A DISCRETE-ELEMENT METHOD OF ANALYSIS FOR ORTHOGONAL SLAB AND GRID BRIDGE FLOOR SYSTEMS

by

John J. Panak
Hudson Matlock

Research Report Number 56-25

Development of Methods for Computer Simulation
of Beam-Columns and Grid-Beam and Slab Systems

Research Project 3-5-63-56

conducted for

The Texas Highway Department

in cooperation with the
U. S. Department of Transportation
Federal Highway Administration

by the

CENTER FOR HIGHWAY RESEARCH
THE UNIVERSITY OF TEXAS AT AUSTIN

May 1972

PREFACE

This report describes a numerical method for the analysis of orthogonal plates with grid-beams. The method is particularly suited for application to highway bridge structures composed of slabs with supporting beam and diaphragm systems. A typical Texas Highway Department bridge is analyzed for various load and stiffness configurations and presented as a series of examples.

The computer program described and included in this report culminates one phase of the effort expended over several years of this research project. The program is compatible with most computers.

This work was supported by the Texas Highway Department in cooperation with the U. S. Department of Transportation Federal Highway Administration, under Research Project 3-5-63-56.

The continued assistance and advice of the project contact representatives and others of the Bridge Division of the Texas Highway Department is deeply appreciated.

John J. Panak
Hudson Matlock

May 1972

LIST OF REPORTS

Report No. 56-1, "A Finite-Element Method of Solution for Linearly Elastic Beam-Columns" by Hudson Matlock and T. Allan Haliburton, presents a solution for beam-columns that is a basic tool in subsequent reports. September 1966.

Report No. 56-2, "A Computer Program to Analyze Bending of Bent Caps" by Hudson Matlock and Wayne B. Ingram, describes the application of the beam-column solution to the particular problem of bridge bent caps. October 1966.

Report No. 56-3, "A Finite-Element Method of Solution for Structural Frames" by Hudson Matlock and Berry Ray Grubbs, describes a solution for frames with no sway. May 1967.

Report No. 56-4, "A Computer Program to Analyze Beam-Columns under Movable Loads" by Hudson Matlcok and Thomas P. Taylor, describes the application of the beam-column solution to problems with any configuration of movable non-dynamic loads. June 1968.

Report No. 56-5, "A Finite-Element Method for Bending Analysis of Layered Structural Systems" by Wayne B. Ingram and Hudson Matlock, describes an alternating-direction iteration method for solving two-dimensional systems of layered grids-over-beams and plates-over-beams. June 1967.

Report No. 56-6, "Discontinuous Orthotropic Plates and Pavement Slabs" by W. Ronald Hudson and Hudson Matlock, describes an alternating-direction iteration method for solving complex two-dimensional plate and slab problems with emphasis on pavement slabs. May 1966.

Report No. 56-7, "A Finite-Element Analysis of Structural Frames" by T. Allan Haliburton and Hudson Matlock, describes a method of analysis for rectangular plane frames with three degrees of freedom at each joint. July 1967.

Report No. 56-8, "A Finite-Element Method for Transverse Vibrations of Beams and Plates" by Harold Salani and Hudson Matlock, describes an implicit procedure for determining the transient and steady-state vibrations of beams and plates, including pavement slabs. June 1968.

Report No. 56-9, "A Direct Computer Solution for Plates and Pavement Slabs" by C. Fred Stelzer, Jr., and W. Ronald Hudson, describes a direct method for solving complex two-dimensional plate and slab problems. October 1967.

Report No. 56-10, "A Finite-Element Method of Analysis for Composite Beams" by Thomas P. Taylor and Hudson Matlock, describes a method of analysis for composite beams with any degree of horizontal shear interaction. January 1968.

Report No. 56-11, "A Discrete-Element Solution of Plates and Pavement Slabs Using a Variable-Increment-Length Model" by Charles M. Pearre, III, and W. Ronald Hudson, presents a method for solving freely discontinuous plates and pavement slabs subjected to a variety of loads. April 1969.

Report No. 56-12, "A Discrete-Element Method of Analysis for Combined Bending and Shear Deformations of a Beam" by David F. Tankersley and William P. Dawkins, presents a method of analysis for the combined effects of bending and shear deformations. December 1969.

Report No. 56-13, "A Discrete-Element Method of Multiple-Loading Analysis for Two-Way Bridge Floor Slabs" by John J. Panak and Hudson Matlock, includes a procedure for analysis of two-way bridge floor slabs continuous over many supports. January 1970.

Report No. 56-14, "A Direct Computer Solution for Plane Frames" by William P. Dawkins and John R. Ruser, Jr., presents a direct method of solution for the computer analysis of plane frame structures. May 1969.

Report No. 56-15, "Experimental Verification of Discrete-Element Solutions for Plates and Slabs" by Sohan L. Agarwal and W. Ronald Hudson, presents a comparison of discrete-element solutions with small-dimension test results for plates and slabs, including some cyclic data. April 1970.

Report No. 56-16, "Experimental Evaluation of Subgrade Modulus and Its Application in Model Slab Studies" by Qaiser S. Siddiqi and W. Ronald Hudson, describes a series of laboratory experiments to evaluate layered foundation coefficients of subgrade reaction for use in the discrete-element method. January 1970.

Report No. 56-17, "Dynamic Analysis of Discrete-Element Plates on Nonlinear Foundations" by Allen E. Kelly and Hudson Matlock, presents a numerical method for the dynamic analysis of plates on nonlinear foundations. July 1970.

Report No. 56-18, "A Discrete-Element Analysis for Anisotropic Skew Plates and Grids" by Mahendrakumar R. Vora and Hudson Matlock, describes a tridirectional model and a computer program for the analysis of anisotropic skew plates or slabs with grid-beams. August 1970.

Report No. 56-19, "An Algebraic Equation Solution Process Formulated in Anticipation of Banded Linear Equations" by Frank L. Endres and Hudson Matlock, describes a system of equation-solving routines that may be applied to a wide variety of problems by using them within appropriate programs. January 1971.

Report No. 56-20, "Finite-Element Method of Analysis for Plane Curved Girders" by William P. Dawkins, presents a method of analysis that may be applied to plane-curved highway bridge girders and other structural members composed of straight and curved sections. June 1971.

Report No. 56-21, "Linearly Elastic Analysis of Plane Frames Subjected to Complex Loading Conditions" by Clifford O. Hays and Hudson Matlock, presents a design-oriented computer solution for plane frames structures and trusses that can analyze with a large number of loading conditions. June 1971.

Report No. 56-22, "Analysis of Bending Stiffness Variation at Cracks in Continuous Pavements," by Adnan Abou-Ayyash and W. Ronald Hudson, describes an evaluation of the effect of transverse cracks on the longitudinal bending rigidity of continuously reinforced concrete pavements. April 1972.

Report No. 56-23, "A Nonlinear Analysis of Statically Loaded Plane Frames Using a Discrete Element Model" by Clifford O. Hays and Hudson Matlock, describes a method of analysis which considers support, material, and geometric nonlinearities for plane frames subjected to complex loads and restraints. May 1972.

Report No. 56-24, "A Discrete-Element Method for Transverse Vibrations of Beam-Columns Resting on Linearly Elastic or Inelastic Supports" by Jack Hsiao-Chieh Chan and Hudson Matlock, presents a new approach to predict the hysteretic behavior of inelastic supports in dynamic problems. June 1972.

(P) indicates Preliminary Report.

ABSTRACT

A method is presented for the discrete-element analysis of isotropic or orthotropic slabs and plates on elastic supports. An integral grid-beam system can also be included which makes it especially useful for highway bridge structures. Other structures such as highway and airfield pavements on elastic foundations, flat or haunched building slabs, two-way mat foundations, stiffened plates, or any grid-type subassemblage of more complex structures can also be investigated.

The developed computer program allows for the free variation of stiffnesses, supports, and loads. In-plane axial thrusts, if their distribution is known, are included and coupled with the stiffnesses of the structure. Inputs to the program may be retained and used for a series of problems for the convenience of the user. Several plotted output options are available including a three-dimensional display of the exaggerated deflected shape of the structure.

Research investigators in other areas have applied preliminary versions of the program to a variety of structures including experimental comparisons and it has been shown to yield satisfactory results.

Four example problems are presented of a typical highway bridge presently in use by the Texas Highway Department. The problems include and discuss most of the brief input calculations and assumptions necessary.

KEY WORDS: bridge decks, orthotropic plates, discrete-element analysis, grid systems, computers.

SUMMARY

This study presents a method for the analysis of highway bridge structures which can be used to supplement and extend existing methods of analysis for the highway bridge investigator. Discrete-element modeling techniques are used to represent the actual structure. The associated computer program with formulations based on the model is used to solve the system of resulting equations by high-speed digital computer. The method can also be directly used for highway pavement analysis in which the pavement is represented as a discretized model on elastic supports representing the subgrade.

Input to the program requires simple engineering judgment in providing appropriate values to represent the stiffnesses, loads, and restraints of the actual structure. Output is arranged in tabular and graphical form to allow ease of interpretation by the engineer-user. Three-dimensional plotted displays of the deformed structure can also be obtained.

This report includes complete documentation for the computer program including

(1) summarized derivation of equations based on the discrete-element model,
(2) computer program listings with brief flow charts, and
(3) detailed program input instructions.

The engineer-user, if he so desires, can apply the program with little study by using the simplified guide for data input and associating his particular structure with the included example problems. The example is a standard Texas Highway Department bridge solved for dead load (non-composite) and three variations of live load (composite).

The computer program allows any number of problems to be run at the same time and utilizes the most recent solution and multiple-loading techniques. Preliminary versions of the program have been used in Research Project 3-5-68-115, "Experimental Verification of Computer Simulation Methods for Slab and Girder Bridge Systems," and the method was shown to give extremely good correlations with experimental data for the various structures considered.

IMPLEMENTATION STATEMENT

This program can be immediately applied by highway bridge and pavement designers to their analysis problems which are now difficult or impossible by conventional methods. It is particularly suited for use on bridge structures since an interconnected grid-beam network is included which can represent beams and diaphragms. Composite action of the beam and slab can be considered by appropriate equivalent beam representations. Extensive experimental comparisons by Research Project 3-5-68-115, "Experimental Verification of Computer Simulation Methods for Slab and Girder Bridge Systems," with versions of this program have been shown to give surprisingly good results for a wide range of bridge types. That project has also shown that even nonlinear or post-cracking behavior under overloads may be predicted by the program by using appropriate elastic estimates of the nonlinear stiffnesses of the structure.

No additional program development is needed to allow immediate use of the program by the sponsors. User-oriented versions of the program could be made which would make application easier for specific classes of problems. These could be data-generation routines for constant thickness slab structures, or automatic computation of effective bending stiffnesses for beam and slab structures.

The availability of this program will make feasible the study of various design options and their overall effect on highway bridge structural capabilities. Among these parameter studies could be the effect of beam spacing as it relates to span-width aspect ratios to better define the present live load distribution coefficients. Diaphragm spacing and positioning could be also studied to aid in defining their effectiveness.

TABLE OF CONTENTS

NOMENCLATURE

Symbol	Typical Units	Definition
a	-	Term in stiffness matrix
[a]	-	Submatrix of stiffness matrix
$\{A\}$	-	Recursion coefficient vector
[b]	-	Submatrix of stiffness matrix
b^1, b^2, b^3	-	Terms in stiffness matrix
[c]	-	Submatrix of stiffness matrix
c^1, c^2, c^3, c^4, c^5	-	Terms in stiffness matrix
[C]	-	Recursion coefficient matrix
C^t	lb-in^2/in/rad	Plate twisting stiffness
[d]	-	Submatrix of stiffness matrix
d^1, d^2, d^3	-	Terms in stiffness matrix
[D]	-	Recursion coefficient multiplier matrix
D^{cr}	lb-in^2/in	Smaller of D^x or D^y at a joint
D^x, D^y	-in^2/in	Plate bending stiffnesses
e		Term in stiffness matrix
[e]	-	Submatrix of stiffness matrix
E	lb/in^2	Modulus of elasticity
[E]	-	Recursion coefficient multiplier matrix
f	lb	Load term

Symbol	Typical Units	Definition
$\{f\}$	-	Load vector
f^s	lb/in^2	Stress in slab
F^x, F^y	$lb\text{-}in^2$	Grid-beam bending stiffnesses
h_x, h_y	-	Discrete-element widths
i	-	Numbering associated with x-direction
j	-	Numbering associated with y-direction
k	lb/in	Subgrade modulus
M^s	in-lb	Slab bending moment
M^x, M^y	in-lb/in	Bending moments in plate
$\overline{M}^x$, $\overline{M}^y$	in-lb	Bending moments in grid-beams
P^x, P^y	lb	Axial thrusts in plate
$\overline{P}^x$, $\overline{P}^y$	lb	Axial thrusts in grid-beams
Q	lb	Applied load
Q^{Rx}, Q^{Ry}	lb	Joint forces from rotational restraints
Q^{Tx}, Q^{Ty}	lb	Joint forces from applied moments
R^x, R^y	lb-in/rad	Rotational restraints
S	lb/in	Support spring
t	in.	Plate or slab thickness
T^x, T^y	lb-in	Applied moments
V^t	lb	Joint force due to twisting
V^x, V^y	lb	Shear forces in plate
$\overline{V}^x$, $\overline{V}^y$	lb	Shear forces in grid-beams

Symbol	Typical Units	Definition
w	in.	Deflection
$\{w\}$	-	Deflection vector
x	-	Coordinate in short direction
y	-	Coordinate in long direction
y^t	in.	Distance from n.a. to top of slab
α	rad	Unit angular mesh rotation
θ^x, θ^y	rad	Mesh angles of rotation
θ^{Jx}, θ^{Jy}	rad	Joint angles of rotation
ν	-	Poisson's ratio
ϕ	rad	Any curvature

CHAPTER 1. INTRODUCTION

This report presents a method for the analysis of isotropic or orthotropic plates on elastic supports which has the added capability of also considering an integral or separate grid system. The major application and emphasis is for use on highway bridge sturctures, but the method can be used for any system which can be represented as a grid or plate.

Problem Definition

Analysis of highway bridge structures has become a much more complex problem in recent years due to the increasing availability of better materials, more complex structural configurations, and the need for greater economy. Presently, most highway bridge structures are one of three distinct types: (1) concrete slab spans, which can be of variable thickness, be post-tensioned, and have various added sections that participate structurally, such as sidewalks and parapet railings; (2) slab and girder arrangements, which consist of steel girders or pre-tensioned concrete girders, with both usually including cross bracing in the form of diaphragms; or (3) a type that might be considered a special case of the second, with the primary difference the use of two-way reinforced decks over widely spaced girders and floor beams. All of these structures have one characteristic in common, the stiffening effect of the basic structure from outstanding members. These outstanding members are the girders on slab and girder-type structures or sidewalks and parapets on slab-type structures. The stiffening contributes significantly to the over-all structural action and must be included for realistic analyses. Thus, a method of analysis which can effectively solve these general structure types would apply to the majority of highway bridge structures and therefore be most useful to the bridge design engineer.

The computer program which is described here is the culmination of several years of development by this research project. It provides an excellent general solution for all of the above structure types, but is not limited to highway bridge structures. Other structures such as highway or airfield pavements on elastic foundations, flat or haunched building slabs, one-way or two-way mat foundations, stiffened steel or aluminum plates, and any plate or grid-type substructure of more complex configurations can also be investigated.

The Analysis Procedure

The basic solution procedure for line members was originally presented in this project for several beam-column applications (Refs 8, 10, 11, and 16). The procedure was extended in various forms and applied to a variety of problems including static, dynamic, and nonlinear solutions of beams and plates (Refs 4, 6, 7, 12, 13, 14, and 15). All of these developments laid the groundwork for the discrete-element solution procedure which is described in Chapter 2.

The mathematical analysis process of the solution has recently been extended to consider all general banded equation systems and at the same time was streamlined for efficient computer operations (Ref 3). The addition of beam-type stiffness members to the basic slab or plate solution was originally presented in this project in 1967 (Ref 6). A beginning direct solution formulation was outlined in 1968 and incorporated in 1969 in a preliminary version of the computer program described in Chapter 3. During the same time, a parallel procedure for skewed, anisotropic plate structures was developed (Ref 18) which also includes the capability of added beam-stiffnesses.

Application to Bridge Structures

Chapter 4 presents example problems involving a typical highway bridge structure to demonstrate use of the computer program. The problems include and discuss most of the basic calculations and assumptions which are made by the engineer in applying the analysis procedure. It is felt that this use of an actual structure expedites effective implementation of the program. The structure is of a type now being used by the Texas Highway Department.

Program Features

This program is the only documented two-dimensional slab or plate program developed on this project, with the exception of the skewed anisotropic program (Ref 18), which stores the image of the read-in data cards for successive use throughout a series of problems, by means of keep options. This data holding feature was first shown to be extremely useful when it was applied to one-dimensional problems in the moving-load beam-column program (Ref 11). The data card images are also proving to be very helpful in making this program useful for special applications in which a user-oriented data-generation routine developed for specific problem types creates card images which are accepted and used by this program to obtain a solution.

Another very useful feature of this program is the ability to selectively profile plot the output values in a variety of ways. In addition, a pseudo three-dimensional display of the deflected shape can also be obtained.

Program Documentation

The computer program is described in Chapter 3 and is documented in Appendix 1. As has been the practice in previous research reports in this project, program flow charts and extensive comment cards in the program listing are included to aid in future developments and modifications by other programmers and researchers.

The program is compatible with IBM 360, UNIVAC 1108, and CDC 6600 computer systems.

CHAPTER 2. THE DISCRETE-ELEMENT MODEL

The Computer Model

The basic computer model for orthotropic plates, which is used to model slabs and bridge decks, is essentially the same as that used in previous developments on this project (Refs 4, 6, 7, 12, 13, 14, and 15). The slab torsional stiffness is represented as a twisting element connected directly to the joints instead of a pair of torsion rods which frame into the bars. The twist reaction forces which result at the joints are the same as in the previous developments, but the model presented in Fig 1 is felt to be simpler to visualize.

Figure 2 presents the discrete-element model of the grid system which acts in conjunction with the plate element shown in Fig 1. Both models are connected at the joints by what may be thought of as ball and socket connections which ensure that the deflection will be the same at the common joint locations.

Figure 3 shows a schematic of the deformed slab model and indicates the general numbering system used.

Equations of Equilibrium

Figures 4 and 5 depict the free-body of a typical joint i, j with all appropriate forces and reactions shown. Forces with a line over the letter (such as $\overline{M}^y_{i,j}$) are related to the grid system. Summation of vertical forces at joint i, j gives

$$Q_{i,j} - V^t_{i,j} + V^t_{i,j+1} + V^t_{i+1,j} - V^t_{i+1,j+1}$$

$$+ V^x_{i,j} - V^x_{i+1,j} + V^y_{i,j} - V^y_{i,j+1}$$

$$+ \overline{V}^x_{i,j} - \overline{V}^x_{i+1,j} + \overline{V}^y_{i,j} - \overline{V}^y_{i,j+1}$$

$$+ Q^{Rx}_{i,j} + Q^{Ry}_{i,j} + Q^{Tx}_{i,j} + Q^{Ty}_{i,j} - S_{i,j}w_{i,j} = 0 \qquad (2.1)$$

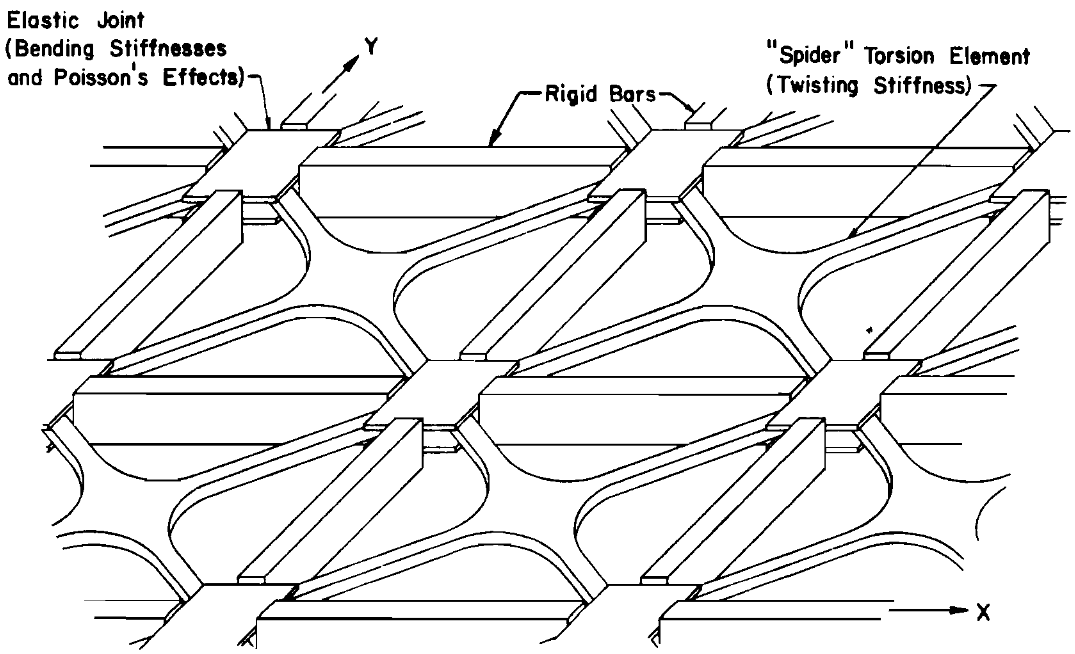


Fig 1. Discrete-element model of slab or plate.

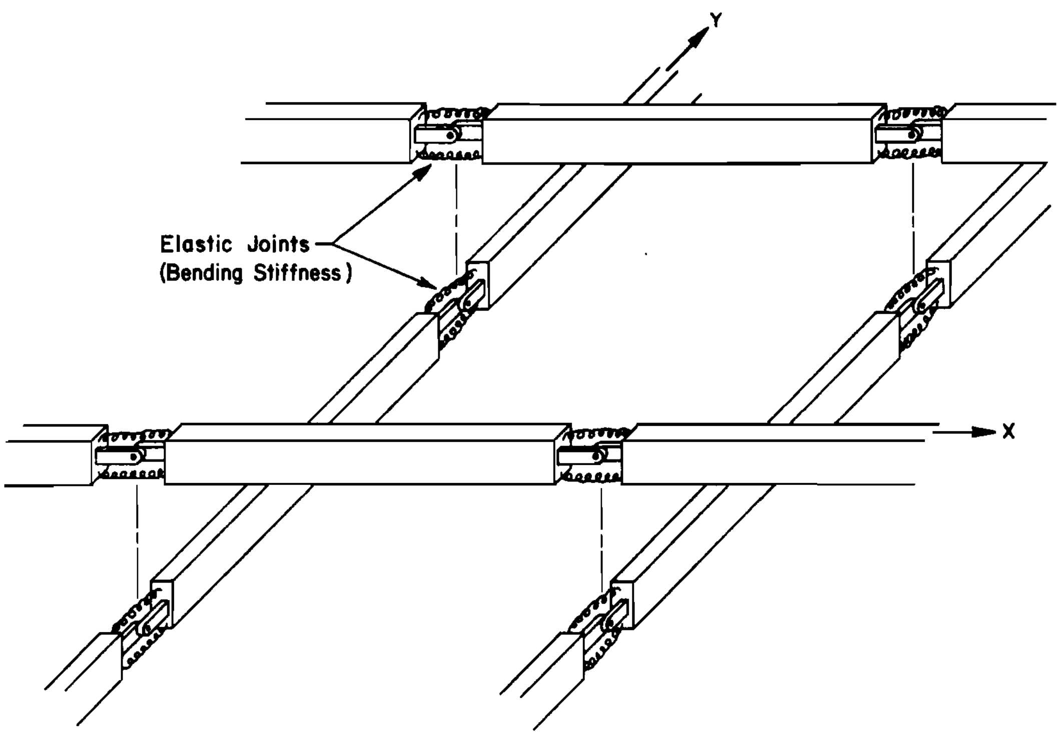


Fig 2. Discrete-element model of grid system.

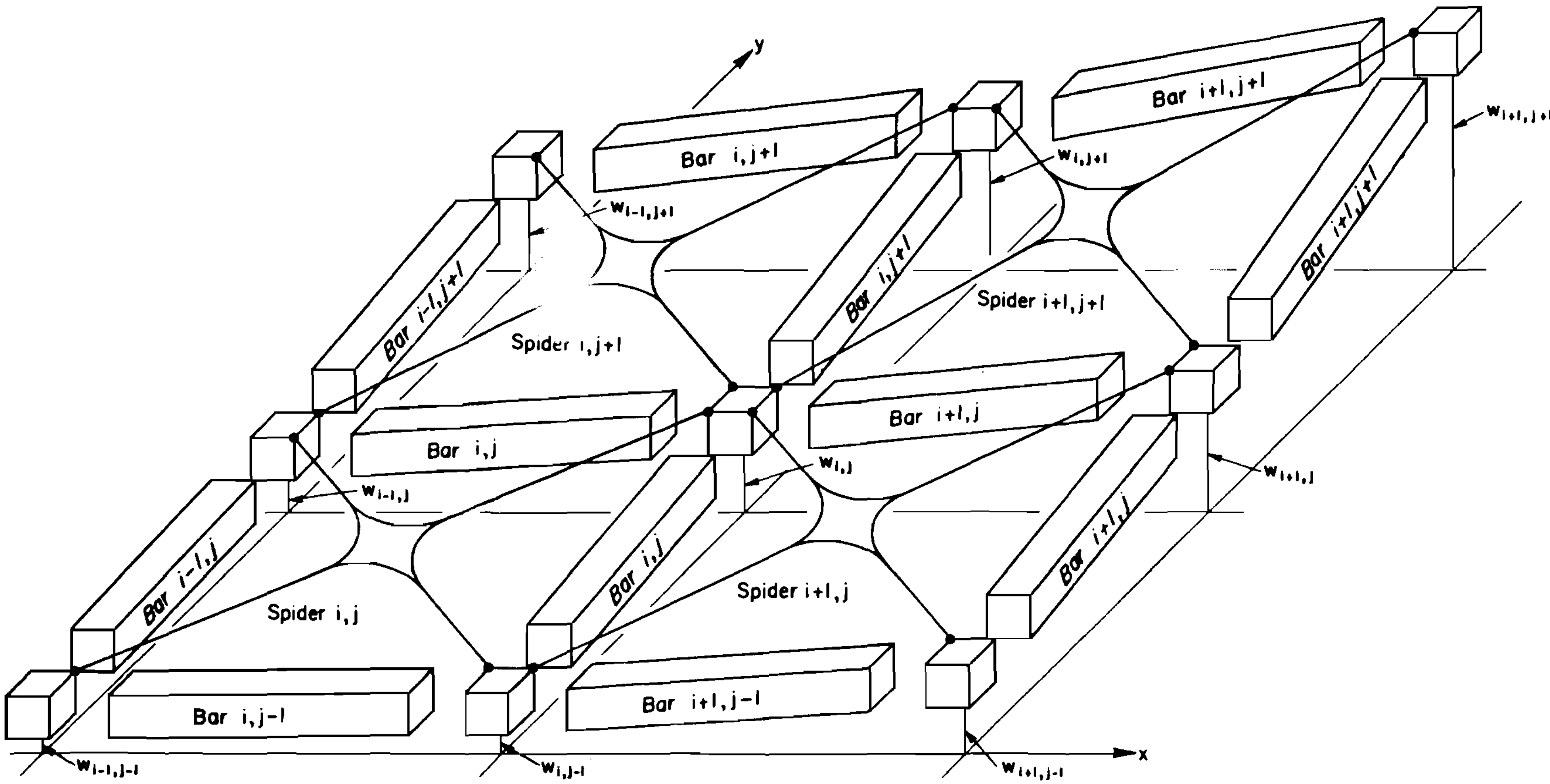


Fig 3. The deformed slab model showing general numbering system.

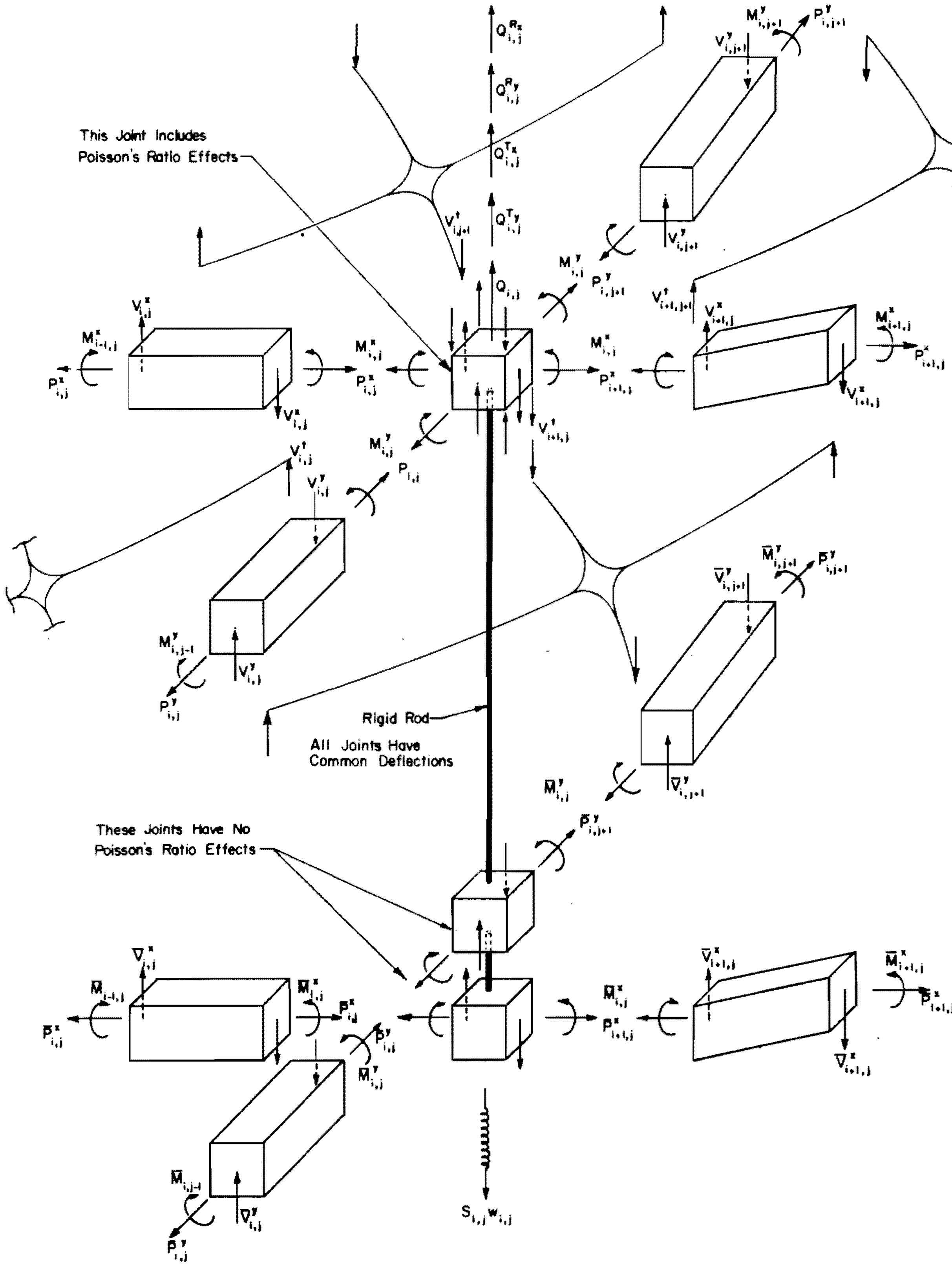


Fig 4. Free-body diagram.

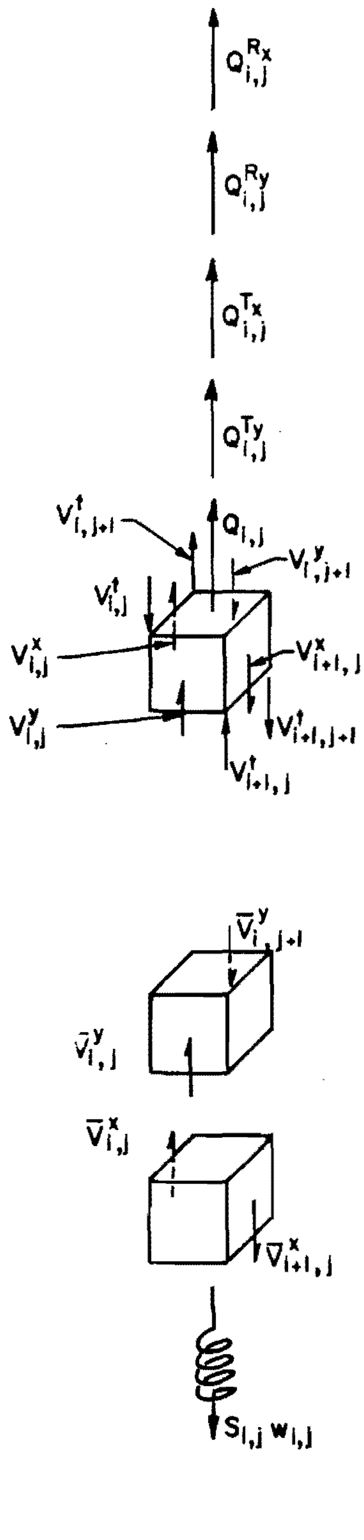


Fig 5. Free body of joint i,j .

The V^t forces are those transferred to joint i,j through the adjacent torsion elements or "spiders" as they are hereafter referred to. A typical spider is shown in Fig 6 for mesh $i+1, j+1$ which is loc... in the area in the increasing x- and y-directions from joint i,j. The spider shown in Fig 7 is deformed in a positive sense; that is, the angle $\theta^x_{i+1,j+1}$ is greater than angle $\theta^x_{i+1,j}$, and angle $\theta^y_{i+1,j+1}$ is greater than angle $\theta^y_{i,j+1}$. The unit angle change $\alpha_{i,j}$ for spider i,j is then seen to be equal to

$$\alpha_{i,j} = \frac{1}{h_y}(\theta^x_{i+1,j+1} - \theta^x_{i+1,j})$$

or

$$\alpha_{i,j} = \frac{1}{h_x}(\theta^y_{i+1,j+1} - \theta^y_{i,j+1}) \tag{2.2}$$

By substituting the appropriate deflection difference relationships for the angles θ, the expression for $\alpha_{i,j}$ in either case is

$$\alpha_{i,j} = \frac{1}{h_x h_y}(w_{i+1,j+1} - w_{i,j+1} - w_{i+1,j} + w_{i,j}) \tag{2.3}$$

The equivalence of the application of a force $V^t_{i,j} = 2\beta$ at the corners of the spider i,j to the application of unit twisting moments β along the edges of a plate element is clearly presented in a previous report (Ref 4). Considering $\beta = C^t_{i,j}\,\alpha_{i,j}$, with $C^t_{i,j}$ being the torsional stiffness per unit of plate width and $\alpha_{i,j}$ the unit angular rotation of mesh i,j, the following expressions are obtained:

$$V^t_{i,j} = \frac{2C^t_{i,j}}{h_x h_y}(w_{i-1,j-1} - w_{i-1,j} - w_{i,j-1} + w_{i,j})$$

$$V^t_{i,j+1} = \frac{2C^t_{i,j+1}}{h_x h_y}(w_{i-1,j} - w_{i-1,j+1} - w_{i,j} + w_{i,j+1})$$

$$V^t_{i+1,j} = \frac{2C^t_{i+1,j}}{h_x h_y}(w_{i,j-1} - w_{i,j} - w_{i+1,j-1} + w_{i+1,j})$$

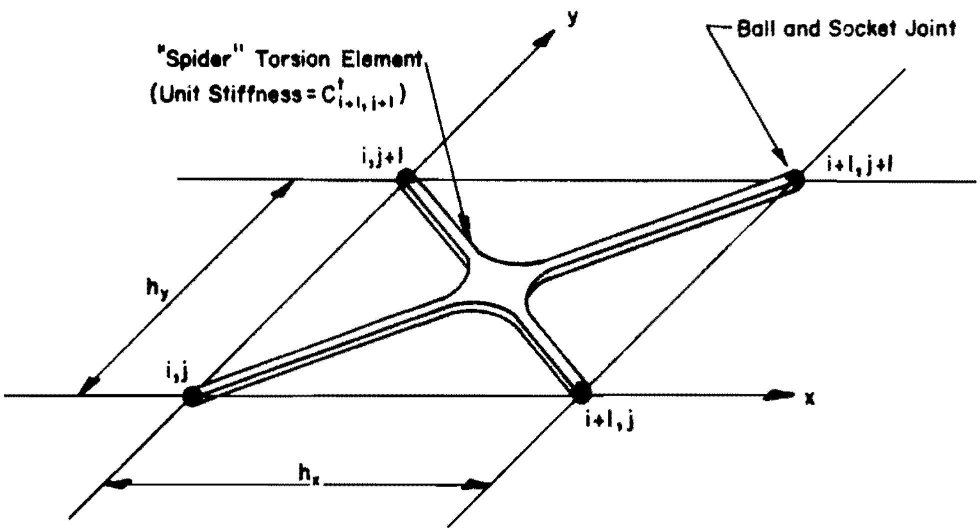


Fig 6. Torsional stiffness model, mesh i+1,j+1 .

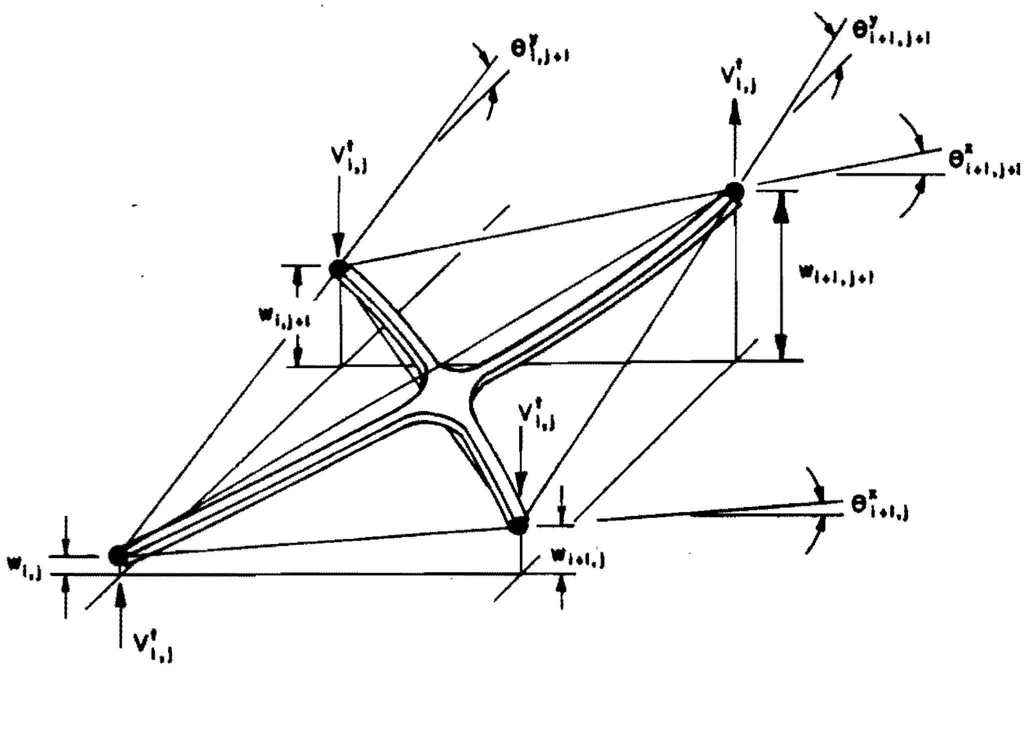


Fig 7. Free body of deformed spider element.

$$V^t_{i+1,j+1} = \frac{2C^t_{i+1,j+1}}{h_x h_y} (w_{i,j} - w_{i,j+1} - w_{i+1,j} + w_{i+1,j+1}) \tag{2.4}$$

Now, considering the bending moments and axial thrusts acting in the slab or plate and summing moments about each bar, the following expressions are obtained for the vertical shear forces.

$$V^x_{i,j} = \frac{1}{h_x} \left[-M^x_{i-1,j} + M^x_{i,j} - P^x_{i,j} (-w_{i-1,j} + w_{i,j}) \right]$$

$$V^x_{i+1,j} = \frac{1}{h_x} \left[-M^x_{i,j} + M^x_{i+1,j} - P^x_{i+1,j} (-w_{i,j} + w_{i+1,j}) \right]$$

$$V^y_{i,j} = \frac{1}{h_y} \left[-M^y_{i,j-1} + M^y_{i,j} - P^y_{i,j} (-w_{i,j-1} + w_{i,j}) \right]$$

$$V^y_{i,j+1} = \frac{1}{h_y} \left[-M^y_{i,j} + M^y_{i,j+1} - P^y_{i,j+1} (-w_{i,j} + w_{i,j+1}) \right] \tag{2.5}$$

Similarly, summation of moments about each bar of the supporting gridwork gives identical equations for the beam shear forces.

$$\overline{V}^x_{i,j} = \frac{1}{h_x} \left[-\overline{M}^x_{i-1,j} + \overline{M}^x_{i,j} - \overline{P}^x_{i,j} (-w_{i-1,j} + w_{i,j}) \right]$$

$$\overline{V}^x_{i+1,j} = \frac{1}{h_x} \left[-\overline{M}^x_{i,j} + \overline{M}^x_{i+1,j} - \overline{P}^x_{i+1,j} (-w_{i,j} + w_{i+1,j}) \right]$$

$$\overline{V}^y_{i,j} = \frac{1}{h_y} \left[-\overline{M}^y_{i,j-1} + \overline{M}^y_{i,j} - \overline{P}^y_{i,j} (-w_{i,j-1} + w_{i,j}) \right]$$

$$\overline{V}^y_{i,j+1} = \frac{1}{h_y} \left[-\overline{M}^y_{i,j} + \overline{M}^y_{i,j+1} - \overline{P}^y_{i,j+1} (-w_{i,j} + w_{i,j+1}) \right] \tag{2.6}$$

The axial thrusts P^x , P^y , $\overline{P}^x$, and $\overline{P}^y$ are the concentrated values of axial thrust that act in the bars of either the slab system or the supporting grid system. It will be noted that since the complete assembly has a common deflection reference plane, the axial thrusts in the slab or gridwork have exactly the same effect. They are input as separate parameters for simplicity and also to allow for independent solution of either a grid or slab.

The bending moments in Eqs 2.5 and 2.6 may be expressed in a central difference form. A complete derivation of the slab bending relationships is presented in a previous report (Ref 4).

$$M^x_{i,j} = D^x_{i,j}\, h_y \left[\frac{w_{i-1,j} - 2w_{i,j} + w_{i+1,j}}{h_x^2}\right]$$

$$+ D^{cr}_{i,j}\, h_y\, \nu \left[\frac{w_{i,j-1} - 2w_{i,j} + w_{i,j+1}}{h_y^2}\right]$$

$$M^y_{i,j} = D^y_{i,j}\, h_x \left[\frac{w_{i,j-1} - 2w_{i,j} + w_{i,j+1}}{h_y^2}\right]$$

$$+ D^{cr}_{i,j}\, h_x\, \nu \left[\frac{w_{i-1,j} - 2w_{i,j} + w_{i+1,j}}{h_x^2}\right] \qquad (2.7)$$

The plate bending stiffnesses $D^x_{i,j}$ and $D^y_{i,j}$ are defined per unit of plate width. $D^{cr}_{i,j}$ is the smaller of either $D^x_{i,j}$ or $D^y_{i,j}$ so that the Poisson's ratio effect is felt only in the parent plate of a stiffened plate area (Ref 12).

$$\overline{M}^x_{i,j} = F^x_{i,j} \left[\frac{w_{i-1,j} - 2w_{i,j} + w_{i+1,j}}{h_x^2}\right]$$

$$\overline{M}^y_{i,j} = F^y_{i,j} \left[\frac{w_{i,j-1} - 2w_{i,j} + w_{i,j+1}}{h_y^2}\right] \qquad (2.8)$$

The beam bending stiffnesses $F^x_{i,j}$ and $F^y_{i,j}$ are the concentrated values of beam bending stiffness and are exactly equivalent to the stiffness terms used in previous beam-column programs (Refs 8, 10, and 11).

Expressions similar to Eqs 2.7 and 2.8 for $M^x_{i-1,j}$, $M^x_{i+1,j}$, $M^y_{i,j-1}$, $M^y_{i,j+1}$, $\overline{M}^x_{i-1,j}$, $\overline{M}^x_{i+1,j}$, $\overline{M}^y_{i,j-1}$, and $\overline{M}^y_{i,j+1}$ are obtained by cyclic permutation of the subscripts i and j.

$Q^{Rx}_{i,j}$, $Q^{Ry}_{i,j}$, $Q^{Tx}_{i,j}$, and $Q^{Ty}_{i,j}$ are the transverse loads at joint i,j due to rotational restraints and applied moments. It should be noted that a rotational restraint or applied moment contributes to the transverse load or restraint one station from where they are placed, as is also true in previous beam-column work (Refs 8 and 11).

$$Q^{Rx}_{i,j} = \frac{1}{2h_x}\left[-R^x_{i-1,j}\ \theta^{Jx}_{i-1,j} + R^x_{i+1,j}\ \theta^{Jx}_{i+1,j}\right]$$

$$Q^{Ry}_{i,j} = \frac{1}{2h_y}\left[-R^y_{i,j-1}\ \theta^{Jy}_{i,j-1} + R^y_{i,j+1}\ \theta^{Jy}_{i,j+1}\right] \qquad (2.9)$$

$$Q^{Tx}_{i,j} = \frac{1}{2h_x}\left[-T^x_{i-1,j} + T^x_{i+1,j}\right]$$

$$Q^{Ty}_{i,j} = \frac{1}{2h_y}\left[-T^y_{i,j-1} + T^y_{i,j+1}\right] \qquad (2.10)$$

where,

$$\theta^{Jx}_{i-1,j} = \frac{1}{2h_x}\left(-w_{i-2,j} + w_{i,j}\right)$$

$$\theta^{Jx}_{i+1,j} = \frac{1}{2h_x}\left(-w_{i,j} + w_{i+2,j}\right)$$

$$\theta^{Jy}_{i,j-1} = \frac{1}{2h_y}\left(-w_{i,j-2} + w_{i,j}\right)$$

$$\theta^{Jy}_{i,j+1} = \frac{1}{2h_y}\left(-w_{i,j} + w_{i,j+2}\right) \qquad (2.11)$$

are the slopes at respective joints and are the average of the slopes in the adjacent bars, which was defined as the central difference slope in the moving-load beam-column report (Ref 11).

Substituting Eqs 2.9 through 2.11 into Eq 2.1, rearranging, and collecting the terms associated with each unknown deflection, gives

$$a_{i,j}\, w_{i,j-2} + b^1_{i,j}\, w_{i-1,j-1} + b^2_{i,j}\, w_{i,j-1} + b^3_{i,j}\, w_{i+1,j-1}$$

$$+ c^1_{i,j}\, w_{i-2,j} + c^2_{i,j}\, w_{i-1,j} + c^3_{i,j}\, w_{i,j} + c^4_{i,j}\, w_{i+1,j} + c^5_{i,j}\, w_{i+2,j}$$

$$+ d^1_{i,j}\, w_{i-1,j+1} + d^2_{i,j}\, w_{i,j+1} + d^3_{i,j}\, w_{i+1,j+1} + e_{i,j}\, w_{i,j+2}$$

$$= f_{i,j} \tag{2.12}$$

where

$$a_{i,j} = \frac{1}{h_y^3}\left[h_x\, D^y_{i,j-1} + F^y_{i,j-1}\right] - \frac{1}{4h_y^2} R^y_{i,j-1}$$

$$b^1_{i,j} = \frac{1}{h_x h_y}\left[\nu(D^{cr}_{i-1,j} + D^{cr}_{i,j-1}) + 2C^t_{i,j}\right]$$

$$b^2_{i,j} = -\frac{2}{h_y^3}\left[h_x(D^y_{i,j-1} + D^y_{i,j}) + F^y_{i,j-1} + F^y_{i,j}\right]$$

$$- \frac{2}{h_x h_y}\left[\nu(D^{cr}_{i,j-1} + D^{cr}_{i,j}) + C^t_{i,j} + C^t_{i+1,j}\right]$$

$$- \frac{1}{h_y}\left[P^y_{i,j} + \overline{P}^y_{i,j}\right]$$

$$b^3_{i,j} = \frac{1}{h_x h_y}\left[\nu(D^{cr}_{i,j-1} + D^{cr}_{i+1,j}) + 2C^t_{i+1,j}\right]$$

$$c^1_{i,j} = \frac{1}{h_x^3}\left[h_y\, D^x_{i-1,j} + F^x_{i-1,j}\right] - \frac{1}{4h_x^2} R^x_{i-1,j}$$

$$c^2_{i,j} = -\frac{2}{h_x^3}\left[h_y(D^x_{i-1,j} + D^x_{i,j}) + F^x_{i-1,j} + F^x_{i,j}\right]$$

$$- \frac{2}{h_x h_y}\left[\nu(D^{cr}_{i-1,j} + D^{cr}_{i,j}) + C^t_{i,j} + C^t_{i,j+1}\right]$$

$$- \frac{1}{h_x}\left[P^x_{i,j} + \overline{P}^x_{i,j}\right]$$

$$c^3_{i,j} = \frac{1}{h_x^3}\left[h_y(D^x_{i-1,j} + 4D^x_{i,j} + D^x_{i+1,j}) + F^x_{i-1,j} + 4F^x_{i,j} + F^x_{i+1,j}\right]$$

$$+ \frac{1}{h_y^3}\left[h_x(D^y_{i,j-1} + 4D^y_{i,j} + D^y_{i,j+1}) + F^y_{i,j-1} + 4F^y_{i,j} + F^y_{i,j+1}\right]$$

$$+ \frac{2}{h_x h_y}\left[4\,\nu D^{cr}_{i,j} + C^t_{i,j} + C^t_{i,j+1} + C^t_{i+1,j} + C^t_{i+1,j+1}\right]$$

$$+ \frac{1}{h_x}\left[P^x_{i,j} + P^x_{i+1,j} + \overline{P}^x_{i,j} + \overline{P}^x_{i+1,j}\right]$$

$$+ \frac{1}{h_y}\left[P^y_{i,j} + P^y_{i,j+1} + \overline{P}^y_{i,j} + \overline{P}^y_{i,j+1}\right]$$

$$+ \frac{1}{4h_x^2}\left[R^x_{i-1,j} + R^x_{i+1,j}\right] + \frac{1}{4h_y^2}\left[R^y_{i,j-1} + R^y_{i,j+1}\right] + S_{i,j}$$

$$c^4_{i,j} = - \frac{2}{h_x^3}\left[h_y(D^x_{i,j} + D^x_{i+1,j}) + F^x_{i,j} + F^x_{i+1,j}\right]$$

$$- \frac{2}{h_x h_y}\left[\nu(D^{cr}_{i,j} + D^{cr}_{i+1,j}) + C^t_{i+1,j} + C^t_{i+1,j+1}\right]$$

$$- \frac{1}{h_x}\left[P^x_{i+1,j} + \overline{P}^x_{i+1,j}\right]$$

$$c^5_{i,j} = \frac{1}{h_x^3}\left[h_y\, D^x_{i+1,j} + F^x_{i+1,j}\right] - \frac{1}{4h_x^2} R^x_{i+1,j}$$

$$d^1_{i,j} = \frac{1}{h_x h_y}\left[\nu(D^{cr}_{i-1,j} + D^{cr}_{i,j+1}) + 2C^t_{i,j+1}\right]$$

$$d^2_{i,j} = - \frac{2}{h_y^3}\left[h_x(D^y_{i,j} + D^y_{i,j+1}) + F^y_{i,j} + F^y_{i,j+1}\right]$$

$$- \frac{2}{h_x h_y} \left[\nu (D^{cr}_{i,j} + D^{cr}_{i,j+1}) + C^{t}_{i,j+1} + C^{t}_{i+1,j+1} \right]$$

$$- \frac{1}{h_y} \left[P^{y}_{i,j+1} + \bar{P}^{y}_{i,j+1} \right]$$

$$d^3_{i,j} = \frac{1}{h_x h_y} \left[\nu (D^{cr}_{i,j+1} + D^{cr}_{i+1,j}) + 2C^{t}_{i+1,j+1} \right]$$

$$e_{i,j} = \frac{1}{h_y^3} \left[h_x \, D^{y}_{i,j+1} + F^{y}_{i,j+1} \right] - \frac{1}{4h_y^2} R^{y}_{i,j+1} \qquad (2.13)$$

and

$$f_{i,j} = Q_{i,j} + \frac{1}{2h_x} \left[-T^{x}_{i-1,j} + T^{x}_{i+1,j} \right] + \frac{1}{2h_y} \left[-T^{y}_{i,j-1} + T^{y}_{i,j+1} \right] \qquad (2.14)$$

The Combined Stiffness Matrix

The thirteen coefficients (Eqs 2.13) of the unknown deflections form a pattern which composes the customary thirteen point operator for orthogonal plate systems. If there are no twisting stiffness or Poisson's ratio effects, the coefficients b^1 , b^3 , d^1 , and d^3 would be eliminated, thus degenerating the operator to nine elements which is appropriate for grid systems.

The ordered system of equations forms a diagonally banded and symmetrical stiffness matrix as shown in Fig 8 which is taken from Ref 12. The coefficient matrix is partitioned into banded submatrices where each horizontal partition represents the equilibrium equations for joints along a constant j-level.

Recursion-Inversion Solution Procedure

A recursion-inversion method developed and presented in previous reports is used to solve the system of equations for unknown displacements. An analogous procedure is the recursive process described for solution of beam-columns (Ref 8). The similarity between the recursive equations developed for slabs and grids and those derived for the recursive solution of beam-columns has been shown (Refs 12 and 15). A restatement of the recursion-inversion method is included here.

GENERAL EQUATION:

$$a_{i,j}w_{i,j-2} + b^1_{i,j}w_{i-1,j-1} + b^2_{i,j}w_{i,j-1} + b^3_{i,j}w_{i+1,j-1} + c^1_{i,j}w_{i-2,j} + c^2_{i,j}w_{i-1,j} + c^3_{i,j}w_{i,j} + c^4_{i,j}w_{i+1,j} + c^5_{i,j}w_{i+2,j} + d^1_{i,j}w_{i-1,j+1} + d^2_{i,j}w_{i,j+1} + d^3_{i,j}w_{i+1,j+1} + e_{i,j}w_{i,j+2} = f_{i,j}$$

OR IN MATRIX FORM:

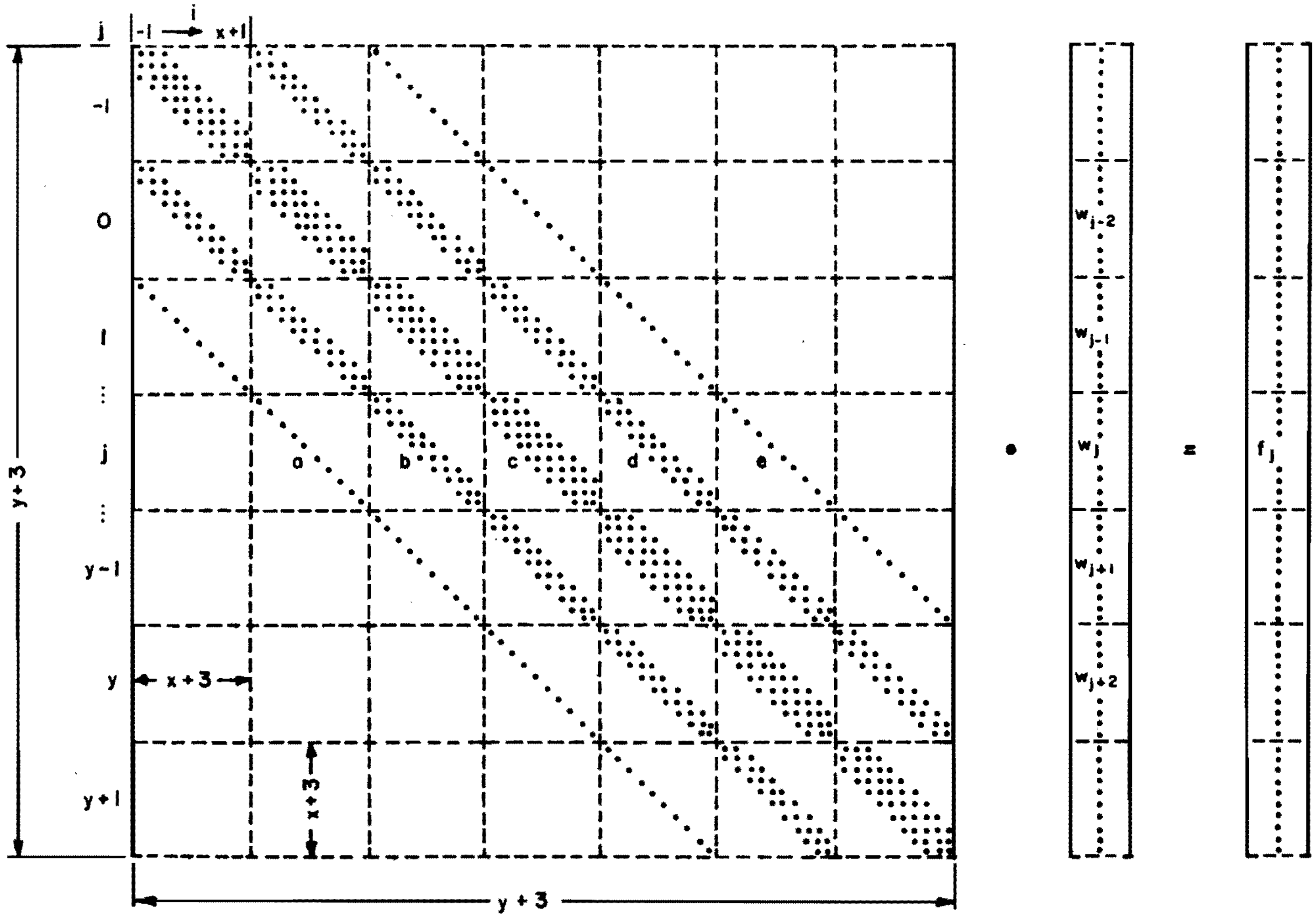


Fig 8. Form of the equations showing partitioned stiffness matrix (Ref 12).

Considering the horizontal partition at j of the equations of equilibrium shown in Fig 8,

$$\left[a_j\right]\left\{w_{j-2}\right\} + \left[b_j\right]\left\{w_{j-1}\right\} + \left[c_j\right]\left\{w_j\right\}$$

$$+ \left[d_j\right]\left\{w_{j+1}\right\} + \left[e_j\right]\left\{w_{j+2}\right\} = \left\{f_j\right\} \tag{2.15}$$

the recursive equations may be written as

$$\left\{A_j\right\} = \left[D_j\right]\Big[\left[E_j\right]\left\{A_{j-1}\right\} + \left[a_j\right]\left\{A_{j-2}\right\} - \left\{f_j\right\}\Big]$$

$$\left[B_j\right] = \left[D_j\right]\Big[\left[E_j\right]\left[C_{j-1}\right] + \left[d_j\right]\Big]$$

$$\left[C_j\right] = \left[D_j\right]\left[e_j\right] \tag{2.16}$$

where

$$\left[D_j\right] = -\Big[\left[a_j\right]\left[C_{j-2}\right] + \left[E_j\right]\left[B_{j-1}\right] + \left[c_j\right]\Big]^{-1}$$

$$\left[E_j\right] = \left[a_j\right]\left[B_{j-2}\right] + \left[b_j\right] \tag{2.17}$$

and the back-substitution equation as

$$\left\{ w_j \right\} = \left\{ A_j \right\} + \left[\quad B_j \quad \right] \left\{ w_{j+1} \right\} + \left[\quad C_j \quad \right] \left\{ w_{j+2} \right\} \tag{2.18}$$

The above matrix equations are seen to be identical in form to those previously given for a beam-column solution (Ref 8). The long, or y, direction of a slab and grid or plate problem determines the number of partitions of the stiffness matrix and is directly analogous to the number of [illegible]ions in a beam-column solution.

For a symmetric stiffness matrix, it has been shown that the above set of recursive equations can be further modified (Ref 3):

$$\left\{ A_j \right\} = \left[\quad D_j \quad \right] \left[\left[\quad E_j \quad \right] \left\{ A_{j-1} \right\} + \left[\diagdown e_{j-2} \diagdown \right]^t \left\{ A_{j-2} \right\} - \left\{ f_j \right\} \right]$$

$$\left[\quad B_j \quad \right] = \left[\quad D_j \quad \right] \left[\quad E_{j+1} \quad \right]^t$$

$$\left[\quad C_j \quad \right] = \left[\quad D_j \quad \right] \left[\diagdown e_j \diagdown \right] \tag{2.19}$$

where

$$\left[\quad D_j \quad \right] = - \left[\left[\diagdown e_{j-2} \diagdown \right]^t \left[\quad C_{j-2} \quad \right] + \left[\quad E_j \quad \right] \left[\quad B_{j-1} \quad \right] + \left[\diagdown\diagdown c_j \diagdown\diagdown \right] \right]^{-1}$$

$$\left[\quad E_j \quad \right] = \left[\diagdown e_{j-2} \diagdown \right]^t \left[\quad B_{j-2} \quad \right] + \left[\diagdown d_{j-1} \diagdown \right]^t \tag{2.20}$$

and

$$\left[E_{j+1} \right] = \left[\diagdown e_{j-1} \diagdown \right]^t \left[B_{j-1} \right] + \left[\diagdown d_j \diagdown \right]^t \tag{2.21}$$

The superscript t indicates the transpose of the appropriate submatrix. Since the stiffness matrix for the discrete element model is a symmetric one, Eqs 2.19 through 2.21 are used with Eq 2.18 in the solution procedure.

Multiple-Loading Technique

The multiple-loading recursive technique presented in the two-way bridge floor slab report (Ref 12) is briefly reviewed here. It is noted that the load vector $\left\{ f_j \right\}$ appears in only the $\left\{ A_j \right\}$ recursion coefficient vector of Eq 2.19. Since $\left\{ A_j \right\}$ is not involved in either the remaining coefficient or multiplier matrices, the $\left[E_j \right]$, $\left[D_j \right]$, $\left[C_j \right]$, and $\left[B_j \right]$ coefficients are computed once and retained on disk, tape, or other auxiliary storage files and then recalled as needed for each successive problem. The $\left\{ A_j \right\}$ recursion vector is simply modified or reformed for each successive loading.

CHAPTER 3. THE COMPUTER PROGRAM

The computer program described in this report applies solution techniques developed during this project study in such fashion as to solve the numerical equations in the most efficient and direct manner. The basic solution procedure is directly analogous to that originally presented (Ref 8) and later extended to two-dimensional systems (Ref 15). The efficiency of multiple-loadings (Ref 12) has been included. A recently developed streamlined and efficient recursion-inversion procedure as applied to banded systems (Ref 3) is also used.

Data Card Images

A significant feature of the program is the technique through which all data input in a series of problems are retained and echo-printed for all problems. This procedure was first applied and found useful for the moving-load beam-column program (Ref 11). The procedure avoids the necessity of recoding and including common data for a problem series. In previous two-dimensional programs written on this project, computer storage for the complete slab was set aside for each different type of input stiffness and load. This program simply stores the input data card images which are searched at each level in the solution process and only the necessary stiffness and load terms are generated. Thus, a significant saving in required computer storage is achieved for a slight increase in computational effort. The technique of storing data card images has also been shown to provide the most convenient method of transferring information from a data-generation routine written for certain specific problem types. The data card images are created by the data-generation routines and then the basic solution program utilizing the general slab and grid system is used for the solution of the structure. By this means, any number of special purpose programs can use this one basic orthogonal system computer solution technique.

The FORTRAN Program

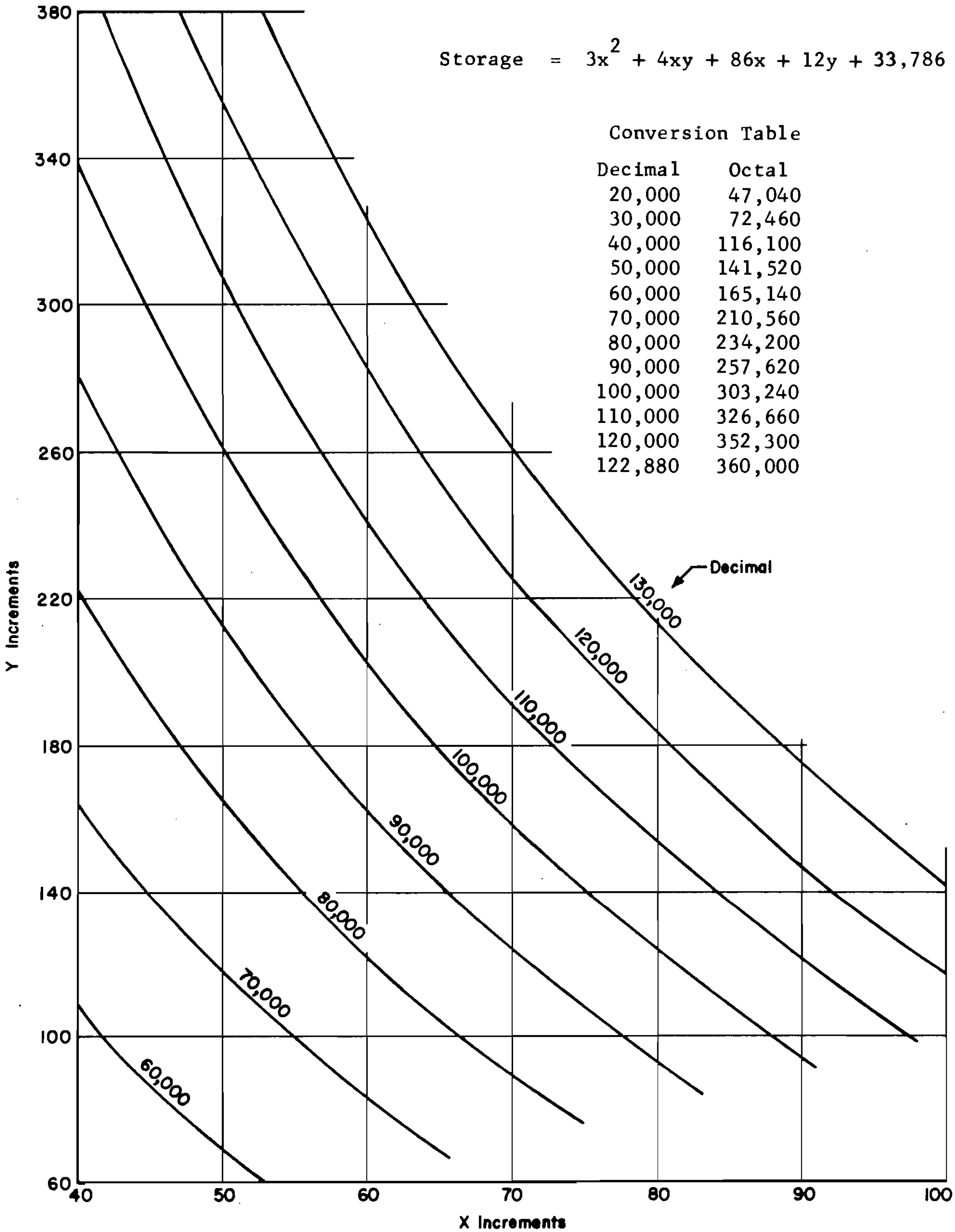
The version of the orthogonal slab and grid program described here is called SLAB 49, which indicates that it is the 49th in the sequence of developments for plate and grid-type structures. The program is written in FORTRAN and follows the basic guidelines given for ASA FORTRAN (Ref 2). The program is written for both CDC 6600 and IBM 360 computers and is easily made compatible with other similar computer systems.

The program listing presented in Appendix 3 is specifically for the IBM 360 computer. Those cards needed to operate on the CDC 6600 computer are included as companion cards and are nulled with a C in column one and have the symbols CDC in columns 78 through 80, following the card preparation date. All necessary cards for IBM operation such as the selective double precision statements are tagged with IBM in columns 78 through 80. Not all variables need to be double precisioned to yield correct solutions with the IBM 360 computer, which saves a significant amount of storage over the customary procedure of complete double precisioning. When converting from one computer system to another, it is recommended that the companion IBM cards be retained and nulled with an added C in column one at the same time the C is omitted from the CDC cards.

Storage Requirements

The storage requirements are variable, depending upon the size of the problem to be run. Cards which must be changed for different sized problems are specified at the beginning of the program and include only the dimension statements and two variables which define the number of increments in both directions. The largest dimensioned storage arrays are designated as R1 , R2 , R3 , PW , PBMX , PBMY , and PSIGO (Refer to the program listing, Appendix 3). The first three are doubly subscripted, with both subscripts equal to the number of increments plus 3 in the shorter slab or grid direction, and are required for the basic solution routine (Ref 3). The last four variables above are doubly subscripted, with subscripts equal to the number of increments plus 3 in both slab directions, and these variables are concerned with plotting of the output values of deflection, bending moments, and principal moments. A simplified version of the program without plots could be easily made which would then allow the length of the problem to be almost without limit as far as computer storage is concerned. Computer time considerations limit all large problems to those solutions that are economically justified.

A plot of the CDC 6600 storage requirements is shown in Fig 9. If it is necessary to gain more storage space, certain variables could be set equal



Storage = $3x^2 + 4xy + 86x + 12y + 33{,}786$

Conversion Table

Decimal	Octal
20,000	47,040
30,000	72,460
40,000	116,100
50,000	141,520
60,000	165,140
70,000	210,560
80,000	234,200
90,000	257,620
100,000	303,240
110,000	326,660
120,000	352,300
122,880	360,000

Fig 9. SLAB 49 storage requirements.

to each other in storage by an equivalence declaration. Layering is another technique to reduce storage requirements that could be used at a slight increase in computational time. These procedures have not been presented with the included program to avoid initial confusion when converting to other computer systems.

A summary flow diagram of the program is shown in Fig 10. A list of the variables used and their definitions is given in the Notation in Appendix 2. A complete listing of the program, including brief flowcharts of each routine, is shown in Appendix 3. All subroutines in the program are variably dimensioned as functions of the short X and long Y lengths, which are specified in the main driving program.

Data Input Tables

The general procedures to be followed for input of a problem are outlined in the Guide for Data Input, Appendix 1. The guide is designed so that additional copies may be made and used for routine reference. A parallel study of the guide will help the reader to understand the following discussion.

The first two cards of a problem series are for identification purposes. Any desired alphanumeric descriptive information can be entered by the user. It is suggested that the date of the run, the user's name, and the chosen units always be included on these two cards. The next card is the problem number card with a brief description of the particular problem. The problem number itself may contain alphabetical characters if desired. The problem series terminates when a blank problem number is encountered. Any number of problems may be run at the same time.

Table 1 is used to input the problem control data and is always comprised of two data cards that include the keep options, multiple load option, number of cards input for this problem, and other ouput options including plots.

The first card of Table 1 contains the keep options. Any data of the preceding problem may be retained if desired. If Table 2 is retained, it may not be added to or modified. Any of the other data tables may be retained and additional data cards may be input, up to the combined maximum total cards for each table. The Multiple Load Option in column 50 of the first Table 1 card is left blank if each successive problem is independent of the preceding problem. If a following problem is for the same slab-and-grid system, and if

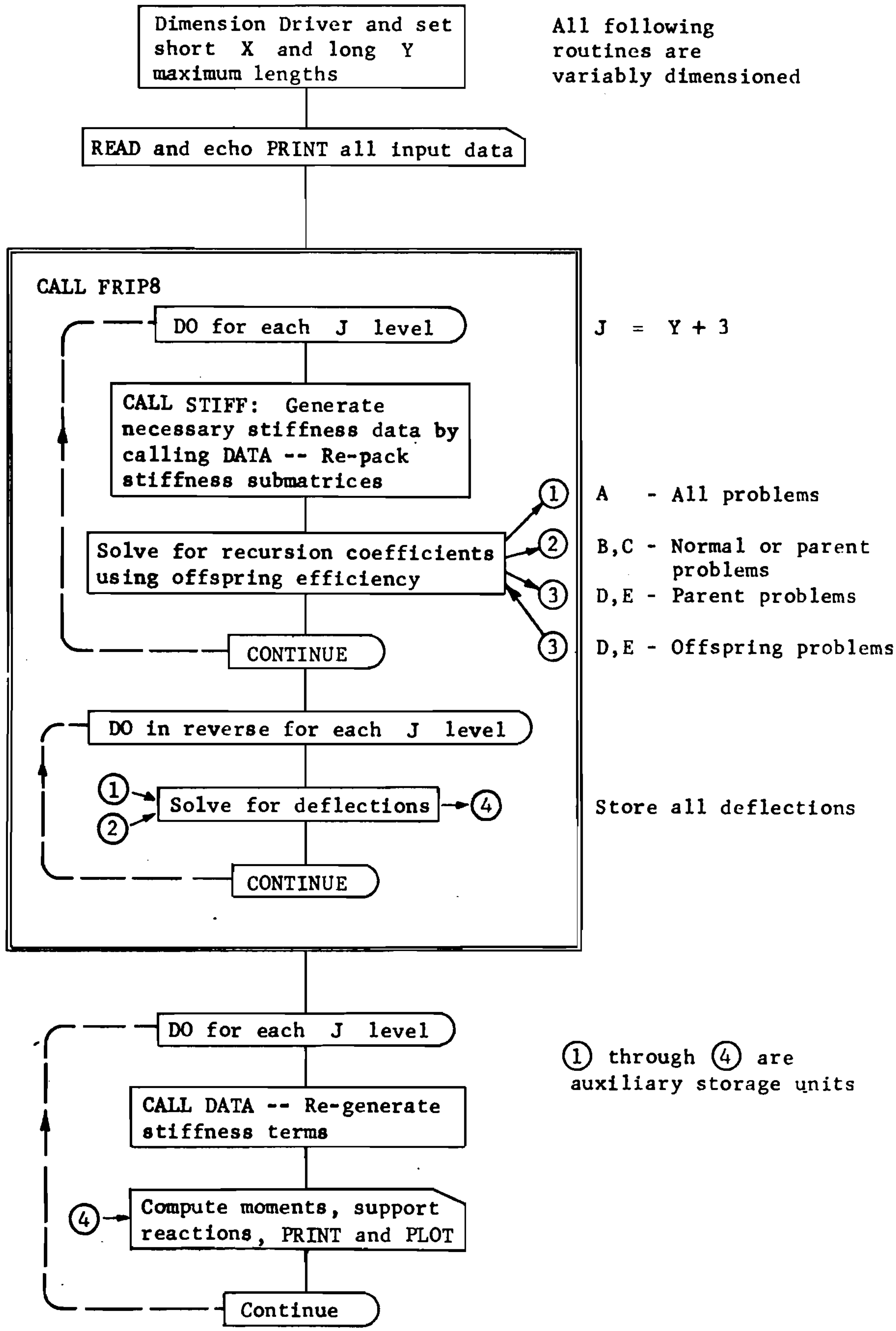


Fig 10. Summary flow diagram for SLAB 49.

only the load pattern and placement given by Table 7 are to change, the first problem in the loading series is specified with a +1 for the option. This will be the "parent" problem. Each successive loading must have a -1 for the option and it will be termed an "offspring" problem. When a blank option or another +1 is encountered, that problem is another independent problem or a new parent problem.

The second card of Table 1 is for the designation of the number of data cards to be input in this problem for Tables 2 through 9. The user should carefully check the card counts to avoid data errors. A number of common types of data errors are checked for by the input routines, but it is possible to create a false problem by incorrect card counts.

The four options in columns 50 through 65 of the second card of Table 1 are for output options. The statics check option may be exercised by entering a 1 in column 50. This is useful for determining if a solution inaccuracy exists, especially for computer systems which have required double precision operations. The statics check is computed internally in the program by reapplication of the governing equations (Eqs 2.12 through 2.14) to the computed values of deflection. Any applied loads and external couples are deducted and the remaining quantity represents the computational error at each joint of the system. This check depends on there being no error in the data or stiffness generation portion of the program since the same data and stiffnesses are used for the back-substitution. The statics check option will normally be left blank, in which case the output value printed will be the concentrated support reaction at each joint. The second output option, in column 55, is exercised by entering a 1 if the user desires that the computed value of principal moment in the slab be converted to an equivalent value of stress. The computed value of stress is correct only for slab areas of uniform thickness with no discontinuities. If this stress option is exercised, an appropriate value of slab thickness must then be available in Table 2. The principal moment is converted to a stress having the same sign by multiplication of the moment by the plate section modulus, which is internally computed from the Table 2 input value of thickness. Axial thrusts if present are not included as part of the stress calculation. The third output option, in column 60, is for control of the type of plotted output associated with the specified areas in Table 8. If the column is left blank or zero, a crude printer plot is obtained along with tabulated output designated by Table 8. If set equal to 1, no tabulated output is obtained for the areas of Table 8 but the plots of those areas are on microfilm (assuming it is available on the particular computer system). If this option is set equal to 2, then a combination of 0 and 1 is obtained; that is, Table 8 areas are tabulated, a printer plot display is made, and a microfilm plot is also made. If the option is set equal to 3, only a line plot on paper is obtained for the Table 8 areas and no tabulation is printed. The fourth option, in column 65, is exercised by entering a 1 to create a pseudo-three-dimensional plotted display of all the computed deflections for the entire slab or grid. This display is discussed further under Computed Results.

Table 2 is used to specify the constants for the problem. These are the number of increments in the X and Y directions, the increment lengths in both directions, and Poisson's ratio. For efficient solution of the program, the number of Y increments must be equal to or greater than the number of X increments. Table 2 must be kept for offspring problems since the constants must be the same as in the parent problem. The constants specified for the parent are retained and used by all successive offspring problems. Table 2 also provides a space for entry of slab or plate pseudo-thickness. The thickness must be entered if the stress option of Table 1 is exercised. The use of the thickness is appropriate only for plates of a constant thickness. For most two-way concrete slabs, the principal moment is a more important design quantity which is automatically obtained if the stress option in Table 1 is left blank. At specified discontinuities in the slab, such as a crack or joint which might be modeled by means of a reduced bending stiffness (Ref 9), the output value of "stress" at that location may be misleading. A better estimate of stress at discontinuities may be obtained by inspecting the variation in computed stress at several stations adjacent to the discontinuity.

Table 3 has the number of cards specified in Table 1. Card counts should be carefully checked. It is recommended that a listing of the data cards be checked by the user prior to submission of the program for a run.

Joint stiffness and load data are entered by a coordinate system notation. The coordinates refer to the discrete-element model numbering system. A joint is defined as occurring at the intersection of the station lines in each X and Y direction. A mesh is defined as that area surrounded by four joints. A bar is defined as the discrete-element length between adjacent joints. Figure 3 in Chapter 2 and Fig A1 of Appendix 1 summarize this notation. Note that mesh data cannot have either a zero X or zero Y coordinate; X-bar data cannot have a zero X-coordinate, and, similarly, the Y-bar cannot have a zero Y-coordinate. If the data occur only at one location (such as a concentrated load), the From and Through coordinates are simply repeated. If the data occur along a line, the coordinates will reflect this by having either both X or both Y coördinates the same. Data distributed over a rectangular area are specified by entering the lower left and upper right coordinates bounding the area.

The orthogonal slab bending stiffnesses D^x and D^y are entered in each direction and are specified on a per unit width basis. If the edge of the slab coincides with a station line, a half-value of stiffness should be input for both D^x and D^y along the edge. If the edge of the real slab is not on a station line, a proportionate value of full stiffness is entered. This is demonstrated by a sample input in Appendix 1, Fig A2. The stiffness apportionment may be thought of as a direct function of the plan area of real slab surrounding each joint.

The orthogonal slab stiffnesses may be varied by the user to provide for any degree of flexibility in both directions. For instance, a crack in a slab can be represented by an appropriate reduced value of stiffness at that location (Ref 9). The customary stiffness relationship that may be used for isotropic plates or slabs is given here for reference. Poisson's ratio for concrete is a difficult constant to define precisely, but a value of 0.15 is used by most investigators.

$$D^x = D^y = \frac{Et^3}{12(1 - \nu^2)} \tag{3.1}$$

The beam bending stiffnesses F^x and F^y are entered in each direction and are the total concentrated values of stiffness which represent beams or combination of beams and a composite slab. The input of beam stiffnesses is more completely discussed in the example problems of Chapter 4. The beam stiffness characterizations and inputs are identical with the models of previous beam-column developments (Refs 8, 10, 11, and 16).

Load Q is concentrated on a per joint basis and may be apportioned at each joint by the contributory area loaded around each joint. Positive loads act upward. Loads that occur between joints may be fractionally proportioned to the adjacent joints. Support springs S are concentrated values input and apportioned exactly like loads. A rigid support may be specified by introducing a large value for the support spring. A maximum value of 1×10^{25} is suggested to avoid computational difficulties for some computers. A uniform subgrade of modulus k can be specified by application of this relationship

$$S = kh_x h_y \tag{3.2}$$

Multi-valued subgrade moduli or foundations with voids may also be modeled by entering the appropriate spring stiffnesses.

Table 4 is for input of rotational restraints and applied couples. These are input as concentrated effects in either the X or Y direction. An effective rotational restraint could be computed for a slab supported by a column which is framed into the slab, or for a girder which is framed into its support members. Both restraints and applied moments are used and input the same as in previous beam-column programs (Refs 8, 10, 11, and 16).

Table 5 is for input of the twisting stiffness associated with the slab portion of the structure. Since it is a quantity that represents stiffness between joints of the model (Fig 1) it is input in a separate table from the bending stiffnesses constants.

The twisting stiffness C^t is input on a per unit width basis for each mesh surrounded by four joints. When the geometric edges of the actual slab do not fall on a station line, proportionate values of unit twisting stiffness may be input similar to bending stiffness apportionment. Computations of twisting stiffnesses for slabs or plates are at best still approximate procedures. This is due to uncertainty in the defining of the shearing modulus of rigidity. One procedure is to ascertain the twisting stiffness experimentally as outlined by Hudson (Ref 4). The formula shown below for reference is correct for uniformly thick isotropic plates.

$$C^t = \frac{Et^3}{12(1 + \nu)} \tag{3.3}$$

An approximate value for twisting stiffness for orthotropic slabs or stiffened plates may be obtained by using procedures outlined by Huffington (Ref 5) or computations summarized by Troitsky (Ref 17). Fortunately, precise values of twisting stiffness are unnecessary to model a slab. The main load-carrying capabilities of a slab are due to its bending stiffness which is easier to define.

Table 6 is for input of the axial thrusts in either the slab or supporting grid-beam network. All the thrusts are concentrated values, and therefore any distributed axial thrusts in slabs must be concentrated over the appropriate increment width. As previously stated, since the slab and grid systems have a common deflection reference, the effect of slab or beam axial thrusts is the same. The provision for inputting them separately is made so that they are easier to visualize and also to allow the independent solution of either grid-beam assemblages or slabs. In either case, the output is shortened, as discussed later.

The beam and slab axial thrusts (Fig 4) have exactly the same effect as their counterparts in previous beam-column solutions (Refs 11 and 16).

Axial thrusts in bridge decks might be due to differential temperature between a slab and floor system, closing of an expansion joint, or traffic braking and acceleration forces. There is no provision in the program for automatic distribution of applied axial forces since no in-plane supports are used which would restrain them. The user must specify the distribution of the axial tensions (+) and compressions (-) in each X-bar and Y-bar of the model (Figs 1 and 4). Since these are bar forces, no data should be input which would represent forces outside the boundaries of the actual slab. A brief sample of data input is given in Fig A2 in Appendix 1.

Table 7 is used for input of loads if they are to change position or magnitude for two or more problems on the same structure. Its use is strictly for convenience. Loads appearing in Table 7 could also have been input as Table 3 loads, and that is why the data field for the load value is in columns 61 through 70 for both Tables 3 and 7. For offspring problems (multiple load option equal to -1), the loads must be input in Table 7, and loads or stiffnesses input in Table 3 are retained. For normal problems or parent problems (multiple load option equal to 0 or +1), loads may be input in either Table 3 or 7.

Table 8 is used to define the lines or areas of selected tabulated and plotted output for deflection, bending moments in the X and Y directions, and either the maximum principal moment or stress, depending on the stress option in Table 1. Thus, concise printout for a specific location, such as near wheel loads and support points, can be obtained. The number of cards is as specified in Table 1 and may include up to a maximum of 10 cards. Each card can encompass up to a maximum of 300 points; for instance, coordinates from 11,11 through 20,40 or from 0,0 through 11,24 could be specified. If a larger area is required, another card covering the adjacent area can be added.

The major advantage of Table 8 is that a crude printer plot display can be obtained for each area specified, if the option in Table 1 is 0 or 2 . This feature is discussed more completely under Computed Results. Table 8 is especially useful when the deflection and moment variations along a line or over a local area are being studied. Table 8 can be omitted, if desired, since all selected output values appear in the complete printout of results, but caution should be used; if all or part of the complete printout of results is suppressed in Table 9, discussed below, a significant amount of computer time will be used but no printout of results will be made. The type of printer or line plotted output depends on the plot control option in Table 1, as discussed above.

Table 9 allows the user to have only selected sections of the complete output printed. This is sometimes very desirable for a problem series in which local areas of the structure are under study for various positions of loads, supports, discontinuities, etc. When Table 9 is omitted the complete output is printed. A partial output is printed when the sections to be printed are specified within the Y-bounds designated on the Table 9 cards. Up to 10 different Y-bounded sections can be printed. The sections may overlap; the doubly defined areas are printed only once. This option is useful for study in areas near span centerlines and in areas near supports for continuous structures; the printout can be deleted for the noncritical areas. Caution is again advised, to assure that output which might be of interest is not suppressed.

General Data Input Comments

As previously stated, it is wise to obtain a listing of the data input for verification prior to program submission, especially for large and time-consuming problems with complex data input.

All data in Tables 3, 4, 5, 6, and 7 are algebraically accumulated to storage as needed and values therefore may be added or subtracted regardless of other values input or held from previous problems.

For offspring problems, only Tables 1 and 7 are required, but Tables 8 and 9 may be again specified if different output areas are desired. When solving an offspring problem, the user must hold Tables 2, 3, 4, 5, and 6, adding no cards. As discussed in Chapter 2, the multiple-loading recursive technique offers a considerable computer time advantage for a problem series in which only the load magnitude and position change. If the user adds any data to that in Tables 3 through 6, the problem must then be considered another parent or independent problem.

Data Errors

All data are checked for compatibility with the geometry of the specified slab and consistency of coordinate input. A count is made of the number of data errors in each table and the problem is then terminated with a message showing the number of data errors. Typical errors are (1) misusing the multiple-load option, such as inputting a -1 to follow a 0 in the preceding problem; (2) having the number of increments in the X-direction exceed those in the Y-direction, which would result in an inefficient and time consuming computer solution; (3) specifying a negative or zero increment length; (4) inputting a negative Poisson's ratio or thickness; (5) making the Through X or Y coordinate in a data specification numerically less than the From coordinate; (6) specifying data outside the geometric limits of the slab; (7) specifying a zero X or Y coordinate for a twisting stiffness; (8) using a zero X coordinate for X-bar axial thrusts or a zero Y coordinate for Y-bar thrusts; (9) specifying a number of increments greater than the dimensioned storage with which the program can operate; and (10) misusing the selected output option.

Computed Results

The computed results are headed by a line which includes the program title, specific version, and latest program revision date. Immediately below are the two alphanumeric information data header cards, followed by the problem number and description. The importance of using run dates and descriptive alphanumeric information for the problem series header cards and problem number cards in order to avoid confusion when running a large number of problems cannot be overemphasized.

Data cards for each table, including those kept from previous problems, are echo printed with explanatory headings exactly as they were input. It is good practice to recheck all data for possible errors prior to inspection of the results.

Tabulated Results. The computed results are listed in Y-station groups, in reverse order because of the computation arrangement set up in the program and for convenience in correlating of results with the sketch of the problem input. The output is arranged to give the X and Y-joint coordinate, the transverse deflection at each joint (upward deflections are positive), the slab and beam bending moments, the slab twisting moment, the principal slab moment or stress and its direction, and either the concentrated value of support reaction or the statics check. Output values of slab moments are per unit width; beam moments are total per beam. Bending moments and stress are positive for compression in the top of the beams or slab. The X-bending moments act in the X-direction and the Y-bending moments in the Y-direction.

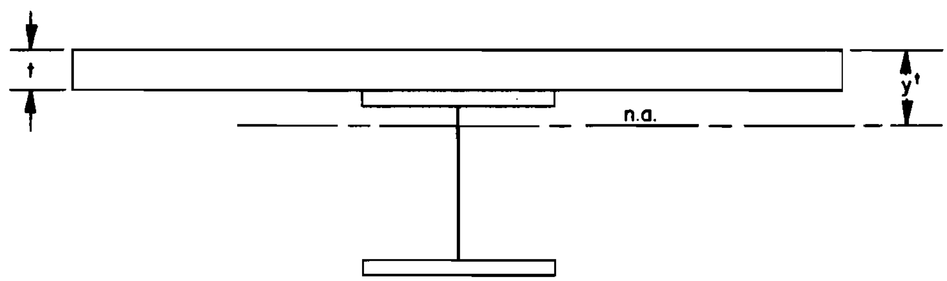
The output for the slabs is automatically given in a reduced form if the input data did not include any beam stiffnesses. The reverse is true if no slab stiffness data were input. The output is arranged so that the X and Y-beam moments are printed directly below each value of slab moment when both slab and beam data are present.

Twisting Moments. The per unit width X-twisting moments are tabulated and are exactly equal to the Y-twisting moments with opposite sign. The X-twisting moments act in the X-direction and are about the Y-axis. Even though the input values of twisting stiffnesses were specified for each mesh, the output values of twisting moment are the average of four adjacent mesh areas and are given at the joints. The user is cautioned that the output values of twisting moment along the edges or other discontinuities of a slab or plate reflect the average and may be a one-quarter, one-half, or some other proportionate value. The output values of largest principal moment or stress at edges are also affected by this averaging.

Principal Moments or Stresses. A Mohr's circle analysis is made at each joint, using the orthogonal slab bending moments and twisting moments to yield the larger numeric value (positive or negative) of principal moment per unit width and the angle from the X-axis of the coordinate system to the acting direction of this larger value. Counterclockwise angles are positive. The principal moment values are converted to the larger numerical value of principal stress if the stress option was specified in Table 1 and a thickness is present in Table 2. A positive stress indicates tension in the bottom of the slab, which follows the same sign convention as do the bending moments. The stress option is properly used only for slabs or plates of constant thickness. A direct conversion can be made for principal stress from the principal moment for plates of variable stiffness and thickness. The output value of stress does not include any in-plane forces that may be present. The user must consider axial thrusts (tension or compression) input in Table 6 when interpreting stress results.

Stresses in Composite Slabs. Figure 11 presents a typical situation when interpreting combined beam and slab results. The input value of beam stiffness was computed from the complete composite section, using appropriate transformed areas, effective flange widths, etc., but excluding the stiffness of the slab about its own axis. The input value of slab stiffness was computed on the basis of the slab thickness and estimated elastic constants for the concrete. The output value of maximum slab stress, which was internally computed, was probably not in a major orthogonal direction due to the presence of loads between beams. Assuming it was nearly orthogonal, it is desired to compute an estimate of the slab stress in the direction of the beam from the results of the computer model. First, the bending stress in the slab in the beam direction is determined; the output value is taken or the stress is computed from the slab bending moment and slab section modulus in which M^s is the slab bending moment in the beam direction. This stress is shown as f^s in Fig 11. Second, this stress is modified by the ratio of twice the computed depth to the neutral axis of the composite cross section divided by the slab thickness, also shown in Fig 11. This total slab stress could then be further modified for added stresses due to axial thrusts. The output values of transverse slab bending moment and the slab twisting moment can then be converted to stress and a Mohr's circle analysis made to determine the maximum slab stress and its direction. All of these calculations could have been made in the computer, but that would require the input of estimated depths to the composite neutral axis at all locations in the slab-beam system or an even more complex input, with all areas, moduli, etc., at all points. A version of the program developed specifically for special

Composite Cross Section:



Stresses in Transformed Section:

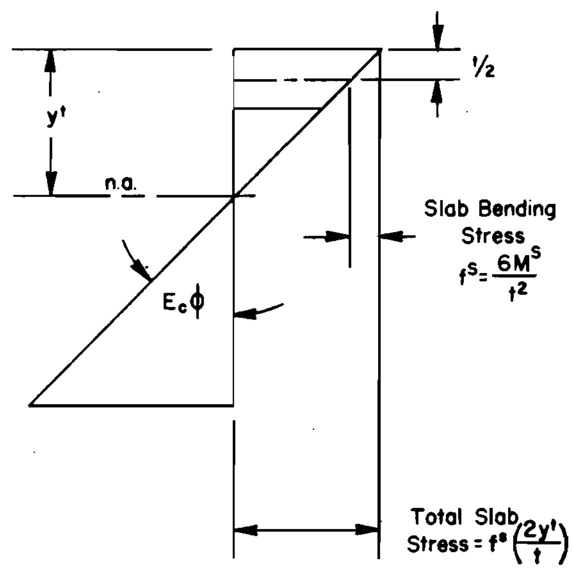


Fig 11. Stress in composite slab.

applications would be best for an automatic stress calculation. However, when local slab stresses have been computed, it has been found that they usually occur in directions which are transverse to the relatively stiff beams. That is, the beams act as almost rigid supports to the slab (Ref 12), and the calculations as shown in Fig 11 are not normally required.

Moments. The values of beam moment in the printout are concentrated values rather than distributed, as are those for slab moments. Slab moments are typically in kip-feet per foot, or in units consistent with the input; and beam moments are in kip-feet. The user can obtain predicted beam stresses by applying usual section modulus values consistent with the input stiffnesses.

Reactions. The last column of output lists the support reactions at each station if the statics check option in Table 1 is blank. The support reaction is the concentrated value of resistance to displacement offered by any support springs that are present. A subgrade modulus spring will reflect the concentrated value of pressure under the slab. If the spring is specified with a large value to represent a rigid support (which is unrealistic, since all structural supports have some degree of elastic response), then the value printed is the rigid support reaction.

A statics check is printed instead of support reaction if the Column 50 statics check option in Table 1 is exercised. This statics check is the summation of all the computed shears, twisting moment forces, restraint and applied moment forces, subgrade reaction, and applied external load at each joint, as shown in Fig 5. The value printed represents the amount of error at that joint which is inherent in the computer solution. This option has no practical application, but if it is suspected that there are computer inaccuracies which are being generated by roundoff, truncation, or errors, this option will help to determine their magnitude.

As a check on the back-substitution process in the computer solution, and as a check on the total load input to the grid-slab structure, a final result is printed at the end of the detailed output. This is the algebraic sum of all the reaction values and should be equal to the sum of all the applied loads. This check should always be inspected to verify that the desired load system was specified and that the problem was properly solved. Another value is printed following this, the maximum statics check error and the station at which it occurred. This value is always printed, whether the statics check option in Table 1 is exercised or not, and can act as an immediate flag to the user if some error has occurred in the computer. The error would normally be expected to be less than about 10 orders of magnitude smaller than the largest load applied. The user should not concern himself with the significant digits of any remnant (approximately zero) values of output.

Profile Output. After the detailed output, areas of selected profile output designated by Table 8 are printed. No tabulated values will be printed if the plot option in Column 60 of Table 1 was 1 or 3. Profile tabulations are obtained if the option was 0 or 2. These tabulations are printed in consecutive groups associated with the largest number of increments designated in the Table 8 rectangular area. For instance, if the area desired is from 10,16 to 12,20 then there would be three X-groups, each with five values. If the area is square, the groups would be for consecutive Y values, which is the same arrangement as the normal output discussed above. Adjacent to the coordinates is the numerical value of the deflection, moment, or stress. Printed to the right of the output values is a series of asterisks whose placement relative to one another is based on the numerical values. Thus, a crude plot of the output values is obtained. The plot has a width of 20 printer characters and a length equal to the number of nodes encompassed by the area specified. The plot has no scale and no zero; the values are relative to one another, increasing positively to the right. The 20-character width is based on the minimum and maximum values in the area to be plotted. The user is cautioned not to misinterpret apparent changes in plot curvature which might be due to very slight numerical changes.

The printer plots have been found to be especially valuable because the user obtains them with the rest of his printed output; unnecessary time is not spent waiting for line plotter output or in hand plotting. They are also useful in understanding slab behavior for areas adjacent to concentrated wheel loads and supports. Deflection areas are printed and plotted first, followed by bending moment areas, which are tabulated and plotted adjacent to each other with a set of common coordinates if both X and Y-moments desired were in the same area. The final selected profile output is for the principal moments or stresses, again depending on the Table 1 option. Plots of principal moments or stresses along slab lines are somewhat misleading since the direction of the moment or stress usually varies along the line. They are valuable, however, in pointing out maximum values which might be overlooked when inspecting a mass of numbers in the normal detailed output.

The final printed output is the computer time used for the problem and the total accumulated time for the problem series. The user should record run times for parent and offspring problems for each problem size run on his computer system to estimate run times required for future problems. For small problems, the offspring times will be from 20 to 50 percent of the parent problem times. Fortunately, the offspring problem time decreases to a very small proportion of the parent problem time as the problem size becomes large. A time as low as 4 percent is possible.

Three-dimensional deflection plots are obtainable with the program if the appropriate plotter routines and hardware are available. The subroutines which generate the three-dimensional plots have been written using standard routines available for Calcomp plot systems. Each plot is arranged to fit within a 7-inch by 10-inch area. The receding angle of the plot measured from the X-axis has a tangent of 0.9 (approximately a 42 degree angle). The receding Y-axis lengths are reduced by a factor of 0.75 of their true length relative to the X-axis lengths. The vertical deflections are distorted (magnified) so that the maximum deflection is 1 inch on the plot. These automatic values for these three parameters were fixed to yield what has been found to be a reasonably optimum implied position of the viewer's eye for most problems. These values can be changed if necessary since they are variables in the subroutines that generate the three-dimensional deflection plots. A typical plot is shown in Fig 12, which is taken from one of the problems in Chapter 4. The problem number is plotted to the left of the plot, for reference. It should be noted that on the plot some areas are hidden due to the magnification of the vertical deflections. The small tics superimposed around the edges of the plot are of the undeformed structure and thus comparisons of relative deflections can be made.

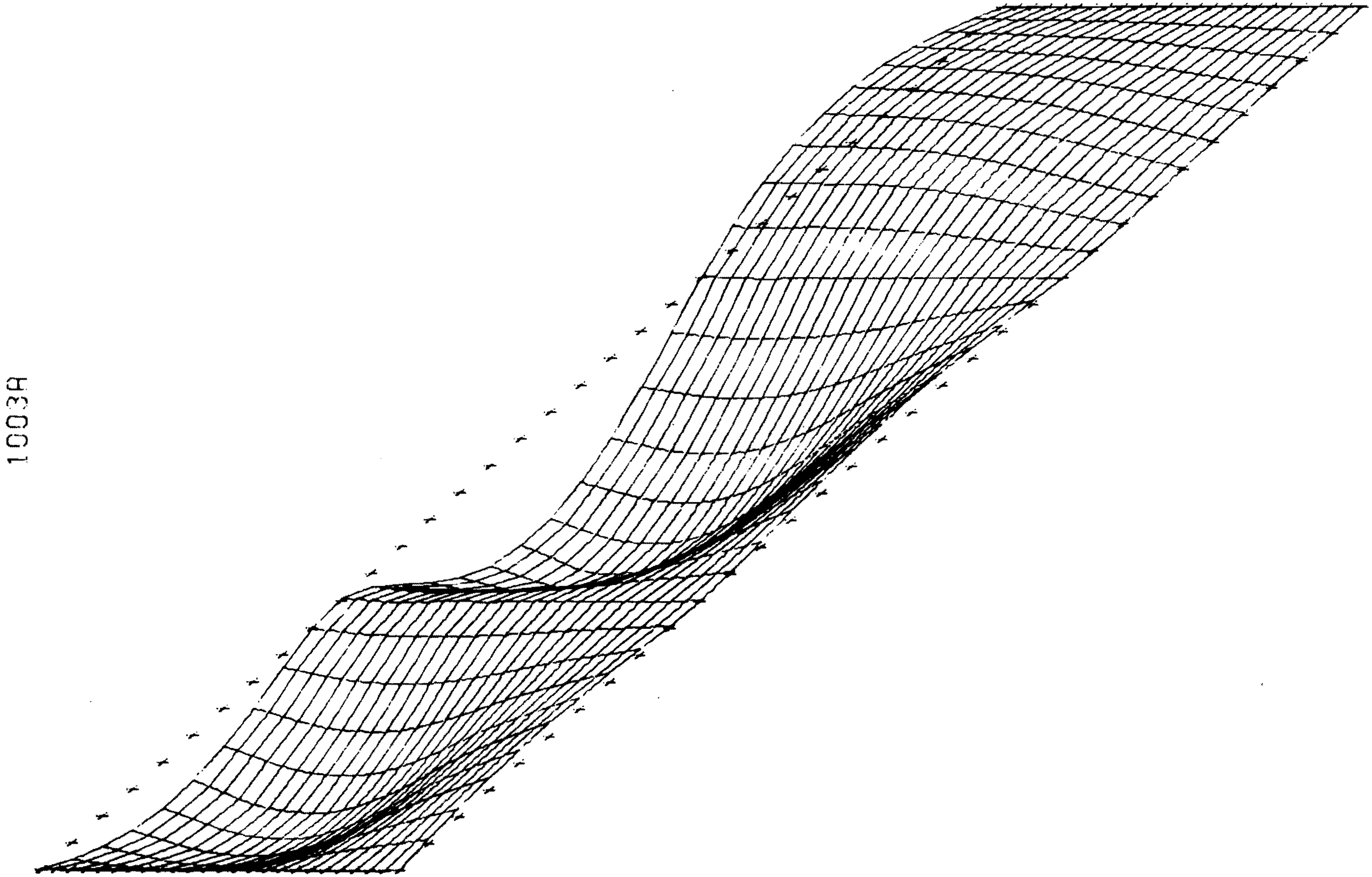


Fig 12. Three-dimensional plot of deflections, Example Problem 1003.

CHAPTER 4. EXAMPLE PROBLEMS

Application of the SLAB 49 program to actual bridge structures is demonstrated by a series of problems which present some of the steps required for the analysis of a typical highway bridge structure. The structure is similar to those currently being designed by the Texas Highway Department and is shown in Fig 13. It consists of a concrete deck resting on a system of longitudinal main beams which are continuous over two intermediate supports with transverse diaphragms framing between the beams.

The structural system is analyzed for

(1) the dead load of beams and concrete deck (PROB 1001).

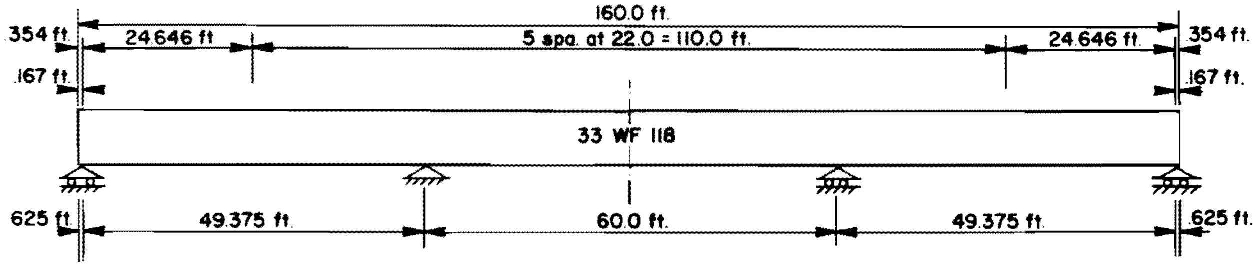
(2) the maximum positive moment in the center span of the main beams due to HS20 truck loading (PROB 1002).

(3) the maximum negative moment due to HS20 lane loading (PROB 1003).

(4) the maximum negative moment in an interior beam due to two HS20 truck loadings in adjacent lanes of two spans (PROB 1004).

The structure geometry as given in Fig 13 must be slightly modified for application of the SLAB 49 program. Slight changes in dimensions are required so the actual geometry may be fitted to suitable increment lengths in both directions. The number of increments and increment lengths in the transverse X-direction is primarily governed by the spacing of the main beams. The 7.25-foot beam spacing is subdivided into five equal increment lengths h_x of 1.45 feet. The concrete deck is extended two increments (2.9 feet) transversely over the outer beams whereas the actual deck overhang is 3.125 feet. The effect of the 0.225-foot excess can be included by using proportionate stiffness and load input data for the exterior stations. If an integral parapet or curb were present, this would also be included by adding its contributory stiffness to the exterior stations (Ref 12).

The selection of the increment length in the Y-direction is influenced by the diaphragm spacing and span lengths. The interior diaphragms are spaced uniformly at 22 feet whereas the end spacing is 24.646 feet. The main beams have a total length of 160 feet and rest on four supports with spans as indicated in Fig 13. The Y-direction increment length was adjusted to $h_y = 5$ feet by resetting the exterior span lengths to 50 feet and relocating the diaphragms at alternating 25-foot and 20-foot intervals. The schematic



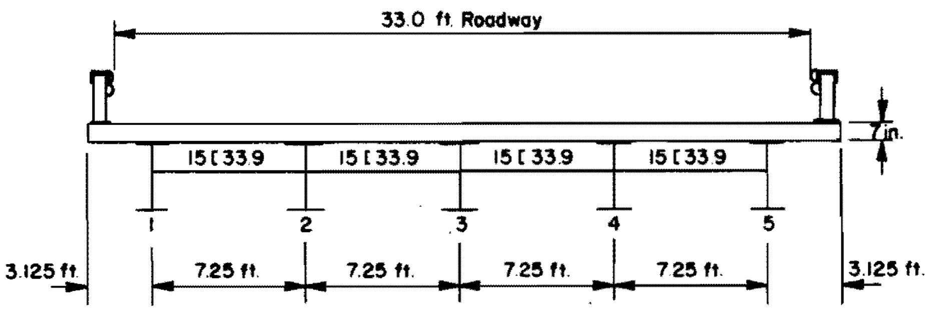


Fig 13. Three-span highway bridge.

plan of the structure as modeled is shown in Fig 14. The interior diaphragms are not offset from their original locations by more than 2 feet, which will not appreciably affect the final distribution of loads among adjacent beams. The exterior span lengths were increased by only 1 percent, which is also a negligible effect.

Stiffnesses are computed for the main beams, which are 33 WF 118, and the transverse diaphragms as well as for the 7-inch slab. Note that the end diaphragms are 14 WF 34 sections with cut bottom flanges whereas interior ones are 15 [33.9 sections. The slab bending and twisting stiffnesses are computed by the conventional formulas $D = \frac{Et^3}{12(1 - \nu^2)}$ and $C^t = \frac{Et^3}{12(1 + \nu)}$ with $\nu = 0.15$ and $E = 3 \times 10^3$ kips per sq. in. All stiffness values are given in Fig 14.

Supports are specified as large springs to effectively restrain the beams vertically. If a support were assumed to settle a given amount, the imposed settlement can be set by applying an appropriate, large downward load at the location of the large support spring. For instance, to set a 0.1-foot displacement, a load of -1×10^{14} kips is placed over the support, which has a fictitious large modulus of 1×10^{15} kips per foot. Thus, the load completely overpowers any other restraint offered by the structure and is resisted directly by the support at the equilibrium deflection of -0.1 foot, which counterbalances the load. A much more realistic simulation of a support settlement problem would be to input an elastic support spring which, when subjected to the actual structure's dead load, would deform to about the same location. If the support is drilled shafts or pilings, load-settlement curves can be used directly to estimate a support modulus. The elastic shortening of the columns can also be included as part of the support spring modulus.

Problem 1001. Dead Load, Non-Composite

The given structural system is first analyzed for the dead load of the 33 WF 118 beams and the overlying 7-inch concrete deck, by considering a grid system composed of the main beams and the transverse diaphragms. The beams carry dead weight of the slab and beams as line loads. It should be noted that the dead load carried by the outer beams is somewhat less than that carried by the interior ones, because the contributory area of the slab is smaller.

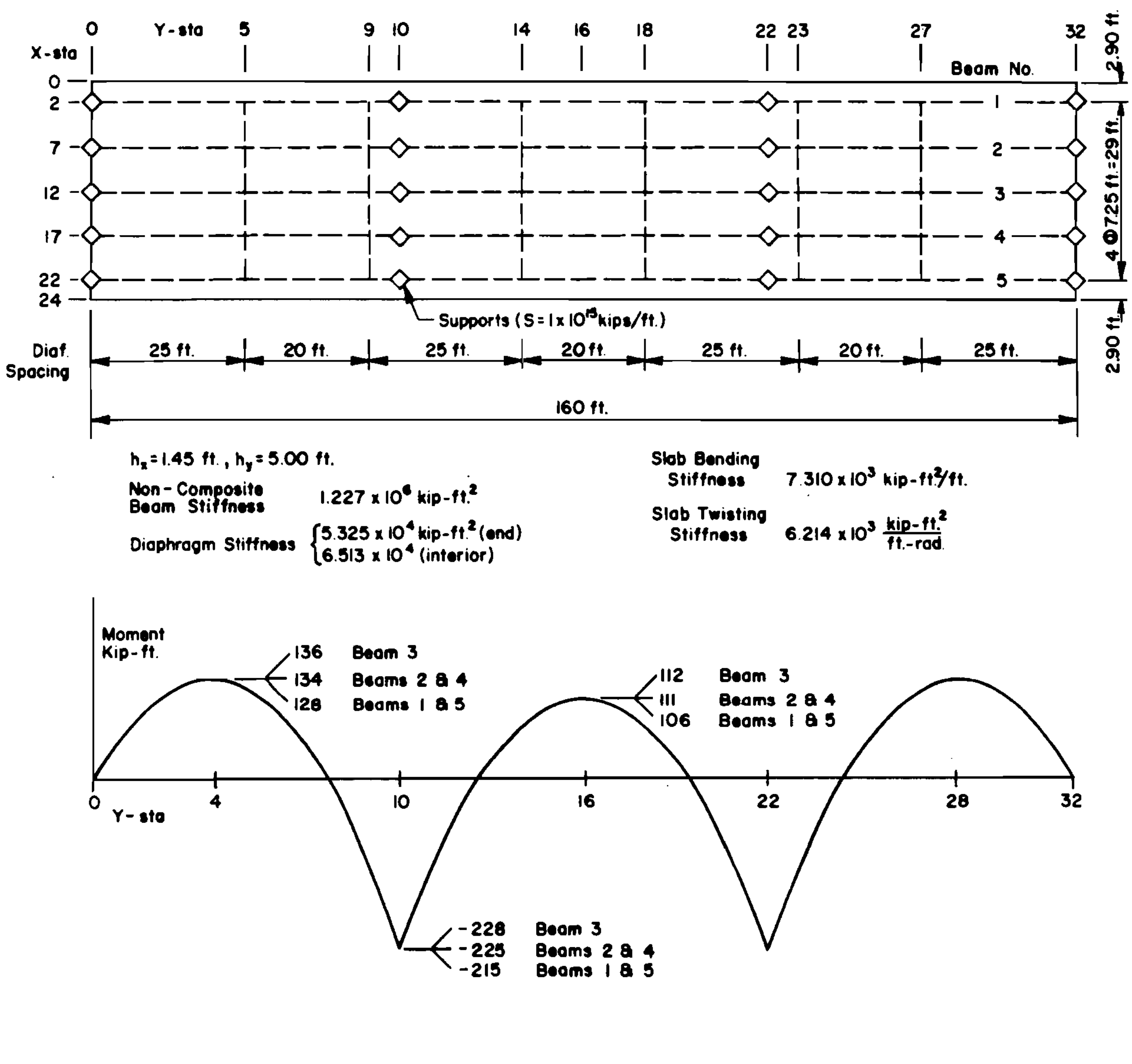


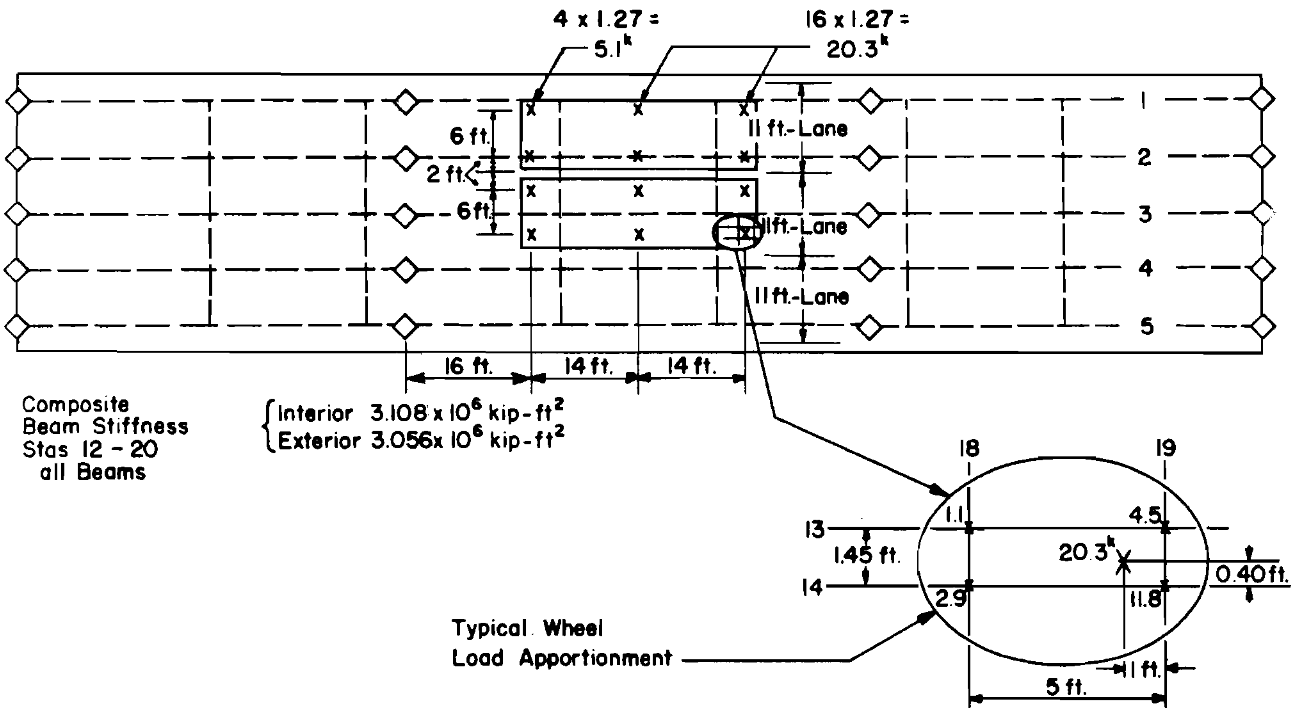
Fig 14. Problem 1001. Three-span structure as modeled, showing dead load beam moments.

Moment variations from the computed results are plotted along the main girders (Fig 14). For the center Beam No. 3, the maximum positive moments on center and exterior spans are 112 and 136 kip-feet, and the maximum negative support moment is -228 kip-feet. Note that the moment curves follow the same pattern for all the beams, with the maximum difference in ordinates not exceeding 3 percent. These moment values are in almost perfect agreement with those obtained from a conventional three-span continuous beam solution (Ref 11). Since the dead load is shared almost equally by all five beams, the diaphragms do not participate.

<u>Problem 1002. Two HS20 Trucks, Maximum Positive Moment for Beam No. 2</u>

Next the structural system is analyzed for the maximum positive moment in the center spans of the main beams, considering HS20 truck loadings (Ref 1). Two trucks are positioned in adjacent lanes with the center axle of the standard three-axle trucks placed at the center line of the middle span, as shown in Fig 15. The wheel loads are apportioned to adjacent stations, as depicted in Fig 15, since the wheel spacing and the lane boundaries do not directly fit the stationing and the increment lengths chosen for the analysis. This spreading or apportionment of the loads causes no appreciable error in the results for the bending moments in the main beams. Local slab results are affected and can be adjusted for actual loadings or another solution run with a fine mesh in the local slab area, considering only a portion of the slab (Ref 12). The wheel loads shown in Fig 15 are increased by an impact factor of 27.0 percent, which is computed using the length of the loaded span under consideration (Ref 1).

In addition to bending stiffnesses computed and input for the slab and diaphragms, a composite-beam stiffness is described for the main beams and the overlying slab. After the concrete deck has hardened, the beams and the slab within an appropriate effective width act compositely when subjected to positive bending. The effective slab width for composite action for this structure is taken to be the beam spacing of 7.25 feet (Ref 1). The composite stiffness is input at stations which fall within the approximate positive moment areas, which are from Y-station 12 through 20. Note that the composite stiffness along the exterior beams is less than for the interior ones, because the slab area on the overhangs is smaller.



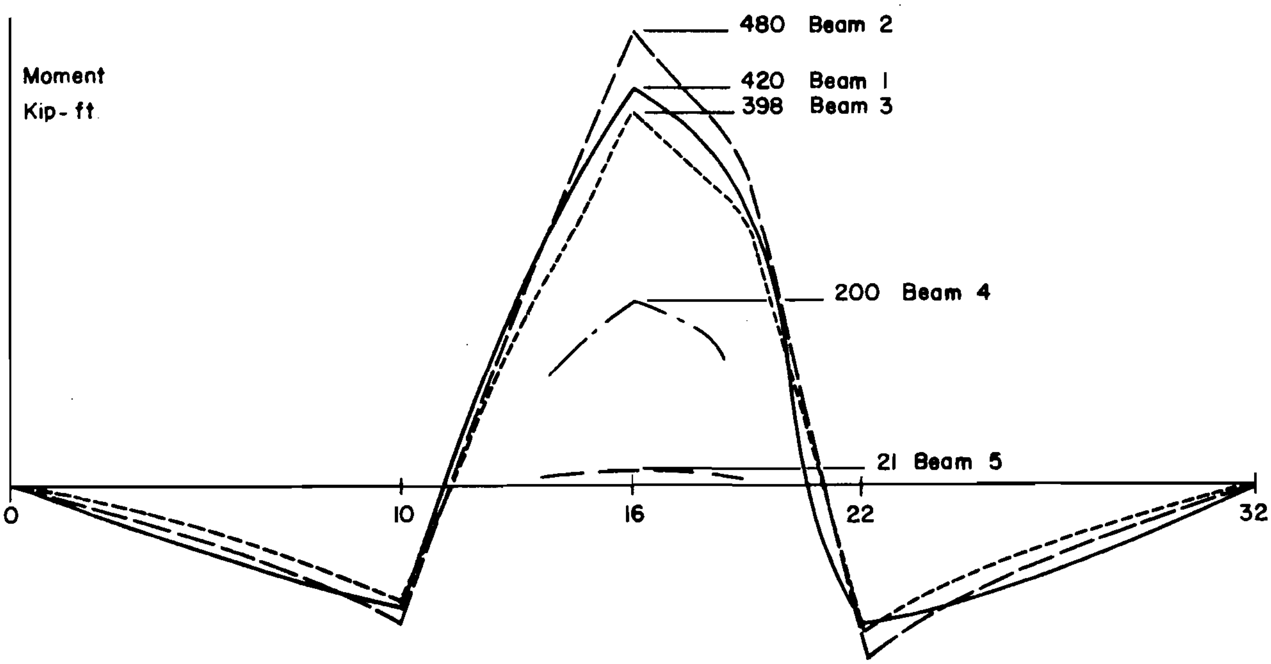


Fig 15. Problem 1002A. HS20 truck loading for maximum positive moment in center span.

The computer results of this solution are summarized by the plots of main beam moments in Fig 15. Dead load moments are not included. As might be expected from the chosen position of the loads, the maximum positive moment occurs in Beam No. 2 and is 480 kip-feet. This value was compared to the conventional line-member analysis obtained from a beam-column solution (Ref 11) using the standard AASHO distribution (Ref 1), which gave a slightly conservative moment value of 495 kip-feet. This demonstrates that the standard distribution formulas are indeed appropriate when applied to this type of structure with these spans and beam spacing.

Problem 1003. Lane Loading, Maximum Negative Moment

The system is next analyzed for the maximum negative moment in the main beams, considering an HS20 lane loading (Ref 1). Two spans are loaded with the uniform lane loading of 818 pounds per foot on two adjacent lanes, which includes an impact factor of 27.8 percent. In addition, two concentrated loads with impact of 23.0 kips are placed at the mid-spans. Both the concentrated and the uniform loads are distributed over a 10-foot lane width as shown in Fig 16.

Slab and diaphragm stiffnesses are the same as in the preceding problem except that the composite stiffnesses of the beams are now input at Y-stations along the center and first spans, which are within the approximate positive moment areas.

The computed moment variations are plotted for the first four main beams in Fig 16. The maximum value of -236 kip-feet can be compared to a conventional analysis value of -247 kip-feet, which was obtained from a beam-column solution considering the specified AASHO distribution (Refs 1 and 11).

Problem 1004. Alternate Truck Loading for Maximum Negative Moment

An alternate solution for possible maximum negative moment is illustrated by considering a two-truck HS20 loading condition in adjacent lanes of the first two spans, as shown in Fig 17.

Wheel loads of the two trucks moving opposite to each other in adjacent lanes are increased by the same impact factor, 27.8 percent, as in Prob 1003. The trucks are positioned to produce the maximum negative support moment for

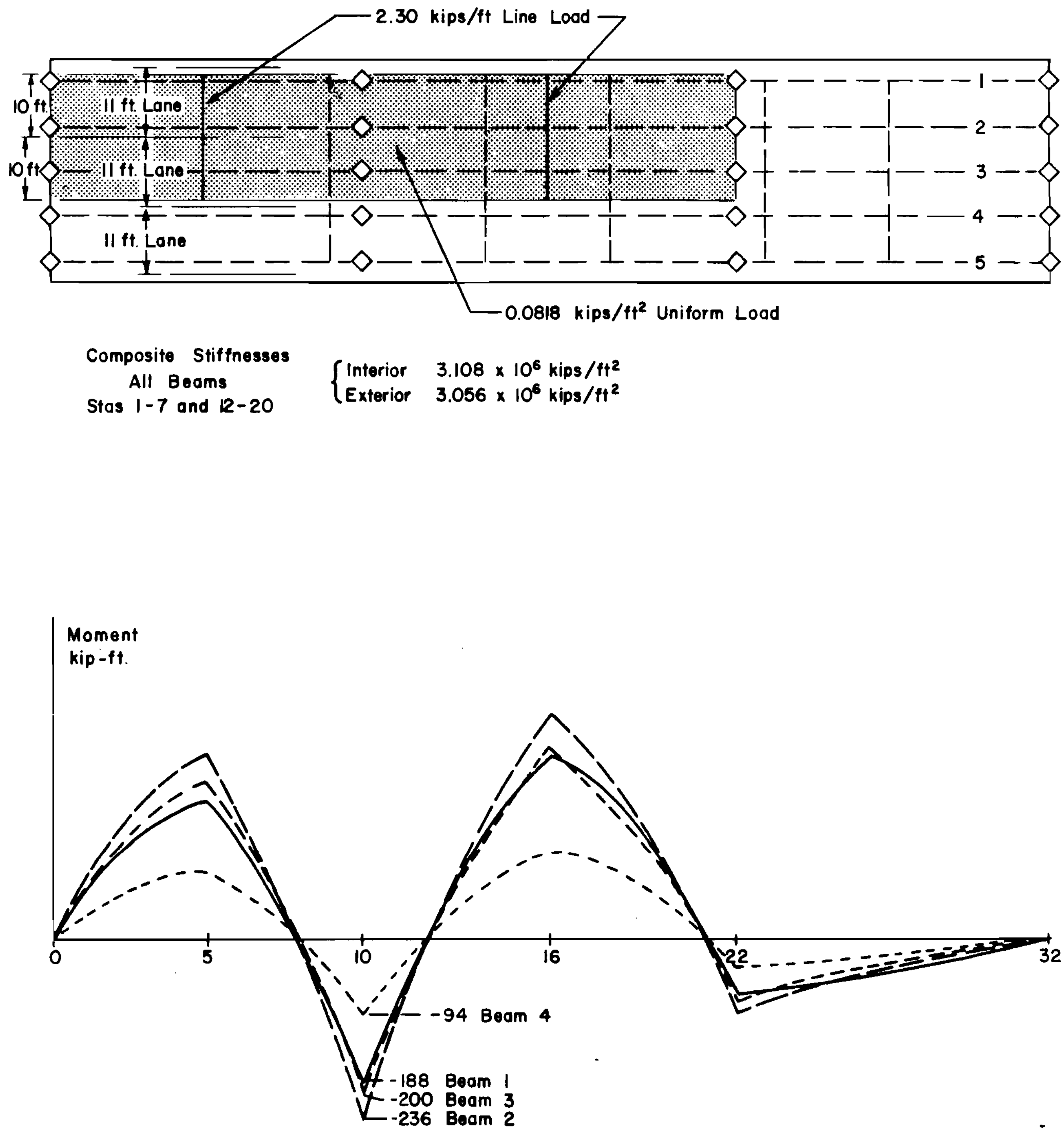
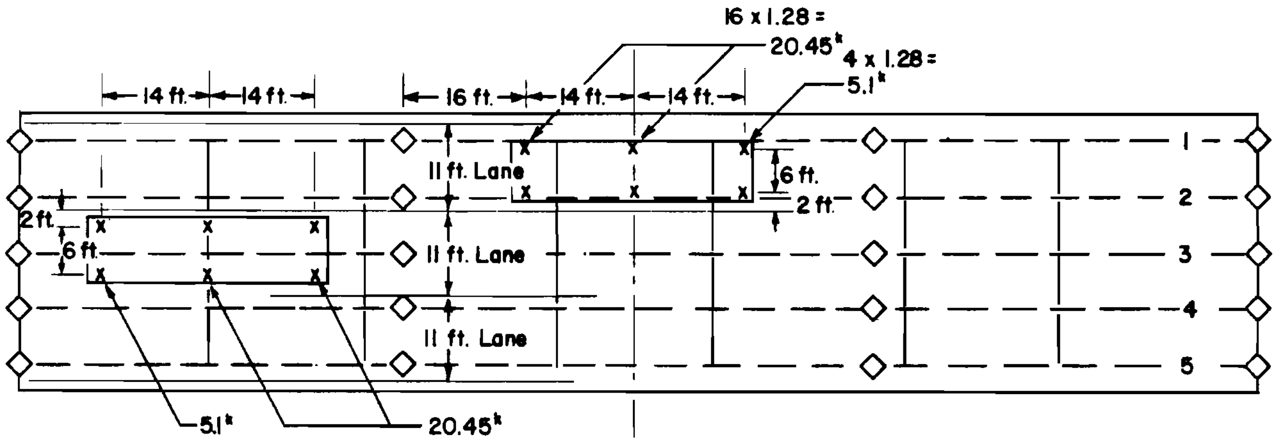


Fig 16. Problem 1003. HS20 lane loading for maximum negative moment.



Beam No.	1	2	3	4	5
Composite Stiffness Sta to Sta	12-20	1-7 and 12-20	1-8 and 13-21	1-9 and 16-24	1-11 and 20-31

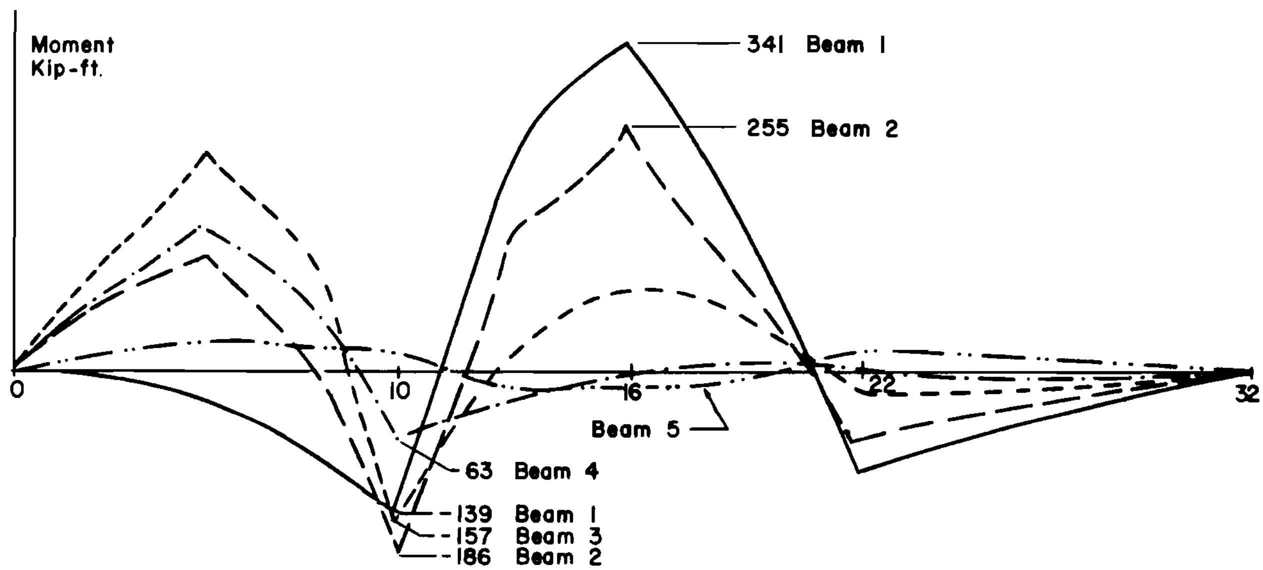


Fig 17. Problem 1004. Two HS20 trucks in adjacent lanes for maximum negative moment.

Beam No. 2. The beam composite stiffnesses are adjusted to include the stations indicated in Fig 17.

The computed moment variations in the beams are shown in Fig 17. The lane loading condition of Problem 1003 is seen to have created a larger maximum, but this arrangement of truck loadings might possibly control for other bridge configurations. In addition, it is a more realistic loading condition than the lane loading of Prob 1003 which is an approximation used for the convenience of designers (Footnote to Art 1.2.7, Ref 1).

Fig 18 shows the moment in the first interior diaphragm which has the central truck wheels of one load pattern placed over it (Fig 17). Note that even though the beams act as supports for the diaphragm, no negative moment occurs at Beam No. 3 since the adjacent beams are also deflecting. The maximum moment of 23.2 kip-feet corresponds to an approximate diaphragm stress of 6600 psi.

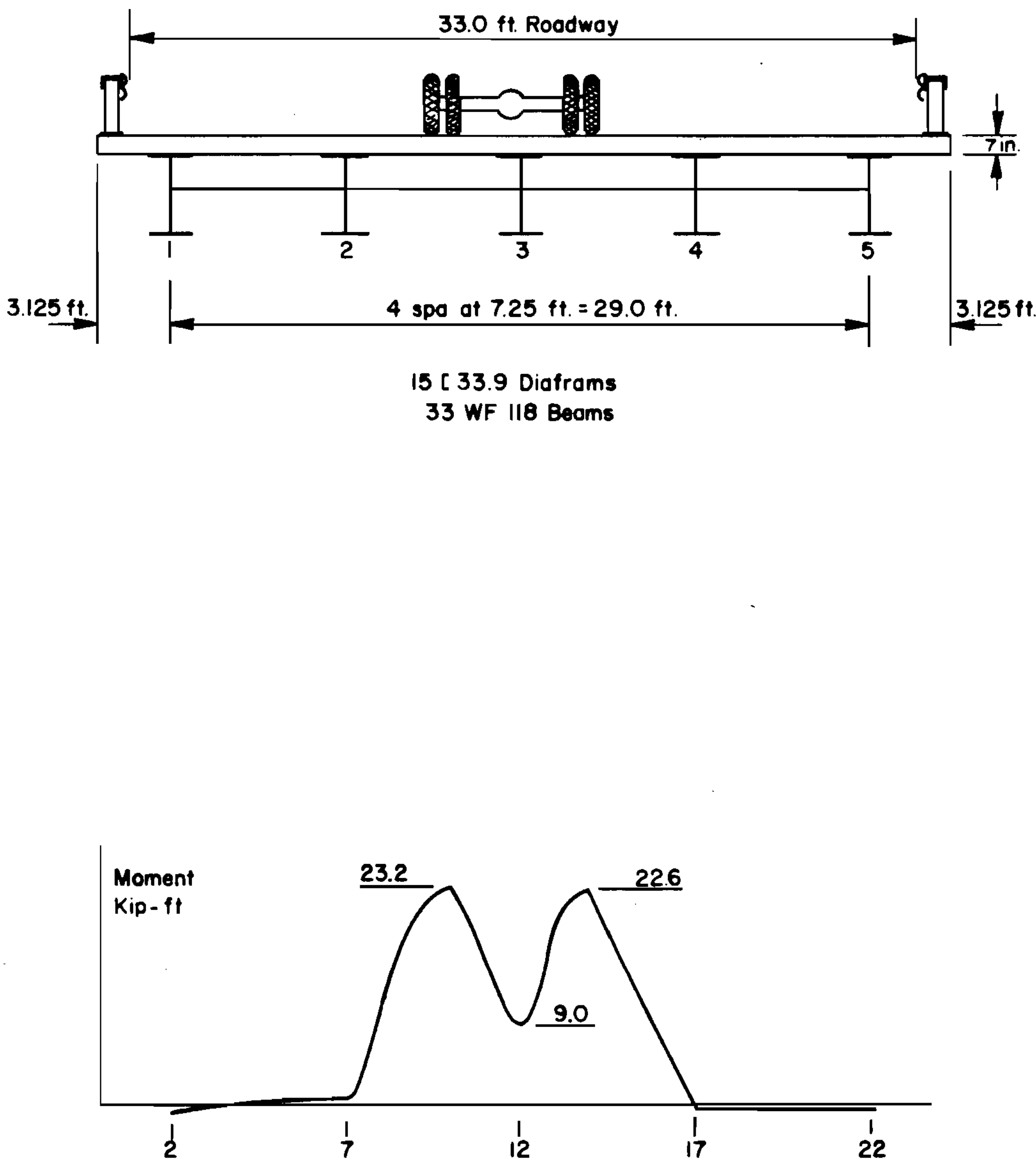


Fig 18. Problem 1004. Moment in first interior diaphragms.

CHAPTER 5. SUMMARY AND RECOMMENDATIONS

The problem of analysis of orthogonal bridge floor systems has been presented. A computer program with documentation is included. This program could be considered the culmination of work on this research project in this area. The method is not limited to bridge floor systems, however. Pavement slabs, building slabs, stiffened and anchored bulkheads, and other two-dimensional plate or grid systems are among the structures to which the analysis can be applied.

The computer program is written and organized in a manner which will facilitate its modification and extension for specific applications. Of primary need is a series of user-oriented data-generation routines which will use this program as the basic solver. Among these would be data generation routines for constant thickness slab structures, and beam and slab systems such as presented in the included example problem.

The existing discrete-element model requires the user to know and specify the distribution of axial or in-plane thrust throughout the slab or plate. A valuable extension of this work would be the modification of the model to include axial deformations, and the development of the force-deformation equations for in-plane thrust. Not only could the axial and bending solutions be coupled for combined axial-bending analysis of plates, but the in-plane analysis could be applied to plane-stress problems. Furthermore, an in-plane solution would be required for the analysis of plates subjected to thermal gradients.

A further valuable extension of the program would be to automatically include the nonlinear stiffness properties of reinforced concrete beams. This would allow the user to make a more realistic determination of the ultimate capacity of the structure. Correlation with experimental data would be extremely useful. Carefully controlled model studies such as those currently being made in Research Study 3-5-71-158, "Diaphragm Requirements for Prestressed Concrete Bridges," could easily be compared to analytical solutions using the included computer program. Adjustments in estimated stiffnesses could then be made for application to prototype concrete structures.

REFERENCES

1. The American Association of State Highway Officials, Standard Specifications for Highway Bridges, Tenth Edition, Washington, D. C., 1969.

2. The American Standards Association, "American Standard FORTRAN," New York, 1966.

3. Endres, F. E., and Hudson Matlock, "An Algebraic Equation Solution Process Formulated in Anticipation of Banded Linear Equations," Research Report 56-19, Center for Highway Research, The University of Texas, Austin, January 1971.

4. Hudson, W. R., and Hudson Matlock, Discontinuous Orthotropic Plates and Pavement Slabs," Research Report 56-6, Center for Highway Research, The University of Texas, Austin, May 1966.

5. Huffington, N. J. Jr., "Theoretical Determination of Rigidity Properties of Orthogonally Stiffened Plates," Applied Mechanics Division, ASME, November 1955.

6. Ingram, W. B., and Hudson Matlock, "A Finite Element Method for Bending Analysis of Layered Structural Systems," Research Report 56-5, Center for Highway Research, The University of Texas, Austin, June 1967.

7. Kelly, Allen E., and Hudson Matlock, "Dynamic Analysis of Discrete-Element Plates on Nonlinear Foundations," Research Report 56-17, Center for Highway Research, The University of Texas, Austin, January 1970.

8. Matlock, Hudson, and T. A. Haliburton, "A Finite-Element Method of Solution for Linearly Elastic Beam-Columns," Research Report 56-1, Center for Highway Research, The University of Texas, Austin, September 1966.

9. Matlock, Hudson, and W. R. Hudson, "Cracked Pavement Slabs with Non-Uniform Support," Journal of the Highway Division, American Society of Civil Engineers, Vol. 93, No. HW1, Proceedings Paper No. 5191, April 1967, pp. 19-42.

10. Matlock, Hudson, and Wayne B. Ingram, "A Computer Program to Analyze Bending of Bent Caps," Research Report 56-2, Center for Highway Research, The University of Texas, Austin, October 1966.

11. Matlock, Hudson, and Thomas P. Taylor, "A Computer Program to Analyze Beam-Columns under Movable Loads," Research Report 56-4, Center for Highway Research, The University of Texas, Austin, June 1968.

12. Panak, John J., and Hudson Matlock, "A Discrete-Element Method of Multiple-Loading Analysis for Two-Way Bridge Floor Slabs," Research Report 56-13, Center for Highway Research, The University of Texas, Austin, January 1970.

13. Pearre, Charles M. III, and W. R. Hudson, "A Discrete-Element Solution of Plates and Pavement Slabs Using a Variable-Increment-Length Model," Research Report 56-11, Center for Highway Research, The University of Texas, Austin, April 1969.

14. Salani, Harold, and Hudson Matlock, "A Finite-Element Method for Transverse Vibrations of Beams and Plates," Research Report 56-8, Center for Highway Research, The University of Texas, Austin, June 1967.

15. Stelzer, C. Fred, Jr., and W. R. Hudson, "A Direct Computer Solution for Plates and Pavement Slabs," Research Report 56-9, Center for Highway Research, The University of Texas, Austin, October 1967.

16. Taylor, Thomas P., and Hudson Matlock, "A Finite-Element Method of Analysis for Composite Beams," Research Report 56-10, Center for Highway Research, The University of Texas, Austin, January 1968.

17. Troitsky, M. S., <u>Orthotropic Bridges Theory and Design</u>, The James F. Lincoln Arc Welding Foundation, Cleveland, 1967.

18. Vora, Mahendrakumar R., and Hudson Matlock, "A Discrete-Element Analysis for Anisotropic Skew Plates and Grids," Research Report 56-18, Center for Highway Research, The University of Texas at Austin, August 1970.

APPENDIX 1

GUIDE FOR DATA INPUT

GUIDE FOR DATA INPUT FOR SLAB 49

with supplementary notes

extract from

A DISCRETE-ELEMENT METHOD OF ANALYSIS FOR ORTHOGONAL SLAB AND GRID BRIDGE FLOOR SYSTEMS

by

John J. Panak and Hudson Matlock

May 1972

IDENTIFICATION OF RUN (2 cards per run)

Enter descriptive alphanumeric information - - date of run, user's name,

and the chosen units should always be included

IDENTIFICATION OF PROBLEM (1 card each problem; program stops if PROB NUM is left blank)

PROB NUM		Alphanumeric problem description
5		11

TABLE 1. CONTROL DATA (2 cards for each problem)

Enter "1" to KEEP prior TABLE

2	3	4	5	6	7	8	9		Multiple Load Option
5	10	15	20	25	30	35	40		46 50

Multiple Load Option: +1 for Parent Problems; -1 for Offspring Problems

Number of cards added for TABLE

2	3	4	5	6	7	8	9		Enter "1" for Statics Check	Enter "1" for Principal Stress	Plot Options Profiles	Plot Options 3-D
*											**	***
5	10	15	20	25	30	35	40		46 50	55	60	65

* Number of cards added must be zero if preceeding table is kept or if this is an offspring problem.

** Profile plots are for areas specified by Table 8. If option is zero or blank, printer plot is made; if 1, microfilm; if 2, printer and microfilm; if 3, paper. The IBM 360 program version has only option zero.

*** Enter 1 to obtain exaggerated isometric (three-dimensional paper plot) display of deflections.

TABLE 2. CONSTANTS (one card, none if Table 2 of preceding problem is kept)

Num X	Incr Y		Increment Length in X-Direction h_x	Increment Length in Y-Direction h_y	Poisson's Ratio ν	Slab Thickness t
5	10		21–30	40	50	60

TABLE 3. JOINT BENDING STIFFNESSES, LOADS, AND SUPPORTS (Maximum of 300 cards including those kept)

From X	From Y	Through X	Through Y	Slab Bending Stiffness D^x	Slab Bending Stiffness D^y	Beam Bending Stiffness F^x	Beam Bending Stiffness F^y	Load Q	Spring S
5	10	15	20	30	40	50	60	70	80

TABLE 4. JOINT RESTRAINTS AND APPLIED MOMENTS (Maximum of 50 cards including those kept)

From X	From Y	Through X	Through Y		Rotational Restraint R^x	Rotational Restraint R^y	Applied Moment T^x	Applied Moment T^y
5	10	15	20		41–50	60	70	80

TABLE 5. MESH TWISTING STIFFNESSES (Maximum of 100 cards including those kept)

From X	From Y	Through X	Through Y	Twisting Stiffness C^t
5	10	15	20	30

TABLE 6. BAR AXIAL THRUSTS (Maximum of 50 cards including those kept)

From X	From Y	Through X	Through Y		Slab Axial Thrust P^x	Slab Axial Thrust P^y	Beam Axial Thrust $\bar{P}^x$	Beam Axial Thrust $\bar{P}^y$
5	10	15	20		41–50	60	70	80

TABLE 7. MULTIPLE LOADS (Maximum of 100 cards including those kept)

From X	From Y	Through X	Through Y	Load Q
5	10	15	20	61–70

TABLE 8. PROFILE OUTPUT AREAS (Maximum of 10 cards, including those kept)

From X	From Y	Through X	Through Y	Defl	X-Mom	Y-Mom	Princ Mom or Stress
					*	*	
5	10	15	20	25	30	35	40

* Enter "1" for slab moments,
"2" for beam moments

TABLE 9. PRINTED OUTPUT LIMITS (Maximum of 10 cards including those kept)

From Y	Through Y
6–10	16–20

If this table is omitted, all results will be printed. Each Y-bounded area specified includes the complete X-width.

TERMINATION OF RUN (one blank PROB NUM card)

5	Alphanumeric information may be punched here if desired such as "END OF DATA".

GENERAL PROGRAM NOTES

The data cards must be assembled in proper order for the program to run.

A consistent system of units must be used for all input data, for example, kips and feet.

All 2 to 5-space words are understood to be right-justified integers or whole decimal numbers. . . . + 4 3 2 1

All 10-space words are floating-point decimal numbers . . . . + 4 . 3 2 1 E + 0 3

Any number of problems may be run together.

If the KEEP option for Table 2 is set equal to 1, there must be no card input for that table.

For Tables 3 through 9, any data from prior problems may be retained in card image storage by the KEEP options. The number of cards input for each table is independent of the KEEP options, except that the cumulative total of cards cannot exceed the specified amount for each table.

Card counts for Tables 3 through 9 should be carefully rechecked after coding is completed.

The multiple-load option is exercised for problem series in which only the load positions and magnitudes will vary. This is done by input of new loads in Table 7. Tables 2 through 6 must be held and no cards may be added to them. The first problem in a series is the Parent and is specified by entering +1; successive loadings are the Offspring and are specified by entering -1. If the option is left blank or zero, the problem is complete within itself. Tables 8 and 9 may be used as desired for all problems.

The options for Statics Check or Principal Stress may be exercised by entering 1. If the Principal Stress is to be used, then a slab or plate thickness must be available in Table 2. The option is useful only if the real slab is of uniform thickness with no discontinuities. The output value of principal stress has the same sign as the principal moment from which it is computed.

Profile plots may be designated for areas specified by Table 8. If the option is left blank or set equal to zero, the printer creates profile plots along with tabulated values; if the option is equal to 1, the plot is made on microfilm with no tabulation; if the option is set equal to 2, the combination of 0 and 1 is obtained; if the option is set equal to 3, only paper plots are made. The IBM version of the program has only option zero which yields printer plots.

A pseudo three-dimensional paper plot of the entire set of deflections may be obtained by exercising the 3-D option (IBM and CDC versions).

TABLE 2. CONSTANTS

Variables:	h_x, h_y	ν	t
Typical Input Units:	in.	none	in.

This table is omitted for Offspring problems

Poisson's ratio will be taken as zero unless specified (always positive). It is not needed when running grid-beam type problems since no Poisson's effects are considered for the beam elements.

Slab or plate thickness must be entered if the Stress Option in Table 1 is used. The stress is computed directly from the value of principal moment and has the same sign.

TABLES 3 and 4. JOINT BENDING STIFFNESSES, LOADS, SUPPORTS, RESTRAINTS, AND APPLIED MOMENTS DATA

Variables:	D^x, D^y	F^x, F^y	Q	S	R^x, R^y	T^x, T^y
Typical Input Units:	$\frac{\text{lb-in}^2}{\text{in.}}$	lb-in^2	lb	$\frac{\text{lb}}{\text{in.}}$	$\frac{\text{in-lb}}{\text{rad}}$	in-lb

Unit stiffness values D^x and D^y for a slab or plate and concentrated stiffness values F^x and F^y for beams are input at all joints. The values are reduced proportionately for edges.

Customary relationships for isotropic slabs or plates and beams of known cross section are given here for reference:

$$D^x = D^y = \frac{Et^3}{12(1-\nu^2)} \qquad C^t = \frac{Et^3}{12(1+\nu)} \qquad F = EI$$

E is the Modulus of Elasticity, t , the plate or slab thickness, ν is Poisson's ratio, and I is the total beam cross section moment of inertia including composite effects if present.

Load values Q and support springs S for any joint are determined by multiplying the unit load or unit support value by the appropriate area of the real slab assigned to that joint. Hinged supports are provided by using large S values. Concentrated loads that occur between joints can be apportioned geometrically to adjacent joints.

All data are described with a coordinate system which is related to the X and Y-station numbers. To distribute data over a rectangular area, the lower left hand and the upper right hand coordinates must be specified. Figure A2 illustrates a sample data input.

To specify data at a single location, the same coordinates must be specified for both the "From" and "Through" coordinates.

The "Through" coordinates must always be equal to or numerically greater than the "From" coordinates.

The user may input values on the edges of the slab and the corners to represent the proportionate area desired as illustrated in Fig A2.

There are no restrictions on the order of cards. The values input are algebraically accumulated at each coordinate.

TABLE 5. MESH TWISTING STIFFNESSES

Variable: C^t

Typical Input Unit: $\frac{\text{lb-in}^2}{\text{in/rad}}$

Unit twisting stiffness C^t is defined for the mesh of the plate or slab surrounded by four rigid bars and four joints. The mesh is numbered according to the joint number at the upper right corner of the mesh as shown in Fig A1.

The same general notes as listed for Tables 3 and 4 are applicable.

TABLE 6. BAR AXIAL THRUSTS

Variables:	P^x, P^y	$\overline{P}^x$, $\overline{P}^y$
Typical Input Units:	lb	lb

All data in this table are concentrated. Distributed data must be summed over the width of the increment involved. Proportionate values can be used along edges.

All tension (+) or compression (-) values P^x are specified for each X-bar in the X-direction. Since it is a bar force, no coordinate should be used which would specify a P^x value in a bar outside the real plate or slab. The vars are numbered according to the joint number at the increasing station end of the bar as shown in Fig A1. P^y values are specified in the Y-direction.

The same general notes as listed for Tables 3 and 4 are applicable.

TABLE 7. MULTIPLE LOADS

Variable: Q

Typical Input Unit: lb

When a problem is such that only the load changes from problem to problem, it is appropriate to enter it in this table and hold all other stiffness, load, and geometrical data of Tables 2, 3, 4, 5, and 6 from the previous problem, thus creating an Offspring problem. Any loads entered or held in Table 3 are added to the loads of Table 7.

The multiple-loading options are specified in Table 1. The greatest amount of computer time is needed for the first problem in a multiple-loading problem series and subsequent problems are then solved in a fraction of the solution time.

Each card may encompass up to a maximum of 300 points. For larger areas, additional cards may be used to the limit of 10, including those kept from previous problems.

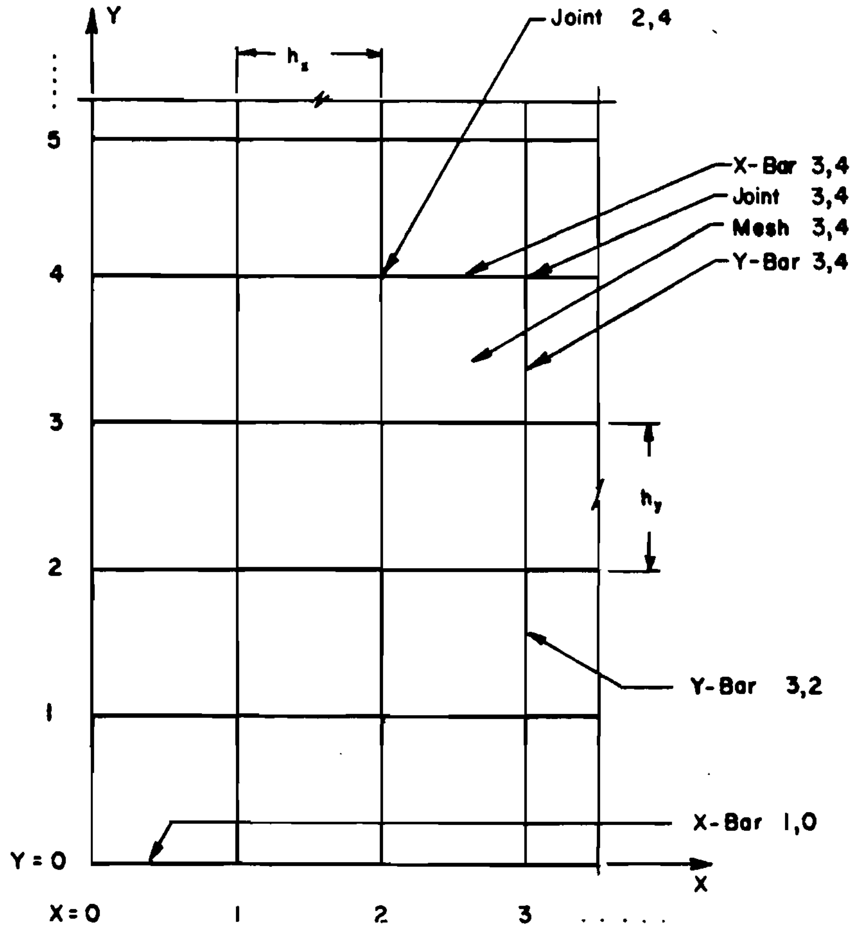
If profile plot options in Table 1 were set to 1 or 3, no tabulated output of Table 8 areas is printed. A blank or 2 option will cause tabulated and printer display of the selected profiles.

Any one or all four types of profile output may be selected by entering a 1 for those desired. Beam moments may be chosen by entering a 2. One limitation for the moment options is that all areas entered or kept from the previous problem must be either for slab or beam X and Y-moments. A mixture of slab and beam profile output within a problem is disallowed. Parent-offspring may have beam-slab options.

TABLE 9. PRINTED OUTPUT LIMITS

If this table is omitted, the complete printout of results is obtained. Partial output may be obtained by specifying the sections to be printed within the Y-bounded limits designated on each card. Up to 10 Y-bounded sections may be printed.

Y-bounded areas may overlap or be contiguous.



Joint Data: D^x, D^y, F^x, F^y, Q, S, R^x, R^y, T^x, T^y

(D^x and D^y are per unit width, all others are concentrated values)

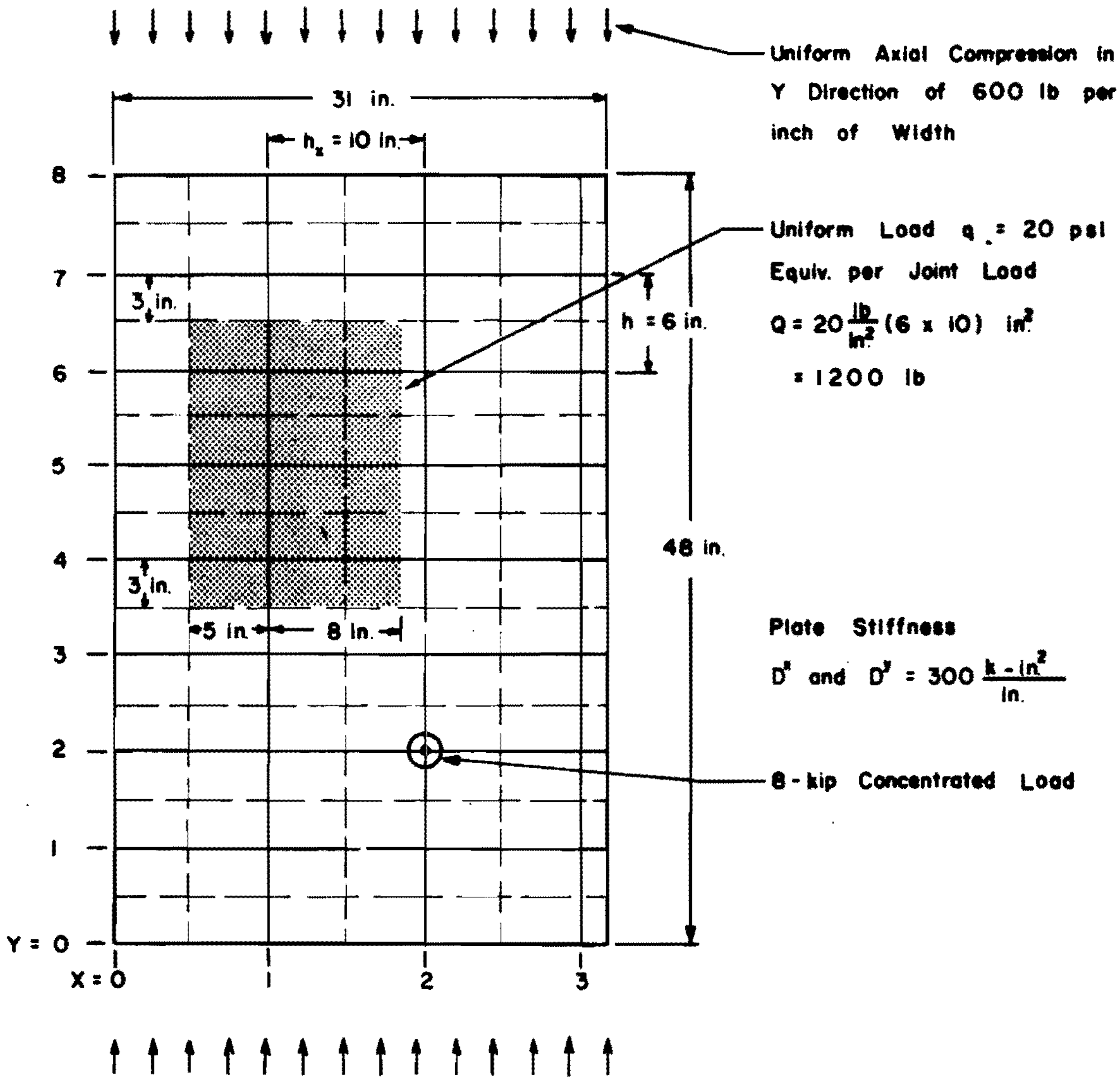
Mesh Data: C^t

(C^t is per unit width)

Bar Data: P^x, P^y, β^x, β^y

(all values are concentrated)

Fig A1. Data coordinate numbering system.



From X	From Y	Through X	Through Y	D^x and D^y	Q	P^y
0	0	2	8	7.500E+04		
0	1	2	7	7.500E+04		
1	1	2	7	1.500E+05		
3	0	3	8	9.000E+04		
3	1	3	7	9.000E+04		
1	4	1	6		-1.200E+03	
2	4	2	6		-3.600E+02	
2	2	2	2		-8.000E+03	
0	1	0	8			-3.000E+03
1	1	2	8			-6.000E+03
3	1	3	8			-3.600E+03

data incomplete for this sample

Fig A2. Sample data input.

APPENDIX 2

GLOSSARY OF NOTATION FOR COMPUTER PROGRAM SLAB 49

```
C---------NOTATION FOR SLAB 49 AND SB49B

C          A( ,1)                  CONTINUITY COEFFICIENT
C          AA( ,1 )                COEFF IN STIFFNESS MATRIX
C          ALP                     ANGLE ON MOHRS CIRCLE
C          AM1( ,1)                CONTINUITY COEFFICIENT A( ,1) AT J-1
C          AM2( ,1)                CONTINUITY COEFFICIENT A( ,1) AT J-2
C          AN1( ) , AN2( )         IDENTIFICATION AND REMARKS ( ALPHA-NUM )
C          AT( ,1)                 TEMP STORAGE FOR A( ,1) CONTINUITY COEFF
C          BB( ,3)                 COEFFS IN STIFFNESS MATRIX
C          BEMX , BEMY             ALPHANUMERIC, = BEAM X, BEAM Y
C          BETA                    HALF THETA ( COUNTER CLOCKWISE IS + )
C          BLNK                    ALPHANUMERIC, = 6 BLANKS
C          BMA                     AVERAGE OF X AND Y BENDING MOMENTS
C          BMBX( )                 BEAM BENDING MOMENT IN X DIRECTION
C          BMBY( )                 BEAM BENDING MOMENT IN Y DIRECTION
C          BMBYM1( )               BMBY AT J-1
C          BMBYP1( )               BMBY AT J+1
C          BMOM                    ALPHANUMERIC, = MOMENT
C          BMP                     BMX MINUS BMA FOR MOHRS CIRCLE
C          BMQ                     FIRST PRINCIPAL BENDING MOMENT
C          BMR                     RADIUS OF MOHRS CIRCLE
C          BMT                     SECOND PRINCIPAL BENDING MOMENT
C          BMX( )                  SLAB BENDING MOMENT IN THE X DIRECTION
C          BMY( )                  SLAB BENDING MOMENT IN THE Y DIRECTION
C          BMYM1( )                BMY AT J-1
C          BMYP1( )                BMY AT J+1
C          CC( ,5)                 COEFFS IN STIFFNESS MATRIX
C          CM( )                   PLATE TWISTING STIFFNESS PER UNIT WIDTH
C          CMK                     ALPHANUMERIC, = CHECK
C          CMN( )                  INPUT VALUE OF CM IN TABLE 5
C          CMP1( )                 CM AT J+1
C          CRD( )                  CROSS BENDING STIFFNESS FOR PR EFFECTS
C          DD( ,3)                 COEFFS IN STIFFNESS MATRIX
C          DX( ) , DY( )           SLAB BENDING STIFFNESSES PER UNIT WIDTH
C          DXM1( )                 DX AT J-1
C          DXP1( )                 DX AT J+1
C          DXN( ) , DYN( )         INPUT VALUES OF DX AND DY IN TABLE 3
C          DYM1( )                 DY AT J-1
C          DYP1( )                 DY AT J+1
C          EE( ,1)                 COEFF IN STIFFNESS MATRIX
C          FF( ,1)                 COEFF IN LOAD VECTOR
C          FX( ) , FY( )           BEAM BENDING STIFFNESSES
C          FXN( ) , FYN( )         INPUT VALUES OF FX AND FY IN TABLE 3
C          FYM1( )                 FY AT J-1
C          FYP1( )                 FY AT J+1
C          HX                      INCREMENT LENGTH IN X DIRECTION
C          HXDHY                   HX DIVIDED BY HY
C          HXDHY3                  HX DIVIDED BY HY CUBED
C          HY                      INCREMENT LENGTH IN Y DIRECTION
C          HYDHX                   HY DIVIDED BY HX
C          HYDHX3                  HY DIVIDED BY HX CUBED
C          I , I1 , I2             X DIRECTION INDEXES (ISTA+2)
C          IBOS                    BEAM OR SLAB MOMENT PROFILE PLOT SWITCH
C          IBOST                   TEMP VALUE OF IBOS
C          IBUF( )                 PLOTTER BUFFER
C          ID                      TEMP VALUE OF ID1 THRU ID4
C          ID1 THRU ID4            ALPHANUMERIC IDENTIFIERS FOR SUBROUTINE ZOT 1   CDC
C          ID1 THRU ID4            RANDOM INDEXES FOR FILES 1 THRU 4               IBM
C          IGSW                    REF GRID TYPE SWITCH FOR 3-D PLOTS
C          IN13( ) THRU            -FROM- X STATION USED IN
C            IN18( )                    TABLES 3 THRU 8
C          IN23( ) THRU            -THRU- X STATION USED IN
C            IN28( )                    TABLES 3 THRU 8
C          IOFFS                   INPUT OPTION TO PRINT PRINCIPAL STRESS
C          IPOP                    INPUT PLOT OPTION
C          IPRINT                  PRINTING CONTROL AT EACH J STEP
C          IROLL                   OPTION TO MOVE TO A NEW PLOT FRAME
C          ISTA                    X STATION
C          ISW                     SWITCH DEPENDANT ON END VALUES OF J
C          ITEMP                   ISTA AT MAX STATICS CHECK ERROR
C          ITEST                   ALPHANUMARIC BLANKS USED TO TERMINATE
C          ITMPP                   TEMP PRINT SWITCH FOR DEBUGGING
C          I3D                     INPUT OPTION FOR 3D PLOT OF DEFLECTIONS
C          I4D                     SWITCH USED TO TERMINATE PLOT FILE
C          J , J1 , J2             Y DIRECTION INDEXES (JSTA+2)
C          JN                      ACTUAL Y STA, EQUALS J-2
C          JN13( ) THRU            -FROM- Y STATION USED IN
C            JN19( )                    TABLES 3 THRU 9
C          JN23( ) THRU            -THRU- Y STATION USED IN
C            JN29( )                    TABLES 3 THRU 9
C          JSTA                    Y STATION
C          JTEMP                   JSTA AT MAX STATICS CHECK ERROR
C          K                       TEMX OR TEMY INDEX FOR SPLOT3 OR SPLOT4
C          KASEP                   TABLE 8 PRINCIPAL MOMENT OR STRESS OPTION
C          KASEW                   TABLE 8 DEFLECTION OPTION
C          KASEX , KASEY           TABLE 8 BENDING MOMENT OPTIONS
C          KEEP2 THRU KEEP9        KEEP OPTIONS FOR DATA TABLES 2 THRU 9
C          KLONG                   MAXIMUM NUMBER OF Y STAS
C          KML                     MULTIPLE LOAD OPTION IN PRIOR PROBLEM
C          KSHORT                  MAXIMUM NUMBER OF X STAS
C          KPROB                   PROBLEM NUMBER FROM PARENT PROBLEM
C          KROPT                   INPUT OPTION TO PRINT STATICS CHECK
C          LL                      DO LOOP INDEX FOR REVERSED J
C          LOP                     LINE OR POINT PLOT OPTION
C          L1                      KSHORT+3, USED FOR VARIABLE DIMENSIONING
C          L2                      KLONG+3, USED FOR VARIABLE DIMENSIONING
C          MC                      NUMBER OF CHARACTERS IN PLOT TITLE
C          ML                      MULTIPLE LOADING SWITCH
C          MOP                     MICROFILM (=0)  OR PAPER (=1) PLOT OPTION
C          MX                      NUMBER OF INCREMENTS IN X DIRECTION
C          MXP1 THRU MXP5          MX+1 THRU MX+5
C          MY                      NUMBER OF INCREMENTS IN Y DIRECTION
C          MYP1 THRU MYP5          MY+1 THRU MY+5
C          N                       INDEX FOR READING CARDS
C          NCD2 THRU NCD9          NUM CARDS IN TABLES 2 THRU 9, THIS PROB
C          NCT3 THRU NCT9          TOTAL NUMBER OF CARDS IN TABLES 3 THRU 9
C          NC13 THRU               STARTING VALUE FOR READING CARDS,
C            NC19                       TABLES 3 THRU 9
C          NDES                    TOTAL NUMBER OF DATA ERRORS
C          NDE1 THRU NDE9          NUMBER OF DATA ERRORS IN TABLES 1 THRU 9
C          NEND                    MAX NUMBER OF POINTS, EACH TABLE 8 CARD
C          NF                      INITIAL COUNTER, SOLUTION ROUTINES (=1)
C          NPROB                   PROBLEM NUMBER (PROG STOPS IF BLANK)
C          NP2                     NEND + 2
C          OOHX                    ONE DIVIDED BY HX
C          OOHXHY                  ONE DIVIDED BY HX TIMES HY
C          OOHX2                   ONE DIVIDED BY HX SQUARED
C          OOHX3                   ONE DIVIDED BY HX CUBED
C          OOHY                    ONE DIVIDED BY HY
C          OOHY2                   ONE DIVIDED BY HY SQUARED
C          OOHY3                   ONE DIVIDED BY HY CUBED
C          PBMX( , )               PLOTTING ARRAY FOR BMX OR BMBX
C          PBMY( , )               PLOTTING ARRAY FOR BMY OR BMBY
C          PBX( )                  AXIAL THRUST IN BEAM IN X DIRECTION
C          PBXN( ), PBYN( )        INPUT VALUES OF PBX AND PBY IN TABLE 6
C          PBY( )                  AXIAL THRUST IN BEAM IN Y DIRECTION
C          PBYP1( )                PBY AT J+1
C          POHXHY                  POISSONS RATIO DIVIDED BY HX TIMES HY
C          PMMAX                   LARGEST PRINCIPAL MOMENT (+ OR -)
C          PR                      POISSONS RATIO
C          PSIGO( , )              PLOTTING ARRAY FOR SIGO
C          PW( , )                 PLOTTING ARRAY FOR W
```

```
C         PX( )                     AXIAL THRUST IN SLAB IN X DIRECTION
C         PXN( ) , PYN( )           INPUT VALUES OF PX AND PY IN TABLE 6
C         PY( )                     AXIAL THRUST IN SLAB IN Y DIRECTION
C         PYP1( )                   PY AT J+1
C         Q( )                      APPLIED LOAD PER JOINT
C         QBMBX , QBMBY             LOAD ABSORBED IN BEAM BENDING
C         QBMX , QBMY               LOAD ABSORBED IN SLAB BENDING
C         QM( )                     APPLIED MULTIPLE LOAD PER JOINT
C         QMN( )                    INPUT VALUES OF LOAD IN TABLE 7
C         QN( )                     INPUT VALUE OF TRANSVERSE LOAD IN TABLE 3
C         QPBX , QPBY               LOAD ABSORBED DUE TO BEAM AXIAL THRUSTS
C         QPX , QPY                 LOAD ABSORBED DUE TO SLAB AXIAL THRUSTS
C         QRX , QRY                 LOAD ABSORBED DUE TO ROTATIONAL RESTRAINTS
C         QTMX , QTMY               LOAD ABSORBED IN TWISTING
C         QTX , QTY                 LOAD ABSORBED DUE TO APPLIED COUPLES
C         RCT                       ALPHANUMERIC, = REACTION
C         RDF                       REDUCTION FACTOR IN Y DIRECTION, 3D PLOT
C         REACT                     SUPPORT REACTION PER JOINT
C         ROCK                      REACTION OR STATICS CHECK
C         ROSC1 , ROSC2             FORMAT CONTROLS (BASED ON KROPT)
C         RX( )                     ROTATIONAL RESTRAINT IN X DIRECTION
C         RXN( ) , RYN( )           INPUT VALUES OF RX AND RY IN TABLE 4
C         RYM1( )                   ROTATIONAL RESTRAINT IN Y DIRECTION AT J-1
C         RYP1( )                   ROTATIONAL RESTRAINT IN Y DIRECTION AT J+1
C         R1( , ) THRU              RECURSION COEFFICIENTS AND MULTIPLIERS
C           R3( , )                         USED IN SOLUTION PACKAGE
C         S( )                      SPRING SUPPORT, VALUE PER JOINT
C         SDT2                      MOMENT MULTIPLIER FOR PLATE STRESS
C         SIGO                      LARGEST PRINCIPAL MOMENT OR STRESS
C         SLBX, SLBY                ALPHANUMERIC, = SLAB X, SLAB Y
C         SLOPE                     TAN OF GRID ANGLE, 3D PLOT
C         SN( )                     INPUT VALUES OF SUPPORT SPRINGS IN TABLE 3
C         SOM                       STRESS OR MOMENT FORMAT CONTROL
C         STACH                     STATICS CHECK
C         STAT                      ALPHANUMERIC, = STATICS
C         STEMP                     MAXIMUM VALUE OF STACH
C         STRS                      ALPHANUMERIC, = STRESS
C         SUMR                      SUMMATION OF SUPPORT REACTIONS
C         SUP                       ALPHANUMERIC, = SUPPORT
C         SWB                       SWITCH TO COMPUTE AND PRINT BEAM OUTPUT
C         SWS                       SWITCH TO COMPUTE AND PRINT SLAB OUTPUT
C         TEMX( ), TEMY( )          TEMPORARY PLOT ARRAY FOR SUBROUTINE ZOT 1
C         THETA                     MOHRS CIRCLE ANGLE BETWEEN X DIRECTION
C                                        AND PRINCIPAL MOMENT
C         THK                       THICKNESS OF SLAB FOR STRESS CALCULATIONS
C         TMX                       TWISTING MOMENT IN X DIRECTION  (ABOUT Y)
C         TMY                       NEGATIVE OF TMX
C         TX( )                     APPLED COUPLE IN X DIRECTION
C         TXN( ) , TYN( )           INPUT VALUES OF TX AND TY IN TABLE 4
C         TYM1( )                   APPLIED COUPLE IN Y DIRECTION AT J-1
C         TYP1( )                   APPLIED COUPLE IN Y DIRECTION AT J+1
C         U                         MAKES FILES 1 THRU 4 RANDOM INDEXED WITH       ISM
C                                        VARIABLE LENGTH RECORDS                   ISM
C         VEF                       VERTICAL EXPANSION FACTOR, 3D PLOT
C         W( )                      DEFLECTION AT EACH JOINT
C         WM1( )                    W AT J-1
C         WM2( )                    W AT J-2
C         WP1( )                    W AT J+1
C         WP2( )                    W AT J+2
C         WSUMM1                    WSUM1 AT J-1
C         WSUMM2                    WSUM2 AT J-2
C         WSUM1 , WSUM2             2ND DIFF OF DEFLECTIONS FOR MOMENTS
C         WSUM3                     2ND DIFF OF DEFLECTIONS FOR TWIST
C         XX( )                     INDEPENDENT VARIABLE FOR SUBROUTINE ZOT 1
C---------NOTATION FOR SUBROUTINE DATA
C         AN1( ) , AN2( )           IDENTIFICATION AND REMARKS ( ALPHA-NUM )
C         CH( )                     PLATE TWISTING STIFFNESS PER UNIT WIDTH
C         CHN( )                    INPUT VALUE OF CH IN TABLE 5
C         CHP1( )                   CH AT J+1
C         DX( ) , DY( )             SLAB BENDING STIFFNESSES PER UNIT WIDTH
C         DXM1( )                   DX AT J-1
C         DXN( ) , DYN( )           INPUT VALUES OF DX AND DY IN TABLE 3
C         DXP1( )                   DX AT J+1
C         DYM1( )                   DY AT J-1
C         DYP1( )                   DY AT J+1
C         FX( ) , FY( )             BEAM BENDING STIFFNESSES
C         FXN( ) , FYN( )           INPUT VALUES OF FX AND FY IN TABLE 3
C         FYM1( )                   FY AT J-1
C         FYP1( )                   FY AT J+1
C         I , I1 , I2               X DIRECTION INDEXES (ISTA+2)
C         IN13( ) THRU              -FROM- X STATION USED IN
C           IN18( )                      TABLES 3 THRU 8
C         IN23( ) THRU              -THRU- X STATION USED IN
C           IN28( )                      TABLES 3 THRU 8
C         IPRINT                    PRINTING CONTROL AT EACH J STEP
C         JN                        ACTUAL Y STA, EQUALS J-2
C         JN13( ) THRU              -FROM- Y STATION USED IN
C           JN19( )                      TABLES 3 THRU 9
C         JN23( ) THRU              -THRU- Y STATION USED IN
C           JN29( )                      TABLES 3 THRU 9
C         KASEP                     TABLE 8 PRINCIPAL MOMENT OR STRESS OPTION
C         KASEW                     TABLE 8 DEFLECTION OPTION
C         KASEX , KASEY             TABLE 8 BENDING MOMENT OPTIONS
C         L1                        KSHORT+3, USED FOR VARIABLE DIMENSIONING
C         ML                        MULTIPLE LOADING SWITCH
C         MXP3                      MX + 3
C         N                         INDEX FOR READING CARDS
C         NC13 THRU                 STARTING VALUE FOR READING CARDS,
C           NC19                         TABLES 3 THRU 9
C         NCT3 THRU NCT9            TOTAL NUMBER OF CARDS IN TABLES 3 THRU 9
C         NOE1 THRU NOE9            NUMBER OF DATA ERRORS IN TABLES 1 THRU 9
C         PBX( )                    AXIAL THRUST IN BEAM IN X DIRECTION
C         PBXN( ), PBYN( )          INPUT VALUES PBX AND PBY IN TABLE 6
C         PBY( )                    AXIAL THRUST IN BEAM IN Y DIRECTION
C         PBYP1( )                  PBY AT J+1
C         PX( )                     AXIAL THRUST IN SLAB IN X DIRECTION
C         PXN( ) , PYN( )           INPUT VALUES OF PX AND PY IN TABLE 6
C         PY( )                     AXIAL THRUST IN SLAB IN Y DIRECTION
C         PYP1( )                   PY AT J+1
C         Q( )                      APPLIED LOAD PER JOINT
C         QM( )                     APPLIED MULTIPLE LOAD PER JOINT
C         QMN( )                    INPUT VALUES OF LOAD IN TABLE 7
C         QN( )                     INPUT VALUE OF TRANSVERSE LOAD IN TABLE 3
C         RX( )                     ROTATIONAL RESTRAINT IN X DIRECTION
C         RXN( ) , RYN( )           INPUT VALUES OF RX AND RY IN TABLE 4
C         RYM1( )                   ROTATIONAL RESTRAINT IN Y DIRECTION AT J-1
C         RYP1( )                   ROTATIONAL RESTRAINT IN Y DIRECTION AT J+1
C         S( )                      SPRING SUPPORT, VALUE PER JOINT
C         SN( )                     INPUT VALUES OF SUPPORT SPRINGS IN TABLE 3
C         TX( )                     APPLED COUPLE IN X DIRECTION
C         TXN( ) , TYN( )           INPUT VALUES OF TX AND TY IN TABLE 4
C         TYM1( )                   APPLIED COUPLE IN Y DIRECTION
C         TYP1( )                   APPLIED COUPLE IN Y DIRECTION AT J+1
```

```
C---------NOTATION FOR SUBROUTINES SPLOT3 AND SPLOT4

C         BETA                        TRANSFORMED VALUE OF PLOTTING ARRAY
C         I , K , L                   DO LOOP INDICES
C         IS                          GENERAL INDEX
C         IOTA                        FIXED VALUE OF BETA
C         IR                          REMAINDER
C         IS                          INDEX OVER X
C         ISKP , ISKPF ,              FOUR-CHARACTER BLOCKS USED
C           ISKPT , ISKP2                 FOR SPACING
C         ISTA                        X STATION
C         IW                          WIDTH OF PLOT ( CHARACTERS )
C         IX                          IW DIVIDED BY 4
C         I1                          IN1S+2
C         I2                          IN2S+2
C         JS                          INDEX OVER Y
C         JSTA                        Y STATION
C         J1                          JN1S+2
C         J2                          JN2S+2
C         NEND                        LENGTH OF ARRAY TO BE PLOTTED
C         OMEGA                       MAXIMUM VALUE IN PLOT ARRAY
C         O2                          OMEGA FOR SECOND PLOT
C         SIGMA                       SCALE FACTOR
C         SPACE                       EMPTY DATA BLOCK FOR SKIPPING CHARACTERS
C         SYMS                        ARRAY OF FOUR LAST BLOCKS TO BE PLOTTED
C         SYMD                        LAST BLOCK TO BE PLOTTED
C         SYMD2                       SYMD FOR SECOND PLOT
C         S2                          SIGMA FOR SECOND PLOT
C         THETA                       MINIMUM VALUE IN PLOT ARRAY
C         T2                          THETA FOR SECOND PLOT
C         WIDTH                       FLOATED VALUE OF IW
C         X( )                        ARRAY TO BE PLOTTED
C         Y( )                        SECOND ARRAY TO BE PLOTTED
```

```
C---------NOTATION FOR SUBROUTINE DIM3

C         H                           HEIGHT OF PLOT
C         HX                          INCREMENT LENGTH IN X DIRECTION
C         HY                          INCREMENT LENGTH IN Y DIRECTION
C         I                           GENERAL INDEX
C         IDEL                        DISTANCE BETWEEN POINTS IN X DIRECTION
C         IGSW                        REF GRID TYPE SWITCH FOR 3-D PLOTS
C         IN                          NUMBER OF Y PARTITIONS
C         INIT                        INITIALIZATION SWITCH
C         IOFX                        OFFSET IN X DIRECTION
C         IOFY                        OFFSET IN Y DIRECTION
C         IPOP                        INPUT PLOT OPTION
C         IZ                          THE MASK
C         J                           GENERAL INDEX
C         JS                          ROTATION SWITCH
C         KN                          NUMBER OF X PARTITIONS
C         L1                          KSHORT+3, USED FOR VARIABLE DIMENSIONING
C         L2                          KLONG+3, USED FOR VARIABLE DIMENSIONING
C         MX                          NUMBER OF INCREMENTS IN X DIRECTION
C         MXP2                        MX+2
C         MY                          NUMBER OF INCREMENTS IN Y DIRECTION
C         MYP2                        MY+2
C         NPROB                       PROBLEM NUMBER (PROG STOPS IF BLANK)
C         PLTMIN                      MINIMUN Y COORD OF MASK
C         RDF                         REDUCTION FACTOR IN Y DIRECTION, 3D PLOT
C         SLOPE                       TAN OF GRID ANGLE, 3D PLOT
C         START                       INITIALIZATION VARIABLE FOR PLOT FILE
C         VEF                         VERTICAL EXPANSION FACTOR
C         VSF                         VERTICAL SCALE FACTOR, 3D PLOT
C         W( )                        WIDTH OF PLOT
C         XS                          X COORD OF RELATIVE ORIGIN FOR GRID 1
C         XX                          TEMP DISTANCE ALONG SLOPE
C         YS                          Y COORD OF RELATIVE ORIGIN FOR GRID 1
C         Z                           FUNCTION TO BE PLOTTED
```

```
C---------NOTATION FOR SUBROUTINE ZOT1

C         ID                          VARIABLE OR ARRAY CONTAINING TITLE OF PLOT
C         INC                         STEP SIZE FOR SCANNING PLOT ARRAY
C         IS                          PLOTTING SYMBOL SELECTOR
C         IROLL                       PLOT SWITCH FOR FRAME POSITION CONTROL
C         IT1 , IT2                   SWITCHES FOR POSITIONING PLOTTER
C         LOP                         LINE OR POINT PLOT OPTION
C         MC                          NUMBER OF CHARACTERS IN PLOT TABLES
C         MOP                         MICROFILM OR PAPER PLOT OPTION
C         NC                          TEMPORARY FOR MC
C         NP                          NUMBER OF POINTS TO BE PLOTTED
C         X                           X AXIS POSITIONER
C         XF                          ARRAY CONTAINING THE X - COORDINATES
C         XF1 , XF2                   TEMPORARIES FOR XF
C         XL                          X AXIS LENGTH IN INCHES
C         XM                          X AXIS POSITION
C         XO                          ORIENTATION AT X AXIS
C         Y                           Y AXIS POSITIONER
C         YF                          ARRAY CONTAINING THE Y - COORDINATES
C         YF1 , YF2                   TEMPORARIES FOR YF
C         YL                          Y AXIS LENGTH IN INCHES
C         YM                          Y AXIS POSITION
C         YO                          ORIENTATION AT Y AXIS
```

```
C---------NOTATION FOR SUBROUTINE GRD

C         IDEL                        DISTANCE BETWEEN POINTS IN X DIRECTION
C         IGSW                        REF GRID TYPE SWITCH FOR 3-D PLOTS
C         IOFX, IOFY                  OFFSETS IN X AND Y DIRECTIONS
C         LOP                         LINE OPTION ( 1 = SOLID, 0 = DASHED )
C         MX, MY                      NUMBER OF INCREMENTS IN X AND Y DIRECTIONS
C         SLOPE                       SLOPE OF Y DIRECTION ARM RELATIVE TO X
C         XS, YS                      COORDS OF RELATIVE ORIGIN FOR GRID 1
```

```
C---------NOTATION FOR SUBROUTINE  TD2

C          AT                         ABSCISSA OF REAL POINT
C          C                          L / 100
C          DEL                        IDEL / 100
C          DELX                       X SHIFT AT ORIGIN
C          DELY                       Y SHIFT AT ORIGIN
C          I                          GENERAL INDEX
C          IDEL                       DISTANCE BETWEEN POINTS IN X DIRECTION
C          IEND                       LENGTH OF MASK
C          II                         NU - 1
C          IN                         NUMBER OF Y PARTITIONS
C          INIT                       INITIALIZATION SWITCH
C          IOFX                       OFFSET IN X DIRECTION
C          IOFY                       OFFSET IN Y DIRECTION
C          IPEN                       PEN STATUS
C          IPOV                       POV * 100
C          IR                         DISTANCE ABOVE THE CURRENT HORIZON
C          IZ                         THE MASK
C          J0                         ROTATION SWITCH
C          K                          INDEX OVER KN
C          KL                         KN - 1
C          KMP                        K + MP (POINT CONTIGUOUS TO POINT K)
C          KN                         NUMBER OF X PARTITIONS
C          L                          ABSCISSA
C          LL                         INDEX OVER IDEL
C          MP                         OPTIMIZING SWEEP SWITCH
C          NU                         ROW NUMBER
C          OFX                        IOFX / 100
C          OFY                        IOFY / 100
C          PLTMIN                     MINIMUM Y COORD OF MASK
C          POV                        MINIMUM VALUE OF Z, SCALED, IN INCHES
C          SI                         SLOPE OF CURRENT X SLICE
C          VSF                        VERTICAL SCALE FACTOR
C          X                          X COORD
C          XH                         TEMP VECTOR FOR VERTICAL SLICES
C          Y                          Y COORD
C          YH                         TEMP VECTOR FOR VERTICAL SLICES
C          Z                          FUNCTION TO BE PLOTTED
```

```
C---------NOTATION FOR SUBROUTINE EDGE1

C          DEL                        IDEL * 0.01
C          I                          DO LOOP INDEX TO INCREMENT IN X DIRECTION
C          IDEL                       DISTANCE BETWEEN POINTS IN X DIRECTION
C          IOFX, IOFY                 OFFSETS IN X AND Y DIRECTIONS
C          MX, MY                     NUMBER OF INCREMENTS IN X AND Y DIRECTIONS
C          OFX, OFY                   HALF OF X AND Y TICK MARK LENGTHS
C          SIZE                       TICK MARK SIZE CONTROL
C          X, Y                       COORDS OF CENTER OF TICK MARK  ( GLOBAL )
C          X0, Y0                     COORDS OF RELATIVE ORIGIN FOR GRID 1
```

```
C---------NOTATION FOR SUBROUTINE CRNR1

C          IDEL                       DISTANCE BETWEEN POINTS IN X DIRECTION
C          IOFX, IOFY                 OFFSETS IN X AND Y DIRECTIONS
C          MX, MY                     NUMBER OF INCREMENTS IN X AND Y DIRECTIONS
C          SIZE                       TICK MARK SIZE CONTROL
C          X, Y                       COORDS OF CENTER OF TICK MARK  ( GLOBAL )
C          X0, Y0                     COORDS OF RELATIVE ORIGIN FOR GRID 1
```

```
C---------NOTATION FOR SUBROUTINE MESH1

C          DEL                        IDEL * 0.01
C          I, J                       DO LOOP INDEXES
C          IDEL                       DISTANCE BETWEEN POINTS IN X DIRECTION
C          IOFX, IOFY                 OFFSETS IN X AND Y DIRECTIONS
C          MP                         REVERSING SWITCH
C          MX, MY                     NUMBER OF INCREMENTS IN X AND Y DIRECTIONS
C          MXP1                       MX PLUS ONE
C          MYP1                       MY PLUS ONE
C          OFX, OFY                   HALF OF X AND Y TICK MARK LENGTHS
C          SIZE                       TICK MARK SIZE CONTROL
C          X, Y                       COORDS OF CENTER OF TICK MARK  ( GLOBAL )
C          XX                         X PLOTTER COORDINATE
C          X0, Y0                     COORDS OF RELATIVE ORIGIN FOR GRID 1
```

```
C---------NOTATION FOR SUBROUTINE GRID1

C          I                          GENERAL INDEX
C          IDEL                       DISTANCE BETWEEN POINTS IN X DIRECTION
C          IM1                        I - 1
C          IN                         NUMBER OF Y PARTITIONS
C          IN1                        IN + 1
C          IOFX                       OFFSET IN X DIRECTION
C          IOFY                       OFFSET IN Y DIRECTION
C          KN                         NUMBER OF X PARTITIONS
C          KN1                        KN + 1
C          LOP                        LINE OPTION ( 1 = SOLID, 0 = DASHED )
C          MP                         OPTIMIZING SWEEP SWITCH
C          XX                         TEMP CONSTANT
C          X0, Y0                     COORD OF RELATIVE ORIGIN
C          X1, Y1                     COORD OF BEGINNING POINT
C          X2, Y2                     COORD OF ENDING POINT
```

```
C---------NOTATION FOR SUBROUTINE MARK2

C          ARM                 LENGTH OF ARM OF TICK MARK
C          SIZE                TICK MARK SIZE CONTROL
C          SLOPE               SLOPE OF Y DIRECTION ARM RELATIVE TO X
C          STEP                0.2 IN.
C          X, Y                COORDS OF CENTER OF TICK MARK  ( GLOBAL )
C          XX, YY              PLOTTER COORDINATES  ( GLOBAL )
C          X1, Y1              COORDS OF TICK MARK ENDS, REL TO (X, Y)
```

```
C---------NOTATION FOR SUBROUTINE  DASH 2

C          DIS                 DIST BETWEEN POINTS
C          I                   GENERAL INDEX
C          INC                 NO OF INCREMENTS BETWEEN POINTS
C          INC2                INC+2
C          IP                  PEN STATUS
C          IS                  PEN STATUS SWITCH
C          LOP                 LINE OPTION ( 1 = SOLID, 0 = DASHED )
C          X, Y                COORD OF PEN POSITION
C          XD, YD              PROJECTED LENGTH OF DASH
C          XT, YT              TEMP COORD FOR FIRST DASH
C          X1, Y1              COORD OF BEGINNING POINT
C          X2, Y2              COORD OF ENDING POINT
```

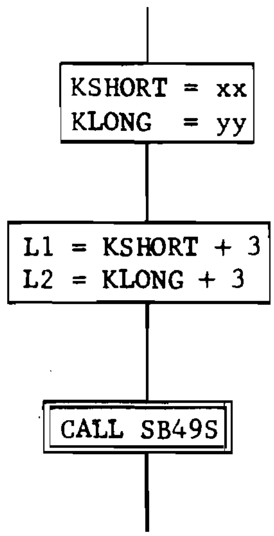
```
C---------NOTATION FOR SUBROUTINE FRIPS

C          A( ,1)              CONTINUITY COEFFICIENT
C          AM1( ,1)            CONTINUITY COEFFICIENT A( ,1) AT J-1
C          AM2( ,1)            CONTINUITY COEFFICIENT A( ,1) AT J-2
C          ATM( ,1)            TEMP STORAGE FOR A( ,1) RECURSION COEFF
C          CC( ,5)             COEFFS IN STIFFNESS MATRIX
C          CM( )               PLATE TWISTING STIFFNESS PER UNIT WIDTH
C          CMP1( )             CM AT J+1
C          DT                  DD TRANSPOSE AT J
C          DX( ) , DY( )       SLAB BENDING STIFFNESSES PER UNIT WIDTH
C          DXM1( )             DX AT J-1
C          DXP1( )             DX AT J+1
C          DYM1( )             DY AT J-1
C          DYP1( )             DY AT J+1
C          EE( ,1)             COEFF IN STIFFNESS MATRIX
C          ET1                 EE TRANSPOSE AT J-1
C          ET2                 EE TRANSPOSE AT J-2 OR AA AT J
C          FF( ,1)             COEFF IN LOAD VECTOR
C          FX( ) , FY( )       BEAM BENDING STIFFNESSES
C          FYM1( )             FY AT J-1
C          FYP1( )             FY AT J+1
C          I                   X DIRECTION INDEX (JSTA+2)
C          ID1 THRU ID4        RANDOM INDEXES FOR FILES 1 THRU 4           IBM
C          IIII                TEMPORARY INDEX
C          ITMPP               TEMPORARY PRINT SWITCH FOR DEBUGGING
C          J                   Y DIRECTION INDEX (JSTA+2)
C          JJ                  J FOR SUBROUTINE MATRIX
C          J1                  J2-1                                        IBM
C          J2                  TWICE THE INDEX  J                          IBM
C          K , L               DO LOOP INDEXES
C          L1                  KSHORT+3, USED FOR VARIABLE DIMENSIONING
C          L2                  KLONG+3, USED FOR VARIABLE DIMENSIONING
C          ML                  MULTIPLE LOADING SWITCH
C          MP                  STARTING VALUE FOR MAIN DO LOOP
C          NK                  ORDER OF SUBMATRICES
C          NKK                 TEMPORARY INDEX FOR NK
C          NL                  MATRIX ORDER OF OVERALL COEFFICIENT MATRIX
C          NLM2                NL MINUS 2
C          N1                  BAND WIDTH OF EE
C          N2                  BAND WIDTH OF DD
C          N3                  BAND WIDTH OF CC
C          PBX( )              AXIAL THRUST IN BEAM IN X DIRECTION
C          PBY( )              AXIAL THRUST IN BEAM IN Y DIRECTION
C          PBYP1               PBY AT J+1
C          PX( )               AXIAL THRUST IN SLAB IN X DIRECTION
C          PY( )               AXIAL THRUST IN SLAB IN Y DIRECTION
C          PYP1( )             PY AT J+1
C          Q( )                APPLIED LOAD PER JOINT
C          QM( )               APPLIED MULTIPLE LOAD PER JOINT
C          RX( )               ROTATIONAL RESTRAINT IN X DIRECTION
C          RYM1( )             ROTATIONAL RESTRAINT IN Y DIRECTION AT J-1
C          RYP1( )             ROTATIONAL RESTRAINT IN Y DIRECTION AT J+1
C          R1( , ) THRU        RECURSION COEFFICIENTS AND MULTIPLIERS
C            R3( , )                USED IN SOLUTION PACKAGE
C          S( )                SPRING SUPPORT, VALUE PER JOINT
C          TX( )               APPLIED COUPLE IN X DIRECTION
C          TYM1( )             APPLIED COUPLE IN Y DIRECTION AT J-1
C          TYP1( )             APPLIED COUPLE IN Y DIRECTION AT J+1
C          WW( , )             COMPLETE ARRAY OF W VECTORS NEEDED
C                                   FOR PLOTTING
```

APPENDIX 3

FLOW DIAGRAMS AND LISTING OF COMPUTER PROGRAM SLAB 49

GENERAL FLOW DIAGRAM FOR SLAB 49

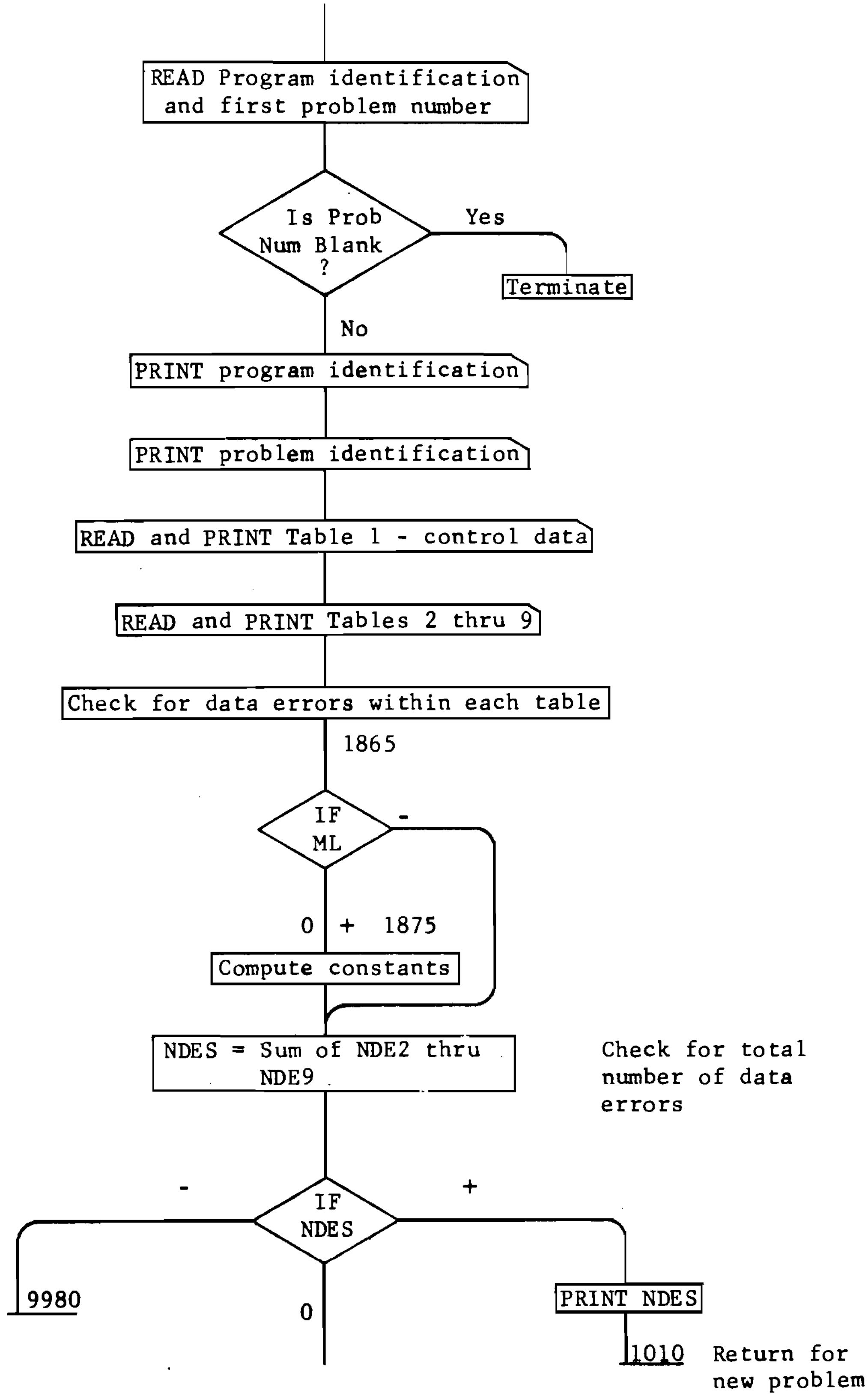
All variables are dimensioned in this main driving routine. For different sized problems only the cards labeled RE-DIMEN columns 73 through 80 must be changed. The main program and all subroutines are variably dimensioned.

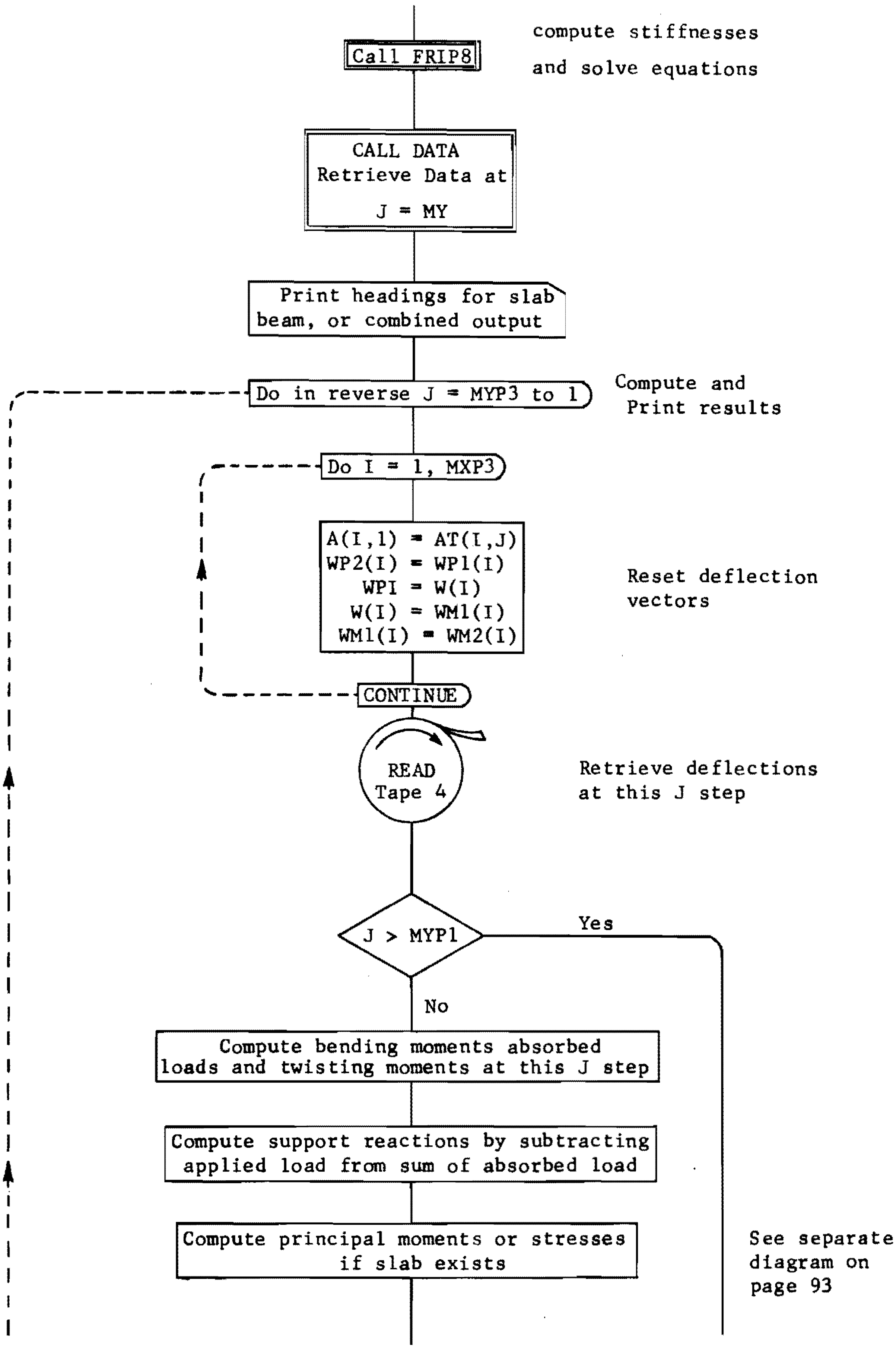


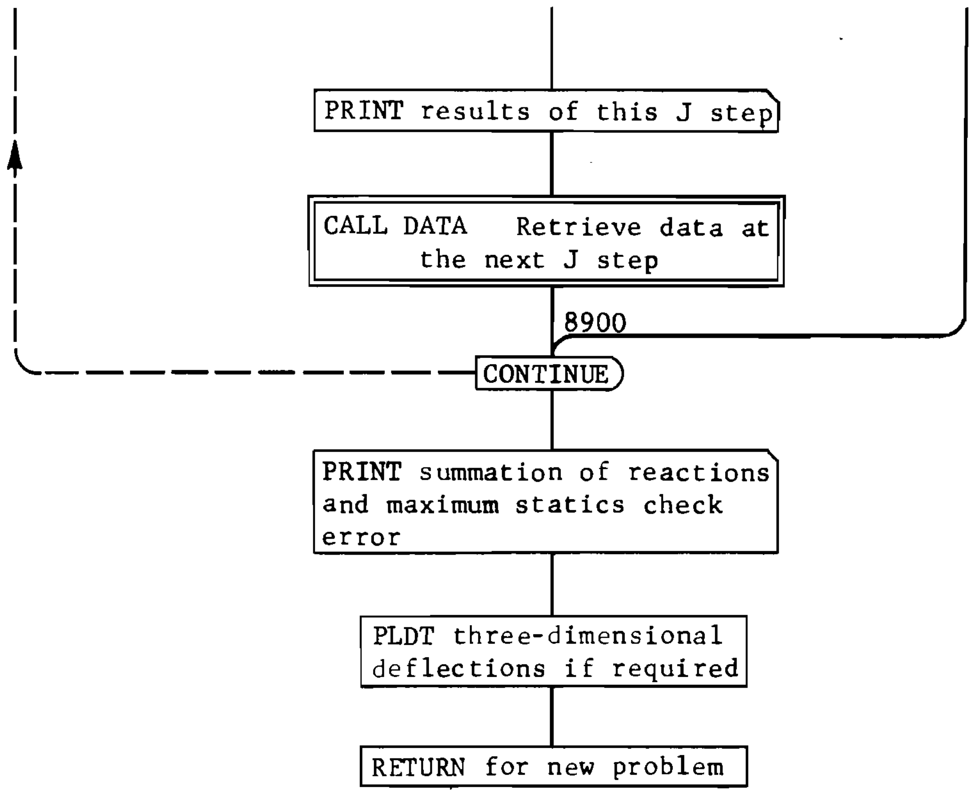
```
C      PROGRAM  SLAB49  ( INPUT, OUTPUT, TAPE1, TAPE2, TAPE3, TAPE4 )    02NOOCDC
C
C-----FOR OTHER PROBLEM SIZES, ONLY THIS DRIVER NEED BE RE-DIMENSIONED.
C      FOR EXAMPLE, AA(S+3,1), CC(S+3,5), DD(S+3,3), R1(S+3,S+3),
C      DX(S+3), PW(S+3,L+3), KSHORT = S, AND KLONG = L   WHERE S AND L
C      ARE THE MAXIMUN X AND Y SIZES OF THE PROBLEM.
C
C-----THIS PROGRAM IS NOW DIMENSIONED TO SOLVE A  24 BY 48  GRID.        RE-DIMEN
C
C-----THIS PROGRAM WILL OPERATE ON CDC 6000 SERIES OR IBM G-LEVEL
C      SYSTEMS.  THE CARDS NEEDED FOR CDC OPERATION HAVE A C IN COLUMN
C      1 AND CDC IN COLUMN 78 THRU 80.  THOSE CARDS NEEDED FOR IBM
C      OPERATION ARE TAGGED WITH IBM.  WHEN CONVERTING FROM ONE SYSTEM
C      TO ANOTHER, THOSE CARDS NOT NEEDED SHOULD BE RETAINED AND NULLED
C      WITH AN ADDED C.  SOME CARDS NOW TAGGED IBM ARE PUNCHED IN THE
C      NECESSARY EBCDIC CODE.  THE CDC CARDS AND THE REMAINDER OF THE
C      CARDS WHICH ARE NOT SPECIALLY TAGGED ARE PUNCHED IN BCD WHICH
C      IS COMPATIBLE WITH BOTH SYSTEMS.  ASA FORTRAN IS USED THRUOUT.
C
       DOUBLE PRECISION  AA, BB, CC, DD, EE, FF, AT,   A, AM1, AM2,       13MY1IBM
3     1                  R1, R2, R3,  W, WM1, WM2, WP1, WP2,  PW           13MY1IBM
       DIMENSION AA( 27 , 1  ),      BB( 27 , 1  ),      CC( 27 , 5  ),  RE-DIMEN
     1           DD( 27 , 3  ),      EE( 27 , 1  ),      FF( 27 , 1  ),  RE-DIMEN
     2           AT( 27 , 1  ),                                          RE-DIMEN
     3            A( 27 , 1  ),     AM1( 27 , 1  ),     AM2( 27 , 1  ),  RE-DIMEN
     4           R1( 27 , 27 ),      R2( 27 , 27 ),      R3( 27 , 27 )   RE-DIMEN
       DIMENSION DX( 27 ),         DXM1( 27 ),         DXP1( 27 ),       RE-DIMEN
     1           DY( 27 ),         DYM1( 27 ),         DYP1( 27 ),       RE-DIMEN
     2           FX( 27 ),                                               RE-DIMEN
     3           FY( 27 ),         FYM1( 27 ),         FYP1( 27 ),       RE-DIMEN
     4            Q( 27 ),            S( 27 ),                           RE-DIMEN
     5           RX( 27 ),         RYM1( 27 ),         RYP1( 27 ),       RE-DIMEN
     6           TX( 27 ),         TYM1( 27 ),         TYP1( 27 ),       RE-DIMEN
     7           CH( 27 ),         CHP1( 27 ),                           RE-DIMEN
     8           PX( 27 ),           PY( 27 ),         PYP1( 27 ),       RE-DIMEN
     9          PBX( 27 ),          PBY( 27 ),        PBYP1( 27 ),       RE-DIMEN
     A           QM( 27 )                                                RE-DIMEN
       DIMENSION  W( 27 ),          WM1( 27 ),          WM2( 27 ),       RE-DIMEN
     1                              WP1( 27 ),          WP2( 27 ),       RE-DIMEN
     2          BMX( 27 ),         BMBX( 27 ),                           RE-DIMEN
     3        BMYM1( 27 ),          BMY( 27 ),        BMYP1( 27 ),       RE-DIMEN
     4       BMBYM1( 27 ),         BMBY( 27 ),       BMBYP1( 27 ),       RE-DIMEN
     5           PW( 27 , 51 ),    PBMX( 27 , 51 ),    PBMY( 27 , 51 ),  RE-DIMEN
     6       PSIGO( 27 , 51 )                                            RE-DIMEN
C-----THE FOLLOWING 5 CARDS PERTAIN TO DIRECT ACCESS FILES AND MUST BE   14JA2IBM
C      CHANGED WHEN PROGRAM IS REDIMENSIONED. FOR EXAMPLE, THE DEFINE    14JA2IBM
C      FILE CARD PARAMETERS ARE  2*(L+3), 2*(S+3)**2), U, ID1 THRU ID4.  14JA2IBM
       COMMON /DA/ ID1, ID2, ID3, ID4                                    14JA2IBM
       DEFINE FILE 1 ( 102, 1458, U, ID1 )                               RE-D IBM
       DEFINE FILE 2 ( 102, 1458, U, ID2 )                               RE-D IBM
       DEFINE FILE 3 ( 102, 1458, U, ID3 )                               RE-D IBM
       DEFINE FILE 4 ( 102, 1458, U, ID4 )                               RE-D IBM
             KSHORT = 24                                                 RE-DIMEN
             KLONG = 48                                                  RE-DIMEN
             L1 = KSHORT + 3                                             05JA8
             L2 = KLONG + 3                                              12MYO
C
       CALL SB49S          (   AA,   BB,   CC,   DD,   EE,   FF,   AT,   22AG9
     1                          A,  AP1,  AM2,   R1,   R2,   R3,         22AG9
     2                         DX, DXM1, DXP1,   DY, DYM1, DYP1,   ,     10OC9
     3                         FX,   FY, FYM1, FYP1,    Q,    S,         10OC9
     4                         RX, RYM1, RYP1,   TX, TYM1, TYP1,   CH,   10OC9
     5                       CHP1,   PX,   PY, PYP1,  PBX,  PBY,PBYP1,   10OC9
     6                         QM,    W,  WM1,  WM2,  WP1,  WP2,         10OC9
     7                        BMX, BMBX,BMYM1,  BMY,BMYP1,               10OC9
     8                     BMBYM1, BMBY,     BMBYP1,                     21JA1
     9                         PW, PBMX, PBMY,PSIGO,   L1,   L2  )       17FE1
       END                                                               05JA9
```

SUBROUTINE SB49S

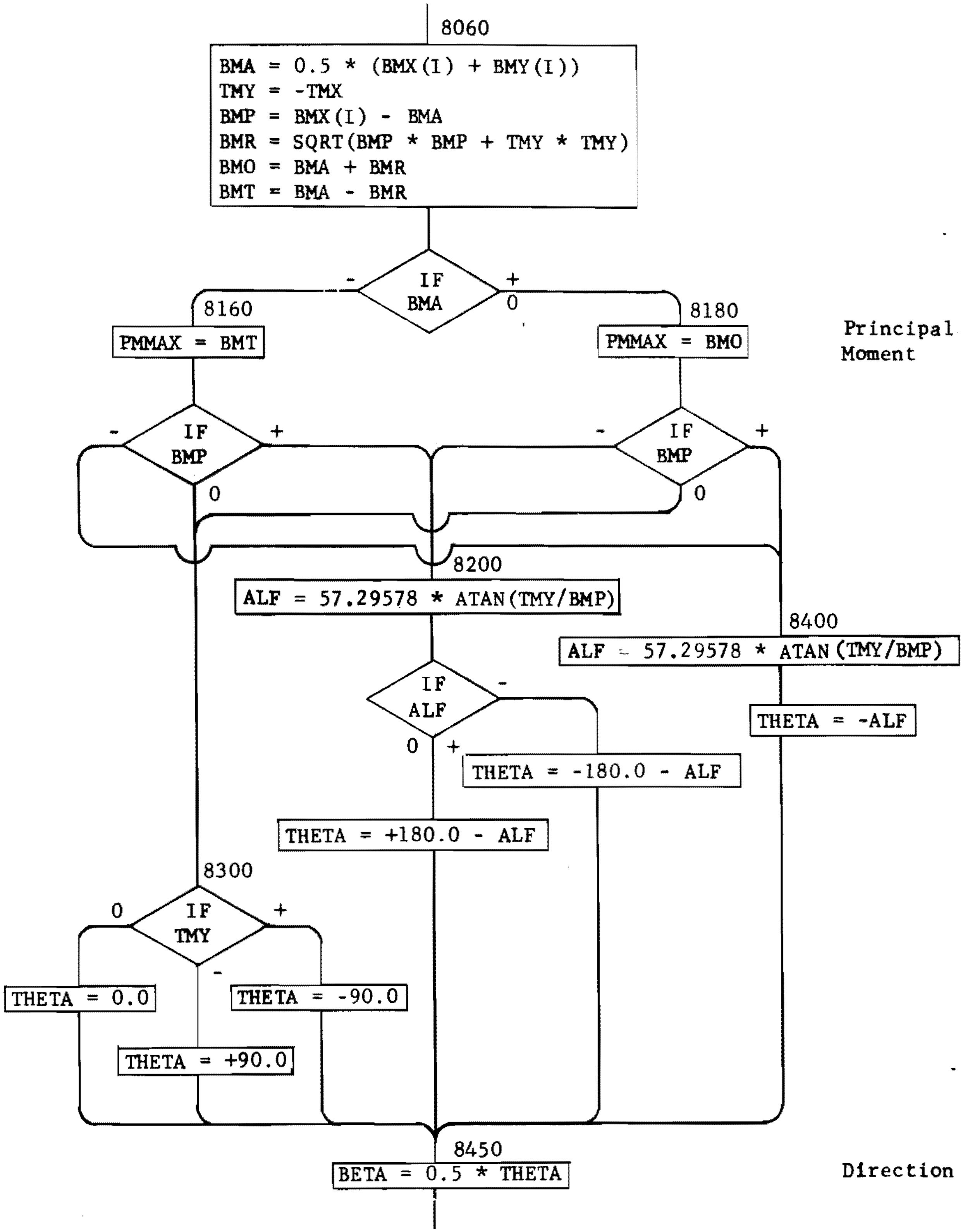
This is the main computational program




compute stiffnesses
and solve equations
Call FRIP8
CALL DATA
Retrieve Data at
J = MY
Print headings for slab
beam, or combined output
Do in reverse J = MYP3 to 1
Compute and
Print results
Do I = 1, MXP3
A(I,1) = AT(I,J)
WP2(I) = WP1(I)
WPI = W(I)
W(I) = WM1(I)
WM1(I) = WM2(I)
Reset deflection
vectors
CONTINUE
READ
Tape 4
Retrieve deflections
at this J step
J > MYP1
Yes
No
Compute bending moments absorbed
loads and twisting moments at this J step
Compute support reactions by subtracting
applied load from sum of absorbed load
Compute principal moments or stresses
if slab exists
See separate
diagram on
page 93


PRINT results of this J step
CALL DATA Retrieve data at the next J step
8900
CONTINUE
PRINT summation of reactions and maximum statics check error
PLDT three-dimensional deflections if required
RETURN for new problem

Compute Principal Moment and Direction



```
      SUBROUTINE SB49S   (   AA,   BB,   CC,   DD,   EE,   FF,   AT,    21JA1
     1                        A,  AM1,  AM2,   R1,   R2,   R3,           22AG9
     2                       DX, DXM1, DXP1,   DY, DYM1, DYP1,           10OC9
     3                       FX,   FY, FYM1, FYP1,    Q,    S,           10OC9
     4                       RX, RYM1, RYP1,   TX, TYM1, TYP1,   CH,     10OC9
     5                     CHP1,   PX,   PY, PYP1,  PBX,  PBY,PBYP1,     10OC9
     6                       QM,    W,  WM1,  WM2,  WP1,  WP2,           10OC9
     7                     BMX, BMBX,BMYM1,  BMY,BMYP1,                  10OC9
     8                   BMBYM1, BMBY,     BMBYP1,                       21JA1
     9                       PW, PBMX, PBMY,PSIGO,   L1,   L2  )         17FE1
C
    1 FORMAT  (  52H      PROGRAM SLAB 49  -TEX HWY DEPT DECK-  MATLOCK,11NO1
     1                        33HPANAK, ENDRES  REV DATE 29 FEB 72  REVISED
     B//  )                                                              21JA1
C
      DOUBLE PRECISION  AA, BB, CC, DD, EE, FF, AT,   A, AM1, AM2,       13MY1IBM
     1                  R1, R2, R3,  W, WM1, WM2, WP1, WP2,  PW          13MY1IBM
      DOUBLE PRECISION  PDHXHY, HYDHX3, DDHX3, DDHX2, DDHX,              13MY1IBM
     1                  DDHXHY, HXDHY3, DDHY3, DDHY2, DDHY, CRD          13MY1IBM
      DOUBLE PRECISION  WSUM1, WSUMM2, WSUM3, HYDHX, HXDXY,              13MY1IBM
     +                   QBMX,  QBMY,  QPX,  QPY, QTMX, QTMY, QTX, QTY,  13MY1IBM
     +                  QBMBX, QBMBY, QPBX, QPBY,  QRX,  QRY, REACT,     13MY1IBM
     +                  SUMR, TMX, BMA, TMY, BMP, BMR, BMD, BMT,         13MY1IBM
     +                  ALF, SDT2, PMMAX, THK, STACH, SUMR, ROCK         13MY1IBM
      DOUBLE PRECISION   ROSC1, ROSC2, SOM, SUP, STAT, STRS, RCT, CHK,   23FE1IBM
     1          BMCM, SLBX, SLBY, BLNK, BEMX, BEMY, ITEST, NPROB, KPROB  23FE1IBM
      DOUBLE PRECISION ID1, ID2, ID3, ID4                                30NO1IBM
      DOUBLE PRECISION STEMP                                             01DE1IBM
      DIMENSION AA( L1 , 1  ),      BB( L1 , 1  ),      CC( L1 , 5  ),   26SE8
     1          DD( L1 , 3  ),      EE( L1 , 1  ),      FF( L1 , 1  ),   26SE8
     2          AT( L1 , 1  ),                                           26SE8
     3           A( L1 , 1  ),     AM1( L1 , 1  ),     AM2( L1 , 1  ),   26SE8
     4          R1( L1 , L1 ),      R2( L1 , L1 ),      R3( L1 , L1 )    22AG9
      DIMENSION DX( L1 ),         DXM1( L1 ),         DXP1( L1 ),        10OC9
     1          DY( L1 ),         DYM1( L1 ),         DYP1( L1 ),        10OC9
     2          FX( L1 ),                                                10OC9
     3          FY( L1 ),         FYM1( L1 ),         FYP1( L1 ),        10OC9
     4           Q( L1 ),            S( L1 ),                            10OC9
     5          RX( L1 ),         RYM1( L1 ),         RYP1( L1 ),        10OC9
     6          TX( L1 ),         TYM1( L1 ),         TYP1( L1 ),        10OC9
     7          CH( L1 ),         CHP1( L1 ),                            10OC9
     8          PX( L1 ),           PY( L1 ),         PYP1( L1 ),        10OC9
     9         PBX( L1 ),          PBY( L1 ),        PBYP1( L1 ),        10OC9
     A          QM( L1 )                                                 10OC9
      DIMENSION  W( L1 ),          WM1( L1 ),          WM2( L1 ),        10OC9
     1                             WP1( L1 ),          WP2( L1 ),        10OC9
     2         BMX( L1 ),         BMBX( L1 ),                            10OC9
     3       BMYM1( L1 ),          BMY( L1 ),        BMYP1( L1 ),        10OC9
     4      BMBYM1( L1 ),         BMBY( L1 ),       BMBYP1( L1 ),        17FE1
     5          PW( L1 , L2 ),    PBMX( L1 , L2 ),    PBMY( L1 , L2 ),   17FE1
     6       PSIGO( L1 , L2 )                                            17FE1
      DIMENSION  TEMX (300), TEMY (300), XX(300)                         11FE1
      DIMENSION IBUF(500)                                                06DE1IBM
C
      COMMON / ZOT / LOP, MC, IROLL, MOP                                 15JE0
C     COMMON /PLOT/ I1, I2, J1, J2                                       25NO9CDC
      COMMON /PLT / I1, I2, J1, J2                                       01DE1IBM
      COMMON /CARDS/ AN1(40), AN2(18),                                   21JA1
     1          IN13(300), JN13(300), IN23(300), JN23(300), DXN(300),    23SE0
     2           DYN(300),  FXN(300),  FYN(300),   QN(300),  SN(300),    23SE0
     3          IN14( 50), JN14( 50), IN24( 50), JN24( 50),              23SE0
     4           RXN( 50),  RYN( 50),  TXN( 50),  TYN( 50),              23SE0
     5          IN15(100), JN15(100), IN25(100), JN25(100), CHN(100),    23SE0
     6          IN16( 50), JN16( 50), IN26( 50), JN26( 50),              23SE0
     7           PXN( 50),  PYN( 50), PBXN( 50), PBYN( 50),              23SE0
     8          IN17(100), JN17(100), IN27(100), JN27(100), QMN(100),    17FE1
     9          IN18( 10), JN18( 10), IN28( 10), JN28( 10),              17FE1
     A        KASEM( 10),KASEX( 10),KASEY( 10),KASEP( 10),               17FE1
     B                    JN19( 10),            JN29( 10)                17FE1
      COMMON /TABLE/NC13, NCT3, NC14, NCT4, NC15, NCT5, NC16, NCT6,      10OC9
     1              NC17, NCT7, NC18, NCT8, NC19, NCT9                   17FE1
      COMMON /STIFF/PDHXHY, HYDHX3, DDHX3, DDHX2, DDHX,                  10OC9
     1              DDHXHY, HXDHY3, DDHY3, DDHY2, DDHY, CRD(5)           24MY1
      COMMON / RI / MXP3, MYP3, NF, ITMPP                                23DE0
```

```
C
      DATA  SUP / 7HSUPPORT /, STAT / 7HSTATICS /, STRS / 6HSTRESS  /,  08FE1
     1       RCT / 8HREACTION/,  CHK / 8H CHECK  /, BMOM / 6HMOMENT  /,  08FE1
     2      SLBX / 6HSLAB X  /, SLBY /  6HSLAB Y  /,BLNK / 6H        /,  08FE1
     3      BEMX / 6HBEAM X  /, BEMY / 6HBEAM Y  /,  ITEST / 5H      /   08FE1
C     DATA ID1, ID2, ID3, ID4  /8HDEFLECTN, 8HBN MOM X,                  15JE0CDC
C    1                          8HBN MOM Y,  8H   SIGO      /            15JE0CDC
C
    6 FORMAT ( )                                                         04MY3
   11 FORMAT (     5H1    , 80X, 10H1-----TRIM )                         17FE1
   12 FORMAT (  20A4   )                                                 18MY0
   13 FORMAT (     5X, 20A4 )                                            18MY0
   14 FORMAT ( A5, 5X, 17A4, A2       )                                  21JA1
   15 FORMAT (///10H      PROB , /5X, A5, 5X, 17A4, A2 )                 07DE0
C  16 FORMAT %///17H      PROB %CONTDL, /5X, A5, 5X, 17A4, A2 L          07DE0CDC
   16 FORMAT (///17H      PROB (CONTD), /5X, A5, 5X, 17A4, A2 )          06DE1IBM
   19 FORMAT (///50H      KEEP RUN TIME RECORDS FOR FUTURE ESTIMATES OF  21JA1
     1             31H PARENT AND OFFSPRING RUN TIMES     )              21JA1
   20 FORMAT ( 8I5, 5X,I5,25X, I5,/, 8I5,5X,3I5,I2,I3,3F5.0 )            06DE1
   21 FORMAT ( 2I5, 10X,    4E10.3 )                                     21JA1
   33 FORMAT (  4( 2X, I3 ),   6E10.3 )                                  19AG8
   43 FORMAT ( 4( 2X, I3 ), 20X,    4E10.3 )                             21JA1
   53 FORMAT ( 4( 2X, I3 ),    E10.3 )                                   21JA1
   63 FORMAT ( 4( 2X, I3 ), 20X,   4E10.3 )                              21JA1
   73 FORMAT ( 4( 2X, I3 ), 40X,   E10.3 )                               21JA1
   83 FORMAT ( 4( 2X, I3 ), 4( 4X, I1 ) )                                03FE1
   93 FORMAT ( 2I10 )                                                    06OC0
C 100 FORMAT % //30H      TABLE 1.  CONTROL DATA     ,                    21JA1CDC
  100 FORMAT ( //30H      TABLE 1.  CONTROL DATA     ,                    06DE1IBM
     1    / 55X, 20H      TABLE NUMBER      ,                            21JA1
     2    / 40X, 45H        2    3    4    5    6    7    8    9  ,      21JA1
C    3   //  5X, 40H      KEEP FROM PRECEDING PROBLEM %1#YESL  , 8I5,    21JA1CDC
     3   //  5X, 40H      KEEP FROM PRECEDING PROBLEM (1=YES)  , 8I5,    06DE1IBM
     4    /  5X, 40H      NUM CARDS INPUT THIS PROBLEM         , 8I5,    21JA1
     5   //  5X, 25H      MULTIPLE LOAD OPTION  ,  15X, I5,              21JA1
     6    /  5X, 25H      STATICS CHECK OPTION  ,  15X, I5,              21JA1
     7    /  5X, 25H      PRIN STRESS OPTION    ,  15X, I5,              21JA1
     8    /  5X, 25H      PROFILE PLOT OPTION   ,  15X, I5,              21JA1
C    9    /  5X, 25H      3-D PLOT OPTION       ,  15X, I5     L         21JA1CDC
     9    /  5X, 25H      3-D PLOT OPTION       ,  15X, I5     )         06DE1IBM
  200 FORMAT ( //24H      TABLE 2.  CONSTANTS )                          16AG8
  201 FORMAT ( / 45H           NUMBER OF INCREMENTS IN X DIRECTION     , 21JA1
     1                 35X, I5,                                          21JA1
     2    / 10X, 35HNUMBER OF INCREMENTS IN Y DIRECTION     , 35X, I5,   21JA1
     3    / 10X, 35HINCREMENT LENGTH IN X DIRECTION    , 30X, 1PE10.3,   21JA1IBM
C    3    / 10X, 35HINCREMENT LENGTH IN X DIRECTION    , 30X,   E10.3,   21JA1CDC
     4    / 10X, 35HINCREMENT LENGTH IN Y DIRECTION    , 30X, 1PE10.3,   21JA1IBM
C    4    / 10X, 35HINCREMENT LENGTH IN Y DIRECTION    , 30X,   E10.3,   21JA1CDC
     5    / 10X, 35HPOISSONS RATIO                     , 30X, 1PE10.3,   21JA1IBM
C    5    / 10X, 35HPOISSONS RATIO                     , 30X,   E10.3,   21JA1CDC
     6    / 10X, 35HSLAB THICKNESS                     , 30X, 1PE10.3 )  21JA1IBM
C    6    / 10X, 35HSLAB THICKNESS                     , 30X,   E10.3 L  21JA1CDC
  300 FORMAT ( //49H      TABLE 3.  JOINT BENDING STIFFNESSES, LOADS,    10DE1
     &             15HAND SUPPORTS                                       10DE1
     1   //        50H      FROM   THRU      DX          DY          FX , 30AG8
     2             35H      FY          Q           S     ,            30AG8
     3    /        20H      JOINT  JOINT , / )                           30AG8
  311 FORMAT ( 5X, 2( 1X, I2, 1X, I3 ), 1P6E11.3 )                       21JA1IBM
C 311 FORMAT % 5X, 2% 1X, I2, 1X, I3 L,   6E11.3 L                        21JA1CDC
  400 FORMAT ( //51H      TABLE 4.  JOINT RESTRAINTS AND APPLIED MOMENTS 10DE1
     1   //        50H      FROM   THRU                              RX , 30AG8
     2             35H      RY          TX          TY    ,            30AG8
     3    /        20H      JOINT  JOINT , / )                           30AG8
  411 FORMAT ( 5X, 2( 1X, I2, 1X, I3 ), 22X, 1P4E11.3 )                  21JA1IBM
C 411 FORMAT % 5X, 2% 1X, I2, 1X, I3 L, 22X,   4E11.3 L                   21JA1CDC
  500 FORMAT ( //50H      TABLE 5.  MESH TWISTING STIFFNESSES            10DE1
     1   //        30H      FROM   THRU        C     ,                   30AG8
     2    /        20H      MESH   MESH  , / )                           30AG8
  511 FORMAT ( 5X, 2( 1X, I2, 1X, I3 ), 1PE11.3 )                        21JA1IBM
C 511 FORMAT % 5X, 2% 1X, I2, 1X, I3 L,   E11.3 L                         21JA1CDC
  600 FORMAT ( //40H      TABLE 6.  BAR AXIAL THRUSTS                    10DE1
     1   //        50H      FROM   THRU                              PX , 30AG8
     2             35H      PY          PBX         PBY   ,            30AG8
     3    /        20H      BAR    BAR   , / )                           30AG8
```

```
  611 FORMAT ( 5X, 2( 1X, I2, 1X, I3 ), 22X, 1P4E11.3 )                21JA1IBM
C 611 FORMAT % 5X, 2% IX, I2, 1X, I3 L, 22X,   4E11.3 L                  21JA1CDC
  700 FORMAT ( //39H      TABLE 7.  MULTIPLE LOADS                       10DE1
     1   //      50H          FROM    THRU                             , 30AG8
     2           20H                        CM,                          30AG8
     3   /       20H          JOINT  JOINT , / )                         30AG8
  711 FORMAT ( 5X, 2( 1X, I2, 1X, I3 ), 44X, 1PE11.3 )                   21JA1IBM
C 711 FORMAT % 5X, 2% 1X, I2, 1X, I3 L, 44X,   E11.3 L                   21JA1CDC
C 800 FORMAT % //35H      TABLE 8.  PROFILE OUTPUT AREAS  ,              21JA1CDC
  800 FORMAT ( //35H      TABLE 8.  PROFILE CUTPUT AREAS  ,              06DE1IBM
     1   //      40H          FROM    THRU      DEFL     X MOMENT        21JA1
     2           35H    Y MCMENT    PRIN MCM OR STRESS    ,              21JA1
C    3    /      42H          JOINT   JOINT  %1#YESL       %1#SLAB,      24FE1CDC
C    4           35H2#BEAML                  %1#YESL            L        24FE1CDC
     3    /      42H          JOINT   JOINT  (1=YES)       (1=SLAB,      06DE1IBM
     4           35H2=BEAM)                  (1=YES)            )        06DE1IBM
  811 FORMAT ( 5X, 2(1X,I2,1X,I3), 4X, I2, 9X, I2, 10X, I2, 14X, I2 )    16AP1
  812 FORMAT ( / 45H             BEAM MCMENTS ARE TOTAL PER BEAM      )  22JA1
  813 FORMAT ( / 25X, A6, 5X, A6, 13X,         20H  LARGEST     BETA     22JA1
     1   /        25X, A6, 5X, A6, 5X, A6, 2X,  20H PRINCIPAL    X TO    10DE1
     2   /        25X, A6, 5X, A6, 33H    TWISTING   SLAB      LARGEST  , 10DE1
     1             A7,                                                   22JA1
     2   /        25H       X , Y    DEFL        , A6, 5X, A6, 5X, A6, 4X, 22JA1
     3             A6, 4X, A6, 2X, A8  )                                 22JA1
  814 FORMAT ( 25X A6, 5X A6, 33X A7,                                    10DE1
     6   /       25H       X , Y     OEFL        , A6, 5X A6, 33X A8 )   20DE1
C 815 FORMAT % / 50H             SLAB X MCMENT AND X TWISTING MOMENT ACT  12FE1CDC
C    1           35HIN THE X DIRECTION %ABCUT Y AXISL     ,              22JA1CDC
C    2     / 10X, 50HY TWISTING MOMENT # -X TWISTING MOMENT, COUNTERCLO  22JA1CDC
  815 FORMAT ( / 50H             SLAB X MOMENT AND X TWISTING MOMENT ACT  06DE1IBM
     1           35HIN THE X DIRECTION (ABOUT Y AXIS)     ,              06DE1IBM
     2     / 10X, 50HY TWISTING MOMENT = -X TWISTING MOMENT, COUNTERCLO  06DE1IBM
     3           25HCKWISE BETA ANGLES ARE                               22JA1
     4     / 10X, 50HPCSITIVE FROM THE X AXIS TO THE DIRECTION OF THE L  22JA1
     5           17HARGEST PRINCIPAL , A6 ,                              22JA1
C    6   /  10X, 35HSLAB MGMENTS ARE PER UNIT WIDTH        L             22JA1CDC
     6   /  10X, 35HSLAB MCMENTS ARE PER UNIT WIDTH        )             06DE1IBM
  851 FORMAT ( 5X, I2, 1X, I3, 1P3E11.3, 28X, 1PE11.3 )                  21JA1IBM
C 851 FORMAT % 5X, I2, 1X, I3,   3E11.3, 28X,   E11.3 L                  21JA1CDC
  852 FORMAT ( 5X, I2, 1X, I3, 1P5E11.3, 0PF6.1, 1PE11.3 )               21JA1IBM
C 852 FORMAT % 5X, I2, 1X, I3,   5E11.3,   F6.1,   E11.3 L               21JA1CDC
  853 FORMAT ( 22X, 1P2E11.3 )                                           21JA1IBM
C 853 FORMAT % 22X,   2E11.3 L                                           21JA1CDC
C 860 FORMAT %///50H      STATICS CHECK.          SUMMATION OF REACTION, 13SE8CDC
C    1           6HS #    ,  E10.3  L                                    21JA1CDC
  860 FORMAT (///50H      STATICS CHECK.          SUMMATION OF REACTION, 06DE1IBM
     1           6HS =    , 1PE10.3  )                                   06DE1IBM
C 861 FORMAT % / 29X 35HMAXIMUM STATICS CHECK ERROR AT STA      ,2I3,    09FE1CDC
C    1           2H #,   E11.3 L                                         16AP1CDC
  861 FORMAT ( / 29X 35HMAXIMUM STATICS CHECK ERROR AT STA      ,2I3,    06DE1IBM
     1           2H =, 1PE11.3 )                                         06DE1IBM
  864 FORMAT (// 25H      PROFILE CUTPUT AREAS  )                        15FE1
C 865 FORMAT %   50H               X MCMENTS ACT IN X DIRECTION %ABOUT Y AX  15FE1CDC
C    1            3HISL,/,10X,  35HTHE PLOTTED RESULTS INDICATE THE RE   15MR1CDC
C    2           40HLATIVE VALUE EACH HAS WITHIN THAT LIST     L         15FE1CDC
  865 FORMAT (   50H               X MCMENTS ACT IN X DIRECTION (ABOUT Y AX  06DE1IBM
     1            3HIS),/,10X,  35HTHE PLOTTED RESULTS (INDICATE THE RE  06DE1IBM
     2           40HLATIVE VALUE EACH HAS WITHIN THAT LIST     )         06DE1IBM
C 866 FCRMAT %// 42H                                        BETWEEN      15FE1CDC
C    1           2H %, I3, 2H ,, I3, 8H L AND %, I3, 2H ,, I3, 2H L L    06DE1CDC
  866 FORMAT (// 42H                                        BETWEEN      29FE2IBM
     1           2H (, I3, 2H ,, I3, 8H ) AND (, I3, 2H ,, I3, 2H ) )    29FE2IBM
  867 FORMAT (    1H+ , 17X,     11HDEFLECTIONS                          10DE1IBM
     1    //     25H       X , Y DEFLECTION    ,                    / )  10DE1IBM
C 867 FORMAT %    1H& , 17X,     11HDEFLECTIONS                          10DE1CDC
C    1    //     25H       X , Y DEFLECTION    ,                    / L  10DE1CDC
  868 FORMAT (    1H+ ,  9X, A6, 5H AND ,A6, 1X, A6,                     10DE1IBM
     1    //     12H       X , Y , A6, 5H MCM , 27X A6,    5H MOM , / )  10DE1IBM
C 868 FORMAT %    1H& ,  9X, A6, 5H AND ,A6, 1X, A6,                     10DE1CDC
C    1    //     12H       X , Y , A6, 5H MCM , 27X A6,    5H MOM , / L  10DE1CDC
  869 FORMAT (    1H+ , 20X, A6, 1X, A6,                                 10DE1IBM
     1    //     12H       X , Y , A6,5H MOM ,                      / )  10DE1IBM
C 869 FORMAT %    1H& , 20X, A6, IX, A6,                                 10DE1CDC
C    1    //     12H       X , Y , A6,5H MOM ,                      / L  10DE1CDC
```

```
  870 FORMAT (    1H+ , 17X,     10HPRINCIPAL , A6,                      10DE1IBM
     1    //     17H       X , Y PRIN , A6,                         / )  10DE1IBM
C 870 FORMAT %    1H& , 17X,     10HPRINCIPAL , A6,                      10DE1CDC
C    1    //     17H       X , Y PRIN , A6,                         / L  10DE1CDC
  900 FORMAT ( //40H       TABLE 9.  PRINTED CUTPUT LIMITS      ,        24FE1
     1    //     20H        FROM    THRU     ,                           21JA1
     2    /      20H       Y STA    Y STA    )                           21JA1
  903 FORMAT ( / 25H                      NONE )                         17FE1
  905 FCRMAT (   46H            USING DATA FROM THE PREVIOUS PROBLEM )   17FE1
  910 FORMAT (   43H            ADDITIONAL CATA FOR THIS PROBLEM  )      31DE4
  911 FORMAT (2I5X,I5))                                                  03FE1
  920 FCRMAT ( //15H       RESULTS               )                       12FE1
  921 FORMAT ( //35H       RESULTS--USING STIFFNESS DATA ,               12FE1
     1              22HFRCM PREVIOUS PROBLEM ,     A5 )                  21JA1
  980 FORMAT (///40H       **** UNDESIGNATED ERROR STOP ****   )         19JE8
  981 FORMAT ( //51H       **** CAUTION.  MULTIPLE LOADING OPTION MISUSED 02FE1
     1              35H FOR THIS OR PRIOR PROBLEM ****     )             02FE1
  982 FORMAT ( //50H       **** MISUSE CF MULTIPLE LOACING OPTION ****  ) 02FE1
  983 FORMAT (//50H      **** IMPROPER NO OF CARDS INPUT OR KEPT **** )  02FE1
  984 FORMAT (  /50H       **** X INCREMENTS EXCEED Y INCREMENTS ****   ) 02FE1
  985 FORMAT (   35H       **** ERRONEOUS DATA INPUT ****       )        02FE1
  986 FORMAT (  /50H       **** THE DIMENSICNED STORAGE IS TOO SMALL FOR  02FE1
     1              30H THIS SIZE OF PROBLEM ****        )               02FE1
  990 FORMAT (   35H       **** TCTAL NUMBER OF SPECIFIED                12FE1
     1              10H PCINTS IS  , I5,                                 03FE1
     2              25H      , 300 IS MAX ****         )                 12FE1
  991 FORMAT (  /10H       **** , I4,                                    24FE1
     1              33H  CATA ERRORS IN THIS TABLE  ****  )              19AG8
  992 FORMAT (///30H       **** PRCBLEM TERMINATED  , I4  ,              23AG8
     1              20H  CATA ERRORS ****  )                             23AG8
C
                 ID1 = 0                                                 30NO1
                 ID2 = 0                                                 30NO1
                 ID3 = 0                                                 30NO1
                 ID4 = 0                                                 30NO1
                 I4D = 0                                                 06DE1IBM
C-----PROGRAM AND PROBLEM IDENTIFICATION
C
      CALL TIC TOC (1)                                                   17FE1
                 KML = 0                                                 05FE1
      READ 12, IAN1(N), N = 1, 40)                                       23FE1
 1010 READ 14, NPROB, (AN2(N), N = 1, 18)                                23FE1
          IF  (  NPROB  -  ITEST  )  1020,  9990,  1020                  04DE0
 1020 PRINT 11                                                           17FE1
 1021 PRINT  1                                                           17FE1
      PRINT 13, IAN1(N), N = 1, 40)                                      23FE1
      PRINT 15, NPROB, (AN2(N), N = 1, 18)                               23FE1
C
C-----INPUT TABLE 1  CCNTROL DATA                                        02FE1
C
      READ    20, KEEP2, KEEP3, KEEP4, KEEP5, KEEP6, KEEP7, KEEP8,       18MY0
     1             KEEP9, ML, ITMPP, NCD2, NCD3, NCD4, NCD5, NCD6, NCD7,  02FE1
     2             NCD8, NCD9, KROPT, IOPPS,IPDP,IGSW, I3D,VEF,RDF,SLOPE 06MY1
                 I4D = I4D + I3D                                         06DE1
      PRINT 100, KEEP2, KEEP3, KEEP4, KEEP5, KEEP6, KEEP7, KEEP8,        18MY0
     1             KEEP9,      NCD2, NCD3, NCD4, NCD5, NCD6, NCD7,       06DC0
     2             NCD8, NCD9, ML, KRCPT, IOPPS,       IPOP, I3D         02FE1
                 NDE1 = 0                                                02FE1
          IF (KML) 1174, 1170, 1172                                      02FE1
 1170     IF ( ML ) 1171, 1175, 1175                                     02FE1
 1171            NDE1 = NDE1 + 1                                         02FE1
      PRINT 982                                                          05FE1
          GO TO 1176                                                     02FE1
 1172     IF ( ML ) 1176, 1173, 1173                                     02FE1
 1173 PRINT 981                                                          02FE1
          GO TO 1175                                                     02FE1
 1174     IF ( ML ) 1176, 1175, 1175                                     02FE1
 1175     IF ( KEEP2 ) 1179, 1177, 1178                                  02FE1
 1176     IF ( KEEP2 ) 1179, 1179, 1178                                  02FE1
 1177     IF (NCD2 ) 1179, 1179, 1180                                    02FE1
 1178     IF ( NCD2 ) 1179, 1180, 1179                                   02FE1
 1179            NDE1 = NDE1 + 1                                         02FE1
      PRINT 983                                                          05FE1
 1180            KML = ML                                                02FE1
```

```
      IF (NDE1)  9980, 1200, 1182                                      23FE1
 1182 PRINT 991, NDE1                                                  05FE1
C
C-----INPUT TABLE 2 -- CONSTANTS                                       02FE1
C
 1200 PRINT 200                                                        02FE1
      IF ( KEEP2 ) 9980, 1201, 1240                                    24FE1
 1201       NDE2 = 0                                                   26AG8
      IF ( NCD2 - 1 ) 1203, 1205, 1203                                 28AG8
 1203       NDE2 = NDE2 + 1                                            26AG8
      PRINT 985                                                        23FE1
 1205 READ   21, MX, MY, HX, HY, PR, THK                               26AG8
      PRINT 201, MX, MY, HX, HY, PR, THK                               19AG8
      IF ( MX - MY ) 1211, 1211, 1210                                  19AG8
 1210       NDE2 = NDE2 + 1                                            16AG8
      PRINT 984                                                        05FE1
 1211 IF ( HX * HY ) 1212, 1212, 1213                                  19AG8
 1212       NDE2 = NDE2 + 1                                            16AG8
      PRINT 985                                                        04FE1
 1213 IF ( PR * THK ) 1214, 1225, 1225                                 23FE1
 1214       NDE2 = NDE2 + 1                                            16AG8
      PRINT 985                                                        04FE1
 1225 IF ( L1 - 3 - MX ) 1226, 1227, 1227                              23FE1
 1226       NDE2 = NDE2 + 1                                            02FE1
      PRINT 986                                                        23FE1
 1227 IF ( L2 - 3 - MY ) 1228, 1250, 1250                              24FE1
 1228       NDE2 = NDE2 + 1                                            02FE1
      PRINT 986                                                        05FE1
      GO TO 1250                                                       19AG8
 1240 PRINT 905                                                        16AG8
      PRINT 201, MX, MY, HX, HY, PR, THK                               19AG8
 1250 IF ( NDE2 ) 9980, 1300, 1270                                     16AG8
 1270 PRINT 991, NDE2                                                  16AG8
C
C-----INPUT TABLE 3 -- JOINT STIFFNESS AND LOAD DATA                   02FE1
C
 1300 PRINT 300                                                        05JE8
      IF ( KEEP3 ) 9980, 1301, 1310                                    05JE8
 1301       NC13 = 1                                                   05JE8
            NCT3 = NCD3                                                05JE8
            NDE3 = 0                                                   19AG8
            SWS  = 0.0                                                 19AG8
            SWB  = 0.0                                                 19AG8
      GO TO 1335                                                       05JE8
 1310 PRINT 905                                                        05JE8
      DO 1325  N = 1, NCT3                                             28JE8
      PRINT 311, IN13(N), JN13(N), IN23(N), JN23(N), DXN(N), DYN(N),   19AG8
     1          FXN(N), FYN(N), CN(N), SN(N)                           19AG8
 1325 CONTINUE                                                         05JE8
      PRINT 910                                                        05JE8
            NC13 = NCT3 + 1                                            05JE8
            NCT3 = NCT3 + NCD3                                         05JE8
 1335 IF ( NCD3 ) 9980, 1337, 1340                                     05JE8
 1337 PRINT 903                                                        05JE8
      GO TO 1372                                                       19AG8
 1340 DO 1370  N = NC13, NCT3                                          05JE8
      READ   33, IN13(N), JN13(N), IN23(N), JN23(N), DXN(N), DYN(N),   19AG8
     1          FXN(N), FYN(N), CN(N), SN(N)                           19AG8
      PRINT 311, IN13(N), JN13(N), IN23(N), JN23(N), DXN(N), DYN(N),   19AG8
     1          FXN(N), FYN(N), CN(N), SN(N)                           19AG8
      IF ( IN13(N) - IN23(N) ) 1342, 1342, 1341                        05JE8
 1341       NDE3 = NDE3 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1342 IF ( JN13(N) - JN23(N) ) 1344, 1344, 1343                        05JE8
 1343       NDE3 = NDE3 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1344 IF ( IN23(N) - MX ) 1346, 1346, 1345                             05JE8
 1345       NDE3 = NDE3 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1346 IF ( JN23(N) - MY ) 1350, 1350, 1347                             09SE8
 1347       NDE3 = NDE3 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1350       SWS = SWS + ABS ( DXN(N) + DYN(N) )                        04SE8
            SWB = SWB + ABS ( FXN(N) + FYN(N) )                        04SE8
 1370 CONTINUE                                                         05JE8
 1372 IF ( NDE3 ) 9980, 1400, 1375                                     19AG8
 1375 PRINT 991, NDE3                                                  19AG8
C
C-----INPUT TABLE 4 -- JOINT STIFFNESS AND LOAD DATA CONTD             02FE1
C
 1400 PRINT 400                                                        05JE8
      IF ( KEEP4 ) 9980, 1401, 1410                                    05JE8
 1401       NC14 = 1                                                   26MR5
            NCT4 = NCD4                                                19OC4
            NDE4 = 0                                                   19AG8
      GO TO 1435                                                       05JE8
 1410 PRINT 905                                                        04JA5
      DO 1425  N = 1, NCT4                                             04JA5
      PRINT 411, IN14(N), JN14(N), IN24(N), JN24(N), RXN(N), RYN(N),   19AG8
     1          TXN(N), TYN(N)                                         19AG8
 1425 CONTINUE                                                         04JA5
      PRINT 910                                                        04JA5
            NC14 = NCT4 + 1                                            21OC4
            NCT4 = NCT4 + NCD4                                         19OC4
 1435 IF ( NCD4 ) 9980, 1437, 1440                                     04JA5
 1437 PRINT 903                                                        04JA5
      GO TO 1472                                                       19AG8
 1440 DO 1470  N = NC14, NCT4                                          05JE8
      READ   43, IN14(N), JN14(N), IN24(N), JN24(N), RXN(N), RYN(N),   19AG8
     1          TXN(N), TYN(N)                                         19AG8
      PRINT 411, IN14(N), JN14(N), IN24(N), JN24(N), RXN(N), RYN(N),   19AG8
     1          TXN(N), TYN(N)                                         19AG8
      IF ( IN14(N) - IN24(N) ) 1442, 1442, 1441                        05JE8
 1441       NDE4 = NDE4 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1442 IF ( JN14(N) - JN24(N) ) 1444, 1444, 1443                        05JE8
 1443       NDE4 = NDE4 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1444 IF ( IN24(N) - MX ) 1446, 1446, 1445                             05JE8
 1445       NDE4 = NDE4 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1446 IF ( JN24(N) - MY ) 1470, 1470, 1447                             29JL9
 1447       NDE4 = NDE4 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1470 CONTINUE                                                         04JA5
 1472 IF ( NDE4 ) 9980, 1500, 1475                                     19AG8
 1475 PRINT 991, NDE4                                                  19AG8
C
C-----INPUT TABLE 5 -- MESH STIFFNESS DATA                             02FE1
C
 1500 PRINT 500                                                        19OC4
      IF ( KEEP5 ) 9980, 1501, 1510                                    04JA5
 1501       NC15 = 1                                                   26MR5
            NCT5 = NCD5                                                19OC4
            NDE5 = 0                                                   19AG8
      GO TO 1535                                                       05JE8
 1510 PRINT 905                                                        04JA5
      DO 1525  N = 1, NCT5                                             04JA5
      PRINT 511, IN15(N), JN15(N), IN25(N), JN25(N), CHN(N)            19AG8
 1525 CONTINUE                                                         21JE8
      PRINT 910                                                        20JA5
            NC15 = NCT5 + 1                                            06NO4
            NCT5 = NCT5 + NCD5                                         19OC4
 1535 IF ( NCD5 ) 9980, 1537, 1540                                     04JA5
 1537 PRINT 903                                                        04JA5
      GO TO 1572                                                       19AG8
 1540 DO 1570  N = NC15, NCT5                                          05JE8
      READ   53, IN15(N), JN15(N), IN25(N), JN25(N), CHN(N)            19AG8
      PRINT 511, IN15(N), JN15(N), IN25(N), JN25(N), CHN(N)            19AG8
      IF ( IN15(N) - IN25(N) ) 1542, 1542, 1541                        05JE8
 1541       NDE5 = NDE5 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1542 IF ( JN15(N) - JN25(N) ) 1544, 1544, 1543                        05JE8
 1543       NDE5 = NDE5 + 1                                            19AG8
      PRINT 985                                                        04FE1
 1544 IF ( IN25(N) - MX ) 1546, 1546, 1545                             05JE8
 1545       NDE5 = NDE5 + 1                                            19AG8
      PRINT 985                                                        04FE1
```

```
 1546      IF ( JN25(N) - MY ) 1548, 1548, 1547                          05JE8
 1547            NDE5 = NDE5 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1548      IF ( IN15(N) * IN25(N) * JN15(N) * JN25(N) ) 9980, 1549, 1570 19AG8
 1549            NDE5 = NDE5 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1570       CONTINUE                                                      04JA5
 1572      IF ( NDE5 ) 9980, 1600, 1575                                   19AG8
 1575 PRINT 991, NDE5                                                     19AG8
C
C-----INPUT TABLE 6 -- BAR STIFFNESS DATA                                  02FE1
C
 1600 PRINT 600                                                           07JE5
           IF ( KEEP6 ) 9980, 1601, 1610                                  05JE8
 1601            NC16 = 1                                                 05JE8
                 NCT6 = NCO6                                              05JE8
                 NDE6 = 0                                                 19AG8
           GO TO 1635                                                     05JE8
 1610 PRINT 905                                                           05JE8
           DO 1625  N = 1, NCT6                                           05JE8
      PRINT 611, IN16(N), JN16(N), IN26(N), JN26(N), PXN(N), PYN(N),      19AG8
     1           PBXN(N), PBYN(N)                                         19AG8
 1625      CONTINUE                                                       05JE8
      PRINT 910                                                           05JE8
                 NC16 = NCT6 + 1                                          05JE8
                 NCT6 = NCT6 + NCO6                                       05JE8
 1635      IF ( NCO6 ) 9980, 1637, 1640                                   05JE8
 1637 PRINT 903                                                           05JE8
           GO TO 1672                                                     19AG8
 1640      DO 1670  N = NC16, NCT6                                        05JE8
      READ   63, IN16(N), JN16(N), IN26(N), JN26(N), PXN(N), PYN(N),      02FE1
     1           PBXN(N), PBYN(N)                                         19AG8
      PRINT 611, IN16(N), JN16(N), IN26(N), JN26(N), PXN(N), PYN(N),      19AG8
     1           PBXN(N), PBYN(N)                                         19AG8
           IF ( IN16(N) - IN26(N) ) 1642, 1642, 1641                      05JE8
 1641            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1642      IF ( JN16(N) - JN26(N) ) 1644, 1644, 1643                      05JE8
 1643            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1644      IF ( IN26(N) - MX ) 1646, 1646, 1645                           05JE8
 1645            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1646      IF ( JN26(N) - MY ) 1648, 1648, 1647                           19AG8
 1647            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1648      IF ( PXN(N) + PBXN(N) ) 1665, 1667, 1665                       19AG8
 1665      IF ( IN16(N) * IN26(N) ) 9980, 1666, 1667                      19AG8
 1666            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1667      IF ( PYN(N) + PBYN(N) ) 1668, 1670, 1668                       19AG8
 1668      IF ( JN16(N) * JN26(N) ) 9980, 1669, 1670                      19AG8
 1669            NDE6 = NDE6 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1670      CONTINUE                                                       05JE8
 1672      IF ( NDE6 ) 9980, 1700, 1675                                   19AG8
 1675 PRINT 991, NDE6                                                     19AG8
C
C-----INPUT TABLE 7 -- MULTIPLE LOAD JOINT DATA                            02FE1
C
 1700 PRINT 700                                                           19AG8
           IF ( KEEP7 ) 9980, 1701, 1710                                  19AG8
 1701            NC17 = 1                                                 19AG8
                 NCT7 = NCO7                                              19AG8
                 NDE7 = 0                                                 19AG8
           GO TO 1735                                                     19AG8
 1710 PRINT 905                                                           19AG8
           DO 1725  N = 1, NCT7                                           19AG8
      PRINT 711, IN17(N), JN17(N), IN27(N), JN27(N), QMN(N)               19AG8
 1725      CONTINUE                                                       19AG8
      PRINT 910                                                           19AG8
                 NC17 = NCT7 + 1                                          19AG8
                 NCT7 = NCT7 + NCO7                                       19AG8
 1735      IF ( NCO7 ) 9980, 1737, 1740                                   19AG8
 1737 PRINT 903                                                           19AG8
           GO TO 1772                                                     19AG8
 1740      DO 1770  N = NC17, NCT7                                        19AG8
      READ   73, IN17(N), JN17(N), IN27(N), JN27(N), QMN(N)               19AG8
      PRINT 711, IN17(N), JN17(N), IN27(N), JN27(N), QMN(N)               19AG8
           IF ( IN17(N) - IN27(N) ) 1742, 1742, 1741                      19AG8
 1741            NDE7 = NDE7 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1742      IF ( JN17(N) - JN27(N) ) 1744, 1744, 1743                      19AG8
 1743            NDE7 = NDE7 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1744      IF ( IN27(N) - MX ) 1746, 1746, 1745                           19AG8
 1745            NDE7 = NDE7 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1746      IF ( JN27(N) - MY ) 1770, 1770, 1747                           19AG8
 1747            NDE7 = NDE7 + 1                                          19AG8
      PRINT 985                                                           04FE1
 1770      CONTINUE                                                       19AG8
 1772      IF ( NDE7 ) 9980, 1800, 1775                                   19AG8
 1775 PRINT 991, NDE7                                                     19AG8
C
C-----INPUT TABLE 8 -- PROFILE OUTPUT AREAS                                02FE1
C
 1800 PRINT 800                                                           02FE1
           IF ( KEEP8 ) 9980, 1801, 1810                                  02FE1
 1801            NC18 = 1                                                 19MY0
                 NCT8 = NCO8                                              19MY0
                 NDE8 = 0                                                 19MY0
                 IBCS = 0                                                 08FE1
                 IBOST = 0                                                08FE1
           GO TO 1835                                                     02FE1
 1810 PRINT 905                                                           02FE1
           DO 1825  N = 1, NCT8                                           02FE1
      PRINT 811, IN18(N), JN18(N), IN28(N), JN28(N), KASEW(N), KASEX(N),12FE1
     1           KASEY(N), KASEP(N)                                       02FE1
 1825      CONTINUE                                                       02FE1
      PRINT 910                                                           19MY0
                 NC18 = NCT8 + 1                                          19MY0
                 NCT8 = NCT8 + NCO8                                       19MY0
 1835      IF ( NCO8 ) 9980, 1837, 1840                                   03FE1
 1837 PRINT 903                                                           03FE1
           GO TO 1872                                                     09FE1
 1840      DO 1870 N = NC18, NCT8                                         04FE1
      READ  83, IN18(N), JN18(N), IN28(N), JN28(N), KASEW(N), KASEX(N),   04FE1
     1           KASEY(N), KASEP(N)                                       04FE1
      PRINT 811, IN18(N), JN18(N), IN28(N), JN28(N), KASEW(N), KASEX(N),04FE1
     1           KASEY(N), KASEP(N)                                       04FE1
                 NEND =  (IN28(N)-IN18(N)+1)*(JN28(N)-JN18(N)+1)          04FE1
           IF ( NEND - 300 )   1842, 1842, 1841                           04FE1
 1841            NDE8 = NDE8 + 1                                          04FE1
      PRINT 990, NEND                                                     04FE1
 1842      IF ( IN28(N) .GE. IN18(N) )   GO TO 1843                       26FE1
                 NDE8 = NDE8 + 1                                          24FE1
      PRINT 985                                                           04FE1
 1843      IF ( JN18(N) - JN28(N) ) 1845, 1845, 1844                      04FE1
 1844            NDE8 = NDE8 + 1                                          04FE1
      PRINT 985                                                           04FE1
 1845      IF ( IN28(N) - MX ) 1847, 1847, 1846                           04FE1
 1846            NDE8 = NDE8 + 1                                          04FE1
      PRINT 985                                                           04FE1
 1847      IF ( JN28(N) - MY ) 1849, 1849, 1848                           17FE1
 1848            NDE8 = NDE8 + 1                                          04FE1
      PRINT 985                                                           04FE1
 1849      CONTINUE                                                       17FE1
           IF ( KASEX(N) + KASEY(N) ) 9980, 1870, 1850                    08FE1
 1850      IF ( KASEX(N) + KASEY(N) - 3 ) 1852, 1851, 1852                08FE1
 1851            NDE8 = NDE8 + 1                                          08FE1
      PRINT 985                                                           08FE1
           GO TO 1870                                                     08FE1
 1852      IF ( KASEX(N).EQ.2 )   GO TO 1856                              16AP1
           IF ( KASEY(N).EQ.2 )   GO TO 1856                              16AP1
                 IBCS = 1                                                 08FE1
           IF ( IBOST .EQ. 2 ) GO TO 1851                                 09FE1
                 IBOST = 1                                                08FE1
```

```
         GO TO 1870                                                     08FE1
 1856           IBOS = 2                                                08FE1
         IF ( IBCST .EQ. 1 ) GO TC 1851                                 08FE1
                IBOST = 2                                               08FE1
 1870    CONTINUE                                                       08FE1
 1872    IF ( NDE8) 5580, 1900, 1875                                    03FE1
 1875 PRINT 991, NDE8                                                   03FE1
C                                                                       06DC0
C-----INPUT TABLE 9 -- PRINTED OUTPUT LIMITS                             03FE1
C                                                                       06DC0
 1900 PRINT 900                                                         03FE1
         IF ( KEEP9 ) 9580, 1901, 1910                                  03FE1
 1901           NCI9 = 1                                                16MR1
                NCT9 = NCD9                                             16MR1
                NDE9 = 0                                                06DC0
         GO TO  1935                                                    03FE1
 1910 PRINT 905                                                         03FE1
         DO 1925  N = 1, NCT9                                           16MR1
      PRINT 911, JN19(N), JN29(N)                                       03FE1
 1925    CONTINUE                                                       03FE1
      PRINT 910                                                         06DC0
                NC19 = NCT9 + 1                                         06DC0
                NCT9 = NC19 + NCD9                                      06DC0
 1935    IF ( NCD9 ) 5980, 1937, 1940                                   03FE1
 1937 PRINT 903                                                         03FE1
         GO TO  1972                                                    03FE1
 1940    DO 1970  N = NC19, NCT9                                        03FE1
      READ 93  , JN19(N), JN29(N)                                       06DC0
      PRINT 911, JN19(N), JN29(N)                                       03FE1
         IF ( JN19(N) - JN29(N) ) 1944, 1944, 1943                      03FE1
 1943           NDE9 = NDE9 + 1                                         03FE1
      PRINT 985                                                         03FE1
 1944    IF ( JN29(N) - MY ) 1970, 1970,1947                            03FE1
 1947           NDE9 = NDE9 + 1                                         03FE1
      PRINT 985                                                         03FE1
 1970    CONTINUE                                                       03FE1
 1972    IF ( NDE9 ) 5980, 4870, 1975                                   16MR1
 1975 PRINT 991, NDE9                                                   03FE1
C
C-----COMPUTE FOR CONVENIENCE
C
 4870    IF ( ML ) 48E6, 4875, 4875                                     16MR1
 4875           MXP1 = MX + 1                                           22MR1
                MYP1 = MY + 1                                           29NO7
                MXP2 = MX + 2                                           08DE7
                MYP2 = MY + 2                                           29NO7
                MXP3 = MX + 3                                           08DE7
                MYP3 = MY + 3                                           08DE7
                MXP4 = MX + 4                                           08DE7
                MYP4 = MY + 4                                           08DE7
                MXP5 = MX + 5                                           08DE7
                MYP5 = MY + 5                                           08DE7
                NF = 1                                                  23OE0
                ODHX = 1.0 / HX                                         14SE6
                ODHY = 1.0 / HY                                         14SE6
                ODHXHY = ODHX * ODHY                                    14NO7
                PDHXHY = PR * ODHXHY                                    14NO7
                ODHX2 = ODHX * ODHX                                     14NO7
                ODHY2 = ODHY * ODHY                                     14NO7
                ODHX3 = ODHX2 * ODHX                                    30MY8
                ODHY3 = ODHY2 * ODHY                                    30MY8
                HXDHY = HX * ODHY                                       22MR1
                HYDHX = HY * ODHX                                       14NO7
                HYDHX3 = HYDHX * ODHX2                                  14NO7
                HXDHY3 = HXDHY * ODHY2                                  14NO7
                KPROB  =  NPROB                                         04DE0
         DO 4880  J = 1, MYP3                                           03FE1
                PBMX(1,J) = 0.0                                         03FE1
                PBMY(1,J) = 0.0                                         03FE1
                PSIGO(1,J) = 0.0                                        03FE1
                PBMY(MXP3,J) = 0.0                                      03FE1
                PBMX(MXP3,J) = 0.0                                      03FE1
                PSIGO(MXP3,J) = 0.0                                     03FE1
 4880    CONTINUE                                                       04DE0
```

```
 4866    DO 4888  I = 1, MXP3                                           04DE0
                WM2(I) = 0.0                                            09FE1
                PBMX(I,MYP3) = 0.                                       15JL0
                PBMX(I,1) = 0.                                          15JL0
                PBMY(I,MYP3) = 0.                                       15JL0
                PBMY(I,1) = 0.                                          15JL0
                EE(I,1) = 0.0                                           24MY1
                BMX(I)     = 0.0                                        08JL8
                BMYM1(I)   = 0.0                                        08JL8
                BMY(I)     = 0.0                                        08JL8
                BMYP1(I)   = 0.0                                        08JL8
                BMBX(I)    = 0.0                                        08JL8
                BMBYM1(I)  = 0.0                                        08JL8
                BMBY(I)    = 0.0                                        08JL8
                BMBYP1(I)  = 0.0                                        08JL8
 4888    CONTINUE                                                       04DE0
                STEMP = 0.0                                             06DC0
                NDES = NDE1+NDE2+NDE3+NDE4+NDE5+NDE6+NDE7+NDE8+NDE9     12FE1
         IF ( NDES ) 5980, 5401, 4900                                   04DE0
 4900 PRINT 952, NDES                                                   04DE0
         GO TO 1010                                                     23AG8
C
C-----MAIN SOLUTION --FORM STIFFNESS MATRIX AND SOLVE EQUATIONS          09FE1
C
 5401 CALL FRIP8          ( L1   , L2   , ML   , 1    , 3    , 5    ,  08FE1
     1                      A    , AM1  , AM2  , AT   , R1   , R2   , R3   , 08FE1
     2                      AA   , BB   , CC   , DD   , EE   , FF   , PW   , 08FE1
     3                      DX   , DXM1, DXP1, DY   , DYM1, DYP1,     08FE1
     4                      FX   , FY   , FYM1, FYP1, Q    , S    ,  08FE1
     5                      RX   , RYM1 , RYP1, TX   , TYM1, TYP1, CH   , 08FE1
     6                      CHP1, PX   , PY   , PYP1, PBX  , PBY  , PBYP1, 08FE1
     7                      CM                                     )    26FE1
C-----SET DEFLECTIONS AT CORNER STATIONS OUTSIDE THE BOUNDARIES
                PW(  1,    1) = 2.0 * PW(   1,    2) - PW(   1,    3)   06MY1
                PW(MXP3,   1) = 2.0 * PW(MXP3,    2) - PW(MXP3,    3)   06MY1
                PW(   1,MYP3) = 2.0 * PW(   1,MYP2) - PW(   1,MYP1)     06MY1
                PW(MXP3,MYP3) = 2.0 * PW(MXP3,MYP2) - PW(MXP3,MYP1)     06MY1
C
C-----COMPUTE AND PRINT RESULTS
C
                ISW = 0                                                 24JL8
                SUMR = 0.0                                              12DE7
                JN = MY                                                 24JL8
      CALL DATA          (   L1,    JN,     1,    MXP3,                 04DE0
     1                      DX, DYM1,   DY, DYP1,   FX, FYM1,           24JE8
     2                      FY, FYP1,    Q,    S,                       24JE8
     3                      RX, RYM1, RYP1,   TX, TYM1, TYP1,           24JE8
     4                      CH, CHP1,                  DXP1, DXM1,      05AG9
     5                      PX,   PY, PYP1,  PBX,  PBY, PBYP1,          24JE8
     6                      CM, IPRINT                              )   15MR1
      PRINT 11                                                          15SE6
      PRINT  1                                                          21JL7
      PRINT 13, AN1                                                     06DC0
      PRINT 16, NPROB, AN2                                              06DC0
         IF ( ML ) 6115, 6110, 6110                                     04SE8
 6110 PRINT 920                                                         12FE1
         GO TO 6120                                                     04SE8
 6115 PRINT 921, KPROB                                                  12FE1
 6120           ROSC1 = SUP                                             15MR1
                ROSC2 = RCT                                             08FE1
         IF ( KROPT.EQ.1 )          ROSC1 = STAT                        08FE1
         IF ( KROPT.EQ.1 )          ROSC2 = CHK                         08FE1
                SCM = BMOM                                              08FE1
         IF ( IOPPS.EQ.1)           SCM = STRS                          08FE1
         IF ( SWS ) 5580, 6125, 6130                                    15MR1
C-----PRINT HEADINGS FOR BEAM ONLY OUTPUT
 6125 PRINT 6                                                           08FE1
      PRINT 812                                                         08FE1
      PRINT 6                                                           08FE1
      PRINT 814, BEMX, BEMY, ROSC1, BMOM, BMOM,               ROSC2     10DE1
         GO TO 6145                                                     08FE1
 6130    IF ( SWB ) 598C, 6135, 6140                                    08FE1
C-----PRINT HEADINGS FOR SLAB ONLY OUTPUT
 6135 PRINT 6                                                           08FE1
```

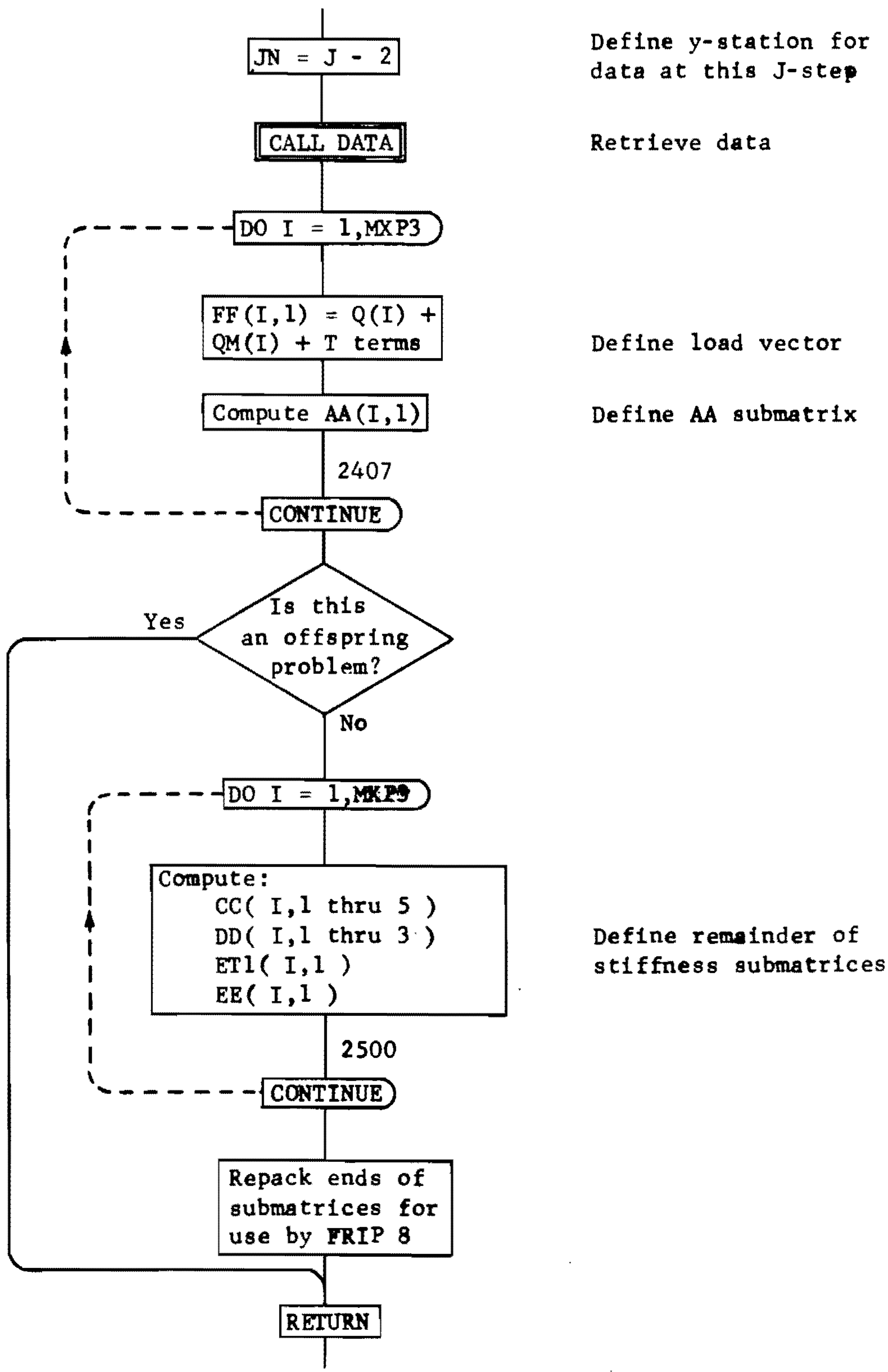
```
      PRINT 815, SGM                                                    16AP1
      PRINT 6                                                           08FE1
      PRINT 813, BLNK, BLNK, BLNK, BLNK, SLBX,                          10DE1
     &            SLBX, SLBY, RDSCI, BMOM, BMOM, BMOM, SOM, SCM, ROSC2  10DE1
         GO TO 6145                                                     08FE1
C-----PRINT HEADINGS FCR CCMBINED SLAB AND BEAM OUTPUT
 6140 PRINT 6                                                           08FE1
      PRINT 815, SDM                                                    16AP1
      PRINT 812                                                         08FE1
      PRINT 6                                                           08FE1
      PRINT 813, SLBX, SLBY, BMOM, BMCM, SLBX,                          10DE1
     &            BEMX, BEMY, RDSC1, BMOM, BMOM, BMOM, SDM, SCM, ROSC2  10DE1
 6145    IF (IOPPS) 9980, 6180, 6175                                    06DC0
 6175    IF ( THK ) 998C, 6180, 6176                                    09FE1
 6176          SDT2 = 6.0 / ( THK * THK )                               06DC0
         GO TO 6200                                                     09FE1
 6180          SDT2 = 1.0                                               06DC0
C
C-----COMPUTE BENDING MOMENTS, REACTICNS AND TWISTING MOMENTS
C
 6200    DO 8650  LL = 1, MYP3                                          15MR1
               J = MYP4 - LL                                            24JL8
         DO 6250  I = 1, MXP3                                           08JL8
               WP2(I) = WP1(I)                                          08JL8
               WP1(I) = W(I)                                            05JL8
               W(I)   = WM1(I)                                          05JL8
               WM1(I) = WM2(I)                                          05JL8
 6250    CONTINUE                                                       08JL8
         DO 6300  I = 1, MXP3                                           12MY0
               WM2(I) = PW(I,J)                                         12MY0
 6300    CONTINUE                                                       12MY0
         IF ( MYP1 - J ) 8650, 6400, 6500                               02MA1
 6400    DO 6450  I = 2, MXP2                                           24JL8
               CRD(2) = AMIN1 ( DX(I), DY(I) )                          04AG9
               WSUMM1     = ODHX2 * ( WM1(I-1) - 2.0 * WM1(I)           05JL8
     I                          + WM1(I+1) )                            05JL8
               WSUMM2     = ODHY2 * ( WM2(I) - 2.0 * WM1(I) + W(I) )    05JL8
               BMYM1(I)   =   DY(I) * WSUMM2 + CRD(2) * PR * WSUMM1     05AG9
               BMBYM1(I)  =   FY(I) * WSUMM2                            26JL8
 6450    CONTINUE                                                       24JL8
         GO TO 8650                                                     09FE1
 6500    DO 7200  I = 2, MXP2                                           08JL8
               CRD(2) = AMIN1 ( DX(I), DY(I) )                          04AG9
               CRD(4) = AMIN1 ( DXM1(I), DYM1(I) )                      04AG9
               BMYP1(I)   = BMY(I)                                      08JL8
               BMY(I)     = BMYM1(I)                                    08JL8
               BMBYP1(I)  = BMBY(I)                                     08JL8
               BMBY(I)    = BMBYM1(I)                                   08JL8
               WSUM1      = ODHX2 * ( W(I-1) - 2.0 * W(I) + W(I+1) )    08JL8
               WSUM2      = ODHY2 * ( WM1(I) - 2.0 * W(I) + WP1(I) )    08JL8
               WSUMM1     = ODHX2 * ( WM1(I-1) - 2.0 * WM1(I)           05JL8
     I                          + WM1(I+1) )                            05JL8
               WSUMM2     = ODHY2 * ( WM2(I) - 2.0 * WM1(I) + W(I) )    05JL8
               BMX(I)     = DX(I) * WSUM1 + CRD(2) * PR * WSUM2         05AG9
               BMYM1(I)   = DYM1(I) * WSUMM2 + CRD(4) * PR * WSUMM1     05AG9
               BMBX(I)    = FX(I) * WSUM1                               08JL8
               BMBYM1(I)  = FYM1(I) * WSUMM2                            26JL8
C-----PBMX AND PBMY ARE STORED HERE FROM J = MYP2 TO 3
         IF ( IBOS .EQ. 2 ) GO TO 6525                                  11FE1
               PBMX(I,J+2) = BMX(I)                                     11FE1
               PBMY(I,J+2) = BMY(I)                                     11FE1
         GO TO 7200                                                     11FE1
 6525          PBMX(I,J+2) = BMBX(I)                                    11FE1
               PBMY(I,J+2) = BMBY(I)                                    11FE1
 7200    CONTINUE                                                       08JL8
               JSTA = J                                                 08JL8
 7300    IF ( IPRINT )  9980, 7310, 7305                                15MR1
 7305 PRINT 6                                                           15MR1
 7310    DO 8550  I = 2, MXP2                                           15MR1
               ISTA = I - 2                                             29NO7
               QBMX       = HYDHX * ( BMX(I-1) - 2.0 * BMX(I) + 0.000   08JL8IBM
C              QBMX       # HYDHX * % BMX%I-1L - 2.0 * BMX%IL           08JL8CDC
     1                          + BMX(I+1) )                            08JL8
               QBMY       = HXDHY * ( BMYM1(I) - 2.0 * BMY(I)+0.000     08JL8IBM
C              QBMY       # HXDHY * % BMYM1%IL - 2.0 * BMY%IL           08JL8CDC
     1                          + BMYP1(I) )                            08JL8
               QBMBX      = ODHX * ( BMBX(I-1) - 2.0 * BMBX(I) + 0.000  08AG8IBM
C              QBMBX      # ODHX * % BMBX%I-1L - 2.0 * BMBX%IL          08AG8CDC
     1                          + BMBX(I+1) )                           08JL8
               QBMBY      = ODHY * ( BMBYM1(I) - 2.0 * BMBY(I) + 0.000  08AG8IBM
C              QBMBY      # ODHY * % BMBYM1%IL - 2.0 * BMBY%IL          08AG8CDC
     1                          + BMBYP1(I) )                           08JL8
               QTMX       = ODHXHY * ( WM1(I-1) * CH(I)                 08JL8
     1                          - W(I-1) * ( CH(I) + CHP1(I) )          08JL8
     2                          + WP1(I-1) * CHP1(I)                    08JL8
     3                          - WM1(I) * ( CH(I) + CH(I+1) )          08JL8
     4                          + W(I) * ( CH(I) + CHP1(I) + CH(I+1)    08JL8
     5                          + CHP1(I+1) )                           08JL8
     6                          - WP1(I) * ( CHP1(I) + CHP1(I+1) )      08JL8
     7                          + WM1(I+1) * CH(I+1)                    08JL8
     8                          - W(I+1) * ( CH(I+1) + CHP1(I+1) )      08JL8
     9                          + WP1(I+1) * CHP1(I+1) )                08JL8
               QTMY       = QTMX                                        08JL8
               QPX        = ODHX * ( W(I-1) * PX(I)                     08JL8
     1                          - W(I) * ( PX(I) + PX(I+1) )            08JL8
     2                          + W(I+1) * PX(I+1) )                    08JL8
               QPY        = CDHY * ( WM1(I) * PY(I)                     08JL8
     1                          - W(I) * ( PY(I) + PYP1(I) )            08JL8
     2                          + WP1(I) * PYP1(I) )                    08JL8
               QPBX       = ODHX * ( W(I-I) * PBX(I)                    08JL8
     1                          - W(I) * ( PBX(I) + PBX(I+1) )          08JL8
     2                          + W(I+1) * PBX(I+1) )                   08JL8
               QPBY       = CDHY * ( WM1(I) * PBY(I)                    08JL8
     1                          - W(I) * ( PBY(I) + PBYP1(I) )          08JL8
     2                          + WP1(I) * PBYP1(I) )                   08JL8
         IF ( I - 2 ) 998C, 7450, 7500                                  29JL8
 7450          QRX        = 0.25 * ODHX2 * ( - RX(I-1) * W(I)           29JL8
     1                          + RX(I+1) * ( - W(I) + W(I+2) ) )       29JL8
         GO TO 7650                                                     24JE9
 7500    IF ( MXP2 - I ) 9980, 7600, 7550                               29JL8
 7550          QRX        = 0.25 * CDHX2 * ( - RX(I-1) * (              29JL8
     1                          - W(I-2) + W(I) )                       29JL8
     2                          + RX(I+1) * ( - W(I) + W(I+2) ) )       29JL8
         GO TO 7650                                                     29JL8
 760C          QRX        = 0.25 * ODHX2 * ( - RX(I-1) * (              29JL8
     1                          - W(I-2) + W(I) )                       29JL8
     2                          - RX(I+I) * W(I) )                      29JL8
 7650          QRY        = 0.25 * ODHY2 * ( - RYM1(I) *                29JL8
     1                          ( - WM2(I) + W(I) )                     29JL8
     2                          + RYP1(I) * ( - W(I) + WP2(I) ) )       29JL8
               QTX        = 0.5 * CDHX * ( - TX(I-1) + TX(I+1) )        29JL8
               QTY        = 0.5 * CDHY * ( - TYM1(I) + TYP1(I) )        29JL8
C-----CCMPUTE TWISTING MCMENTS
               WSUM3      = 0.0625 * ODHXHY * ( WM1(I-1) - WP1(I-1)     08JL8
     I                          - WM1(I+1) + WP1(I+1) )                 08JL8
               TMX        = WSUM3 * ( CH(I) + CHP1(I)                   08JL8
     1                          + CH(I+I) + CHP1(I+1) )                 08JL8
C-----CCMPUTE SUPPORT REACTIONS
               REACT = -S(I) * W(I)                                     06DC0
C-----CCMPUTE STATICS CHECK ERROR
               STACH      = CBMX + QBMY + QBMBX + QBMBY + QTMX + QTMY   06DC0
     1                          - QPX - CPY - QPBX - QPBY - QRX - QRY   29JL8
     2                          - QTX - QTY - Q(I) - QM(I) - REACT      06DC0
         IF ( S(I) .LT. 1.E+19 )        GC TO  7700                     01DE1
               REACT = STACH                                            01DE1
               STACH =  C.                                              01DE1
C-----SUMMATICN OF REACTICNS
 7700          SUMR       = SUMR + REACT                                01DE1
               ROCK = REACT                                             09FE1
         IF ( KRCPT .EQ. 1 )  ROCK = STACH                              11FE1
         IF ( DABS(STACH) .LE. DABS(STEMP) )  GO TO 8040                14MY1IBM
C        IF %  ABS%STACHL .LE.  ABS%STEMPL L  GO TO 8040                14MY1CDC
               STEMP = STACH                                            06DC0
               ITEMP = ISTA                                             06DC0
               JTEMP = JSTA                                             06DC0
C-----PRINT BEAM CNLY OUTPUT
 8040    IF ( SWS ) 998C, 8050, 8060                                    04SE8
 8050    IF ( IPRINT ) 9980, 8058, 8054                                 15MR1
```

```
 8054 PRINT 851, ISTA, JSTA, W(I), BMBX(I), BMBY(I), ROCK                15MR1
 8058      GO TO 8550                                                     15MR1
C
C-----CCMPUTE PRINCIPAL MCMENTS DR STRESSES                               14NO7
C
 8060            BMA    = C.5 * ( BMX(I) + BMY(I) )                       08JL8
                 TMY    = - TMX                                           07DE0
                 BMP    = BMX(I) - BMA                                    08JL8
                 BMR    =DSCRT ( BMP * BMP + TMY * TMY )                  07DE0IBM
C                BMR    # SQRT % BMP * BMP C TMY * TMY L                  07DE0CDC
                 BMO    = BMA + BMR                                       08JL8
                 BMT    = BMA - BMR                                       08JL8
C-----TEST TO PRINT ONLY THE MAXIMUM VALUE                                15OE7
          IF ( BMA ) 8160, 8180, 8180                                     08JL8
 8160            PMMAX  = BMT                                             08JL8
          IF ( BMP ) 8400, 8300, 8200                                     08JL8
 8180            PMMAX  = BMO                                             08JL8
          IF ( BMP ) 8200, 8300, 8400                                     08JL8
 8200            ALF    = TMY / BMP                                       07DE0
                 ALF    =DATAN ( ALF ) * 57.29578                         07DE0IBM
C                ALF    # ATAN % ALF L * 57.29578                         07DE0CDC
          IF ( ALF ) 8220, 8240, 8240                                     07DE0
 8220            THETA  =  - ALF - 180.0                                  07DE0
          GO TO 8450                                                      08JL8
 8240            THETA  =  + 180.0 - ALF                                  07DE0
          GO TO 8450                                                      08JL8
 8300     IF ( TMY ) 8320, 8340, 8360                                     07DE0
 8320            THETA  =  + 90.0                                         07DE0
          GO TO 8450                                                      08JL8
 8340            THETA  = 0.0                                             08JL8
          GO TO 8450                                                      08JL8
 8360            THETA  =  - 90.0                                         07DE0
          GO TO 8450                                                      08JL8
 8400            ALF    = TMY / BMP                                       07DE0
                 ALF    =DATAN ( ALF )    * 57.29578                      07DE0IBM
C                ALF    # ATAN % ALF L    * 57.29578                      07DE0CDC
                 THETA  = - ALF                                           07DE0
C-----CLOCKWISE ANGLES ARE NEGATIVE                                       15DE7
 8450            BETA   = 0.5 * THETA                                     09FE1
                 SIGO   = PMMAX * SOT2                                    08JL8
                 PSIGO(I,J+2) = SIGO                                      15JL0
C-----      SLAB ONLY CUTPUT
          IF ( IPRINT ) 5980, 8458, 8454                                  15MR1
 8454 PRINT 852, ISTA, JSTA, W(I), BMX(I), BMY(I), TMX, SIGO,             15MR1
     1     BETA, ROCK                                                     09FE1
C-----PRINT BEAM OUTPUT IF COMBINEC
 8458     IF ( SWB ) 5980, 8550, 8460                                     15MR1
 8460     IF ( IPRINT ) 5980, 8550, 8465                                  15MR1
 8465 PRINT 853, BMBX(I), BMBY(I)                                         15MR1
 8550     CONTINUE                                                        08JL8
          IF ( ISW ) 5980, 8600, 8650                                     09FE1
 8600            JN = J - 1                                               24JL8
      CALL DATA        (    L1,    JN,     1,    MXP3,                    09FE1
     1                      DX, DYM1,   DY, DYP1,   FX, FYM1,             24JE8
     2                      FY, FYP1,    Q,    S,                         24JE8
     3                      RX, RYM1, RYP1,   TX, TYM1, TYP1,             24JE8
     4                      CH, CHP1,                    DXP1, DXM1,      05AG9
     5                      PX,   PY, PYP1,  PBX,  PBY, PBYP1,            24JE8
     6                      QM, IPRINT                                    15MR1
          IF ( J - 1 ) 9980, 8610, 8650                                   07DE0
 8610            ISW = 1                                                  07DE0
          DO 8620  I = 1, MXP3                                            09FE1
              WP2(I) = WP1(I)                                             24JL8
              WP1(I) = W(I)                                               05JL8
              W(I)   = WM1(I)                                             05JL8
              WM1(I) = WM2(I)                                             05JL8
              WM2(I) = 0.0                                                24JL8
 8620     CONTINUE                                                        07DE0
          DO 8630  I = 2, MXP2                                            07DE0
              CRD(2) = AMIN1 ( DX(I), DY(I) )                             04AG9
              BMYP1(I)   = BMY(I)                                         08JL8
              BMY(I)     = BMYM1(I)                                       08JL8
              BMYM1(I)   = 0.0                                            24JL8
              BMBYP1(I)  = BMBY(I)                                        08JL8
              BMBY(I)     = BMBYM1(I)                                     08JL8
              BMBYM1(I)   = 0.0                                           24JL8
              WSUM1       = ODHX2 * ( W(I-1) - 2.0 * W(I) + W(I+1) )      08JL8
              WSUM2       = ODHY2 * ( WM1(I) - 2.0 * W(I) + WP1(I) )      08JL8
              BMX(I)      = DX(I) * WSUM1 + CRD(2) * PR * WSUM2           05AG9
              BMBX(I)     = FX(I) * WSUM1                                 08JL8
C-----BMX AND BMY ARE AT STATION 0, PBMX AND PBMY ARE STORED AT J = 2
          IF ( IBOS .EQ. 2 ) GO TC 8625                                   11FE1
              PBMX(I, 2 ) = BMX(I)                                        21AP1
              PBMY(I, 2 ) = BMY(I)                                        21AP1
          GO TO 8630                                                      11FE1
 8625         PBMX(I, 2 ) = BMBX(I)                                       21AP1
              PBMY(I, 2 ) = BMBY(I)                                       21AP1
 8630     CCNTINUE                                                        07DE0
              JSTA = 0                                                    26JL8
              J = 0                                                       15JL0
          GO TO 7300                                                      24JL8
 8650     CONTINUE                                                        07DE0
      PRINT 860, SUMR                                                     13SE8
      PRINT 861, ITEMP, JTEMP, STEMP                                      09FE1
C ******* NO ZOT, SPLOT CNLY, FOR IBM ***********                         30NO1
              IPOP = 0                                                    30NO1IBM
          IF(IPCP .GT. 0 .OR. I3D .EQ. 1)  CALL PLOTS( IBUF,500,10)       30NO1IBM
          IF ( NCT8 )  9980, 8950, 8660                                   15MR1
C
C---- PRINT PROFILE CLTPLT WITH PRINTER PLCTS                             09FE1
C
 8660 PRINT 11                                                            09FE1
      PRINT  1                                                            24SE9
      PRINT 13, AN1                                                       06OC0
      PRINT 16, NPROB, AN2                                                06OC0
      PRINT 864                                                           09FE1
      PRINT 865                                                           09FE1
          DO 8670 I = 1,300                                               12FE1
              XX(I) = I - 1                                               15JE0
 8670     CONTINUE                                                        12FE1
              LGP = -1                                                    15JE0
              NC =  10                                                    15JE0
              IROLL = 1                                                   15JE0
              MOP = 0                                                     15JE0
          IF ( IPOP.EC.3 )      MOP = 1                                   15JE0
              IPOP = IPOP - 1                                             20MY0
C-----PLOT DEFLECTICN AREAS                                               11FE1
          DO 8718  N = 1, NCT8                                            11FE1
          IF ( KASEW(N) ) 5980, 8718, 8702                                11FE1
 8702 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                       11FE1
      PRINT 867                                                           11FE1
              I1 = IN18(N) + 2                                            26FE1
              I2 = IN28(N) + 2                                            26FE1
              J1 = JN18(N) + 2                                            26FE1
              J2 = JN28(N) + 2                                            26FE1
              NENO = (I2-I1+1)*(J2-J1+1)                                  20AP0
              K = 0                                                       20AP0
          IF ( (J2-J1)-(I2-I1) ) 8704, 8704, 8708                         09DE0
 8704     DO  8706  J = J1, J2                                            09DE0
          DO  8706  I = I1, I2                                            09DE0
              K = K + 1                                                   20AP0
              TEMX(K) = PW(I,J)                                           11FE1
 8706     CONTINUE                                                        09DE0
          GO TO 8712                                                      09DE0
 8708     DO  8710  I = I1, I2                                            09DE0
          DO  8710  J = J1, J2                                            09DE0
              K = K + 1                                                   20AP0
              TEMX(K) = PW(I,J)                                           11FE1
 8710     CONTINUE                                                        09DE0
 8712     IF ( IPOP ) 8714, 8716, 8716                                    09DE0
 8714 CALL SPLOT 3   ( TEMX, NENO, 20.0 )                                 03MA1
          GO TO 8718                                                      09DE0
 8716     CCNTINLE                                                        09DE0
              ID = 101                                                    15JE0
      CALL ZOT 1  ( XX, TEMX, NENO, 10 )                                  15JE0
              NP2 = NENO + 2                                              19JE0
          DO 8717  I = 1, NP2                                             09DE0
 8717         XX(I) = I - 1                                               09DE0
```

```
      IF ( IPOP.EQ.I )    GO TC  8714                               15MR1
 8718     CONTINUE                                                  09DE0
C----PLOT MOMENT AREAS - SLAB IF IBOS = 1, BEAM IF IBOS = 2         11FE1
      DO  8780  N = 1, NCT8                                         11FE1
      IF ( KASEX(N) + KASEY(N) ) 9980, 8724, 8721                   11FE1
 8721     IF ( KASEX(N) - 1 ) 9980, 8722, 8723                      11FE1
 8722 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 868, SLBX, SLBY, BMOM, SLBX, SLBY                       11FE1
      GO TO 8732                                                    11FE1
 8723 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 868, BEMX, BEMY, BMOM, BEMX, BEMY                       11FE1
      GO TO 8732                                                    11FE1
 8724     IF ( KASEX(N) - 1 ) 8727, 8725, 8726                      11FE1
 8725 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 869, SLBX, BMCM, SLBX                                   11FE1
      GO TO 8732                                                    11FE1
 8726 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 869, BEMX, BMCM, BEMX                                   11FE1
      GO TO 8732                                                    11FE1
 8727     IF ( KASEY(N) - 1 ) 8732, 8728, 8729                      11FE1
 8728 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 869, SLBY, BMCM, SLBY                                   11FE1
      GO TO 8732                                                    11FE1
 8729 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 869, BEMY, BMCM, BEMY                                   11FE1
 8732     CONTINUE                                                  09DE0
              I1 = IN18(N) + 2                                      09DE0
              I2 = IN28(N) + 2                                      09DE0
              J1 = JN18(N) + 2                                      09DE0
              J2 = JN28(N) + 2                                      11DE0
              NEND = (I2-I1+1)*(J2-J1+1)                            11NO9
              K = 0                                                 25NO9
      DO  8736  J = J1, J2                                          11DE0
      DO  8734  I = I1, I2                                          26FE1
          K = K + 1                                                 25NO9
          TEMX(K) = PBMX(I,J)                                       11FE1
          TEMY(K) = PBMY(I,J)                                       11FE1
 8734     CONTINUE                                                  11DE0
 8736     CONTINUE                                                  11DE0
      IF ( IPOP ) 8738, 8765, 8765                                  11DE0
 8738     IF ( KASEX(N) + KASEY(N) ) 9980, 8742, 8740               11FE1
C----PLOT X AND Y MOMENTS                                           11FE1
 8740 CALL SPLOT 4    ( TEMX, TEMY, NEND )                          03MA1
      GO TO 8780                                                    11DE0
 8742     IF ( KASEX(N) - 1 ) 8744, 8743, 8743                      11FE1
C----PLOT X MOMENTS ONLY                                            11FE1
 8743 CALL SPLOT 3    ( TEMX, NEND, 20.0 )                          03MA1
      GO TO 8780                                                    11DE0
 8744     IF ( KASEY(N) - 1 ) 8780, 8746, 8746                      11FE1
 8746     IF ( (J2-J1)-(I2-I1) ) 8754, 8754, 8748                   11DE0
 8748         K = 0                                                 11DE0
      DO  8752  I = I1, I2                                          26FE1
      DO  8750  J = J1, J2                                          11DE0
          K = K + 1                                                 12DE9
          TEMY(K) = PBMY(I,J)                                       11FE1
 8750     CONTINUE                                                  11DE0
 8752     CONTINUE                                                  11DE0
C----PLOT Y MOMENTS CNLY                                            11FE1
 8754 CALL SPLOT 3    ( TEMY, NEND, 20.0 )                          03MA1
      GO TO 8780                                                    13JL1
 8765     CONTINUE                                                  11DE0
      IF ( KASEX(N).GE.1 )      ID = ID2                            12FE1
      IF ( KASEX(N).GE.1 )      CALL ZCT 1  ( XX, TEMX, NEND, ID )  12FE1
          NP2 = NEND + 2                                            19JE0
      DO  8770  I = 1, NP2                                          11DE0
 8770     XX(I) = I - 1                                             11DE0
      IF ( KASEY(N).GT.1 )      ID = ID3                            11FE1
      IF ( KASEY(N).GT.1 )      CALL ZCT 1  ( XX, TEMY, NEND, ID )  11FE1
      DO  8772  I = 1, NP2                                          11DE0
 8772     XX(I) = I - 1                                             11DE0
      IF ( IPOP.EQ.I )    GO TO  8738                               11DE0
 8780     CCNTINUE                                                  11DE0
C----PLCT PRINCIPAL MCMENT CR STRESS AREAS                           
      DO  8884  N = 1, NCT8                                         11FE1
```

```
      IF (KASEP(N) ) 9980, 8884, 8781                               11FE1
 8781     IF ( IOPPS ) 99E0, 8782, 8783                             13JL1
 8782 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 870, BMCM, BMOM                                         11FE1
      GC TO 8784                                                    11FE1
 8783 PRINT 866, IN18(N), JN18(N), IN28(N), JN28(N)                 11FE1
      PRINT 870, STRS, STRS                                         11FE1
 8784     CONTINUE                                                  11DE0
              I1 = IN18(N) + 2                                      26FE1
              I2 = IN28(N) + 2                                      26FE1
              J1 = JN18(N) + 2                                      26FE1
              J2 = JN28(N) + 2                                      26FE1
              NEND = (I2-I1+1)*(J2-J1+1)                            12MY0
              K = 0                                                 12MY0
      IF ( (J2-J1) - (I2-I1) ) 8785, 8785, 8787                     11DE0
 8785     DO  8786  J = J1, J2                                      11DE0
          DO  8786  I = I1, I2                                      11DE0
              K = K + 1                                             02JE0
              TEMX(K) = PSIGO(I,J)                                  11FE1
 8786     CONTINUE                                                  11DE0
          GO TO 8790                                                11DE0
 8787     DO  8788  I = I1, I2                                      11DE0
          DO  8788  J = J1, J2                                      11DE0
              K = K + 1                                             02JE0
              TEMX(K) = PSIGO(I,J)                                  11FE1
 8788     CCNTINUE                                                  11DE0
 879C     CONTINUE                                                  11DE0
      IF ( IPOP ) 8865, 8870, 8870                                  11DE0
 8865 CALL SPLOT 3    ( TEPX, NEND, 20.0 )                          03MA1
      GO TO 8884                                                    11DE0
 8870     CONTINUE                                                  11DE0
              ID = ID4                                              15JE0
      CALL ZOT 1  ( XX, TEPX, NEND, ID )                            15JE0
              NP2 = NEND + 2                                        19JE0
          DC  8872  I = 1, NP2                                      11DE0
 8872         XX(I) = I - 1                                         11DE0
      IF ( IPOP.EC.1 )     GO TC  8865                              11DE0
 8884     CONTINUE                                                  11DE0
 88E5     CONTINUE                                                  11DE0
 8950     CONTINUE                                                  12MY0
      IF ( I3D.EQ.1 )                                               260C0
     *CALL DIM3      (PL, L1, L2, PX, MY, HX, HY, VEF , ROF, SLOPE, 26FE1
     1                  IPOP, NPRC8, IGSW )                         06MY1
      CALL TIC TOC (4)                                              25SE6
          GO TO 1010                                                26AG3
 9980 PRINT 980                                                     19JE8
 9990     CONTINUE                                                  19MR5
 9999     CONTINUE                                                  04MY3
      IF ( IPOP.GT.O .OR.I4D.GE.1 ) CALL PLOT(O., O., 999)          06DE118M
C     IF X IPOP.GT.O .OR.I3O.EQ.1 L CALL ENDPLT                     14MY1COC
      PRINT 11                                                      08MY3
      PRINT I                                                       21JL7
      PRINT 13, AN1                                                 060C0
      CALL TIC TOC (2)                                              26SE6
      PRINT 19                                                      26AG3
      RETURN                                                        23NO1
      END
```

SUBROUTINE STIFF

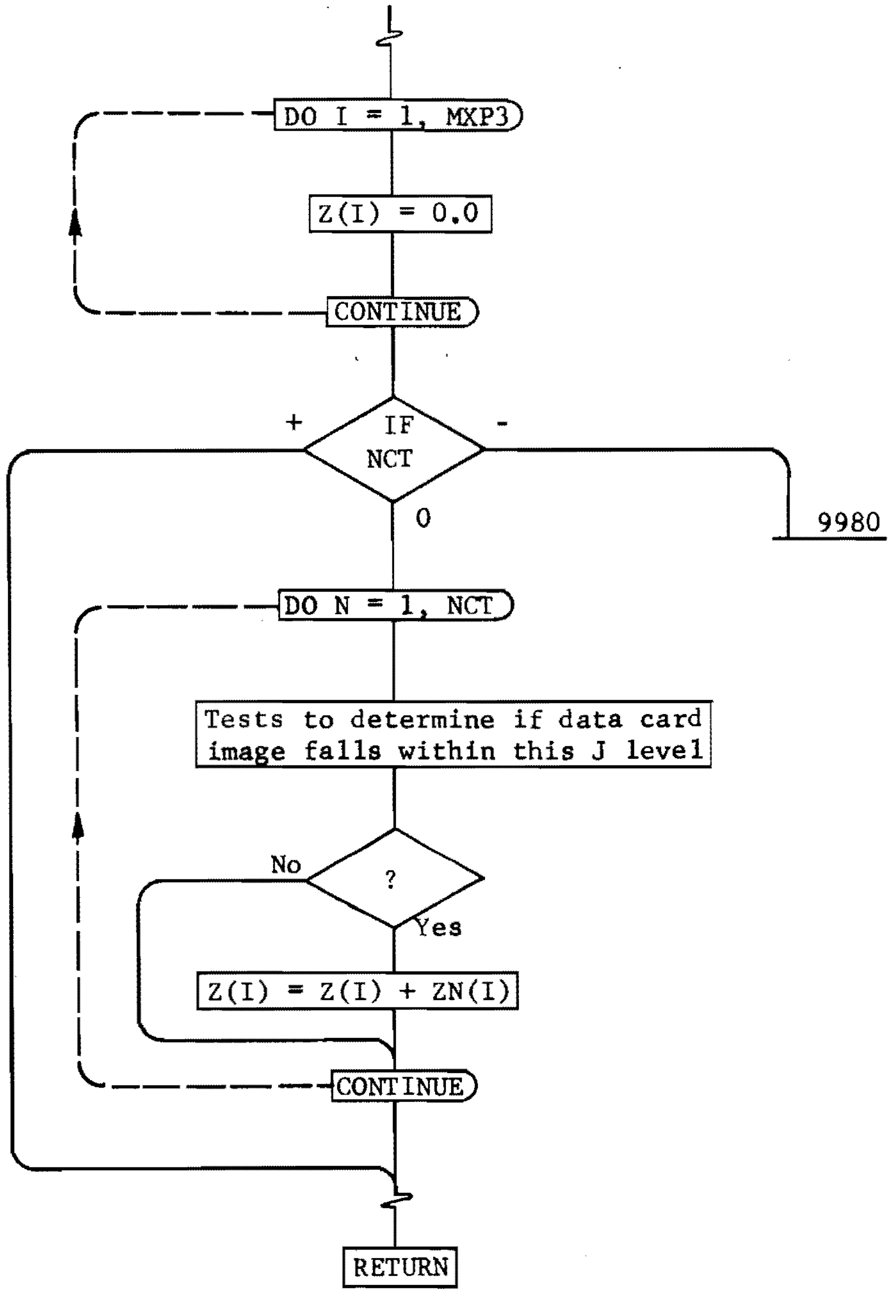


```
      SUBROUTINE STIF    ( L1     ,  AA     ,  ET1   ,  CC     ,  DD     ,      23NO1
     1                    EE,    FF,    ML,    J1,    N1,    N2,    N3,        26SE8
     2                    DX, DYM1,    DY, DYP1,    FX, FYM1,                  26SE8
     3                    FY, FYP1,     Q,     S,                              26SE8
     4                    RX, RYM1, RYP1,    TX, TYM1, TYP1,    CH, CHP1,      26SE8
     5                    PX,    PY, PYP1,   PBX,  PBY,      PBYP1,     QM,    29JL9
     6                   DXP1, DXM1 )                                          05AG9
      DOUBLE PRECISION  PDHXHY, HYDHX3, ODHX3, ODHX2, ODHX,                    24MY11BM
     1                  ODHXHY, HXDHY3, ODHY3, ODHY2, ODHY, CRD                24MY11BM
      DOUBLE PRECISION  AA, ET1, CC, DD, EE, FF                                23NO11BM
      DIMENSION AA( L1 , 1  ),       ET1( L1 , 1  ),       CC( L1 , 5  ),      26SE8
     1          DD( L1 , 3  ),        EE( L1 , 1  ),       FF( L1 , 1  )       26SE8
      DIMENSION DX( L1 ),           DXM1( L1 ),          DXP1( L1 ),           03MA1
     1          DY( L1 ),           DYM1( L1 ),          DYP1( L1 ),           03MA1
     2          FX( L1 ),                                                      03MA1
     3          FY( L1 ),           FYM1( L1 ),          FYP1( L1 ),           03MA1
     4           Q( L1 ),              S( L1 ),                                03MA1
     5          RX( L1 ),           RYM1( L1 ),          RYP1( L1 ),           03MA1
     6          TX( L1 ),           TYM1( L1 ),          TYP1( L1 ),           03MA1
     7          CH( L1 ),           CHP1( L1 ),                                03MA1
     8          PX( L1 ),             PY( L1 ),          PYP1( L1 ),           03MA1
     9         PBX( L1 ),            PBY( L1 ),         PBYP1( L1 ),           03MA1
     A          QM( L1 )                                                       10OC9
C
      COMMON / RI / MXP3, MYP3, NF, ITMPP                                      02MA1
      COMMON /STIFF/PDHXHY, HYDHX3, ODHX3, ODHX2, ODHX,                        02MA1
     1              ODHXHY, HXDHY3, ODHY3, ODHY2, ODHY, CRD(5)                 24MY1
C
C-----A SPRING IS PLACED AT PTS BEYOND BOUNORIES OF THE REAL SLAB
C        TO MAKE SOLUTION OF NON-RECTANGULAR SLABS OR SLABS WITH
C        HOLES POSSIBLE. THIS IS DONE BY TESTING ON THE CC$I,3L
C        TERMS, AND IF ZERO, SET EQUAL TO 1.0                                  12FE1
C
   98 FORMAT ( //40H       ***  UNDESIGNATED ERROR STOP  ***   )               25SE8
               JN = J1 - 2                                                     13SE8
      CALL DATA        (   L1,    JN,    ML,    MXP3,                          12FE1
     1                    DX, DYM1,    DY, DYP1,    FX, FYM1,                  24JE8
     2                    FY, FYP1,     Q,     S,                              24JE8
     3                    RX, RYM1, RYP1,    TX, TYM1, TYP1,                   24JE8
     4                    CH, CHP1,                         DXP1, DXM1,        05AG9
     5                    PX,    PY, PYP1,   PBX,   PBY, PBYP1,                24JE8
     6                    QM, IPRINT                                     )     15MR1
         DO 2407  I = 1, MXP3                                                  18JL8
         IF ( I - 1 ) 9980, 2402, 2403                                         18JL8
 2402        FF(I,1)     = 0.5 * ODHX * TX(I+1)                                18JL8
         GO TO 2406                                                            18JL8
 2403    IF ( MXP3 - I ) 9980, 2405, 2404                                      18JL8
 2404        FF(I,1)     = Q(I) + QM(I)                                        24JE8
     1                         + 0.5 * ODHX * ( - TX(I-1) + TX(I+1) )          24JE8
     2                         + 0.5 * ODHY * ( - TYM1(I) + TYP1(I) )          24JE8
         GO TO 2406                                                            18JL8
 2405        FF(I,1)     = - 0.5 * ODHX * TX(I-1)                              18JL8
 2406        AA(I,1)     = HXDHY3 * DYM1(I) + ODHY3 * FYM1(I)                  18JL8
     1                         - 0.25 * ODHY2 * RYM1(I)                        24JE8
 2407    CONTINUE                                                              18JL8
         IF ( ML ) 2700, 2408, 2408                                            13SE8
 2408    DO 2500  I = 1, MXP3                                                  18JL8
         IF ( I - 1 ) 9980, 2421, 2422                                         13SE8
 2421        CC(I,3)     = HYDHX3 * DX(I+1) + ODHX3 * FX(I+1)                  25SE8
     1                         + 0.25 * ODHX2 * RX(I+1)                        24JE8
             CRD(3) = AMIN1 ( DX(I+1)   , DY(I+1) )                            29JL9
         IF ( CC(I,3) ) 2441, 2440, 2441                                       24JE8
 2440        CC(I,3)     = 1.0                                                 29JL9
 2441        CC(I,4)     = - 2.0 * ( HYDHX3 * DX(I+1) + ODHX3 * FX(I+1)21JE8
     1                         + PDHXHY * CRD(3)  )                            29JL9
     2                         - ODHX * ( PX(I+1) + PBX(I+1) )                 24JE8
             CC(I,5)     = HYDHX3 * DX(I+1) + ODHX3 * FX(I+1)                  24JE8
     1                         - 0.25 * ODHX2 * RX(I+1)                        24JE8
             DD(I,2)     = 0.D                                                 08DE7
             DD(I,1)     = PDHXHY * CRD(3)                                     29JL9
             ET1(I,1)    = EE(I,1)                                             25SE8
             EE(I,1)     = D.D                                                 03JA8
         GO TO 2500                                                            24JE8
 2422    IF ( MXP3 - I ) 9980, 2429, 2423                                      24JE8
 2423        CRD(1) = AMIN1 ( DX(I-1)   , DY(I-1) )                            29JL9
             CRD(2) = AMIN1 ( DX(I  )   , DY(I  ) )                            29JL9
             CRD(3) = AMIN1 ( DX(I+1)   , DY(I+1) )                            29JL9
             CRD(5) = AMIN1 ( DXP1(I  ), DYP1(I) )                             29JL9
             CC(I,2)     = - 2.0 * ( HYDHX3 * ( DX(I-1) + DX(I) )              29JL9
     1                         + ODHX3 * ( FX(I-1) + FX(I) )                   19JE8
     2                         + PDHXHY * ( CRD(1) + CRD(2)  )                 29JL9
     3                         + ODHXHY * ( CH(I) + CHP1(I) ) )                19JE8
     4                         - ODHX * ( PX(I) + PBX(I) )                     19JE8
             CC(I,3)     = HYDHX3 * ( DX(I-1) +4.0D0* DX(I) + DX(I+1) )19JE81BM
     1                         + ODHX3 * ( FX(I-1) +4.0D0* FX(I) +             19JE81BM
     2                         FX(I+1) ) + HXDHY3 * ( DYM1(I) +                19JE81BM
     3                        4.0D0* DY(I) + DYP1(I) ) + ODHY3 *               21JE81BM
     4                         ( FYM1(I) +4.0D0* FY(I) + FYP1(I) )             19JE81BM
     5                         + PDHXHY * 4.0 * ( CRD(2) +CRD(2)  )            29JL91BM
     6                         + ODHXHY * 2.0 * ( CH(I) + CH(I+1) + 0.0D019JE81BM
     7                         + CHP1(I) + CHP1(I+1 ) )                        19JE81BM
     8                         + ODHX * ( PX(I) + PX(I+1) + PBX(I)+0.0D0  21JE81BM
     9                       + PBX(I+1) ) + ODHY * ( PY(I) + PYP1(I) +         23NO11BM
     A                           0.0D0 + PBY(I) + PBYP1(I) ) + S(I)            19JE81BM
     B                         - 0.25 * ODHX2 * ( - RX(I-1) - RX(I+1) )        19JE81BM
     C                         - 0.25 * ODHY2 * ( - RYM1(I) - RYP1(I) )        19JE81BM
C            CC$I,3L     # HYDHX3 * $ DX$I-1L & 4.0 * DX$IL & DX$I&1L L19JE8CDC
C    1                         & ODHX3 * $ FX$I-1L & 4.0 * FX$IL &             19JE8CDC
C    2                         FX$I&1L L & HXDHY3 * $ DYM1$IL &                19JE8CDC
C    3                         4.0 * DY$IL & DYP1$IL L & ODHY3 *               21JE8CDC
C    4                         $ FYM1$IL & 4.D * FY$IL & FYP1$IL L             19JE8CDC
C    5                         & PDHXHY * 4.D * $ CRD$2L &CRD$2L   L           29JL9CDC
C    6                         & CDHXHY * 2.D * $ CH$IL & CH$I&1L              19JE8CDC
C    7                         & CHP1$IL & CHP1$I&1 L L                        19JE8CDC
C    8                         & ODHX * $ PX$IL & PX$I&1L & PBX$IL &           21JE8CDC
C    9                         PBX$I&1L L & ODHY * $ PY$IL & PYP1$IL &         19JE8CDC
C    A                                PBY$IL & PBYP1$IL L & S$IL               19JE8CDC
C    B                         - 0.25 * ODHX2 * $ - RX$I-1L - RX$I&1L L        19JE8CDC
C    C                         - 0.25 * ODHY2 * $ - RYM1$IL - RYP1$IL L        19JE8CDC
         IF ( CC(I,3) ) 2425, 2424, 2425                                       21JE8
 2424        CC(I,3)     = 1.0                                                 29JL9
 2425        CC(I,4)     = - 2.0 * ( HYDHX3 * ( DX(I) + DX(I+1) )              05AG8
     1                         + ODHX3 * ( FX(I) + FX(I+1) )                   05AG8
     2                         + PDHXHY * ( CRD(3) + CRD(2)  )                 29JL9
     3                         + ODHXHY * ( CH(I+1) + CHP1(I+1) ) )            19JE8
     4                         - ODHX * ( PX(I+1) + PBX (I+1) )                19JE8
             DD(I,3)     = PDHXHY * ( CRD(1) + CRD(5)  )                       29JL9
     1                         + ODHXHY * 2.0 * CHP1(I)                        19JE8
             DD(I,2)     = - 2.0 * ( HXDHY3 * ( DY(I) + DYP1(I) )              05AG8
     1                         + ODHY3 * ( FY(I) + FYP1(I) )                   05AG8
     2                         + PDHXHY * ( CRD(2) + CRD(5)  )                 29JL9
     3                         + ODHXHY * ( CHP1(I) + CHP1(I+1) ) )            21JE8
     4                         - ODHY * ( PYP1(I) + PBYP1(I) )                 19JE8
             DD(I,1)     = PDHXHY * ( CRD(3) + CRD(5)  )                       29JL9
     1                         + ODHXHY * 2.0 * CHP1(I+1)                      19JE8
             ET1(I,1)    = EE(I,1)                                             25SE8
             EE(I,1)     = HXDHY3 * DYP1(I) + ODHY3 * FYP1(I)                  19JE8
     1                         - 0.25 * ODHY2 * RYP1(I)                        19JE8
         IF ( I - 2 ) 9980, 2427, 2426                                         19JE8
 2426        CC(I,1)     = HYDHX3 * DX(I-1) + ODHX3 * FX(I-1)                  24JE8
     1                         - 0.25 * ODHX2 * RX(I-1)                        19JE8
 2427    IF ( MXP3 - 1 - I ) 9980, 2500, 2428                                  02JU1
 2428        CC(I,5)     = HYDHX3 * DX(I+1) + ODHX3 * FX(I+1)                  19JE8
     1                         - 0.25 * ODHX2 * RX(I+1)                        19JE8
         GO TO 2500                                                            19JE8
 2429        CRD(1) = AMIN1 ( DX(I-1)   , DY(I-1) )                            29JL9
             CC(I,1)     = HYDHX3 * DX(I-1) + ODHX3 * FX(I-1)                  25SE8
     1                         - 0.25 * ODHX2 * RX(I-1)                        21JE8
             CC(I,2)     = - 2.0 * ( HYDHX3 * DX(I-1) + ODHX3 * FX(I-1)19JE8
     1                         + PDHXHY * CRD(1)  )                            29JL9
             CC(I,3)     = HYDHX3 * DX(I-1) + ODHX3 * FX(I-1)                  21JE8
     1                         + 0.25 * ODHX2 * RX(I-1)                        19JE8
         IF ( CC(I,3) ) 2444, 2443, 2444                                       19JE8
 2443        CC(I,3)     = 1.0                                                 29JL9
 2444        DD(I,3)     = PDHXHY * CRD(1)                                     29JL9
             DD(I,2)     = 0.0                                                 08DE7
             ET1(I,1)    = EE(I,1)                                             25SE8
             EE(I,1)     = 0.0                                                 03JA8
```

```
2500     CONTINUE                                                       19JE8
         DO 2610  L = 2, MXP3                                           13SE8
              J = MXP3 - L + 2                                          21OC8
              DD(J,1)      = DD(J-1,1)                                  13SE8
2610     CONTINUE                                                       13SE8
              DD(1,1)      = 0.0                                        13SE8
         DO 2620  J = 1, MXP3                                           13SE8
              DD(J,3)      = DD(J+1,3)                                  13SE8
2620     CONTINUE                                                       13SE8
              DD(MXP3,3) = 0.0                                          13SE8
              DD(1,1)      = DD(1,2)                                    13SE8
              DD(1,2)      = DD(1,3)                                    13SE8
              DD(1,3)      = 0.0                                        13SE8
              CC(1,1)      = CC(1,3)                                    13SE8
              CC(1,2)      = CC(1,4)                                    13SE8
              CC(1,3)      = CC(1,5)                                    13SE8
              CC(2,1)      = CC(2,2)                                    13SE8
              CC(2,2)      = CC(2,3)                                    13SE8
              CC(2,3)      = CC(2,4)                                    13SE8
              CC(2,4)      = CC(2,5)                                    13SE8
              CC(2,5)      = 0.0                                        13SE8
              CC(1,4)      = 0.0                                        13SE8
              CC(1,5)      = 0.0                                        13SE8
         I = MXP3                                                       13SE8
              DD(I,3)      = DD(I,2)                                    13SE8
              DD(I,2)      = DD(I,1)                                    13SE8
              DD(I,1)      = 0.0                                        13SE8
              CC(I,5)      = CC(I,3)                                    13SE8
              CC(I,4)      = CC(I,2)                                    13SE8
              CC(I,3)      = CC(I,1)                                    13SE8
              CC(I-1,5)    = CC(I-1,4)                                  13SE8
              CC(I-1,4)    = CC(I-1,3)                                  13SE8
              CC(I-1,3)    = CC(I-1,2)                                  13SE8
              CC(I-1,2)    = CC(I-1,1)                                  13SE8
              CC(I-1,1)    = 0.0                                        13SE8
              CC(I,2)      = 0.0                                        13SE8
              CC(I,1)      = 0.0                                        13SE8
2700     CONTINUE                                                       13SE8
      RETURN                                                            13SE8
9980  PRINT 98                                                          25SE8
      END                                                               13SE8
```

SUBROUTINE DATA

This routine is called at each J (y-direction) level to distribute data from tables 3, 4, 5, 6, 7, and 9. Z(I) refers to the distributed value in each table. ZN(I) is the data input from the data cards.



```
      SUBROUTINE DATA (   L1,   JN,   ML,   MXP3,                          11DE7
     1                    DX, DYM1,   DY, DYP1,   FX, FYM1,                 24JE8
     2                    FY, FYP1,    Q,    S,                             24JE8
     3                    RX, RYM1, RYP1,   TX, TYM1, TYP1,                 24JE8
     4                    CH, CHP1,                     DXP1, DXM1,         05AG9
     5                    PX,    PY, PYP1,  PBX,  PBY, PBYP1,               24JE8
     6                    QM, IPRINT                                    )   15MR1
C
C-----THIS SUBROUTINE IS CALLED AT EACH J STEP IN THE STIFFNESS MATRIX  12JE8
C         GENERATION AND AGAIN AT EACH J STEP WHEN COMPUTING RESULTS.   12JE8
C
      DIMENSION DX( L1 ),          DXM1( L1 ),          DXP1( L1 ),     10OC9
     1          DY( L1 ),          DYM1( L1 ),          DYP1( L1 ),     10OC9
     2          FX( L1 ),                                                10OC9
     3          FY( L1 ),          FYM1( L1 ),          FYP1( L1 ),     10OC9
     4           Q( L1 ),             S( L1 ),                           10OC9
     5          RX( L1 ),          RYM1( L1 ),          RYP1( L1 ),     10OC9
     6          TX( L1 ),          TYM1( L1 ),          TYP1( L1 ),     10OC9
     7          CH( L1 ),          CHP1( L1 ),                           10OC9
     8          PX( L1 ),            PY( L1 ),          PYP1( L1 ),     10OC9
     9         PBX( L1 ),           PBY( L1 ),         PBYP1( L1 ),     10OC9
     A          QM( L1 )                                                 10OC9
      COMMON /CARDS/ AN1(40),  AN2(18),                                  12FE1
     1          IN13(300), JN13(300), IN23(300), JN23(300), DXN(300),    23SE0
     2           DYN(300),  FXN(300),  FYN(300),   QN(300),  SN(300),    23SE0
     3          IN14( 50), JN14( 50), IN24( 50), JN24( 50),              23SE0
     4           RXN( 50),  RYN( 50),  TXN( 50),  TYN( 50),              23SE0
     5          IN15(100), JN15(100), IN25(100), JN25(100), CHN(100),    23SE0
     6          IN16( 50), JN16( 50), IN26( 50), JN26( 50),              23SE0
     7           PXN( 50),  PYN( 50), PBXN( 50), PBYN( 50),              23SE0
     8          IN17(100), JN17(100), IN27(100), JN27(100), QMN(100),    03MA1
     9          IN18( 10), JN18( 10), IN28( 10), JN28( 10),              17FE1
     A        KASEW( 10),KASEX( 10),KASEY( 10),KASEP( 10),               17FE1
     B                   JN19( 10),            JN29( 10)                 17FE1
      COMMON /TABLE/NC13, NCT3, NC14, NCT4, NC15, NCT5, NC16, NCT6,      10OC9
     1              NC17, NCT7, NC18, NCT8, NC19, NCT9                   15MR1
C
   98 FORMAT ( //30H     UNDESIGNATED ERROR STOP     )                   21AG8
C
C-----DISTRIBUTE DATA FROM TABLE 3                                       12JE8
C
  300     DO 305  I = 1, MXP3                                            12JE8
             DX(I) = 0.0                                                 12JE8
             DXP1(I) = 0.0                                               29JL9
             DXM1(I) = 0.0                                               05AG9
             DYM1(I) = 0.0                                               28JE8
             DY(I) = 0.0                                                 12JE8
             DYP1(I) = 0.0                                               12JE8
             FX(I) = 0.0                                                 12JE8
             FYM1(I) = 0.0                                               12JE8
             FY(I) = 0.0                                                 12JE8
             FYP1(I) = 0.0                                               30MY8
             Q(I) = 0.0                                                  12JE8
             S(I) = 0.0                                                  12JE8
  305     CONTINUE                                                       28JE8
          IF ( NCT3 ) 980, 400, 310                                      21AG8
  310     DO 360  N = 1, NCT3                                            28JE8
             I1 = IN13(N) + 2                                            02JL8
             I2 = IN23(N) + 2                                            12JE8
          IF ( JN - JN13(N) ) 345, 315, 315                              02JL8
  315     IF ( JN23(N) - JN ) 330, 320, 320                              09JL8
  320     DO 325  I = I1, I2                                             02JL8
             DX(I) = DX(I) + DXN(N)                                      12JE8
             DY(I) = DY(I) + DYN(N)                                      12JE8
             FX(I) = FX(I) + FXN(N)                                      12JE8
             FY(I) = FY(I) + FYN(N)                                      12JE8
             Q(I) = Q(I) + QN(N)                                         21JE8
             S(I) = S(I) + SN(N)                                         12JE8
  325     CONTINUE                                                       02JL8
  330     IF ( (JN-1) - JN13(N) ) 345, 333, 333                          09JL8
  333     IF ( JN23(N) - (JN-1) ) 345, 335, 335                          09JL8
  335     DO 340  I = I1, I2                                             02JL8
             DYM1(I) = DYM1(I) + DYN(N)                                  02JL8
             DXM1(I) = DXM1(I) + DXN(N)                                  05AG9
             FYM1(I) = FYM1(I) + FYN(N)                                  12JE8
  340     CONTINUE                                                       02JL8
  345     IF ( (JN+1) - JN13(N) ) 360, 347, 347                          03JL8
  347     IF ( JN23(N) - (JN+1) ) 360, 350, 350                          02JL8
  350     DO 355  I = I1, I2                                             02JL8
             DXP1(I) = DXP1(I) + DXN(N)                                  29JL9
             DYP1(I) = DYP1(I) + DYN(N)                                  02JL8
             FYP1(I) = FYP1(I) + FYN(N)                                  12JE8
  355     CONTINUE                                                       02JL8
  360     CONTINUE                                                       12JE8
C
C-----DISTRIBUTE DATA FROM TABLE 4                                       12JE8
C
  400     DO 405  I = 1, MXP3                                            28JE8
             RX (I) = 0.0                                                12JE8
             RYM1(I) = 0.0                                               12JE8
             RYP1(I) = 0.0                                               12JE8
             TX(I) = 0.0                                                 12JE8
             TYM1(I) = 0.0                                               12JE8
             TYP1(I) = 0.0                                               12JE8
  405     CONTINUE                                                       28JE8
          IF ( NCT4 ) 980, 500, 410                                      15MR1
  410     DO 460  N = 1, NCT4                                            21JE8
             I1 = IN14(N) + 2                                            03JL8
             I2 = IN24(N) + 2                                            12JE8
          IF ( JN - JN14(N) ) 445, 415, 415                              03JL8
  415     IF ( JN24(N) - JN ) 430, 420, 420                              21AG8
  420     DO 425  I = I1, I2                                             02JL8
             RX(I) = RX(I) + RXN(N)                                      12JE8
             TX(I) = TX(I) + TXN(N)                                      12JE8
  425     CONTINUE                                                       02JL8
  430     IF ( (JN-1) - JN14(N) ) 445, 433, 433                          21AG8
  433     IF ( JN24(N) - (JN-1) ) 445, 435, 435                          21AG8
  435     DO 440  I = I1, I2                                             21AG8
             RYM1(I) = RYM1(I) + RYN(N)                                  02JL8
             TYM1(I) = TYM1(I) + TYN(N)                                  12JE8
  440     CONTINUE                                                       21AG8
  445     IF ( (JN+1) - JN14(N) ) 460, 447, 447                          03JL8
  447     IF ( JN24(N) - (JN+1) ) 460, 450, 450                          03JL8
  450     DO 455  I = I1, I2                                             02JL8
             RYP1(I) = RYP1(I) + RYN(N)                                  02JL8
             TYP1(I) = TYP1(I) + TYN(N)                                  12JE8
  455     CONTINUE                                                       02JL8
  460     CONTINUE                                                       12JE8
C
C-----DISTRIBUTE DATA FROM TABLE 5                                       12JE8
C
  500     DO 505  I = 1, MXP3                                            28JE8
             CH(I) = 0.0                                                 12JE8
             CHP1(I) = 0.0                                               12JE8
  505     CONTINUE                                                       28JE8
          IF ( NCT5 ) 980, 600, 510                                      21AG8
  510     DO 560  N = 1, NCT5                                            21AG8
             I1 = IN15(N) + 2                                            03JL8
             I2 = IN25(N) + 2                                            12JE8
          IF ( JN - JN15(N) ) 545, 515, 515                              03JL8
  515     IF ( JN25(N) - JN ) 545, 520, 520                              03JL8
  520     DO 525  I = I1, I2                                             02JL8
             CH(I) = CH(I) + CHN(N)                                      12JE8
  525     CONTINUE                                                       02JL8
  545     IF ( (JN+1) - JN15(N) ) 560, 547, 547                          03JL8
  547     IF ( JN25(N) - (JN+1) ) 560, 550, 550                          03JL8
  550     DO 555  I = I1, I2                                             02JL8
             CHP1(I) = CHP1(I) + CHN(N)                                  03JL8
  555     CONTINUE                                                       02JL8
  560     CONTINUE                                                       12JE8
C
C-----DISTRIBUTE DATA FROM TABLE 6                                       12JE8
C
  600     DO 605  I = 1, MXP3                                            28JE8
             PX(I) = 0.0                                                 12JE8
             PY(I) = 0.0                                                 12JE8
             PYP1(I) = 0.0                                               12JE8
             PBX(I) = 0.0                                                12JE8
```

```
             PBY(I) = 0.0                                              12JE8
             PBYP1(I) = 0.0                                            12JE8
  605    CONTINUE                                                      28JE8
         IF ( NCT6 ) 960, 700, 610                                     21AG8
  610    DO 660  N = 1, NCT6                                           12JE8
             I1 = IN16(N) + 2                                          03JL8
             I2 = IN26(N) + 2                                          12JE8
         IF ( JN - JN16(N) ) 645, 615, 615                             21AG8
  615    IF ( JN26(N) - JN ) 645, 620, 620                             30AG8
  620    DO 625  I = I1, I2                                            02JL8
             PX(I) = PX(I) + PXN(N)                                    12JE8
             PY(I) = PY(I) + PYN(N)                                    12JE8
             PBX(I) = PBX(I) + PBXN(N)                                 12JE8
             PBY(I) = PBY(I) + PBYN(N)                                 12JE8
  625    CONTINUE                                                      02JL8
  645    IF ( (JN+1) - JN16(N) ) 660, 647, 647                         21AG8
  647    IF ( JN26(N) - (JN+1) ) 660, 650, 650                         21AG8
  650    DO 655  I = I1, I2                                            21AG8
             PYP1(I) = PYP1(I) + PYN(N)                                03JL8
             PBYP1(I) = PBYP1(I) + PBYN(N)                             12JE8
  655    CONTINUE                                                      21AG8
  660    CONTINUE                                                      12JE8
C
C-----DISTRIBUTE DATA FROM TABLE 7
C
  700    DO 705  I = 1, MXP3                                           21AG8
             QM(I) = 0.0                                               12JE8
  705    CONTINUE                                                      21AG8
         IF ( NCT7 ) 980, 800, 710                                     21AG8
  710    DO 760  N = 1, NCT7                                           21AG8
             I1 = IN17(N) + 2                                          21AG8
             I2 = IN27(N) + 2                                          21AG8
         IF ( JN - JN17(N) ) 760, 715, 715                             21AG8
  715    IF ( JN27(N) - JN ) 760, 720, 720                             21AG8
  720    DO 725  I = I1, I2                                            21AG8
             QM(I) = QM(I) + QMN(N)                                    21JE8
  725    CONTINUE                                                      21AG8
  760    CONTINUE                                                      21AG8
  800    CONTINUE                                                      21AG8
C
C-----DISTRIBUTE DATA FROM TABLE 9
C
  900        IPRINT = 0                                                16MR1
         IF ( NCT9 ) 980, 965, 910                                     16MR1
  910    DO 960  N = 1, NCT9                                           15MR1
         IF ( JN - JN19(N) ) 920, 965, 915                             16MR1
  915    IF ( JN29(N) - JN ) 920, 965; 965                             16MR1
  920        IPRINT = 0                                                15MR1
  960    CONTINUE                                                      15MR1
         GO TO 975                                                     16MR1
  965        IPRINT = 1                                                16MR1
  975    CONTINUE                                                      15MR1
      RETURN                                                           12JE8
  980 PRINT 98                                                         21AG8
      END                                                              12JE8
```

PRINTER PLOT ROUTINES

These subroutines generate plots of the selected profile output areas specified by Table 8. Subroutines SPLOT3 and SPLOT4 control the printer plots. SPLOT 3 is used for single column plots and SPLOT 4 for double column plots. The columns of values are tabulated vertically at the left of the printer plot display. Subroutine ZOT 1 utilizes the same profile values and creates only line plots on either microfilm or paper depending on the plot option control of Table 1. The IBM version of the program does not have subroutine ZOT 1.

```
      SUBROUTINE SPLOT 3 ( X, NEND, WIDTH )                                   18NO9
C * * * * THE LATEST REVISION DATE FOR THIS ROUTINE IS  -  -  12 DEC 69 REVISED
      DIMENSION X(NEND), SPACE(15), SYMB(4)                                   10DC9
      COMMON /PLT / I1, I2, J1, J2                                            01DE1IBM
C     COMMON /PLOT/ I1, I2, J1, J2                                            25NO9CDC
      DATA SPACE / 15*4H    /, SYMB / 4H*   ,4H *  ,4H  * ,4H   * /           10DC9
C******** THIS ROUTINE PRINTS AND PLOTS THE NEND VALUES OF X                  10DC9
C         BEGINNING WITH THE INITIAL VALUE TRANSFERRED,                       10DC9
C         THE PAPER SHOULD BE POSITIONED PROPERLY AND ALL                     10DC9
C         HEADINGS PRINTED BEFORE CALLING. F. L. E.                           10DC9
C  ****   INPUT -   X, THE FUNCTION TO BE PLOTTED                             10DC9
C                NEND, THE NUMBER OF X TO BE PLOTTED                          14DC9
C               WIDTH, WIDTH OF PLOT% LESS THAN 61 L                          14DC9
C  ****   OUTPUT-   NO ACTUAL VALUES ARE RETURNED TO THE                      10DC9
C                   CALLING ROUTINE - THE VALUES ARE PRINTED                  10DC9
C                   AND PLOTTED VERTICALLY.                                   10DC9
   10 FORMAT ( 5X, I2, 1X, I3, 1X, 1PE10.3,16A4 )                             25NO9IBM
C  10 FORMAT % 5X, I2, 1X, I3, 1X,   E10.3,16A4 L                             25NO9CDC
   15 FORMAT ( 5X,I2,1X,I3,1X, 1PE10.3 )                                      25NO9IBM
C  15 FORMAT % 5X,I2,1X,I3,1X,   E10.3 L                                      25NO9CDC
         IF ( NEND .LE. 0 )              GO TO 990                            11NO9
         IF ( WIDTH.GT.60. .OR. WIDTH.LT.1. )    WIDTH = 60.                  15DC9
             ISKP = WIDTH / 8 + 1                                             14DC9
             SYMD = SYMB(1)                                                   10DC9
             OMEGA = X(1)                                                     10DC9
             THETA = X(1)                                                     10DC9
         IF ( NEND .EQ. 1 )              GO TO 120                            25NO9
         DO 50  I = 2, NEND                                                   10DC9
         IF ( OMEGA.LT.X(I) )            OMEGA = X(I)                         10DC9
         IF ( THETA.GT.X(I) )            THETA = X(I)                         10DC9
   50    CONTINUE                                                             10DC9
         IF ( OMEGA.EQ.THETA )           GO TO  60                            25NO9
             SIGMA = ( WIDTH - 1. ) / ( OMEGA - THETA )                       10DC9
   60    CONTINUE                                                             25NO9
         IF ( (J2-J1-I2+I1) .GT. 0 )   GO TO 150                              12DE9
             I = 0                                                            25NO9
         DO 110 JS = J1, J2                                                   25NO9
             JSTA = JS - 2                                                    25NO9
         DO 100 IS = I1, I2                                                   25NO9
             ISTA = IS - 2                                                    25NO9
             I = I + 1                                                        25NO9
         IF ( OMEGA.EQ.THETA )           GO TO  80                            10DC9
             BETA = SIGMA * ( X(I) - THETA ) + 1.                             10DC9
             IOTA = BETA                                                      10DC9
         IF ( (BETA - IOTA).GE.0.5 )   IOTA = IOTA + 1                        10DC9
             ISKP = ( IOTA - 1 ) / 4                                          10DC9
             IR   = IOTA - 4 * ISKP                                           10DC9
             ISKP = ISKP + 1                                                  10DC9
             SYMD = SYMB(IR)                                                  10DC9
   80 PRINT 10, ISTA, JSTA, X(I), ( SPACE(L), L=1, ISKP ), SYMD               25NO9
  100    CONTINUE                                                             25NO9
  110    CONTINUE                                                             25NO9
         GO TO 990                                                            25NO9
  150    CONTINUE                                                             12DE9
             I = 0                                                            12DE9
         DO  210  IS = I1, I2                                                 12DE9
             ISTA = IS - 2                                                    12DE9
             DO  200  JS = J1, J2                                             12DE9
             JSTA = JS - 2                                                    12DE9
             I = I + 1                                                        25NO9
         IF ( OMEGA.EQ.THETA )           GO TO  180                           12DE9
             BETA = SIGMA * ( X(I) - THETA ) + 1.                             10DC9
             IOTA = BETA                                                      10DC9
         IF ( (BETA - IOTA).GE.0.5 )   IOTA = IOTA + 1                        10DC9
             ISKP = ( IOTA - 1 ) / 4                                          10DC9
             IR   = IOTA - 4 * ISKP                                           10DC9
             ISKP = ISKP + 1                                                  10DC9
             SYMD = SYMB(IR)                                                  10DC9
  180 PRINT 10, ISTA, JSTA, X(I), ( SPACE(L), L=1, ISKP ), SYMD               12DE9
  200        CONTINUE                                                         12DE9
  210        CONTINUE                                                         12DE9
             GO TO 990                                                        12DE9
  120 PRINT 15, I1, J1, X(1)                                                  25NO9
  990 RETURN                                                                  11NO9
C     END SPLOT 3                                                             18MY0
      END                                                                     10DC9
```

```
      SUBROUTINE SPLOT 4 ( X, Y, NEND )                                       18NO9
C * * * * THE LATEST REVISION DATE FOR THIS ROUTINE IS  -  -  25 NOV 69 REVISED
      DIMENSION X(NEND), Y(NEND), SPACE( 8), SYMB(4)                          15DC9
      COMMON /PLT / I1, I2, J1, J2                                            01DE1IBM
C     COMMON /PLOT/ I1, I2, J1, J2                                            25NO9CDC
      DATA SPACE /  8*4H    /, SYMB / 4H*   ,4H *  ,4H  * ,4H   * /           15DC9
C******** THIS ROUTINE PRINTS AND PLOTS THE NEND VALUES OF X AND Y            10DC9
C         BEGINNING WITH THE INITIAL VALUE TRANSFERRED,                       10DC9
C         THE PAPER SHOULD BE POSITIONED PROPERLY AND ALL                     10DC9
C         HEADINGS PRINTED BEFORE CALLING. F. L. E.                           10DC9
C  ****   INPUT -   X, THE FUNCTION TO BE PLOTTED                             10DC9
C                   Y, THE FUNCTION TO BE PLOTTED                             14DC9
C                NEND, THE NUMBER OF X  OR Y TO BE PLOTTED                    14DC9
C  ****   OUTPUT-   NO ACTUAL VALUES ARE RETURNED TO THE                      10DC9
```

```
C                   CALLING ROUTINE - THE VALUES ARE PRINTED                  100C9
C                   AND PLOTTED VERTICALLY.                                   100C9
   10 FORMAT ( 5X, I2,1X,I3,1X, 1PE10.3,7A4, 1PE10.3,7A4 )                    25ND918M
C  10 FORMAT % 5X, I2,1X,I3,1X,   E10.3,7A4,   E10.3,7A4 L                    25ND9CDC
   15 FORMAT ( 5X,I2,1X,I3,1X, 1PE10.3,28X, 1PE10.3 )                         25ND918M
C  15 FORMAT % 5X,I2,1X,I3,1X,   E10.3,28X,   E10.3 L                         25ND9CDC
         IF ( NEND .LE. 0 )              GO TO 990                            11NO9
              IW = 20                                                         18N39
              IX = IW / 4                                                     140C9
              ISKP  = IW / 8                                                  140C9
              ISKP2 = ISKP                                                    11NO9
              ISKPT = 4                                                       14NO9
              ISKPF = ISKP                                                    140C9
              SYMD  = SYMB(1)                                                 140C9
              SYMD2 = SYMB(1)                                                 140C9
              WIDTH = IW                                                      140C9
              OMEGA = X(1)                                                    100C9
              THETA = X(1)                                                    100C9
              O2 = Y(1)                                                       140C9
              T2 = Y(1)                                                       140C9
         IF ( NEND .EQ. 1 )              GO TO 120                            25NO9
         DO 50  I = 2, NEND                                                   100C9
         IF ( OMEGA.LT.X(I) )            OMEGA = X(I)                         100C9
         IF ( THETA.GT.X(I) )            THETA = X(I)                         100C9
         IF (    O2.LT.Y(I) )            O2 = Y(I)                            140C9
         IF (    T2.GT.Y(I) )            T2 = Y(I)                            140C9
   50    CONTINUE                                                             100C9
         IF ( OMEGA.EQ.THETA )           GO TO  60                            25NO9
              SIGMA = ( WIDTH - 1. ) / ( OMEGA - THETA )                      100C9
   60    IF ( O2.EQ.T2 )                 GO TO  70                            140C9
              S2 = ( WIDTH - 1 ) / ( O2 - T2 )                                140C9
   70    CONTINUE                                                             25NO9
              I = 0                                                           25NO9
         DO 110 JS = J1, J2                                                   25NO9
              JSTA = JS - 2                                                   25NO9
         DO 100 IS = I1, I2                                                   25NO9
              ISTA = IS - 2                                                   25NO9
              I = I + 1                                                       25NO9
         IF ( OMEGA.EQ.THETA )           GO TO  80                            100C9
              BETA = SIGMA * ( X(I) - THETA ) + 1.                            100C9
              IOTA = BETA                                                     100C9
         IF ( (BETA - IOTA).GE.0.5 )   IOTA = IOTA + 1                        100C9
              ISKP = ( IOTA - 1 ) / 4                                         100C9
              IR   = IOTA - 4 * ISKP                                          100C9
              ISKP = ISKP + 1                                                 100C9
              SYMD = SYMB(IR)                                                 100C9
   80    IF (    O2.EQ.T2    )           GO TO  90                            140C9
              BETA = S2 * ( Y(I) - T2 ) + 1.                                  150C9
              IOTA = BETA                                                     140C9
         IF ( (BETA - IOTA).GE.0.5 )   IOTA = IOTA + 1                        140C9
              ISKP2 = ( IOTA - 1 ) / 4                                        140C9
              IR    = IOTA - 4 * ISKP2                                        140C9
              SYMD2 = SYMB(IR)                                                140C9
              ISKP2 = ISKP2 + 1                                               140C9
              ISKPT = IX - ISKP + 1                                           150C9
   90 PRINT 10, ISTA, JSTA, X(I), (SPACE(K),K=1,ISKP), SYMD                   25NO9
    1           , (SPACE(L),L=1,ISKPT), Y(I), (SPACE(K),K=1,ISKP2), SYMD2     25NO9
  100    CONTINUE                                                             25NO9
  110    CONTINUE                                                             25N39
         GO TO 990                                                            25NO9
  120 PRINT 15, I1, J1, X(1), Y(1)                                            25NO9
  990 RETURN                                                                  11NO9
C     END SPLOT 4                                                             18MYO
      END                                                                     100C9
```

```
      SUBROUTINE ZOT 1  ( XF, YF, NP, ID )                                    11JEO
C * * * * THE LATEST REVISION DATA FOR THIS ROUTINE IS  -  -  13 DEC 71 REVISED
      COMMON / ZCT /  LOP, MC, IROLL, MOP                                     11JEO
      DIMENSION  XF(1), YF(1), ID(1)                                          09JEO
      DATA INC, IQ, X, Y, XL, YL, XO, YO / 1,12,0.,0.,9.,4.,0.,90. /          09JEO
      DATA IT1, IT2 / -1, 0 /                                                 09JEO
C           XF     -  ARRAY CONTAINING THE X - COORDINATES                    09JEO
C           YF     -  ARRAY CONTAINING THE Y - COORDINATES                    09JEO
C           NP     -  NUMBER OF POINTS TO BE PLOTTED                          09JEO
C           LOP    -  LINE OR POINT PLOT OPTION                               09JEO
C                     #  0 , LINE PLOT                                        09JEO
C                     # -J , POINT PLOT AT EVERY J-TH POINT                   D9JEO
C                     # &J , LINE PLOT WITH A POINT PLOT AT EVERY J-TH PT.    09JEO
C           ID     -  VARIABLE OR ARRAY CONTAINING TITLE OF PLOT              09JEO
C           MC     -  NUMBER OF CHARACTERS IN TITLE % 0 IF NO TITLE L         09JEO
C           IROLL -   OPTION TO MOVE TO A NEW FRAME - AFTER THIS PLOT         09JEO
C                     # 0 , SAME FRAME                                        09JEO
C                     GREATER THAN 0 , NEW FRAME                              09JEO
C                     LESS THAN 0 , TERMINATE                                 09JEO
C           MOP       MICROFILM OR PAPER PLOT OPTION                          09JEO
C                     # 1 , PAPER PLOTS                                       09JEO
C                     @ 1 , MICROFILM                                         09JEO
C  ****     FRANK L ENDRES  -  PAX 1892    ****                               09JEO
                 NC = MC                                                      09JEO
                 IT1 = IT1 + 1                                                09JEO
            IF ( IT1.NE.0 )      GO TO  20                                    09JEO
C           IF % MOP.EQ.1 L      CALL BGNPLT                                  09JEOCDC
C           IF % MOP.NE.1 L      CALL BGNPLT % 6LFILMPL L                     09JEOCDC
   20       IF (  NC.NE.0 )      GO TO  50                                    09JEO
                 NC = 10                                                      09JEO
C                ID # 10H                                                     09JEOCDC
   50       CONTINUE                                                          09JEO
                 XF(NP+1) = XF1                                               13DE1
                 XF(NP+2) = XF2                                               13DE1
                 YF(NP+1) = YF1                                               13DE1
                 YF(NP+2) = YF2                                               13DE1
C - - - - POSITION ORIGIN                                                     09JEO
C           IF % IT2.EQ.0L       CALL PLT %1.0, 1.0, -3L                      09JEOCDC
            IF ( IT2.EQ.1 )      GO TO  100                                   09JEO
C - - - - SCALE X - AXIS                                                      09JEO
      CALL SCALE  ( XF, XL, NP, INC )                                         09JEO
                 XF1 = XF(NP+1)                                               13DE1
                 XF2 = XF(NP+2)                                               13DE1
C - - - - SCALE Y - AXIS                                                      D9JEO
      CALL SCALE  ( YF, YL, NP, INC )                                         09JEO
                 YF1 = YF(NP+1)                                               13DE1
                 YF2 = YF(NP+2)                                               13DE1
C - - - - SET UP X-AXIS                                                       09JEO
                 YM = - YF(NP+1) / YF(NP+2)                                   09JEO
      CALL AXIS   ( X, YM,1H , -1, XL, XO, XF(NP+1), XF(NP+2) )               09JEO
C - - - - SET UP Y-AXIS                                                       09JEO
                 XM = - XF(NP+1) / XF(NP+2)                                   09JEO
      CALL AXIS   ( XM, Y, 1H , 1, YL, YO, YF(NP+1), YF(NP+2) )               09JEO
C - - - - PRINT TITLE                                                         09JEO
      CALL SYMBOL ( 1., -.75,.14, ID, XO, NC )                                29NO1
C - - - - PLOT THE FUNCTION                                                   09JEO
  100 CALL LINE   ( XF, YF, NP, 1, LOP, IQ )                                  09JEO
            IF ( IT2.EQ.1 )      IT2 = 0                                      09JEO
C           IF % IROLL.GT.0 L    CALL PLT % .0, .0, 999 L                     09JEOCDC
C           IF % IROLL.LT.0 L    CALL ENDPLT                                  09JEOCDC
            IF ( IROLL.EQ.0 )    IT2 = 1                                      09JEO
            IF ( IROLL.LT.0 )    IT1 = -1                                     11JEO
      RETURN                                                                  09JEO
      END                                                                     09JEO
```

3-D PLOT SUBROUTINES

These nine routines are used to generate the paper plot display of the exaggerated deflected shape of the slab or grid. The routines default to values that have been found to give a reasonable optimum implied position of the viewer's eye. The plot may be controlled by input of different options in the second card of Table 1. The IGSW reference grid switch may be input in column 62, the value of the plotted maximum deflection in columns 66-70, the reduction factor of Y lengths relative to X lengths in columns 71-75, and the tangent of the receding plot angle in columns 76-80.

DIM3 - Defines the plot size and orientation and controls subsequent routines. PLT and SYMBOL are CALCOMP routines.

TD2 - Plots the relative deflection value at the appropriate position, connecting all grid points. Hidden points are masked out.

GRD - Controls the reference grid to be plotted as a function of the reference grid switch IGSW:

0 = Tic marks at edge points only (default)
1 = No reference marks
2 = Tic marks at the 4 corner points
3 = Tic marks at all grid points
4 = Dashed lines along edges only
5 = Solid lines along edges only
6 = Dashed grid lines throughout
7 = Solid grid lines throughout
8 = Dashed grid shifted 12 inches to right
9 = Solid grid shifted 12 inches to right

EDGE 1 - Plots tics at edge points

CRNR 1 - Plots tics at the 4 corner points

MESH 1 - Plots tics at all grid points

GRID 1 - Plots dashed or solid lines at edges or throughout

MARK 2 - Plots four arm tic oriented in the proper direction

DASH 2 - Plots a dashed or solid line

```
      SUBROUTINE DIM3 ( Z, L1, L2, MX, MY, HX, HY, VEF , RDF, SLOPE,     09FE1
     1                  IPOP, NPROB, IGSW )                              06MY1
C
C-----BY TEMPORARILY NULLING OUT THIS ROUTINE, THE 3 - D PLOT IS
C     ELIMINATED, AND THESE ROUTINES MAY BE REMOVED FROM THE DECK --
C     DIM3, TD2, GRID1, GRD, CRNR1, EDGE1, MESH1, MARK2, AND DASH2. THUS
C     SAVING 7150 OCTAL STOR AND 4500 OCTAL STOR FOR SYSTEM PLOT CALLS.
C
C
C-----THE CARDS TAGGED WITH PLT ARE FOR CDC SYSTEM PLOT ROUTINES AND ARE
C     IN GENERAL COMPATIBLE WITH ALL SYSTEMS USING STANDARD CALCOMP PLOT
C     ROUTINES. MODIFICATIONS FOR PARTICULAR PLOTTER INSTALLATIONS MAY
C     BE NECESARY HOWEVER.
C
      DOUBLE PRECISION  Z, NPROB                                         10DE1IBM
      DIMENSION  Z(L1,L2) , IZ(2500)                                     26OC0
C         IF % IPOP.LE.0 .A. START.NE.1.314 L     CALL BGNPLT            19OC0CDC
      CALL PLOT    ( 1., 1., -3 )                                        26OC0IBM
C     CALL PLT     % 1., 1., -3 L                                        26OC0CDC
      CALL SYMBOL (-.1, 3., .14, NPROB, 90., 5 )                         12MY1
              IF ( ABS ( VEF ) .LT. 0.01 ) VEF   = 1.0                   12FE1
              IF ( ABS (SLOPE) .LT. 0.01 )  SLOPE = 0.9                  11FE1
              IF (        ROF   .LT. 0.01 )  RDF   = 0.75                11FE1
              H = 6.5                                                    02NO0
              IOFY = H * 100 / MY                                        29OC0
              IOFX = IOFY / SLOPE                                        29OC0
              XX = IOFX * IOFX + IOFY * IOFY                             30OC0
              IDEL = SQRT( XX ) * HX / ( HY * RDF )                      30OC0
              W = .01 * ( IDEL * MX + IABS(IOFX) * MY )                  10MY1
         IF ( W.LE.10 )      GO TO  5                                    29OC0
              H = 10. * H / W                                            02NO0
              IOFY = H * 100 / MY                                        29OC0
              IOFX = IOFY / SLOPE                                        29OC0
              IDEL = 10. / W * IDEL                                      02NO0
              W = 10.                                                    29OC0
    5         VSF = 0.                                                   29OC0
              MXP2 = MX + 2                                              02NO0
              MYP2 = MY + 2                                              02NO0
         DO  10  I = 2, MXP2                                             11DE0
         DO 10 J = 2, MYP2                                               11DE0
         IF (DABS( Z(I,J)).GT.VSF   )          VSF =DABS( Z(I,J) )       26OC0IBM
C        IF % ABS% Z%I,JLL.GT.VSF  L           VSF # ABS% Z%I,JL L       26OC0CDC
   10    CONTINUE                                                        26OC0
              VSF = VSF / VEF                                            09FE1
              IN = MY + 1                                                26OC0
              KN = MX + 1                                                26OC0
              JS = 1                                                     26OC0
              PLTMIN = -.5                                               26OC0
              INIT = 1                                                   26OC0
         DO 100  I = 1, IN                                               26OC0
  100 CALL TD2   ( Z(2,I+1), IN, KN, IDEL, VSF, IOFX,IOFY,IZ,PLTMIN,JS,  26OC0
     1             INIT )                                                26OC0
              XS = 0                                                     11DE0
         IF ( IOFX.LT.0 )           XS = .01 * IABS (IOFX) * MY          11FE1
      CALL GRD  ( IGSW, MX, MY, IOFX, IOFY, IDEL, XS, YS )               06MY1
      CALL PLOT  (15., 0., -3  )                                         30OC0IBM
C     CALL PLT   %15., 0., -3  L                                         30OC0CDC
              START = 1.314                                              26OC0
      RETURN                                                             26OC0
      END                                                                26OC0
```

```
      SUBROUTINE TD2  ( Z, IN, KN, IDEL, VSF, IOFX, IOFY, IZ, PLTMIN,JS,26OC0
     1                 INIT )                                           11FE1
      DOUBLE PRECISION   Z                                              24MY1IBM
      DIMENSION Z(1), IZ(1)                                             26OC0
      DIMENSION XM(50), YM(50)                                          26OC0
           DO 5 I = 1, KN                                               26OC0
    5            XM(I) = PLTMIN - 1.                                    26OC0
                 IEND = KN * IDEL + IABS(IOFX) * (IN - 1) + 1           26OC0
                 INIT = 0                                               26OC0
                 NU = 0                                                 26OC0
                 L = 0                                                  26OC0
                 MP = 1                                                 26OC0
                 KL = KN - 1                                            26OC0
                 POV = PLTMIN / VSF                                     26OC0
                 IPOV = POV * 100.                                      26OC0
                 DEL = IDEL / 100.                                      26OC0
                 OFY = IOFY / 100.                                      26OC0
                 OFX = IOFX / 100.                                      26OC0
           IF ( IOFX .GE. 0 )        GO TO  8                           26OC0
                 L = -IOFX * (IN - 1) + 1                               26OC0
    8      CONTINUE                                                     26OC0
           DO 9  I = 1, IEND                                            26OC0
                 IZ(I) = IPOV                                           26OC0
    9      CONTINUE                                                     26OC0
                 IPEN = 3                                               26OC0
   10      CONTINUE                                                     26OC0
                 II = NU - 1                                            12FE1
                 NU = NU + 1                                            26OC0
                 II = NU - 1                                            26OC0
                 DELY = OFY * II                                        26OC0
                 DELX = OFX * II                                        26OC0
           DO 31  I = 1, KL                                             26OC0
           IF ( MP ) 15, 15, 16                                         26OC0
   15            K = KN + 1 - I                                         26OC0
           GO TO  17                                                    26OC0
   16            K = I                                                  26OC0
   17      CONTINUE                                                     26OC0
C-----COMPUTE INTERPOLATION CONSTANT                                    11FE1
                 KMP = K + MP                                           26OC0
                 SI = ( Z(KMP) - Z(K) ) / VSF                           26OC0
                 SI = .01 * SI / DEL                                    26OC0
                 AT = Z(K) / VSF + DELY - SI                            26OC0
C-----INTERPOLATE POINTS                                                11FE1
           DO 30  LL = 1, IDEL                                          26OC0
                 L = LL + MP                                            26OC0
                 AT = AT + SI                                           26OC0
C-----COMPUTE NEW ATTITUDE                                              12FE1
                 IR = AT * 100. *IZ(L)                                  12FE1
           IF ( IR .LE. 0 )          GO TO  19                          12FE1
                 IZ(L) = SIGN (( ABS (AT * 100.) + .5) , AT )           26OC0
   19      CONTINUE                                                     26OC0
C-----GENERATE ABSCISSA                                                 11FE1
                 C = L * .01                                            26OC0
           GO TO ( 24, 25 ) , JS                                        26OC0
   24            X = C                                                  26OC0
                 Y = IZ(L) * .01                                        26OC0
           GO TO  26                                                    26OC0
   25            Y = -C                                                 26OC0
                 X = IZ(L) * .01                                        26OC0
   26 CALL PLOT( X, Y, IPEN )                                           26OC0IBM
C  26 CALL PLT % X, Y, IPEN L                                           26OC0CDC
           IF ( I.EQ.1 .OR.LL.EQ.1 )      GO TO 265                     26OC0
                 XM(K) = X                                              26OC0
                 YM(K) = Y                                              26OC0
  265            IPEN = 2                                               26OC0
   30      CONTINUE                                                     03MA1
```

```
      IF ( XM(KMP).GE.PLTMIN )      CALL PLOT (XM(KMP),YM(KMP),2)      12FE1IBM
C     IF % XM%KMPL.GE.PLTMIN L      CALL PLT  %XM%KMPL,YM%KMPL,2L      12FE1CDC
      CALL PLOT ( X, Y, 3 )                                              26DC0IBM
C     CALL PLT  % X, Y, 3 L                                              26DC0CDC
            XM(KMP) = X                                                  12FE1
            YM(KMP) = Y                                                  12FE1
   31     CONTINUE                                                       26DC0
            L = L + IOFX + MP                                            26DC0
            MP = - MP                                                    26DC0
      RETURN                                                             26DC0
      END                                                                26DC0
```

```
      SUBROUTINE EDGE1  ( MX, MY, IOFX, IOFY, IDEL, XO, YO )             15AP1
      COMMON /PGRID/ SLOPE, SIZE                                         27AP1
            X = XO                                                       15AP1
            Y = YO                                                       15AP1
            SIZE = 2.                                                    15AP1
            OFX = IOFX * .01                                             15AP1
            OFY = IOFY * .01                                             15AP1
            DEL = IDEL * .01                                             15AP1
      CALL MARK2 ( X, Y )                                                27AP1
         DO 100  I = 1, MX                                               15AP1
            X = X + DEL                                                  15AP1
      CALL MARK2 ( X, Y )                                                27AP1
  100    CONTINUE                                                        15AP1
            DO 200  I = 1, MY                                            15AP1
            X = X + OFX                                                  15AP1
            Y = Y + OFY                                                  15AP1
      CALL MARK2 ( X, Y )                                                27AP1
  200    CONTINUE                                                        15AP1
         DO 300  I = 1, MX                                               15AP1
            X = X - DEL                                                  15AP1
      CALL MARK2 ( X, Y )                                                27AP1
  300    CONTINUE                                                        15AP1
         DO 400  I = 1, MY                                               15AP1
            X = X - OFX                                                  15AP1
            Y = Y - OFY                                                  15AP1
      CALL MARK2 ( X, Y )                                                27AP1
  400    CONTINUE                                                        15AP1
      RETURN                                                             15AP1
      END                                                                15AP1
```

```
      SUBROUTINE GRD  ( IGSW, MX, MY, IOFX, IOFY, IDEL, XS, YS )         27AP1
      COMMON /PGRID/ SLOPE, SIZE                                         27AP1
            SLOPE = IOFY / IOFX + .01                                    27AP1
            IGSW = IGSW + 1                                              15AP1
      IF ( IGSW ) 999, 999, 105                                          27AP1
  105 GO TO (110, 300, 130, 140, 150, 160, 170, 180, 190, 200),IGSW      23NO1
  110 CALL EDGE1 (MX, MY, IOFX, IOFY, IDEL, XS, 0.0)                     10MY1
      GO TO 300                                                          15AP1
  130 CALL CRNR1  (MX, MY, IOFX, IOFY, IDEL, XS, 0.0)                    10MY1
      GO TO 300                                                          15AP1
  140 CALL MESH1  (MX, MY, IOFX, IOFY, IDEL, XS, 0.0)                    10MY1
      GO TO 300                                                          15AP1
  150       LOP = 0                                                      15AP1
  155 CALL GRID1 (1, 1, IOFX*MY, ICFY*MY, IDEL*MX, XS, 0.0,LOP )         16AP1
      GO TO 300                                                          15AP1
  160       LOP = 1                                                      15AP1
      GO TO 155                                                          15AP1
  170       LOP = 0                                                      15AP1
  175 CALL GRID1 (MY, MX, IOFX, IOFY, IDEL, XS, 0.0,LOP )                10MY1
      GO TO 300                                                          15AP1
  180       LOP = 1                                                      15AP1
      GO TO 175                                                          15AP1
  190       XS = XS + IABS(IOFX) * MX * .01 + 2.                         15AP1
      GO TO 170                                                          15AP1
  200       XS = XS + IABS(IOFX) * MX * .01 + 2.                         15AP1
      GO TO 180                                                          15AP1
  300 CONTINUE                                                           15AP1
  999 CONTINUE                                                           15AP1
      RETURN                                                             26AP1
      END                                                                26AP1
```

```
      SUBROUTINE  CRNR1  ( MX, MY, ICFX, IOFY, IDEL, XO, YO )            15AP1
      COMMON /PGRID/ SLOPE, SIZE                                         27AP1
            X = XO                                                       15AP1
            Y = YO                                                       15AP1
            SIZE = 4.                                                    15AP1
      CALL MARK2 ( X, Y )                                                27AP1
            X = X + MX *  IDEL * .01                                     15AP1
      CALL MARK2 ( X, Y )                                                27AP1
            Y = Y + MY * IOFY * .01                                      15AP1
            X = X + MY *IOFX * .01                                       15AP1
      CALL MARK2 ( X, Y )                                                27AP1
            X = X - MX * IDEL * .01                                      15AP1
      CALL MARK2 ( X, Y )                                                27AP1
      RETURN                                                             15AP1
      END                                                                15AP1
```

```
      SUBROUTINE MESH1  ( MX, MY, IOFX, IOFY, IDEL, XO, YO )          15AP1
      COMMON /PGRID/ SLOPE, SIZE                                      27AP1
              SIZE = 1.                                               15AP1
              OFX = IOFX * .01                                        15AP1
              OFY = IOFY * .01                                        15AP1
              DEL = IDEL * .01                                        15AP1
              MYP1 = MY + 1                                           15AP1
              MXP1 = MX + 1                                           15AP1
              XX = XO                                                 15AP1
              MP = 1                                                  15AP1
              Y = YO - OFY                                            15AP1
          DO 2000  J = 1, MYP1                                        15AP1
              X = XX - DEL * MP                                       15AP1
              Y = Y + OFY                                             15AP1
          DO 1000  I = 1, MXP1                                        15AP1
              X = X + DEL * MP                                        15AP1
      CALL MARK2 ( X, Y )                                             27AP1
 1000     CONTINUE                                                    15AP1
              XX = XX + OFX + DEL * MX * MP                           15AP1
 2000         MP = -MP                                                15AP1
      RETURN                                                          15AP1
      END                                                             15AP1

      SUBROUTINE GRID1 ( IN, KN, IOFX, IOFY, IDEL, XO, YO, LOP )      290C0
              XX = IDEL * KN * .01                                    290C0
              IN1 = IN + 1                                            290C0
              MP = -1                                                 290C0
          DO 100  I = 1, IN1                                          290C0
              IM1 = I - 1                                             290C0
              X1 = XO + IOFX * IM1 * .01                              290C0
              Y1 = YO + IOFY * IM1 * .01                              290C0
              X2 = X1 + XX                                            290C0
          IF ( MP ) 80, 100, 90                                       290C0
   80 CALL DASH2 ( X1,Y1 , X2,Y1, LOP )                               290C0
          GO TO  99                                                   290C0
   90 CALL DASH2 ( X2,Y1 , X1,Y1, LOP )                               290C0
   99         MP = -MP                                                290C0
  100     CONTINUE                                                    290C0
              XX  = IN * IOFX * .01                                   290C0
              KN1 = KN + 1                                            290C0
              Y1 = YO                                                 290C0
              Y2 = YO + IN * IOFY * .01                               290C0
          DO 200  I = 1, KN1                                          290C0
              IM1 = I - 1                                             290C0
              X1 = XO + IDEL * IM1 * .01                              290C0
              X2 = X1 + XX                                            290C0
          IF ( MP ) 180, 200, 190                                     290C0
  180 CALL DASH2 ( X1,Y1 , X2,Y2, LOP )                               290C0
          GO TO  199                                                  290C0
  190 CALL DASH2 ( X2,Y2 , X1,Y1, LOP )                               290C0
  199         MP = -MP                                                290C0
  200     CONTINUE                                                    290C0
      RETURN                                                          290C0
      END                                                             290C0

      SUBROUTINE MARK2 ( X, Y )                                       16AP1
      COMMON /PGRID/ SLOPE, SIZE                                      16AP1
              STEP = .02                                              27AP1
              ARM =  STEP * SIZE                                      27AP1
              X1 = ARM * SQRT ( 1. / ( 1. + SLOPE*SLOPE ) )           28AP1
              Y1 = X1 * SLOPE                                         27AP1
         IF ( X1.EQ.O. )            Y1 = ARM                          28AP1
              XX = X - ARM                                            27AP1
      CALL PLOT  ( XX, Y, 3 )                                         16AP1IBM
C     CALL PLT   $ XX, Y, 3 L                                         16AP1CDC
              XX = X + ARM                                            27AP1
      CALL PLOT  ( XX, Y, 2 )                                         16AP1IBM
C     CALL PLT   $ XX, Y, 2 L                                         16AP1CDC
              XX = X + X1                                             27AP1
              YY = Y + Y1                                             27AP1
      CALL PLOT  ( XX, YY, 3 )                                        16AP1IBM
C     CALL PLT   $ XX, YY, 3 L                                        16AP1CDC
              XX = X - X1                                             27AP1
              YY = Y - Y1                                             27AP1
      CALL PLOT  ( XX, YY, 2 )                                        16AP1IBM
C     CALL PLT   $ XX, YY, 2 L                                        16AP1CDC
      RETURN                                                          16AP1
      END                                                             16AP1

      SUBROUTINE DASH2  ( X1,Y1 , X2,Y2, LOP )                        290C0
C - - -  LOP # 1 FOR SOLID,  OTHERWISE DASHED                         290C0
      CALL PLOT  ( X1, Y1, 3 )                                        01DE1IBM
C     CALL PLT   $ X1, Y1, 3 L                                        01DE1CDC
              IP = 2                                                  290C0
              IS = -1                                                 290C0
         IF ( LOP - 1 ) 6, 5, 6                                       290C0
    5         IS = 0                                                  290C0
    6         DIS  = SQRT ( (X2-X1)**2 + (Y2-Y1)**2 )                 290C0
              INC  = DIS * 5.                                         290C0
              XD   = (X2-X1) / (INC+1) * .5                           290C0
              YO   = (Y2-Y1) / (INC+1) * .5                           290C0
              X    = X1                                               290C0
              Y    = Y1                                               290C0
         IF ( INC ) 101, 101, 10                                      290C0
   10         INC2 = INC * 2                                          290C0
              XT = X1 + XD / 4.                                       290C0
              YT = Y1 + XO / 4.                                       290C0
      CALL PLOT  ( XT, YT, 2 )                                        290C0IBM
C     CALL PLT   $ XT, YT, 2 L                                        290C0CDC
         DO 100  I = 1, INC2                                          290C0
              X = X + XD                                              290C0
              Y = Y + YD                                              290C0
              IS = - IS                                               290C0
              IP = IP + IS                                            290C0
      CALL PLOT  ( X, Y, IP )                                         290C0IBM
C     CALL PLT   $ X, Y, IP L                                         290C0CDC
  100    CONTINUE                                                     290C0
  101         X = X + (X2-X) / 2                                      290C0
              Y = Y + (Y2-Y) / 2                                      290C0
              IS = -IS                                                290C0
              IP = IP + IS                                            290C0
      CALL PLOT  ( X, Y, IP )                                         290C0IBM
C     CALL PLT   $ X, Y, IP L                                         290C0CDC
      CALL PLOT  ( X2, Y2, 2 )                                        290C0IBM
C     CALL PLT   $ X2, Y2, 2 L                                        290C0CDC
  999 RETURN                                                          290C0
      END                                                             290C0
```

SUBROUTINE FRIP8

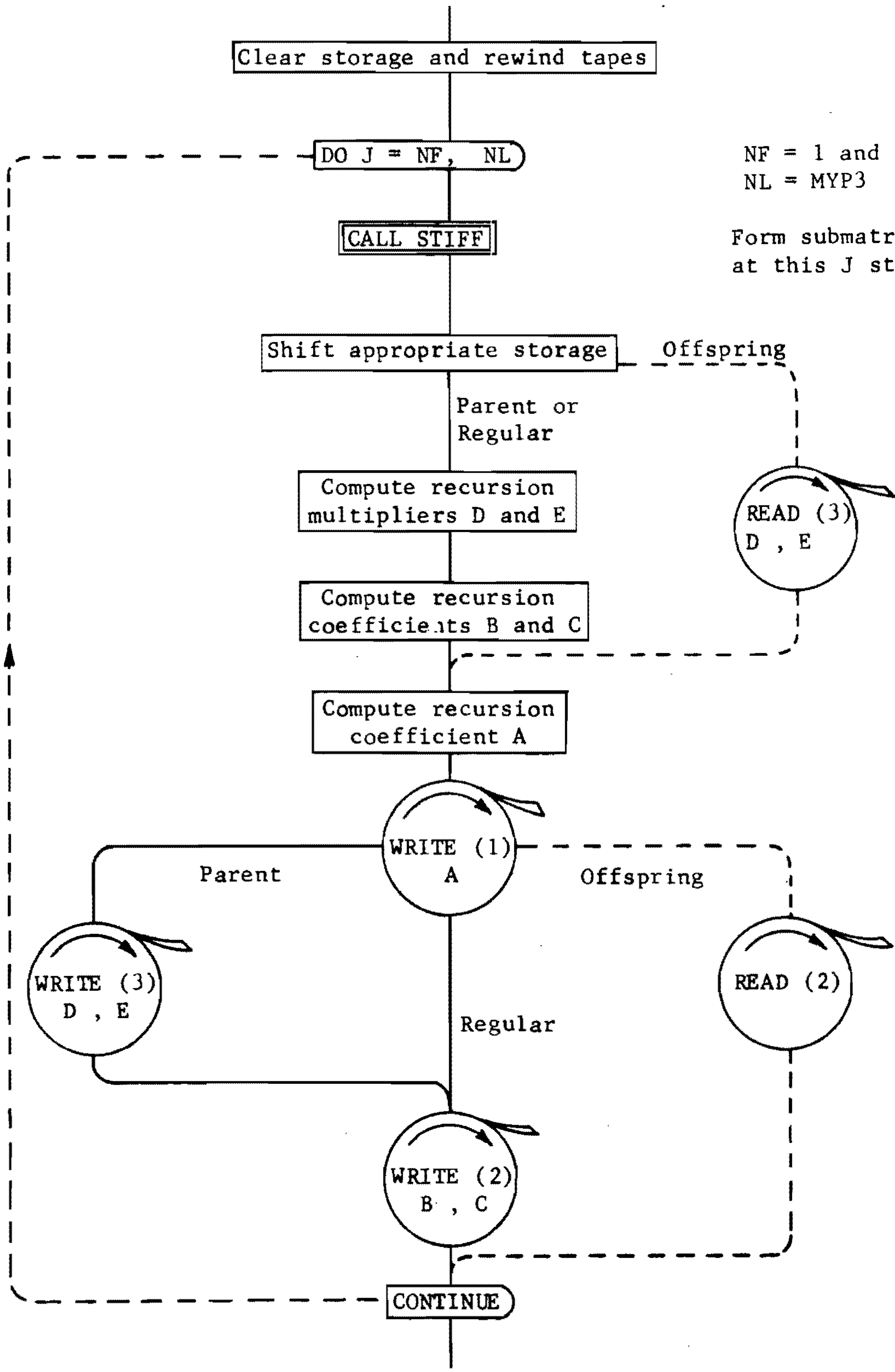
This subroutine is a general version of the FRIP4 Five-Wide Recursion-Inversion Solution Process which is documented in Research Report 56-19, "An Alegebraic Equation Solution Process Formulated in Anticipation of Banded Linear Equations," by Frank L. Endres and Hudson Matlock (Ref 3).

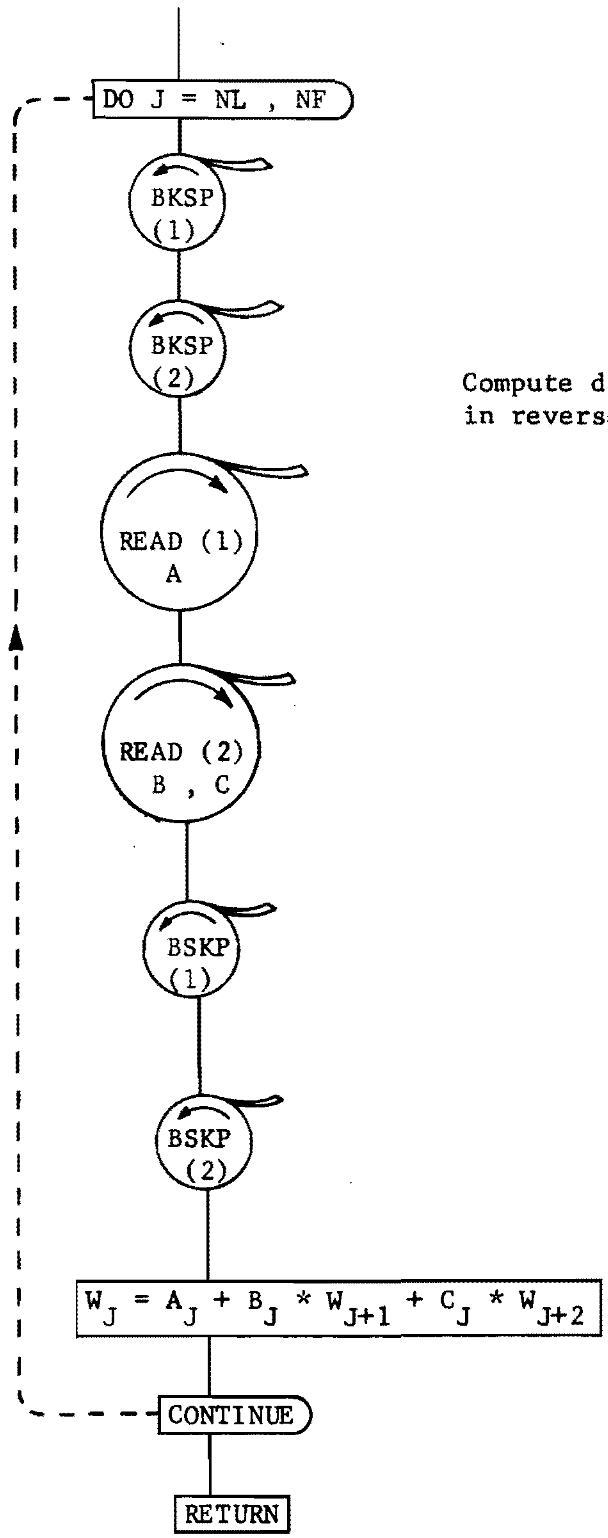
This group of 14 subroutines provides an efficient solver for a sparsely banded matrix of equations. It can handle up to 5 groups of bands, each of arbitrary width. As used here, the matrix is assumed to be symmetric and positive definite.

These routines are called by FRIP8 and are completely documented in Ref 3.

RFV		-	Replaces a full matrix or a vector by another
MBFV		-	Multiplies a banded (packed) matrix times a full matrix or a vector
ABF		-	Adds a banded matrix to a full matrix
MFFV		-	Multiplies full (square) matrix times a full (square) matrix or a vector
ASFV		-	Adds or subtracts two full matrices or two vectors
SMFF		-	Symmetric multiplication of a full times a full matrix
INVR6	DCOM1 INVLT1 MLTXL FIX1	-	Takes inverse of symmetric positive definite matrix
CFV		-	Multiples of full matrix or a vector by a constant
MFB		-	Multiplies a full matrix times a banded (packed) matrix
MFFT		-	Multiplies a full times the transpose of a full matrix

SUBROUTINE FRIP8





Compute deflections in reverse J-direction

```
      SUBROUTINE FRIPB  ( L1   , L2   , ML   , N1   , N2   , N3   ,         08FE1
     1                    A    , AM1  , AM2  , ATM  , R1   , R2   , R3   ,   08FE1
     2                    ET2  , ET1  , CC   , DT   , EE   , FF   , WW   ,   08FE1
     3                    DX   , DXM1, DXP1, DY   , DYM1, DYP1,              08FE1
     4                    FX   , FY   , FYM1, FYP1, Q    , S    ,            08FE1
     5                    RX   , RYM1, RYP1, TX   , TYM1, TYP1, CH   ,       08FE1
     6                    CHP1, PX   , PY   , PYP1, PBX  , PBY  , PBYP1,     08FE1
     7                    QM                                           )   08FE1
C
C*IBM*IBM*THIS VERSION OF FRIPB HAS BEEN MODIFIED TO USE THE DIRECT          IBM
C         ACCESS CAPABILITIES OF THE 360 SERIES — CARDS TAGGED WITH          IBM
C         IBM ARE NOT ASA FORTRAN AND THEREFORE NOT GENERALLY COMPATIBLE     IBM
C         THE CARDS TAGGED CDC ARE ASA FORTRAN AND MAY BE USED, BUT          IBM
C         DIRECT REFRENCING OF FILES IS MORE DESIREABLE.                     IBM
C
C*******             THE LATEST REVISION DATE FOR THIS PROGRAM IS - - - - 22OC9
C******* THIS GROUP OF 15 SUBROUTINES PROVIDES THE USER WITH AN            20MY8
C        EFFICIENT GENERAL SPARSELY BANDED EQUATION SOLVER                 04JA8
C        %THE MATRIX IS ASSUMED TO BE SYMMETRIC AND POSITIVE DEFINITEL     12MR8
C        WHICH CAN HANDLE UP TO 5 GROUPS OF BANDS , EACH                   04JA8
C        OF ARBITRARY WIDTH                                                04JA8
C******* THIS ROUTINE SUPERVISES 14 SUBROUTINES , 13 OF WHICH              20MY8
C        ARE SELF-SUFFICIENT AND COME AS A PACKAGE , THE                   04JA8
C        REMAINING ONE GENERATES AND PACKS THE STIFFNESS                   04JA8
C******* MATRIX IS OUTLINED IN THE APPENDIX OF REPORT 56-19.               28MY1
C******* FILE 4 IS WHERE THE SOLUTION VECTOR IS STORED                     22AG9
C        E%J&1L # AA%J&1L*B%J-1L & BB%J&1L                                 22AG9
C        D%JL   # -1 / % AA%JL*C%J-2L & E%JL*B%J-1L & CC%JL L              22AG9
C        C%JL   # D%JL*EE%JL                                               22AG9
C        B%JL   # D%JL*E%J&1LT                                             22AG9
C        A%JL   # D%JL*% E%JL*A%J-1L & AA%JL*A%J-2L - FF%JL L              22AG9
      DOUBLE PRECISION  ET2, ET1, CC, DT,  EE,  FF,  A, AM1, AM2, ATM,     25MY0IBM
     1                  R1,  R2, R3, WW                                    07JU1IBM
      DIMENSION  A(L1,N1)  , AM1(L1,N1)  , AM2(L1,N1)  , ATM(L1,N1)  ,     05MR1
     2           R1(L1,L1)  ,  R2(L1,L1)  ,  R3(L1,L1)  ,                  22AG9
     4           DT(L1,N2)  ,  CC(L1,N3)  , ET1(L1,N1)  ,  EE(N1,L1)  ,    23MR8
     5                         FF(L1,1),     ET2(L1,N1)                    05MR1
      DIMENSION   DX(L1), DYM1(L1),   DY(L1), DYP1(L1),   FX(L1),          24JE8
     1           FYM1(L1),    FY(L1), FYP1(L1),    Q(L1),     S(L1),       24JE8
     2             RX(L1), RYM1(L1), RYP1(L1),                             24JE8
     3             TX(L1), TYM1(L1), TYP1(L1),                             24JE8
     4             CH(L1), CHP1(L1),               DXP1(L1), DXM1(L1),     05AG9
     5             PX(L1),    PY(L1), PYP1(L1),                            24JE8
     6            PBX(L1),  PBY(L1),PBYP1(L1),                             24JE8
     7             QM(L1)                                                  24JE8
      DIMENSION  WW( L1, L2 )                                              25MY0
      COMMON /RI/ NK , NL , NF ,  ITMPP                                    12FE1
      COMMON /DA/ ID1, ID2, ID3, ID4                                       14JA2IBM
   92 FORMAT (1X,1PE10.3, 1X,1PE10.3, 1X,1P5E10.3, 1X,1P3E10.3, 1X,        29JL9IBM
     1        1PE10.3,  1X,1PE10.3 )                                       29JL9IBM
C  92 FORMAT %1X,  E10.3, 1X,  E10.3, 1X,  5E10.3, 1X,  3E10.3, 1X,        29JL9CDC
C    1         E10.3,  1X,  E10.3 L                                        29JL9CDC
   93 FORMAT (/( 1X,1P11E10.3 ) )                                          29JL9IBM
C  93 FORMAT %/% 1X,  11E10.3 L L                                          29JL9CDC
C     REWIND 1                                                             04JA8CDC
C     REWIND 2                                                             04JA8CDC
C     REWIND 3                                                             17JA8CDC
C     REWIND 4                                                             22AG9CDC
```

```
          IF( ML ) 140, 100, 100                                           04JA8
C        SET INITIAL CONDITIONS                                            04JA8
  100     DO 135  J = 1 , NK                                               01FE8
          DO 130  I = 1 , NK                                               01FE8
                R1(I,J) = 0.0                                              22AG9
                R3(I,J) = 0.0                                              22AG9
  130     CONTINUE                                                         04JA8
  135     CONTINUE                                                         04JA8
      WRITE ( 3 ' 1 )      ( ( R1(I,K) , I = 1, NK ), K = 1, NK )            IBM
      WRITE ( 4 ' 1 )      ( ( R1(I,K) , I = 1, NK ), K = 1, NK )          01DE1IBM
C     WRITE%3L  %% R1%I,KL , I # 1,NKL , K # 1,NK L                        22AG9CDC
C     WRITE%4L  %% R1%I,KL , I # 1,NKL , K # 1,NK L                        22AG9CDC
C     REWIND 3                                                             22AG9CDC
C     REWIND 4                                                             22AG9CDC
  140     DO 150  I = 1 , NK                                               01FE8
                A(I,1) = 0.0                                               23NO1
                AM1(I,1) = 0.0                                             23NO1
  150     CONTINUE                                                         04JA8
C***************************************************************************
C        BEGIN FORWARD PASS  --  SOLVE FOR RECURSION COEFFICIENTS          04JA8
C***************************************************************************
C                  . . . . . . . . . . . . . . . . . . . . . . . . . . . .
          DO 1000  J = NF , ML                                             01FE8  .
                JJ = J                                                     04JA8  .
                J2 = J + J                                                 25MY0IBM
                J1 = J2 - 1                                                25MY0IBM
C        FORM SUB-MATRICES                                                 04JA8  .
      CALL STIF            ( L1 ,   ET2 ,   ET1 ,   CC  ,   DT  ,          23NO1
     1                        EE,    FF,    ML,    JJ,    N1,   N2,   N3,  26SE8  .
     2                        DX, DYM1,    DY, DYP1,    FX, FYM1,          26SE8  .
     3                        FY, FYP1,    Q,    S,                        26SE8  .
     4                        RX, RYM1, RYP1,   TX, TYM1, TYP1,   CH, CHP1, 26SE8 .
     5                        PX,   PY, PYP1,  PBX,  PBY,      PBYP1,   QM, 29JL9 .
     6                      DXP1, DXM1 )                                   05AG9  .
          IF ( ITMPP.GE.1 )   PRINT 92,  (ET2(I,1),ET1(I,1),               29JL9  .
     1        (CC(I,J),J=1,5 ),(DT(I,J),J=1,3),EE(I,1),FF(I,1),I=1,NK)     23NO1  .
      CALL RFV    ( AM2, AM1, L1 , 1   , NK )                              20MY8  .
      CALL RFV    ( AM1, A  , L1 , 1   , NK )                              20MY8  .
          IF ( ML.GE.0 )      GO TO  220                                   22AG9  .
C        READ E AND D MULTIPLIERS FROM FILE 3                              22AG9  .
      READ  ( 3'J1) (( R1(I,K), I=1,NK), K=1,NK )                          01DE1IBM
      READ  ( 3'J2) (( R2(I,K), I=1,NK), K=1,NK )                          01DE1IBM
C     READ %3L  %% R1%I,KL , I # 1,NKL , K # 1,NK L                        22AG9CDC
C     READ %3L  %% R2%I,KL , I # 1,NKL , K # 1,NK L                        22AG9CDC
          GO TO  230                                                       22AG9  .
C        CALCULATE RECURSION MULTIPLIER EP1                                23MR8  .
CEEEE    EP1 # ET2*BM1 & DT                                                22AG9  .
  220 CALL MBFV  ( ET1,  R3,  R2, L1 , L1 , NK , N1 )                      22AG9  .
      CALL ABF   ( DT ,  R2,  R2, L1 , NK , N2 )                           22AG9  .
          IF ( ITMPP.GE.2 )   PRINT 93, ((R1(I,J),J=1,NK),I=1,NK )         22AG9  .
C        BEGIN CALULATION RECURSION COEFFIECENT A                          22AG9  .
  230 CALL MFFV  ( R1 , AM1, A  , L1 , 1   , NK )                          22AG9  .
          IF ( ITMPP.GE.2 )   PRINT 93,  ( A(I,1),I=1,NK)                  29JL9  .
      CALL MBFV  ( ET2, AM2, ATM, L1 , 1   , NK , N1 )                     20MY8  .
      CALL ASFV  ( A  , ATM, AM2, L1 , 1   , NK , +1 )                     20MY8  .
      CALL ASFV  ( AM2, FF , ATM, L1 , 1   , NK , -1 )                     20MY8  .
          IF ( ML.LT.0 )      GO TO  280                                   22AG9  .
      WRITE ( 1'1 ) (( R2(I,K), I=1,NK), K=1,NK )                          01DE1IBM
```

```
C      WRITE%1L   %% R2%I,KL , I # 1,NKL , K # 1,NK L                    22AG9CDC
C      REWIND 1                                                            25MYOCDC
C         CALCULATE RECURSION MULTIPLIER D                                 04JA8  .
CDDDD     D # -I / % ET2*CM2 & E*BM1 &CC L                                 22AG9  .
       CALL SMFF  ( R1 , R3 , R2 , L1 , NK )                               22AG9  .
           IF ( ITMPP.GE.2 )   PRINT 93, ((R2(I,J),J=I,NK),I=1,NK )         22AG9  .
C      READ %3L   %% R3%I,KL , I # 1,NKL , K # I,NK L                      22DC9CDC
C      BACKSPACE 3                                                         22AG9CDC
       READ  ( 3'J1) (( R3(I,K), I=1,NK), K=I,NK )                             18M
C         SAVE RECURSION MULTIPLIER E ON TAPE 3                            22AG9  .
                ID3 = J1                                                   01DE1I8M
       WRITE ( 3'ID3)(( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
       READ  ( 4'1 ) (( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
       WRITE ( 4'1 ) (( R3(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      WRITE%3L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22DC9CDC
C      READ %4L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22DC9CDC
C      REWIND 4                                                            22AG9CDC
C      WRITE%4L   %% R3%I,KL , I # 1,NKL , K # 1,NK L                      22DC9CDC
C      REWIND 4                                                            22AG9CDC
       CALL MBFV  ( ET2,  R1, R3 , L1 , L1 , NK , N1 )                     22DC9  .
           IF ( ITMPP.GE.2 )   PRINT 93, ((R3(I,J),J=1,NK),I=1,NK )         22DC9  .
       CALL ASFV  ( R2 , R3 , R2 , L1 , L1 , NK , +1 )                     22AG9  .
           IF ( ITMPP.GE.2 )   PRINT 93, ((R2(I,J),J=1,NK),I=1,NK )         22AG9  .
       CALL ABF   ( CC , R2 , R2 , L1 , NK , N3 )                          22AG9  .
           IF ( ITMPP.GE.1 )   PRINT 93, ((R2(I,J),J=1,NK),I=1,NK )         22AG9  .
       CALL INVR6 ( R2 , L1 , NK )                                         22AG9  .
       CALL CFV   ( R2 , L1 , L1 , NK , -1.)                               22AG9  .
           IF ( ITMPP.GE.1 )   PRINT 93, ((R2(I,J),J=1,NK),I=1,NK )         22AG9  .
C         CALCULATE RECURSION COEFFIECENT C                                04JA8  .
CCCCC     C # D*EE                                                         22AG9  .
       CALL MFB   ( R2 , EE , R1 , L1 , NK , N1 )                          22AG9  .
C         SAVE RECURSION COEFFICIENT C ON FILE 2                           22AG9  .
       WRITE ( 2'J1) (( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      WRITE%2L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
           IF ( ITMPP.GE.2 )   PRINT 93, ((R1(I,J),J=1,NK),I=1,NK )         22AG9  .
C         SAVE RECURSION MULTIPLIER D ON TAPE 3                            22AG9  .
       WRITE ( 3'ID3)(( R2(I,K), I=1,NK), K=1,NK )                         01DE1I8M
           IF ( J.EQ.NL )    GO TO 280                                     01DE1I8M
       WRITE ( 3'ID3)(( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
       READ  ( 1'1 ) (( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      WRITE%3L   %% R2%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      WRITE%3L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      BACKSPACE 3                                                         22AG9CDC
C      READ %1L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      REWIND 1                                                            25MYOCDC
C         CALCULATE RECURSION COEFFIECENT B                                04JA8  .
CBBBB     B # D*EP1                                                        22AG9  .
       CALL MFFT  ( R2 , R1 , R3 , L1 , NK )                               22AG9  .
C         SAVE RECURSION COEFFICIENT B ON FILE 2                           22AG9  .
       WRITE ( 2'J2) (( R3(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      WRITE%2L   %% R3%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
           IF ( ITMPP.GE.2 )   PRINT 93, ((R3(I,J),J=1,NK),I=1,NK )         22AG9  .
C         FINAL CALULATION RECURSION COEFFIECENT A                         22AG9  .
CAAAA     A # D*% E*AM1 & ET2*AM2 - FF L                                   22AG9  .
  280  CALL MFFV  ( R2 , ATM, A  , L1 , 1  , NK )                          22AG9  .
           IF ( ITMPP.GE.2 )   PRINT 93,  ( A(I,1),I=1,NK)                 29JL9  .
                NKK = NK + NF - 1                                          25MYO  .
           DO 300  I = NF, NKK                                             25MYO  .
                IIII = I - NF + 1                                          23NO1  .
  300           WW(I,J) = A(IIII,1)                                        23NO1  .
C          IF % ML.GE.0 L       GO TO  1000                                22AG9CDC
C      READ %2L                                                            22AG9CDC
C      READ %2L                                                            22AG9CDC
 1000      CONTINUE                                                        04JA8  .
C                 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
C*****************************************************************************
C          BEGIN BACKWARD PASS  --  COMPUTE RECURSION EQUATION             04JA8
C*****************************************************************************
                ID2 = 2*NL - 3                                             01DE1I8M
       READ  ( 2'ID2)(( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
       READ  ( 2'ID2)(( R3(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      BACKSPACE 2                                                         20MY8CDC
C      BACKSPACE 2                                                         22AG9CDC
C      BACKSPACE 2                                                         20MY8CDC
C      BACKSPACE 2                                                         20MY8CDC
C      READ %2L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      READ %2L   %% R3%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      BACKSPACE 2                                                         20MY8CDC
C      BACKSPACE 2                                                         20MY8CDC
       CALL MFFV  ( R3, WW(NF,NL), AM1, L1, 1, NK )                        25MYO
       CALL ASFV  ( WW(NF,NL-1), AM1, WW(NF,NL-1), L1, 1, NK, +1 )         25MYO
                NLM2 = NL - 2                                              20MY8
C                 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
           DO 2000  L = NF , NLM2                                          20MY8  .
                J = NLM2 + NF - L                                          20MY8  .
           IF ( J .EQ. 1 )  GO TO 2100                                     01DE1I8M
       FIND ( 2'2*J - 3 )                                                  01DE1I8M
 2100      CONTINUE                                                        01DE1I8M
C      BACKSPACE 2                                                         17JA8CDC
C      BACKSPACE 2                                                         22AG9CDC
C         READ C AND B COEFFICIENTS FROM FILE 2                            22AG9  .
C      READ %2L   %% R1%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
C      READ %2L   %% R3%I,KL , I # 1,NKL , K # 1,NK L                      22AG9CDC
                ID2 = 2*J - 1                                              01DE1I8M
       READ  ( 2'ID2)(( R1(I,K), I=1,NK), K=1,NK )                         01DE1I8M
       READ  ( 2'ID2)(( R3(I,K), I=1,NK), K=1,NK )                         01DE1I8M
C      BACKSPACE 2                                                         17JA8CDC
C      BACKSPACE 2                                                         22AG9CDC
       CALL MFFV  ( R3, WW(NF,J+1), AM1, L1, 1, NK )                       25MYO  .
       CALL MFFV  ( R1, WW(NF,J+2), AM2, L1, 1, NK )                       25MYO  .
       CALL ASFV  ( AM1, AM2, AM1,  L1,   1,  NK, +1 )                     22AG9  .
       CALL ASFV  ( WW(NF,J), AM1, WW(NF,J), L1, 1, NK, +1 )               25MYO  .
 2000      CONTINUE                                                        04JA8  .
C                 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
       RETURN                                                              04JA8
       END                                                                 04JA8
```

```
      SUBROUTINE RFV ( X , Y , L1 , L5 , L2 )
      DOUBLE PRECISION  X, Y
C******** THIS ROUTINE REPLACES A FULL MATRIX OR A VECTOR
C         % X # Y L
      DIMENSION  X(L1,L5) , Y(L1,L5)
            M = 1
         IF( L1 .EQ. L5 ) M = L2
         DO 110  J = 1,M
         DO 100  I = 1 , L2
            X(I,J) = Y(I,J)
  100    CONTINUE
  110    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MBFV ( XB , YF , ZF , L1 , L5 , L2 , L8 )
      DOUBLE PRECISION  XB, YF, ZF, SUM
C******** THIS ROUTINE MULTIPLIES A BANDED MATRIX
C         TIMES A FULL MATRIX OR A VECTOR
C         % XB * YF # ZF L
      DIMENSION  XB( L1,L8 ) , YF( L1,L5 ) , ZF( L1,L5 )
            M1 = 1
         IF( L1 .EQ. L5 ) M1 = L2
            L4 = L8/2
            L6 = L4 + 1
            N1 = L2 - L4
         DO 110  M = 1,M1
         DO 105  I = L6,N1
            J = I - L6
            SUM = 0.0
         DO 100  K = 1,L8
            SUM = XB(I,K) * YF(K+J,M) + SUM
  100    CONTINUE
            ZF(I,M) = SUM
  105    CONTINUE
  110    CONTINUE
            K1 = 0
            I1 = 1
            I2 = L4
            I3 = 1
            I4 = L8
         IF( I2 ) 150, 900, 150
  150    DO 210  M = 1,M1
         DO 205  I = I1,I2
            SUM = 0.0
            N = 1
         DO  200 K = I3, I4
            SUM = XB(I,N) * YF(K,M) + SUM
            N = N + 1
  200    CONTINUE
            ZF(I,M) = SUM
  205    CONTINUE
  210    CONTINUE
         IF( K1 ) 900,300,900
  300       I1 = L2 - L4 + 1
            I2 = L2
            I3 = L2 - L8 + 1
            I4 = L2
            K1 = 1
         GO TO 150
  900 RETURN
      END
```

```
      SUBROUTINE ABF ( YB , XF , ZF , L1 , L2 , L8 )
      DOUBLE PRECISION  YB, XF, ZF
C******** THIS ROUTINE ADDS A BANDED MATRIX
C*        TO A FULL MATRIX
C         % YB & XF # ZF  OR  XF & YB # ZF L
      DIMENSION YB( L1,L8 ) , XF( L1,L1 ) , ZF(L1,L1)
            L4 = L8/2
            N1 = L2 - L4
            L6 = L4 + 1
         DO 50  I = 1,L2
         DO 40  J = 1,L2
            ZF(I,J) = XF(I,J)
   40    CONTINUE
   50    CONTINUE
         DO 110  I = L6,N1
            J = I - L6
         DO 100  K = 1,L8
            ZF(I,K+J) = YB(I,K) + XF(I,K+J)
  100    CONTINUE
  110    CONTINUE
            K1 = 0
            I1 = 1
            I2 = L4
            I3 = 1
            I4 = L8
         IF( I2 ) 150, 900, 150
  150    DO 210 I = I1,I2
            N = 1
         DO  200 K = I3, I4
            ZF(I,K) = YB(I,N) + XF(I,K)
            N = N + 1
  200    CONTINUE
  210    CONTINUE
         IF(K1) 900, 300, 900
  300       I1 = L2 - L4 + 1
            I2 = L2
            I3 = L2 - L8 + 1
            I4 = L2
            K1 = 1
         GO TO 150
  900    RETURN
         END
```

```
      SUBROUTINE MFFV ( X , Y , Z , L1 , L5 , L2 )
      DOUBLE PRECISION  X, SUM, Y, Z
C******** THIS ROUTINE MULTIPLIES A FULL MATRIX
C         TIMES A FULL MATRIX OR A VECTOR
C         % X * Y # Z L
      DIMENSION X(L1,L1) , Y(L1,L5) , Z(L1,L5)
            M = 1
         IF( L1 .EQ. L5 ) M = L2
         DO 110  J = 1,M
         DO 105  I = 1,L2
            SUM = 0.0
         DO 100  K = 1,L2
            SUM = SUM + X(I,K) * Y(K,J)
  100    CONTINUE
            Z(I,J) = SUM
  105    CONTINUE
  110    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ASFV ( X , Y , Z , L1 , L5 , L2 , SIGN )                       20MY8
      DOUBLE PRECISION   X, Y, Z                                                    18M
C******* THIS ROUTINE ADDS OR SUBTRACTS 2 FULL MATRICES OR 2 VECTORS            20MY8
C         % X - Y # Z  OR  X & Y # Z L                                           13DE7
      DIMENSION X(L1,L5) , Y(L1,L5) , Z(L1,L5)                                  13DE7
            M = 1                                                               20MY8
         IF( L1 .EQ. L5 ) M = L2                                                20MY8
         IF ( SIGN ) 190, 50, 50                                                13DE7
   50    DO 110  J = 1,M                                                        13DE7
         DO 100  I = 1,L2                                                       13DE7
            Z(I,J) = X(I,J) + Y(I,J)                                            13DE7
  100    CONTINUE                                                               13DE7
  110    CONTINUE                                                               13DE7
         GO TO 300                                                              13DE7
  190    DO 210  J = 1,M                                                        13DE7
         DO 200  I = 1,L2                                                       13DE7
            Z(I,J) = X(I,J) - Y(I,J)                                            13DE7
  200    CONTINUE                                                               13DE7
  210    CONTINUE                                                               13DE7
  300 RETURN                                                                    13DE7
      END                                                                       13DE7
```

```
      SUBROUTINE SMFF  ( X , Y , Z , L1, L2 )                                   19FE8
      DOUBLE PRECISION  X, SUM, Y, Z                                               18M
C******* THIS ROUTINE MULTIPLIES TWO FULL MATRICES UNDER THE ASSUMPTION 05MR8
C         THAT THEIR PRODUCT WILL BE SYMMETRIC % X,Y, AND Z ARE FULL       05MR8
C         DIMENSIONED BUT ONLY THE LOWER HALF OF EACH IS USED L            05MR8
      DIMENSION  X(L1,L1) , Y(L1,L1) , Z(L1,L1)                            19FE8
         DO 110  J = 1 , L2                                                19FE8
         DO 105  I = 1 , J                                                 19FE8
            SUM = 0.0                                                      19FE8
         DO 100  K = 1 , L2                                                19FE8
            SUM = SUM + X(J,K) * Y(K,I)                                    19FE8
  100    CONTINUE                                                          19FE8
            Z(J,I) = SUM                                                   19FE8
  105    CONTINUE                                                          19FE8
  110    CONTINUE                                                          19FE8
      RETURN                                                               19FE8
      END                                                                  19FE8
```

```
      SUBROUTINE INVR6 ( X , L1 , L2 )                                     19FE8
      DOUBLE PRECISION  X, S, S1                                           31MY118M
C******* THIS ROUTINE TAKES THE INVERSE OF A SYMMETRIC POSITIVE - DEF      05MR8
C         MATRIX USING A COMPACTED CHOLESKI DECOMPOSITION PROCEDURE .      05MR8
C         A FULL DIMENSIONED MATRIX IS REQUIRED BUT ONLY THE LOWER         05MR8
C         HALF IS USED BY THE 3 ROUTINES DRIVEN BY INVR6                   05MR8
      DIMENSION X(L1,L1)                                                   19FE8
         IF ( L2 - 1 ) 600, 10, 20                                         01OC8
   10    IF (DABS (X(1,1)).LT.1.0E-10 )        GO TO  600                  02DE118M
C  10    IF % ABSF%X%1,1LL.LT.1.0E-10 L        GO TO  600                  02JU1COC
            X(1,1) = 1. / X(1,1)                                           05MR1
         GO TO  500                                                        01OC8
   20    IF ( L2 - 2 ) 30, 30, 40                                          01OC8
   30       S1 = X(1,1) * X(2,2) - X(1,2) * X(2,1)                         05MR1
         IF (DABS (S1).LT.1.0E-10 )      GO TO  600                        02DE118M
C        IF % ABSF%S1L.LT.1.0E-10 L      GO TO  600                        02JU1CDC
            S1 = 1./ S1                                                    01OC8
            S = X(1,1)                                                     05MR1
            X(1,1) = S1 * X(2,2)                                           05MR1
            X(2,2) = S1 * S                                                01OC8
            X(1,2) = -S1 * X(1,2)                                          01OC8
            X(2,1) = -S1 * X(2,1)                                          01OC8
         GO TO 500                                                         01OC8
   40 CALL FIX1 ( X, L1, L2 )                                              04MR018M
C  40    CONTINUE                                                          19OCOCDC
      CALL DCOM1  ( X , L1, L2 )                                           04MR0
      CALL INVLT1 ( X , L1 , L2 )                                          19FE8
      CALL MLTXL  ( X , L1 , L2 )                                          05MR8
      CALL FIX1 ( X, L1, L2 )                                              29JL918M
         DO 100  I = 2 , L2                                                19FE8
            KC = I - 1                                                     19FE8
         DO 50  J = 1 , KC                                                 19FE8
            X(J,I) = X(I,J)                                                19FE8
   50    CONTINUE                                                          19FE8
  100    CONTINUE                                                          19FE8
  500 RETURN                                                               01OC8
  600 PRINT 601,((X(I,J), J=1,L2), I=1,L2)                                 01OC8
  601 FORMAT ( 1H1,30H   SINGULAR MATRIX ENCOUNTERED ,/, 2(5X,1P2E15.7) )01OC818M
C 601 FORMAT % 1H1,30H   SINGULAR MATRIX ENCOUNTERED ,/, 2%5X,  2E15.7L L01OC8CDC
      STOP
      END                                                                  19FE8
```

```
      SUBROUTINE DCOM1 ( X , L1 , L2 )                                  19FE8
      DOUBLE PRECISION  X, S, S1, T                                     01DE11BM
      DIMENSION  X(L1,L1) , T(100)                                      12MR8
   10 FORMAT (/, 45H     NON-POSITIVE DEFINITE MATRIX ENCOUNTERED    )  05MR1
   15 FORMAT ( /,5X,1P13E9.2 )                                          29FE21BM
C  15 FORMAT % /,5X,  13E10.3 L                                         20MR9CDC
         DO 20  I = 1 , L2                                              12MR8
            T(I) = X(I,I)                                               12MR8
   20    CONTINUE                                                       12MR8
         IF ( X(1,1) .LE. 0.0 )  GO TO 4000                             05MR8
            X(1,1) = DSQRT ( X(1,1) )                                   19FE81BM
C           X%1,1L #  SQRT % X%1,1L L                                   19FE8CDC
            S1 = 1.0 / X(1,1)                                           19FE8
         DO 50  I = 2 , L2                                              19FE8
            X(I,1) = X(I,1) * S1                                        19FE8
   50    CONTINUE                                                       19FE8
            L2M1 = L2 - 1                                               19FE8
C                         . . . . . . . . . . . . . . . . . . . . . . . . . . . .
         DO 200  J = 2 , L2M1                                           19FE8  .
            S = 0.0                                                     19FE8  .
            JM1 = J - 1                                                 19FE8  .
         DO 120  K = 1 , JM1                                            19FE8 ..
            S = S + X(J,K) * X(J,K)                                     19FE8 ..
  120    CONTINUE                                                       19FE8 ..
         IF ( X(J,J) .LE. S )  GO TO 4000                               05MR8  .
            X(J,J) = DSQRT ( X(J,J) - S )                               19FE81BM
C           X%J,JL #  SQRT % X%J,JL - S L                               19FE8CDC
           S1 = 1.0 / X(J,J)                                            19FE8  .
           JP1 = J + 1                                                  19FE8  .
         DO 190  I = JP1 , L2                                           19FE8 ..
            S = 0.0                                                     19FE8 ..
         DO 180  K = 1 , JM1                                            19FE8...
            S = S + X(I,K) * X(J,K)                                     19FE8...
  180    CONTINUE                                                       19FE8...
            X(I,J) = ( X(I,J) - S ) * S1                                19FE8 ..
  190    CONTINUE                                                       19FE8 ..
  200    CONTINUE                                                       19FE8  .
C                         . . . . . . . . . . . . . . . . . . . . . . . . . . . .
            S = 0.0                                                     19FE8
         DO 250  K = 1, L2M1                                            19FE8
            S = S + X(L2,K) * X(L2,K)                                   19FE8
  250    CONTINUE                                                       19FE8
            S = X(L2,L2) - S                                            05MR8
         IF ( S .LE. 0.0 )  GO TO 4000                                  05MR8
            X(L2,L2) = DSQRT( S )                                       05MR81BM
C           X%L2,L2L #  SQRT% S L                                       05MR8CDC
      RETURN                                                            19FE8
 4000 PRINT 10                                                          12MR8
            X(1,1) = T(1)                                               12MR8
         DO 400  I = 2 , L2                                             09AP8
            K = I - 1                                                   12MR8
            X(I,I) = T(I)                                               12MR8
         DO 350  J = 1 , K                                              12MR8
            X(I,J) = X(J,I)                                             12MR8
  350    CONTINUE                                                       12MR8
  400    CONTINUE                                                       12MR8
         DO 500  I = 1, L2                                              20MR9
  500 PRINT 15, ( X(I,J), J=1,L2 )                                      20MR9
      STOP
      END                                                               19FE8
```

```
      SUBROUTINE INVLT1 ( X , L1 , L2 )                                 19FE8
      DOUBLE PRECISION  X, SUM                                             IBM
      DIMENSION X(L1,L1)                                                19FE8
         DO 50  I = 1 , L2                                              19FE8
            X(I,I) = 1.0 / X(I,I)                                       19FE8
   50    CONTINUE                                                       19FE8
            L2M1 = L2 - 1                                               19FE8
         DO 200  J = 1 , L2M1                                           19FE8
            JP1 = J+1                                                   19FE8
         DO 150  I = JP1 , L2                                           19FE8
            IM1 = I-1                                                   19FE8
            SUM = 0.0                                                   19FE8
         DO 120  K = J , IM1                                            19FE8
            SUM = SUM - X(I,K) * X(K,J)                                 19FE8
  120    CONTINUE                                                       19FE8
            X(I,J) = X(I,I) * SUM                                       19FE8
  150    CONTINUE                                                       19FE8
  200    CONTINUE                                                       19FE8
      RETURN                                                            19FE8
      END                                                               19FE8
```

```
      SUBROUTINE MLTXL ( X , L1 , L2 )                                  19FE8
      DOUBLE PRECISION  X, SUM                                             IBM
      DIMENSION X(L1,L1)                                                19FE8
         DO 200  I = 1 , L2                                             19FE8
         DO 150  J = 1 , I                                              19FE8
            SUM = 0.0                                                   19FE8
         DO 100  K = I , L2                                             19FE8
            SUM = SUM + X(K,I) * X(K,J)                                 19FE8
  100    CONTINUE                                                       19FE8
            X(I,J) = SUM                                                19FE8
  150    CONTINUE                                                       19FE8
  200    CONTINUE                                                       19FE8
      RETURN                                                            19FE8
      END                                                               19FE8
```

```
      SUBROUTINE CFV ( X , L1 , L5 , L2 , C )                           20MY8
      DOUBLE PRECISION  X                                                  IBM
C******** THIS ROUTINE MULTIPLIES A FULL MATRIX OR A VECTOR BY A CONSTANT13DE7
C          % X # C*X L                                                  13DE7
      DIMENSION X(L1,L5)                                                13DE7
            M = 1                                                       20MY8
         IF( L1 .EQ. L5 ) M = L2                                        20MY8
         DO 110  J = 1,M                                                13DE7
         DO 100  I = 1,L2                                               13DE7
            X(I,J) = X(I,J) * C                                         13DE7
  100    CONTINUE                                                       13DE7
  110    CONTINUE                                                       13DE7
      RETURN                                                            13DE7
      END                                                               13DE7
```

```
      SUBROUTINE MFB ( XF , YB , ZF , L1 , L2 , LB )                   07DE7
      DOUBLE PRECISION  XF, YB, ZF, SUM                                   IBM
C******* THIS ROUTINE MULTIPLIES A FULL MATRIX                          07DE7
C        TIMES A BANDED MATRIX                                          07DE7
C        % XF * YB # ZF L                                               07DE7
      DIMENSION  XF( L1,L1 ) , YB( LB,L1 ) , ZF( L1,L1 )                07DE7
            L4 = LB/2                                                   07DE7
            L6 = L4 + 1                                                 07DE7
            N1 = L2 - L4                                                07DE7
        DO 110  I = L6,N1                                               07DE7
            J = I - L6                                                  07DE7
        DO 105  M = 1,L2                                                07DE7
            SUM = 0.D                                                   06MY8
        DO 100  K = 1,LB                                                07DE7
            SUM = YB(K,I) * XF(M,K+J) + SUM                             06MY8
  100   CONTINUE                                                        10NO7
            ZF(M,I) = SUM                                               06MY8
  105   CONTINUE                                                        10NO7
  110   CONTINUE                                                        10NO7
            K1 = 0                                                      10NO7
            I1 = 1                                                      10NO7
            I2 = L4                                                     07DE7
            I3 = 1                                                      08DE7
            I4 = LB                                                     07DE7
        IF( I2 ) 150, 900, 150                                          07DE7
  150   DO  210 I = I1, I2                                              10NO7
        DO 205  M =  1,L2                                               07DE7
            SUM = 0.0                                                   06MY8
            N = 1                                                       08DE7
        DO  200 K = I3, I4                                              08DE7
            SUM = YB(N,I) * XF(M,K) + SUM                               06MY8
            N = N + 1                                                   08DE7
  200   CONTINUE                                                        10NO7
            ZF(M,I) = SUM                                               06MY8
  205   CONTINUE                                                        10NO7
  210   CONTINUE                                                        10NO7
        IF( K1 ) 900,300,900                                            10NO7
  300       I1 = L2 - L4 + 1                                            07DE7
            I2 = L2                                                     07DE7
            I3 = L2 - LB + 1                                            07DE7
            I4 = L2                                                     07DE7
            K1 = 1                                                      10NO7
        GO TO 150                                                       10NO7
  900 RETURN                                                            10NO7
      END                                                               10NO7
```

```
      SUBROUTINE MFFT ( X  , Y  , Z  , L1 , L2 )                        18MR8
      DOUBLE PRECISION  X, SUM, Y, Z                                      IBM
C******* THIS ROUTINE MULTIPLIES A FULL MATRIX                          18MR8
C        TIMES THE TRANSPOSE OF A SECOND FULL MATRIX                    18MR8
C        % X * YT # Z L                                                 18MR8
      DIMENSION  X(L1,L1) , Y(L1,L1) , Z(L1,L1)                         18MR8
        DO 110  J = 1 , L2                                              18MR8
        DO 105  I = 1 , L2                                              18MR8
            SUM = 0.0                                                   18MR8
        DO 100  K = 1 , L2                                              18MR8
            SUM = SUM + X(I,K) * Y(J,K)                                 18MR8
  100   CONTINUE                                                        18MR8
            Z(I,J) = SUM                                                18MR8
  105   CONTINUE                                                        18MR8
  110   CONTINUE                                                        18MR8
      RETURN                                                            18MR8
      END                                                               18MR8
```

```
      SUBROUTINE FIX1(D,L1,NK)                                          03JL9
      DOUBLE PRECISICN  D                                               01DE11BM
      DATA L3,L4/-1,-1/                                                 03JL9
      DIMENSION D(L1,L1),IFIX(100)                                      03JL9
            L3 = L3*L4                                                  03JL9
        IF (L3.LT.0)GO TO 500                                           03JL9
        DO 100 I = 1, NK                                                03JL9
            IFIX(I) = 0                                                 03JL9
        IF ( DABS( D(I,I) ).LT.1.0E+19 )     GO TO  100                 29JL9IBM
C       IF %  ABS% D%I,IL L.LT.1.0E&19 L     GO TO  100                 29JL9CDC
        DO 50 L= 1, NK                                                  03JL9
            D(I,L) = 0.0                                                03JL9
            D(L,I) = 0.0                                                03JL9
   50   CONTINUE                                                        03JL9
            D(I,I) = 1.0                                                03JL9
            IFIX(I) = 1                                                 03JL9
  100   CONTINUE                                                        03JL9
        GO TO 900                                                       03JL9
  500   CONTINUE                                                        03JL9
        DO 600 I = 1,NK                                                 03JL9
        IF (IFIX(I).EQ.1)  D(I,I) = 0.0                                 03JL9
  600   CONTINUE                                                        03JL9
  900   CONTINUE                                                        03JL9
      RETURN                                                            03JL9
      END                                                               03JL9
```

```
      SUBROUTINE TIC TOC (J)                                            01SE7
C-----THIS ROUTINE IS SPECIFICALLY FOR THE CDC6600. WHEN USING THE      01SE7
C     IBM360/50 SYSTEM THE INDICATED IBM CARDS WILL CALL THE SUBROUTINE 01SE7
C     PRTIME TO PRINTCUT THE REQUIRED TIME.                             01SE7
C                                                                       01SE7
C-----  TIC TOC %1L  #  COMPILE TIME                                    01SE7
C       TIC TOC %2L  #  ELAPSED CPU TIME                                01SE7
C       TIC TOC %3L  #  TIME FOR THIS PROBLEM                           01SE7
C       TIC TOC %4L  #  TIME FOR THIS PROBLEM  AND  ELAPSED CPU TIME    01SE7
C  10 FORMAT%///30X19HELAPSED CPU TIME # I5,8H MINUTESF9.3,8H SECONDS L 01SE7CDC
C  11 FORMAT%///30X15HCOMPILE TIME # ,I5,8H MINUTES,F9.3,8H SECONDS   L 01SE7CDC
C  12 FORMAT%///30X24HTIME FOR THIS PRCBLEM # ,I5,8H MINUTES,F9.3,      01SE7CDC
C    1          8H SECONDS  L                                           01SE7CDC
C           I # J - 2                                                   01SE7CDC
C       IF % I-1 L 40, 30, 30                                           01SE7CDC
C  30       FI4 # F                                                     01SE7CDC
C  40 CALL SECOND %FL                                                   01SE7CDC
   40 CALL PRTIME                                                       01DE11BM
        GO TO 990                                                       01DE11BM
C           II1 # F                                                     01SE1CDC
C           I1 # II1 / 60                                               01SE1CDC
C           FI2 #  F - I1*60                                            01SE1CDC
C       IF % I L 50, 70, 60                                             01SE1CDC
C  50 PRINT 11, I1, FI2                                                 01SE1CDC
C       GO TO 990                                                       01SE1CDC
C  60       FI3 # F - FI4                                               01SE1CDC
C           I2 # FI3 / 60                                               01SE1CDC
C           FI3 # FI3 - I2*60                                           01SE1CDC
C     PRINT 12, I2, FI3                                                 01SE1CDC
C       IF % I-1 L 990, 990, 70                                         01SE1CDC
C  70 PRINT 10, I1, FI2                                                 01SE1CDC
  990   CONTINUE                                                        01SE1
      RETURN                                                            01SE1
      END                                                               01SE1
```

APPENDIX 4

LISTING OF INPUT DATA FOR PROBLEMS 1001 THROUGH 1004

```
I-BEAM STRUCTURE        SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF TWO THREE-SPAN CONTINUOUS       RUN 29 FEB  1972  JJP    FT-KIP UNITS
1001A      DEAD LOAD OF GIRDERS AND NON-COMPOSITE DECK
    0    0    0    0    0    0    0    0
    1   47    0    0    0    0    3    4                      1
   24   32               1.450E+00 5.000E+00
    0    0   24   32                          1.000E-03 1.000E-03
    2    0    2   32                                    6.135E+05-1.772E+00
    2    1    2   31                                    6.135E+05-1.772E+00
    7    0    7   32                                    6.135E+05-1.881E+00
    7    1    7   31                                    6.135E+05-1.881E+00
   12    0   12   32                                    6.135E+05-1.881E+00
   12    1   12   31                                    6.135E+05-1.881E+00
   17    0   17   32                                    6.135E+05-1.881E+00
   17    1   17   31                                    6.135E+05-1.881E+00
   22    0   22   32                                    6.135E+05-1.772E+00
   22    1   22   31                                    6.135E+05-1.772E+00
    2    0   22    0                          2.663E+04
    3    0   21    0                          2.663E+04
    2    5   22    5                          3.256E+04
    3    5   21    5                          3.256E+04
    2    9   22    9                          3.256E+04
    3    9   21    9                          3.256E+04
    2   14   22   14                          3.256E+04
    3   14   21   14                          3.256E+04
    2   18   22   18                          3.256E+04
    3   18   21   18                          3.256E+04
    2   23   22   23                          3.256E+04
    3   23   21   23                          3.256E+04
    2   27   22   27                          3.256E+04
    3   27   21   27                          3.256E+04
    2   32   22   32                          2.663E+04
    3   32   21   32                          2.663E+04
    2    0    2    0                                              1.000E+14
    7    0    7    0                                              1.000E+14
   12    0   12    0                                              1.000E+14
   17    0   17    0                                              1.000E+14
   22    0   22    0                                              1.000E+14
    2   10    2   10                                              1.000E+14
    7   10    7   10                                              1.000E+14
   12   10   12   10                                              1.000E+14
   17   10   17   10                                              1.000E+14
   22   10   22   10                                              1.000E+14
    2   22    2   22                                              1.000E+14
    7   22    7   22                                              1.000E+14
   12   22   12   22                                              1.000E+14
   17   22   17   22                                              1.000E+14
   22   22   22   22                                              1.000E+14
    2   32    2   32                                              1.000E+14
    7   32    7   32                                              1.000E+14
   12   32   12   32                                              1.000E+14
   17   32   17   32                                              1.000E+14
   22   32   22   32                                              1.000E+14
   12    0   12   32    1    0    2    0
   17    0   17   32    1    0    2    0
   22    0   22   32    1    0    2    0
         0         0
         4         5
         9        10
        14        18
```

```
1002A      2 HS20 TRUCKS -- MAX POS MOMENT CNTR SPAN, GIRDER NO 2
    0    0    0    0    0    0    0    0         0
    1   55    0    1    0   40    5    5                             1
   24   32               1.450E+00 5.000E+00 1.500E-01
    0    0   24   32 1.830E+03 1.830E+03
    0    1   24   31 1.830E+03 1.830E+03
    1    0   23   32 1.830E+03 1.830E+03
    1    1   23   31 1.830E+03 1.830E+03
    2    0    2   32                                  6.135E+05
    2    1    2   31                                  6.135E+05
    2   12    2   20                                  1.829E+06
    7    0    7   32                                  6.135E+05
    7    1    7   31                                  6.135E+05
    7   12    7   20                                  1.881E+06
   12    0   12   32                                  6.135E+05
   12    1   12   31                                  6.135E+05
   12   12   12   20                                  1.881E+06
   17    0   17   32                                  6.135E+05
   17    1   17   31                                  6.135E+05
   17   12   17   20                                  1.881E+06
   22    0   22   32                                  6.135E+05
   22    1   22   31                                  6.135E+05
   22   12   22   20                                  1.829E+06
    2    0   22    0                        2.663E+04
    3    0   21    0                        2.663E+04
    2    5   22    5                        3.256E+04
    3    5   21    5                        3.256E+04
    2    9   22    9                        3.256E+04
    3    9   21    9                        3.256E+04
    2   14   22   14                        3.256E+04
    3   14   21   14                        3.256E+04
    2   18   22   18                        3.256E+04
    3   18   21   18                        3.256E+04
    2   23   22   23                        3.256E+04
    3   23   21   23                        3.256E+04
    2   27   22   27                        3.256E+04
    3   27   21   27                        3.256E+04
    2   32   22   32                        2.663E+04
    3   32   21   32                        2.663E+04
    2    0    2    0                                            1.000E+14
    7    0    7    0                                            1.000E+14
   12    0   12    0                                            1.000E+14
   17    0   17    0                                            1.000E+14
   22    0   22    0                                            1.000E+14
    2   10    2   10                                            1.000E+14
    7   10    7   10                                            1.000E+14
   12   10   12   10                                            1.000E+14
   17   10   17   10                                            1.000E+14
   22   10   22   10                                            1.000E+14
    2   22    2   22                                            1.000E+14
    7   22    7   22                                            1.000E+14
   12   22   12   22                                            1.000E+14
   17   22   17   22                                            1.000E+14
   22   22   22   22                                            1.000E+14
    2   32    2   32                                            1.000E+14
    7   32    7   32                                            1.000E+14
   12   32   12   32                                            1.000E+14
   17   32   17   32                                            1.000E+14
   22   32   22   32                                            1.000E+14
    1    1   24   32 6.210E+03
```

```
  2   13    2   13                                       -1,300E+00
  3   13    3   13                                       -2,000E+00
  6   13    6   13                                       -7,000E-01
  7   13    7   13                                       -3,400E+00
  9   13    9   13                                       -1,700E+00
 10   13   10   13                                       -2,400E+00
 13   13   13   13                                       -1,100E+00
 14   13   14   13                                       -3,000E+00
  2   14    2   14                                       -3,000E-01
  2   14    2   14                                       -7,000E-01
  6   14    6   14                                       -2,000E-01
  7   14    7   14                                       -0,000E-01
  9   14    9   14                                       -4,000E-01
 10   14   10   14                                       -6,000E-01
 13   14   13   14                                       -3,000E-01
 14   14   14   14                                       -7,000E-01
  2   16    2   16                                       -6,300E+00
  3   16    3   16                                       -1,400E+01
  6   16    6   16                                       -3,500E+00
  7   16    7   16                                       -1,600E+01
  9   16    9   16                                       -8,400E+00
 10   16   10   16                                       -1,190E+01
 13   16   13   16                                       -5,600E+00
 14   16   14   16                                       -1,470E+01
  2   10    2   10                                       -1,300E+00
  3   10    3   10                                       -2,000E+00
  6   10    6   10                                       -6,000E-01
  7   10    7   10                                       -3,400E+00
  9   10    9   10                                       -1,700E+00
 10   10   10   10                                       -2,400E+00
 13   10   13   10                                       -1,100E+00
 14   10   14   10                                       -2,900E+00
  2   19    2   19                                       -5,000E+00
  3   10    3   19                                       -1,120E+01
  6   19    6   19                                       -2,000E+00
  7   19    7   19                                       -1,340E+01
  9   19    9   19                                       -6,700E+00
 10   19   10   19                                       -9,900E+00
 13   19   13   19                                       -4,500E+00
 14   19   14   19                                       -1,180E+01
  2    0    2   32     1         2
  7    0    7   32     1         2
 12    0   12   32     1         2
 17    0   17   32     1         2
 22    0   22   32     1         2
       0         0
       9        11
      13        19
      21        23
      32        32
1003A     LANE LOADING FOR MAX NEG MOMENT    COMPOSITE DECK WHERE MOM IS POS
  0    0    0    0     0    0    0    0         0
  1   60    0    1     0   13    5    6                          1
 24   32              1,450E+00 5,000E+00 1,500E-01
  0    0   24   32 1,030E+03 1,030E+03
  0    1   24   31 1,030E+03 1,030E+03
  1    0   03   32 1,030E+03 1,030E+03
  1    1   23   31 1,030E+03 1,030E+03
  2    0    2   32                                  6,135E+05
  2    1    2   31                                  6,135E+05
  2    1    2    7                                  1,029E+06
  2   12    2   20                                  1,029E+06
  7    0    7   32                                  6,135E+05
  7    1    7   31                                  6,135E+05
  7    1    7    7                                  1,001E+06
  7   12    7   20                                  1,001E+06
```

```
 12    0   12   32                                  6,135E+05
 12    1   12   31                                  6,135E+05
 12    1   12    7                                  1,881E+06
 12   12   12   20                                  1,881E+06
 17    0   17   32                                  6,135E+05
 17    1   17   31                                  6,135E+05
 17    1   17    7                                  1,881E+06
 17   12   17   20                                  1,881E+06
 22    0   22   32                                  6,135E+05
 22    1   22   31                                  6,135E+05
 22    1   22    7                                  1,829E+06
 22   12   22   20                                  1,829E+06
  2    0   22    0                        2,663E+04
  3    0   21    0                        2,663E+04
  2    5   22    5                        3,256E+04
  3    5   21    5                        3,256E+04
  2    9   22    9                        3,256E+04
  3    9   21    9                        3,256E+04
  2   14   22   14                        3,256E+04
  3   14   21   14                        3,256E+04
  2   10   22   10                        3,256E+04
  3   10   21   10                        3,256E+04
  2   23   22   23                        3,256E+04
  3   23   21   23                        3,256E+04
  2   27   22   27                        3,256E+04
  3   27   21   27                        3,256E+04
  2   32   22   32                        2,663E+04
  3   32   21   32                        2,663E+04
  2    0    2    0                                                      1,000E+14
  7    0    7    0                                                      1,000E+14
 12    0   12    0                                                      1,000E+14
 17    0   17    0                                                      1,000E+14
 22    0   22    0                                                      1,000E+14
  2   10    2   10                                                      1,000E+14
  7   10    7   10                                                      1,000E+14
 12   10   12   10                                                      1,000E+14
 17   10   17   10                                                      1,000E+14
 22   10   22   10                                                      1,000E+14
  2   22    2   22                                                      1,000E+14
  7   22    7   22                                                      1,000E+14
 12   22   12   22                                                      1,000E+14
 17   22   17   22                                                      1,000E+14
 22   22   22   22                                                      1,000E+14
  2   32    2   32                                                      1,000E+14
  7   32    7   32                                                      1,000E+14
 12   32   12   32                                                      1,000E+14
 17   32   17   32                                                      1,000E+14
 22   32   22   32                                                      1,000E+14
  1    1   24   32 6,210E+03
  1    0    1   22                                          -5,600E-02
  1    1    1   21                                          -5,600E-02
 15    0   15   22                                          -1,790E-01
 15    1   15   21                                          -1,790E-01
  2    0   14    0                                          -2,960E-01
  2   22   14   22                                          -2,960E-01
  2    1   14   21                                          -5,930E-01
  1    5    1    5                                          -6,330E-01
 15    5   15    5                                          -2,013E+00
  2    5   14    5                                          -3,335E+00
  1   16    1   16                                          -6,330E-01
 15   16   15   16                                          -2,013E+00
  2   16   14   16                                          -3,335E+00
```

```
   2    0    2   32    1    0    2    0
   7    0    7   32    1    0    2    0
  12    0   12   32    1    0    2    0
  17    0   17   32    1    0    2    0
  22    0   22   32    1    0    2    0
        0         0
        4         6
        9        11
       14        18
       22        22
       32        32
1004A      2 HS20 TRUCKS STAGGERED IN ADJACENT SPANS FOR MAX NEG SUPPORT MOMENT
   0    0    0    0    0    0                   0
   1   59    0    1    0   40    6    6                          1
  24   32            1,450E+00 5,000E+00 1,500E-01
   0    0   24   32 1,030E+03 1,030E+03
   0    1   24   31 1,030E+03 1,030E+03
   0    1   23   32 1,030E+03 1,030E+03
   1    1   23   31 1,030E+03 1,030E+03
   2    0    2   32                                 6,135E+05
   2    1    2   31                                 6,135E+05
   2   12    2   20                                 1,029E+06
   7    0    7   32                                 6,135E+05
   7    1    7   31                                 6,135E+05
   7    1    7    7                                 1,001E+06
   7   12    7   20                                 1,001E+06
  12    0   12   32                                 6,135E+05
  12    1   12   31                                 6,135E+05
  12    1   12    8                                 1,001E+06
  12   13   12   21                                 1,001E+06
  17    0   17   32                                 6,135E+05
  17    1   17   31                                 6,135E+05
  17    1   17    9                                 1,001E+06
  17   16   17   28                                 1,001E+06
  22    0   22   32                                 6,135E+05
  22    1   22   31                                 6,135E+05
  22    1   22   11                                 1,029E+06
  22   20   22   31                                 1,029E+06
   2    0   22    0                      2,663E+04
   3    0   21    0                      2,663E+04
   2    5   22    5                      3,256E+04
   3    5   21    5                      3,256E+04
   2    9   22    9                      3,256E+04
   3    9   21    9                      3,256E+04
   2   14   22   14                      3,256E+04
   3   14   21   14                      3,256E+04
   2   18   22   18                      3,256E+04
   3   18   21   18                      3,256E+04
   2   23   22   23                      3,256E+04
   3   23   21   23                      3,256E+04
   2   27   22   27                      3,256E+04
   3   27   21   27                      3,256E+04
   2   32   22   32                      2,663E+04
   3   32   21   32                      2,663E+04
   2    0    2    0                                          1,000E+14
   7    0    7    0                                          1,000E+14
  12    0   12    0                                          1,000E+14
  17    0   17    0                                          1,000E+14
  22    0   22    0                                          1,000E+14
   2   10    2   10                                          1,000E+14
   7   10    7   10                                          1,000E+14
  12   10   12   10                                          1,000E+14
  17   10   17   10                                          1,000E+14
  22   10   22   10                                          1,000E+14
```

```
   2   22    2   22                                                      1,000E+14
   7   22    7   22                                                      1,000E+14
  12   22   12   22                                                      1,000E+14
  17   22   17   22                                                      1,000E+14
  22   22   22   22                                                      1,000E+14
   2   32    2   32                                                      1,000E+14
   7   32    7   32                                                      1,000E+14
  12   32   12   32                                                      1,000E+14
  17   32   17   32                                                      1,000E+14
  22   32   22   32                                                      1,000E+14
   1    1   24   32 6,210E+03
   9    2    9    2                                          -1,700E+00
  10    2   10    2                                          -2,400E+00
  13    2   13    2                                          -1,100E+00
  14    2   14    2                                          -3,000E+00
   9    3    9    3                                          -4,000E-01
  10    3   10    3                                          -6,000E-01
  13    3   13    3                                          -3,000E-01
  14    3   14    5                                          -7,000E-01
   9    5    9    5                                          -8,500E+00
  10    5   10    5                                          -1,200E+01
  13    5   13    5                                          -5,600E+00
  14    5   14    5                                          -1,480E+01
   9    7    9    7                                          -1,700E+00
  10    7   10    7                                          -2,400E+00
  13    7   13    7                                          -1,100E+00
  14    7   14    7                                          -3,000E+00
   9    8    9    8                                          -6,800E+00
  10    8   10    8                                          -9,600E+00
  13    8   13    8                                          -4,500E+00
  14    8   14    8                                          -1,100E+01
   2   13    2   13                                          -5,100E+00
   3   13    3   13                                          -1,130E+01
   6   13    6   13                                          -2,800E+00
   7   13    7   13                                          -1,350E+01
   2   14    2   14                                          -1,300E+00
   3   14    3   14                                          -2,800E+00
   6   14    6   14                                          -7,000E-01
   7   14    7   14                                          -3,400E+00
   2   16    2   16                                          -6,400E+00
   3   16    3   16                                          -1,410E+01
   6   16    6   16                                          -3,500E+00
   7   16    7   16                                          -1,690E+01
   2   18    2   18                                          -3,000E-01
   3   18    3   18                                          -7,000E-01
   6   18    6   18                                          -2,000E-01
   7   18    7   18                                          -8,000E-01
   2   19    2   19                                          -1,300E+00
   3   19    3   19                                          -2,800E+00
   6   19    6   19                                          -7,000E-01
   7   19    7   19                                          -3,400E+00
   2    0    2   32    1         2
   7    0    7   32    1         2
  12    0   12   32    1         2
  17    0   17   32    1         2
  22    0   22   32    1         2
   0    5   24    5    1    2
        0         0
        5         5
        8        11
       14        18
       21        23
       32        32
```

APPENDIX 5

SELECTED OUTPUT FOR EXAMPLE PROBLEMS INCLUDING THREE-DIMENSIONAL PLOTS

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THD THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB
1001A DEAD LOAD OF GIRDERS AND NON-COMPOSITE DECK

TABLE 1. CONTROL DATA

	TABLE NUMBER 2	3	4	5	6	7	8	9
KEEP FROM PRECEDING PROBLEM (1=YES)	0	0	0	0	0	0	0	0
NUM CARDS INPUT THIS PROBLEM	1	47	0	0	0	0	3	4

MULTIPLE LOAD OPTION	0
STATICS CHECK OPTION	0
PRIN STRESS OPTION	0
PROFILE PLOT OPTION	0
3-D PLOT OPTION	1

TABLE 2. CONSTANTS

NUMBER OF INCREMENTS IN X DIRECTION	24
NUMBER OF INCREMENTS IN Y DIRECTION	32
INCREMENT LENGTH IN X DIRECTION	1.450E 00
INCREMENT LENGTH IN Y DIRECTION	5.000E 00
POISSONS RATIO	0.0
SLAB THICKNESS	0.0

These small "beam" stiffnesses added so that 3-D plot will be clear

TABLE 3. JOINT BENDING STIFFNESSES, LOADS, AND SUPPORTS

FROM JOINT		THRU JOINT		DX	DY	FX	FY	Q	S
0	0	24	32	0.0	0.0	1.000E-03	1.000E-03	0.0	0.0
2	0	2	32	0.0	0.0	0.0	6.135E 05	-1.772E 00	0.0
2	1	2	31	0.0	0.0	0.0	6.135E 05	-1.772E 00	0.0
7	0	7	32	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
7	1	7	31	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
12	0	12	32	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
12	1	12	31	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
17	0	17	32	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
17	1	17	31	0.0	0.0	0.0	6.135E 05	-1.881E 00	0.0
22	0	22	32	0.0	0.0	0.0	6.135E 05	-1.772E 00	0.0
22	1	22	31	0.0	0.0	0.0	6.135E 05	-1.772E 00	0.0
2	0	22	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	0	21	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	5	22	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	5	21	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	9	22	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	9	21	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	14	22	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	14	21	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	18	22	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	18	21	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	23	22	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	23	21	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	27	22	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	27	21	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	32	22	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	32	21	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	0	2	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	0	7	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	0	12	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	0	17	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	0	22	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	10	2	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	10	7	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	10	12	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	10	17	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	10	22	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	22	2	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	22	7	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	22	12	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	22	17	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	22	22	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	32	2	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	32	7	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	32	12	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	32	17	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	32	22	32	0.0	0.0	0.0	0.0	0.0	1.000E 14

Beam stiffness and dead load

Diafram stiffness

Supports

TABLE 4. JOINT RESTRAINTS AND APPLIED MOMENTS

FROM JOINT	THRU JOINT	RX	RY	TX	TY

NONE

TABLE 5. MESH TWISTING STIFFNESSES

FROM MESH	THRU MESH	C

NONE

TABLE 6. BAR AXIAL THRUSTS

FROM BAR	THRU BAR	PX	PY	PBX	PBY

NONE

TABLE 7. MULTIPLE LOADS

FROM JOINT	THRU JOINT	QM

NONE

TABLE 8. PROFILE OUTPUT AREAS

FROM JOINT		THRU JOINT		DEFL (1=YES)	X MOMENT (1=SLAB,2=BEAM)	Y MOMENT	PRIN MOM OR STRESS (1=YES)
12	0	12	32	1	0	2	0
17	0	17	32	1	0	2	0
22	0	22	32	1	0	2	0

TABLE 9. PRINTED OUTPUT LIMITS

FROM Y STA	THRU Y STA
0	0
4	5
9	10
14	18

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF TWO THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB (CONTD)
1001A DEAD LOAD OF GIRDERS AND NON-COMPOSITE DECK

RESULTS

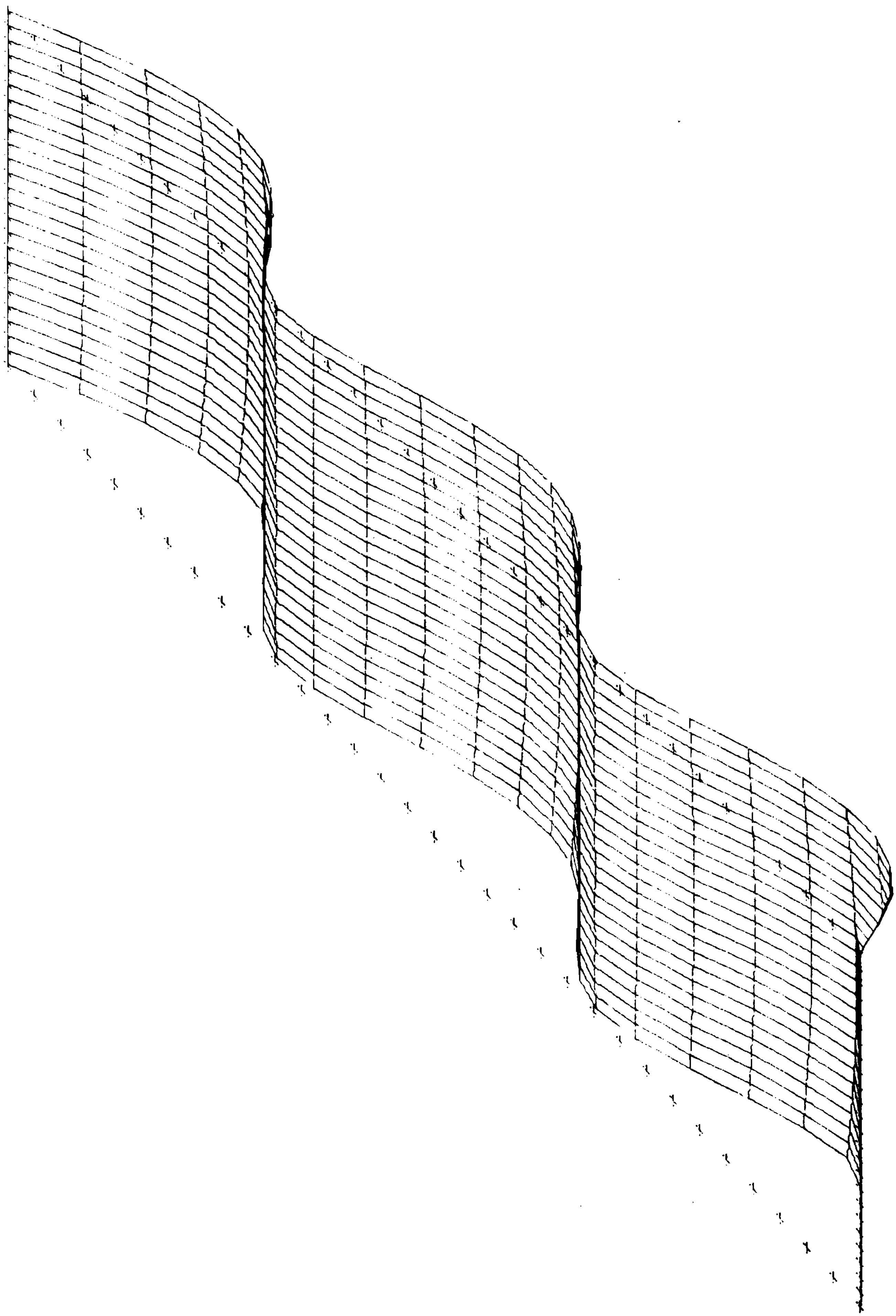
BEA MOMENTS ARE TOTAL PER BEAM

X	Y	DEFL	BEAM X MOMENT	BEAM Y MOMENT	SUPPORT REACTION
0	18	-1.534D-02	-5.776E-21	5.891E-08	0.0
1	18	-1.551D-02	6.428E-09	5.844E-08	0.0
2	18	-1.566D-02	1.492E-05	7.051E 01	0.0
3	18	-1.582D-02	1.464E-01	5.825E-08	0.0
4	18	-1.597D-02	2.929E-01	5.931E-08	0.0
5	18	-1.611D-02	4.393E-01	5.987E-08	0.0
6	18	-1.624D-02	5.857E-01	5.980E-08	0.0
7	18	-1.635D-02	7.321E-01	7.324E 01	0.0
8	18	-1.644D-02	6.805E-01	6.080E-08	0.0
9	18	-1.650D-02	6.288E-01	6.179E-08	0.0
10	18	-1.654D-02	5.771E-01	6.199E-08	0.0
11	18	-1.657D-02	5.255E-01	6.143E-08	0.0
12	18	-1.657D-02	4.738E-01	7.458E 01	0.0
13	18	-1.657D-02	5.255E-01	6.143E-08	0.0
14	18	-1.654D-02	5.771E-01	6.199E-08	0.0
15	18	-1.650D-02	6.288E-01	6.179E-08	0.0
16	18	-1.644D-02	6.805E-01	6.080E-08	0.0
17	18	-1.635D-02	7.321E-01	7.324E 01	0.0
18	18	-1.624D-02	5.857E-01	5.980E-08	0.0
19	18	-1.611D-02	4.393E-01	5.987E-08	0.0
20	18	-1.597D-02	2.929E-01	5.931E-08	0.0
21	18	-1.582D-02	1.464E-01	5.825E-08	0.0
22	18	-1.566D-02	1.492E-05	7.051E 01	0.0
23	18	-1.551D-02	6.428E-09	5.844E-08	0.0
24	18	-1.534D-02	8.251E-22	5.891E-08	0.0
0	17	-1.830D-02	-5.363E-21	7.684E-08	0.0
1	17	-1.851D-02	3.236E-09	7.782E-08	0.0
2	17	-1.872D-02	1.000E-08	9.709E 01	0.0
3	17	-1.890D-02	4.316E-09	7.977E-08	0.0
4	17	-1.908D-02	2.679E-09	8.031E-08	
5	17	-1.925D-02	4.858E-09	8.102E-08	
6	17	-1.941D-02	1.087E-08	8.18[illegible]	
7	17	-1.955D-02	2.1[illegible]		
8	17	[illegible]			
					0.0
					0.0
				...E-08	0.0
			...E-01	5.931E-08	0.0
		...11D-02	4.393E-01	5.987E-08	0.0
6	14	-1.624D-02	5.857E-01	5.980E-08	0.0
7	14	-1.635D-02	7.321E-01	7.324E 01	0.0
8	14	-1.644D-02	6.805E-01	6.080E-08	0.0
9	14	-1.650D-02	6.288E-01	6.179E-08	0.0
10	14	-1.654D-02	5.771E-01	6.199E-08	0.0
11	14	-1.657D-02	5.255E-01	6.143E-08	0.0
12	14	-1.657D-02	4.738E-01	7.458E 01	0.0
13	14	-1.657D-02	5.255E-01	6.143E-08	0.0
14	14	-1.654D-02	5.771E-01	6.199E-08	0.0
15	14	-1.650D-02	6.288E-01	6.179E-08	0.0
16	14	-1.644D-02	6.805E-01	6.080E-08	0.0
17	14	-1.635D-02	7.321E-01	7.324E 01	0.0
18	14	-1.624D-02	5.857E-01	5.980E-08	0.0
19	14	-1.611D-02	4.393E-01	5.987E-08	0.0
20	14	-1.597D-02	2.929E-01	5.931E-08	0.0
21	14	-1.582D-02	1.464E-01	5.825E-08	0.0
22	14	-1.566D-02	1.492E-05	7.051E 01	0.0
23	14	-1.551D-02	6.428E-09	5.844E-08	0.0
24	14	-1.534D-02	-1.650E-20	5.891E-08	0.0
0	10	-3.428D-04	-5.366E-22	-1.479E-07	0.0
1	10	-1.421D-04	-2.790E-08	-1.634E-07	0.0
2	10	-4.347D-13	-9.469E-08	-2.150E 02	4.347D 01
3	10	-5.700D-05	-9.307E-09	-1.720E-07	0.0
4	10	-1.336D-04	3.286E-08	-1.676E-07	0.0
5	10	-1.410D-04	3.645E-08	-1.689E-07	0.0
6	10	-7.186D-05	1.277E-09	-1.761E-07	0.0
7	10	-4.563D-13	-7.793E-08	-2.250E 02	4.563D 01
8	10	-9.199D-05	3.558E-09	-1.772E-07	0.0
9	10	-1.765D-04	4.119E-08	-1.712E-07	0.0
10	10	-1.744D-04	4.023E-08	-1.718E-07	0.0
11	10	-8.776D-05	5.233E-10	-1.787E-07	0.0
12	10	-4.608D-13	-8.348E-08	-2.278E 02	4.608D 01
13	10	-8.776D-05	5.233E-10	-1.787E-07	0.0
14	10	-1.744D-04	4.023E-08	-1.718E-07	0.0
15	10	-1.765D-04	4.119E-08	-1.712E-07	0.0
16	10	-9.199D-05	3.558E-09	-1.772E-07	0.0
17	10	-4.563D-13	-7.793E-08	-2.250E 02	4.563D 01
18	10	-7.186D-05	1.277E-09	-1.761E-07	0.0
19	10	-1.410D-04	3.645E-08	-1.689E-07	0.0
20	10	-1.336D-04	3.286E-08	-1.676E-07	0.0
21	10	-5.700D-05	-9.307E-09	-1.720E-07	0.0
22	10	-4.347D-13	-9.469E-08	-2.150E 02	4.347D 01
23	10	-1.421D-04	-2.790E-08	-1.634E-07	0.0
24	10	-3.428D-04	-1.031E-22	-1.479E-07	0.0
0	9	-2.460D-03	1.882E-21	-1.002E-07	0.0
1	9	-2.471D-03	-7.309E-09	-9.593E-08	0.0
2	9	-2.499D-03	2.361E-06	-1.134E 02	0.0
3	9	-2.526D-03	2.769E-02	-9.556E-08	
4	9	-2.552D-03	5.537E-02	-9.919E-08	
5	9	-2.577D-03	8.305E-0[illegible]		
					0.0
				...388E-22	0.0
		...13	-3.937E-08	-2.129E-14	1.349D 01
3	0	-2.457D-13	-8.503E-09	-1.388E-22	0.0
4	0	-6.921D-13	8.417E-09	-1.388E-22	0.0
5	0	-8.063D-13	1.130E-08	-1.388E-22	0.0
6	0	-4.743D-13	4.335E-12	-1.388E-22	0.0
7	0	-1.421D-13	-2.569E-08	-4.257E-14	1.421D 01
8	0	-8.240D-13	2.365E-09	-1.388E-22	0.0
9	0	-1.413D-12	1.588E-08	-1.388E-22	0.0
10	0	-1.374D-12	1.491E-08	-1.388E-22	0.0
11	0	-7.475D-13	-5.630E-10	-1.388E-22	0.0
12	0	-1.431D-13	-3.063E-08	-4.257E-14	1.431D 01
13	0	-7.475D-13	-5.630E-10	-1.388E-22	0.0
14	0	-1.374D-12	1.491E-08	-1.388E-22	0.0
15	0	-1.413D-12	1.588E-08	-1.388E-22	0.0
16	0	-8.240D-13	2.365E-09	-1.388E-22	0.0
17	0	-1.421D-13	-2.569E-08	-4.257E-14	1.421D 01
18	0	-4.743D-13	4.335E-12	-1.388E-22	0.0
19	0	-8.063D-13	1.130E-08	-1.388E-22	0.0
20	0	-6.921D-13	8.417E-09	-1.388E-22	0.0
21	0	-2.457D-13	-8.503E-09	-1.388E-22	0.0
22	0	-1.349D-13	-3.937E-08	-2.129E-14	1.349D 01
23	0	-3.132D-12	-1.291E-08	-1.388E-22	0.0
24	0	-2.714D-05	8.057E-24	-1.388E-22	0.0

STATICS CHECK. SUMMATION OF REACTIONS = 5.880D 02

MAXIMUM STATICS CHECK ERROR AT STA 17 10 = -6.956D-05

```
PROGRAM SLAB 49  -TEX HWY DEPT DECK-  MATLOCK,PANAK          REV DATE 29 FEB 72

I-BEAM STRUCTURE          SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF TWO THREE-SPAN CONTINUOUS        RUN 29 FEB  1972  JJP    FT-KIP UNITS


PROB (CONTD)
1001A      DEAD LOAD OF GIRDERS AND NON-COMPOSITE DECK


PROFILE OUTPUT AREAS
     X MOMENTS ACT IN X DIRECTION (ABOUT Y AXIS)
     THE PLOTTED RESULTS INDICATE THE RELATIVE VALUE EACH HAS WITHIN THAT LIST


             DEFLECTIONS      BETWEEN ( 12 ,  0 ) AND ( 12 , 32 )

 X , Y DEFLECTION

12  0 -1.431E-13                         *
12  1 -8.219E-03                 *
12  2 -1.517E-02           *
12  3 -1.998E-02       *
12  4 -2.213E-02    *
12  5 -2.153E-02     *
12  6 -1.842E-02        *
12  7 -1.348E-02            *
12  8 -7.749E-03                 *
12  9 -2.651E-03                      *
12 10 -4.608E-13                         *
12 11 -1.990E-03                      *
12 12 -6.507E-03                  *
12 13 -1.182E-02              *
12 14 -1.657E-02          *
12 15 -1.981E-02       *
12 16 -2.095E-02     *
12 17 -1.981E-02       *
12 18 -1.657E-02          *
12 19 -1.182E-02              *
12 20 -6.507E-03                  *
12 21 -1.990E-03                      *
12 22 -4.608E-13                         *
12 23 -2.651E-03                      *
12 24 -7.749E-03                 *
12 25 -1.348E-02            *
12 26 -1.842E-02        *
12 27 -2.153E-02     *
12 28 -2.213E-02    *
12 29 -1.998E-02       *
12 30 -1.517E-02           *
12 31 -8.219E-03
12


             BEAM Y MOMENT BETWEEN ( 17 ,  0 ) AND ( 17 , 32 )

 X , Y BEAM Y MOM

17  0 -4.257E-14                 *
17  1  6.166E 01                    *
17  2  1.045E 02                      *
17  3  1.286E 02                         *
17  4  1.338E 02                         *
17  5  1.202E 02                        *
17  6  8.877E 01                      *
17  7  3.851E 01                   *
17  8 -3.056E 01               *
17  9 -1.184E 02           *
17 10 -2.250E 02    *
17 11 -1.222E 02          *
17 12 -3.825E 01               *
17 13  2.690E 01                  *
17 14  7.324E 01                      *
17 15  1.015E 02                       *
17 16  1.109E 02                        *
17 17  1.015E 02                       *
17 18  7.324E 01                      *
17 19  2.690E 01                  *
17 20 -3.825E 01               *
17 21 -1.222E 02          *
17 22 -2.250E 02     *
17 23 -1.184E 02           *
17 24 -3.056E 01               *
17 25  3.851E 01                   *
17 26  8.877E 01                       *
17 27  1.202E 02                        *
17 28  1.338E 02                         *
17 29  1.286E 02                         *
17 30  1.045E 02                       *
17 31  6.166E 01                    *
17 32 -2.129E-14                 *


             BEAM Y MOMENT BETWEEN ( 22 ,  0 ) AND ( 22 , 32 )

 X , Y BEAM Y MOM

22  0 -2.129E-14                 *
22  1  5.859E 01                    *
22  2  9.945E 01                       *
22  3  1.226E 02                         *
22  4  1.280E 02                         *
22  5  1.157E 02                        *
22  6  8.504E 01                       *
22  7  3.663E 01                   *
22  8 -2.951E 01               *
22  9 -1.134E 02           *
22 10 -2.150E 02     *
22 11 -1.171E 02          *
22 12 -3.682E 01               *
22 13  2.570E 01                  *
22 14  7.051E 01                     *
22 15  9.709E 01                      *
22 16  1.059E 02                        *
22 17  9.709E 01                      *
22 18  7.051E 01                     *
22 19  2.570E 01                  *
22 20 -3.682E 01               *
22 21 -1.171E 02          *
22 22 -2.150E 02     *
22 23 -1.134E 02           *
22 24 -2.951E 01               *
22 25  3.663E 01                   *
22 26  8.504E 01                      *
22 27  1.157E 02                        *
22 28  1.280E 02                         *
22 29  1.226E 02                         *
22 30  9.945E 01                      *
22 31  5.859E 01                    *
22 32  0.0                       *

*****  TIME SINCE BEGINNING OF EXECUTION IS      233.82 SECONDS ,
       TIME SINCE LAST CALL OF PRTIME IS      233.82 SECONDS  *****
```


1001A

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THD THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB
1002A 2 HS20 TRUCKS — MAX POS MOMENT CNTR SPAN, GIRDER NO 2

TABLE 1. CONTROL DATA

	TABLE NUMBER 2	3	4	5	6	7	8	9
KEEP FROM PRECEDING PROBLEM (1=YES)	0	0	0	0	0	0	0	0
NUM CARDS INPUT THIS PROBLEM	1	55	0	1	0	40	5	5
MULTIPLE LOAD OPTION	0							
STATICS CHECK OPTION	0							
PRIN STRESS OPTION	0							
PROFILE PLOT OPTION	0							
3-D PLOT OPTION	1							

TABLE 2. CONSTANTS

NUMBER OF INCREMENTS IN X DIRECTION	24
NUMBER OF INCREMENTS IN Y DIRECTION	32
INCREMENT LENGTH IN X DIRECTION	1.450E 00
INCREMENT LENGTH IN Y DIRECTION	5.000E 00
POISSONS RATIO	1.500E-01
SLAB THICKNESS	0.0

TABLE 3. JOINT BENDING STIFFNESSES, LOADS, AND SUPPORTS

Slab bending stiffness (CX, DY, rows 0 0 – 1 1); Beam stiffness with composite zones (FY, rows 2 0 – 22 12); Diafram stiffness (rows 2 0 – 3 32); Supports (S, rows 2 0 – 22 32)

FROM JOINT		THRU JOINT		CX	DY	FX	FY	Q	S
0	0	24	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
0	1	24	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
1	0	23	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
1	1	23	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
2	0	2	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	1	2	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	12	2	20	0.0	0.0	0.0	1.829E 06	0.0	0.0
7	0	7	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	1	7	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	12	7	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
12	0	12	32	0.0	C.0	0.0	6.135E 05	0.0	0.0
12	1	12	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
12	12	12	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
17	0	17	32	0.0	C.0	0.0	6.135E 05	0.0	0.0
17	1	17	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
17	12	17	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
22	0	22	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	1	22	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	12	22	20	0.0	0.0	0.0	1.829E 06	0.0	0.0
2	0	22	0	0.0	C.0	2.663E 04	0.0	0.0	0.0
3	0	21	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	5	22	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	5	21	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	9	22	9	0.0	C.0	3.256E 04	0.0	0.0	0.0
3	9	21	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	14	22	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	14	21	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	18	22	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	18	21	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	23	22	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	23	21	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	27	22	27	0.0	C.0	3.256E 04	0.0	0.0	0.0
3	27	21	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	32	22	32	0.0	C.0	2.663E 04	0.0	0.0	0.0
3	32	21	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	0	2	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	0	7	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	0	12	0	0.0	C.0	0.0	0.0	0.0	1.000E 14
17	0	17	0	0.0	C.0	0.0	0.0	0.0	1.000E 14
22	0	22	0	0.0	C.0	0.0	0.0	0.0	1.000E 14
2	10	2	10	0.0	C.0	0.0	0.0	0.0	1.000E 14
7	10	7	10	0.0	C.0	0.0	0.0	0.0	1.000E 14
12	10	12	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	10	17	10	0.0	C.0	0.0	0.0	0.0	1.000E 14
22	10	22	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	22	2	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	22	7	22	0.0	C.0	0.0	0.0	0.0	1.000E 14
12	22	12	22	0.0	C.0	0.0	0.0	0.0	1.000E 14
17	22	17	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	22	22	22	0.0	C.0	0.0	0.0	0.0	1.000E 14
2	32	2	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	32	7	32	0.0	C.0	0.0	0.0	0.0	1.000E 14
12	32	12	32	0.0	C.0	0.0	0.0	0.0	1.000E 14
17	32	17	32	0.0	C.0	0.0	0.0	0.0	1.000E 14
22	32	22	32	0.0	0.0	0.0	0.0	0.0	1.000E 14

TABLE 4. JOINT RESTRAINTS AND APPLIED MOMENTS

FROM JOINT	THRU JOINT	RX	RY	TX	TY

NONE

TABLE 5. MESH TWISTING STIFFNESSES

Slab twisting stiffness

FROM MESH		THRU MESH		C
1	1	24	32	6.210E 03

TABLE 6. BAR AXIAL THRUSTS

FROM BAR	THRU BAR	PX	PY	PBX	PBY

NONE

TABLE 7. MULTIPLE LOADS

FROM JOINT		THRU JOINT		QM
2	13	2	13	-1.300E 00
3	13	3	13	-2.800E 00
6	13	6	13	-7.000E-01
7	13	7	13	-3.400E 00
9	13	9	13	-1.700E 00
10	13	10	13	-2.400E 00
13	13	13	13	-1.100E 00
14	13	14	13	-3.000E 00
2	14	2	14	-3.000E-01
2	14	2	14	-7.000E-01
6	14	6	14	-2.000E-01
7	14	7	14	-8.000E-01
9	14	9	14	-4.000E-01
10	14	10	14	-6.000E-01
13	14	13	14	-3.000E-01
14	14	14	14	-7.000E-01
2	16	2	16	-6.300E 00
3	16	3	16	-1.400E 01
6	16	6	16	-3.500E 00
7	16	7	16	-1.680E 01
9	16	9	16	-8.480E 00
10	16	10	16	-1.190E 01
13	16	13	16	-5.600E 00
14	16	14	16	-1.470E 01
2	18	2	18	-1.300E 00
3	18	3	18	-2.800E 00
6	18	6	18	-6.000E-01
7	18	7	18	-3.400E 00
9	18	9	18	-1.700E 00
10	18	10	18	-2.400E 00
13	18	13	18	-1.100E 00
14	18	14	18	-2.900E 00
2	19	2	19	-5.000E 00
3	19	3	19	-1.120E 01
6	19	6	19	-2.800E 00
7	19	7	19	-1.340E 01
9	19	9	19	-6.700E 00
10	19	10	19	-9.500E 00
13	19	13	19	-4.500E 00
14	19	14	19	-1.180E 01

Two HS-20 trucks apportioned as shown in Fig 15

TABLE 8. PROFILE OUTPUT AREAS

FROM JOINT		THRU JOINT		DEFL (1=YES)	X MOMENT (1=SLAB,2=BEAM)	Y MOMENT	PRIN MOM OR STRESS (1=YES)
2	0	2	32	1	0	2	0
7	0	7	32	1	0	2	0
12	0	12	32	1	0	2	0
17	0	17	32	1	0	2	0
22	0	22	32	1	0	2	0

TABLE 9. PRINTED OUTPUT LIMITS

FROM Y STA	THRU Y STA
0	0
9	11
13	19
21	23
32	32

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THE THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB (CONTD)
1002A 2 HS20 TRUCKS -- MAX POS MOMENT CNTR SPAN, GIRDER NO 2

RESULTS

SLAB X MOMENT AND X TWISTING MOMENT ACT IN THE X DIRECTION (ABOUT Y AXIS)
Y TWISTING MOMENT = -X TWISTING MOMENT, COUNTERCLOCKWISE BETA ANGLES ARE POSITIVE FROM THE X AXIS TO THE DIRECTION OF THE LARGEST PRINCIPAL MOMENT
SLAB MOMENTS ARE PER UNIT WIDTH

BEAM MOMENTS ARE TOTAL PER BEAM

X	Y	DEFL	SLAB X MOMENT / BEAM X MOMENT	SLAB Y MOMENT / BEAM Y MOMENT	SLAB X TWISTING MOMENT	LARGEST PRINCIPAL SLAB MOMENT	BETA X TO LARGEST MOMENT	SUPPORT REACTION
0	32	8.951D-05	1.593E-15	3.010E-16	1.400D-02	1.400E-02	45.0	0.0
			0.0	0.0				
1	32	3.286D-05	4.050E-02	-1.546E-09	3.538D-02	6.102E-02	30.1	0.0
			0.0	0.0				
2	32	3.562D-14	2.964E-02	5.897E-03	4.550D-02	7.130E-02	34.8	-3.562D 00
			2.885E-01	-8.550E-03				
3	32	-1.008D-05	1.452E-02	-4.746E-10	4.661D-02	5.443E-02	40.6	0.0
			2.162E-01	0.0				
4	32	-1.162D-05	7.240E-03	-4.713E-10	5.141D-02	5.516E-02	43.0	0.0
			1.078E-01	0.0				
5	32	-8.910D-06	2.599E-03	-7.752E-11	5.382D-02	5.514E-02	44.3	0.0
			3.869E-02	0.0				
6	32	-4.672D-06	7.376E-04	-2.301E-11	5.732D-02	5.769E-02	44.8	0.0
			1.098E-02	0.0				
7	32	2.657D-14	2.010E-03	2.989E-04	6.308D-02	6.424E-02	44.6	-2.657D 00
			2.925E-02	-4.334E-04				
8	32	5.827D-06	-3.903E-03	3.812E-10	6.929D-02	-7.127E-02	-44.2	0.0
			-5.810E-02	0.0				
9	32	9.360D-06	-6.518E-03	2.504E-10	7.416D-02	-7.749E-02	-43.7	0.0
			-9.703E-02	0.0				
10	32	9.063D-06	-6.020E-03	4.798E-10	7.886D-02	-8.193E-02	-43.9	0.0
			-8.961E-02	0.0				
11	32	5.228D-06	-2.371E-03	7.419E-11	8.472D-02	-8.592E-02	-44.6	0.0
			-3.529E-02	0.0				
12	32	2.146D-14	4.739E-03	7.049E-04	9.246D-02	9.520E-02	44.4	-2.146D 00
			6.897E-02	-1.022E-03				
13	32	-2.505D-06	1.110E-03	-4.200E-11	1.002D-01	1.007E-01	44.8	0.0
			1.652E-02	0.0				
14	32	-4.358D-06	7.606E-04	-3.552E-11	1.059D-01	1.063E-01	44.9	0.0
			1.132E-02	0.0				
15	32	-5.764D-06	3.199E-03	-8.541E-11	1.104D-01	1.120E-01	44.6	0.0
			4.762E-02	0.0				
16	32	-5.290D-06	8.196E-03	-2.490E-10	1.165D-01	1.207E-01	44.0	0.0
			1.220E-01	0.0				
17	32	1.781D-14	1.596E-02	2.374E-03	1.224D-01	1.318E-01	43.4	-1.781D 00
			2.323E-01	-3.443E-03				
18	32	1.446D-05	7.617E-04	-3.275E-11	1.271D-01	1.275E-01	44.9	0.0
			1.134E-02	0.0				
19	32	2.937D-05	-1.257E-02	7.856E-10	1.297D-01	-1.362E-01	-43.6	0.0
			-1.871E-01	0.0				
20	32	3.690D-05	-2.520E-02	8.965E-10	1.280D-01	-1.412E-01	-42.2	0.0
			-3.752E-01	0.0				
21	32	2.961D-05	-3.798E-02	1.313E-09	1.181D-01	-1.386E-01	-40.4	0.0
			-5.654E-01	0.0				
22	32	1.211D-15	-8.386E-02	-1.248E-02	1.159D-01	-1.694E-01	-36.4	-1.211D-01
			-6.105E-01	1.809E-02				

23	32	-7.781D-05	-9.242E-02	4.166E-09	9.127D-02	-1.485E-01	-31.6	0.0
			0.0	0.0				
24	32	-2.099D-04	4.361E-17	-1.673E-17	3.520D-02	3.520E-02	45.0	0.0
			0.0	0.0				
0	23	8.595D-03	9.142E-09	-3.244E-01	1.996D-03	-3.244E-01	-89.6	0.0
			0.0	0.0				
1	23	8.505D-03	5.773E-02	-6.727E-01	2.237D-03	-6.727E-01	-89.8	0.0
			0.0	0.0				
2	23	8.462D-03	1.574E-02	-7.177E-01	1.171D-03	-7.177E-0[illegible]		
			5.615E-01	-1.235E 02				
3	23	8.456D-03	-8.227E-02	-7.734E-01				
			3.071E-01					
4	23	8.459D-03						
							-63.2	0.0
					-4.288D-01	5.072E-01	-60.6	0.0
				0.0				
			-2.180E-01	1.914E-01	-4.298D-01	-4.893E-01	32.3	0.0
			-2.245E 00	0.0				
21	18	-5.422D-03	-1.711E-01	1.093E-01	-4.218D-01	-4.754E-01	35.8	0.0
			-1.706E 00	0.0				
22	18	-2.304D-03	-1.756E-01	1.255E-02	-4.073D-01	-4.995E-01	38.5	0.0
			-8.074E-01	1.661E 01				
23	18	7.625D-04	-1.556E-01	-7.992E-02	-3.948D-01	-5.143E-01	42.3	0.0
			0.0	0.0				
24	18	3.787D-03	7.015E-10	-7.122E-02	-1.959D-01	-2.347E-01	50.2	0.0
			0.0	0.0				
0	17	-4.102D-02	-9.775E-09	4.872E-01	4.540D-02	4.914E-01	84.7	0.0
			0.0	0.0				
1	17	-4.209D-02	4.628E-02	5.637E-01	1.058D-01	9.757E-01	83.5	0.0
			0.0	0.0				
2	17	-4.319D-02	1.519E-01	5.411E-01	1.678D-01	9.753E-01	78.5	0.0
			0.0	3.922E 02				
3	17	-4.429D-02	7.446E-01	9.299E-01	1.192D-01	9.882E-01	63.9	0.0
			0.0	0.0				
4	17	-4.521D-02	1.070E 00	1.002E 00	-1.681D-02	1.074E 00	-13.0	0.0
			0.0	0.0				
5	17	-4.586D-02	1.285E 00	1.057E 00	-7.099D-02	1.308E 00	-18.6	0.0
			0.0	0.0				
6	17	-4.618D-02	1.477E 00	1.188E 00	-7.150D-02	1.494E 00	-13.2	0.0
			0.0	0.0				
7	17	-4.612D-02	1.716E 00	1.257E 00	3.586D-02	1.719E 00	4.4	0.0
			0.0	4.343E 02				
8	17	-4.561D-02	1.904E 00	1.163E 00	1.294D-01	1.926E 00	9.6	0.0
			0.0	0.0				
9	17	-4.459D-02	2.105E 00	1.047E 00	2.201D-02	2.105E 00	1.2	0.0
			0.0	0.0				
10	17	-4.300D-02	2.028E 00	1.001E 00	-1.969D-01	2.065E 00	-10.5	0.0
			0.0	0.0				
11	17	-4.086D-02	1.642E 00	1.036E 00	-2.883D-01	1.757E 00	-21.8	0.0
			0.0	0.0				
12	17	-3.828D-02	1.322E 00	1.031E 00	-1.002D-01	1.353E 00	-17.3	0.0
			0.0	3.618E 02				
13	17	-3.536D-02	1.334E 00	8.313E-01	4.249D-02	1.338E 00	4.8	0.0
			0.0	0.0				
14	17	-3.208D-02	1.281E 00	6.362E-01	-1.112D-01	1.300E 00	-9.5	0.0
			0.0	0.0				
15	17	-2.845D-02	8.630E-01	5.437E-01	-2.956D-01	1.039E 00	-30.8	0.0
			0.0	0.0				
16	17	-2.460D-02	2.994E-01	4.778E-01	-3.505D-01	7.502E-01	-52.1	0.0
			0.0	0.0				
17	17	-2.067D-02	-2.707E-01	4.011E-01	-3.041D-01	5.183E-01	-68.9	0.0
			0.0	1.919E 02				
18	17	-1.685D-02	-3.241E-01	3.372E-01	-2.308D-01	4.098E-01	-72.5	0.0
			0.0	0.0				
19	17	-1.313D-02	-3.196E-01	2.561E-01	-1.917D-01	-3.776E-01	16.8	0.0
			0.0	0.0				
20	17	-9.522D-03	-2.919E-01	1.707E-01	-1.701D-01	-3.478E-01	18.2	0.0
			0.0	0.0				
21	17	-6.005D-03	-2.612E-01	8.715E-02	-1.568D-01	-3.214E-01	21.0	0.0

			0.0	0.0				
22	17	-2.569D-03	-2.423E-01	9.193E-03	-1.509D-01	-3.130E-01	25.1	0.0
			0.0	1.945E 01				
23	17	7.958D-04	-1.221E-01	-4.304E-02	-1.537D-01	-2.413E-01	37.8	0.0
			0.0	0.0				
24	17	4.126D-03	7.184E-10	-4.766E-02	-7.810D-02	-1.055E-01	53.5	0.0
			0.0	0.0				
0	16	-4.180D-02	-9.436E-09	3.438E-01	-3.654D-02	3.476E-01	-84.0	0.0
			0.0	0.0				
1	16	-4.300D-02	-2.398E-01	7.779E-01	-7.437D-02	7.833E-01	-85.8	0.0
			0.0	0.0				
2	16	-4.431D-02	-5.800E-01	8.573E-01	-7.459D-02	9.011E-01	-87.1	0.0
			0.0	4.204E 02				
3	16	-4.583D-02	2.325E 00	1.625E 00	-7.062D-02	2.332E 00	-5.7	0.0
			0.0	0.0				
4	16	-4.674D-02	1.733E 00	1.575E 00	-6.133D-02	1.754E 00	-18.9	0.0
			0.0	0.0				
5	16	-4.721D-02	1.428E 00	1.464E 00	-4.921D-02	1.498E 00	-55.0	0.0
			0.0	0.0				
6	16	-4.733D-02	1.236E 00	1.346E 00	-3.506D-02	1.356E 00	-73.8	0.0
			0.0	0.0				
7	16	-4.714D-02	5.942E-03	1.106E 00	-1.798D-02	1.107E 00	-89.1	0.0
			0.0	4.802E 02				
8	16	-4.700D-02	1.848E 00	1.599E 00	5.788D-04	1.848E 00	0.1	0.0
			0.0	0.0				
9	16	-4.638D-02	3.760E 00	2.110E 00	1.889D-02	3.760E 00	0.7	0.0
			0.0	0.0				
10	16	-4.476D-02	3.861E 00	2.086E 00	3.666D-02	3.862E 00	1.2	0.0
			0.0	0.0				
11	16	-4.209D-02	1.210E 00	1.357E 00	5.197D-02	1.374E 00	72.4	0.0
			0.0	0.0				
12	16	-3.912D-02	-1.501E 00	6.905E-01	6.477D-02	-1.502E 00	-1.7	0.0
			0.0	3.977E 02				
13	16	-3.663D-02	1.683E 00	1.366E 00	7.529D-02	1.700E 00	12.7	0.0
			0.0	0.0				
14	16	-3.370D-02	3.429E 00	1.775E 00	8.316D-02	3.433[illegible]		
			0.0	0.0				
15	16	-2.984D-02	1.693E 00	1.325E 00	8.[illegible]			
			0.0	0.0				
16	16	-2.555D-02	3.019E-01					
			0.0					-1.848D 00
17	16	-2.120D-0[illegible]						
						[illegible].561E-02	-44.8	0.0
					-8.950D-02	8.989E-02	-44.9	0.0
				-5.819E-11	-9.292D-02	9.432E-02	-44.6	0.0
			4.135E-02	0.0				
		[illegible]D-06	6.828E-03	-6.264E-10	-9.768D-02	1.012E-01	-44.0	0.0
			1.016E-01	0.0				
17	0	1.516D-14	1.311E-02	1.950E-03	-1.022D-01	1.099E-01	-43.4	-1.516D 00
			1.908E-01	-2.827E-03				
18	0	1.202D-05	5.807E-04	-2.595E-11	-1.059D-01	1.062E-01	-44.9	0.0
			8.644E-03	0.0				
19	0	2.439D-05	-1.044E-02	3.936E-10	-1.076D-01	-1.132E-01	43.6	0.0
			-1.554E-01	0.0				
20	0	3.062D-05	-2.090E-02	6.767E-10	-1.063D-01	-1.173E-01	42.2	0.0
			-3.112E-01	0.0				
21	0	2.457D-05	-3.150E-02	1.167E-09	-9.803D-02	-1.150E-01	40.4	0.0
			-4.689E-01	0.0				
22	0	1.455D-15	-6.953E-02	-1.034E-02	-9.617D-02	-1.406E-01	36.4	-1.455D-01
			-5.060E-01	1.500E-02				
23	0	-6.451D-05	-7.665E-02	8.166E-09	-7.575D-02	-1.232E-01	31.6	0.0
			0.0	0.0				
24	0	-1.741D-04	-4.595E-17	8.587E-19	-2.921D-02	-2.921E-02	45.0	0.0
			0.0	0.0				

STATICS CHECK.   SUMMATION OF REACTIONS =   1.828D 02

MAXIMUM STATICS CHECK ERROR AT STA   2 14 =   8.897D-05

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF TWO THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB (CONTD)
1002A 2 HS20 TRUCKS -- MAX POS MOMENT CNTR SPAN, GIRDER NO 2

PROFILE OUTPUT AREAS
X MOMENTS ACT IN X DIRECTION (ABOUT Y AXIS)
THE PLOTTED RESULTS INDICATE THE RELATIVE VALUE EACH HAS WITHIN THAT LIST

DEFLECTIONS BETWEEN (2 , 0) AND (2 , 32)

X	Y	DEFLECTION
2	0	3.095E-14
2	1	4.459E-03
2	2	8.638E-03
2	3	1.226E-02
2	4	1.506E-02
2	5	1.675E-02
2	6	1.706E-02
2	7	1.577E-02
2	8	1.263E-02
2	9	7.439E-03
2	10	-2.307E-13
2	11	-9.825E-03
2	12	-1.988E-02
2	13	-2.915E-02
2	14	-3.675E-02
2	15	-4.200E-02
2	16	-4.431E-02
2	17	-4.319E-02
2	18	-3.886E-02
2	19	-3.161E-02
2	20	-2.197E-02
2	21	-1.109E-02
2	22	-2.950E-13
2	23	8.462E-03
2	24	1.441E-02
2	25	1.803E-02
2	26	1.954E-02
2	27	1.921E-02
2	28	1.729E-02
2	29	1.409E-02
2	30	9.930E-03
2	31	5.127E-03
2	32	3.562E-14

DEFLECTIONS BETWEEN (7 , 0) AND (7 , 32)

X	Y	DEFLECTION
7	0	2.287E-14
7	1	3.890E-03
7	2	7.558E-03
7	3	1.078E-02
7	4	1.332E-02
7	5	1.496E-02
7	6	1.543E-02
7	7	1.448E-02
7	8	1.181E-02
7	9	7.099E-03
7	10	-2.781E-13
7	11	-9.912E-03
7	12	-2.039E-02
7	13	-3.019E-02
7	14	-3.839E-02
7	15	-4.430E-02
7	16	-4.714E-02
7	17	-4.612E-02
7	18	-4.160E-02
7	19	-3.392E-02
7	20	-2.350E-02
7	21	-1.173E-02
7	22	-4.072E-13
7	23	8.338E-03
7	24	1.383E-02
7	25	1.692E-02
7	26	1.800E-02
7	27	1.742E-02
7	28	1.550E-02
7	29	1.253E-02
7	30	8.782E-03
7	31	4.519E-03
7	32	2.657E-14

DEFLECTIONS BETWEEN (12 , 0) AND (12 , 32)

X	Y	DEFLECTION
12	0	1.848E-14
12	1	3.103E-03
12	2	6.034E-03
12	3	8.618E-03
12	4	1.068E-02
12	5	1.202E-02
12	6	1.246E-02
12	7	1.175E-02
12	8	9.631E-03
12	9	5.820E-03
12	10	-2.304E-13
12	11	-8.193E-03
12	12	-1.689E-02
12	13	-2.504E-02
12	14	-3.186E-02
12	15	-3.677E-02
12	16	-3.912E-02
12	17	-3.828E-02
12	18	-3.452E-02
12	19	-2.813E-02
12	20	-1.948E-02
12	21	-9.709E-03
12	22	-3.354E-13
12	23	6.858E-03
12	24	1.133E-02
12	25	[illegible]

X	Y	DEFLECTION
[illegible]	[illegible]	[illegible]
22	16	-2.675E-03
22	17	-2.569E-03
22	18	-2.304E-03
22	19	-1.903E-03
22	20	-1.390E-03
22	21	-7.841E-04
22	22	2.216E-14
22	23	9.234E-04
22	24	1.794E-03
22	25	2.486E-03
22	26	2.917E-03
22	27	3.047E-03
22	28	2.671E-03
22	29	2.418E-03
22	30	1.741E-03
22	31	9.094E-04
22	32	1.211E-15

BEAM Y MOMENT BETWEEN (2 , 0) AND (2 , 32)

X	Y	BEAM Y MOM
2	0	-8.031E-03
2	1	-1.370E 01
2	2	-2.726E 01
2	3	-4.075E 01
2	4	-5.414E 01
2	5	-6.736E 01
2	6	-7.917E 01
2	7	-9.044E 01
2	8	-1.009E 02
2	9	-1.102E 02
2	10	-1.171E 02
2	11	-1.134E 01
2	12	9.659E 01
2	13	2.030E 02
2	14	2.889E 02
2	15	3.575E 02
2	16	4.204E 02
2	17	3.922E 02
2	18	3.576E 02
2	19	2.910E 02
2	20	1.522E 02
2	21	1.019E 01
2	22	-1.289E 02
2	23	-1.235E 02
2	24	-1.142E 02
2	25	-1.032E 02
2	26	-9.093E 01
2	27	-7.779E 01
2	28	-6.264E 01
2	29	-4.720E 01
2	30	-3.159E 01
2	31	-1.588E 01
2	32	-8.550E-03

BEAM Y MOMENT BETWEEN (7 , 0) AND (7 , 32)

X	Y	BEAM Y MOM
7	0	-1.974E-04
7	1	-1.090E 01
7	2	-2.195E 01
7	3	-3.321E 01
7	4	-4.475E 01
7	5	-5.671E 01
7	6	-7.011E 01
7	7	-8.442E 01
7	8	-1.000E 02
7	9	-1.173E 02
7	10	-1.380E 02
7	11	-2.778E 01
7	12	8.417E 01
7	13	1.985E 02
7	14	2.859E 02
7	15	3.808E 02
7	16	4.802E 02
7	17	4.343E 02
7	18	3.944E 02
7	19	3.393E 02
7	20	1.679E 02
7	21	-1.687E 00
7	22	-1.666E 02
7	23	-1.396E 02
7	24	-1.180E 02
7	25	-9.877E 01
7	26	-8.148E 01
7	27	-6.551E 01
7	28	-5.160E 01
7	29	-3.825E 01
7	30	-2.528E 01
7	31	-1.255E 01
7	32	-4.334E-04

BEAM Y MOMENT BETWEEN (12 , 0) AND (12 , 32)

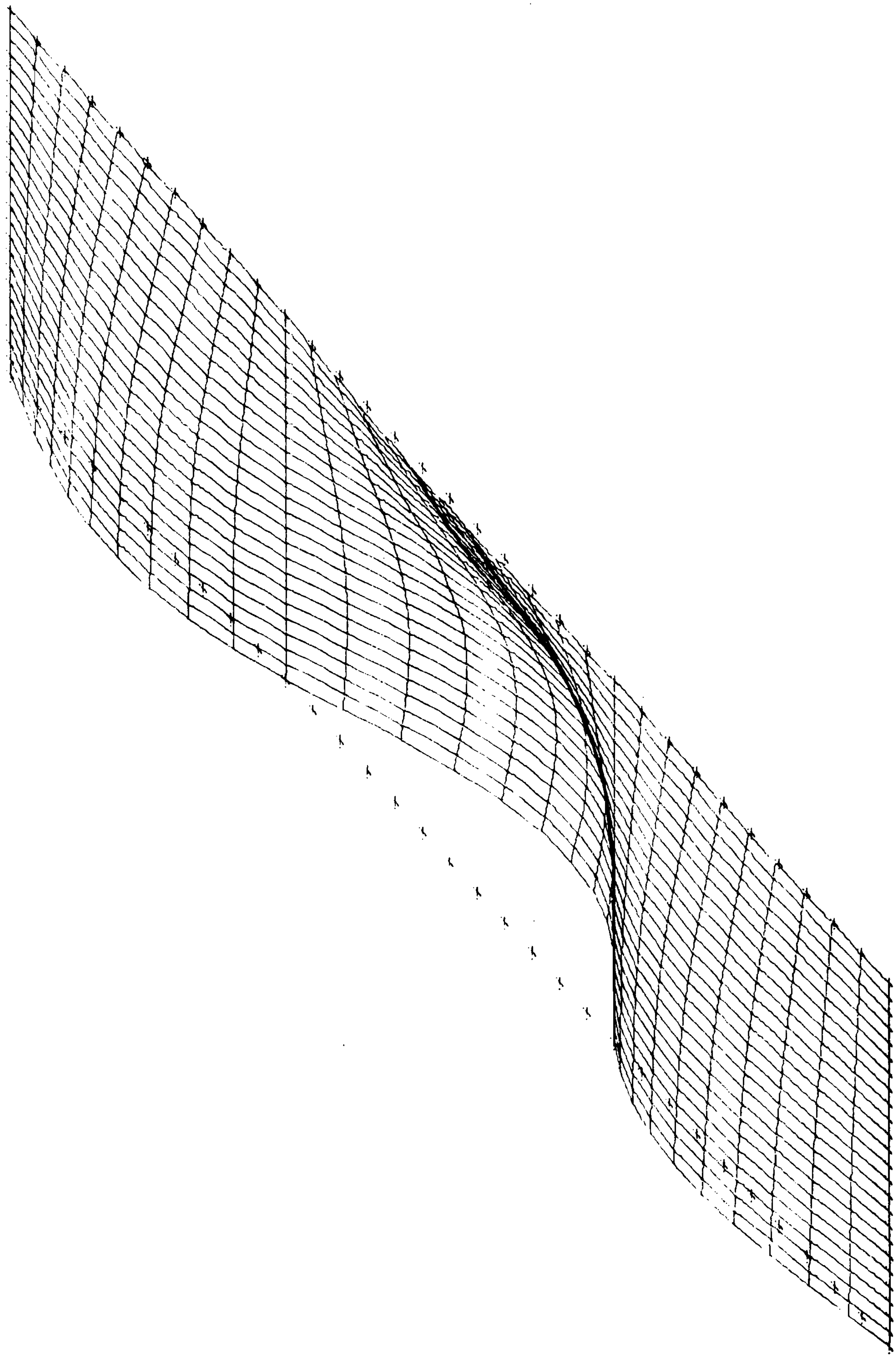
X	Y	BEAM Y MOM
12	0	-6.389E-04
12	1	-8.438E 00
12	2	-1.700E 01
12	3	-2.579E 01
12	4	-3.496E 01
12	5	-4.466E 01
12	6	-5.634E 01
12	7	-6.906E 01
12	8	-8.310E 01
12	9	-9.858E 01
12	10	-1.165E 02
12	11	-2.483E 01
12	12	6.884E 01
12	13	1.642E 02
12	14	2.383E 02
12	15	3.174E 02
12	16	3.977E 02
12	17	3.618E 02
12	18	3.282E 02
12	19	2.802E 02
12	20	1.3[illegible]

[illegible] 0) AND (22 , 32)

X	Y	BEAM Y MOM
[illegible]	0	1.500E-02
22	1	-3.371E 00
22	2	-6.778E 00
22	3	-9.792E 00
22	4	-1.218E 01
22	5	-1.374E 01
22	6	-1.371E 01
22	7	-1.229E 01
22	8	-9.142E 00
22	9	-3.918E 00
22	10	4.360E 00
22	11	8.168E 00
22	12	1.300E 01
22	13	1.703E 01
22	14	2.005E 01
22	15	2.116E 01
22	16	2.021E 01
22	17	1.945E 01
22	18	1.661E 01
22	19	1.364E 01
22	20	1.143E 01
22	21	6.740E 00
22	22	6.835E 00
22	23	-2.570E 00
22	24	-8.822E 00
22	25	-1.277E 01
22	26	-1.474E 01
22	27	-1.509E 01
22	28	-1.354E 01
22	29	-1.097E 01
22	30	-7.632E 00
22	31	-3.799E 00
22	32	1.809E-02

**** TIME SINCE BEGINNING OF EXECUTION IS 209.87 SECONDS ,

TIME SINCE LAST CALL OF PRTIME IS 209.87 SECONDS *****


1002A

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THD THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB
1003A LANE LOADING FOR MAX NEG MOMENT COMPOSITE DECK WHERE MOM IS POS

TABLE 1. CONTROL DATA

	TABLE NUMBER 2	3	4	5	6	7	8	9
KEEP FROM PRECEDING PROBLEM (1=YES)	0	0	0	0	0	0	0	0
NUM CARDS INPUT THIS PROBLEM	1	60	0	1	0	13	5	6

MULTIPLE LOAD OPTION	0
STATICS CHECK OPTION	0
PRIN STRESS OPTION	0
PROFILE PLOT OPTION	0
3-D PLOT OPTION	1

TABLE 2. CONSTANTS

NUMBER OF INCREMENTS IN X DIRECTION	24
NUMBER OF INCREMENTS IN Y DIRECTION	32
INCREMENT LENGTH IN X DIRECTION	1.450E 00
INCREMENT LENGTH IN Y DIRECTION	5.000E 00
POISSONS RATIO	1.500E-01
SLAB THICKNESS	0.0

TABLE 3. JOINT BENDING STIFFNESSES, LOADS, AND SUPPORTS

Slab bending stiffness (DX, DY); Beam stiffness with composite zones (FY); Diafram stiffness (FX); Supports (S)

FROM JOINT		THRU JOINT		DX	DY	FX	FY	Q	S
0	0	24	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
0	1	24	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
1	0	23	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
1	1	23	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
2	0	2	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	1	2	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	1	2	7	0.0	0.0	0.0	1.829E 06	0.0	0.0
2	12	2	20	0.0	0.0	0.0	1.829E 06	0.0	0.0
7	0	7	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	1	7	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	1	7	7	0.0	0.0	0.0	1.881E 06	0.0	0.0
7	12	7	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
12	0	12	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
12	1	12	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
12	1	12	7	0.0	0.0	0.0	1.881E 06	0.0	0.0
12	12	12	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
17	0	17	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
17	1	17	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
17	1	17	7	0.0	0.0	0.0	1.881E 06	0.0	0.0
17	12	17	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
22	0	22	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	1	22	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	1	22	7	0.0	0.0	0.0	1.829E 06	0.0	0.0
22	12	22	20	0.0	0.0	0.0	1.829E 06	0.0	0.0
2	0	22	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	0	21	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	5	22	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	5	21	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	9	22	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	9	21	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	14	22	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	14	21	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	18	22	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	18	21	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	23	22	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	23	21	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	27	22	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	27	21	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	32	22	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	32	21	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	0	2	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	0	7	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	0	12	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	0	17	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	0	22	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	10	2	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	10	7	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	10	12	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	10	17	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	10	22	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	22	2	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	22	7	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	22	12	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	22	17	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	22	22	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	32	2	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	32	7	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	32	12	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	32	17	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	32	22	32	0.0	0.0	0.0	0.0	0.0	1.000E 14

TABLE 4. JOINT RESTRAINTS AND APPLIED MOMENTS

FROM JOINT	THRU JOINT	RX	RY	TX	TY

NONE

TABLE 5. MESH TWISTING STIFFNESSES

Slab twisting stiffness

FROM MESH		THRU MESH		C
1	1	24	32	6.210E 03

TABLE 6. BAR AXIAL THRUSTS

FROM BAR	THRU BAR	PX	PY	PBX	PBY

NONE

TABLE 7. MULTIPLE LOADS

FROM JOINT		THRU JOINT		QM
1	0	1	22	-5.600E-02
1	1	1	21	-5.600E-02
15	0	15	22	-1.790E-01
15	1	15	21	-1.790E-01
2	0	14	0	-2.960E-01
2	22	14	22	-2.960E-01
2	1	14	21	-5.930E-01
1	5	1	5	-6.330E-01
15	5	15	5	-2.013E 00
2	5	14	5	-3.335E 00
1	16	1	16	-6.330E-01
15	16	15	16	-2.013E 00
2	16	14	16	-3.335E 00

Lane loading apportioned as shown in Fig 16

TABLE 8. PROFILE OUTPUT AREAS

FROM JOINT		THRU JOINT		DEFL (1=YES)	X MOMENT (1=SLAB,2=BEAM)	Y MOMENT	PRIN MOM OR STRESS (1=YES)
2	0	2	32	1	0	2	0
7	0	7	32	1	0	2	0
12	0	12	32	1	0	2	0
17	0	17	32	1	0	2	0
22	0	22	32	1	0	2	0

TABLE 9. PRINTED OUTPUT LIMITS

FROM Y STA	THRU Y STA
0	0
4	6
9	11
14	18
22	22
32	32

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THD THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB (CONTD)
1003A LANE LOADING FOR MAX NEG MOMENT COMPOSITE DECK WHERE MOM IS POS

RESULTS

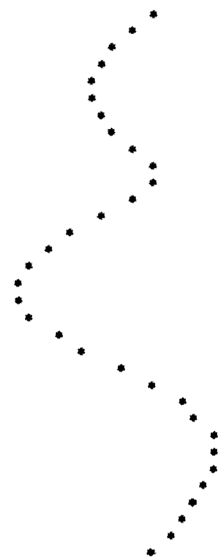
SLAB X MOMENT AND X TWISTING MOMENT ACT IN THE X DIRECTION (ABOUT Y AXIS)
Y TWISTING MOMENT = -X TWISTING MOMENT, COUNTERCLOCKWISE BETA ANGLES ARE
POSITIVE FROM THE X AXIS TO THE DIRECTION OF THE LARGEST PRINCIPAL MOMENT
SLAB MOMENTS ARE PER UNIT WIDTH

BEAM MOMENTS ARE TOTAL PER BEAM

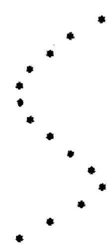
X	Y	DEFL	SLAB X MOMENT / BEAM X MOMENT	SLAB Y MOMENT / BEAM Y MOMENT	SLAB X TWISTING MOMENT	LARGEST PRINCIPAL SLAB MOMENT	BETA X TO LARGEST MOMENT	SUPPORT REACTION
0	32	4.349D-05	6.563E-16	1.311E-17	6.769D-03	6.769E-03	45.0	0.0
			0.0	0.0				
1	32	1.594D-05	1.975E-02	-7.236E-10	1.708D-02	2.960E-02	30.0	0.0
			0.0	0.0				
2	32	1.872D-14	1.945E-02	2.853E-03	2.202D-02	3.470E-02	34.7	-1.872D 00
			1.415E-01	-4.194E-03				
3	32	-4.768D-06	6.968E-03	-2.455E-10	2.263D-02	2.638E-02	40.6	0.0
			1.037E-01	0.0				
4	32	-5.442D-06	3.338E-08	-9.855E-11	2.508D-02	2.681E-02	43.1	0.0
			4.969E-02	0.0				
5	32	-4.154D-06	1.126E-03	-5.308E-11	2.646D-02	2.703E-0[illegible]		
			1.676E-02	0.0				
6	32	-2.204D-06	4.329E-04	-1.696E-11				
			6.444E-03					
7	32	1.413D-14						
						[illegible]E-01	39.3	0.0
				[illegible]-02	2.666D-01	3.751E-01	38.0	0.0
				1.425E 00				
			[illegible].203E-02	6.911E-02	2.468D-01	3.225E-01	44.3	0.0
			0.0	0.0				
24	11	5.179D-04	-9.362E-10	5.188E-02	1.212D-01	1.499E-01	51.0	0.0
			0.0	0.0				
0	10	5.302D-04	9.445E-09	-5.133E-01	7.248D-03	-5.134E-01	-89.2	0.0
			0.0	0.0				
1	10	2.404D-04	5.997E-03	-1.057E 00	1.204D-02	-1.057E 00	-89.4	0.0
			0.0	0.0				
2	10	-3.940D-13	-9.283E-02	-1.110E 00	8.909D-03	-1.110E 00	-89.5	3.940D 01
			0.0	-1.880E 02				
3	10	-2.187D-04	2.261E-01	-1.092E 00	1.813D-02	-1.093E 00	-89.2	0.0
			0.0	0.0				
4	10	-3.228D-04	3.145E-01	-1.126E 00	3.943D-02	-1.127E 00	-88.4	0.0
			0.0	0.0				
5	10	-2.849D-04	1.781E-01	-1.214E 00	5.917D-02	-1.216E 00	-87.6	0.0
			0.0	0.0				
6	10	-1.412D-04	-2.110E-01	-1.350E 00	6.555D-02	-1.353E 00	-86.7	0.0
			0.0	0.0				
7	10	-5.546D-13	-9.214E-01	-1.514E 00	4.493D-02	-1.518E 00	-85.7	5.546D 01
			0.0	-2.359E 02				
8	10	-6.279D-05	-2.710E-01	-1.393E 00	2.309D-02	-1.394E 00	-88.8	0.0
			0.0	0.0				
9	10	-1.438D-04	4.685E-02	-1.299E 00	2.542D-02	-1.300E 00	-88.9	0.0
			0.0	0.0				

```
10  10 -1.5380-04   6.670E-02 -1.248E 00   3.7020-02 -1.249E 00 -88.4   0.0
                    0.0        0.0
11  10 -8.3250-05 -1.429E-01 -1.242E 00   4.3850-02 -1.243E 00 -87.7   0.0
                    0.0        0.0
12  10 -4.6340-13 -6.774E-01 -1.266E 00   3.1520-02 -1.267E 00 -86.9   4.6340 01
                    0.0       -1.996E 02
13  10 -6.0000-05 -1.573E-01 -1.069E 00   1.9380-02 -1.069E 00 -88.8   0.0
                    0.0        0.0
14  10 -1.1910-04   7.313E-02 -9.004E-01   2.6960-02 -9.011E-01 -88.4   0.0
                    0.0        0.0
15  10 -1.1710-04   6.805E-02 -7.735E-01   4.0510-02 -7.755E-01 -87.3   0.0
                    0.0        0.0
16  10 -6.0920-05 -8.554E-02 -6.788E-01   4.9690-02 -6.829E-01 -85.2   0.0
                    0.0        0.0
17  10 -2.0390-13 -2.887E-01 -5.943E-01   5.0680-02 -6.025E-01 -80.8   2.0390 01
                    0.0       -9.449E 01
18  10  2.2770-06 -1.500E-01 -4.467E-01   4.8910-02 -4.546E-01 -80.9   0.0
                    0.0        0.0
19  10 -1.9820-05 -4.620E-02 -3.050E-01   4.9650-02 -3.142E-01 -79.5   0.0
                    0.0        0.0
20  10 -4.2050-05   4.579E-02 -1.717E-01   5.2800-02 -1.838E-01 -77.0   0.0
                    0.0        0.0
21  10 -4.3260-05   1.442E-01 -4.742E-02   5.8640-02  1.608E-01  15.7   0.0
                    0.0        0.0
22  10  2.8970-14   2.678E-01  6.585E-02   6.5880-02  2.874E-01  16.6  -2.8970 00
                    0.0        4.403E 00
23  10  1.1900-04   1.253E-01  1.183E-01   6.8510-02  1.904E-01  43.5   0.0
                    0.0        0.0
24  10  2.6970-04 -4.682E-09  8.427E-02   3.3780-02  9.614E-02  70.6   0.0
                    0.0        0.0

 0   9 -7.0630-05   9.574E-09 -2.393E-01   5.6700-02 -2.520E-01
                    0.0        0.0
 1   9 -4.3300-04   7.663E-02 -5.072E-01   1.3160-01
                    0.0        0.0
 2   9 -7.5040-04   3.789E-02 -5.631E-
                    5.568E-01
 3   9 -1.0320-03   1.2
                                                                 38.4   0.0
 4   9
                                                     1.213E-01  44.2   0.0

                              E-03   1.4740-01 -1.858E-01 -39.9   1.4810 01
                               1.349E-02
                       1E-05  3.856E-12   1.7270-01 -1.727E-01 -45.0   0.0
                   -1.108E-03  0.0
      3.0670-05   2.824E-02 -1.132E-09   1.9880-01  2.134E-01  43.0   0.0
                   4.203E-01  0.0
15   0 -2.9440-05   2.364E-02 -1.041E-09   2.1360-01  2.258E-01  43.4   0.0
                   3.519E-01  0.0
16   0 -1.4310-05 -1.397E-03  4.029E-11   2.2100-01 -2.217E-01 -44.9   0.0
                  -2.080E-02  0.0
17   0 -6.1980-14 -2.826E-02 -4.203E-03   2.1210-01 -2.287E-01 -43.4   6.1980 00
                  -4.113E-01  6.095E-03
18   0 -1.9290-06 -1.576E-02  5.292E-10   1.9730-01 -2.054E-01 -43.9   0.0
                  -2.347E-01  0.0
19   0 -1.3120-05 -2.312E-05  6.935E-13   1.8750-01 -1.875E-01 -45.0   0.0
                  -3.441E-04  0.0
20   0 -2.4330-05   1.906E-02 -7.744E-10   1.7580-01  1.856E-01  43.4   0.0
                   2.837E-01  0.0
21   0 -2.4340-05   4.142E-02 -1.511E-09   1.5690-01  1.790E-01  41.2   0.0
                   6.167E-01  0.0
22   0  1.3450-14   1.090E-01  1.622E-02   1.4940-01  2.191E-01  36.4  -1.3450 00
                   7.936E-01 -2.352E-02
23   0  8.6990-05   1.182E-01 -1.027E-08   1.1540-01  1.888E-01  31.4   0.0
                   0.0        0.0
24   0  2.4350-04   9.437E-17  1.416E-17   4.4390-02  4.439E-02  45.0   0.0
                   0.0        0.0


STATICS CHECK.          SUMMATION OF REACTIONS =     2.7190 02

                        MAXIMUM STATICS CHECK ERROR AT STA  12 10 = -7.6480-05
```

```
PROGRAM SLAB 49  -TEX HWY DEPT DECK-  MATLOCK,PANAK          REV DATE 29 FEB 72

I-BEAM STRUCTURE          SPANS 50-60-50, 33 FT RDWY, HS20 LDG
ANALYSIS OF THE THREE-SPAN CONTINUOUS        RUN 29 FEB  1972  JJP   FT-KIP UNITS


PROB (CONTD)
1003A     LANE LOADING FOR MAX NEG MOMENT    COMPOSITE DECK WHERE MOM IS POS


PROFILE OUTPUT AREAS
     X MOMENTS ACT IN X DIRECTION (ABOUT Y AXIS)
     THE PLOTTED RESULTS INDICATE THE RELATIVE VALUE EACH HAS WITHIN THAT LIST


             DEFLECTIONS      BETWEEN (  2 ,  0 ) AND (  2 , 32 )

 X , Y DEFLECTION

 2   0 -1.162E-13
 2   1 -3.723E-03
 2   2 -6.984E-03
 2   3 -9.398E-03
 2   4 -1.067E-02
 2   5 -1.059E-02
 2   6 -9.065E-03
 2   7 -6.523E-03
 2   8 -3.488E-03
 2   9 -7.504E-04
 2  10 -3.940E-13
 2  11 -3.079E-03
 2  12 -8.063E-03
 2  13 -1.310E-02
 2  14 -1.752E-02
 2  15 -2.075E-02
 2  16 -2.237E-02
 2  17 -2.202E-02
 2  18 -1.986E-02
 2  19 -1.614E-02
 2  20 -1.128E-02
 2  21 -5.753E-03
 2  22 -1.686E-13
 2  23  4.414E-03
 2  24  7.529E-03
 2  25  9.435E-03
 2  26  1.024E-02
 2  27  1.007E-02
 2  28  9.073E-03
 2  29  7.399E-03
 2  30  5.216E-03
 2  31  2.694E-03
 2  32  1.872E-14
```



```
             DEFLECTIONS      BETWEEN (  7 ,  0 ) AND (  7 , 32 )

 X , Y DEFLECTION

 7   0 -1.775E-13
 7   1 -4.901E-03
 7   2 -9.208E-03
 7   3 -1.244E-02
 7   4 -1.423E-02
 7   5 -1.429E-02
 7   6 -1.242E-02
 7   7 -9.202E-03
 7   8 -5.279E-03
 7   9 -1.494E-03
 7  10 -5.546E-13
 7  11 -3.313E-03
 7  12 -8.936E-03
 7  13 -1.463E-02
```



X	Y	VALUE
7	14	-1.966E-02
7	15	-2.339E-02
7	16	-2.523E-02
7	17	-2.472E-02
7	18	-2.215E-02
7	19	-1.790E-02
7	20	-1.243E-02
7	21	-6.261E-03
7	22	-2.482E-13
7	23	4.435E-03
7	24	7.352E-03
7	25	8.988E-03
7	26	9.554E-03
7	27	9.240E-03
7	28	8.219E-03
7	29	6.643E-03
7	30	4.655E-03
7	31	2.395E-03
7	32	[illegible]

BEAM Y MOMENT BETWEEN (2 , 0) AND (2 , 32)

X	Y	BEAM Y MOM
2	0	-1.813E-02
2	1	5.647E 01
2	2	1.036E 02
2	3	1.400E 02
2	4	1.646E 02
2	5	1.767E 02
2	6	1.245E 02
2	7	6.022E 01
2	8	-1.456E 01
2	9	-9.754E 01
2	10	-1.880E 02
2	11	-9.347E 01
2	12	-5.912E 00
2	13	7.463E 01
2	14	1.448E 02
2	15	1.986E 02
2	16	2.400E 02
2	17	2.212E 02
2	18	1.897E 02
2	19	1.409E 02
2	20	8.112E 01
2	21	1.104E 01
2	22	-6.571E 01
2	23	-6.374E 01
2	24	-5.938E 01
2	25	-5.395E 01
2	26	-4.772E 01
2	27	-4.098E 01
2	28	-3.305E 01
2	29	-2.493E 01
2	30	-1.670E 01
2	31	-8.392E 00
2	32	-4.194E-03

BEAM Y MOMENT BETWEEN (7 , 0) AND (7 , 32)

X	Y	BEAM Y MOM
7	0	2.312E-02
7	1	7.382E 01
7	2	1.333E 02
7	3	1.800E 02
7	4	2.151E 02
7	5	2.384E 02
7	6	1.691E 02
7	7	8.734E 01
7	8	-6.855E 00
7	9	-1.124E 02
7	10	-2.359E 02
7	11	-1.134E 02
7	12	-9.104E 00
7	13	8.290E 01
7	14	1.621E 02
7	15	2.340E 02
7	16	2.924E 02
7	17	2.568E 02
7	18	2.080E 02
7	19	1.529E 02
7	20	8.575E 01
7	21	4.630E 00
7	22	-8.961E 01
7	23	-7.453E 01
7	24	-6.285E 01
7	25	-5.250E 01
7	26	-4.322E 01
7	27	-3.466E 01
7	28	-2.728E 01
7	29	-2.020E 01
7	30	-1.334E 01
7	31	-6.624E 00
7	32	-3.200E-04

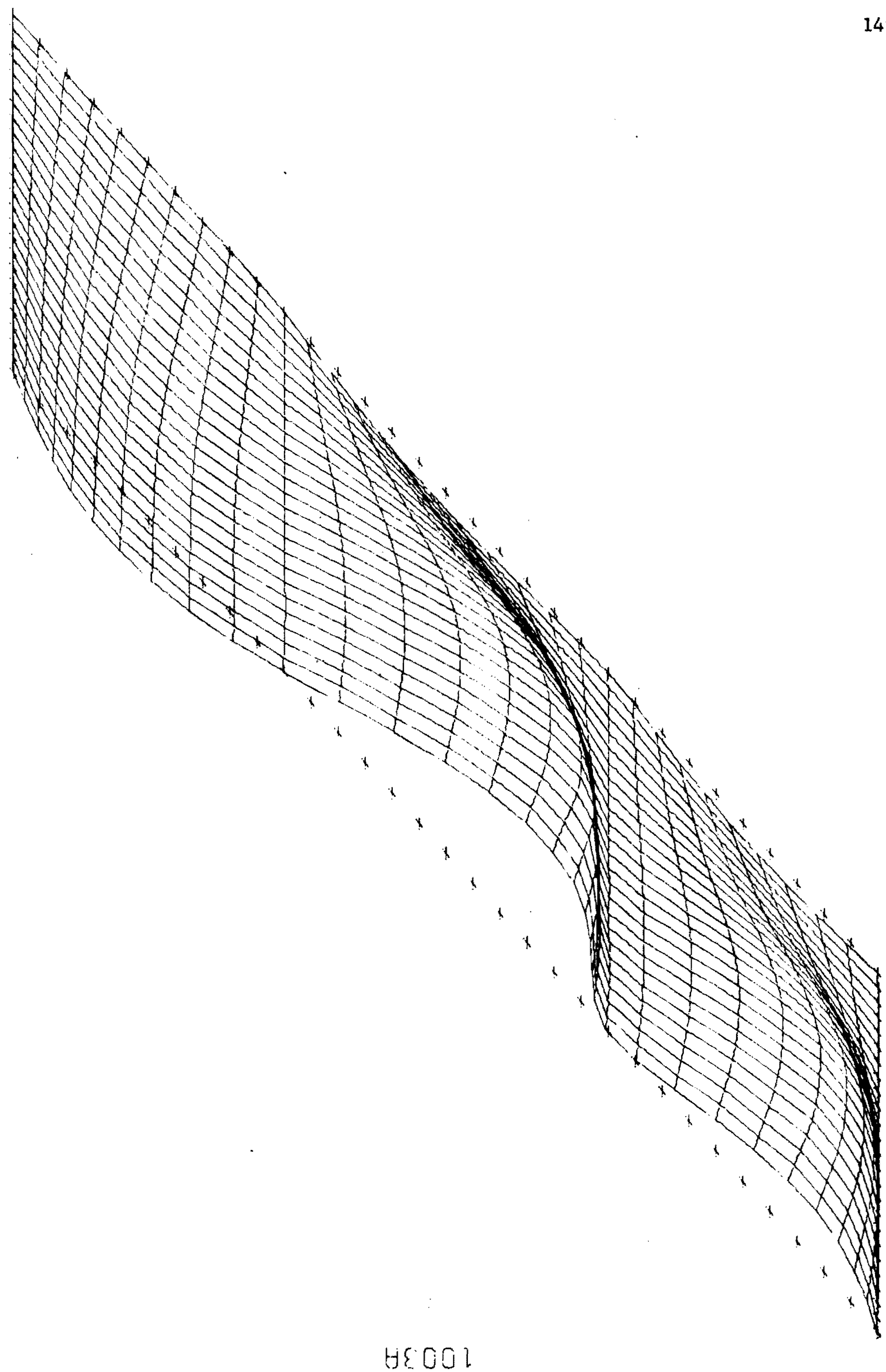
BEAM Y MOMENT BETWEEN (12 , 0) AND (12 , 32)

X	Y	BEAM Y MOM
12	0	1.349E-02
12	1	6.318E 01
12	2	1.142E 02
12	3	1.542E 02
12	4	1.844E 02
12	5	2.049E 02
12	6	1.453E 02
12	7	7.551E 01
12	8	-5.059E 00
12	9	-9.556E 01
12	10	-1.996E 02
12	11	-9.691E 01
12	12	-8.929E 00
12	13	6.878E 01
12	14	1.354E 02
12	15	1.975E 02
12	16	2.493E 02
12	17	2.167E 02
12	18	1.739E 02
12	19	1.279E 02

X	Y	VALUE
[illegible]	[illegible]	[illegible] E 00
[illegible]	25	-5.765E 00
22	26	-7.099E 00
22	27	-7.524E 00
22	28	-6.850E 00
22	29	-5.599E 00
22	30	-3.912E 00
22	31	-1.948E 00
22	32	9.966E-03

***** TIME SINCE BEGINNING OF EXECUTION IS 351.27 SECONDS ,

TIME SINCE LAST CALL OF PRTIME IS 141.40 SECONDS *****



1003A

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LOG
ANALYSIS OF TWO THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB
1004A 2 HS20 TRUCKS STAGGERED IN ADJACENT SPANS FOR MAX NEG SUPPORT MOMENT

TABLE 1. CONTROL DATA

	TABLE NUMBER 2	3	4	5	6	7	8	9
KEEP FROM PRECEDING PROBLEM (1=YES)	0	0	0	0	0	0	0	0
NUM CARDS INPUT THIS PROBLEM	1	59	0	1	0	40	6	6

MULTIPLE LOAD OPTION	0
STATICS CHECK OPTION	0
PRIN STRESS OPTION	0
PROFILE PLOT OPTION	0
3-D PLOT OPTION	1

TABLE 2. CONSTANTS

NUMBER OF INCREMENTS IN X DIRECTION	24
NUMBER OF INCREMENTS IN Y DIRECTION	32
INCREMENT LENGTH IN X DIRECTION	1.450E 00
INCREMENT LENGTH IN Y DIRECTION	5.000E 00
POISSONS RATIO	1.500E-01
SLAB THICKNESS	0.0

TABLE 3. JOINT BENDING STIFFNESSES, LOADS, AND SUPPORTS

(Annotations: DX, DY of first rows — Slab bending stiffness; FY values 6.135E 05 to 1.829E 06 — Beam stiffness with composite zones; FX values 2.663E 04 / 3.256E 04 — Diafram stiffness; S values 1.000E 14 — Supports)

FROM JOINT		THRU JOINT		DX	DY	FX	FY	Q	S
0	0	24	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
0	1	24	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
0	1	23	32	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
1	1	23	31	1.830E 03	1.830E 03	0.0	0.0	0.0	0.0
2	0	2	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	1	2	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
2	12	2	20	0.0	0.0	0.0	1.829E 06	0.0	0.0
7	0	7	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	1	7	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
7	1	7	7	0.0	0.0	0.0	1.881E 06	0.0	0.0
7	12	7	20	0.0	0.0	0.0	1.881E 06	0.0	0.0
12	0	12	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
12	1	12	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
12	1	12	8	0.0	0.0	0.0	1.881E 06	0.0	0.0
12	13	12	21	0.0	0.0	0.0	1.881E 06	0.0	0.0
17	0	17	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
17	1	17	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
17	1	17	9	0.0	0.0	0.0	1.881E 06	0.0	0.0
17	16	17	24	0.0	0.0	0.0	1.881E 06	0.0	0.0
22	0	22	32	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	1	22	31	0.0	0.0	0.0	6.135E 05	0.0	0.0
22	1	22	11	0.0	0.0	0.0	1.829E 06	0.0	0.0
22	20	22	31	0.0	0.0	0.0	1.829E 06	0.0	0.0
2	0	22	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	0	21	0	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	5	22	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	5	21	5	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	9	22	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	9	21	9	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	14	22	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	14	21	14	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	18	22	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	18	21	18	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	23	22	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	23	21	23	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	27	22	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
3	27	21	27	0.0	0.0	3.256E 04	0.0	0.0	0.0
2	32	22	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
3	32	21	32	0.0	0.0	2.663E 04	0.0	0.0	0.0
2	0	2	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	0	7	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	0	12	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	0	17	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	0	22	0	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	10	2	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	10	7	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	10	12	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	10	17	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	10	22	10	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	22	2	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	22	7	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	22	12	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	22	17	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	22	22	22	0.0	0.0	0.0	0.0	0.0	1.000E 14
2	32	2	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
7	32	7	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
12	32	12	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
17	32	17	32	0.0	0.0	0.0	0.0	0.0	1.000E 14
22	32	22	32	0.0	0.0	0.0	0.0	0.0	1.000E 14

TABLE 4. JOINT RESTRAINTS AND APPLIED MOMENTS

FROM JOINT	THRU JOINT	RX	RY	TX	TY

NONE

TABLE 5. MESH TWISTING STIFFNESSES

(Annotation: C — Slab twisting stiffness)

FROM MESH		THRU MESH		C
1	1	24	32	6.210E 03

TABLE 6. BAR AXIAL THRUSTS

FROM BAR	THRU BAR	PX	PY	PBX	PBY

NONE

TABLE 7. MULTIPLE LOADS

FROM JOINT		THRU JOINT		QM
9	2	9	2	-1.700E 00
10	2	10	2	-2.400E 00
13	2	13	2	-1.100E 00
14	2	14	2	-3.000E 00
9	3	9	3	-4.000E-01
10	3	10	3	-6.000E-01
13	3	13	3	-3.000E-01
14	3	14	3	-7.000E-01
9	5	9	5	-8.500E 00
10	5	10	5	-1.200E 01
13	5	13	5	-5.600E 00
14	5	14	5	-1.480E 01
9	7	9	7	-1.700E 00
10	7	10	7	-2.400E 00
13	7	13	7	-1.100E 00

14	7	14	7	-3.000E 00
9	8	9	8	-6.800E 00
10	8	10	8	-9.600E 00
13	6	13	8	-4.500E 00
14	8	14	8	-1.180E 01
2	13	2	13	-5.100E 00
3	13	3	13	-1.130E 01
6	13	6	13	-2.800E 00
7	13	7	13	-1.350E 01
2	14	2	14	-1.300E 00
3	14	3	14	-2.800E 00
6	14	6	14	-7.000E-01
7	14	7	14	-3.400E 00
2	16	2	16	-6.400E 00
3	16	3	16	-1.410E 01
6	16	6	16	-3.500E 00
7	16	7	16	-1.690E 01
2	18	2	18	-3.000E-01
3	18	3	18	-7.000E-01
6	18	6	18	-2.000E-01
7	18	7	18	-8.000E-01
2	19	2	19	-1.300E 00
3	19	3	19	-2.800E 00
6	19	6	19	-7.000E-01
7	19	7	19	-3.400E 00

Two HS-20 trucks apportioned as shown in Fig 17

TABLE 8. PROFILE OUTPUT AREAS

FROM JOINT		THRU JOINT		DEFL (1=YES)	X MOMENT (1=SLAB,2=BEAM)	Y MOMENT	PRIN MOM OR STRESS (1=YES)
2	0	2	32	1	0	2	0
7	0	7	32	1	0	2	0
12	0	12	32	1	0	2	0
17	0	17	32	1	0	2	0
22	0	22	32	1	0	2	0
0	5	24	5	1	2	0	0

TABLE 9. PRINTED OUTPUT LIMITS

FROM Y STA	THRU Y STA
0	0
5	5
8	11
14	18
21	23
32	32

PROGRAM SLAB 49 -TEX HWY DEPT DECK- MATLOCK,PANAK REV DATE 29 FEB 72

I-BEAM STRUCTURE SPANS 50-60-50, 33 FT RDWY, HS20 LOG
ANALYSIS OF THD THREE-SPAN CONTINUOUS RUN 29 FEB 1972 JJP FT-KIP UNITS

PROB (CONTD)
1004A 2 HS20 TRUCKS STAGGERED IN ADJACENT SPANS FOR MAX NEG SUPPORT MOMENT

RESULTS

SLAB X MOMENT AND X TWISTING MOMENT ACT IN THE X DIRECTION (ABOUT Y AXIS)
Y TWISTING MOMENT = -X TWISTING MOMENT, COUNTERCLOCKWISE BETA ANGLES ARE
POSITIVE FROM THE X AXIS TO THE DIRECTION OF THE LARGEST PRINCIPAL MOMENT
SLAB MOMENTS ARE PER UNIT WIDTH

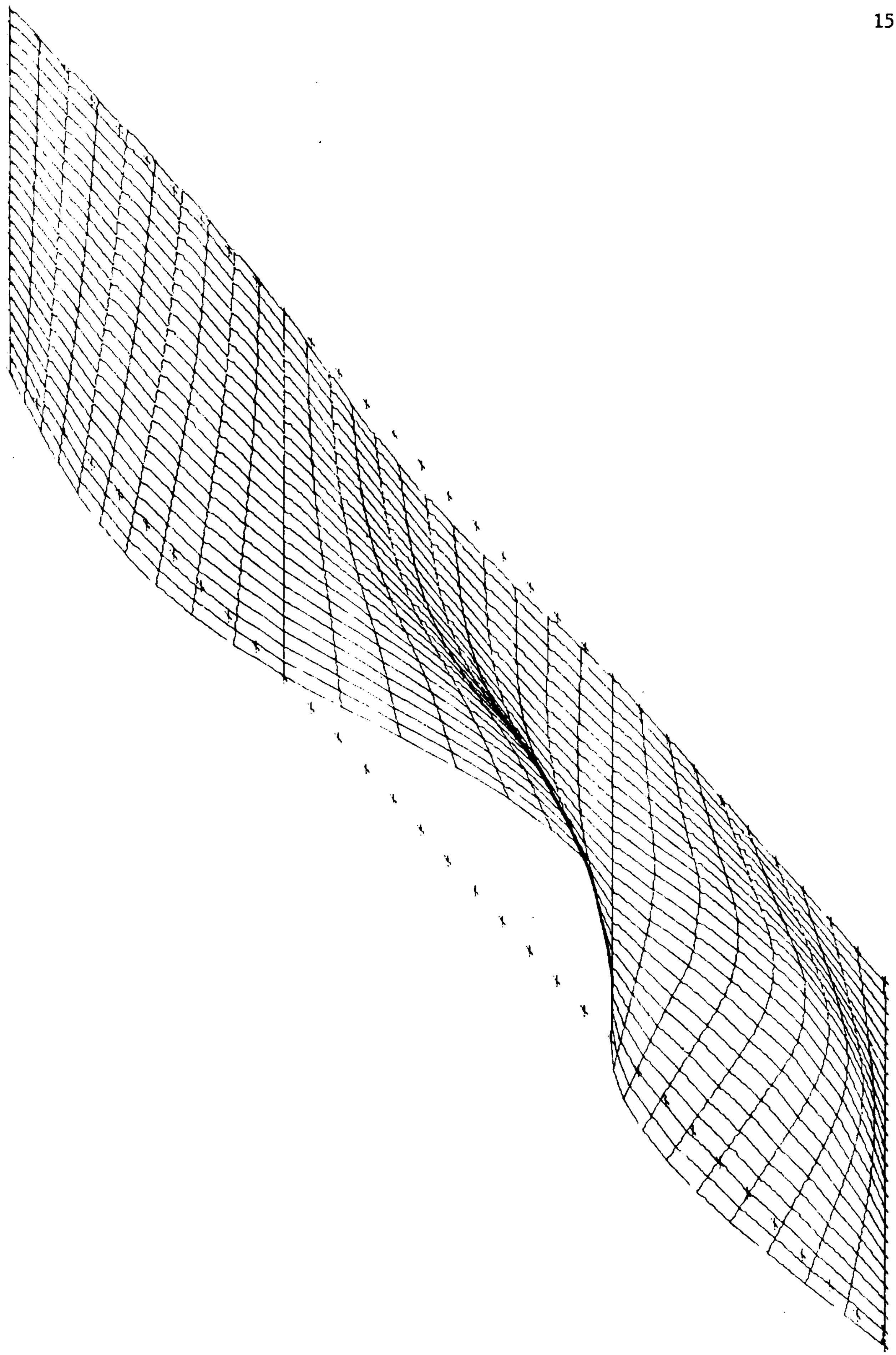
BEAM MOMENTS ARE TOTAL PER BEAM

X	Y	DEFL	SLAB X MOMENT / BEAM X MOMENT	SLAB Y MOMENT / BEAM Y MOMENT	SLAB X TWISTING MOMENT	LARGEST PRINCIPAL SLAB MOMENT	BETA X TO LARGEST MOMENT	SUPPORT REACTION
0	32	1.537D-04	1.424E-15	2.679E-16	2.473D-02	2.473E-02	45.0	0.0
			0.0	0.0				
1	32	5.686D-05	6.810E-02	-5.248E-09	6.363D-02	1.062E-01	30.9	0.0
			C.0	0.0				
2	32	2.183D-14	6.345E-C2	9.443E-03	8.094D-02	1.218E-01	35.8	-2.183D 00
			4.620E-01	-1.369E-02				
3	32	-2.038D-C5	2.7G1E-02	-1.136E-09	8.192D-02	9.653E-02	40.3	0.0
			4.021E-01	0.0				
4	32	-2.490D-05	1.672E-C2	-6.498E-10	8.845D-02	9.721E-02	42.3	0.0
			2.489E-C1	0.0				
5	32	-1.958D-C5	7.768E-C3	-3.874E-10	8.922D-02	9.319E-02	43.8	0.0
			1.156E-01	0.0				
6	32	-9.706D-06	-2.913E-04	2.662E-11	8.793D-02	-8.808E-02	-45.0	0.0
			-4.336E-03	0.0				
7	32	9.635D-15	-7.996E-03	-1.189E-03	8.628D-02	-9.094E-02	-43.9	-9.635D-01
			-1.164E-01	1.725E-03				
8	32	5.112D-06	-6.087E-03	4.17CE-10	8.409D-02	-8.719E-02	-44.0	0.0
			-9.062E-02	0.0				
9	32	6.646D-06	-4.296E-03	4.349E-10	8.113D-02	-8.331E-02	-44.2	0.0
			-6.395E-02	0.0				
10	32	5.656D-C6	-2.588E-03	8.335E-11	7.894D-02	-8.025E-02	[illegible]	
			-3.853E-02	0.0				
11	32	3.145D-06	-1.079E-C3	3.399E-11	7.660D-[illegible]			
			-1.606E-02	0.0				
12	32	6.442D-15	1.010E-[illegible]					
13	32						[illegible]	0.0
					[illegible]-01	-1.214E 00	-51.0	0.0
				-6.141E-01	7.346D-01	-1.241E 00	-49.5	0.0
			0.0	0.0				
		[illegible].636D-05	-5.585E-01	-6.441E-01	6.484D-01	-1.251E 00	-46.9	0.0
			0.0	0.0				
12	11	6.169D-04	-7.67CE-01	-6.611E-01	5.474D-01	-1.264E 00	-42.2	0.0
			0.0	-9.365E 01				
13	11	1.134D-03	-6.302E-01	-6.140E-01	4.401D-01	-1.062E 00	-44.5	0.0
			0.0	0.0				
14	11	1.493D-C3	-5.222E-C1	-5.552E-01	3.340D-01	-8.731E-01	-46.4	0.0
			0.0	0.0				
15	11	1.723D-03	-4.411E-01	-4.861E-01	2.369D-01	-7.016E-01	-47.7	0.0
			0.0	0.0				
16	11	1.844D-03	-3.794E-01	-4.101E-01	1.536D-01	-5.491E-01	-47.9	0.0
			0.0	0.0				
17	11	1.873D-C3	-3.251E-01	-3.314E-01	8.725D-02	-4.156E-01	-46.0	0.0
			0.0	-4.847E 01				

```
18  11  1.820D-03 -2.339E-01 -2.523E-01  3.570D-02 -2.800E-01 -52.2  0.0
                   0.0        0.0
19  11  1.710D-03 -1.382E-01 -1.750E-01 -4.144D-03 -1.754E-01  83.6  0.0
                   0.0        0.0
20  11  1.567D-03 -3.879E-02 -9.934E-02 -3.253D-02 -1.135E-01  66.5  0.0
                   0.0        0.0
21  11  1.417D-03  6.563E-02 -2.579E-02 -5.062D-02  8.812E-02 -24.0  0.0
                   0.0        0.0
22  11  1.287D-03  1.770E-01  4.515E-02 -6.029D-02  2.004E-01 -21.2  0.0
                   0.0        7.943E 00
23  11  1.207D-03  8.554E-02  8.162E-02 -6.385D-02  1.475E-01 -44.1  0.0
                   0.0        0.0
24  11  1.149D-03 -2.110E-09  5.852E-02 -3.165D-02  7.236E-02 -66.4  0.0
                   0.0        0.0

 0  10  2.912D-04  1.290E-08 -5.384E-01  3.453D-01 -7.071E-01 -64.0  0.0
                   0.0        0.0
 1  10  1.246D-04  2.923E-02 -7.577E-01  6.889D-01 -1.158E 00 -59.9  0.0
                   0.0        0.0
 2  10 -2.966D-13  2.919E-03 -8.127E-01  7.142D-01 -1.227E 00 -59.9  2.966D 01
                   0.0       -1.394E 02
 3  10 -8.791D-05  5.507E-02 -8.488E-01  7.530D-01 -1.275E 00 -60.5  0.0
                   0.0        0.0
 4  10 -1.222D-04  4.509E-02 -9.007E-01  7.890D-01 -1.348E 00 -60.5  0.0
                   0.0        0.0
 5  10 -1.036D-04 -2.529E-02 -9.711E-01  8.218D-01 -1.446E 00 -60.0  0.0
                   0.0        0.0
 6  10 -4.961D-05 -1.735E-01 -1.057E 00  8.486D-01 -1.572E 00 -58.8  0.0
                   0.0        0.0
 7  10 -4.259D-13 -4.409E-01 -1.149E 00  8.654D-01 -1.730E 00 -56.1  4.259D 01
                   0.0       -1.856E 02
 8  10 -2.931D-05 -1.825E-01 -1.110E 00  8.518D-01 -1.616E 00 -59.3  0.0
                   0.0        0.0
 9  10 -6.331D-05 -5.804E-02 -1.070E 00  7.910D-01 -1.503E 00 -61.3  0.0
                   0.0        0.0
10  10 -6.723D-05 -4.010E-02 -1.031E 00  6.899D-01 -1.385E 00 -62.8  0.0
                   0.0        0.0
11  10 -3.751D-05 -1.229E-01 -9.957E-01  5.592D-01 -1.269E 00 -64.0  0.0
                   0.0        0.0
12  10 -3.939D-13 -3.199E-01 -9.643E-01  4.002D-01 -1.156E 00 -64.4  3.939D 01
                   0.0       -1.571E 02
13  10 -1.399D-05 -1.358E-01 -8.417E-01  2.358D-01 -9.132E-01 -73.1  0.0
                   0.0        0.0
14  10 -3.078D-05 -5.246E-02 -7.204E-01  9.145D-02 -7.327E-01 -82.3  0.0
                   0.0        0.0
15  10 -3.123D-05 -3.762E-02 -6.023E-01 -2.927D-02 -6.038E-01  87.0  0.0
                   0.0        0.0
16  10 -1.619D-05 -6.973E-02 -4.910E-01 -1.194D-01 -5.224E-01  75.2  0.0
                   0.0        0.0
17  10 -1.715D-13 -1.362E-01 -3.895E-01 -1.780D-01 -4.813E-01  62.7  1.715D 01
                   0.0       -6.329E 01
18  10 -6.649D-06 -6.991E-02 -2.724E-01 -2.129D-01 -4.070E-01  57.7  0.0
                   0.0        0.0
19  10 -2.183D-05 -1.560E-02 -1.675E-01 -2.330D-01 -3.366E-01  54.0  0.0
                   0.0        0.0
20  10 -3.422D-05  3.890E-02 -7.395E-02 -2.409D-01 -2.649E-01  51.6  0.0
                   0.0        0.0
21  10 -3.191D-05  1.022E-01  1.003E-02 -2.377D-01  2.983E-01 -39.5  0.0
                   0.0        0.0
22  10  3.709D-14  1.829E-01  8.561E-02 -2.234D-01  3.629E-01 -38.9 -3.709D 00
                   0.0        2.485E 01
23  10  8.188D-05  8.426E-02  1.210E-01 -2.094D-01  3.129E-01 -47.5  0.0
                   0.0        0.0
24  10  1.832D-04 -2.960E-09  7.716E-02 -1.028D-01  1.484E-01 -55.3  0.0
                   0.0        0.0

 0   9  7.278D-03  9.943E-09 -5.240E-01  3.589D-01 -7.064E-01 -63.1  0.0
                   0.0        0.0
 1   9  6.236D-03 -4.083E-02 -6.840E-01  7.281D-01 -1.158E 00 -56.9  0.0
                   0.0        0.0
 2   9  5.212D-03 -9.383E-02 -6.659E-01  7.512D-01 -1.186E 00 -55.5  0.0
                   3.028E-02 -1.125E 02
 3   9  4.190D-03 -1.007E-01 -6.475E-01  7.839D-01 -1.204E 00 -54.6  0.0
                  -3.289E-02  0.0
 4   9  3.167D-03 -1.076E-01 -6.190E-01  8.238D-01 -1.226E 00 -53.6  0.0
                  -1.343E-01  0.0
 5   9  2.140D-03 -1.142E-01 -5.858E-01  8.664D-01 -1.248E 00 -52.6  0.0
                  -2.396E-01  0.0
 6   9  1.104D-03 -1.135E-01 -5.557E-01  9.101D-01 -1.271E 00 -51.8  0.0
                  -2.741E-01  0.0
 7   9  6.049D-05 -9.178E-02 -5.494E-01  9.548D-01 -1.302E 00 -51.7  0.0
                  -8.526E-02 -9.185E 01
 8   9 -9.863D-04  1.821E-01 -5.792E-01  9.234D-01 -1.197E 00 -56.2  0.0
                   2.448E 00  0.0
 9   9 -1.954D-03  4.301E-01 -5.885E-01  7.208D-01 -9.618E
                   4.718E 00  0.0
10   9 -2.769D-03  5.800E-01 -4.839E-01
                   5.940E 00  0.0
11   9 -3.393D-03  6.346E
12   9 -3.
```

```
BEAM X MOMENT BETWEEN (  0 ,  5 ) AND ( 24 ,  5 )

X , Y BEAM X MOM

 0   5  0.0          *
 1   5  0.0          *
 2   5 -6.543E-01    *
 3   5 -4.245E-01    *
 4   5  9.912E-02    *
 5   5  6.023E-01    *
 6   5  9.905E-01    *
 7   5  1.130E 00    *
 8   5  1.057E 01            *
 9   5  2.030E 01                    *
10   5  2.321E 01                      *
11   5  1.611E 01                *
12   5  5.041E 00           *
13   5  1.825E 01                  *
14   5  2.260E 01                      *
15   5  1.442E 01               *
16   5  6.831E 00         *
17   5 -6.216E-01    *
18   5 -4.260E-01    *
19   5 -4.138E-01    *
20   5 -4.917E-01    *
21   5 -5.933E-01    *
22   5 -4.868E-01    *
23   5  0.0          *
24   5  0.0          *
```

```
*****  TIME SINCE BEGINNING OF EXECUTION IS      491.96 SECONDS ,

       TIME SINCE LAST CALL OF PRTIME IS      140.69 SECONDS  *****


   PROGRAM SLAB 49  -TEX HWY DEPT DECK-  MATLOCK,PANAK          REV DATE 29 FEB 72

   I-BEAM STRUCTURE          SPANS 50-60-50, 33 FT RDWY, HS20 LOG
   ANALYSIS OF THD THREE-SPAN CONTINUOUS        RUN 29 FEB  1972  JJP    FT-KIP UNITS


*****  TIME SINCE BEGINNING OF EXECUTION IS      492.61 SECONDS ,

       TIME SINCE LAST CALL OF PRTIME IS       0.65 SECONDS  *****

   KEEP RUN TIME RECORDS FOR FUTURE ESTIMATES OF PARENT AND OFFSPRING RUN TIMES
```



1004A

THE AUTHORS

John J. Panak is a research engineer associate with the Center for Highway Research. His experience includes work with the Bridge Division of the Texas Highway Department, and highway design and construction with the California Division of Highways. He is the author and co-author of several technical papers and reports. His primary areas of interest include (1) development and application of computer methods for complex civil engineering problems, (2) structural analysis, and (3) reinforced building and foundation slab analysis, including soil interaction.

Hudson Matlock is a Professor and Chairman of the Civil Engineering Department at The University of Texas at Austin. He has a broad base of experience, including research at the Center for Highway Research and as a consultant for other organizations such as the Alaska Department of Highways, Shell Development Company, Humble Oil and Refining Company, Penrod Drilling Company, and Esso Production Research Company. He is the author of over 50 technical papers and 25 research reports and received the 1967 ASCE J. James R. Croes Medal. His primary areas of interest in research include (1) soil-structure interaction, (2) experimental mechanics, and (3) development of computer methods for simulation of civil engineering problems.