NASA CR 159396
LMSC-HREC TR D497204-II

CELFE: COUPLED EULERIAN-LAGRANGIAN FINITE ELEMENT PROGRAM FOR HIGH VELOCITY IMPACT

PART II PROGRAM USER'S MANUAL

C. H. Lee

LOCKHEED MISSILES & SPACE COMPANY, INC.
HUNTSVILLE RESEARCH & ENGINEERING CENTER

prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NASA Lewis Research Center
Contract NAS3-18908

C. C. Chamis, Project Manager

1 Report No	2 Government Accession No	3 Recipient's Catalog No
NASA CR 159396		

4 Title and Subtitle	5 Report Date
CELFE: Coupled Eulerian-Lagrangian Finite Element Program for High Velocity Impact, Part II Program User's Manual	January 1978
	6 Performing Organization Code

7. Author(s)	8 Performing Organization Report No
C. H. Lee	LMSC-HREC TR D497204-II
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No
Lockheed Missiles and Space Company, Inc. Huntsville Research and Engineering Center P. O. Box 1103 Huntsville, AL 18908	NAS 3-18908
	13. Type of Report and Period Covered
12 Sponsoring Agency Name and Address	Final 7-75 to 12-77
National Aeronautics & Space Administration Washington DC 20546	14. Sponsoring Agency Code

15 Supplementary Notes
Project Manager, C. C. Chamis
Materials and Structures Division
NASA-Lewis Research Center
Cleveland, Ohio 44135

16. Abstract
The CELFE computer program and User's Manual, together with the execution of the CELFE/NASTRAN system, are described. The execution procedure and the transfer of data between the CELFE and NASTRAN programs are controlled through the use of "DATA" files in the Univac 1100 system. Five data files are used to control the runstream and data transfer, and three files are used to hold the programs. These files are contained on a single tape. Changes in NASTRAN routines required by the present analysis are also discussed in this report. All the program listings, except the last two files (where the absolute and relocatable elements are stored), are included in the appendixes.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
High velocity impact, computer program, 3-D finite element, Eulerian-Lagrangian, Coupled Eulerian-Lagrangian, non-linear analysis, material flow, isoparametric, anisotropic material, NASTRAN	Unclassified, Unlimited.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22 Price*
Unclassified	Unclassified	220	

* For sale by the National Technical Information Service, Springfield, Virginia 22161

FOREWORD

This User's Manual is Part II of the final report for Contract NAS3-18908. The report (Parts I and II) details the efforts and accomplishments of the second phase of the contract covering the period June 1975 through September 1977. The study was conducted for the NASA-Lewis Research Center, Cleveland, Ohio, by personnel of the Engineering Sciences Section, Lockheed-Huntsville Research & Engineering Center, Huntsville, Alabama. The NASA-Lewis Project Engineer is Dr. C. C. Chamis, Mail Stop 49-3. Dr. C.H. Lee was the principal investigator for the study, and the study was supervised by B. H. Shirley.

Work was begun on this contract on 28 June 1974. Phase I of the study was completed in June 1975 and an interim report was submitted and published as NASA CR-134933. Efforts on the study were directed toward the local analysis of a three-dimensional high velocity impact problem based on Eulerian representation. During the follow-on study in Phase II, efforts were directed toward the global finite element analysis of the problem by coupling the Eulerian mode with the Lagrangian mode, based on the results obtained in Phase I. An interfacing procedure for coupling the presently developed program, CELFE, with NASTRAN was also accomplished to increase the range of applicability for the program.

ACKNOWLEDGMENT

The author is indebted to Dr. S. T. K. Chan of Lockheed-Huntsville for the use of his subroutines BNDEQ, SFCALL, SFCALQ and SFCALC; Mr. R. Schmitz of Sperry-Rand Corporation for his assistance on modeling the NASTRAN part of the analysis, and developing the "read" and "write" subroutines MATRX, OPENTP and INTVEC; and Dr. A. L. Lee of Lockheed-Huntsville for his advice on computer software.

ABSTRACT

The descriptions and user's manual of the CELFE program, together with the execution of the CELFE/NASTRAN system, are presented as Part II of the final report. The execution procedure and the transfer of data between the CELFE and NASTRAN programs are controlled through the use of "DATA" files in the Univac 1100 system. Five data files are used to control the run-stream and data transfer, and three files are used to hold the programs. These files are contained on a single tape.

Changes in NASTRAN routines required by the present analysis are also discussed in this report. All the program listings, except the last two files (where the absolute and relocatable elements are stored), are included in the appendixes.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Descriptions of the CELFE program developed for high velocity impact analysis, together with the user's manual, are presented. The execution of CELFE and NASTRAN programs and the transfer of data between these two programs are controlled through the use of "DATA" files containing runstream control cards and user data cards in the Univac 1100 system. The complete CELFE/NASTRAN system consists of eight files contained in the delivery tape in the following sequence:

File	Name	Type	Purpose
1	RUN1	DATA (SDF)	Exec 8 control attaching the required files and executing the example problem
2	LEE	Program	CELFE program, source, relocatable and absolute elements
3	NAS1	DATA (SDF)	NASTRAN runstream includes executive, case and bulk data decks
4	START	DATA (SDF)	Assign disk files and executes CELFE for 1st run
5	LOOP	DATA (SDF)	Exec 8 control cards for looping from NASTRAN to CELFE
6	PCH2	DATA (SDF)	GRID cards for the CELFE model
7	NASTRAN	Program	14 NASTRAN absolute elements
8	OBJ	Program	NASTRAN relocatable and selected source elements.

Descriptions of routines in the CELFE program that are stored in the LEE file are presented in Section 2, while the complete listings of the program are given in Appendix A. In Section 3, the interfacing procedure for CELFE/NASTRAN is presented. Some modifications on NASTRAN required for the procedure are also discussed, and the changed routines are listed in Appendix B. A CELFE/NASTRAN finite element model is

demonstrated in the last part of this section for illustration. The procedure for executing the program, including the descriptions of all the files stored in the program tape, are presented in Section 4, and the complete listing of the five data files are tabulated in Appendix C.

2. DESCRIPTION OF CELFE PROGRAM ELEMENTS

The CELFE program consists of one main routine, CELFE; two flags, NASTRN and NFLAG; one PDP PROC, CELFEP; and 31 subroutines. All the routines are stored in LEE file, and listed alphabetically in Appendix A. The functions of the routines are tabulated in Table 2-1.

2.1 SUBROUTINE ALMS

Subroutine ALMS [A-1]* determines the pressure distribution using the Los Alamos equation of state. This subroutine is called after the solution in density, momentum and energy are determined. The calling sequence is written as

```
Call ALMS(NPMT, NPS, NPPJ, NPPJT, INTF, INTFT, NFRAC,
     RHOT, RHOP, VER, PRS)
```

The descriptions of the arguments of ALMS are given in Table 2-2.

2.2 SUBROUTINE BNDEQ

This is the in-core banded matrix equation solver [A-4] used for solving all the governing equations. The calling sequence reads

```
Call BNDEQ(A, I, J, K, L)
```

where

- A = compact form of the banded coefficient matrix plus loading vector
- I = maximum number of rows
- J = maximum number of columns
- K = total number of unknowns, and
- L = half bandwidth.

*The number in the bracket indicates the compiled listing page number of the described subroutine.

Table 2-1

LIST OF SUBPROGRAMS IN CELFE

Subprogram Title	Subprogram Function	Compiled Listing Page
ALMS	Los Alamos equation of state	A-1
BNDEQ	In-core banded matrix equation solver	A-4
CELANI	Routine for generating and assembling element matrices of constitutive equations	A-7
CELDIS	Routine for generating and assembling element matrices in coupling modes	A-12
CELFE	Main program	A-19
CELFEP	FORTRAN procedure definition processor	A-32
CELTRN	Principal routine for interfacing CELFE/NASTRAN system	A-38
CEL3JB	Jacobian updating routine	A-45
CEL3CV	Routine for generating and assembling element matrices of conservation equations	A-48
ELAGVA	Subroutine computing element matrices for momentum equations in L_c zone	A-53
ELCELA	Subroutine computing element matrices for constitutive equations in impact zone	A-56
ELCELD	Subroutine computing element matrices for displacements in impact zone	A-59
ELCELV	Subroutine computing element matrices for conservation equations in impact zone	A-61
ELDISA	Subroutine computing element matrices for displacements in L_c zone	A-64
ELMATS	Translation of element matrices for constitutive equations from CELFE	A-86
ELMATX	Translation of element matrices for conservation equations and displacements from CELFE	A-88
FRCRIT	Chamis' failure criterion for composites	A-67
GEOMTR	Subroutine defining initial finite-element mesh, free surface nodes, and related geometric conditions from input data	A-72
INTVEC	Reading subroutine for interpreting a vector in NASTRAN OUTPUT2 format	A-75
INVDET	Matrix inversion routine	A-79
MATMUL	Matrix multiplication routine	A-80
MATRX	Writing subroutine for CELFE data in NASTRAN INPUTT2 format	A-81
MESHUP	Mesh updating routine	A-90
OPENTP	Reading subroutine for data written in NASTRAN OUTPUT2 format	A-95
OUTMAT	Bridging routine for element matrices translated by ELMATS and ELMATX to MATRX	A-99
SFCALC	Shape functions for cubic isoparametric element	A-103
SFCALL	Shape functions for linear isoparametric element	A-101
SFCALQ	Shape functions for quadratic element	A-102
SLTEST	Routine for testing rebound, sliding or penetration	A-108
SORT	Sorting routine	A-119
STIFFG	Reading routine for material stiffness coefficients of anisotropic materials	A-120
STIFF	Computting routine for material stiffness coefficients for orthotropic materials	A-123
STRENG	Viscoplastic yielding criterion for isotropic materials	A-125

Table 2-2

DESCRIPTIONS OF PROGRAM VARIABLES

Word	Description
A	Time step multiplicity, a
ALPHA	Relaxation factor, α
BFS (I, J)	Body force, F_j (I = number of nodes)
C (I, J, K)	Material stiffness coefficients C_{jk} (I = number of nodes)
DIS (I, J)	Displacements, u_j, at present integration step (I = number of nodes)
DISN (I, J)	u_j at INNER = 1
DISPRV (I, J)	u_j at previous time step
DT	Time increment
IBDIS (I, J)	Nodal number of boundary node for displacements u_j (I = 1, 2, . . . , IBDIST(J))
IBDIST (J)	Total number of boundary nodes for u_j
IBDS (I, J, K)	Nodal number of boundary nodes for deviatoric stresses S_{jk} (I = 1, 2, . . . , IBDST(J, K))
IBDST(J, K)	Total number of boundary nodes for S_{jk}
IBDV(I, J)	Nodal number of boundary nodes for V_1, V_2, V_3, E and ρ (I = 1, 2, . . . , IBDVT(J))
IBDVT(J)	Total number of boundary nodes for V_1, V_2, V_3, E and ρ
ICOND	Parameter for generation of initial data: ICOND = 0 – Read data from input cards = 1 – Generate data from main program
INNER	= 1 and 2, the number of iterations in two-step time integration procedure
INTF(I)	Nodal number of impact nodes (I = 1, . . . , INTFT)
INTFT	Total number of impact nodes
IORDC, IORDL	Order of finite elements in impact zone and Lagrangian zone, respectively
ISOP, ISOT	Materials of target and projectile, respectively: = 0 – Isotropic = 1 – Orthotropic
ITER	Number of time steps
ITGIV	Total number of time steps
IUNIT	Unit: = 1 – ISU = 2 – English
NASTRN	Flag controlling the run: NASTRN = 0 – In-core CELFE run = 1 – CELFE/NASTRAN run
NELG, NPLG	Number of elements and nodes, respectively, for starting the L_c mesh

Table 2-2 - (Concluded)

Word	Description
NELT, NPST	Total number of elements and nodes, respectively, for the CELFE substructure
NEPJ, NPPJ	Number of element and node, respectively, for starting the projectile mesh
NEPJT, NPPJT	Number of last element and node, respectively, for the projectile mesh
NET, NPS	Total number of elements and nodes, respectively, in impact zone
NETR, NPTR	Number of element and node, respectively, for starting the interface nodes between the impact zone and the Lagrangian zone
NFLAG	Flag controlling the CELFE run
NFRAC	Total number of failure nodes
NFS(I)	Nodal number of free surface nodes (I = 1, . . . , NFST)
NFST	Total number of free-surface nodes
PMIU, TMIU	Viscosities for projectile and target materials if ISOP, ISOT = 0, respectively
PNIU, PPNIU	Poisson's ratio for target and projectile when ISOP, ISOT = 0, respectively
PRS(I)	Pressure distribution in impact zone at present step (I = number of node)
PRSN(I)	Pressure at INNER = 1
PRSPRV(I)	Pressure at previous time step
PYLD, TYLD	Yielding stresses for projectile and target when ISOP, ISOT = 0, respectively
RHOP, RHOT	Initial densities for projectile and target
SLC, SLT	Unidirectional failure stresses under compressive and tensile loads, respectively
SMIU	Coulomb's friction coefficient
STS(I, J, K)	Deviatoric stresses S_{jk} at present step (I = number of node)
STSN(I, J, K)	Deviatoric stresses S_{jk} at INNER = 1
STSPRV(I, J, K)	Deviatoric stresses S_{jk} at previous time step
TIME	Time in μsec
V(J)	Impact velocity V_{oj}, J = 1, 2, 3
VBD, SBD, DISBD	Boundary conditions for VER, STS, and DIS, respectively
VER(I, J)	Momentums, energy and density, V_1, V_2, V_3, E and ρ (J = 1, . . . , 5) at present step (I = number of node)
VERN(I, J)	V_1, V_2, V_3, E and ρ at INNER = 1
VERPRV(I, J)	V_1, V_2, V_3, E and ρ at previous time steps

2.3 SUBROUTINES CELANI, CELDIS AND CEL3CV

Subroutines CELANI [A-7], CELDIS [A-12] and CEL3CV [A-48] are principally and logically identical routines for generating and assembling the element matrices of constitutive equations, displacement equations, and conservative equations, respectively. The respective calling sequences are read as follows:

```
Call CELANI(INTF, INTFT, CINT, L1, L2)
Call CELDIS(I)
Call CEL3CV(J)
```

Here, INTF, INTFT and CINT are described in Table 2-1; and

L1, L2, I = 1, 2, 3, representing the coordinates x_1, x_2, x_3, respectively.

J = 1, . . . , with 1, 2, 3 representing momentums on x_1, x_2, x_3; 4 representing the total energy; and 5 the density.

FORTRAN PROC PARAM1 is inserted in all three subroutines to define the sizes of dimensional variables. The FORTRAN PROCs CONSTV, DISPLV and CONSRV are also inserted, respectively, to subroutines CELANI, CELDIS and CEL3CV, for storing the common blocks.

As an illustration, the flow chart for CELANI is depicted in Fig. 2-1.

2.4 CELFE – MAIN PROGRAM

CELFE [A-19] reads the flag NASTRN for the option (whether the in-core or coupled CELFE/NASTRAN procedure) to be utilized in the computations as the starting point. Following NASTRN, the flag NFLAG and related parameters controlling the procedure, and numerical data for mesh, material properties, etc., are read subsequently from data cards. Then, the program provides the main runstream for the entire impact analysis. The procedure description is illustrated in the flow chart depicted in Fig. 2-2.

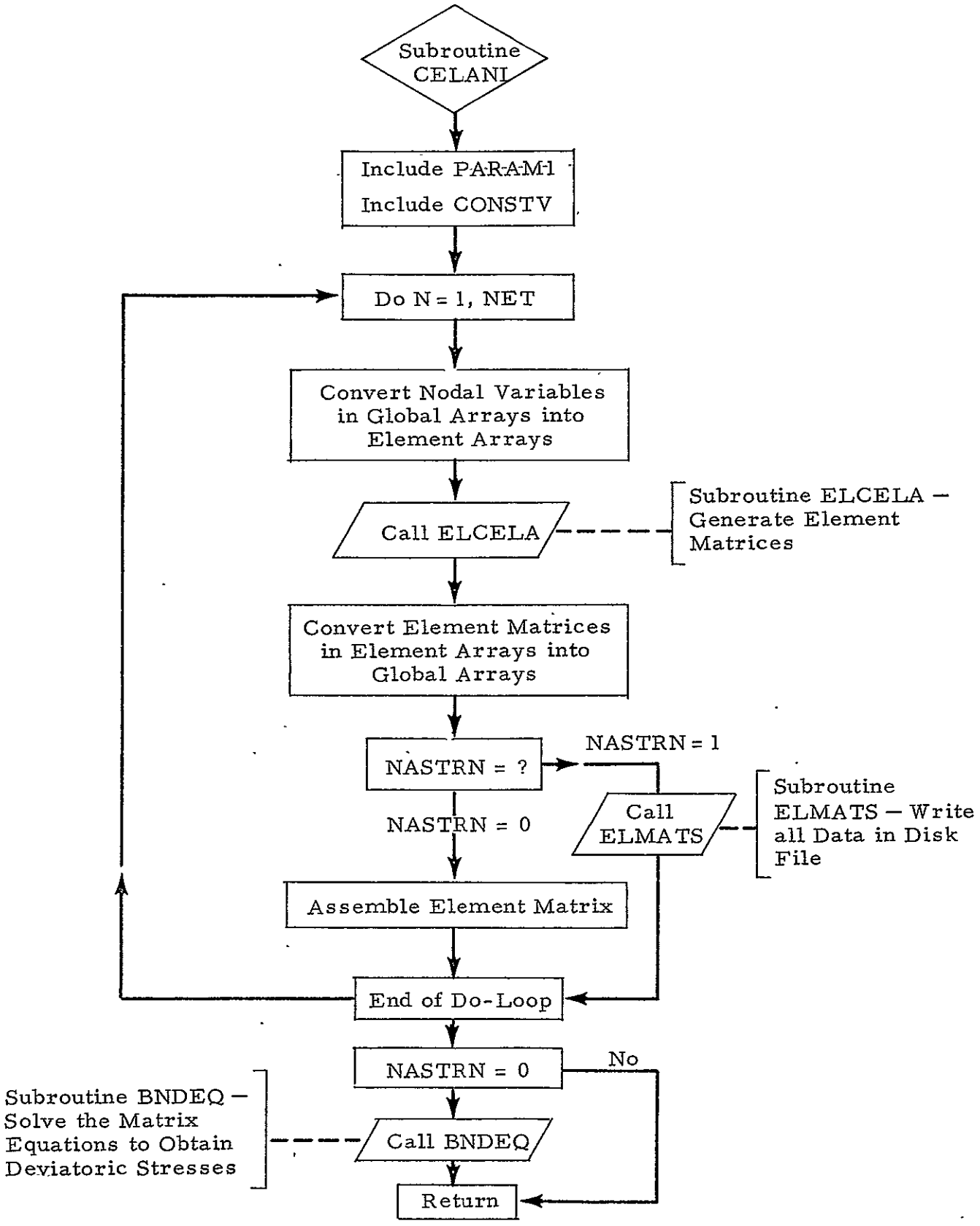


Fig. 2-1 - Flow Chart for Subroutine CELANI

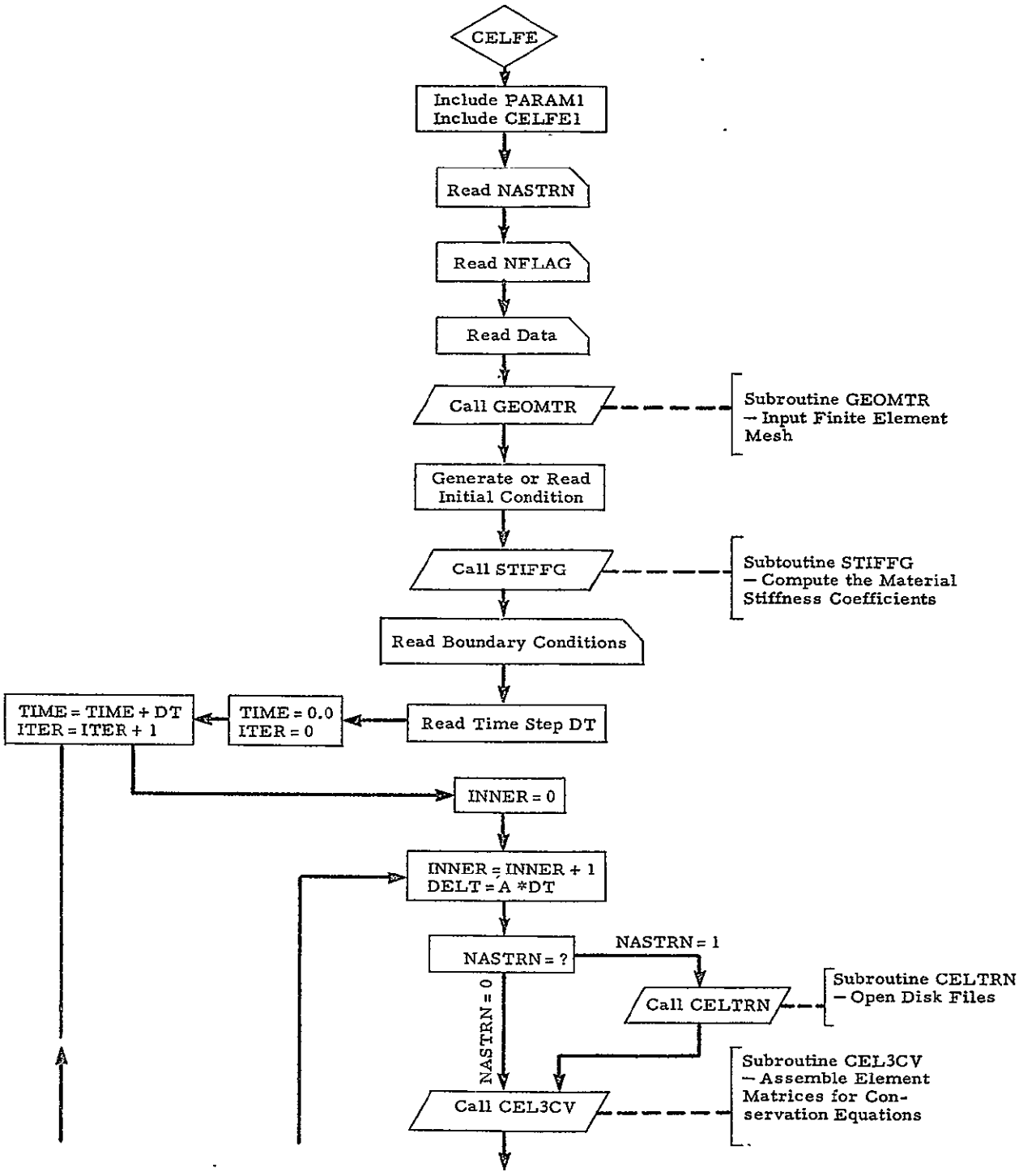


Fig. 2-2 - Flow Chart for Main Program CELFE

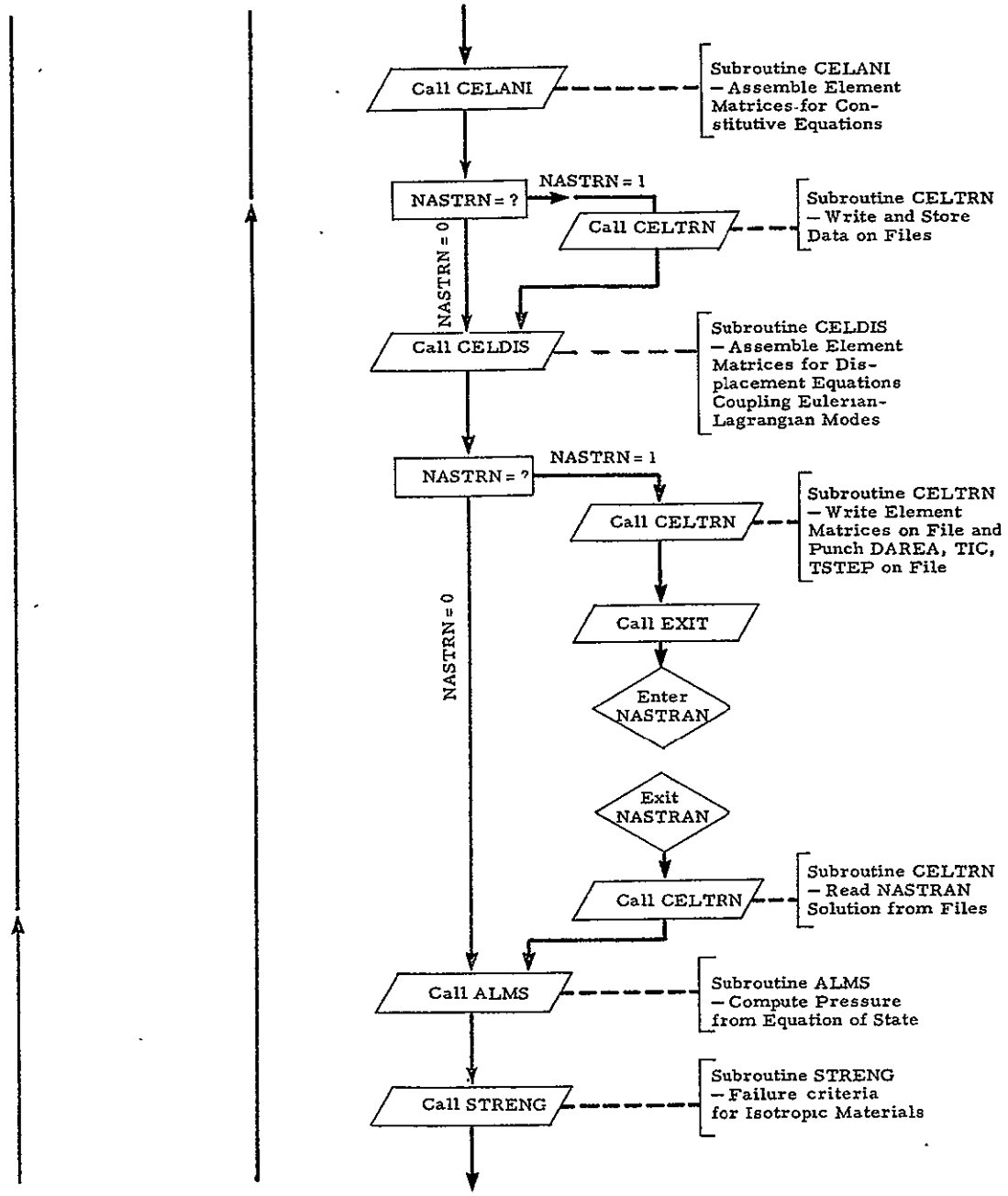


Fig. 2-2 (Continued)

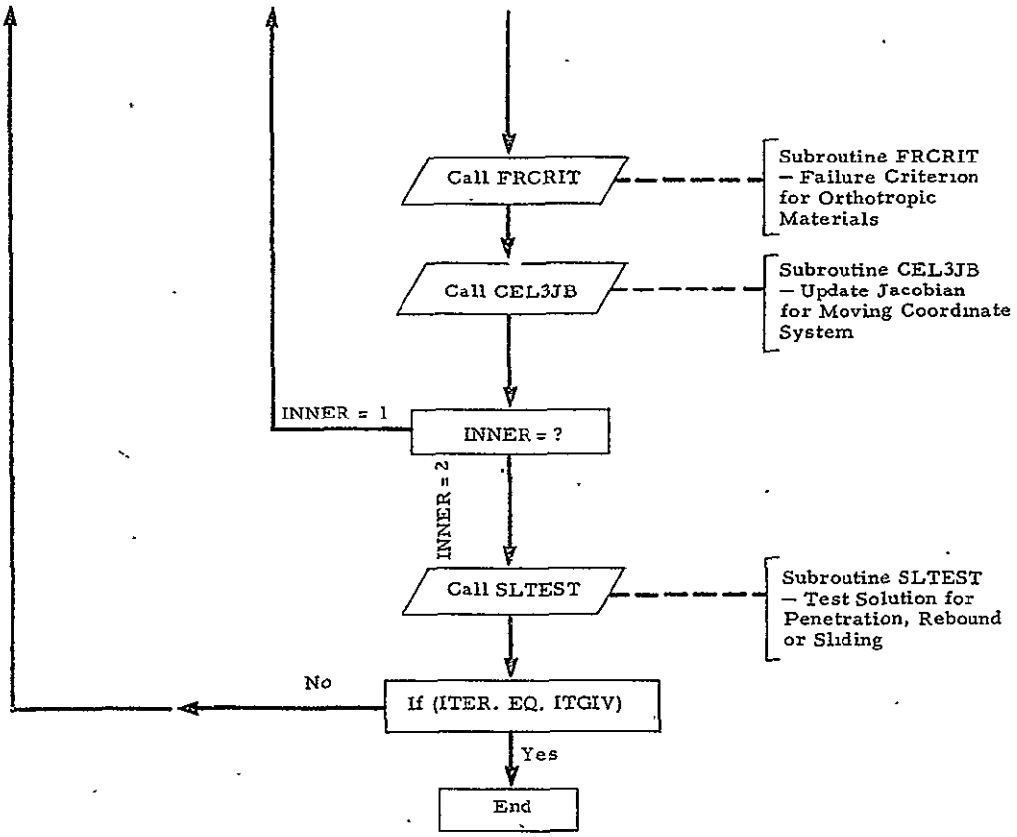


Fig 2-2 (Concluded)

2.5 PDP CELFEP

The procedure definition processor CELFEP [A-32] consists of ten FORTRAN PROCs:

PARAM1 PROC defines the sizes of the parameters for the main program and all major subroutines except GEOMTR and CELTRN.

PARAM2 PROC defines the sizes of the parameters for GEOMTR.

CELFE1 PROC stores related dimensions and common blocks in the main routine CELFE.

CONSRV PROC stores the common blocks for subroutine CEL3CV.

CONSTV PROC stores the common blocks for subroutine CELANI.

DISPLV PROC stores the common blocks for subroutine CELDIS.

GENMSH PROC stores the common blocks for subroutine GEOMTR.

TESTNG PROC stores the related parameters, dimensions, and common blocks for subroutine SLTEST.

CELSTR PROC defines the parameters, data, and dimensions, as well as stores the common blocks for subroutine CELTRN.

STATEQ PROC defines the data for subroutine ALMS.

The parameters are described in Table 2-3.

2.6 SUBROUTINE CELTRN

This subprogram [A-38] serves as the principal routine for interfacing the CELFE/NASTRAN system. It is called only when the flag NASTRN=1, and the calling sequence reads

Call CELTRN (K)

where K = 1, . . . , 4.

The runstream of CELTRN can be described according to the value of K as

K = 1: Store all the solutions of the previous time step, related data and parameters in disk file 16; also, call subroutine OPNNAS to open the disk file KFILE for storing the matrices computed from subroutines CELANI and CEL3CV in NASTRAN INPUTT2 format.

Table 2-3

DESCRIPTIONS OF PARAMETERS IN FORTRAN PROCs

Word	Description
IU = 5	Dimension of the conservation equations.
NBV, NBS, NBD	Dimensions of the boundary nodes for conservation equations, constitutives equations, and displacements, respectively, in CELFE zone.
NCMAX, NRMAX	For in-core CELFE run: Dimensions of column and row, respectively, for banded matrices; for CELFE/NASTRAN run: NCMAX = 2, NRMAX = MAXDOF (number of degrees of freedom).
NDFRS	Dimension of the free surface nodes.
NDINT	Dimension of the impact nodes.
NDPJT	Dimension of the projectile nodes.
NEM	Dimension of elements in impact zone.
NEMT	Dimension of elements in CELFE zone.
NFRC	Dimension of failure nodes.
NGP	Number of Gaussian points used in impact zone.
NGPLG	Number of Gaussian points used in L_c zone.
NPE	Dimension of nodes for each element in impact zone.
NPELG	Dimension of nodes for each L_c-element.
NPM	Dimension of nodes in impact zone.
NPMT	Dimension of nodes in CELFE zone.
NTRL = 7	Dimension of trailer.
NTSTEP	Dimension of time-step in the run.
NWK	Dimension of working space in interfacing with NASTRAN.

K = 2: Call subroutine OPNNAS to open the disk file LFILE for storing matrices computed from CELDIS in NASTRAN INPUTT2 format.

K = 3: Read the coordinates from breakpointed disk file 18 in NASTRAN bulk data format GRID, and update the mesh. Then punch the image cards of the coordinates and store in breakpointed disk file 19 in the same format. At the same time, update the DAREA, DIT and TSTEP, and punch and store in the same breakpointed disk file 19.

K = 4: Call subroutine OPENTP to open the disk files MFILE and NFILE. Then read the stored data in file 16, and solutions computed by NASTRAN and stored in MFILE and NFILE in OUTPUT2 format.

2.7 SUBROUTINE CEL3JB

This subroutine [A-45] updates the Jacobian of the general moving coordinate system with respect to the fixed frame. The computations cover only the mesh in the impact zone. The calling sequence reads

```
Call CEL3JB(NPM, NEM, VERPRV, VERN, VER, NOD, NPE, FJPRV,
     FJ, KODE)
```

where the arguments are defined in Tables 2-2 and 2-3.

2.8 SUBROUTINES ELAGVA, ELCELA, ELCELD, ELCELV AND ELDISA

These logically identical subroutines compute the element matrices following the finite element formulations of governing equations in various zones:

ELCELV [A-61] — For conservation equations in the impact zone

ELCELA [A-56] — For constitutive equations in the impact zone

ELCELD [A-59], ELAGVA [A-53], and ELDISA [A-64] — For coupling Eulerian-Lagrangian modes using lateral displacements and velocities as the coupling variables. Here, the first subroutine is for the impact zone, and the last two are for the L_c zone.

Three-dimensional isoparametric elements are used in all subroutines. ELCELD, ELAGVA and ELDISA are called only when the L_c zone is not empty. (cf. Part I of the present report for the discussions and finite element formulations.)

2.9 SUBROUTINES FRCRIT AND STRENG

FRCRIT [A-67] is the principal subroutine for predicting the failure of composites based on Chamis' failure criterion; and STRENG [A-125] is for isotropic materials based on von Mises' criterion. The numerical procedures follow directly the discussions given in Section 6.4 of Part I.

The calling sequences are read, respectively,

```
Call FRCRIT(NPMT, NPM, STS, INTF, C, CSTF, NCNT, NFRC,
     NDFRC, NFS, EF, PNF, CKF, KODE)
```

and

```
Call STRENG (NPM, STS, YIELD, NFS, INTF, KODE)
```

where the arguments are described in Tables 2-2 and 2-3.

2.10 SUBROUTINE GEOMTR

This subroutine [A-72] reads the finite element mesh from input data for the CELFE substructure, i.e., for the impact zone and the L_c zone. Its calling sequence is

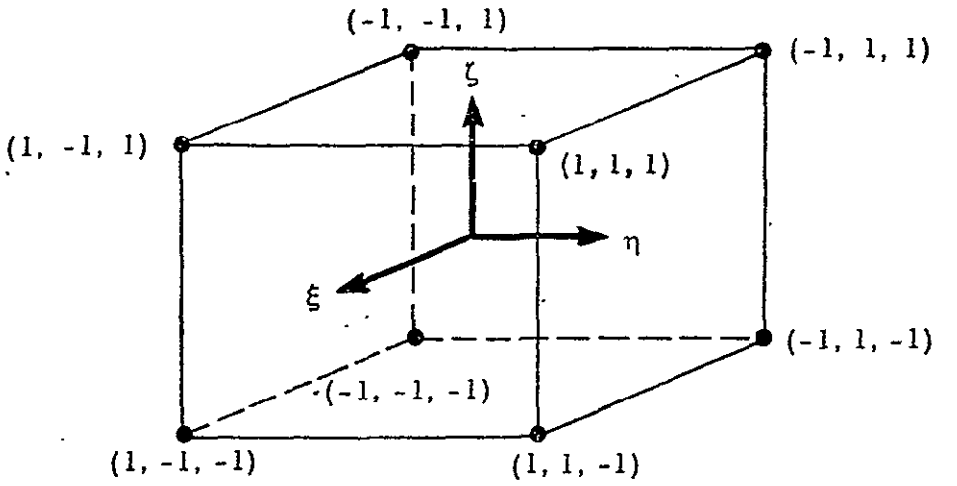
```
Call GEOMTR(X, Y)
```

where

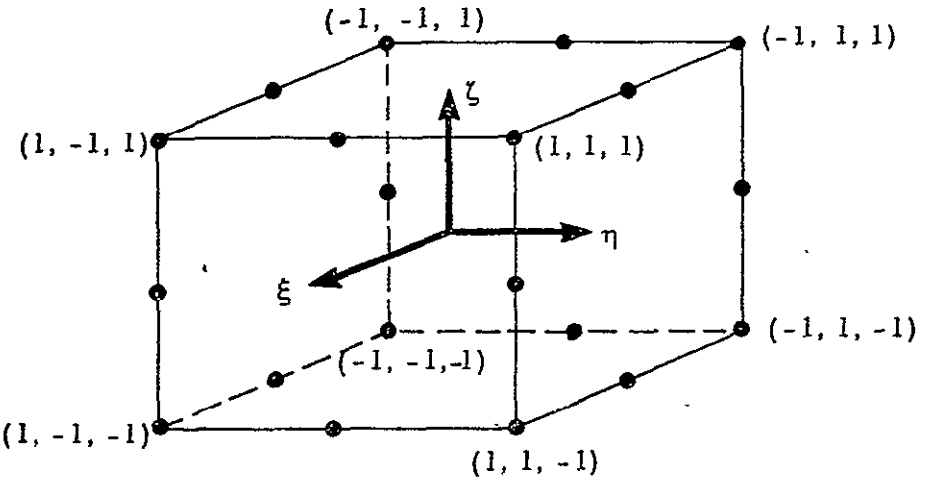
X = nodal coordinates for the entire CELFE substructure

Y = respective dimensions of the impact zone

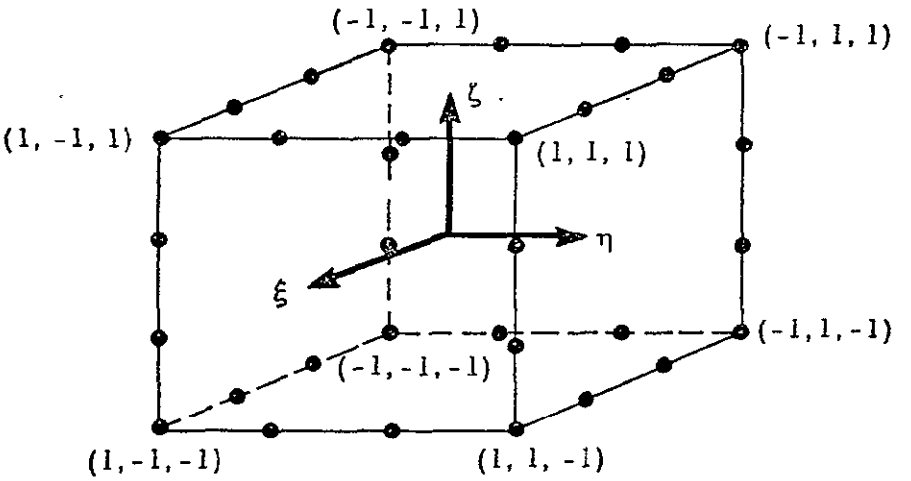
The 3-D quadralateral elements must be used in the CELFE substructure. The serendipity family available in CELFE consists of linear, quadratic and cubic elements as shown in Fig. 2-3.



a. Linear Element



b. Quadratic Element



c. Cubic Element

Fig. 2-3 - Location of Nodal Points in Three-Dimensional Elements

In GEOMTR, the free surface nodes for the impact zone, the impact surface nodes, and the interfacial nodes are also defined from the input data. The flow chart of the routine is depicted in Fig. 2-4.

It is noted that the geometry of the CELFE substructure must be divided a priori into the impact zone (including the projectile) and the L_c zone. The dimensions of the impact zone chosen must be large enough to cover the possible failure zone. The partition is illustrated in Fig. 2-5.

The elements and the corresponding nodes must first be numbered in the impact zone, this process repeated in the L_c zone; or vice versa. The numbering cannot be mixed among different zones, which is also applied when the L_N zone (NASTRAN zone) is included.

The input data are stored in START file together with those read in main routine CELFE.

METHOD FOR 'GEOMTR'

1. Generating Elements for the Impact Zone: The statements are written as

```
      DO 210  N = 1, NET
      READ 1000, (NOD(N, I), I = 1, NPE)
210   PRINT 1150, N, (NOD(N, I), I = 1, NPE)
1000  FORMAT (20I4)
```

Here, we denote

NET	=	total number of elements in the impact zone
NPE	=	number of nodes in each element for the impact zone
NOD(N, I)	=	global numbering of the node at the N^{th} element corresponding to the I^{th} local numbering.

2. Generating Elements for the L_c Zone: Following the statements in (1), we write

```
      NN = NET + 1
      DO 211  N = NN, NELT
      REAL 1000, (NOD(N,I), I = 1, NPELG)
211   PRINT  1150, N, (NOD(N,I), I = 1, NPELG)
```

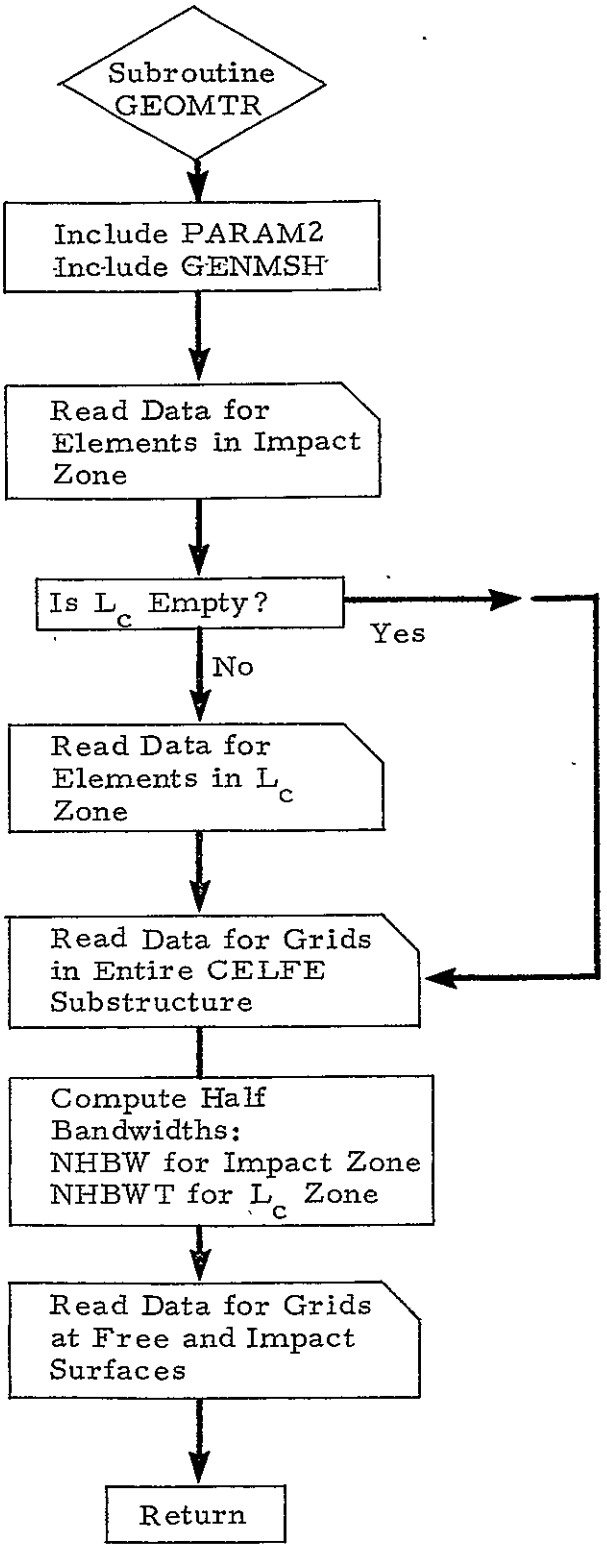


Fig. 2-4 - Flow Chart for Subroutine GEOMTR

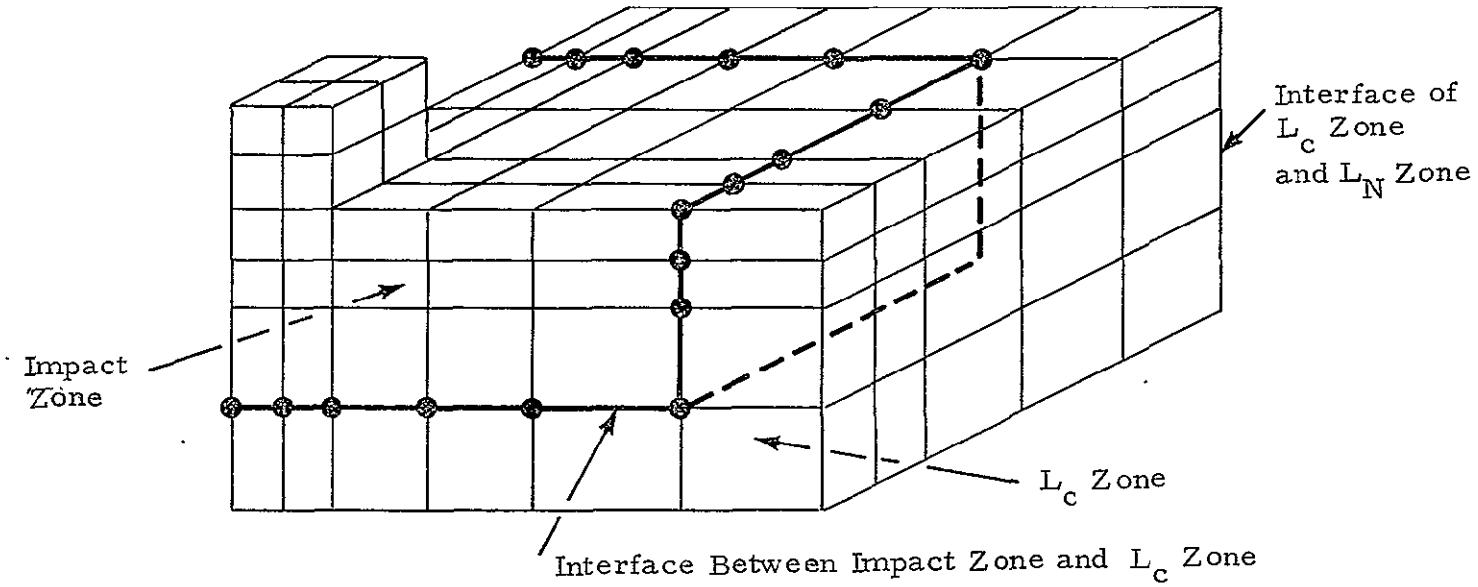


Fig. 2-5 - Partition of the Impact Zone and L_c Zone at t = 0

Here, the integers

NELT = total number of elements in the CELFE substructure

NPELG = number of nodes in each element of the L_c zone.

Thus, the total number of elements in the L_c zone is equal to NELT-NET. In addition, note that the parameters NPE and NPELG may have different values depending on the order of elements used in the impact zone and the L_c zone. For example, if linear elements are used in the impact zone, and quadratic elements in the L_c zone, then NPE = 8, NPELG = 20.

The numbering of nodes, NOD(N, I), in the finite element mesh must start from axes parallel to the impact plane, as illustrated in Fig. 2-6.

3. Generating Mesh in the Entire CELFE Substructure: The mesh in the CELFE structure is represented in different coordinate systems characterized by the integer KODE(I). The read statement can be written as

```
      DO  220  N = 1, NPST
      READ 1300, G, KODE(N), (X(N, I), I = 1, 3)
220   PRINT  1305, G, KODE(N), (X(N, I), I = 1, 3)
1300  FORMAT (16A1, I8, 3F8.3)
```

Here, the integers

NPST = total number of grids (nodes) in the CELFE substructure

KODE(N) = 0 Eulerian mesh
= 1 E-L mesh
= 2 Interfacial mesh for E-L zone and Lagrangian zone
= 3 L_c mesh

The format of the input cards is similar to the NASTRAN bulk data format GRID, with the given number of KODE(N) punched on the third (CP) field:

	N	KODE(N)	X(N, 1)	X(N, 2)	X(N, 3)	
GRID	8	0	1.0	2.5	0.0	
1	8	16	24	32	40	48

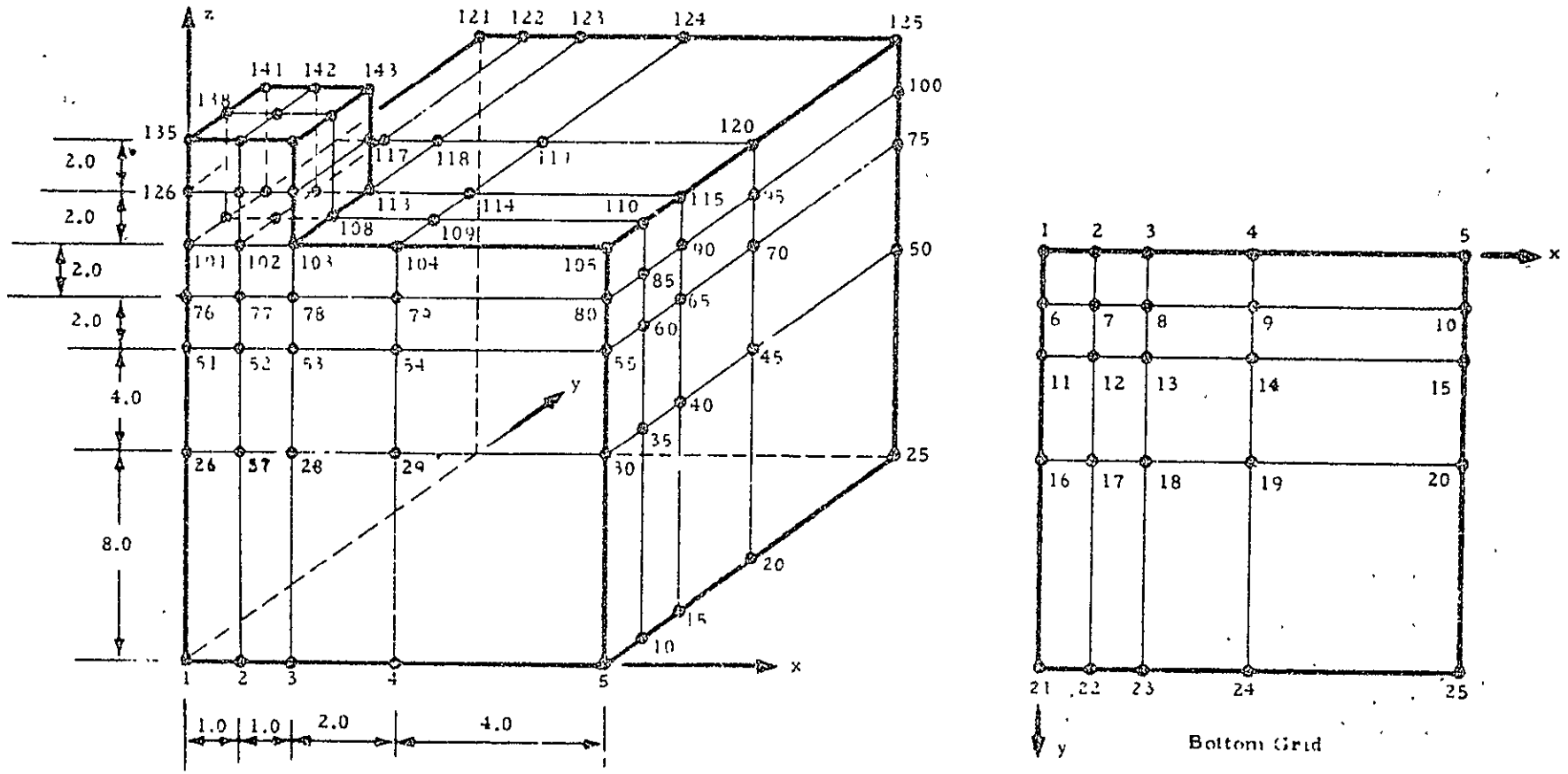


Fig. 2-6 - Numbering of Nodes in the Finite Element Mesh

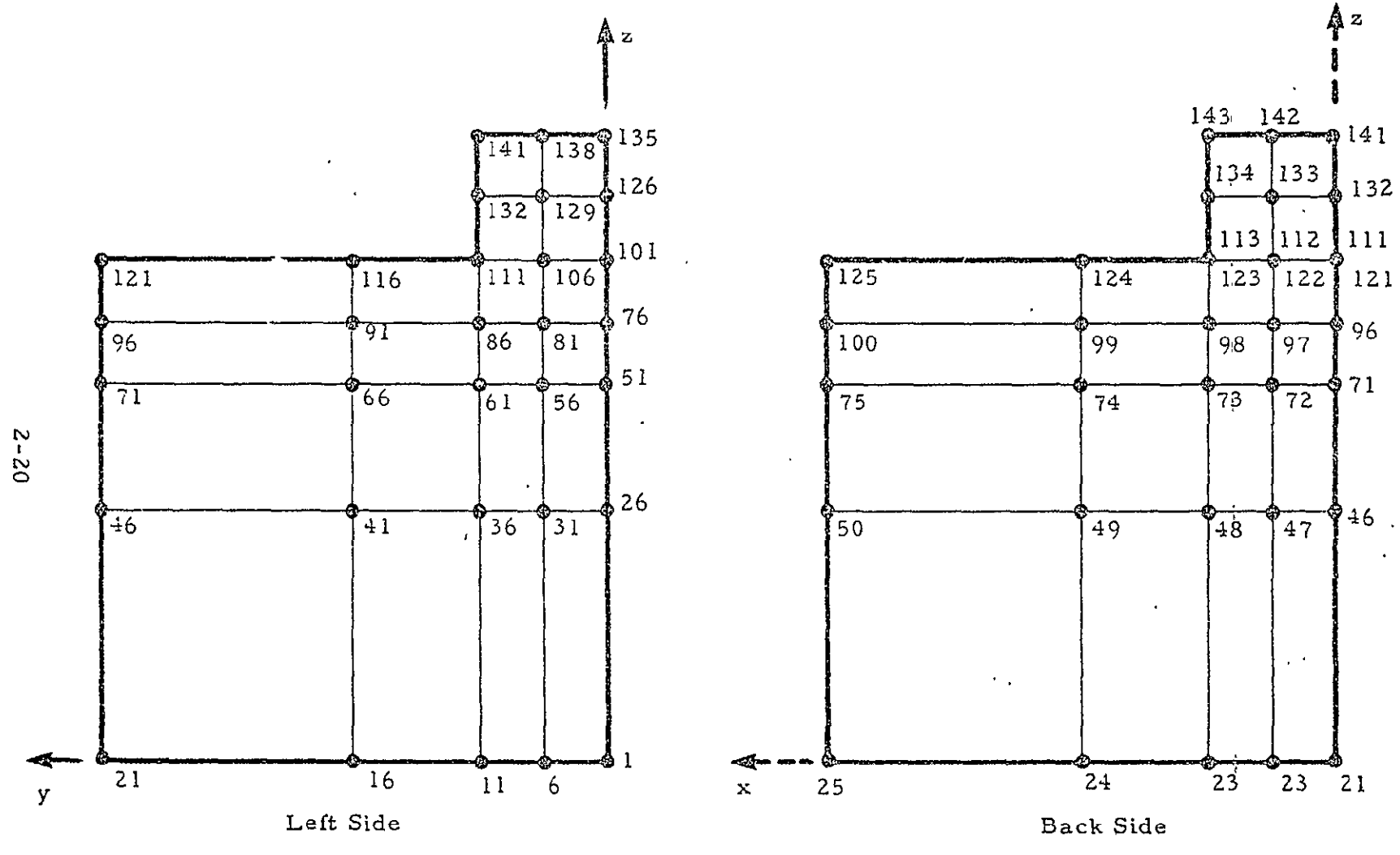


Fig. 2-6 (Continued)

4. Read Free Surface Grids and Impact Surface Grids: The statements read as

```
      READ   1000, (NFS(I), I = 1, NFST)
      READ   1000, (INTF(I), I = 1, INTFT)
1000  FORMAT (20I4)
```

Remarks

- In general, all the KODE(I)s in the impact zone are set to be 1, except the interface nodes between the impact zone and the Lagrangian zone where the number of KODE(I) is set to be 2.
- If the mesh in the impact zone is too coarse and cannot be refined, it is suggested that at the interior nodes of the impact zone KODE(I) = 0; on free surface, KODE(I) = 1; and at the interface, KODE(I) = 2.
- When sliding is likely to occur due to oblique impact, the CELFE substructure must be defined to be a band covering the entire sliding path on the target (see Fig. 2-7). The integer KODE(I) $\equiv$ 1 for the entire CELFE substructure, except those interface nodes of the CELFE substructure and the L_N zone (NASTRAN zone), where KODE(I) = 2.

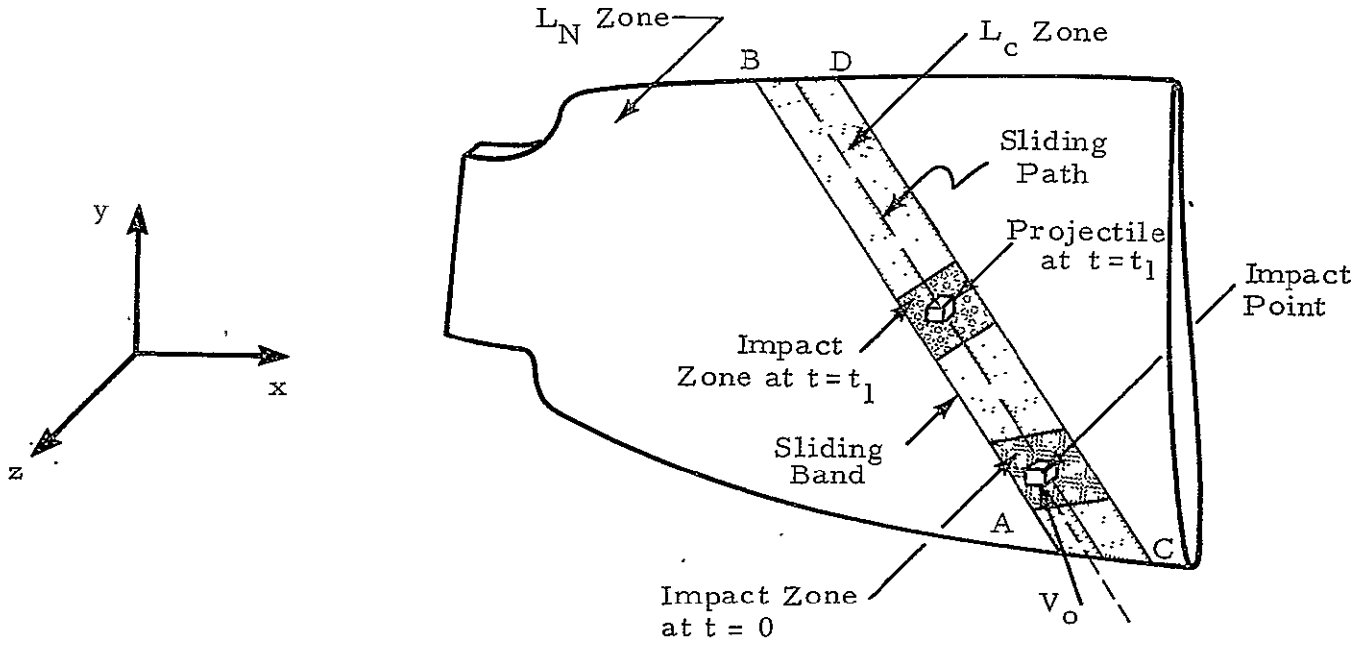


Fig. 2-7 - Illustration of Sliding Band (as Denoted by ABCD)

2.11 SUBROUTINES INTVEC, MATRX AND OPENTP

These multiple entry subroutines are key routines for interfacing CELFE/NASTRAN.

Writing Subroutines

MATRX [A-81] — Write NASTRAN formatted data block for INPUTT2

Entry Points:
- OPNNAS — Open file, write data on header
- MATRX — Write second record name and trailer
- COLUMN — Write current column index
- ELEMNT — Write one row index and value into file
- WEOF — Write EOF for current data block.

Reading Subroutines

INTVEC [A-75] — Interpret a vector from OUTPUT2

OPENTP [A-95] — Read NASTRAN data block from OUTPUT2

Entry Points:
- OPENTP — first call to process NASTRAN tape made by OUTPUT2
- OPENF — READ second record, name and trailer
- READF — NASTRAN data block begins from third record.

METHOD FOR 'MATRX'

1. Open file — Call OPNNAS (FILE, WORK, NWD, IDATE, LABEL, 1)
2. Write matrix name trailer — Call MATRX (NAME, NCOL, NROW, NFORM, NTYPE)
3. Write matrix terms by columns

```
DO   I = 1, NCOL
CALL COLUMN (I)
DO   J = 1, NROW
```

```
CALL  ELEMNT(J,D)        where D is value of matrix
CONTINUE                   D(J,I)
```

4. Close — Call WEOF

Word	Definition
IDATE(3)	Month, day, year
LABEL(2)	Value of P3
NAME(2)	Matrix name, NASTRAN data block name
NCOL	Number of columns
NROW	Number of rows
NFORM	Form of Matrix 1 — Square 2 — Rectangular 3 — Diagonal 4 — Low Triangular 5 — Upper Triangular 6 — Symmetric 7 — Row Vector 8 — Identity
NTYPE	Type 1 — S. P. 2 — D.P. 3 — CPSP 4 — CPDP

METHOD FOR "READ"

OBJECTIVE — To read a matrix written by NASTRAN OUTPUT2.

1. Open File Reads Tape Header — Call OPENTP (FILE, N, NWD, IERR, ILB)
2. Read Matrix (or Table) Name and Trailer — Call OPENF(IERR, ITRL)
3. Read Matrix (or Table) Values by Column — Call READF($\$N_1$, $\$N_2$, A, WD, FLAG, IERR)

Word	Definition
FILE	FORTRAN unit number (12, 14, 15, etc.)
N	Working buffer
NWD	Size of buffer

IERR	Flag use to locate fatal error
ILB	If, 0, no tape label record; if, +1, read tape label record
ITRL(7)	Seven work trailer
N_1$	Statement number for 1st return, READF encountered EOR while reading
N_2$	Statement number for 2nd return, call SKIPF
A	Array for matrix
WD	Size of working buffer
FLAG	Number of row left in current record.

Remarks

1. If an error return occurs from OPENTP,

IERR	Meaning
-5	Incorrect number of words in date label
-4	Incorrect number of words in header
-3	Incorrect number of words in tape ID (P3)
-2	EOR for first record missing
-1	EOF for first record missing.

2. If an error return occurs for OPENF,

IERR	Meaning
-14	Incorrect number of words in data block name, or unexpected EOF encountered
-13	EOR missing
-12	Incorrect number of words in data block trailer
-11	EOR missing after data block name.

3. If an error return occurs from READF,

IERR	Meaning
-22	EOR encountered when reading key
-23	Value for EOR is not correct for this record
-24	EOR is not correct value
-25	Insufficient working core to hold FORTRAN logical record. Increase the value of NWD in OPENTP.
-26	Unexpected EOR encountered.

4. If an error return occurs from SKIPF,

IERR	Meaning
-31	EOR is not correct for this record
-32	Number of words to be read is incorrect
-34	Number of words to be read is greater than the number of words in buffer area.

5. Entry point READF is for general use in reading tables or matrices.

6. The subroutine INTVEC may be used for reading a single matrix vector.

2.12 SUBROUTINES ELMATS AND ELMTRX

These routines bring the element matrices, each element at a time, from the subroutines CELANI, CEL3CV and CELDIS into the subroutine OUTMAT. Here,

ELMATS [A-86] — Brings element matrices out from CELANI

ELMAT [A-88] — Brings element matrices out from CEL3CV and CELDIS

The subroutines are logically identical and are called, respectively, in CELANI, and CEL3CV, CELDIS (cf. Fig. 2-1). As an illustration, Fig. 2-8 depicts the flow chart for ELMATS which follows the flow chart given in Fig. 2-1.

2.13 SUBROUTINE OUTMAT

This subroutine [A-99] serves as the bridging routine to extract all the element matrices and loading vectors from CELFE, and write in files opened in MATRX on NASTRAN INPUTT2 format. The subroutine is called by ELMATS, and ELMTRX as

```
Call OUTMAT (NAME, CD, NUMNP, NFORM, NOD,
     NELT, NPELG, NN, L, IPARAM)
```

where

NAME	=	Matrix name, NASTRAN D. B. name
CD	=	Element matrix or loading vector
NUMNP	=	1 — Element matrix
	=	2 — Element loading vector

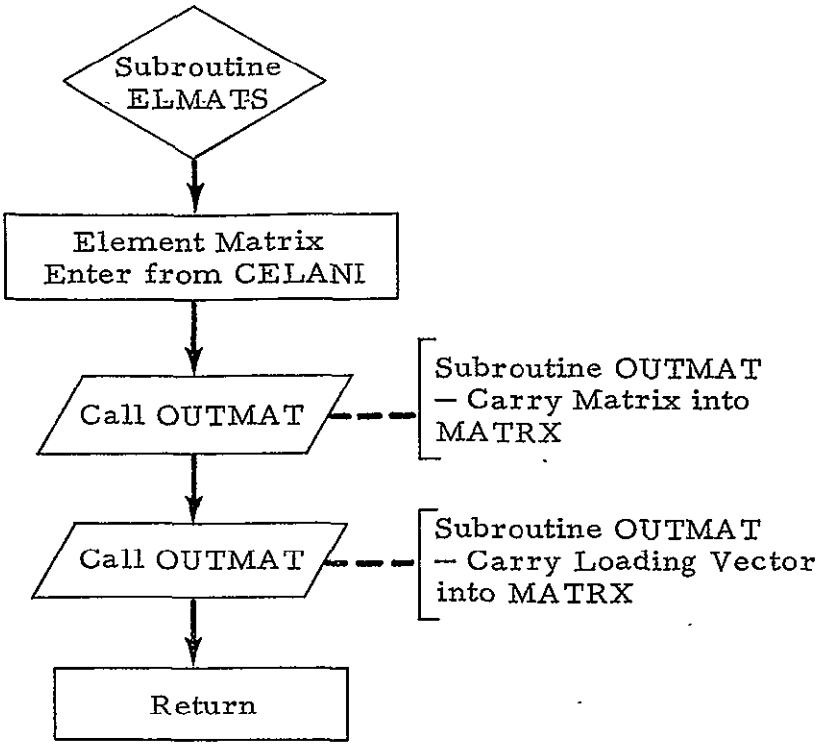


Fig. 2-8 - Flow Chart for ELMATS

NFORM	=	Form of the matrix
NN	=	The number of the element being carried
L	=	The number of coordinates, 1 for X_1, 2 for X_2, and 3 for X_3
IPARAM	=	1 No coupling involved
	=	2 Coupling procedure

The remaining arguments in OUTMAT are described in Table 2-1, and the flow chart is shown in Fig. 2-9.

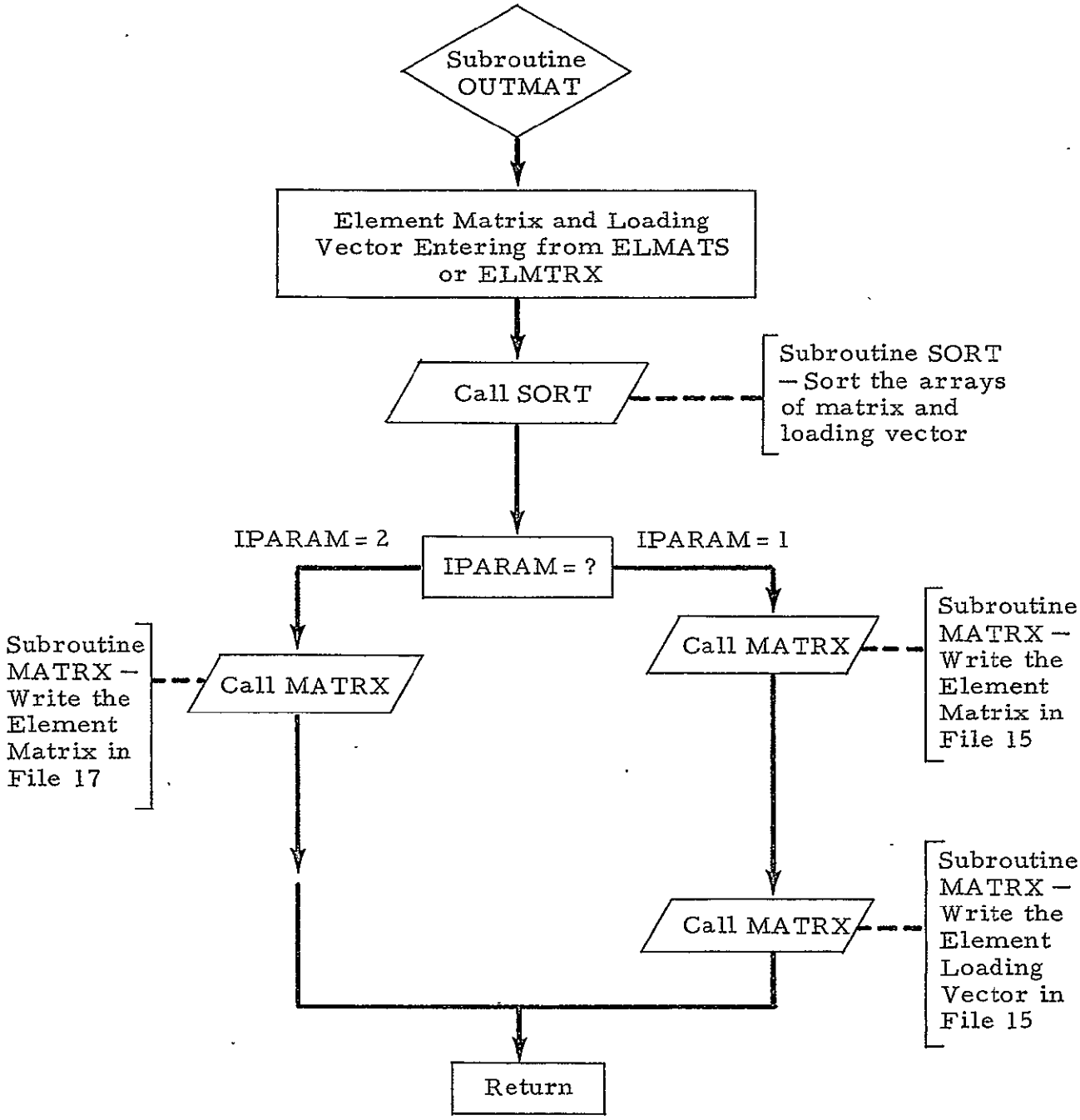


Fig. 2-9 - Flow Chart for OUTMAT

2.14 SUBROUTINES SFCALL, SFCALQ, AND SFCALC

This package of subroutines provides the shape functions of three-dimensional isoparametric elements, along with their first-order derivatives with respect to spatial variables. Here

SFCALL [A-101] – Provides shape functions and their derivatives for linear elements

SFCALQ [A-102] – Provides shape functions and their derivatives for quadratic elements

SFCALC [A-103] – Provides shape functions and their derivatives for cubic elements.

2.15 SUBROUTINE SLTEST

This routine [A-108] tests the solution of each time step for the possibilities of

- Rebound
- Sliding
- Penetration.

FORTRAN PROCs PARAM1 and TESTNG are included to provide the sizes of the variables, and the storage of the common blocks.

The criteria for various cases are discussed in Part I of this report. The routine tests the possibility of rebound first using the momentum vector computed each time step. If all nodes in the projectile begin to reverse the direction of velocities normal to the impact surface, the projectile is assumed to rebound. Then the computations in the following time step will exclude the projectile. Otherwise, the runstream proceeds to test the sliding case.

If the impact is not oblique, jump to the third case. If yes, use the criterion discussed in Part I to test the onset of the sliding. If yes, the mesh of the entire CELFE substructure is regenerated, and the original impact surface becomes a moving contact surface and moves along the sliding band (cf. Section 2.10). Otherwise, the runstream proceeds to the third case.

The penetration follows the failure criteria tested in subroutines STRENG or FRCRIT. Thus, if there is no failure, the runstream returns. Otherwise, test the possibilities of degeneracy of certain elements due to free surface motion, and/or other means. If there is a degenerate element, then regenerate the mesh and return (Fig. 2-10).

2.16 MISCELLANEOUS SUBPROGRAMS

There are miscellaneous subroutines for the needs of various computations:

INVDET [A-79] — Computes the determinant and its inverse for a matrix

MATMUL [A-80] — Provides the multiplication of matrices

MESHUP [A-90] — Update the coordinate system from computed displacement vector and smooth out the spurious oscillation when the mesh is too coarse

SORT [A-119] — Sorts the global arrays of submatrices

STIFFG [A-120] — Computes the material stiffness coefficients for anisotropic materials by input data cards

STIFF [A-123] — Computes the material stiffness coefficients for unidirectional composite materials.

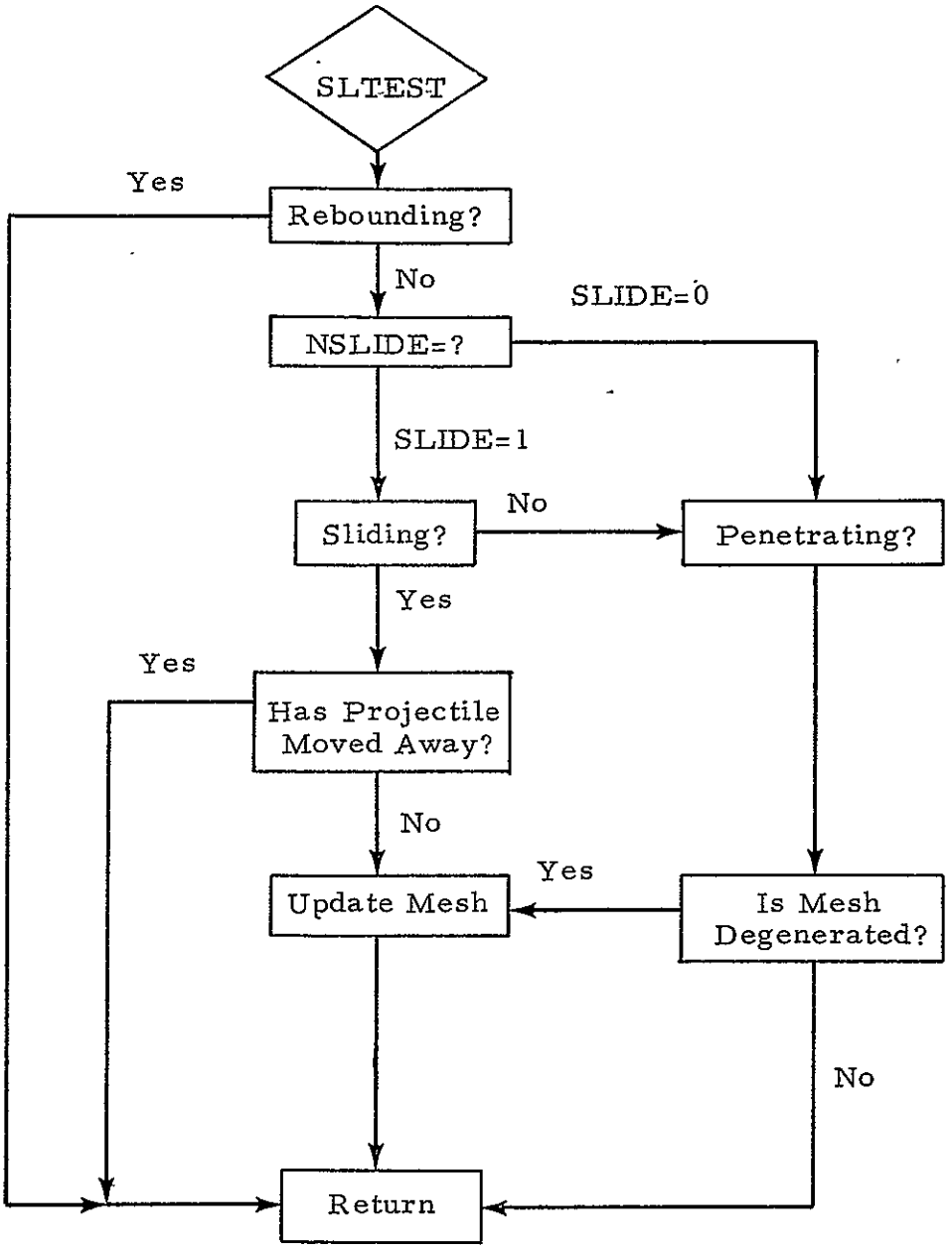


Fig. 2-10 - Flow Chart for SLTEST

3. CELFE/NASTRAN INTERFACING PROCEDURE

The theoretical background and formulation of the CELFE/NASTRAN interfacing procedure are given in Part I of this report. In this section, related software aspects of the procedure are discussed, together with corrections and modifications made in the NASTRAN program, for making the procedure operational.

3.1 GENERAL DISCUSSIONS

The NASTRAN program is interfaced with the CELFE program to serve the following purposes:

- Utilizing complete NASTRAN capabilities for tackling high velocity impact problems, and
- Resolving any limitation of the size of the CELFE program.

These goals can be accomplished by treating the dynamic behavior in the vicinity of the impact point simulated by CELFE as a substructure of the entire NASTRAN model, and utilizing the NASTRAN program to perform all large matrix manipulation requirements.

3.1.1 Geometric Considerations

In the CELFE program, the three-dimensional serendipity family of elements is used (cf. Section 2-10). The interfacing procedure permits the three-dimensional mesh of a CELFE substructure to be connected with any type of finite element mesh in the NASTRAN structure via suitably chosen linking elements available in NASTRAN.

An illustration is shown in Fig. 3-1a for a blade subjected to high velocity impact of a projectile near the leading edge: the CELFE zone containing the

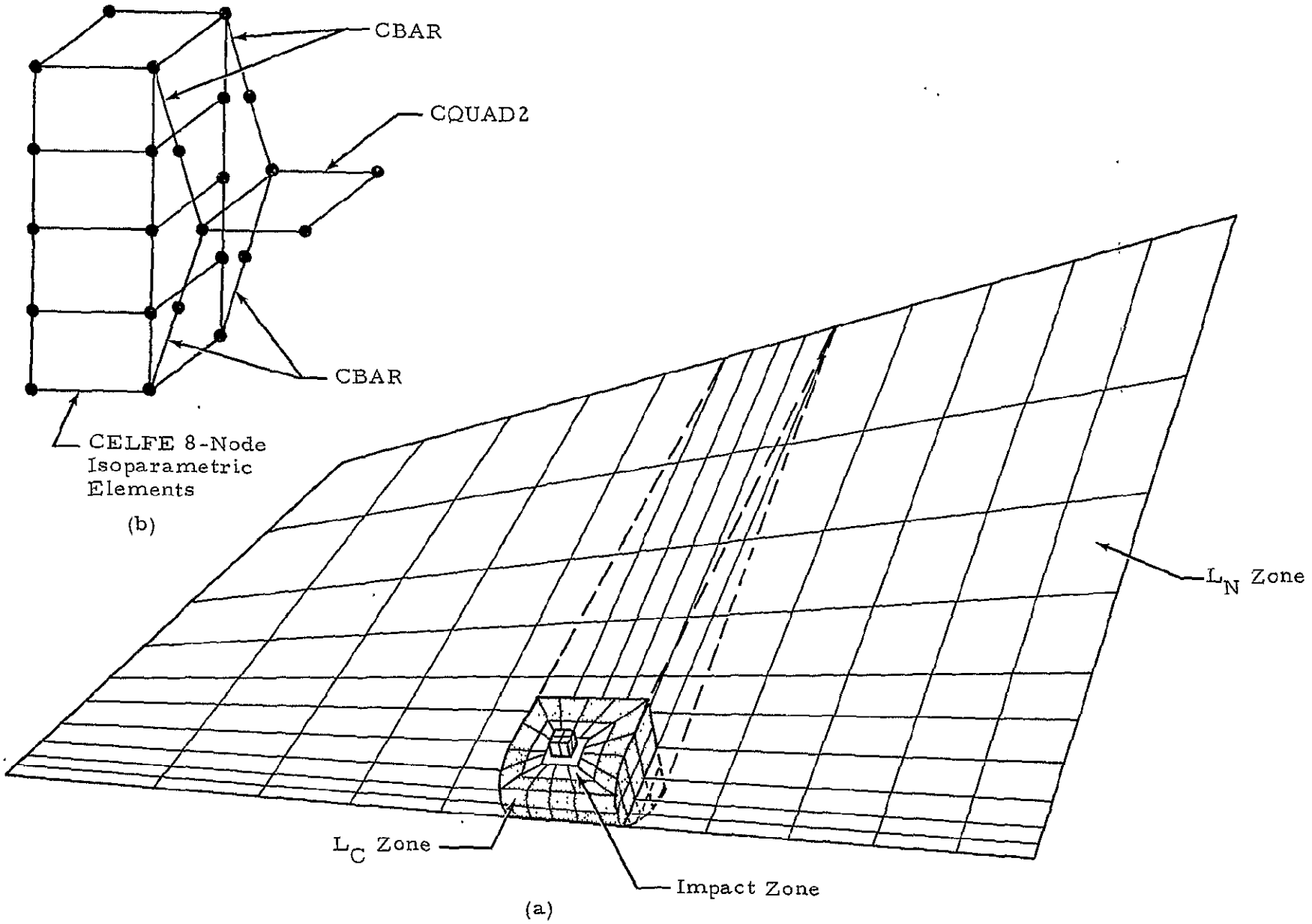


Fig. 3-1 - An Illustration of CELFE/NASTRAN Mesh – A Blade Impacted by a Projectile Near the Leading Edge

projectile and the vicinity of the impact point is discretized by three-dimensional linear quadrilateral mesh; in the remaining region of the blade, NASTRAN plate elements are used. In order to join the two regions, typical CBAR elements are used at the interface nodes between the L_C zone and L_N zone as shown in Fig. 3-1b. In Fig. 3-2, a wedge element family is used as the linking elements.

3.1.2 Interfacing Procedure

The CELFE in-core execution and CELFE/NASTRAN run are controlled by a flag NASTRN described in the previous section. For the in-core run, where NASTRN=0, the element stiffness matrices of all variables are assembled and solved inside CELFE. For the CELFE/NASTRAN interfacing procedure, where NASTRN=1, the element matrices computed by CELFE for "primary variables" as well as the "coupling variables" in CELFE substructure are transmitted to NASTRAN. These matrices will be assembled and solved in NASTRAN for each iteration in each time step. The solution will be returned to CELFE to continue the execution. A general flow diagram for CELFE/NASTRAN runstream is illustrated in Fig. 3-3 (cf. also Fig. 2-2).

Five data files were prepared to contain the input data for CELFE and NASTRAN, and control the runstream during program execution. These files are set-up as "permanent" files. Only the input data in the files need to be changed by the user for different problems. In addition, several scratch files will be generated within those permanent files to store the temporary data during the execution. All these files are compiled in Table 3-1.

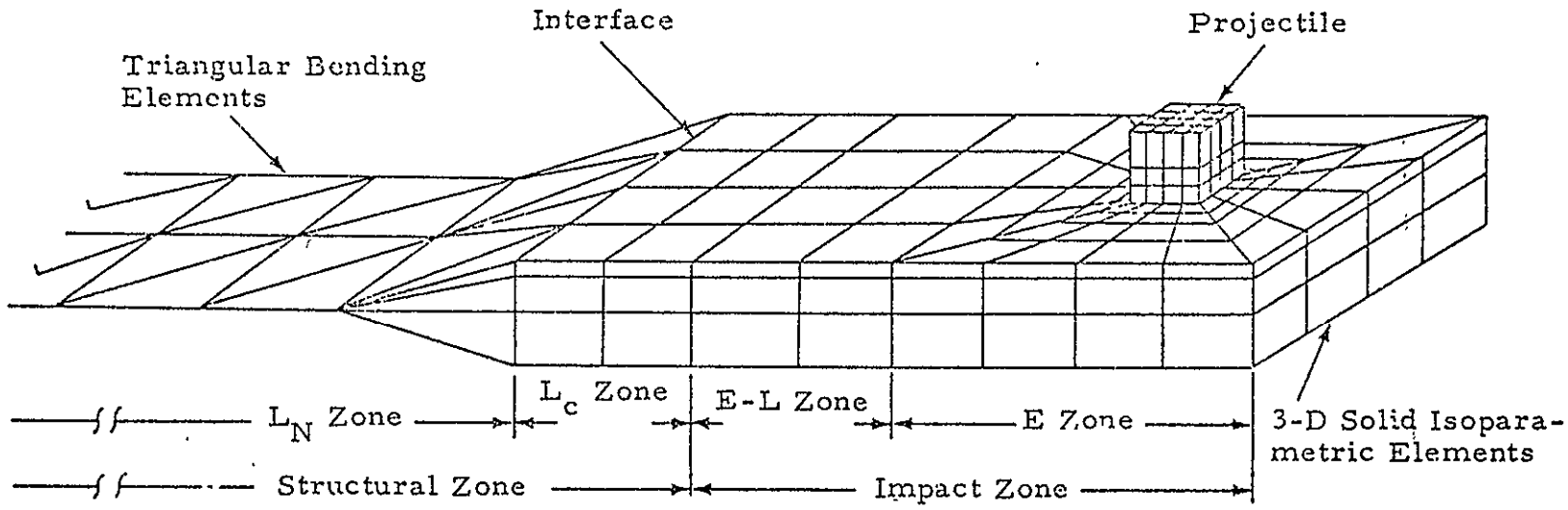


Fig. 3-2 - Typical Finite Element Sketch in Global Analysis of High Velocity Impact

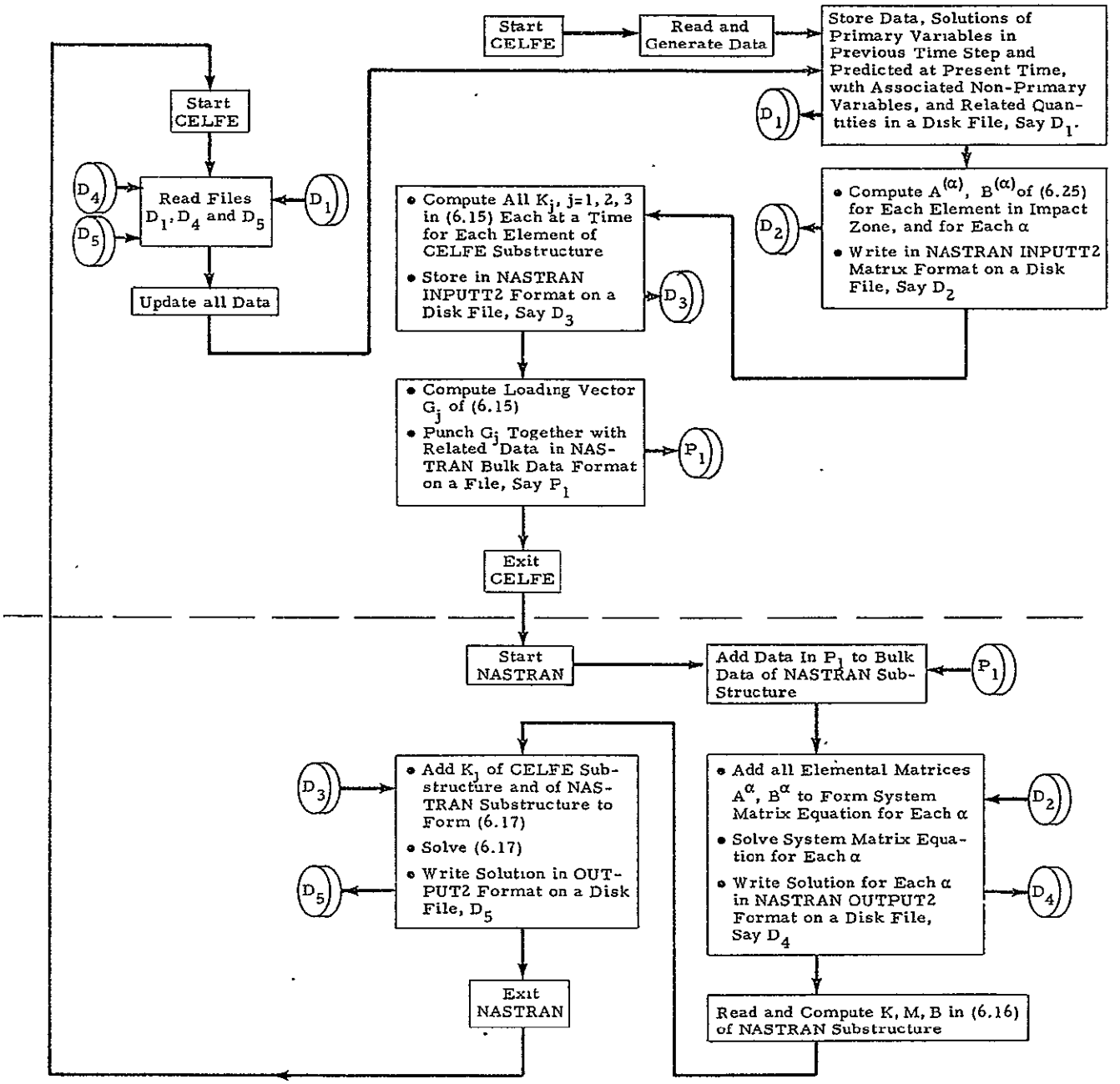


Fig. 3-3 - Interfacing Procedure for CELFE/NASTRAN

Table 3-1

LISTING OF FILES USED IN CELFE/NASTRAN PROCEDURE

File Name	Type	Purpose
INPT (14)	Disk	Output solution for primary variables computed by NASTRAN in OUTPUT 2 format.
INP1 (15)	Disk	Input element matrices and load vectors for primary variables to NASTRAN from CELFE in INPUTT2 format.
INP2 (16)	Disk	Store variables and constants in CELFE for restart.
INP3 (17)	Disk	Input element stiffness matrices of CELFE substructure to NASTRAN in INPUTT 2 format.
LEE	Program	Store the complete CELFE program: source, relocalable and absolute elements.
LOOP	Data (SDF)	Loop NASTRAN and CELFE executions after fresh start.
NASTRAN	Program	Store 14 NASTRAN absolute elements
NAS1	Data (SDF)	Execute NASTRAN portion after fresh start.
OBJ	Program	Store NASTRAN relocalable and selected source elements.
PCH1	Disk	Store break-point-punch data in PCH2, DAREA, TIC, TSTED for NASTRAN.
PCH2 (18)	Data (SDF)	Store punched GRID cards of CELFE model for NASTRAN.
PCH3 (19)	Disk	Store updated GRID computed from CELFE.
PLT1 (12)	Disk	Output solution vector UVN from NASTRAN in OUTPUT2 format.
RUN1	Data (SDF)	Illustrate Exec 8 control for attaching the required files and executing the example problem.
START	Data (SDF)	Assign disk files and execute CELFE for fresh start.

3.2 UTILIZATION OF NASTRAN PROGRAM

3.2.1 Specific Tasks

The specific tasks performed by NASTRAN are:

- Sum "primary variables" calculated by CELFE for each CELFE isoparametric element and solve the matrix equations, $Ax = B$.
- Sum the CELFE element stiffness matrices with the NASTRAN system stiffness matrix.

$$\Sigma_i \, K1_i + K2_i + K3_i + KGGX = KGG$$

- Solve the equation of motion for each CELFE iteration.

$$Kx + B\dot{x} + M\ddot{x} = P(t)$$

To accomplish these tasks and to communicate information between CELFE and NASTRAN, several I/O routines were installed in CELFE which read and write matrices in an acceptable NASTRAN format. The standard NASTRAN Rigid Format 9 was then altered to perform the required matrix manipulation and output results to the CELFE program. The runstream sequence can be described as in Fig. 3-4 (cf. also Fig. 3-3).

3.2.2 File Allocation for NASTRAN

When communicating between a FORTRAN program (CELFE) and NASTRAN several restrictions and procedures must be followed.

- Data blocks (either tables or matrices) can be input or output from NASTRAN via the INPUTT2 and OUTPUT2 modules. These modules operate normally using FORTRAN unformatted read and write instructions.
- The NASTRAN rules for writing the FORTRAN records must be rigorously adhered to. Therefore, a set of subroutines has been prepared which when added to the user FORTRAN program, will read or write the FORTRAN records in the required NASTRAN form.

CELFE (Iteration 1, 3, 5...)

1. Read Model Data — File 'START' (Iter. 1 only)
2. Calc Mesh and Elem. Coeff. (Iter. 1 only)
3. Store Variables & Const. for Restart — File 16
4. Calculate and Output NASTRAN Cards — Read File PCH2
   DAREA, GRID, TIC, TSTEP — File PCH1
   GRID — File PCH3.
5. Calculate "Primary Variables" A&B
6. Output A&B — File 15
7. Calculate Element Stiffness Matrices
8. Output K1, K2 & K3 — File 17

NASTRAN (All Iterations)

1. Add File PCH1 to Bulk Data
2. Initialize Files 14 & 15
3. Read A, B — File 15
4. Sum Element Matrices A, B
5. Solve X and Output — File 14
6. Repeat 3, 4, 5 for 11 Parameters
7. Calculate NASTRAN Model, KGGX
8. Read CELFE Stiffness K1, K2, K3 — File 17
9. Assemble CELFE/NASTRAN Stiffness
10. Apply System Constraints
11. Solve Transient Equations for Given Time Step
12. Output Solution Vector UVN — File 12
13. Print NASTRAN Solution

CELFE (Iteration 2, 4, 6...)

1. Load Stored Constants — File 16
2. Jump to Inner Loop
3. Load Solution Vector (Disp., Velo.) — File 12
4. Update Grid Locations, TIC, DAREA and TSTEP —
   File PCH1
   GRID — File PCH3
5. Update "Primary Variables" A&B
6. Output A&B — File 15
7. Calculate Element Stiffness Matrices
8. Output K1, K2 & K3 — File 17

Fig. 3-4 - CELFE/NASTRAN Runstream Description

- The simplest and least costly method for temporarily storing the data is through the use of disk files rather than tape. Since all but a few files used by NASTRAN are restricted to NASTRAN read and write, the user must determine beforehand which files are available in NASTRAN for FORTRAN formatted files. These files are highly dependent on the particular version of NASTRAN to be used. However, all versions permit the PLT1 (12) file to be used, if plotter number 1, is not being used. Additional files can be obtained by checking the total number of NTRAN files used. The DIAG 2 option will print the file allocation table (FIAT). The NTRAN files may be counted directly from this table. The MAXFILES parameter on the NASTRAN card can then be decreased by a number (N), where N is the number of FORTRAN files available for use by INPUTT2 and OUTPUT2. The available files begin where the NTRAN files stop; e.g., if the program had a maximum of 30 NTRAN files, and the last file was 46, the user can use file 46 for a FORTRAN file by setting MAXFILES = 29 (30-N, here N=1) or files 44, 45 and 46 by setting MAXFILES = 27. The Lewis Research Center's version of NASTRAN permits files 14 through 17 to be used for FORTRAN written files without exercising any special options. For that reason disk files 12, 14, 15, 16 and 17 were utilized to communicate between CELFE and NASTRAN (cf. Table 3-1).

3.2.3 Program Changes for NASTRAN

In order to interface NASTRAN with CELFE, some subprograms in NASTRAN must be modified. The following is a summary of the program changes made to the NASTRAN program.

- Modify modules INPUTT2 and OUTPUT2 to use formatted FORTRAN read and write statements. Format used: (22A6). Reason – The formatted read and write was selected to minimize disk or tape space when passing a large number of matrices between CELFE and NASTRAN. When the standard nonformatted write is used, a minimum of one 28 word sector is used for each write statement. The small elemental matrices generated by CELFE may require only 8 or less non-zero terms per write statement (matrix column). The formatted write therefore reduces the disk or tape requirement by a factor of from 3 to 14, since a null column requires 2 words and a full column 9.
- The modules INPUTT2, OUTPUT2 and MATPRN were added to NASTRAN LINK7. This was accomplished by changing the subroutine XSEMO7 and data block routine XBSBD. The map symbolic was modified for LINK7, and LINK1, 2, 8 and 14 were recollected to include the changes in the block data routines. Reason – When the elemental matrices are being assembled, a DMAP loop is used to input, add, input, add, etc., solve the matrix equation and output the result. In the standard NASTRAN program, link switching occurs in this loop. INPUTT2 is in LINK2,

ADD in LINK7, SOLVE in LINK7, MATPRN in LINK8 and OUTPUT2 in LINK14. When a LINK change occurs on the Univac 1100 Exec 8 system, FORTRAN files are automatically rewound. This rewind feature would utilize excessive CPU time as well as physically wear out a tape. In the example problem, the inner loop of the DMAP alter is repeated for 33 elements and 11 parameters (33 x 11) or a total of 363 times for each iteration by CELFE. By placing all modules used in the DAMP loop into LINK7, the file retains its position at the end of each INPUTT2 read, and each elemental matrix can be read sequentially. The same is true for the solution vector [X]. When written to a disk or tape file, each of the 11 primary variable solution vectors are written sequentially on the output file.

3.2.4 Transient Analysis in NASTRAN

Rigid format 9, "Direct Transient Analysis," was selected to perform the NASTRAN portion of the analysis [3]. The basic rigid format provides all of the module operations required for a normal finite element transient analysis.

A DMAP alter program is added to Rigid Format 9 to read elemental matrices generated by CELFE, sum the matrices and solve the matrix equation AX = B for the "primary variables" and to add the CELFE stiffness matrices (K1, K2, and K3) to the NASTRAN system stiffness matrix (KGGX). Alters are also used to output to disk, the solution vector X for the "primary variables" and the NASTRAN data block UVN which contains the displacement, velocity and acceleration vectors from each time step solved by the NASTRAN transient response analysis.

3.3 FINITE ELEMENT MODELING — AN EXAMPLE

The general discussions on the finite element mesh arrangement for CELFE/NASTRAN have been presented in Section 3.1 (e.g., Fig. 3-1). In this subsection, a specific procedure for the finite element modeling will be presented by the following example.

3.3.1 Normal Impact of a Projectile onto a Taped Composite Plate

Consider a taped rectangular plate subjected to high velocity impact by a foreign object. Assume that the projectile hits on the center of the plate surface normally as shwon in Fig. 3-5. Thus, due to the symmetry of the problem, it suffices to consider only a quadrant of the plate, say, the first quadrant.

The finite element model (for the quarter of the plate together with the quarter of the projectile) is divided into two substructures. The first substructure (CELFE model) is made up of CELFE 8-node isoparametric solid elements and the second substructure (NASTRAN model) is made up of any of the finite elements contained in the NASTRAN element library [2]. If the NASTRAN model utilizes bars and/or plate elements, special modeling techniques must be employed to ensure inter-boundary displacement compatibility.

3.3.2 CELFE Modeling Considerations

The finite element modeling of the CELFE substructure must follow the procedure described in Section 2.(specifically, Section 2.10 and Figs. 2-6 and 2-7). To prepare the modeling, it is important to recall the following:

- The CELFE substructure consists of the impact zone and the L_c zone (Fig. 2-5). The impact zone, in turn, is composed of the E zone and E-L zone whose sizes change with respect to time [1].
- For normal impact, the size of the impact zone should a priori be chosen large enough to cover the terminated failure zone. Naturally, this can only be done by an estimation (or a guess). If too small a size is chosen the accuracy of the result will be affected; too large a size will consume computation time.
- For oblique impact, the configuration of the CELFE model should be able to allow the possible sliding taking place within the substructure as remarked in Section 2.10 and illustrated in Fig. 2-7.
- The finite element mesh for the CELFE substructure should be arranged according to the READing sequence in subroutine GEOMTR (Section 2.10), namely, mesh numbering should start from the impact zone, and then the L_c zone. The coordinates and node numbering are illustrated in Fig. 2-6.

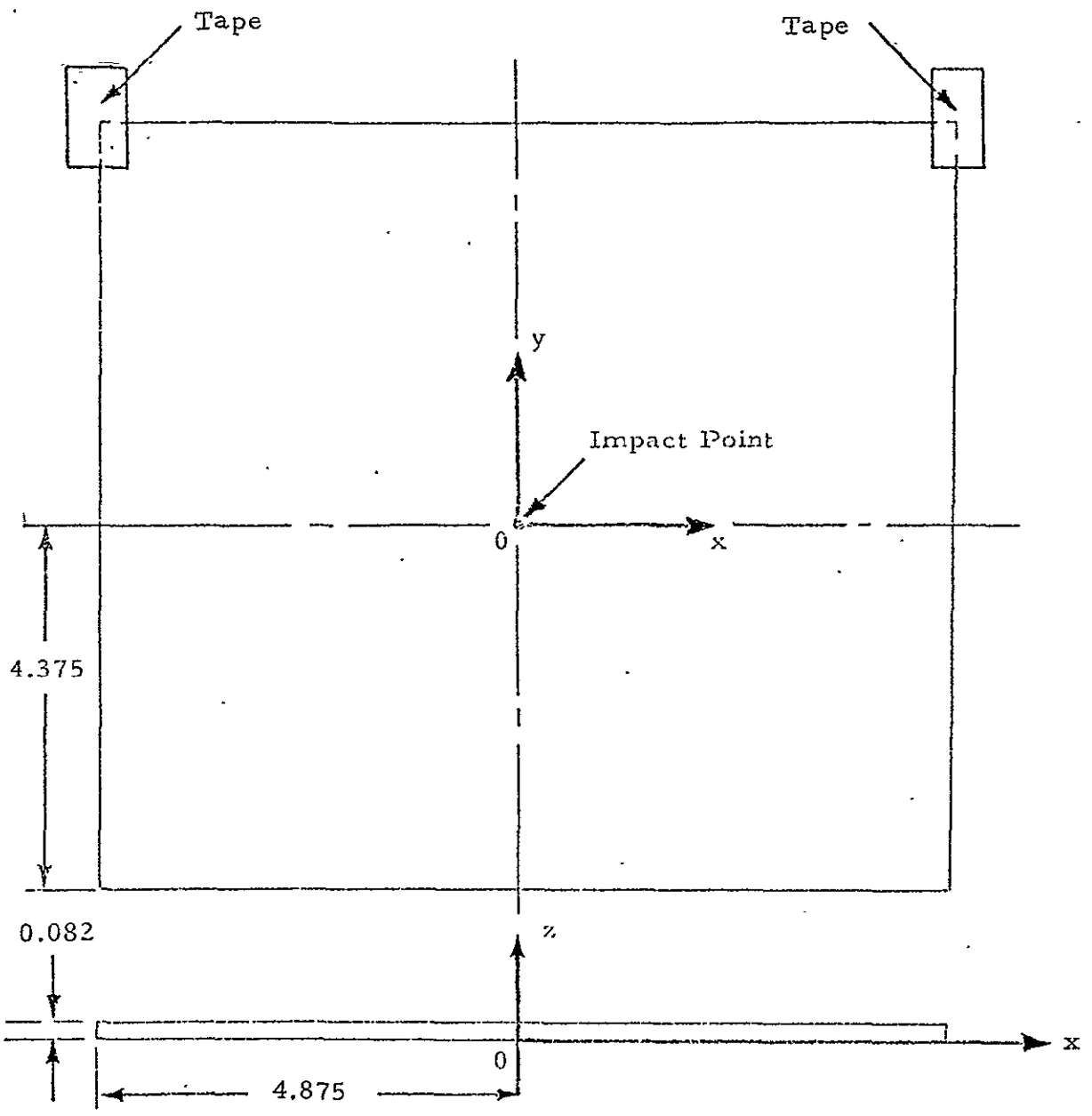


Fig. 3-5 - Geometric Sketch of a Three-Dimensional Demonstration Problem

The CELFE mesh can now readily be modeled and is shown in Fig. 3-6. The zoning is as follows:

- The impact zone consists of nodes 1 through 61 (or elements 1 through 25)*.
- The transition (E-L) zone starts from node 47 to node 76 (i.e., element 26 to element 33).
- The L_c zone selected in this example is empty.

Thus, in this example, the CELFE substructure is connected with NASTRAN substructure through nodes 62 through 76 in the E-L zone.

3.3.3 NASTRAN Modeling Considerations

Figure 3-7 illustrates the NASTRAN model. The process interconnecting the CELFE model and NASTRAN model is depicted in Fig. 3-8. To prepare the NASTRAN bulk data, the following procedure should be utilized:

- All grid points contained in the CELFE model (e.g., 1 through 76), including boundary points, must be defined in the NASTRAN bulk data. This is accomplished by adding the RCH1 file (at ADD, P PCH1), after the BEGIN BULK card of the NAS1 file. The cards in file PCH1 have been generated by CELFE and are updated each iteration by adding the displacements calculated in the transient analysis to the current grid point coordinate location. The CELFE program obtaines the previous grid location by reading the PCH2 file. The user must place the original CELFE GRID cards into the PCH2 file (cf. Table 3-1)
- All grid points contained in the NASTRAN model excluding the CELFE/NASTRAN interfacial boundary points (163 through 326 in the example) are placed in file NAS1 by the user.
- CELFE grid points must be internally sequenced before the NASTRAN grid points. This simplifies the partitioning required to combine the two substructures for transient analysis in NASTRAN and for reading and writing data blocks in CELFE.
- All constraints are applied during the NASTRAN analysis (NAS1 file).

*The numbering sequence of the elements must follow the sequence of the nodes.

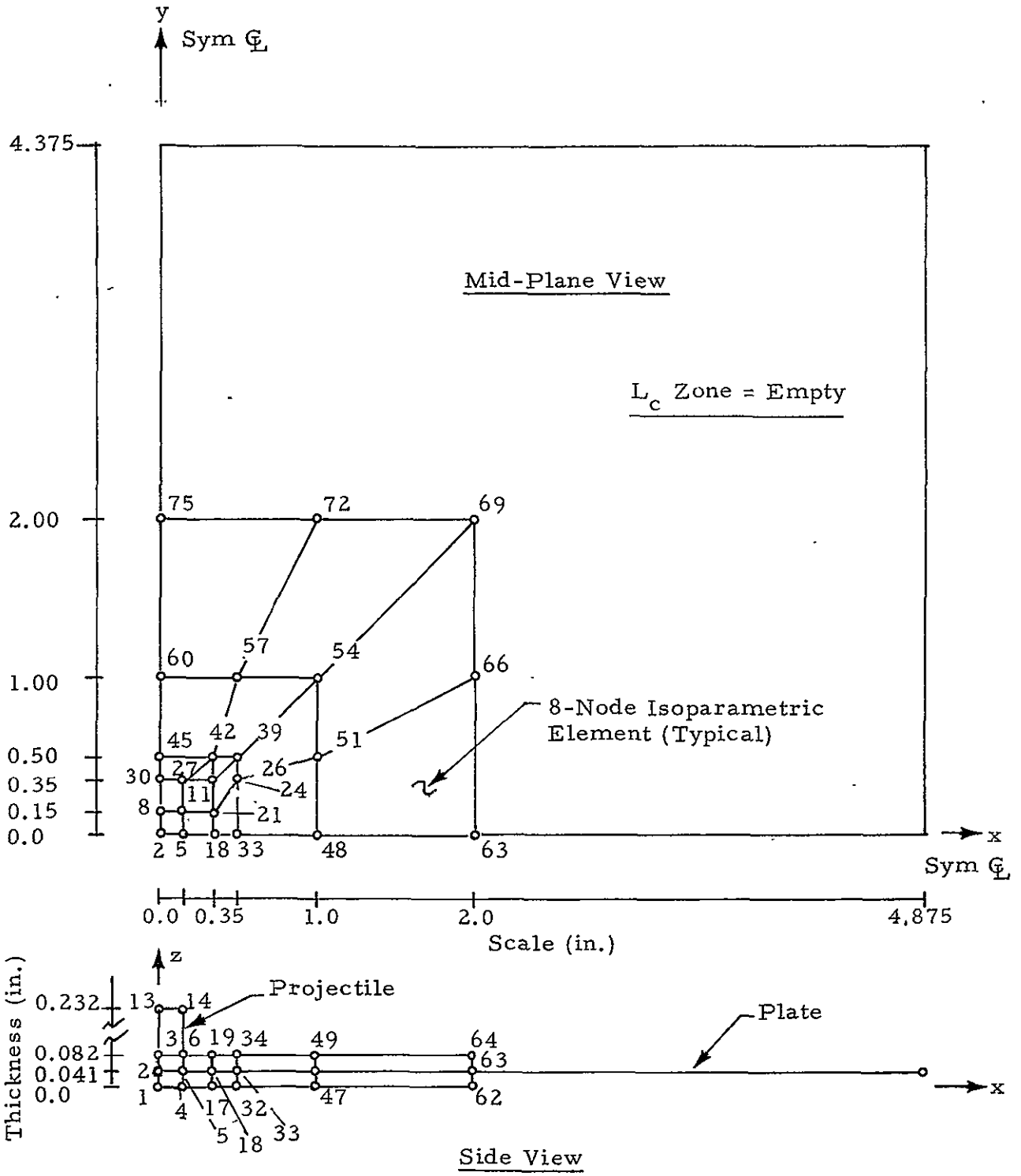


Fig. 3-6 - CELFE Finite Element Model

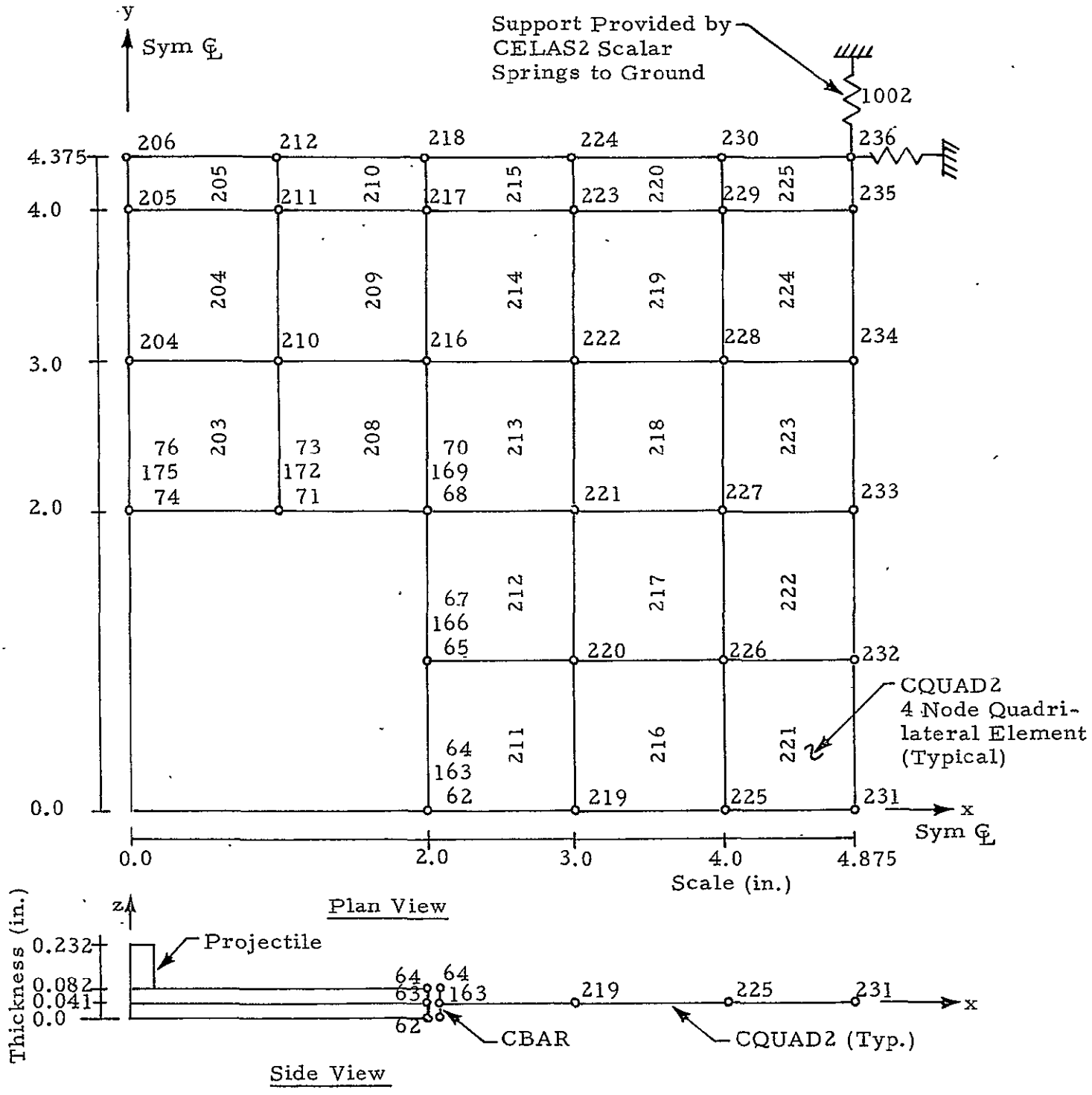


Fig. 3-7 - NASTRAN Finite Element Model

MPC – Equations (typ)

$X_{163} = X_{63}$ $\quad$ $X_{166} = X_{66}$

$Y_{163} = Y_{63}$ $\quad$ $Y_{166} = Y_{66}$

$Z_{163} = Z_{63}$ $\quad$ $Z_{166} = Z_{66}$

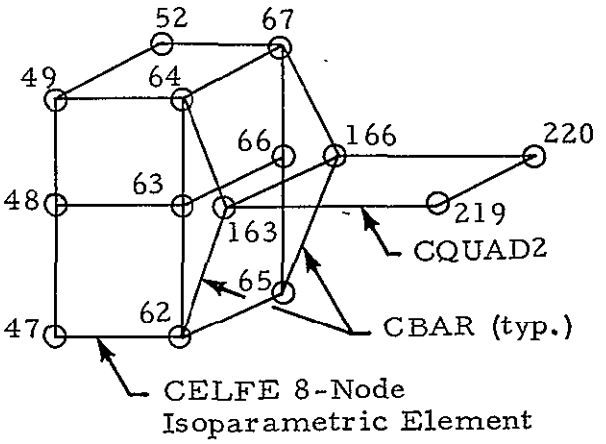


Fig. 3-8 - An Illustration of Connecting CELFE Model to NASTRAN Model

- If multipoint constraints (MPC) are used to ensure displacement compatibility across the interface boundary, the following technique is recommended:

  A NASTRAN grid point has been added to each CELFE mid-point grid lying on the boundary between the two substructures (e.g., 163, 166, 169, 172 and 175). The NASTRAN grid exists at the same location as the CELFE grid (e.g., 63, 66, 69, 72 and 75). The mid-point grids are coupled by MPCs for the three translational degrees of freedom (DOF); rotational displacement is transmitted from the CQUAD2 plate to the CELFE 8-node solid elements by using rigid bars (CBAR with arbitrarily large section properties) connected between the mid-point node (163) and the top and bottom nodes of the solid (64 and 62, respectively).

- Permanent grid point constraints are applied using the GRDSET card and field 8 of the GRID card. These constraints remove the inplane rotation DOF (6) on all CQUAD2 elements and all rotational DOFs (4, 5 and 6) on all CELFE 8-node solid elements.

- Symmetry constraints and supporting constraints are applied using SPC1 cards.

- Loads for the transient analysis are provided on DAREA, TABLED1 and TLOAD1 cards. The CELFE program will automatically place a new set of DAREA cards into file PCH1 for each iteration. The user must provide TLOAD1 and TABLED1 cards which will apply a unit scale factor to the DAREA load.

- The integration time step (TSTEP) is determined by CELFE and placed into the PCH1 file; no user card is required.

- The initial conditions (displacement and velocity) for each iteration are obtained from CELFE which places TIC cards into file PCH1. The values are obtained from the previous solution vector.

- Several PARAM cards must be defined in the bulk data; they are:

  BLOK=0, used to control file positioning by the INPUTT2 module.

  SYM=0, use unsymmetric solution techniques for module SOLVE.

  NOEQU= number of "primary variables" to be solved.

  NUMNP=N-1, where N = number of CELFE elements.

4. EXECUTION OF CELFE/NASTRAN

4.1 RUNSTREAM FOR UNIVAC 1100 SYSTEM

The execution of the CELFE and NASTRAN programs and the transfer of data between these two programs is controlled through the use of "DATA" files containing runstream control cards and user data cards. The options for using the in-core CELFE program and coupled CELFE/NASTRAN are controlled by the flag NASTRN, as described in Section 2, where

NASTRN = 0 for in-core CELFE
NASTRN = 1 for CELFE/NASTRAN

4.1.1 Program Tape

There are five files used to control the runstream and three files used to hold the programs. These files are contained on a delivery tape in the following sequence:

File	Name	Type	Purpose
1	RUN1	DATA (SDF)	Exec 8 control attaching the required files and executing the example problem
2	LEE	Program	CELFE program, source, relocatable and absolute elements
3	NAS1	DATA (SDF)	NASTRAN runstream includes executive, case and bulk data decks
4	START	DATA (SDF)	Assigns disk files and executes CELFE for 1st run
5	LOOP	DATA (SDF)	Exec 8 control cards for looping from NASTRAN to CELFE
6	PCH2	DATA (SDF)	GRID cards for the CELFE model
7	NASTRAN	Program	14 NASTRAN absolute elements
8	OBJ	Program	NASTRAN relocatable and selected source elements.

The complete listing of the program LEE File is given in Appendix A. The modified source elements of NASTRAN contained in the program OBJ File is listed in Appendix B. A complete listing of each Data File contained on the Program Tape is presented in Appendix C (Tables C-1 through C-5).

4.1.2 Data File Descriptions

The Data Files listed in Appendix C are based on the demonstration example given in Section 3.3. The following is a brief description of the cards found in the Data Files:

RUN1 — The RUN1 file is used to start (at START) a CELFE/NASTRAN run. All files required for execution are attached. Two data files START and LOOP are added (at ADD) to execute the programs.

START — The START file provides the runstream and initial data for executing CELFE (cf. Section 4.2 for details)

	Card No.*
Assigns all necessary disk files	1-6
Defines the required "USE" relationships	7-15
Breakpoints the CELFE punch cards to file PCH1	16
Executes the CELFE program (first time only)	17
All CELFE data cards	18-163
Closes Breakpoint of punch cards	164
Breaks the "USE" relationship so NASTRAN can use files 18 and 19	165-166
Update the PCH2 file with new GRID cards from PCH3 file (PCH3 was written by CELFE at the same time card images were punched into PCH1.)	167

NAS1 — The NAS1 file contains the DMAP alter program added to Rigid Format 9 to read the element matrices generated by CELFE, sum the matrices and solve the matrix equations for the primary variables; at the same time, the element stiffness matrices are added from CELFE to solve the stiffness matrix equation for the entire system (cf. Section 4.3 for details).

*The card numbering in this file is based on the demonstration example described in Section 3.3.

LOOP — Provides the necessary control cards for looping between NASTRAN and CELFE

	Card No.
Add NAS1 file to execute NASTRAN	1
Breakpoint PUNCH$ to PCH1	6
Define "USE" relationships	7-8
Execute CELFE for second iteration of inner loop	9-11
Close Breakpoint PUNCH$	12
Break "USE" relationship	13-14
Update the PCH2 file with new GRID cards from PCH3 file	15
Repeat procedure for first iteration of inner loop	17-26

PCH2 — Contains GRID cards for CELFE substructure at time t = 0. The cards are prepared for the NASTRAN program for the NASTRAN program to solve the system stiffness matrix equation.

4.1.3 Basic Considerations for Input and Output

The input data for CELFE and NASTRAN are prepared separately, and are stored in the different files, so that the CELFE in-core execution can be carried out more efficiently.

The CELFE input data are given in the 'START' File, and will be described in detail in Section 4.2.

The NASTRAN input data, on the other hand, are prepared and stored in two separate files:

- The data for the NASTRAN substructure, together with the parameters required for solving the entire problem, are embedded in DMAP alters given in 'NAS1' File. The details will be described in Section 4.3.
- The GRID data for CELFE substructure at t = 0 prepared for NASTRAN is stored in 'PCH2' File. The data cards must be in the NASTRAN GRID format.

The present program is developed for both SI and British units. The unit will be identified by the assigned value of IUNIT:

IUNIT = 1 – SI Unit

IUNIT = 2 – British Unit

The units used for the input and output variables are listed in Table 4-1.

Table 4-1

UNITS FOR INPUT/OUTPUT VARIABLES

Variable	SI (IUNIT = 1)	British (IUNIT = 2)
Length	cm	in.
Velocity	cm/sec	in./sec
Time (ΔT)	μsec	μsec
Density	gm/cm^3	lb/in^3
Pressure, Stresses (and Other Similar Quantities)	megabar	psi
Energy	megabar	$lbf\text{-}in./in^3$

4.2 INPUT DATA FOR CELFE

As aforementioned, the input data for the CELFE program are all stored in 'START' file. For the demonstration example given in Section 3.3, these data are found on the card numbers 18 through 163. As readily described in Section 4.1.2, the remaining cards (card Nos. 1 through 17, and Nos. 164 through 167) in the 'START' file are all control cards. All these control cards should be kept unchanged for all time in utilizing the program.

The complete listing of the 'START' file is given in Table 4-2, and its card images are shown in Fig. 4-1. In what follows, the input data cards will be described sequentially starting from card number 18 of Table 4-2.

Table 4-2

LISTING OF FILE 'START'

```
SYM

@DATA,L START.
DATA 8R1 SL73R1 10/31/78 21:19:15 (0)
      1.  @ASG,T PLT1,F///2000
      2.  @USE 12.,PLT1.
      3.  @ASG,T INPT,F///2000
      4.  @ASG,T INP1,F///2000
      5.  @ASG,T INP2,F///2000
      6.  @ASG,T INP3,F///2000
      7.  @USE 14.,INPT.
      8.  @USE 15.,INP1.
      9.  @USE 16.,INP2.
     10.  @USE 17.,INP3.
     11.  @ASG,A PCH2.
     12.  @USE 18.,PCH2.
     13.  @ASG,T PCH1,F///1000
     14.  @ASG,C PCH3,F///1000
     15.  @USE 19.,PCH3.
     16.  @ERKPT PUNCHS/PCH1
     17.  @XQT CELFE
     18.  +1      CHANGE THIS CARD TO 0 FOR CLEFE RUN ONLY
     19.  0
     20.    HIGH VELOCITY IMPACT --- DEMONSTRATION PROBLEM
     21.     1   1  51   4   1   0   2   8   1   0   0
     22.     4.CD       0.          0.
     23.          0.0        0.0 -7.5600+3   2.0000+0   2.0000+0   8.2000-2
     24.    7.3CC0-2  5.1200-2         0.0   2.7027+3        0.0   6.0000+3   1.0000+0
     25.    76  33  13   3  16   3  62  30  77  34  76  33 648 103
     26.    20  20   0  15  15  10  10  10  10  10  10  10  10  10  20  20   0
     27.     1   4  10   7   2   5  11   8
     28.     2   5  11   8   3   6  12   9
     29.     3   6  12   9  13  14  16  15
     30.     4  17  20  10   5  18  21  11
     31.     5  18  21  11   6  19  22  12
     32.    10  20  23  26  11  21  24  27
     33.    11  21  24  27  12  22  25  28
     34.     7  10  26  29   8  11  27  30
     35.     8  11  27  30   9  12  28  31
     36.    17  32  35  20  18  33  36  21
     37.    18  33  36  21  19  34  37  22
     38.    20  35  38  23  21  36  39  24
     39.    21  36  39  24  22  37  40  25
     40.    26  23  38  41  27  24  39  42
     41.    27  24  39  42  28  25  40  43
     42.    29  26  41  44  30  27  42  45
     43.    30  27  42  45  31  28  43  46
     44.    32  47  50  35  33  48  51  36
     45.    33  48  51  36  34  49  52  37
     46.    35  50  53  38  36  51  54  39
     47.    36  51  54  39  37  52  55  40
     48.    41  38  53  56  42  39  54  57
     49.    42  39  54  57  43  40  55  58
     50.    44  41  56  59  45  42  57  60
     51.    45  42  57  60  46  43  58  61
     52.    47  62  65  50  48  63  66  51
     53.    48  63  66  51  49  64  67  52
     54.    50  65  68  53  51  66  69  54
```

(Continued)

Table 4-2 (Continued)

```
SYM
55.     51  66  69  54  52  67  70  55
56.     56  53  68  71  57  54  69  72
57.     57  54  69  72  58  55  70  73
58.     59  56  71  74  60  57  72  75
59.     60  57  72  75  61  58  73  76
60.   GRID   1                20.      0.      0.
61.   GRID   2                10.      0.      0.041
62.   GRID   3                10.      0.      0.082
63.   GRID   4                20.15    0.      0.
64.   GRID   5                10.15    0.      0.041
65.   GRID   6                10.15    0.      0.082
66.   GRID   7                20.      0.15    0.
67.   GRID   8                10.      0.15    0.041
68.   GRID   9                10.      0.15    0.082
69.   GRID  10                20.15    0.15    0.
70.   GRID  11                10.15    0.15    0.041
71.   GRID  12                10.15    0.15    0.082
72.   GRID  13                10.      0.      0.232
73.   GRID  14                10.15    0.      0.232
74.   GRID  15                10.      0.15    0.232
75.   GRID  16                10.15    0.15    0.232
76.   GRID  17                20.35    0.      0.
77.   GRID  18                10.35    0.      0.041
78.   GRID  19                10.35    0.      0.082
79.   GRID  20                20.35    0.15    0.
80.   GRID  21                10.35    0.15    0.041
81.   GRID  22                10.35    0.15    0.082
82.   GRID  23                20.35    0.35    0.
83.   GRID  24                10.35    0.35    0.041
84.   GRID  25                10.35    0.35    0.082
85.   GRID  26                20.15    0.35    0.
86.   GRID  27                10.15    0.35    0.041
87.   GRID  28                10.15    0.35    0.082
88.   GRID  29                20.      0.35    0.
89.   GRID  30                10.      0.35    0.041
90.   GRID  31                10.      0.35    0.082
91.   GRID  32                20.5     0.      0.
92.   GRID  33                10.5     0.      0.041
93.   GRID  34                10.5     0.      0.082
94.   GRID  35                20.5     0.35    0.
95.   GRID  36                10.5     0.35    0.041
96.   GRID  37                10.5     0.35    0.082
97.   GRID  38                20.5     0.5     0.
98.   GRID  39                10.5     0.5     0.041
99.   GRID  40                10.5     0.5     0.082
100.  GRID  41                20.35    0.5     0.
101.  GRID  42                10.35    0.5     0.041
102.  GRID  43                10.35    0.5     0.082
103.  GRID  44                20.      0.5     0.
104.  GRID  45                10.      0.5     0.041
105.  GRID  46                10.      0.5     0.082
106.  GRID  47                21.      0.      0.
107.  GRID  48                11.      0.      0.041
108.  GRID  49                11.      0.      0.082
109.  GRID  50                21.      0.5     0.
110.  GRID  51                11.      0.5     0.041
```

Table 4-2 (Concluded)

```
      SYM
  111.      GRID     52                11.       0.5       0.082
  112.      GRID     53                21.       1.        0.
  113.      GRID     54                11.       1.        0.041
  114.      GRID     55                11.       1.        0.082
  115.      GRID     56                20.5      1.        0.
  116.      GRID     57                10.5      1.        0.041
  117.      GRID     58                10.5      1.        0.082
  118.      GRID     59                20.       1.        0.
  119.      GRID     60                10.       1.        0.041
  120.      GRID     61                10.       1.        0.082
  121.      GRID     62                22.       0.        0.
  122.      GRID     63                22.       0.        .041
  123.      GRID     64                22.       0.        .082
  124.      GRID     65                22.       1.        0.
  125.      GRID     66                22.       1.        .041
  126.      GRID     67                22.       1.        .082
  127.      GRID     68                22.       2.        0.
  128.      GRID     69                22.       2.        .041
  129.      GRID     70                22.       2.        .082
  130.      GRID     71                21.       2.        0.
  131.      GRID     72                21.       2.        .041
  132.      GRID     73                21.       2.        .082
  133.      GRID     74                20.       2.        0.
  134.      GRID     75                20.       2.        .041
  135.      GRID     76                20.       2.        .082
  136.        1   4   6   7   9  10  12  13  14  15  16  17  19  20  22  23  25  26  28  29
  137.       31  32  34  35  37  38  40  41  43  44  46  47  49  50  52  53  55  56  58  59
  138.       61  62  64  65  67  68  70  71  73  74  76
  139.        3   6   9  12
  140.      8.1000+3  9.1000+3  8.9000+3  9.1000+3  0.1990+6  9.1000+3  8.9000+3  9.1000+3
  141.      8.1000+3
  142.     17.9000+3  9.1000+3  8.9000+3  9.1000+3  0.2320+6  9.1000+3  8.9000+3  9.1000+3
  143.     17.9000+3
  144.      3.1500+6  0.7800+6  0.6000+6  0.7800+6 29.2000+6  0.7800+6  0.6000+6  0.7800+6
  145.      3.1500+6
  146.        0.00      0.17      0.53      0.17      0.00      0.17      0.53      0.17
  147.        0.00
  148.        1   2   3   7   8   9  13  15  29  30  31  44  45  46  59  60  61  74  75  76
  149.        1   2   3   4   5   6  13  14  17  18  19  32  33  34  47  48  49  62  63  64
  150.       62  63  64  65  66  67  68  69  70  71  72  73  74  75  76
  151.       62  63  64  65  66  67  68  69  70  71  72  73  74  75  76
  152.       62  64  65  67  68  70  71  73  74  76
  153.       62  64  65  67  68  70  71  73  74  76
  154.       62  64  65  67  68  70  71  73  74  76
  155.       62  64  65  67  68  70  71  73  74  76
  156.       62  64  65  67  68  70  71  73  74  76
  157.       62  64  65  67  68  70  71  73  74  76
  158.       62  64  65  67  68  70  71  73  74  76
  159.       62  64  65  67  68  70  71  73  74  76
  160.       62  64  65  67  68  70  71  73  74  76
  161.        1   2   3   7   8   9  13  15  29  30  31  44  45  46  59  60  61  74  75  76
  162.        1   2   3   4   5   6  13  14  17  18  19  32  33  34  47  48  49  62  63  64
  163.        0.001     0.001     0.002     0.002     0.002     0.002     0.005     0.005
  164.    $BRKPT PUNCH$
  165.    $FREE,B 18.
  166.    $FREE,B 19.

  167.    $COPY PCH3.,PCH2.
END DATA.
```

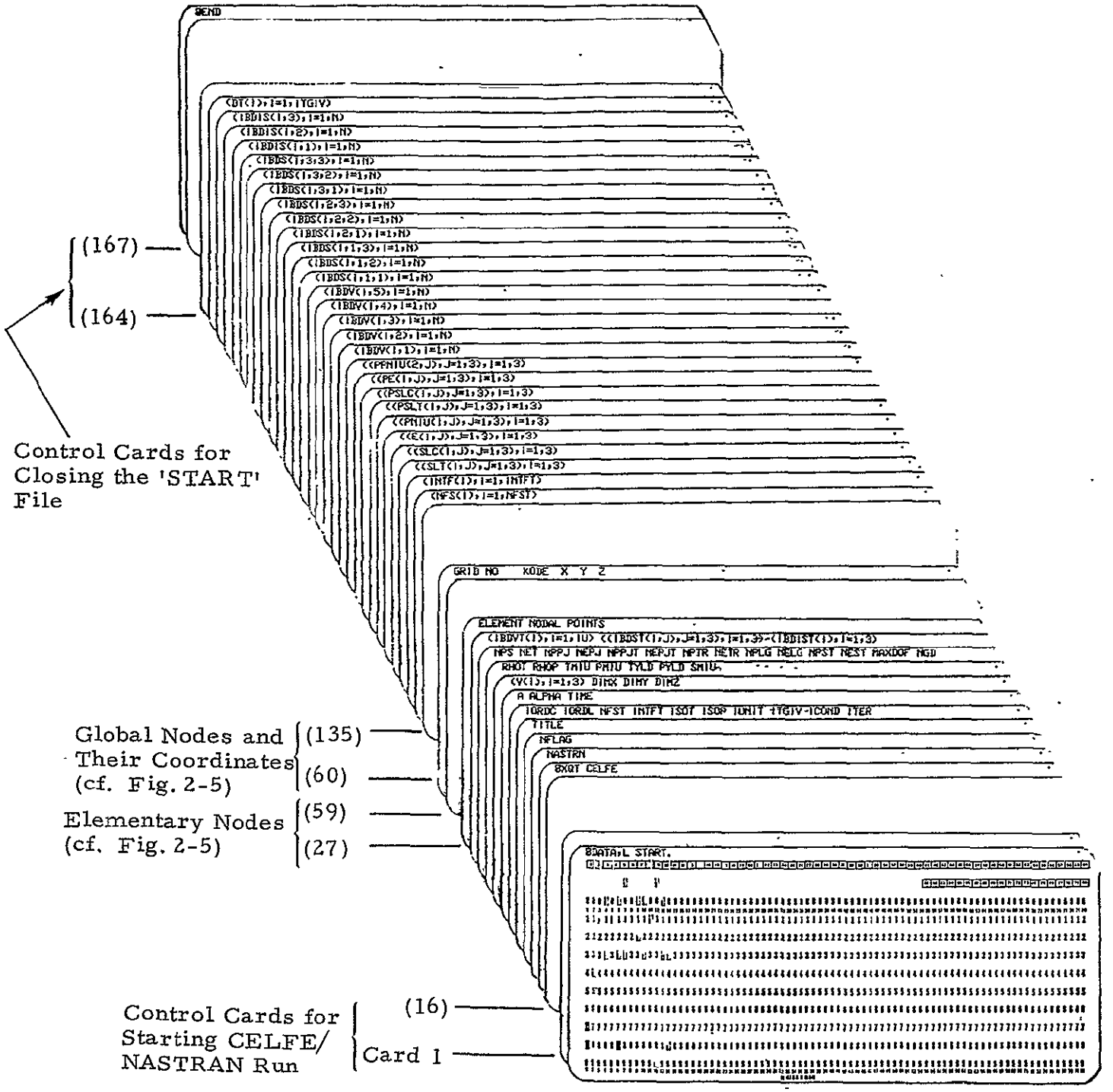


Fig. 4-1 - Input Data Cards for CELFE Stored in 'START' File

4.2.1 CELFE/NASTRAN Execution Control Flag — NASTRN Card

Function: Defines the option of the execution —

NASTRN = 0 for in-core CELFE

NASTRN = +1 for CELFE/NASTRAN

Format: (I2)

Example: (Card No. 18)

+1

*The number in the bracket refers to the card numbers in Table 4-2 for the given example.

4.2.2 Iteration Loop Control Flag – NFLAG Card

Function: Controls the iteration loops INNER and ITER.

Format: (I2)

Example: (Card 19)

0

Remarks: (1) NFLAG is always 0 in 'START' file.

4.2.3 Problem Control — Title Card

Function: Defines the title of the problem.

Format: (12A6)

Example: (Card 20)

```
HIGH VELOCITY IMPACT -- DEMONSTRATION PROBLEM
```

4.2.4 Problem Control Parameters — Integer Parameters Card

Function: Defines the geometric, time and material parameters for a given problem.

Format: (11I4)

Variable	Column	Format	Description
IORDC	1-4	I4	Order of the element in the impact zone
IORDL	5-8	I4	Order of the element in L_c zone
NFST	9-12	I4	Total number of the free surface nodes in CELFE substructure at t = 0
INTFT	13-16	I4	Total number of the impact nodes at t = 0
ISOT	17-20	I4	Target material: ≥ 0 (see Remark 1)
ISOP	21-24	I4	Projectile material: $\geq$ (see Remark 1)
IUNIT	25-28	I4	Unit of input data: = 1 Metric unit = 2 British unit (see Table 4-1)
ITGIV	29-32	I4	Total time steps assigned for a specific run
ICOND	33-36	I4	Parameter for generating initial data: = 0 read data from input data card (default) = 1 generate data from main program
ITER	37-40	I4	Number of time steps at the start of a specific run: Fresh start = 0 (default) Restart > 0 Stop = ITGIV
NSLIDE	41-44	I4	= 0 normal impact (default) = 1 oblique impact

Example: (Card 21)

1	1	51	4	1	0	2	8	1	0

Remarks: (1) Both ISOT and ISOP may be assigned to the following values:
= 0 isotropic materials
= 1 unidirectional composites
> 1 other types of anisotropic materials yet to be formulated. (At present, all materials other than isotropic and unidirectional composite must be treated as a general anisotropic material.)

4.2.5 Computation Control Parameters — A, ALPHA, and TIME Card

Function: Defines the relaxation factors for the multistep iteration scheme and the starting time for a specific run.

Format: (3F10.4)

Variable	Column	Format	Description
A	1-10	F10.4	Multiplier of time step (Remark 1)
ALPHA	11-20	F10.4	Relaxation factor in multistep procedure (Remark 1)
TIME	21-30	F10.4	Starting time at present run

Example: (Card 22)

4.0	0.0	0.0

Remarks: (1) The definitions of A and ALPHA are defined in Eq. (5.36) of Part I. The values of these parameters depend on the specific problem. Computational experiments indicated that

$$1.0 \leq A \leq 4.0$$

$$0.0 \leq ALPHA \leq 1.0$$

A will be increased as the impact velocity increases.

(2) The default value of TIME = 0, representing a fresh start.

4.2.6 Initial Data Card

Function: Specifies the impact velocity vector and the dimensions of priori assigned impact zone.

Format: (6F10.5)

Variable	Column	Format	Description
V(1)	1-10	F10.5	Impact velocity
V(2)	11-20	F10.5	Components along x, y, and z-axis, respectively
V(3)	21-30	F10.5	
DIMX	31-40	F10.5	Dimensions of impact zone along x, y, and z-axis, respectively
DIMY	41-50	F10.5	
DIMZ	51-60	F10.5	

Example: (Card 23)

```
0.0     0.0     -7.5600+3     2.0000+0
2.0000+0     0.0820+0
```

4.2.7 Material Properties – ρ_T, ρ_P, μ_T, μ_P, Y_T, Y_P and μ_s Card

Function: Specifies initial material data.

Format: (7E10.4)

Variable	Column	Format	Description
RHOT	1-10	E10.4	Initial density for the target and projectile materials, respectively
RHOP	11-20	E10.4	
TMIU	21-30	E10.4	Isotropic shear moduli for target and projectile materials, respectively
PMIU	31-40	E10.4	
TYLD	41-50	E10.4	Isotropic yield strength for target and projectile materials, respectively
PYLD	51-60	E10.4	
SMIU	61-70	E10.4	Coulomb's friction coefficient

Example: (Card 24)

7.3000-2	5.1200-2	0.0	2.7027+3	0.0

6.0000+3	1.0

Remarks: (1) When ISOT $\neq$ 0, set TMIU, TYLD = 0.
When ISOP $\neq$ 0, set PMIU, PYLD = 0.

4.2.8 Mesh Data Card

Function: Specifies the nodes and elements in various zones.

Format: (14I4)

Variable	Column	Format	Description
NPS	1-4	I4	Total number of nodal points in the impact zone
NET	5-8	I4	Total number of elements in the impact zone
NPPJ	9-12	I4	Number of the first node in the projectile
NEPJ	13-16	I4	Number of the first element in the projectile
NPPJT	17-20	I4	Number of the last node in the projectile
NEPJT	21-24	I4	Number of the last element in the projectile
NPTR	25-28	I4	Number of the first node in the transition zone
NETR	29-32	I4	Number of the first element in the transition zone
NPLG	33-36	I4	Number of the first node in L_c zone
NELG	37-40	I4	Number of the first element in L_c zone
NPST	41-44	I4	Total number of nodes in CELFE substructure
NELT	45-48	I4	Total number of elements in CELFE substructure
MAXDOF	49-52	I4	Maximum numbering of degree of freedom in the entire system
NGD	53-56	I4	Total number of nodes in the entire system

Example: (Card 25)

76	33	13	3	16	3
62	30	77	34	76	33
648	103				

Remarks: (1) The finite element mesh numbering must start from the CELFE structure (cf. Section 2.10).

(2) MAXDOF is the maximum numbering of the degree of freedom; it may be larger or equal to the actual total number of degrees of freedom in the entire system, depending on the NASTRAN mesh numbering.

4.2.9 Boundary Condition Specification Card

Function: Specifies the total nodal numbers of boundary conditions for momentum, energy, density, stresses, and displacements in CELFE substructure.

Format: (17I4)

Variable	Column	Format	Description
IBDVT(1)	1-4	I4	Total number of nodes for momentum boundary conditions along the x, y, and z-directions, respectively
IBDVT(2)	5-8	I4	
IBDVT(3)	9-12	I4	
IBDVT(4)	13-16	I4	Total number of nodes for energy boundary conditions
IBDVT(5)	17-20	I4	Total number of nodes for density boundary conditions
IBDST(1, 1)	21-24	I4	Total number of nodes for stress boundary conditions
IBDST(1, 2)	25-28		
.	.	.	
.	.	.	
.	.	.	
IBDST(3, 3)	53-56	I4	
IBDIST(1)	57-60	I4	Total number of nodes for translational displacement boundary conditions
IBDIST(2)	61-64	I4	
IBDIST(3)	65-68	I4	

Example: Card 26)

20	20	0	15	115	15	10	10	10

20	10	10	10	10	10	20	20	0

4.2.10 Mesh Generation — NOD(I, J) Cards

Function: Generate the finite element mesh for CELFE model. Here,

NOD(I, J) = Global number of the J^{th} node in the element I

I = 1, . . . , NELT (total number of nodes in CELFE)

J = 1, . . . , NPE (total number of nodes in an element)

IORDC or IORDL	1	2	3
NPE	8	20	32
Format	8I4	20I4	32I4
No. of Cards per Element	1	1	2

Example: (Cards 27 through 59)

1	4	10	7	2
5	11	8		

Remarks: (1) Cf. Section 2.10 for details.

(2) When only one layer of elements are used for any direction in a given problem, at least quadratic elements should be employed.

4.2.11 Nodal Coordinates — GRID Cards

Function: Specify the nodal coordinates for the CELFE model

Format: (8A1, 2I8, 3F8.4)

Variable	Column	Format	Description
GRID	1-8	8A1	
N	9-16	I8	Number of node
KODE(N)	17-24	I8	Node I.D. number: = 0 Eulerian mesh = 1 E-L mesh = 2 Interfacial mesh for impact (E-L) zone and L_c zone = 3 L_c mesh (cf. Remarks 2, 3)
X(N, 1)	25-32	F8.4	Nodal coordinates
X(N, 2)	33-40	F8.4	
X(N, 3)	41-48	F8.4	

Example: (Cards 60 through 135)

GRID	10	2	0.15	0.15	0.0

Remarks: (1) Each card per node is required.

(2) In order to prevent possible error when the projectile penetrates to the point near the opposite side, it is suggested that the KODE value of surface nodes opposite the impact surface is set to 2 (see example).

(3) To reduce computation time for problems with high enough impact velocity when penetration is likely to occur, the nodes in the vicinity of the impact point may be set to 0 a priori. Otherwise, all nodes in the impact zone should be set as 1 at t = 0.

4.2.12 Free Surface Nodes — NFS(I) Cards

Function: Identify the free surface nodes in CELFE model.

NFS(I) — Global number of nodes on the free surface

I = 1, . . . , NFST (total number of free surface nodes)

Format: (20I4) per card

Example: (Cards 136 through 138)

1	4	6	7	9	....

4.2.13 Initial Impact Surface Nodes — INTF(I) Cards

Function: Identify the nodes on the impact surface at t = 0.

INTF(I) — Global number of nodes on the impact surface

I = 1, . . . , INTFT (total number of nodes on the initial impact surface

Format: (20I4) per card

Example: (Card 139)

3	6	9	12

Remarks: (1) Impact surface may consist of only one node.

4.2.14 Anisotropic Material Stiffness Coefficients — C(N, I, J) Card(s)

Function: Specifies stiffness coefficients C_{ij}, i, j = 1, . . . 6, of anisotropic materials for the entire CELFE system on the global coordinate systems.

Format: (2I4) for each given (I, J), and follows by (8E10.4) per card for (C(N, I, J), N = 1, ... NPST).

Variable	Format	No. of Cards	Description
(I, J)	2I4	One for each pair of (I, J)	Indices of C_{ij}. Set I ≥ 7 for ending the reading. For C_{ij}= C_{ij}, i ≠ j, choose either one
C(N, I, J)	8E10.4	NPST/8	Stiffness coefficients on global coordinate system

Example: (Nil, since ISOT = 1, ISOP = 0)

Remarks: (1) This set of cards must be used when either ISOT > 1 or ISOP > 1.

(2) When ISOT and ISOP ≤ 1, skip the set.

4.2.15 Anisotropic Material Properties for Target – SLT(I, J), SLC(I, J), E(I, J) and PNIU(I, J) Cards

Function: Specify anisotropic material properties for target on global coordinate system.

Format: (8E10.4) per card.

Variable	Format	No. of Cards per Variable	Description
((SLT(I, J), J=1, 3), I=1, 3)	9E10.4	2	For I=J: SLT(ImJ) = $S_{\ell ijT}$ (uniaxial tensil strength) For I≠J: SLT(I, J) = $S_{\ell ijS}$ (uniaxial shear strength)
((SLC(I, J), J=1, 3), I=1, 3)	9E10.4	2	For I=J: SLC(I, J) = $S_{\ell ijC}$ (uniaxial compression strength) For I≠J: SLC(I, J) = $S_{\ell ijS}$ (uniaxial shear strength)
((E(I, J), J=1, 3), I=1, 3)	9E10.4	2	For I=J: E(I, J) = $E_{\ell ii}$ (elastic modulus) For I≠J: E(I, J) = $G_{\ell ij}$ (shear modulus)
((PNIU(I, J), J=1, 3), I=1, 3)	9E10.4	2	Poisson's ratio. (Set 0 values for I=J)

Example: (Cards 140 through 147)

((SLT(I, J), J = 1, 3), I = 1, 3)

8.1+3	9.1+3	8.9+3	9.1+3	0.199+6	9.1+3	8.9+3	9.1+3
8.1+3							

Remarks: (1) When ISOT = 0, skip these cards.

4.2.16 Anisotropic Material Properties for Projectile – PSLT(I, J), PSLC(I, J) PE(I, J), and PPNIU(I, J) Cards

Function: Specify anisotropic material properties for projectile on global coordinate system.

Format: (Same as Section 4.2.15)

Example: (ISOP = 0, Nil)

4.2.17 Boundary Nodes for Conservation Equations — IBDV(I, J) Cards

Function: Identify the nodal number of boundary condition nodes for momentum, energy and density.

IBDV(I, J) — Global number of the I^{th} boundary node for the J^{th} variable

I = 1, . . . , IBDVT (defined in Section 4.2.9)

J = 1, 2, 3 — momentum in x, y, and z-axis, respectively

= 4 — energy

= 5 — density

Format: (20I4) per card per J.

Example: (Cards 148 through 151)

(IBDV(I, 1), I = 1, . . . , 20)

1	2	3	7	....	76

Remarks: (1) When IBDVT(I, J) = 0, skip the card.

4.2.18 Boundary Nodes for Constitutitve Equations – IBDS(I, J, K) Card

Function: Identify the nodal number of boundary condition nodes for stresses.

IBDV(I, J, K) – Global number of the I^{th} boundary node for $(J, K)^{th}$ component of the deviatoric stress tensor

I = 1, . . . , IBDST (defined in Section 4.2.9)

Format: (20I4) per card per component; the cards should be arranged according to ((IBDS(I, J, K), K = 1, 3), J = 1, 3)

Example: (Cards 152 through 160)

(IBDS(I, 1, 1), I = 1, 10)

62	64	65	67	68	70	71	73	74	76

Remarks: (1) If IBDST(I, J, K) = 0, skip the card.

4.2.19 Boundary Nodes for Displacement Vector — IBDIS(I, J) Cards

Function: Identify the nodal number of boundary condition nodes for displacements.

IBDIS(I, J) — Global number of the I^{th} boundary node for J^{th} component of the displacement vector.

I = 1, . . . , IBDIST (defined in Section 4.2.9)

Format: (20I4) per card per component

Example: (Cards 161, 162)

1	2	3	7	8	....	76

Remarks: (1) When IBDIS(I, J) = 0, skip the card.

4.2.20 Time Integration Control — Time Interval DT(I) Cards

Function: Assign time steps for ITER loop (time integration procedure).

DT(I) — Time step (μsec)

I = 1, . . . , ITGIV (defined in Section 4.2.4)

Format: (8F10.4) per card.

Example: (Card 163)

0.2	0.2	. . . . .

4.3 INPUT PROCEDURE FOR NASTRAN

Rigid format 9, "Direct Transient Analysis," is selected to perform the NASTRAN portion of the analysis. As aforementioned, a DMAP alter is added to the format to control the CELFE/NASTRAN execution. The basic input preparation for the NASTRAN portion in CELFE/NASTRAN analysis follows the original rigid format 9. The user is presumably familiar with the applications of the NASTRAN program, and thus the detailed description of the input for the rigid format will not be repeated here (cf. Ref. 3 for instruction). Instead, in what follows, only the uncommon input procedure will be presented.

4.3.1 DMAP Alters for Rigid Format 9

The basic rigid format 9 provides all of the module operations required for the transient analysis. A DMAP alter program is added to the rigid format to read elemental matrices generated by CELFE, sum the matrices and solve the matrix equation AX = B for the "primary variables" and to add the CELFE stiffness matrices (K1, K2, and K3) to the NASTRAN system stiffness matrix (KGGX). Alters are also used to output to disk, the solution vector X for the "primary variables" and the NASTRAN data block UVN which contains the displacement, velocity and acceleration vectors from each time step solved by the NASTRAN transient response analysis.

The runstream for the entire alters is stored in the file 'NAS1' as shown in Table 4-3 (for the example described in Section 3.3). Within the rigid format 9 (e.g., cards 73 through 191 in Table 4-3), any NASTRAN element model can be used, provided the linking elements are properly chosen (within NASTRAN) for connecting the CELFE and NASTRAN mesh. The preparation of bulk data for all admissible models follow the original rigid format as mentioned. The example given in Section 3.3 demonstrated a typical model together with the procedure utilized to prepare its data.

Table 4-3

DMAP ALTER LISTINGS – NAS1 FILE

```
 $DATA,L NAS1,,NAS1,
DATA 7 RL70-5 12/29-17:05:36
     1.      $HDG,N
     2.      $XQT *NASTRAN,LINK1
     3.      NASTRAN BUFFSIZE=871
     4.      ID-NASTRAN,CELFE
     5.      APP DISP
     6.      SOL 9,0
     7.      TIME 60
     8.      DIAG 8,21
     9.      ALTER 2
    10.      $
    11.      $ ASSEMBLE AND SOLVE 11 PRIMARY VARIABLES
    12.      PARAM    //C,N,NOP/V,N,TRUE=-1 $
    13.      PARAM    //C,N,NOP/V,N,FALSE=+1 $
    14.      PARAM    //C,N,ADD/Y,N,NUEL/V,Y,NUMNP $
    15.      OUTPUT2  ,,,,//C,N,-1/C,N,14 $
    16.      INPUTT2  /,,,,/C,N,-1/C,N,15 $
    17.      $
    18.      LABEL OUTER $
    19.      PARAM    //C,N,SUB/V,Y,NOEQU/V,Y,NOEQU $
    20.      COND     BOTTOM1,NOEQU $
    21.      INPUTT2  /A1,B1,,,/C,N,0/C,N,15 $
    22.      ADD      A1,/AX $
    23.      ADD      B1,/BX $
    24.      $
    25.      LABEL INNER $
    26.      PARAM    //C,N,SUB/V,Y,NUMNP/V,Y,NUMNP $
    27.      COND     BOTTOM2,NUMNP $
    28.      EQUIV    AX,AT/FALSE $
    29.      EQUIV    BX,BT/FALSE $
    30.      INPUTT2  /A,B,,,/V,Y,BLOK/C,N,15 $
    31.      ADD      A,AX/AT $
    32.      EQUIV    AT,AX/TRUE $
    33.      ADD      B,BX/BT $
    34.      EQUIV    BT,BX/TRUE $
    35.      REPT INNER,31 $
    36.      $
    37.      LABEL    BOTTOM2 $
    38.      MATPRN   AT,BT,,,// $
    39.      SOLVE    AT,BT/X/V,Y,SYM/C,N,1/C,N,1/C,N,1 $
    40.      MATPRN   X,,,,// $
    41.      OUTPUT2  X,,,,//C,N,0/C,N,14 $
    42.      PARAM    //C,N,SUB/V,Y,NUMNP/V,N,NUEL $
    43.      REPT OUTER,11 $
    44.      $
    45.      LABEL  BOTTOM1 $
    46.      OUTPUT2  ,,,,//C,N,-9/C,N,14 $
    47.      $
    48.      ALTER 30
    49.      $
    50.      $ ASSEMBLE STIFFNESS MATRIX KGG
    51.      PARAM    //C,N,SUB/V,Y,NUMNP/V,N,NUEL $
    52.      INPUTT2  /,,,,/C,N,-1/C,N,17 $
    53.      LABEL KLOOP $
    54.      PARAM    //C,N,SUB/V,Y,NUMNP/V,Y,NUMNP $
```

(Continued)

Table 4-3 (Continued)

```
  55.       COND      KEND,NUMNP $
  56.       EQUIV     KGGX,KGGG/FALSE $
  57.       INPUTT2   /K1,K2,K3,,/V,Y,BLOK/C,N,17 $
  58.       ADDS      K1,K2,K3,KGGX,/KGGG $
  59.       EQUIV     KGGG,KGGX/TRUE $
  60.       REPT KLOOP,32 $
  61.       LABEL     KEND $
  62.       MATPRN    KGGX,,,,//-1 $
  63.       $
  64.       ALTER 141
  65.       $
  66.       $ OUTPUT DISP AND VELOCITY TO CELFE
  67.       UMERGE    USETD,UDVT,/UVN/C,N,N/C,N,D/C,N,S $
  68.       MATPRN    UVN,,,,//-1 $
  69.       OUTPUT2   UVN,,,,//C,N,-1/C,N,12 $
  70.       $
  71.       ENDALTER
  72.       CEND
  73.       TITLE= 5.0 X 5.0 X 0.0820 INCH THICK PLATE
  74.       SUBTITLE= ASSEMBLE AND SOLVE PRIMARY VARIABLES
  75.       LABEL= ONE TIME STEP TRANSIENT ANALYSIS
  76.       IC=1
  77.       MPC=1
  78.       SPC=1
  79.       TSTEP=1
  80.       DLOAD=1
  81.       OUTPUT
  82.       SET 1= 62 THRU 76,206,231,236
  83.       DISP=1
  84.       VELO=1
  85.       BEGIN BULK
  86.       $ PCH1 CONTAINS AREA, GRID AND TSTEP CARDS FROM CELFE, CHANGE EACH DT
  87.       WADD,P PCH1,
  88.       PARAM   BLOK    0
  89.       PARAM   SYM     0
  90.       PARAM   NOEQU   1
  91.       PARAM   NUMNP   32
  92.       CELAS2  1001    1.+3    236     1
  93.       CELAS2  1002    1.+3    236     2
  94.       CELAS2  1003    1.+3    236     3
  95.       GRDSET                                                      456
  96.       CORD2R          1       0       0.0     0.0     0.0     0.0     0.0     1.0+222
  97.       +222            1.0     0.0     0.0
  98.       PQUAD2          201     123     .082
  99.       CQUAD2          203     201     175     204     210     172
 100.       CQUAD2          204     201     204     205     211     210
 101.       CQUAD2          205     201     205     206     212     211
 102.       CQUAD2          208     201     172     210     216     169
 103.       CQUAD2          209     201     210     211     217     216
 104.       CQUAD2          210     201     211     212     218     217
 105.       CQUAD2          211     201     163     166     220     219
 106.       CQUAD2          212     201     166     169     221     220
 107.       CQUAD2          213     201     169     216     222     221
 108.       CQUAD2          214     201     216     217     223     222
 109.       CQUAD2          215     201     217     218     224     223
 110.       CQUAD2          216     201     219     220     226     225
```

Table 4-3 (Continued)

```
111.    CQUAD2    217    201    220    221    227    226
112.    CQUAD2    218    201    221    222    228    227
113.    CQUAD2    219    201    222    223    229    228
114.    CQUAD2    220    201    223    224    230    229
115.    CQUAD2    221    201    225    226    232    231
116.    CQUAD2    222    201    226    227    233    232
117.    CQUAD2    223    201    227    228    234    233
118.    CQUAD2    224    201    228    229    235    234
119.    CQUAD2    225    201    229    230    236    235
120.    GRID   163    1       2.      .0     .041   0      6
121.    GRID   166    1       2.      1.     .041   0      6
122.    GRID   169    1       2.0     2.0    .041   0      6
123.    GRID   172    1       1.      2.     .041   0      6
124.    GRID   175    1       .0      2.0    .041   0      6
125.    GRID          204     1      .000   3.000   .041      0      6
126.    GRID          205     1      .000   4.000   .041      0      6
127.    GRID          206     1      .000   4.375   .041      0      6
128.    GRID          210     1     1.000   3.000   .041      0      6
129.    GRID          211     1     1.000   4.000   .041      0      6
130.    GRID          212     1     1.000   4.375   .041      0      6
131.    GRID          216     1     2.000   3.000   .041      0      6
132.    GRID          217     1     2.000   4.000   .041      0      6
133.    GRID          218     1     2.000   4.375   .041      0      6
134.    GRID          219     1     3.000   .000    .041      0      6
135.    GRID          220     1     3.000   1.000   .041      0      6
136.    GRID          221     1     3.000   2.000   .041      0      6
137.    GRID          222     1     3.000   3.000   .041      0      6
138.    GRID          223     1     3.000   4.000   .041      0      6
139.    GRID          224     1     3.000   4.375   .041      0      6
140.    GRID          225     1     4.000   .000    .041      0      6
141.    GRID          226     1     4.000   1.000   .041      0      6
142.    GRID          227     1     4.000   2.000   .041      0      6
143.    GRID          228     1     4.000   3.000   .041      0      6
144.    GRID          229     1     4.000   4.000   .041      0      6
145.    GRID          230     1     4.000   4.375   .041      0      6
146.    GRID          231     1     4.875   .000    .041      0      6
147.    GRID          232     1     4.875   1.000   .041      0      6
148.    GRID          233     1     4.875   2.000   .041      0      6
149.    GRID          234     1     4.875   3.000   .041      0      6
150.    GRID          235     1     4.875   4.000   .041      0      6
151.    GRID          236     1     4.875   4.375   .041      0      6
152.    MAT2   123    3.16+06 0.537+06 0.0    29.29+06 0.0    0.78+06 0.073
153.    CBAR   1      1       62      163     0.0     1.0     0.0     1
154.    CBAR   2      1       163     64      0.0     1.0     0.0     1
155.    CBAR   3      1       65      166     0.0     1.0     0.0     1
156.    CBAR   4      1       166     67      0.0     1.0     0.0     1
157.    CBAR   5      1       68      169     0.0     1.0     0.0     1
158.    CBAR   6      1       169     70      0.0     1.0     0.0     1
159.    CBAR   7      1       71      172     0.0     1.0     0.0     1
160.    CBAR   8      1       172     73      0.0     1.0     0.0     1
161.    CBAR   9      1       74      175     0.0     1.0     0.0     1
162.    CBAR   10     1       175     76      0.0     1.0     0.0     1
163.    PBAR   1      2       10.     10.     18.     20.
164.    MAT1   2      3.15+06          0.17    0.073
165.    MPC    1      163     1       1.0     63      1       -1.0
166.    MPC    1      163     2       1.0     63      2       -1.0
```

Table 4-3 (Concluded)

```
  167.      MPC     1       163     3       1.0     63      3       -1.0
  168.      MPC     1       166     1       1.0     66      1       -1.0
  169.      MPC     1       166     2       1.0     66      2       -1.0
  170.      MPC     1       166     3       1.0     66      3       -1.0
  171.      MPC     1       169     1       1.0     69      1       -1.0
  172.      MPC     1       169     2       1.0     69      2       -1.0
  173.      MPC     1       169     3       1.0     69      3       -1.0
  174.      MPC     1       172     1       1.0     72      1       -1.0
  175.      MPC     1       172     2       1.0     72      2       -1.0
  176.      MPC     1       172     3       1.0     72      3       -1.0
  177.      MPC     1       175     1       1.0     75      1       -1.0
  178.      MPC     1       175     2       1.0     75      2       -1.0
  179.      MPC     1       175     3       1.0     75      3       -1.0
  180.      SPC1    1       15      1       2       3       7       8       9
  181.      SPC1    1       15      13      15      29      30      31      44
  182.      SPC1    1       15      45      46      59      60      61
  183.      SPC1    1       1       74      75      76
  184.      SPC1    1       24      1       2       3       4       5       6
  185.      SPC1    1       24      13      14      17      18      19      32
  186.      SPC1    1       24      33      34      47      48      49
  187.      SPC1    1       2       62      63      64
  188.      TLOAD1  1       2                               10
  189.      TABLED1 10                                                      +TD1
  190.      +TD1    0.      1.      1000.   1.      ENDT
  191.      ENDDATA
  192.      $ADD,P-,NASTRAN,CONTRL
END DATA.
```

4.3.2 User's Guide for Inputs in DMAP Alter Program – 'NAS1' File

The preparation of input data for the alters can be described by illustrating the 'NAS1' file for the demonstration example as follows:

DESCRIPTION OF NAS1 FILE

Card No.	Description
1	Suppress any heading previously defined.
2	Begin execution of NASTRAN program.
3	Set NASTRAN buffer size to 871 words.
6	Select rigid format 9 for transient analysis.
9	Insert DMAP alter cards 10 through 47 into the rigid format program after DMAP No. 2.
12	Initialize DMAP parameter TRUE = -1.
13	Initialize DMAP parameter FALSE = +1.
14	Calculate the number of CELFE elements (NUEL) in current analysis, NUEL = NUMNP+1. Where the user defines NUMNP on a bulk data PARAM input card (see card 91).
15	Write a tape header record on file 14; do not rewind.
16	Read the tape header record on file 15; do not rewind.
18	Top of outer loop.
19	Decrease the number of primary variable equations solved in NASTRAN by 1. NOEQU = NOEQU - 1, where NOEQU is defined by the user on a bulk data PARAM input card (see card 90).
20	If all equations have been solved, jump outside outer loop (Card 45), LABEL BOTTOM1.
21	Read first pair of element matrices A1 and B1 from file 15.
22	Transfer A1 to AX.
23	Transfer B1 to BX.
25	Top of inner loop which sums the element matrices.
26	Decrease the element counter NUMNP=NUMNP-1.
27	Jump to LABEL BOTTOM2 if NUMNP=-1 (all element matrices have been summed).
28	Break the equivalence between AX at AT.
29	Break the equivalence between BX at BT.
30	Read the A and B matrices for the next element from file 15. Do not rewind file 15.
31	Add the element matrix A to the total assembled matrix AT. A + AX = AT.
32	Equivalence the matrices AX = AT.
33	Add the element matrix B to the total assembled matrix BT. B + BX = BT.
34	Equivalence the matrices BX = BT.
35	Jump to the top of INNER loop (Card 25), repeat 31 times. Note the number 31 may be set to an arbitrarily large number, looping will then be terminated by the conditional jump at card 27.

(Continued)

37	LABEL BOTTOM2.
38	Print the assembled matrices AT, BT. This statement may be removed to minimize printout.
39	Solve the matrix equation A • X=B using unsymmetric matrix decomposition.
40	Print the solution Vector X.
41	Write the solution vector onto file 14; do not rewind.
42	Reset the parameter NUMNP=NUEL-1. NUMNP will then be used in the inner loop for the next primary variable to be solved by NASTRAN.
43	Jump to the top of the OUTER loop (card 18); repeat 11 times. Note: If the user requires less than 11 parameters the parameter NOEQU should be reset on the PARAM bulk data card (card 90).
45	LABEL BOTTOM1.
46	Close file 14 with a rewind.
48	Insert DMAP alter cards 49 through 63 into the rigid format program after DMAP No. 30.
51	Reset the parameter NUMNP=NUEL-1 for stiffness matrix loop.
52	Read tape header record on file 17; do not rewind.
53	LABEL KLOOP.
54	Decrease the element counter NUMNP=NUNMP-1.
55	Jump to LABEL KEND if NUMNP=-1 (all element stiffness matrices have been summed).
56	Break the equivalence between KGGX and KGGG.
57	Read CELFE elemental stiffness matrices K1, K2 and K3 from file 17, do not rewind.
58	Add the CELFE stiffness matrices to the NASTRAN stiffness matrix (KGGX).
59	Equivalence matrix KGGG = KGGX.
60	Jump to LABEL KLOOP (Card No. 52). If all CELFE elements have been assembled, terminate the looping and continue with the normal rigid format DMAP instructions.
61	LABEL KEND.
62	Print the assembled stiffness matrix KGGX. The user may remove this statement to minimize the printout.
64	Insert DMAP alter cards 65 through 70 into the rigid format program after DMAP No. 141.
67	Merge the S-set (single point constraints) with the D-set solution vector (UDVT) to form the N-set (UVN) which can be read by the CELFE program.
68	Print NASTRAN data block UVN. This data block contains the displacement, velocity and acceleration vectors for the transient analysis.
69	Write a header record followed by the UVN data block on file 12. Continue with the remaining NASTRAN DMAP instructions.
71	End of DMAP alters.
72	End of Exec control deck.
73-85	NASTRAN case control deck. Insert the DAREA, GRID, TIC and TSTEP cards generated by CELFE. These cards are updated every CELFE iteration.
88-91	Parameters input to NASTRAN: BLOK=0, do not rewind file SYM=0, use unsymmetric SOLVE logic NOEQU=11, number of primary variables NUMNP=32, one less than the number of CELFE elements present in problem.
92-191	Bulk data defining the NASTRAN finite element model.
192	NASTRAN control card file for execution of the NASTRAN program.

It is important to note that several PARAM cards must be defined in the bulk data:

BLOCK = 0	Used to control file positioning by the INPUTT2 module.
SYM = 0	Indicates the unsymmetric solution technique is used for module SOLVE.
NOEQU = 11	Number of "primary variables" to be solved. For $0 \leq$ ISOT, ISOP ≤ 5, there is always 11 variables; 3 components of momentum vector, total energy, density and 6 components of stress tensor. Otherwise, NOEQU = 14.
NUMNP = N-1	With N = number of CELFE elements.

In addition, for different problems with different CELFE elements, card 35 should be changed to its correct values, NUMNP-1 (or N-2).

4.4 OUTPUT IN CELFE

The output of the present program is also divided into two parts, a NASTRAN output printed in NASTRAN, and a CELFE output composed of the following sets.

Output Listings: Output Set No.	Description
1	Nodal no., energy, density, pressure
2	Nodal no., stresses (σ_{ij})
3	Nodal no., free surface coordinates x_1, x_2, x_3
4	Nodal no., KODE, updated mesh x_1, x_2, x_3
5	Nodal no., displacements u_1, u_2, u_3, velocities v_1, v_2, v_3
6(*)	Failure number, nodal number of the failure node
7(*)	Nodal no., stiffness coefficients, C_{ij}

Remark: (*) If there is no failure, skip the 6th and 7th sets of output.

Cards Punched

All the data for the last time step, i.e., ITER = ITGIV, are punched using E-format for variables, and I-format for integers. When restart is needed, replace the cards from 27 to 147 in file 'START' by this set of punched cards.

Units:

The units used in the output are identical to their counterparts in the input, and were summarized in Table 4-1.

4.5 EXECUTIONS

The executions of the present program vary according to the problems and the options desired by the user. In what follows, only those basic ones will be described.

4.5.1 In-Core CELFE Executions

For in-core CELFE run, only 'LEE' file and 'START' file are needed, e.g.,

```
@ ASG, T              LEE, F///500
@ ASG, T              START, F
@ USE TPF$., LEE.
@ ADD, P              START
```

For restarting the in-core run, replace the cards from 27 to 147 in file 'START' by the output cards punched from the previous run.

An alternative may be used by assigning a 'RESTR' tape for each run, and store all data in internal file 16 into 'RESTR' file for the subsequent restart. For example, in the fresh start, assign a 'RESTR' tape as follows:

```
                .
                .
                .
@ASG, T                 RESTR, T, _____
                .
                .
                .
                .
@ADD, P                 START
@COPY, GMC              16, RESTR
                .
                .
                .
```

Then, for each restart, reassign the 'RESTR' tape. In addition, replace the 'START' file with a data file 'RESTART', e.g.,

```
                .
                .
                .
@ASG, T                 RESTR, T, _____
                .
                .
                .
@ADD, P                 RESTART.
@COPY, GMC              16, RESTR
                .
                .
                .
```

where the 'RESTART' file reads

```
@DATA, L        RESTART
@ASG, T PLT1, F///2000
@USE 12., PLT1.
@ASG, T INPT, F///2000
@ASG, T INP1, F///2000
@ASG, T INP2, F///2000
@ASG, T INP3, F///2000
@USE 14., INPT.
@USE 15., INP1.
@USE 16., INP2.
@USE 17., INP3.
```

```
@ ASG,A PCH2.
@ USE 18., PCH2.
@ ASG,T PCH1,F///1000
@ ASG,C PCH3,F///1000
@ USE 19., PCH3.
@ COPY,        RESTR,16
@ BRKPT PUNCH$/PCH1
@ XQT CELFE
+0
+0
@ BRKPT        PUNCH$
@ FREE,B       18.
@ FREE,B       19.
@ COPY         PCH3.,PCH2.
@ END
```

4.5.2 Fresh Start of CELFE/NASTRAN Run

The control cards for starting the CELFE/NASTRAN run are illustrated in Table 4-4.

Table 4-4

CONTROL CARDS FOR CELFE/NASTRAN RUN – FILE 'RUN1'

```
@DATA,L RUN1.
DATA 7 RL70-5 01/09-23:42:57
     1.      @ASG,A START.
     2.      @ASG,A LEE.
     3.      @ASG,A NAS1.
     4.      @ASG,A PCH2.
     5.      @ASG,A LOOP.
     6.      @ASG,A NASTRAN.
     7.      @ASG,A OBJ.
     8.      @USE TPF$.,LEE.
     9.      @ADD,P START.
    10.      @CPDMPH,DO
    11.       PRT 15,30
    12.       PRT 17,30
    13.      @ADD,P LOOP.
    14.      @ADD,P LOOP.
    15.      @ADD,P LOOP.
    16.      @ADD,P LOOP.
    17.      @ADD,P LOOP.
    18.      @PMD,E
    19.      @FIN
END DATA.
```

4.5.3 Restart CELFE/NASTRAN Run

As in the in-core option, data file 'START' will not be used in the restart of CELFE/NASTRAN run. Instead, a 'RESTR' tape should be prepared to store all data in relevant files for the subsequent executions. The overall runstream reads

```
@  Run Card
@  ASG,T          IN,T,_____
@  ASG,T          RESTR,T,
                  .
                  .
                  .
                  .
                  .
@  PREP
@  REWIND         RESTR.
@  ASG,T          PLT1,F///2000
@  ASG,T          INPT,F///2000
@  ASG,T          INP1,F///2000
@  ASG,T          INP2,F///2000
@  ASG,T          INP3,F///2000
@  USE      12., PLT1.
@  USE      14., INPT.
@  USE      15., INP1.
@  USE      16., INP2.
@  USE      17., INP3.
@  ASG,A    PCH2.
@  ASG,T    PCH1,F///1000
@  ASG,C    PCH3,F///1000
@  USE      18., PCH2.
@  USE      19., PCH3.
@  COPY,G         RESTR,PLT1
@  COPY,G         RESTR,14
@  COPY,G         RESTR,15
```

```
@ COPY,G          RESTR,16
@ COPY,G          RESTR,17
@ COPY,G          RESTR,PCH1
@ COPY,G          RESTR,PCH2
@ COPY,G          RESTR,PCH3

@ ADD,P           LOOP.

    .
    .
    .

@ ADD,P           LOOP.
@ REWIND          RESTR.
@ COPY,GMC        PLT1,RESTR
@ COPY,GMC        14,RESTR
@ COPY,GMC        15,RESTR
@ COPY,GMC        16,RESTR
@ COPY,GMC        17,RESTR
@ COPY,GMC        PCH1,RESTR
@ COPY,GMC        PCH2,RESTR
@ COPY,GMC        PHC3,RESTR
@ ADD,P           LOOP.
    .
    .
    .
    .
@ FIN
```

C-2

Appendix A
CELFE PROGRAM LISTINGS
(File 'LEE')

```
@FOR,S    ALMS,ALMS
HSA  E3 -01/09/78-23:43:02 (1,)

   SUBROUTINE ALMS      ENTRY POINT 000550

   STORAGE USED: CODE(1) 000604; DATA(0) 000114; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000172 10L      0001   000063 1426     0001   000122 1526     0001   000212 20L      0001   000245 2076
    0001   000275 2166     0001   000417 2416     0001   000426 2476     0001   000345 29L      0001   000367 30L
    0001   000371 35L      0001   000476 40L      0001   000516 50L      0000 R 000040 A        0000 R 000020 AMU
    0000 R 000010 AP1      0000 R 000011 AP2      0000 R 000000 AT1      0000 R 000001 AT2      0000 R 000025 A1
    0000 R 000026 A2       0000 R 000041 B        0000 R 000012 BP0      0000 R 000013 BP1      0000 R 000014 BP2
    0000 R 000002 BT0      0000 R 000003 BT1      0000 R 000004 BT2      0000 R 000027 B0       0000 R 000042 B01
    0000 R 000030 B1       0000 R 000031 B2       0000 R 000022 C        0000 R 000015 CP0      0000 R 000016 CP1
    0000 R 000005 CT0      0000 R 000006 CT1      0000 R 000032 C0       0000 R 000033 C1       0000 R 000021 E
    0000 R 000035 ETA      0000 I 000034 I        0000 I 000043 II       0000   000046 INJP$    0000 I 000037 J
    0000 R 000023 PHI      0000 R 000017 PHIP     0000 R 000007 PHIT     0000 R 000024 RHO      0000 R 000036 SUMV
```

```
00101     1*          SUBROUTINE ALMS(NPM,NP,NN,NPJ,INTF,INT,NFRAC,RHOT,RHOP,VER,P)                  000033
00101     2*    C                                                                                    000033
00101     3*    C     LOS ALAMOS EQUATION OF STATE                                                   000033
00101     4*    C                                                                                    000033
00103     5*          INCLUDE STATEQ,LIST                                                            000033
00103     5*    STATEQ PROC                                                                          000033
00103     5*    C                                                                                    000033
00104     5*          DIMENSION VER(NPM,5),P(1),INTF(1)                                              000033
00104     5*    C                                                                                    000033
00104     5*    C                                                                                    000033
00104     5*    C    TARGET  ---  BORON/EPOXY                                                        000033
00104     5*    C                                                                                    000033
00105     5*          DATA AT1,AT2,BT0,BT1,BT2,CT0,CT1,PHIT/1.8212,4.3509,0.3764,0.3287,             000033
00105     5*         1  1.0801,0.5531,0.6346,0.250/                                                 000033
00105     5*    C                                                                                    000033
00105     5*    C     PROJECTILE ---  SILASTIC                                                       000033
00105     5*    C                                                                                    000033
00116     5*          DATA AP1,AP2,BP0,BP1,BP2,CP0,CP1,PHIP/0.004794,0.04684,0.33969,                000033
00116     5*         1  0.02377,0.50767,0.4925,0.5721,0.3000/                                        000033
00116     5*      END                                                                               000033
00127     6*          PFCN(X,Y)=(X*AMU+Y*E+C*E*E)/(E+PHI)                                            000033
00127     7*    C                                                                                    000033
00127     8*    C    TARGET                                                                          000033
00127     9*    C                                                                                    000033
00130    10*          RHO=RHOT                                                                       000033
```

```
00131     11*          A1=AT1                                                    000035
00132     12*          A2=AT2                                                    000037
00133     13*          B0=BT0                                                    000041
00134     14*          B1=BT1                                                    000043
00135     15*          B2=BT2                                                    000045
00136     16*          C0=CT0                                                    000047
00137     17*          C1=CT1                                                    000051
00140     18*          PHI=PHIT                                                  000053
00141     19*          DO 20 I=1,NP                                              000063
00144     20*          IF (I .GE. NN .AND. I .LE. NPJ) GO TO 20                  000070
00146     21*          ETA=VER(I,5)/RHO                                          000106
00147     22*          AMU = ETA - 1.0                                           000111
00150     23*          SUMV = 0.0                                                000113
00151     24*          DO 8 J = 1,3                                              000122
00154     25*        8 SUMV = SUMV + VER(I,J)*VER(I,J)                           000122
00156     26*          SUMV = 0.5*SUMV/VER(I,5)                                  000126
00157     27*          E = (VER(I,4) - SUMV)/ETA                                 000131
00160     28*          A = A1 + A2*AMU                                           000134
00161     29*          B = B0 + AMU*(B1 + B2*AMU)                                000140
00162     30*          C = C0 + C1*AMU                                           000146
00163     31*          IF(ETA .LT. 1.0) GO TO 10                                 000152
00163     32*      C.....COMPRESSION....                                         000152
00165     33*          P(I)=PFCN(A,B)                                            000156
00166     34*          GO TO 20                                                  000170
00166     35*      C.....EXPANSION.....                                          000170
00167     36*       10 B01=B0+B1*AMU                                             000172
00170     37*          P(I)=PFCN(A1,B01)                                         000175
00171     38*       20 CONTINUE                                                  000216
00173     39*          RHO=RHOP                                                  000216
00174     40*          A1=AP1                                                    000220
00175     41*          A2=AP2                                                    000222
00176     42*          B0=BP0                                                    000224
00177     43*          B1=BP1                                                    000226
00200     44*          B2=BP2                                                    000230
00201     45*          C0=CP0                                                    000232
00202     46*          C1=CP1                                                    000234
00203     47*          PHI=PHIP                                                  000236
00204     48*          IF (NFRAC .EQ. 0) GO TO 35                                000240
00206     49*          DO 30 II=1,INT                                            000245
00211     50*          I=INTF(II)                                                000245
00212     51*          ETA=VER(I,5)/RHO                                          000246
00213     52*          AMU = ETA - 1.0                                           000254
00214     53*          SUMV = 0.0                                                000256
00215     54*          DO 28 J= 1,3                                              000275
00220     55*       28 SUMV = SUMV + VER(I,J)*VER(I,J)                           000275
00222     56*          SUMV = 0.5*SUMV/VER(I,5)                                  000301
00223     57*          E = (VER(I,4) - SUMV)/ETA                                 000304
00224     58*          A = A1 + A2*AMU                                           000307
00225     59*          B = B0 + AMU*(B1 + B2*AMU)                                000313
00226     60*          C = C0 + C1*AMU                                           000321
00227     61*          IF(ETA .LT. 1.0) GO TO 29                                 000325
00227     62*      C.....COMPRESSION....                                         000325
00231     63*          P(I)=PFCN(A,B)                                            000331
00232     64*          GO TO 30                                                  000343
00232     65*      C.....EXPANSION.....                                          000343
00233     66*       29 B01=B0+B1*AMU                                             000345
```

```
00234     67*            P(I)=PFCN(A1,B01)                                    000352
00235     68*         30 CONTINUE                                             000371
00237     69*         35 CONTINUE                                             000371
00237     70*      C                                                          000371
00237     71*      C     PROJECTILE                                           000371
00237     72*      C                                                          000371
00240     73*            DO 50 I=NN,NPJ                                       000371
00243     74*            ETA=VER(I,5)/RHO                                     000417
00244     75*            AMU = ETA - 1.0                                      000421
00245     76*            SUMV = 0.0                                           000423
00246     77*            DO 38 J= 1,3                                         000426
00251     78*         38 SUMV = SUMV + VER(I,J)*VER(I,J)                      000426
00253     79*            SUMV = 0.5*SUMV/VER(I,5)                             000432
00254     80*            E = (VER(I,4) - SUMV)/ETA                            000435
00255     81*            A = A1 + A2*AMU                                      000440
00256     82*            B = B0 + AMU*(B1 + B2*AMU)                           000444
00257     83*            C = C0 + C1*AMU                                      000452
00260     84*            IF(ETA .LT. 1.0) GO TO 40                            000456
00260     85*      C.....COMPRESSION....                                      000456
00262     86*            P(I)=PFCN(A,B)                                       000462
00263     87*            GO TO 50                                             000474
00263     88*      C.....EXPANSION.....                                       000474
00264     89*         40 B01=B0+B1*AMU                                        000476
00265     90*            P(I)=PFCN(A1,B01)                                    000501
00266     91*         50 CONTINUE                                             000524
00270     92*            RETURN                                               000524
00271     93*            END                                                  000603

     END OF COMPILATION:      NO  DIAGNOSTICS.

@HDG,P      BNDE
```

```
BNDE                                                                    DATE 010978

@FOR,S    BNDE,BNDE
HSA. E3 -01/09/78-23:43:05 (0,)

   SUBROUTINE BNDEQ     ENTRY POINT 000470

   STORAGE USED: CODE(1) 000516; DATA(0) 000117; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NPRT$
    0004   NIO2$
    0005   NIO1$
    0006   NSTOP$
    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000311 1000L      0001   000051 111G      0001   000314 1200L      0001   000114 130G      0001   000146 137G
    0001   000376 1500L      0001   000206 154G      0001   000405 1600L      0000   000026 1601F     0000   000047 1603F
    0001   000265 162G       0001   000303 172G      0001   000335 204G       0001   000344 207G      0001   000427 224G
    0001   000433 230G       0001   000057 410L      0000   000013 420F       0001   000071 430L      0001   000154 500L
    0001   000306 900L       0000 R 000010 B         0000 R 000000 CERO       0000 I 000004 I         0000   000053 INJP$
    0000 I 000007 J          0000 I 000012 JFIRST     0000 I 000005 JLAST      0000 I 000011 K         0000 I 000006 L
    0000 I 000003 NPM        0000 I 000002 NBND       0000 R 000001 PARE

00101      1*            SUBROUTINE BNDEQ(A,NRMAX,NCMAX,N,ITERM)                                    000023
00101      2*       C    EQUATION SOLVER FOR BANDED NON-SYMMETRIC SYSTEM OF EQUATIONS               000023
00101      3*       C    SOLUTION STORED IN THE COLUMN  A(N,2*ITERM)                                000023
00103      4*            DIMENSION A(NRMAX,NCMAX)                                                   000023
00104      5*            CERO = 1.D-6                                                               000023
00105      6*            PARE = CERO**2                                                             000025
00106      7*            NBND=2*ITERM                                                               000027
00107      8*            NBM = NBND - 1                                                             000034
00107      9*       C    BEGINS ELIMINATION OF THE LOWER LEFT                                       000034
00110     10*            DO 1000 I=1, N                                                             000041
00113     11*            IF ( ABS(A(I,ITERM))  .LT.  CERO)  GO TO 410                               000051
00115     12*            GO TO 430                                                                  000055
00116     13*        410 IF ( ABS(A(I,ITERM))  .LT.  PARE)   GO TO 1600                            000057
00120     14*            PRINT 420, A(I,ITERM), I                                                   000062
00124     15*        420 FORMAT (' WARNING. ILL-CONDITIONED A-MATRIX. A=',E16.6,' I=',I2)           000071
00125     16*        430 JLAST = MIN0(I+ITERM-1, N)                                                 000071
00126     17*            L = ITERM + 1                                                              000100
00127     18*            DO 500 J=I, JLAST                                                          000114
00132     19*            L = L - 1                                                                  000114
00133     20*            IF ( ABS(A(J,L))  .LT.  PARE)  GO TO 500                                   000117
00135     21*            B = A(J,L)                                                                 000127
00136     22*            DO 450 K=L, NPND                                                           000146
00141     23*        450 A(J,K) = A(J,K) / B                                                        000146
00143     24*            IF (I  .EQ.  N)  GO TO 1200                                                000151
```

```
      BNDE                                                                DATE 010978

00145     25*        500 CONTINUE                                                    000155
00147     26*            L=0                                                         000155
00150     27*            JFIRST = I + 1                                              000156
00151     28*            IF (JLAST .LE. I) GO TO 1000                                000161
00153     29*            DO 900 J= JFIRST, JLAST                                     000177
00156     30*            L=L+1                                                       000216
00157     31*            IF ( ABS(A(J,ITERM-L)) .LT. PARE) GO TO 900                 000221
00161     32*            DO 600 K=ITERM, NBM                                         000265
00164     33*        600 A(J,K-L) = A(J-L,K) - A(J,K-L)                              000265
00166     34*            A(J,NBND) = A(J-L,NBND) - A(J,NBND)                         000270
00167     35*            IF (I .GE. N-ITERM+1) GO TO 900                             000273
00171     36*            DO 800 K=1, L                                               000303
00174     37*        800 A(J,NBND-K) = -A(J,NBND-K)                                  000303
00176     38*        900 CONTINUE                                                    000314
00200     39*       1000 CONTINUE                                                    000314
00202     40*       1200 L = ITERM - 1                                               000314
00203     41*            DO 1500 I=2, N                                              000316
00206     42*            DO 1500 J=1, L                                              000344
00211     43*            IF (N+1-I+J .GT. N) GO TO 1500                              000344
00213     44*            A(N+1-I,NBND) = A(N+1-I,NBND) - A(N+1-I+J,NBND)*A(N+1-I,ITERM+J)  000351
00214     45*       1500 CONTINUE                                                    000401
00217     46*            RETURN                                                      000401
00220     47*       1600 PRINT 1601                                                  000405
00222     48*       1601 FORMAT (' COMPUTATION STOPPED IN BNDEQ BECAUSE ZERO APPEARED ON  000410
00222     49*           1MAIN DIAGONAL.  THE MATRIX FOLLOWS.')                       000410
00223     50*            DO 1602 I=1, N                                              000410
00226     51*       1602 PRINT 1603, (A(I,J), J=1, NBND)                             000427
00235     52*       1603 FORMAT (10E12.4)                                            000442
00236     53*            STOP                                                        000442
00237     54*            END                                                         000515

      END OF COMPILATION:       NO DIAGNOSTICS.

@HDG,P    CELA
```

```
@FOR,S    CELA,CELA
HSA  E3 -01/09/78-23:43:07 (1,)

   SUBROUTINE CELANI    ENTRY POINT 001160

   STORAGE USED: CODE(1) 001236; DATA(0) 000111; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   SYSMTX 007641
    0004   CELZ1  000015
    0005   CELZ3  001140
    0006   ORTHPC 000060
    0007   MATRL  000005
    0010   NCASE  000003
    0011   GEOM1  000524
    0012   IMPCT1 006650
    0013   IMPCT3 000500
    0014   ELMLAG 000410
    0015   PARDOF 000002
    0016   STRESS 002530
    0017   STSELM 000110
    0020   IMPCT2 000230
    0021   SOLSTR 001254
    0022   BDSTR  002541

   EXTERNAL REFERENCES (BLOCK, NAME)

    0023   ELCELA
    0024   ELMATS
    0025   BNDEQ
    0026   NERR2$
    0027   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000352 105L      0001    000434 115L      0001    000443 118L      0001    000023 141G      0001    000024 144G
    0001    000621 150L      0001    000625 152L      0001    000041 153G      0001    000646 155L      0001    000077 157G
    0001    000110 164G      0001    000125 176G      0001    000714 200L      0001    000731 201L      0001    001015 203L
    0001    000151 204G      0001    001022 204L      0001    000152 207G      0001    000175 217G      0001    001075 220L
    0001    000176 222G      0001    000211 230G      0001    000245 240G      0001    000273 245G      0001    000312 252G
    0001    000333 260G      0001    000357 270G      0001    000366 274G      0001    000367 277G      0001    000413 306G
    0001    000430 315G      0001    000441 323G      0001    000512 333G      0001    000547 344G      0001    000551 347G
    0001    000573 361G      0001    000613 367G      0001    000661 404G      0001    000671 411G      0001    000767 430G
    0001    001002 435G      0001    001054 454G      0001    000032 46L       0001    001066 460G      0001    001124 471G
    0001    000051 48L       0001    000164 60L       0001    000167 68L       0001    000232 85L       0004    000001 A
    0004 R  000002 ALPHA     0012 R  001370 C         0014 R  000000 CD        0013 R  000040 CFRC      0006    000055 CK
    0014 R  000200 CL        0014 R  000100 CS        0004 R  000013 DELT      0014 R  000250 DF        0006    000033 E
    0014 R  000310 ELXYZ     0013 R  000070 FJCB      0012 R  001140 FJN       0012 R  001254 FJPRV     0000 I  000000 I
    0022 I  000011 IBDS      0022 I  000000 IBDST     0000 I  000012 IE        0000 I  000011 II        0000 I  000004 IN
    0010    000000 INDEX     0000    000041 INJPS     0004 I  000011 INNER     0004 I  000000 IORDC     0007    000004 JSOP
    0007    000203 ISOT      0004 I  000012 ITLR      0000 I  000001 J         0000 I  000006 K         0011 I  000000 KODE
```

```
0006 I 000022 LI     0015   000000 MAXDOF   0000 I 000002 N      0010 I 000001 NASTRN   0004 I 000006 NBW
0000 I 000010 NC     0004 I 000010 NEPJ     0004 I 000003 NET    0007   000000 NFRAC    0004   000014 NFST
0015   000001 NGD    0004 I 000005 NHBW     0000 I 000003 NI     0000 I 000005 NJ       0011 I 000114 NOD
0004   000007 NPPJ   0004 I 000004 NPS      0000 I 000007 NR     0010   000002 NSLIDE   0003 I 007640 NSTAB
0013 R 000000 OM     0014 R 000300 PM       0007   000002 PMIU   0006   000044 PNIU    0020 R 000000 PRSN
0020 R 000114 PRSPRV 0003 R 000000 S        0022 R 001265 SBD    0014 R 000210 SF       0006   000011 SLC
0006   000000 SLT    0017 R 000000 STL      0021 R 000000 STS    0016 R 000000 STSN     0016 R 001254 STSPRV
0007   000001 TMIU   0012 R 000344 VERN     0005 R 000344 VERPRV 0014 R 000340 VL       0014 R 000220 W
0012 R 000000 XN     0005 R 000000 XPRV
```

```
00101      1*          SUBROUTINE CELANI(INIF,INIFT,CINI,L1,L2)                                  000016
00101      2*        C                                                                           000016
00101      3*        C**************************                                                 000016
00101      4*        C                                                                           000016
00101      5*        C       ASSEMBLE SYSTEM MATRICES FOR CONSTITUTIVE EQUATIONS                 000016
00101      6*        C                                                                           000016
00101      7*        C**************************                                                 000016
00101      8*        C                                                                           000016
00103      9*              INCLUDE PARAM1,LIST                                                   000016
00103      9*        PARAM1 PROC                                                                 000016
00103      9*        C                                                                           000016
00104      9*              PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                           000016
00105      9*              PARAMETER NPELG= 8,NPE= 8                                             000016
00106      9*              PARAMETER NDFRS=53,NDINT=10                                           000016
00106      9*        C                                                                           000016
00107      9*              PARAMETER NRMAX=2000,NCMAX=2                                          000016
00110      9*              PARAMETER NGPLG=2,NGP=2                                               000016
00111      9*              PARAMETER NCNT=6, NFRC=NPM                                            000016
00112      9*              PARAMETER IU=5,NDPJT= 10,NTSTEP=200                                   000016
00113      9*              PARAMETER NBV=NPM,NBS=NPM,NBD=NPM                                     000016
00113      9*         END                                                                        000016
00114     10*              INCLUDE CONSTV,LIST                                                   000016
00114     10*         CONSTV PROC                                                                000016
00115     10*              COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB                                  000016
00116     10*              COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,        000016
00116     10*             1  ITER,DELT,NFST                                                     000016
00117     10*              COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)                          000016
00120     10*              COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)      000016
00121     10*              COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                              000016
00122     10*              COMMON /NCASE/ INDEX,NASTRN,NSLIDE                                    000016
00123     10*              COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)                             000016
00124     10*              COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPMJ),FJPRV(NPM),        000016
00124     10*             1C(NPMT,NCNT,NCNT)                                                     000016
00125     10*              COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)             000016
00126     10*              COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),            000016
00126     10*             1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),            000016
00126     10*             2VL(NPELG,IU)                                                          000016
00127     10*              COMMON /PARDOF/ MAXDOF,NGD                                            000016
00130     10*              COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)                         000016
00131     10*              COMMON /STSELM/ STL(NPE,3,3)                                          000016
00132     10*              COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)                                 000016
00133     10*              COMMON /SOLSTR/ STS(NPM,3,3)                                          000016
00134     10*              COMMON /BDSTR/ BDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)                   000016
```

```
00134     10*       END                                                                  000016
00135     11*       DIMENSION CINT(INTFT,NCNT,NCNT),INTF(1)                              000016
00135     12*      C                                                                     000016
00135     13*      C                                                                     000016
00136     14*            IF (NASTRN .NE. 0) GO TO 46                                     000016
00140     15*            DO 45 I=1,NRMAX                                                 000024
00143     16*            DO 45 J=1,NCMAX                                                 000024
00146     17*         45 S(I,J)=0.0                                                      000024
00151     18*         46 CONTINUE                                                        000032
00152     19*            DO 200 N=1,NET                                                  000032
00155     20*            GO TO (48,85),INNER                                             000041
00156     21*         48 DO 80 I=1,NPE                                                   000051
00161     22*            NI=NOD(N,I)                                                     000077
00162     23*            FJCB(I)=FJPRV(NI)                                               000100
00163     24*            DO 50 J=1,3                                                     000110
00166     25*            ELXYZ(I,J)=XPRV(NI,J)                                           000110
00167     26*         50 VL(I,J)=VERPRV(NI,J)/VERPRV(NI,5)                               000111
00171     27*            IF(ITER  .GT. 1 .OR. N .GE. NEPJ) GO TO 68                      000115
00173     28*            VL(I,3)=0.0                                                     000117
00174     29*            VL(I,4)=0.0                                                     000120
00175     30*            DO 60  IN=1,INTFT                                               000125
00200     31*            NJ=INTF(IN)                                                     000133
00201     32*            IF(NJ .NE. NI) GO TO 60                                         000135
00203     33*            DO 55 J=1,NCNT                                                  000137
00206     34*            DO 55 K=1,NCNT                                                  000152
00211     35*         55 C(NI,J,K)=CINT(IN,J,K)                                          000152
00214     36*         60 CONTINUE                                                        000167
00216     37*         68 DO 75 J=1,3                                                     000167
00221     38*            DO 75 K=1,3                                                     000176
00224     39*         75 STL(I,J,K)=STSPRV(NI,J,K)                                       000176
00227     40*            DO 78 J=1,3                                                     000211
00232     41*         78 STL(I,J,J)=STL(I,J,J)+PRSPRV(NI)                                000211
00234     42*         80 CONTINUE                                                        000230
00236     43*            GO TO 105                                                       000230
00237     44*         85 DO 100 I=1,NPE                                                  000232
00242     45*            NI=NOD(N,I)                                                     000245
00243     46*            FJCB(I)=ALPHA*FJPRV(NI)+(1.0-ALPHA)*FJN(NI)                     000256
00244     47*            DO 90 J=1,3                                                     000273
00247     48*            ELXYZ(I,J)=ALPHA*XPRV(NI,J)+(1.0-ALPHA)*XN(NI,J)                000273
00250     49*            VL(I,J)=ALPHA*VERPRV(NI,J)/VERPRV(NI,5)+(1.0-ALPHA)*            000300
00250     50*           1  VERN(NI,J)/VERN(NI,5)                                        000300
00251     51*            DO 90 K=1,3                                                     000312
00254     52*         90 STL(I,J,K)=ALPHA*STSPRV(NI,J,K)+(1.0-ALPHA)*STSN(NI,J,K)        000312
00257     53*            DO 95 J=1,3                                                     000333
00262     54*         95 STL(I,J,J)=STL(I,J,J)+ALPHA*PRSPRV(NI)+(1.0-ALPHA)*PRSN(NI)     000333
00264     55*        100 CONTINUE                                                        000352
00266     56*        105 CONTINUE                                                        000352
00267     57*            DO 110 I=1,NPE                                                  000352
00272     58*            NI=NOD(N,I)                                                     000357
00273     59*            DO 110 J=1,NCNT                                                 000361
00276     60*            DO 110 K=1,NCNT                                                 000367
00301     61*        110 CFRC(I,J,K)=C(NI,J,K)                                           000367
00305     62*            DO 118 I=1,NPE                                                  000413
00310     63*            NI=NOD(N,I)                                                     000416
00311     64*            IF(KODE(NI)) 112,112,115                                        000420
00314     65*        112 DO 113 J=1,3                                                    000430
```

```
00317     66*        113 OM(I,J)=VL(I,J)                                                    000430
00321     67*            GO TO 118                                                          000432
00322     68*        115 DO 116 J=1,3                                                       000434
00325     69*        116 OM(I,J)=0.0                                                        000441
00327     70*        118 CONTINUE                                                           000445
00331     71*            CALL ELCELA(NGP,NPE,ELXYZ,SF,DF,W,VL,OM,FJCB,STL,CD,CS,CL,LI,CFRC,  000445
00331     72*           1NCNT,DELT,L1,L2,IORDC)                                             000445
00332     73*            DO 120 I=1,NPE                                                     000473
00335     74*            NI=NOD(N,I)                                                        000512
00336     75*            PM(I)=STSPRV(NI,L1,L2)                                             000514
00337     76*            IF (L1 .EQ. L2) PM(I)=PM(I)+PRSPRV(NI)                             000520
00341     77*        120 CONTINUE                                                           000535
00343     78*            DO 140 I=1,NPE                                                     000535
00346     79*            DO 140 J=1,NPE                                                     000551
00351     80*        140 CL(I)=CL(I)+CD(I,J)*PM(J)                                          000551
00354     81*            IF (NASTRN .EQ. 0) GO TO 155                                       000562
00356     82*            IF (NSTAB .NE. 0) GO TO 152                                        000564
00360     83*            DO 150 I=1,NPE                                                     000573
00363     84*            NI=NOD(N,I)                                                        000576
00364     85*            IF (KODE(NI) .LT. 2) GO TO 150                                     000600
00366     86*            DO 148 J=1,NPE                                                     000613
00371     87*        148 CS(I,J)=0.0                                                        000613
00373     88*            CS(I,I)=1.0                                                        000614
00374     89*            CL(I)=STSPRV(NI,L1,L2)                                             000616
00375     90*        150 CONTINUE                                                           000625
00377     91*        152 CONTINUE                                                           000625
00400     92*            CALL ELMATS(CS,CL,NPS,NEMT,NPELG,NOD,N,CD,IBDST,IBDS,SBD,NBS,      000625
00400     93*           1  L1,L2)                                                           000625
00401     94*            GO TO 200                                                          000644
00402     95*        155 CONTINUE                                                           000646
00403     96*            DO 160 I=1,NPE                                                     000646
00406     97*            NR=NOD(N,I)                                                        000661
00407     98*            S(NR,NBW)=S(NR,NBW)+CL(I)                                          000664
00410     99*            DO 160 J=1,NPE                                                     000671
00413    100*            NC=NOD(N,J)-NR+NHBW                                                000671
00414    101*        160 S(NR,NC)=S(NR,NC)+CS(I,J)                                          000676
00417    102*        200 CONTINUE                                                           000723
00421    103*            IF (NASTRN .EQ. 0) GO TO 201                                       000723
00423    104*            RETURN                                                             000725
00424    105*        201 CONTINUE                                                           000731
00425    106*            IF (NSTAB .NE. 0) GO TO 204                                        000731
00427    107*            DO 203 I=1,NPS                                                     000767
00432    108*            IF (KODE(I) .LT. 2) GO TO 203                                      000772
00434    109*            DO 202 J=1,NBW                                                     001002
00437    110*        202 S(I,J)=0.0                                                         001002
00441    111*            S(I,NHBW)=1.0                                                      001003
00442    112*            S(I,NBW)=STSPRV(I,L1,L2)                                           001005
00443    113*            IF (L1 .EQ. L2) S(I,NBW)=S(I,NBW)+PRSPRV(I)                        001007
00445    114*        203 CONTINUE                                                           001022
00447    115*        204 CONTINUE                                                           001022
00450    116*            IF(IBDST(L1,L2).EQ. 0) GO TO 220                                   001022
00452    117*            II=IBDST(L1,L2)                                                    001026
00453    118*            DO 210 I=1,II                                                      001049
00454    119*            II=IBDS(I,L1,L2)                                                   001054
00457    120*            DO 205 J=1,NBW                                                     001065
00461    121*        205 S(II,J)=0.0                                                        001086
```

```
      CELA                                                                    DATE 010978

00464     122*              S(IE,NHBW)=1.0                                    001067
00465     123*        210   S(IE,NBW)=SBD(I,L1,L2)                            001071
00467     124*        220   CALL BNDEQ(S,NRMAX,NCMAX,NPS,NHBW)                001075
00470     125*              DO 240 I=1,NPS                                    001103
00473     126*        240   STS(I,L1,L2)=S(I,NBW)                            001124
00475     127*              RETURN                                            001126
00476     128*              END                                               001235

        END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      CELD
```

CELD DATE 010978

```
@FOR,S    CELD,CELD
HSA  E3 -01/09/78-23:43:11 (4,)

   SUBROUTINE CELDIS    ENTRY POINT 001431

   STORAGE USED: CODE(1) 001456; DATA(0) 000100; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   SYSMTX 007641
    0004   CELZ1  000015
    0005   CELZ3  001140
    0006   ORTHPC 000060
    0007   MATRL  000005
    0010   NCASE  000003
    0011   GEOM1  000524
    0012   IMPCT1 006650
    0013   IMPCT3 000500
    0014   ELMLAG 000410
    0015   PARDOF 000002
    0016   DISPLC 001620
    0017   DISELM 000060
    0020   SOLDIS 000344
    0021   BODIS  000713
    0022   LAGRGN 000005

   EXTERNAL REFERENCES (BLOCK, NAME)

    0023   ELCELD
    0024   ELMTRX
    0025   ELDISA
    0026   BNDEQ
    0027   NERR2$
    0030   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000232 110L    0001   000256 115L    0001   000265 118L    0001   000006 142G    0001   000411 142L
    0001   000415 144L    0001   000007 145G    0001   000457 150L    0001   000022 155G    0001   000467 158L
    0001   000023 160G    0001   000535 165L    0001   000543 170L    0001   000610 171L    0001   000046 175G
    0001   000645 180L    0001   000720 185L    0001   001074 192L    0001   001100 193L    0001   001142 195L
    0001   001153 196L    0001   001221 200L    0001   001223 201L    0001   000067 203G    0001   001230 204L
    0001   001277 206L    0001   001304 208L    0001   000130 212G    0001   000141 217G    0001   001352 220L
    0001   000170 234G    0001   000205 241G    0001   000235 251G    0001   000252 260G    0001   000263 266G
    0001   000316 276G    0001   000330 304G    0001   000332 307G    0001   000363 324G    0001   000403 332G
    0001   000447 347G    0001   000502 363G    0001   000512 370G    0001   000016 40L     0001   000547 404G
    0001   000571 412G    0001   000614 421G    0001   000625 426G    0001   000654 441G    0001   000671 446G
    0001   000031 45L     0001   000037 46L     0001   000722 460G    0001   000730 464G    0001   000731 467G
    0001   001001 477G    0001   000106 48L     0001   001013 506G    0001   001015 511G    0001   001046 526G
    0001   001066 534G    0001   001132 551G    0001   001166 565G    0001   001176 572G    0001   001256 613G
    0001   001271 620G    0001   001331 635G    0001   001343 641G    0001   001376 653G    0001   000152 80L
    0001   000161 85L     0004   000001 A       0004 R 000002 ALPHA   0017 R 000030 BF      0015 R 001254 BFS
```

```
0012 R 001370 C        0014 R 000000 CD       0013 R 000040 CFRC     0006   000055 CK       0014 R 000200 CL
0014 R 000100 CS       0004 R 000013 DELT     0014 R 000250 DF       0020 R 000000 DIS      0021 R 000347 DISBD
0016 R 000000 DISN     0016 R 000344 DISPRV   0017 R 000000 DL       0006   000033 E        0014 R 000310 ELXYZ
0013 R 000030 FJCB     0012 R 001140 FJN      0012 R 001254 FJPRV    0000 I 000000 I        0021 I 000003 IBDIS
0021 I 000000 IBDIST   0000 I 000015 IE       0000 I 000014 II       0010 I 000000 INDEX    0000   000042 INJP$
0004 I 000011 INNER    0004 I 000000 IORDC    0022 I 000002 IORDL    0000 I 000010 IPARAM   0007   000004 ISOP
0007   000003 ISOT     0004 I 000012 ITER     0000 I 000001 J        0000 I 000013 K        0000 I 000006 KKK
0011 I 000000 KODE     0000 I 000004 LEE      0006 I 000022 LI       0000 I 000005 LIE      0000 I 000002 LIN
0015   000000 MAXDOF   0022 I 000003 MBW      0022 I 000004 MHBW     0000 I 000003 N        0010 I 000001 NASTRN
0004   000006 NBW      0000 I 000012 NC       0022 I 000001 NELT     0004 I 000010 NEPJ     0004 I 000003 NET
0007   000000 NFRAC    0004   000014 NFST     0015   000001 NGD      0004   000005 NHBW     0000 I 000007 NI
0011 I 000114 NOD      0004   000007 NPPJ     0004 I 000004 NPS      0022 I 000000 NPST     0000 I 000011 NR
0010   000002 NSLIDE   0003 I 007640 NSTAB    0013 R 000000 OM       0014 R 000300 PM       0007   000002 PMIU
0006   000044 PNIU     0003 R 000000 S        0014 R 000210 SF       0006   000011 SLC      0006   000000 SLT
0007   000001 TMIU     0012 R 000344 VERN     0005 R 000344 VERPRV   0014 R 000340 VL       0014 R 000220 W
0016 R 000710 X        0012 R 000000 XN       0005 R 000000 XPRV
```

```
00100      1*      C                                                                    000000
00100      2*      C*************************                                           000000
00100      3*      C                                                                    000000
00100      4*      C     ASSEMBLE SYSTEM MATRICES FOR DISPLACEMENTS IN CELFE ZONE       000000
00100      5*      C                                                                    000000
00100      6*      C*************************                                           000000
00100      7*      C                                                                    000000
00101      8*            SUBROUTINE CELDIS(L)                                           000000
00103      9*            INCLUDE PARAM1,LIST                                            000000
00103      9*      PARAM1 PROC                                                          000000
00103      9*      C                                                                    000000
00104      9*            PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                    000000
00105      9*            PARAMETER NPELG= 8,NPE= 8                                      000000
00106      9*            PARAMETER NDFRS=53,NDINT=10                                    000000
00106      9*      C                                                                    000000
00107      9*            PARAMETER NRMAX=2000,NCMAX=2                                   000000
00110      9*            PARAMETER NGPLG=2,NGP=2                                        000000
00111      9*            PARAMETER NCNT=6, NFRC=NPM                                     000000
00112      9*            PARAMETER IU=6,NDPUT= 10,NTSTEP=200                            000000
00113      9*            PARAMETER NBV=NPM,NBS=NPM,NBD=NPM                              000000
00113      9*       END                                                                 000000
00114     10*            INCLUDE DISPLV,LIST                                            000000
00114     11*      C                                                                    000000
00114     11*      DISPLV PROC                                                          000000
00115     11*            COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB                           000000
00116     11*            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER, 000000
00116     11*           1  ITER,DELT,NFST                                               000000
00117     11*            COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)                   000000
00120     11*            COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3) 000000
00121     11*            COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                       000000
00122     11*            COMMON /NCASE/ INDEX,NASTRN,NSLIDE                             000000
00123     11*            COMMON /CLOM1/ KODE(NPMT),NOD(NEMT,NPELG)                      000000
00124     11*            COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),  000000
00124     11*           1C(NPMT,NCNT,NCNT)                                              000000
00125     11*            COMMON /IMPCT2/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)      000000
00126     11*            COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),     000000
```

```
00126     11*          1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),                    000000
00126     11*         2VL(NPELG,IU)                                                                   000000
00127     11*           COMMON /PARDOF/ MAXDOF,NGD                                                    000000
00130     11*           COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMI,3)             000000
00131     11*           COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)                                       000000
00132     11*           COMMON /SOLDIS/ DIS(NPMT,3)                                                   000000
00133     11*           COMMON /BDDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)                            000000
00133     11*       END                                                                               000000
00134     12*           COMMON /LAGRGN/ NPST,NELT,IORDL,MBW, MHBW                                     000000
00134     13*       C                                                                                 000000
00134     14*       C                                                                                 000000
00135     15*           IF (NASTRN) 45,20,40                                                          000000
00140     16*        20 CONTINUE                                                                      000007
00141     17*           DO 30 I=1,NRMAX                                                               000007
00144     18*           DO 30 J=1,NCMAX                                                               000007
00147     19*        30 S(I,J)=0.0                                                                    000007
00152     20*           GO TO 45                                                                      000014
00153     21*        40 CONTINUE                                                                      000016
00154     22*           DO 41 I=1,NPST                                                                000016
00157     23*           DO 41 J=1,3                                                                   000023
00162     24*        41 DIS(I,J)=0.0                                                                  000023
00165     25*        45 CONTINUE                                                                      000031
00166     26*           IF (INDEX .EQ. 1) GO TO 46                                                    000031
00170     27*           LIN=1                                                                         000033
00171     28*           GO TO 170                                                                     000035
00172     29*        46 CONTINUE                                                                      000037
00173     30*           LIN=NET+1                                                                     000037
00174     31*           DO 165 N=1,NET                                                                000046
00177     32*           LEE=1                                                                         000051
00200     33*           IF (NASTRN .EQ. 1) LEE=3                                                      000053
00202     34*           DO 150 LIE=1,LEE                                                              000060
00205     35*           KKK=L                                                                         000067
00206     36*           IF (NASTRN .EQ. 1) KKK=LIE                                                    000071
00210     37*           GO TO (48,85),INNER                                                           000076
00211     38*        48 DO 80 I=1,NPE                                                                 000106
00214     39*           NI=NOD(N,I)                                                                   000130
00215     40*           FJCB(I)=FJPRV(NI)                                                             000131
00216     41*           DO 50 J=1,3                                                                   000141
00221     42*           ELXYZ(I,J)=XPRV(NI,J)                                                         000141
00222     43*           VL(I,J)=VERPRV(NI,J)/VERPRV(NI,5)                                             000142
00223     44*        50 CONTINUE                                                                      000146
00225     45*           IF(ITER  .GT. 1 .OR. N .GE. NEPJ) GO TO 80                                    000146
00227     46*           VL(I,3)=0.0                                                                   000150
00230     47*        80 CONTINUE                                                                      000157
00232     48*           GO TO 110                                                                     000157
00233     49*        85 DO 100 I=1,NPE                                                                000161
00236     50*           NI=NOD(N,I)                                                                   000170
00237     51*           FJCB(I)=ALPHA*FJPRV(NI)+(1.0-ALPHA)*FJN(NI)                                   000173
00240     52*           DO 100 J=1,3                                                                  000205
00243     53*           ELXYZ(I,J)=ALPHA*XPRV(NI,J)+(1.0-ALPHA)*XN(NI,J)                              000205
00244     54*           VL(I,J)=ALPHA*VERPRV(NI,J)/VERPRV(NI,5)+(1.0-ALPHA)*                          000212
00244     55*          1  VERN(NI,J)/VERN(NI,5)                                                      000212
00245     56*       100 CONTINUE                                                                     000235
00250     57*       110 DO 118 I=1,NPE                                                               000235
00253     58*           NI=NOD(N,I)                                                                   000240
00254     59*           IF(KODE(NI)) 112,112,115                                                      000243
```

```
00257     60*       112 DO 113 J=1,3                                                    000252
00262     61*       113 OM(I,J)=VL(I,J)                                                 000252
00264     62*           GO TO 118                                                       000254
00265     63*       115 DO 116 J=1,3                                                    000256
00270     64*       116 OM(I,J)=0.0                                                     000263
00272     65*       118 CONTINUE                                                        000267
00274     66*           CALL ELCELD(NGP,NPE,ELXYZ,SF,DF,W,VL,OM,FJCB,CD,CS,CL,DELT,KKK,  000267
00274     67*          1IORDC)                                                          000267
00275     68*           DO 120 I=1,NPE                                                  000310
00300     69*           NI=NOD(N,I)                                                     000316
00301     70*       120 PM(I)=DISPRV(NI,KKK)                                            000321
00303     71*           DO 140 I=1,NPE                                                  000332
00306     72*           DO 140 J=1,NPE                                                  000332
00311     73*       140 CL(I)=CL(I)+CD(I,J)*PM(J)                                       000332
00314     74*           IF (NASTRN .EQ. 0) GO TO 150                                    000343
00316     75*           IPARAM=2                                                        000345
00317     76*           IF (NASTRN .EQ. -1) IPARAM=1                                    000347
00321     77*           IF (NSTAB .NE. 0) GO TO 144                                     000354
00323     78*           DO 142 I=1,NPE                                                  000363
00326     79*           NI=NOD(N,I)                                                     000366
00327     80*           IF (KODE(NI) .LT. 2) GO TO 142                                  000370
00331     81*           DO 141 J=1,NPE                                                  000403
00334     82*       141 CS(I,J)=0.0                                                     000403
00336     83*           CS(I,I)=1.0                                                     000404
00337     84*           CL(I)=DISPRV(NI,KKK)                                            000406
00340     85*       142 CONTINUE                                                        000415
00342     86*       144 CONTINUE                                                        000415
00343     87*           CALL ELMTRX(CS,CL,NPS,NEMT,NPELG,NOD,N,CD,IBDIST,IBDIS,DISBD,NBD,  000415
00343     88*          1  LIE,IPARAM)                                                   000415
00344     89*           IF (NASTRN .NE. 1) GO TO 150                                    000434
00346     90*           DO 145 I=1,NPE                                                  000447
00351     91*           NR=NOD(N,I)                                                     000447
00352     92*       145 DIS(NR,KKK)=DIS(NR,KKK)+CL(I)                                   000452
00354     93*       150 CONTINUE                                                        000463
00356     94*           IF (NASTRN .EQ. 0) GO TO 158                                    000463
00360     95*           GO TO 165                                                       000465
00361     96*       158 CONTINUE                                                        000467
00362     97*           DO 160 I=1,NPE                                                  000467
00365     98*           NR=NOD(N,I)                                                     000502
00366     99*           S(NR,MBW)=S(NR,MBW)+CL(I)                                       000505
00367    100*           DO 160 J=1,NPE                                                  000512
00372    101*           NC=NOD(N,J)-NR+MHBW                                             000512
00373    102*       160 S(NR,NC)=S(NR,NC)+CS(I,J)                                       000517
00376    103*       165 CONTINUE                                                        000536
00400    104*           IF (LIN .GT. NELT) GO TO 201                                    000536
00402    105*       170 CONTINUE                                                        000543
00403    106*           DO 200 N=LIN,NELT                                               000543
00406    107*           LEE=1                                                           000552
00407    108*           IF (NASTRN .EQ. 1) LEE=3                                        000554
00411    109*           DO 195 LIE=1,LEE                                                000561
00414    110*           KKK=L                                                           000571
00415    111*           IF (NASTRN .EQ. 1) KKK=LIE                                      000573
00417    112*           GO TO (171,180),INNER                                           000600
00420    113*       171 DO 175 I=1,NPELG                                                000614
00423    114*           NI=NOD(N,I)                                                     000614
00424    115*           VL(I,5)=VERPRV(NI,5)                                            000617
```

```
00425    116*          DO 172 J=1,3                                                    000625
00430    117*          ELXYZ(I,J)=X(NI,J)                                              000625
00431    118*          BF(I,J)=BFS(NI,J)                                               000626
00432    119*          DL(I,J)=DISPRV(NI,J)                                            000630
00433    120*      172 VL(I,J)=VERPRV(NI,J)                                            000632
00435    121*      175 CONTINUE                                                        000643
00437    122*          GO TO 185                                                       000643
00440    123*      180 DO 184 I=1,NPELG                                                000645
00443    124*          NI=NOD(N,I)                                                     000654
00444    125*          VL(I,5)=ALPHA*VERPRV(NI,5)+(1.0-ALPHA)*VERN(NI,5)               000657
00445    126*          DO 184 J=1,3                                                    000671
00450    127*          ELXYZ(I,J)=X(NI,J)                                              000671
00451    128*          DL(I,J)=ALPHA*DISPRV(NI,J)+(1.0-ALPHA)*DISN(NI,J)               000672
00452    129*          VL(I,J)=ALPHA*VERPRV(NI,J)+(1.0-ALPHA)*VERN(NI,J)               000700
00453    130*      184 BF(I,J)=BFS(NI,J)                                               000706
00456    131*      185 CONTINUE                                                        000722
00457    132*          DO 187 I=1,NPELG                                                000722
00462    133*          NI=NOD(N,I)                                                     000722
00463    134*          DO 187 J=1,NCNT                                                 000723
00466    135*          DO 187 K=1,NCNT                                                 000731
00471    136*      187 CFRC(I,J,K)=C(NI,J,K)                                           000731
00475    137*          CALL ELDISA(NGPLG,NPELG,ELXYZ,SF,DF,W,DL,VL,BF,CD,CS,CL,LI,CFRC,  000747
00475    138*         1 NCNT,DELT,KKK,IORDL)                                           000747
00476    139*          DO 188 I=1,NPELG                                                000773
00501    140*          NI=NOD(N,I)                                                     001001
00502    141*          PM(I)=DISPRV(NI,KKK)                                            001004
00503    142*      188 CONTINUE                                                        001015
00505    143*          DO 189 I=1,NPELG                                                001015
00510    144*          DO 189 J=1,NPELG                                                001015
00513    145*      189 CL(I)=CL(I)+CD(I,J)*PM(J)                                       001015
00516    146*          IF (NASTRN .EQ. 0) GO TO 195                                    001026
00520    147*          IPARAM=2                                                        001030
00521    148*          IF (NASTRN .EQ. -1) IPARAM=1                                    001032
00523    149*          IF (NSTAB .NE. 0) GO TO 193                                     001037
00525    150*          DO 192 I=1,NPELG                                                001046
00530    151*          NI=NOD(N,I)                                                     001051
00531    152*          IF (KODE(NI) .LT. 2) GO TO 192                                  001053
00533    153*          DO 191 J=1,NPELG                                                001066
00535    154*      191 CS(I,J)=0.0                                                     001066
00540    155*          CS(I,I)=1.0                                                     001067
00541    156*          CL(I)=DISPRV(NI,KKK)                                            001071
00542    157*      192 CONTINUE                                                        001100
00544    158*      193 CONTINUE                                                        001100
00545    159*          CALL ELMTRX(CS,CL,NPST,NEMT,NPELG,NOD,N,CD,IBDIST,IBDIS,DISBD,  001100
00545    160*         1  NBD,LIE,IPARAM)                                              001100
00546    161*          IF (NASTRN .NE. 1) GO TO 195                                    001117
00550    162*          DO 194 I=1,NPELG                                                001132
00553    163*          NR=NOD(N,I)                                                     001132
00554    164*      194 DIS(NR,KKK)=DIS(NR,KKK)+CL(I)                                   001135
00556    165*      195 CONTINUE                                                        001147
00560    166*          IF (NASTRN .EQ. 0) GO TO 196                                    001147
00562    167*          GO TO 200                                                       001151
00563    168*      196 CONTINUE                                                        001153
00564    169*          DO 198 I=1,NPELG                                                001153
00567    170*          NR=NOD(N,I)                                                     001166
00570    171*          S(NR,NBW)=S(NR,NBW)+CL(I)                                       001171
```

```
CELD                                                                    DATE 010978

00571    172*          DO 198 J=1,NPELG                                  001176
00574    173*          NC=NOD(N,J)-NR+MHBW                               001176
00575    174*      198 S(NR,NC)=S(NR,NC)+CS(I,J)                         001203
00600    175*      200 CONTINUE                                          001223
00602    176*      201 CONTINUE                                          001223
00603    177*          IF (NASTRN .EQ. 0) GO TO 204                      001223
00603    178*    C                                                       001223
00605    179*          RETURN                                            001224
00605    180*    C                                                       001224
00606    181*      204 CONTINUE                                          001230
00607    182*          KKK=L                                             001230
00610    183*          IF (NSTAB .NE. 0) GO TO 208                       001231
00612    184*          DO 206 I=1,NPST                                   001256
00615    185*          IF (KODE(I) .LT. 2) GO TO 206                     001261
00617    186*          DO 205 J=1,MBW                                    001271
00622    187*      205 S(I,J)=0.0                                        001271
00624    188*          S(I,MHBW)=1.0                                     001272
00625    189*          S(I,MBW)=DISPRV(I,KKK)                            001274
00626    190*      206 CONTINUE                                          001304
00630    191*      208 CONTINUE                                          001304
00631    192*          IF(IBDIST(KKK) .EQ. 0) GO TO 220                  001304
00633    193*          II=IBDIST(KKK)                                    001306
00634    194*          DO 210 I=1,II                                     001321
00637    195*          IE=IBDIS(I,KKK)                                   001331
00640    196*          DO 209 J=1,MBW                                    001332
00643    197*      209 S(IE,J)=0.0                                       001343
00645    198*          S(IE,MHBW)=1.0                                    001344
00646    199*      210 S(IE,MBW)=DISBD(I,KKK)                            001346
00650    200*      220 CONTINUE                                          001352
00651    201*          CALL BNDEQ(S,NRMAX,NCMAX,NPST,MHBW)               001352
00652    202*          DO 250 I=1,NPST                                   001363
00655    203*      250 DIS(I,KKK)=S(I,MBW)                               001376
00657    204*          RETURN                                            001400
00660    205*          END                                               001455

      END OF COMPILATION:       NO  DIAGNOSTICS.

@HDG,P      CELFE
```

```
@FOR,S   CELFE,CELFE
HSA  E3 -10/31/78-20:56:12 (15,)

    MAIN PROGRAM

    STORAGE USED: CODE(1) 003571; DATA(0) 002461; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     C043   SYSMTX 007641
     C034   DATAX  000010
     C035   CELZ1  000015
     C036   CELZ3  001140
     C037   ORTHPC 000060
     C010   MATRL  000005
     C011   NCASE  000003
     C012   UNITCV 000015
     C013   GEOM1  000524
     C014   IMPCT1 006650
     C015   IMPCT3 006500
     C016   IMPCT4 006660
     C017   ELMLAG 000410
     C020   PARDOF 000002
     C021   LACRGN 000005
     C022   NODEPJ 000003
     C023   SRFACE 000077
     C024   CELZ2  000007
     C025   TSTEP  000312
     C026   SURFCE 000010
     C027   MATRL2 000230
     C030   PROJMT 000047
     C031   STRESS 002530
     C032   DISPLC 001620
     C033   STSELM 000110
     C034   DISELM 000060
     C035   IMPCT2 000230
     C036   SOLCNV 000574
     C037   SOLSTR 001254
     C040   SOLDIS 000344
     C041   BDVER  001375
     C042   BDSTR  002541
     C043   BDDIS  000713
     C044   DATE   000003

    EXTERNAL REFERENCES (BLOCK, NAME)

     C045   GEOMTR
     C046   STIFFG
     C047   CELTRN
     C050   CEL3CV
     C051   CELANI
     C052   CELDIS
     C053   EXIT
     C054   MESHUP
     C055   ALMS
```

```
C056   STRENG
C057   FRCRIT
C060   CEL3JB
C061   SLTEST
C062   NINTR$
C063   NRDC$
C064   NI02$
C065   NRDU$
C066   NI03$
C067   NPRT$
C070   NI01$
C071   SQRT
C072   NERR2$
C073   NWDC$
C074   NSTOP$
```

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

```
C001  000016 1L    0000  002122 1000F   0000  002123 1001F   0000  002125 1002F   0001  001350 1007G
C001  001356 1014G 0001  001366 1020G   0001  001404 1027G   0001  001406 1032G   0001  001431 1044G
C001  001437 1051G 0001  001461 1062G   0001  001467 1067G   0001  001470 1071G   0000  002126 1100F
C001  001513 1102G 0000  002127 1105F   0001  001527 1111G   0001  001544 1120G   0001  001553 1126G
C001  001574 1136G 0001  001602 1143G   0001  001606 1147G   0001  001631 1157G   0001  001637 1164G
C001  001640 1166G 0000  002130 1200F   0001  001670 1201G   0001  001671 1204G   0001  001733 1224G
C001  001773 1247G 0000  002132 1250F   0000  002136 1255F   0001  002013 1260G   0001  002027 1266G
C001  001655 130L  0000  002142 1300F   0001  002061 1304G   0001  002111 1326G   0001  002112 1331G
C001  002176 1344G 0001  001705 135L    0001  002210 1354G   0001  002236 1363G   0001  002246 1366G
C001  002276 1373G 0000  002143 1400F   0001  002315 1406G   0001  002316 1411G   0001  002317 1414G
C000  002145 1420F 0001  002362 1427G   0001  002363 1432G   0001  002414 1445G   0001  002444 1465G
C001  002445 1470G 0001  002457 1476G   0001  001766 150L    0000  002151 1500F   0001  002464 1503G
C001  002465 1506G 0001  002503 1515G   0000  002170 1520F   0001  002504 1526G   0000  002173 1530F
C001  002525 1532G 0001  002526 1535G   0000  002176 1550F   0001  002564 1551G   0001  002572 1556G
C001  002614 1567G 0001  002622 1574G   0001  002623 1576G   0000  002223 1580F   0000  002226 1582F
C000  002244 1585F 0000  002262 1600F   0001  002646 1607G   0000  002263 1612F   0001  002662 1615G
C001  002677 16256 0001  002706 1633G   0001  002727 1643G   0001  002735 1650G   0001  002741 1654G
C001  002767 1666G 0000  002307 1682F   0000  002314 1683F   0000  002315 1684F   0000  002325 1685F
C000  002335 1686F 0000  002345 1667F   0000  002355 1700F   0001  003007 1700G   0001  003015 1705G
C001  003016 1707G 0000  002363 1720F   0001  003565 2000L   0001  003205 2006G   0001  003206 2010G
C001  003223 2017G 0001  003227 2023G   0001  003250 2032G   0001  003254 2036G   0001  003267 2044G
C001  003276 2052G 0001  003314 2061G   0001  003325 2067G   0001  003333 2074G   0001  002011 210L
C001  003337 2100G 0001  003351 2106G   0001  003373 2114G   0001  003377 2120G   0001  003413 2127G
C001  003417 2133G 0001  003434 2142G   0001  003445 2150G   0001  003453 2155G   0001  003457 2161G
C001  003474 2170G 0001  003505 2176G   0000  002400 2200F   0001  003506 2200G   0001  003523 2207G
C001  003524 2211G 0001  003536 2217G   0001  003537 2222G   0001  003543 2226G   0001  000463 25L
C001  002043 260L  0001  000146 270G    0001  002055 270L    0001  000147 272G    0001  002075 320L
C001  000260 325G  0001  000337 346G    0001  000344 353G    0001  000357 361G    0001  000367 367G
C001  000402 375G  0001  000654 40L     0001  000403 400G    0001  002077 400L    0001  000414 407G
C001  000431 415G  0001  000442 423G    0001  000446 430G    0001  000447 433G    0001  000507 444G
C001  000513 450G  0001  000525 456G    0001  000547 464G    0001  000553 470G    0001  000567 477G
C001  000772 48L   0001  002102 500L    0001  000573 503G    0001  001116 51L     0001  000610 512G
C001  000621 520G  0001  002121 526L    0001  000627 525G    0001  002152 525L    0001  000633 531G
C001  002213 533L  0001  002301 535L    0001  001211 54L     0001  000650 540G    0001  002332 540L
C001  000664 546G  0001  000665 551G    0001  001275 56L     0001  000700 560G    0001  000701 563G
C001  000721 575G  0001  001341 60L     0001  000732 604G    0001  000733 607G    0001  002374 615L
C001  001010 625G  0001  001011 627G    0001  001026 636G    0001  001027 640G    0001  002417 645L
```

```
C001  001041 6466    0001  001054 6526    C001  001055 6556    0001  001075 6676    0001  001076 6726
C001  001102 6766    0001  002423 700L    C001  002434 701L    0001  001146 7066    0001  001173 7156
C001  002543 720L    C001  001202 7226    C001  001215 7326    0001  001224 7356    0001  003033 739L
C001  001257 7446    C001  001266 7516    C001  001304 7626    0001  001326 7716    0001  001335 7766
C001  003034 800L    C001  003560 820L    C001  003564 850L    0005 R 000001 A       0005 R 000002 ALPHA
C034  000030 BF      C032 R 001254 BFS     C014 R 001370 C       0017  000000 CD      0015  000040 CFRC
C016 R 000110 CINT    C007  000055 CK      CCC0 R 000245 CKF     0017  000200 CL      0026 R 000002 CONST1
C000 R 002100 CONST2  C017  000100 CS      C016 R 000044 CSPF    0016 R 000000 CSTF    0005 R 000013 DELT
C017  000250 DF      CC26 R 000001 DIFF    CC12 P 000013 DIFFS   0000 R 000000 DIM     0000 R 002113 DIML
C000 R 002112 DIMV    C024 R 000004 DIMX    C024 R 000005 DIMY    0024 R 000006 DIMZ    0047 R 000000 DIS
C043 R 000347 DISBD   L032 R 000000 DISN    C032 R 000344 DISPRV  0034  000000 DL      0025 R 000002 DT
C007  000033 E       CC03 R 000000 EF      CC17  000310 ELXYZ    0000 R 002076 FAC     0012 R 000007 FACTE
C012 R 000001 FACTL   C012 R 000005 FACTR   CC12 R 000011 FACTS   0012 R 000003 FACTV   0000 R 002075 FACTVO
C000 R 000131 FJ      CC15  000030 FJCB    CC14 R 001140 FJN     0014 R 001254 FJPRV   0000 I 002373 I
C043 I 000003 IBDIS   CC43 I 000000 IBDIST  CC42 I 000011 IBDS    0042 I 000000 IBDST   0041 I 000005 IBDV
C041 I 000000 IBDVT   C004 I 000007 ICOND   CC44  000000 IDATE   0011 I 000000 INDEX   0005 I 000011 INNER
C023 I 000065 INTF    C022 I 000002 INTFT   CC05 I 000000 IORDC   0000 I 002065 IORDER  0021 I 000002 IOROL
C010 I 000004 ISOP    C010 I 000003 ISOT    CC05 I 000012 ITER    0025 I 000000 ITGIV   0012 I 000000 IUNIT
CC00 I 002074 J       C000 I 002102 K       CC00 I 002106 KI      0000 I 002126 KK      0013 I 000000 KODE
CC00 I 002104 L       C000 I 002117 LEE     CC07  000022 LI      0000 I 002103 LIE     0000 I 002114 LL
C003 I 002115 L1      0000 I 002116 L2      CCC0 I 002107 M       0020 I 000000 MAXDOF  0000 I 002105 N
C011 I 000001 NASTRN  CC05 I 000006 NBW     CC21 I 000003 NBWT    0024 I 000003 NELG    0021 I 000001 NELT
C005 I 000010 NEPJ    CC22 I 000000 NEPJT   CC05 I 000003 NET     0024 I 000001 NETR    0000 I 002110 NF
C000 I 002072 NFLAG   C010 I 000000 NFRAC   CC23 I 000000 NFS     0005 I 000014 NFST    0020 I 000001 NGO
C005 I 000005 NHBW    CC21 I 000004 NHBWT   CCC0 I 002101 NI      0000 I 002121 NN      0013 I 000114 NOD
C027 I 000114 NOLFRC  0024 I 000002 NPLG    CC05 I 000007 NPPJ    0022 I 000001 NPPJT   0005 I 000004 NPS
C021 I 000000 NPST    CC24 I 000000 NPTR    CC11 I 000002 NSLIDE  0007 I 007640 NSTAB   0015  000000 OM
C030  000044 PCK      C037  000022 PE      CC17  000300 PM      0010 R 000002 PMIU    0000 R 000611 PNF
C007  000044 PNIU     C030  000033 PPNIU    CCC0 R 000015 PRS     0035 R 000000 PRSN    0035 R 001114 PRSPRV
C030  000011 PSLC     C030  000000 PSLT     CC04 R 000006 PYLD    0004 R 000001 RHOP    0004 R 000000 RHOT
C003  000000 S        C042 R 001265 SBD     CC17  000210 SF      0007  000011 SLC     0007  000000 SLT
C026 R 000000 SMIU    CC33  000000 STL     CC37 R 000000 STS     0031 R 000000 STSN    0031 R 001254 STSPRV
C025 R 000001 TIME    C000 R 000001 TITLE   CC10 R 000001 TMIU    0004 R 000005 TYLD    0004 R 000002 V
C041 R 000601 VBL     CC36 R 000000 VER     CC14 R 000344 VERN    0006 R 000344 VERPRV  0017  000340 VL
C026 R 000003 VMAX    C026 R 000004 VMIN    CC00 R 002077 VO      0017  000220 W       0032 R 000710 X
C026 R 000005 XDIM    C000 R 002111 XDL     CC14 R 000000 XN      0006 R 000000 XPRV    0027 R 000000 YIELD
```

```
001000      1*      C ***************************************************************      000000
001000      2*      C  *                                                             *      000000
001000      3*      C  ** CELFE - COUPLED EULERIAN-LAGRANGIAN FINITE ELEMENT PROGRAM FOR  *      000000
001000      4*      C  *  HIGH VELOCITY IMPACT ANALYSIS.  THE PRESENT PROGRAM INCLUDES THE *      000000
001000      5*      C  *  INTERFACING PROCEDURE FOR CELFE AND NASTRAN (15.7) PROGRAMS      *      000000
001000      6*      C  *  -- (CELFE/NASTRAN 1.0)  DEVELOPED BY CHUN-HIAN LEE, LOCKHEED    *      000000
001000      7*      C  *  MISSILES & SPACE CO., INC., HREC, HUNTSVILLE, ALA. DEC. 1977.   *      000000
001000      8*      C  *                                                             *      000000
001000      9*      C     THREE-DIMENSIONAL FINITE ELEMENT ANALYSIS OF HIGH VELOCITY IMPACT      000000
001000     10*      C  *                                                             *      000000
001000     11*      C ***************************************************************      000000
001000     12*      C                                                                      000000
000101     13*           INCLUDE PARAM1,LIST                                               000000
000102     13*      PARAM1 PROC                                                            000001
001000     13*      C                                                                      000001
001001     13*           PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                       000001
```

```
00104     13*           PARAMETER NPELG= 8,NPE= 8                                    000C01
00105     13*           PARAMETER NDFRS=53,NDINT=10                                  000001
00105     13*     C                                                                  000001
00106     13*           PARAMETER NRMAX=2000,NCMAX=2                                 000001
00107     13*           PARAMETER NGPLG=2,NGP=2                                      000C01
00110     13*           PARAMETER NCNT=6, NFRC=NPM                                   000C01
00111     13*           PARAMETER IU=5,NDPJT= 10,NTSTEP=200                          000C01
00112     13*           PARAMETER NBV=NPM,NBS=NPM,NBD=NPM                            000C01
00112     13*      END                                                               000C01
00113     14*           INCLUDE CELFE1,LIST                                          000001
00113     15*     C                                                                  000C01
00113     15*     CELFE1 PROC                                                        000001
00114     15*           DIMENSION TITLE(12)                                          000C01
00115     15*             DIMENSION PRS(NPM),FJ(NPM)                                 000001
00116     15*           DIMENSION CKF(NFRC,3),EF(NFRC,3,3),PNF(NFRC,3,3)             000001
00117     15*           DIMENSION IORDER(IU)                                         000C01
00117     15*     C                                                                  000001
00120     15*           COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB                         000001
00121     15*           COMMON /DATAX/ RHOT,RHOP,V(3),TYLD,PYLD,ICOND                000001
00122     15*           COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,  000001
00122     15*          1  ITER,DELT,NFST                                            000C01
00123     15*           COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)                 000C01
00124     15*           COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)  000001
00125     15*           COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                     000C01
00126     15*           COMMON /NCASE/ INDEX,NASTRN,NSLIDE                           000C01
00127     15*           COMMON /UNITCV/ IUNIT,FACTL(2),FACTV(2),FACTR(2),FACTE(2),   000C01
00127     15*          1 FACTS(2),DIFFS(2)                                           000C01
00130     15*           COMMON /CCOM1/ KODE(NPMT),NOD(NPMT,NPELG)                    000C01
00131     15*           COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),  000C01
00131     15*          1C(NPMT,NCNT,NCNT)                                            000001
00132     15*           COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)    000C01
00133     15*           COMMON /IMPCT4/ CSTF(NCNT,NCNT),CSPF(NCNT,NCNT),             000C01
00133     15*          1CINT(NDINT,NCNT,NCNT)                                        000C01
00134     15*           COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),   000C01
00134     15*          1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),   000001
00134     15*          2VL(NPELG,IU)                                                 000C01
00135     15*           COMMON /PARDOF/ MAXDOF,NGD                                   000C01
00136     15*           COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                   000C01
00137     15*           COMMON /NODEPJ/ NEPJT,NPPJT,INTFT                            000C01
00140     15*           COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)                        000C01
00141     15*           COMMON /CELZ2/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ            000C01
00142     15*           COMMON /TSTEP/ ITGIV,TIME,DT(NTSTEP)                         000C01
00143     15*           COMMON /SURFCE/ SMIU,DIFF,CONST1,VMAX,VMIN,XDIM(3)           000C01
00144     15*           COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)                     000C01
00145     15*           COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3)  000C01
00146     15*           COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)                000001
00147     15*           COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)  000C01
00150     15*           COMMON /STSELM/ STL(NPE,3,3)                                 000C01
00151     15*           COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)                      000C01
00152     15*           COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)                        000C01
00153     15*           COMMON /SOLCNV/ VER(NPMT,IU)                                 000C01
00154     15*           COMMON /SOLSTR/ STS(NPM,3,3)                                 000C01
00155     15*           COMMON /SOLDIS/ DIS(NPMT,3)                                  000C01
00156     15*           COMMON /BDVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)            000C01
00157     15*           COMMON /BDSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)         000C01
00160     15*           COMMON /BDDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)           000C01
```

```
00161    15*           COMMON /DATE/ IDATE(3)                                    000001
00162    15*           EQUIVALENCE (S(1),EF(1))                                  000001
00162    15*      END                                                            000001
00163    16*           DATA IORDER/1,2,3,4,5/                                    000001
00163    17*      C                                                              000001
00163    18*      C                                                              000001
00163    19*      C**********************                                       000001
00163    20*      C                                                              000001
00163    21*      C     NFLAG = 0     1ST XQT, READ DATA                         000001
00163    22*      C     NFLAG =+1     2ND XQT, GO TO LOOP1                       000001
00163    23*      C     NFLAG =-1     3RD XQT, GO TO LOOP2                       000001
00163    24*      C                                                              000001
00163    25*      C                                                              000001
00163    26*      C     NASTRN = 0 -----; IN-CORE CELFE RUN                      000001
00163    27*      C            =+1 ----- CELFE/NASTRAN RUN                       000001
00163    28*      C            =-1 ----- CELFE RUN USING NASTRAN SUBROUTINE SOLVE ONLY  000001
00163    29*      C                                                              000001
00163    30*      C                                                              000001
00163    31*      C     ISOT = 0, ISOTROPIC,                                     000001
00163    32*      C          = 1, UNIDIRECTIONAL FIBER COMPOSITES                000001
00163    33*      C          = 2,3,... FOR OTHER ANISOTROPIC MATERIALS           000001
00163    34*      C                                                              000001
00163    35*      C**********************                                       000001
00163    36*      C                                                              000001
00163    37*      C                                                              000001
00165    38*           READ 1002, NASTRN                                         000001
00170    39*           READ 1002, NFLAG                                          000006
00173    40*           IF (NFLAG .NE. 0) GO TO 400                               000013
00175    41*         1 READ (5,1001,END=2000)TITLE                               000016
00200    42*           PRINT 1001, TITLE                                         000026
00203    43*           INDEX = 1                                                 000035
00203    44*      C                                                              000035
00203    45*      C                                                              000035
00203    46*      C     READ DATA                                                000035
00203    47*      C                                                              000035
00203    48*      C                                                              000035
00204    49*           READ 1100,IORDC,IORDL,NFST,INTFT,ISOT,ISOP,IUNIT,ITGIV,ICOND,ITER  000037
00204    50*          1 ,NSLIDF                                                   000037
00221    51*           READ 1400,A,ALPHA,TIME                                    000056
00226    52*           READ 1200, (V(I),I=1,3),DIMX,DIMY,DIMZ                    000065
00234    53*           READ 1200, RHOT,RHOP,TMIU,PMIU,TYLD,PYLD,SHIU             000100
00245    54*           READ  1100, NPS,NET,NPPJ,NEPJ,NPPJT,NEPJT,NPTR,NETR,NPLG,NELG,  000113
00245    55*          1NPST,NELT,MAXDOF,NGD                                      000113
00265    56*           READ 1100, (IBDVT(I),I=1,IU),((IBDST(I,J),J=1,3),I=1,3),  000135
00265    57*          1(IBDIST(I),I=1,3)                                         000135
00265    58*      C                                                              000135
00265    59*      C**********************************                           000135
00265    60*      C                                                              000135
00265    61*      C     NPS ------ TOTAL NODES IN IMPACT ZONE                    000135
00265    62*      C     NET ------ TOTAL ELEMENTS IN IMPACT ZONE                 000135
00265    63*      C     NPPJ ------ NO. OF NODE BEGINNING THE PROJECTILE         000135
00265    64*      C     NEPJ ------ NO. OF ELEMENT BEGINNING THE PROJECTILE      000135
00265    65*      C     NPPJT ----- TOTAL NODES OF PROJECTILE                    000135
00265    66*      C     NEPJT ----- TOTAL ELEMENTS OF PROJECTILE                 000135
00265    67*      C     NPTR ------ NO. OF NODE BEGINNING THE TRANSITION LAYER   000135
00265    68*      C     NETR ------ NO. OF ELEMENT BEGINNING THE TRANSITION LAYER  000135
```

```
00265    69*     C     NPLG -+--- NO. OF NODE BEGINNING THE LAGRANGIAN ZONE                000135
00265    70*     C     NELE ----- NO. OF ELEMENT BEGINNING THE LAGRANGIAN ZONE              000135
00265    71*     C     NPST ----- TOTAL NODES IN COMPLETE STRUCTURE                        000135
00265    72*     C     NELT ----- TOTAL ELEMENTS IN COMPLETE STRUCTURE                     000135
00265    73*     C                                                                         000135
00265    74*     C*****************************                                          000135
00265    75*     C                                                                         000135
00265    76*     C     KODE(I) = 0 --------- EULERIAN NODES                                000135
00265    77*     C             = 1 --------- MOVING NODES IN COUPLING ZONE (INCLUDING      000135
00265    78*     C                           FREE SURFACE IN EULERIAN ZONE)                000135
00265    79*     C             = 2 --------- INTERFACE NODES BETWEEN IMPACT AND           000135
00265    80*     C                           LAGRANGIAN ZONES                              000135
00265    81*     C             = 3 --------- LAGRANGIAN NODES                             000135
00265    82*     C                                                                         000135
00265    83*     C*****************************                                          000135
00265    84*     C                                                                         000135
00300    85*           FACTV0=1.0UE-6                                                     000161
00301    86*           FACTL(1)=1.0                                                       000163
00302    87*           FACTV(1)=1.0                                                       000165
00303    88*           FACTR(1)=1.0                                                       000167
00304    89*           FACTE(1)=1.0                                                       000171
00305    90*           FACTS(1)=1.0                                                       000173
00306    91*           DIFFS(1)=1.0                                                       000175
00307    92*           FACTL(2)=2.54                                                      000177
00310    93*           FACTV(2)=FACTL(2)                                                  000201
00311    94*           FACTR(2)=27.67990                                                  000202
00312    95*           FACTE(2)=FACTV(2)*FACTV(2)                                         000204
00313    96*           FACTS(2)=100000000.0/6.894757                                      000207
00314    97*           FAC=1.00E-12                                                       000211
00315    98*           DIFFS(2)=12.0*32.174*FAC                                           000213
00315    99*     C                                                                        000213
00315   100*     C                                                                        000213
00315   101*     C     MESH GENERATION                                                    000213
00315   102*     C                                                                        000213
00315   103*     C*********************************                                     000213
00315   104*     C                                                                        000213
00316   105*           DIM=DIMX                                                           000215
00317   106*           IF (DIM .GE. DIMY) DIM=DIMY                                        000217
00321   107*           IF (DIM .GE. DIMZ) DIM=DIMZ                                        000225
00323   108*           CALL GEOMTR(X)                                                     000233
00324   109*           DO 12 J=1,3                                                        000236
00327   110*           V(J)=V(J)*FACTV0                                                   000260
00330   111*           XDIM(J)=DIM                                                        000262
00331   112*       12  CONTINUE                                                         000265
00332   113*           DIFF=1.0                                                           000265
00334   114*           VMAX=V0*RHOT                                                       000267
00335   115*           VMIN=0.0                                                           000272
00336   116*           V0=SQRT(V(1)**2+V(2)**2+V(3)**2)                                   000273
00337   117*           CONST1=RHOT*V0                                                     000310
00340   118*           CONST2=0.50*CONST1*V0                                              000312
00341   119*           NBW=2*NHBW                                                         000315
00342   120*           NBWT=2*NHBWT                                                       000320
00342   121*     C                                                                        000320
00342   122*     C                                                                        000320
00342   123*     C     INITIAL CONDITIONS                                                 000320
00342   124*     C                                                                        000320
```

```
00342    125*     C                                                              000320
00343    126*           IF (ICOND .EQ. 0) GO TO 25                               000323
00345    127*           DO 14 I=1,NPST                                           000337
00350    128*           YIELD(I)=TYLD                                            000337
00351    129*           VER(I,5)=RHOT                                            000340
00352    130*           DO 14 J=1,4                                              000344
00355    131*        14 VER(I,J)=0.0                                             000344
00360    132*           DO 15 I=NPPJ,NPPJT                                       000357
00363    133*           YIELD(I)=PYLD                                            000357
00364    134*           VER(I,4)=0.50*RHOP*V0**2                                 000360
00365    135*           VER(I,5)=RHOP                                            000362
00366    136*           DO 15 J=1,3                                              000367
00371    137*        15 VER(I,J)=RHOP*V(J)                                       000367
00374    138*           DO 16 I=1,NPST                                           000403
00377    139*           DO 16 J=1,3                                              000403
00402    140*           DIS(I,J)=0.0                                             000403
00403    141*        16 BFS(I,J)=0.0                                             000403
00406    142*           DO 17 I=1,INTFT                                          000414
00411    143*           NI=INTF(I)                                               000414
00412    144*           VER(NI,4)=0.50*RHOP*V0**2                                000415
00413    145*           VER(NI,5)=RHOP                                           000420
00414    146*           DO 17 J=1,3                                              000431
00417    147*        17 VER(NI,J)=RHOP*V(J)                                      000431
00422    148*           DO 20 I=1,NPS                                            000442
00425    149*           FJ(I)=1.0                                                000442
00426    150*           PRS(I)=0.0                                               000443
00427    151*           DO 20 J=1,3                                              000447
00432    152*           DO 20 K=1,3                                              000447
00435    153*        20 STS(I,J,K)=0.0                                           000447
00439    154*     C                                                              000447
00441    155*           GO TO 40                                                 000461
00442    156*        25 CONTINUE                                                 000463
00443    157*           DO 28 J=1,IU                                             000463
00446    158*        28 READ 2200, (VER(I,J),I=1,NPST)                           000507
00455    159*           DO 30 J=1,3                                              000525
00460    160*           LIE=1                                                    000525
00461    161*           IF (ISOT .GE. 0) LIE=J                                   000527
00463    162*           DO 30 K=LIE,3                                            000533
00466    163*        30 READ 2200,(STS(I,J,K),I=1,NPS)                           000547
00476    164*           DO 32 J=1,3                                              000567
00501    165*        32 READ 2200, (DIS(I,J),I=1,NPST)                           000567
00516    166*           READ 2200, (PRS(I),I=1,NPS)                              000602
00516    167*           READ 2200, (FJ(I),I=1,NPS)                               000613
00524    168*           DO 35 J=1,3                                              000627
00527    169*        35 READ 2200, (BFS(I,J),I=1,NPST)                           000627
00536    170*           READ 2200, (YIELD(I),I=1,NPST)                           000642
00536    171*     C                                                              000642
00544    172*        40 CONTINUE                                                 000654
00545    173*           DO 44 I=1,NPST                                           000654
00550    174*           DO 44 L=1,IU                                             000665
00553    175*           VERN(I,L)=VER(I,L)                                       000665
00554    176*        44 VERPRV(I,L)=VER(I,L)                                     000666
00557    177*           DO 45 I=1,NPST                                           000701
00562    178*           DO 45 J=1,3                                              000701
00565    179*           DISN(I,J)=DIS(I,J)                                       000701
00566    180*           DISPRV(I,J)=DIS(I,J)                                     000702
```

```
00567   181*            XN(I,J)=X(I,J)                                          000704
00570   182*            XPRV(I,J)=X(I,J)                                        000706
00571   183*        45  CONTINUE                                                000721
00574   184*            DO 47 I=1,NPS                                           000721
00577   185*            FJN(I)=FJ(I)                                            000721
00600   186*            FJPRV(I)=FJ(I)                                          000722
00601   187*            PRSN(I)=PRS(I)                                          000724
00602   188*            PRSPRV(I)=PRS(I)                                        000726
00603   189*            DO 47 J=1,3                                             000733
00606   190*            DO 47 K=1,3                                             000733
00611   191*            STSN(I,J,K)=STS(I,J,K)                                  000733
00612   192*        47  STSPRV(I,J,K)=STS(I,J,K)                               000734
00612   193*    C                                                               000734
00612   194*    C                                                               000734
00612   195*    C       INITIAL MATERIAL PROPERTIES                             000734
00612   196*    C                                                               000734
00612   197*    C                                                               000734
00616   198*            IF (ITER .NE. 0) GO TO 48                               000750
00620   199*            CALL STIFFG(NPMT,C,NPPJ,NPPJT,INTF,INTFT,NPST,CSTF,CSPF,NCNT,   000752
00620   200*           1  CINT,KODE)                                            000752
00621   201*            GO TO 51                                                000770
00622   202*        48  CONTINUE                                                000772
00623   203*            READ 2200, ((CSTF(I,J),J=1,NCNT),I=1,NCNT)              000772
00634   204*            READ 2200, ((CSPF(I,J),J=1,NCNT),I=1,NCNT)              001020
00645   205*            DO 49 K=1,INTFT                                         001041
00650   206*            N=INTF(K)                                               001041
00651   207*            DO 49 I=1,NCNT                                          001042
00654   208*            DO 49 J=1,NCNT                                          001055
00657   209*            CINT(N,I,J)=CSTF(I,J)                                   001055
00660   210*            IF (KODE(N) .EQ. 0) CINT(N,I,J)=CSPF(I,J)               001056
00662   211*        49  CONTINUE                                                001076
00666   212*            DO 50 J=1,NCNT                                          001076
00671   213*            DO 50 K=1,NCNT                                          001076
00674   214*        50  READ 2200, (C(I,J,K),I=1,NPST)                        001076
00704   215*        51  CONTINUE                                                001116
00704   216*    C                                                               001116
00704   217*    C                                                               001116
00704   218*    C       GENERATE BOUNDARY CONDITIONS                            001116
00704   219*    C                                                               001116
00704   220*    C                                                               001116
00705   221*            DO 54 J=1,IU                                            001116
00710   222*            N=IBDVT(J)                                              001155
00711   223*            IF (N.EQ.0) GO TO 54                                    001157
00713   224*            READ 1100,(IBDV(I,J),I=1,N)                             001161
00721   225*            DO 53 K=1,N                                             001202
00724   226*            KI=IBDV(K,J)                                            001202
00725   227*        53  VRD(K,J)=VER(KI,J)                                     001203
00727   228*        54  CONTINUE                                                001215
00731   229*            DO 56 J=1,3                                             001215
00734   230*            DO 56 K=1,3                                             001224
00737   231*            N= IBDST(J,K)                                           001241
00740   232*            IF (N .EQ. 0) GO TO 56                                  001243
00742   233*            READ 1100,( IBDS(I,J,K),I=1,N)                          001245
00750   234*            DO 55 M=1,N                                             001266
00753   235*            KI=IBDS(M,J,K)                                          001266
00754   236*        55  SRD(M,J,K)=STS(KI,J,K)                                 001267
```

```
00756   237*          56 CONTINUE                                              001304
00761   238*             DO 60 J=1,3                                           001304
00764   239*             N=IBDIST(J)                                           001310
00765   240*             IF(N .EQ. 0) GO TO 60                                 001312
00767   241*              READ 1100,(IBDIS(I,J),I=1,N)                         001314
00775   242*             DO 59 K=1,N                                           001335
01000   243*             KI=IBDIS(K,J)                                         001335
01001   244*          59 DISBC(K,J)=0.0                                        001336
01003   245*          60 CONTINUE                                              001342
01003   246*       C                                                           001342
01003   247*       C                                                           001342
01005   248*             READ 1400,(DT(I),I=1,ITGIV)                           001342
01013   249*             DO 92 I=1,INTFT                                       001356
01016   250*             NI=INTF(I)                                            001356
01017   251*             DO 90 J=1,3                                           001357
01022   252*          90 VER(NI,J)=V(J)/FACTVO                                 001366
01024   253*          92 VER(NI,4)=VER(NI,4)/DIFFS(IUNIT)                      001371
01026   254*             DO 100 I=NPPJ,NPPJT                                   001406
01031   255*             DO 95 J=1,3                                           001406
01034   256*          95 VER(I,J)=V(J)/FACTVO                                  001406
01036   257*         100 VER(I,4)=VER(I,4)/DIFFS(IUNIT)                        001411
01036   258*       C   *******                                                 001411
01036   259*       C                                                           001411
01036   260*       C    ASSEMBLE AND SOLVE SYSTEM OF EQUATIONS                 001411
01036   261*       C    ITERATION WITH TIME                                    001411
01036   262*       C                                                           001411
01036   263*       C*********                                                  001411
01040   264*             PRINT 1500, TIME                                      001420
01043   265*             DO 120 I=1,NPS                                        001431
01046   266*         120 PRINT 1520, I,(VER(I,J),J=4,IU),PRS(I)                001431
01057   267*             PRINT 1550                                            001451
01061   268*             DO 121 I=1,NPS                                        001461
01064   269*         121 PRINT 1580, I,((STS(I,J,K),K=1,3),J=1,3)              001461
01077   270*             PRINT 1585                                            001504
01101   271*             DO 122 I=1,NFST                                       001513
01104   272*             NF=NFS(I)                                             001513
01105   273*         122 PRINT 1520,I,(X(NF,J),J=1,3)                          001515
01115   274*             PRINT 1582                                            001534
01117   275*             DO 123 I=1,NPST                                       001544
01122   276*         123 PRINT 1530, I,KODE(I),(X(I,J),J=1,3)                  001544
01133   277*             PRINT 1612                                            001563
01135   278*             DO 124 I=1,NPST                                       001574
01140   279*         124 PRINT 1520,I,(DIS(I,J),J=1,3),(VER(I,J),J=1,3)        001574
```

```
01154   280*          PRINT 1686                                              001621
01156   281*          DO 125 I=1,NPST                                         001631
01161   282*      125 PRINT 1687, I,((C(I,J,K),K=1,NCNT),J=1,NCNT)            001631
01174   283*      130 CONTINUE                                                001655
01175   284*          ITER=ITER+1                                             001655
01176   285*          IF (ITER .GT. ITGIV) GO TO 800                          001657
01176   286*    C                                                             001657
01200   287*          DO 132 I=1,NPST                                         001664
01203   288*          DO 132 J=1,3                                            001671
01206   289*          DIS(I,J)=0.0                                            001671
01207   290*          DISN(I,J)=0.0                                           001671
01210   291*      132 DISPRV(I,J)=0.0                                         001672
01213   292*          TIME=TIME+DT(ITER)                                      001700

01214   293*          INNER=0                                                 001703
01215   294*      135 INNER=INNER+1                                           001705
01216   295*          IF (INNER .EQ. 1) DELT=A*DT(ITER)                       001707
01220   296*          IF (INNER .EQ. 2) DELT=DT(ITER)                         001720
01222   297*          XDL=XDIM(1)                                             001726
01223   298*          DO 138 I=2,3                                            001733
01226   299*          IF (XDL .GE. XDIM(I)) XDL=XDIM(I)                       001733
01230   300*      138 CONTINUE                                                001742
01232   301*          NSTAB=0                                                 001742
01233   302*          DIMV=TIME*VO                                            001743
01234   303*          DIML=DIM-XDL                                            001746
01235   304*          IF (DIMV .GT. DIML) NSTAB=1                             001751
01237   305*          IF(NASTRN)150,150,140                                   001757
01237   306*    C                                                             001757
01237   307*    CLOOP2                                                        001757
01237   308*    C                                                             001757
```

```
01242    309*      140 CONTINUE                                                  001762
01243    310*          CALL CELTRN(1)                                            001762
01243    311*    C                                                               001762
01244    312*      150 CONTINUE                                                  001766
01245    313*          CALL CELTRN(5)                                            001766
01246    314*          DO 200 LL=1,10                                            001773
01251    315*          L=IORDER(LL)                                              001773
01252    316*          CALL CEL3CV(L)                                            001774
01253    317*      200 CONTINUE                                                  002000
01255    318*          GO TO (210,260,850),INDEX                                 002000
01256    319*      210 CONTINUE                                                  002013
01257    320*          DO 250 L1=1,3                                             002013
01262    321*          LIE=1                                                     002013
01263    322*          IF(ISOT .GE. 0) LIE=L1                                    002015
01265    323*          DO 250  L2=LIE,3                                          002022
01270    324*          CALL CELANI(INTF,INTFT,CINT,L1,L2)                        002027
01271    325*      250 CONTINUE                                                  002043
01274    326*      260 CONTINUE                                                  002043
01275    327*          LEE=3                                                     002043
01276    328*          IF (NASTRN .NE. 1) GO TO 270                              002044
01300    329*          LEE=1                                                     002047
01301    330*          CALL CELTRN(2)                                            002051
01302    331*      270 CONTINUE                                                  002055
01302    332*          DO 300 KK=1,LEE                                           002055
01306    333*          CALL CELDIS(KK)                                           002061
01307    334*      300 CONTINUE                                                  002066
01311    335*          IF (NASTRN) 320,500,310                                   002066
01314    336*      310 CONTINUE                                                  002071
01315    337*          CALL CELTRN(3)                                            002071
01316    338*      320 CONTINUE                                                  002075
01317    339*          CALL EXIT                                                 002075
01317    340*    C                                                               002075
01317    341*    C LOOP1                                                         002075
01317    342*    C                                                               002075
01317    343*    C     READ VECTOR IN FROM NASTRAN                               002075
01317    344*    C                                                               002075
01320    345*      400 CONTINUE                                                  002077
01321    346*          CALL CELTRN(4)                                            002077
01321    347*    C                                                               002077
01322    348*      500 CONTINUE                                                  002102
```

```
01323    349*          IF (INNER .NE. 1) GO TO 520                                    002102
01329    350*          DO 510 I=1,NPST                                                002112
0133C    351*          DO 510 J=1,IU                                                  002112
01333    352*      510 VERN(I,J)=VER(I,J)                                             002112
01336    353*      520 CONTINUE                                                       002121
01337    354*          CALL MESHUP(NPMT,NEMT,NPELG,NPE,IU,IORDC,NET,NPST,DELT,KODE,   002121
01337    355*         1  NOE,DIS,X,XPRV,VERPRV)                                       002121
0134C    356*          GO TO (525,645,850),INDEX                                      002141
01341    357*      525 CONTINUE                                                       002152
01342    358*          CALL ALMS(NPMT,NPS,NPPJ,NPPJT,INTF,INTFT,NFRAC,RHOT,RHOP,VER,PRS)  002152
01343    359*          DO 530 I=1,NFST                                                002166
01346    360*          NF=NFS(I)                                                      002176
01347    361*      530 PRS(NF)=0.0                                                    002177
01351    362*          IF(INNER .NE. 1) GO TO 533                                     002202
01353    363*          DO 532 J=1,NPS                                                 002210
01356    364*      532 PRSN(J)=PRS(J)                                                 002210
0136C    365*      533 CONTINUE                                                       002213
01361    366*          CALL STRENG(NPM,STS,YIELD,NFS,INTF,KODE)                       002213
01362    367*          DO 536 N=1,NPS                                                 002222
01365    368*          DO 536 I=1,3                                                   002246
0137C    369*          IF (ISOT .LT. 0) GO TO 535                                     002266
01372    370*          DO 534 J=I,3                                                   002276
01375    371*      534 STSN(N,J,I)=STS(N,I,J)                                         002276
01377    372*      535                 STSN(N,I,I)=STS(N,I,I)-PRS(N)                  002301
0140C    373*      536 CONTINUE                                                       002307
01403    374*          IF(INNER .NE. 1) GO TO 540                                     002307
01405    375*          DO 538 N=1,NPS                                                 002317
0141C    376*          DO 538 I=1,3                                                   002317
01413    377*          DO 538 J=1,3                                                   002317
01416    378*      538 STSN(N,I,J)=STS(N,I,J)                                         002317
01422    379*      540 CONTINUE                                                       002332
01423    380*          CALL FRCRIT(NPMT,NPM,STS,INTF,C,CSTF,CSPF,NCNT,NFRC,NODFRC,    002332
01423    381*         1  NFS,EF,PNF,CRF,KODE)                                         002332
01424    382*          IF (INNER .NE. 1) GO TO 615                                    002352
01426    383*          DO 610 I=1,NPST                                                002363
01431    384*          DO 610 J=1,3                                                   002363
01434    385*          DISN(I,J)=DIS(I,J)                                             002363
01435    386*      610 XN(I,J)=X(I,J)                                                 002364
0144C    387*      615 CONTINUE                                                       002374
01441    388*          CALL CEL3JB(NPMT,VERPRV,VERN,VER,FJPRV,FJ,KODE)                002374
01442    389*          IF(INNER .NE. 1) GO TO 645                                     002404
01444    390*          DO 642 I=1,NPS                                                 002414
01447    391*      642 FJN(I)=FJ(I)                                                   002414
01451    392*      645 CONTINUE                                                       002417
01452    393*          IF(INNER .EQ. 2)   GO TO 700                                   002417
01454    394*          GO  TO 135                                                     002421
01455    395*      700 CONTINUE                                                       002423
01456    396*          IF (INDEX .GT. 1) GO TO 701                                    002423
0146C    397*          CALL SLTEST                                                    002426
01461    398*          IF (INDEX .EQ. 3) GO TO 820                                    002430
01461    399*     C                                                                   002430
01463    400*      701 CONTINUE                                                       002434
01464    401*          DO 705 I=1,NPST.                                               002434
01467    402*          DO 705 J=1,IU                                                  002445
01472    403*      705 VERPRV(I,J)=VER(I,J)                                           002445
01475    404*          DO 708 I=1,NPS                                                 002457
```

```
01500   405*          FJPRV(I)=FJ(I)                                                  002457
01501   406*          PRSPRV(I)=PRS(I)                                                002460
01502   407*          DO 708 J=1,3                                                    002465
01505   408*          DO 708 K=1,3                                                    002465
01510   409*      708 STSPRV(I,J,K)=STS(I,J,K)                                        002465
01514   410*          DO 710 I=1,NPST                                                 002504
01517   411*          DO 710 J=1,3                                                    002504
01522   412*          XPRV(I,J)=X(I,J)                                                002504
01523   413*          DISPRV(I,J)=DIS(I,J)                                            002505
01524   414*      710 CONTINUE                                                        002514
01527   415*          IF(IUNIT .EQ. 1) GO TO 720                                      002514
01531   416*          DO 715 I=1,NPST                                                 002517
01534   417*          DO 712 J=1,3                                                    002526
01537   418*      712 VER(I,J)=1000000.0*VERPRV(I,J)/VERPRV(I,5)                      002526
01541   419*          VER(I,4)=VERPRV(I,4)/DIFFS(IUNIT)                               002532
01542   420*      715 CONTINUE                                                        002543
01544   421*      720 CONTINUE                                                        002543
01545   422*          PRINT 1500, TIME                                                002543
01550   423*          DO 725 I=1,NPS                                                  002564
01553   424*      725 PRINT 1520, I,(VER(I,J),J=4,IU),PRS(I)                          002564
01564   425*          PRINT 1550                                                      002604
01566   426*          DO 730 I=1,NPS                                                  002614
01571   427*      730 PRINT 1580, I,((STS(I,J,K),K=1,3),J=1,3)                        002614
01604   428*          PRINT 1585                                                      002637
01606   429*          DO 731 I=1,NFST                                                 002646
01611   430*          NF=NFS(I)                                                       002646
01612   431*      731 PRINT 1520,I,(X(NF,J),J=1,3)                                    002650
01622   432*          PRINT 1582                                                      002667
01624   433*          DO 732 I=1,NPST                                                 002677
01627   434*      732 PRINT 1530,  I,KODE(I),(X(I,J),J=1,3)                           002677
01640   435*          PRINT 1612                                                      002716
01642   436*          DO 733 I=1,NPST                                                 002727
01645   437*      733 PRINT 1520, I,(DIS(I,J),J=1,3),(VER(I,J),J=1,3)                 002727
01661   438*          IF(NFRAC .EQ. 0) GO TO 739                                      002754
01663   439*          PRINT 1682                                                      002756
01665   440*          DO 735 I=1,NFRAC                                                002762
01670   441*      735 PRINT 1683, I,NODFRC(I)                                         002767
01675   442*          PRINT 1686                                                      002777
01677   443*          DO 738 I=1,NPST                                                 003007
01702   444*      738 PRINT 1687, I,((C(I,J,K),K=1,NCNT),J=1,NCNT)                    003007
01715   445*      739 CONTINUE                                                        003033
01715   446*        C                                                                 003033
01716   447*          GO TO 130                                                       003033
01717   448*      800 PRINT 1720                                                      003034
01721   449*          ICOND=0                                                         003037
01722   450*          PUNCH 1100, IORDC,IORDL,NFST,INTFT,ISOT,ISOP,IUNIT,ITGIV,ICOND  003040
01722   451*         1  ,ITER,NSLIDE                                                  003040
01731   452*          PUNCH 1400, A,ALPHA,TIME                                        003057
01744   453*          PUNCH 1400, (V(I),I=1,3),DIMX,DIMY,DIMZ                         003066
01752   454*          PUNCH 1400, RHOT,RHOP,TMIU,PMIU,TYLD,PYLD,SMIU                  003101
01763   455*          PUNCH 1100, NPS,NET,NPPJ,NEPJ,NPPJT,NEPJT,NPTR,NETR,NPLG,NELG,  003114
01763   456*         1  NPST,NELT,MAXDOF,NGD                                          003114
02003   457*          PUNCH 1100,(IBDVT(I),I=1,IU),((IBDST(I,J),J=1,3),I=1,3),        003136
02003   458*         1(IBCIST(I),I=1,3)                                               003136
02016   459*          DO 801 N=1,NET                                                  003223
02021   460*      801 PUNCH 1100,(NOD(N,I),I=1,NPE)                                   003223
```

```
02030   461*          NN=NELT+1                                                    003236
02031   462*          DO 802 N=NN,NELT                                             003241
02034   463*      802 PUNCH 1100,(NOD(N,I),I=1,NPEL6)                              003250
02043   464*          DO 803 N=1,NPST                                              003267
02046   465*      803 PUNCH 1420, N,KODE(N),(X(N,I),I=1,3)                         003267
02057   466*          PUNCH 1100, (NFS(I),I=1,NFST)                                003306
02065   467*          PUNCH 1100, (INTF(I),I=1,INTFT)                              003317
02073   468*          DO 805 J=1,IU                                                003333
02076   469*      805 PUNCH 2200,(VERPRV(I,J),I=1,NPST)                            003333
02105   470*          DO 808 J=1,3                                                 003351
02110   471*          LIE=1                                                        003351
02111   472*          IF(ISOT .GE. 0) LIE=I                                        003353
02113   473*          DO 808 K=LIE,3                                               003357
02116   474*      808 PUNCH 2200,(STS(I,J,K),I=1,NPS)                              003373
02126   475*          DO 810 J=1,3                                                 003413
02131   476*      810 PUNCH 2200,(DIS(I,J),I=1,NPST)                               003413
02140   477*          PUNCH 2200,(PRS(I),I=1,NPS)                                  003426
02146   478*          PUNCH 2200,(FJ(I),I=1,NPS)                                   003437
02154   479*          DO 811 J=1,3                                                 003453
02157   480*      811 PUNCH 2200,(BFS(I,J),I=1,NPST)                               003453
02166   481*          PUNCH 2200,(YIELD(I),I=1,NPST)                               003466
02174   482*          PUNCH 2200, ((CSTF(I,J),J=1,NCNT),I=1,NCNT)                  003477
02205   483*          PUNCH 2200, ((CSPF(I,J),J=1,NCNT),I=1,NCNT)                  003515
02216   484*          DO 818 J=1,NCNT                                              003537
02221   485*          DO 818 K=1,NCNT                                              003537
02224   486*      818 PUNCH 2200, (C(I,J,K),I=1,NPST)                              003537
02234   487*          GO TO 650                                                    003556
02235   488*      820 PRINT 1700                                                   003560
02237   489*      850 GO TO 1                                                      003564
02237   490*        C                                                              003564
02237   491*        C                                                              003564
02240   492*     1000 FORMAT (12A6)                                                003565
02241   493*     1001 FORMAT (1H1,12A6/)                                           003565
02242   494*     1002 FORMAT (I2)                                                  003565
02243   495*     1100 FORMAT (20I4)                                                003565
02244   496*     1105 FORMAT(4I8)                                                  003565
02245   497*     1200 FORMAT(8E10.4)                                               003565
02246   498*     1250 FORMAT(//6X,'NHBW =',I4//)                                   003565
02247   499*     1255 FORMAT(//6X,'NHBWS =',I4// )                                 003565
02250   500*     1300 FORMAT (2I5)                                                 003565
02251   501*     1400 FORMAT (8F10.4)                                              003565
02252   502*     1420 FORMAT ('GRID    ',2I8,3F8.3)                                003565
02252   503*        C                                                              003565
02253   504*     1500 FORMAT (1H1,2X,'TIME= ',E11.4/1HO,6X,'NODE',5X,'ENERGY',9X,  003565
02253   505*         1'DENSITY',7X,'PRESSURE'/)                                    003565
02254   506*     1520 FORMAT (I10,1P6E15.4)                                        003565
02255   507*     1530 FORMAT(2I10,1P6E15.4)                                        003565
02256   508*     1550 FORMAT (1HO,2X,'NODE',6X,'S11',10X,'S12',10X,'S13',10X,'S21',10X,  003565
02256   509*         1'S22',10X,'S23',10X,'S31',10X,'S32',10X,'S33'/)              003565
02257   510*     1580 FORMAT (I6,1P9E13.4)                                         003565
02260   511*     1582 FORMAT(1HO,2X,'COORDINATES'/1HO,6X,'NODE',6X,'KODE',10X,'X',12X,  003565
02260   512*         1'Y',12X,'Z'/)                                                003565
02261   513*     1585 FORMAT(1HO,2X,'MESH OF FREE SURFACE'/1HO,6X,'NODE',7X,'X',12X,  003565
02261   514*         1'Y',12X,'Z'/)                                                003565
02262   515*     1600 FORMAT (3314)                                                003565
02263   516*     1612 FORMAT(1HO,2X,'DISPLACEMENT AND VELOCITY'/                   003565
```

```
02261   517*          1 1HO,6X,'NODE',8X,'UX',11X,'UY',11X,'UZ',11X,'VX',11X,'VY',11X,    003565
02263   518*            2 'VZ'/)                                                           003565
02263   519*      C1                                                                       003565
02264   520*        1682 FORMAT (1HO,8X,'FAILURE NODE'/)                                   003565
02265   521*        1683 FORMAT(2I2O)                                                      003565
02266   522*        1684 FORMAT(1HO,2X,3OHPOISSON RATIO                /)                  003565
02267   523*        1685 FORMAT(1HO,2X,3OHFRACTURE CRITERION COEFFICIENT/)                 003565
02270   524*        1686 FORMAT(1HO,2X,3OHSTIFFNESS COEFFICIENTS       /)                  003565
02271   525*        1687 FORMAT(I10,1P12E10.4/10X,1P12E10.4/10X,1P12E10.4)                 003565
02271   526*      C                                                                        003565
02272   527*        1700 FORMAT ('OSTATIC SOLUTION ATTAINED')                              003565
02273   528*        1720 FORMAT ('OFAIL TO SATISFY PRESCRIBED CRITERION FOR ATTAINING      003565
02273   529*           1STATIC SOLUTION')                                                  003565
02273   530*      C                                                                        003565
02274   531*        2200 FORMAT (1P8E10.4)                                                 003565
02275   532*        2000 STOP                                                              003565
02276   533*             END                                                               003570

        END OF COMPILATION:       NO  DIAGNOSTICS.

@HOB,P      CELFEP
```

```
@POP,L WF  CELFEP,CELFEP
POP12R1 M72-16 10/31/78 20:56:37 (0,0)      RI
  0001      C
  0002      C
PE0003      PARAM1 PROC
  0004      C
  0005            PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33
  0006            PARAMETER NPELG= 8,NPE= 8
  0007            PARAMETER NDFRS=53,NDINT=10
  0008      C
  0009            PARAMETER NRMAX=2000,NCMAX=2
  0010            PARAMETER NGPLG=2,NGP=2
  0011            PARAMETER NCNT=6, NFRC=NPM
  0012            PARAMETER IU=5,NOPJT= 10,NTSTEP=200
  0013            PARAMETER NRV=NPM,NBS=NPM,NBD=NPM
  0014       END
PE0015      PARAM2 PROC
  0016            PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33
  0017            PARAMETER NPELG= 8,NPE= 8
  0018            PARAMETER NDFRS=53,NDINT=10
  0019      C
  0020      C THE PARAMETERS ARE SAME AS THOSE FIRST THREE CARDS IN PARAM1
  0021      C
  0022       END
PE0023      CELFE1 PROC
  0024            DIMENSION TITLE(12)
  0025              DIMENSION PRS(NPM),FJ(NPM)
  0026            DIMENSION CKF(NFRC,3),EF(NFRC,3,3),PNF(NFRC,3,3)
  0027            DIMENSION IORDER(IU)
  0028      C
  0029            COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAP
  0030            COMMON /DATAX/ RHOT,RHOP,V(3),TYLD,PYLD,ICOND
  0031            COMMON /CEL21/ IORUC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
  0032           1 ITER,DELT,NFST
  0033            COMMON /CEL23/ XPRV(NPMT,3), VERPRV(NPMT,IU)
  0034            COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)
  0035            COMMON /MATPL/ NFRAC,THIU,PMIU,ISOT,ISOP
  0036            COMMON /NCASE/ INDEX,NASTRN,NSLIDE
  0037            COMMON /UNITCV/ IUNIT,FACTL(2),FACTV(2),FACTR(2),FACTE(2),
  0038           1 FACTS(2),DIFFS(2)
  0039            COMMON /GEOM1/ NODE(NPMT),NOD(NEMT,NPELG)
  0040            COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
  0041           1C(NPMT,NCNT,NCNT)
  0042            COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)
  0043            COMMON /IMPCT4/ CSTF(NCNT,NCNT),CSPF(NCNT,NCNT),
  0044           1CINT(NDINT,NCNT,NCNT)
  0045            COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),
  0046           1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),
  0047           2VL(NPELG,IU)
  0048            COMMON /PARDOF/ MAXDOF,NGD
  0049            COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT
  0050            COMMON /NOLEPJ/ NEPJT,NPPJT,INTFT
  0051            COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)
  0052            COMMON /CEL22/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ
  0053            COMMON /TSTEP/ ITCIV,TIME,DT(NTSTEP)
  0054            COMMON /SURFCE/ SMIU,DIFF,CONST1,VMAX,VMIN,XDIM(3)
```

```
      0055            COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)
      0056            COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3)
      0057            COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)
      0058            COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)
      0059            COMMON /STSELM/ STL(NPE,3,3)
      0060            COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)
      0061            COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)
      0062            COMMON /SOLCNV/ VER(NPMT,IU)
      0063            COMMON /SOLSTR/ STS(NPM,3,3)
      0064            COMMON /SOLDIS/ DIS(NPMT,3)
      0065            COMMON /BCVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)
      0066            COMMON /BCSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)
      0067            COMMON /BCDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)
      0068            COMMON /DATE/ IDATE(3)
      0069            EQUIVALENCE (S(1),EF(1))
      0070          END
PL    0071          CONSRV PROC
      0072            COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB
      0073            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
     1  0074        ITER,DELT,NFST
      0075            COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)
      0076            COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)
      0077            COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP
      0078            COMMON /NCASE/ INDEX,NASTRN,NSLIDE
      0079            COMMON /UNITCV/ IUNIT,FACTL(2),FACTV(2),FACTR(2),FACTE(2),
     1  0080        1 FACTS(2),FACTOR(2)
      0081            COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)
      0082            COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
     1  0083        1C(NPMT,NCNT,NCNT)
      0084            COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)
      0085            COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),
     1  0086        1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),
     2  0087        2VL(NPELG,IU)
      0088            COMMON /PARDOF/ MAXDOF,NGD
      0089            COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)
      0090            COMMON /STSELM/ STL(NPE,3,3)
      0091            COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)
      0092            COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)
      0093            COMMON /SOLCNV/ VER(NPMT,IU)
      0094            COMMON /BCVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)
      0095          END
PE    0096          CONSTV PROC
      0097            COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB
      0098            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
     1  0099        1 ITER,DELT,NFST
      0100            COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)
      0101            COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)
      0102            COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP
      0103            COMMON /NCASE/ INDEX,NASTRN,NSLIDE
      0104            COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)
      0105            COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
     1  0106        1C(NPMT,NCNT,NCNT)
      0107            COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)
      0108            COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),
     1  0109        1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),
     2  0110        2VL(NPELG,IU)
```

```
      0111          COMMON /PARDOF/ MAXDOF,NGD
      0112          COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)
      0113          COMMON /STSELM/ STL(NPE,3,3)
      0114          COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)
      0115          COMMON /SOLSTR/ STS(NPM,3,3)
      0116          COMMON /BCSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)
      0117       END
PE    0118       DISPLV PROC
      0119          COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB
      0120          COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
      0121         1  ITER,DELT,NFST
      0122          COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)
      0123          COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)
      0124          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP
      0125          COMMON /NCASE/ INDEX,NASTRN,NSLIDE
      0126          COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)
      0127          COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
      0128         1C(NPMT,NCNT,NCNT)
      0129          COMMON /IMPCT3/ GM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)
      0130          COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),
      0131         1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),
      0132         2VL(NPELG,IU)
      0133          COMMON /PARDOF/ MAXDOF,NGD
      0134          COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)
      0135          COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)
      0136          COMMON /SCLDIS/ DIS(NPMT,3)
      0137          COMMON /BCDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)
      0138       END
PE    0139       GENMSH PROC
      0140          COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)
      0141          COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)
      0142          COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
      0143         1  ITER,DELT,NFST
      0144          COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT
      0145          COMMON /CEL22/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ
      0146          COMMON /NODEPJ/ NEPJT,NPPJT,INTFT
      0147          DIMENSION X(NPMT,3),G(16)
      0149       END
PE    0149       TESTNG PROC
      0150          PARAMETER NDIM=NEM
      0151          DIMENSION NDEG(32),INTELM(NDIM)
      0152       C
      0153          COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
      0154         1  ITER,DELT,NFST
      0155          COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)
      0156          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP
      0157          COMMON /NCASE/ INDEX,NASTRN,NSLIDE
      0158          COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)
      0159          COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT
      0160          COMMON /NODEPJ/ NEPJT,NPPJT,INTFT
      0161          COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)
      0162          COMMON /SURFCE/ SMIU,DIFFPR,VO,VMAXPR,VMINPR,XDIM(3)
      0163          COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)
      0164          COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)
      0165          COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
      0166         1  C(NPMT,NCNT,NCNT)
```

```
0167            COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)
0168            COMMON /SOLCNV/ VER(NPMT,IU)
0169            COMMON /SOLSTR/ STS(NPM,3,3)
0170            COMMON /SOLDIS/ DIS(NPMT,3)
0171            COMMON /BDVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)
0172            COMMON /BDSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)
0173            COMMON /BDDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)
0174            COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)
0175        END
PEJ176      CELSTR PROC
0177            PARAMETER NWK=2000,NTRL=7
0178            DIMENSION WORK(NWK),P3(2),ITRL(NTRL)
0179            COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAP
0180            COMMON /DATAX/ RHOT,RHOP,V(3),TYLD,PYLD,ICOND
0181            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,
0182           1  ITER,DELT,NFST
0183            COMMON /CEL23/ XPRV(NPMT,3), VERPRV(NPMT,IU)
0184            COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LT(3,3),E(3,3),PNIU(3,3),CK(3)
0185            COMMON /MATRL/ NFRAC,THIU,PMIU,ISOT,ISOP
0186            COMMON /NCASE/ INDEX,NASTRN,NSLIDE
0187            COMMON /UNITCV/ IUNIT,FACTL(2),FACTV(2),FACTR(2),FACTE(2),
0188           1 FACTS(2),CIFFS(2)
0189            COMMON /CEGM1/ KODE(NPMT),NOD(NEMT,NPELG)
0190            COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),
0191           1C(NPMT,NCNT,NCNT)
0192            COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)
0193            COMMON /IMPCT4/ CSTF(NCNT,NCNT),CSPF(NCNT,NCNT),
0194           1CINT(NDINT,NCNT,NCNT)
0195            COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),
0196           1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),
0197           2VL(NPELG,IU)
0198            COMMON /PARDOF/ MAXDOF,NGD
0199            COMMON /CEL22/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ
0200            COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT
0201            COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3)
0202            COMMON /NODEPJ/ NEPJT,NPPJT,INTFT
0203            COMMON /SURFCE/ SMIU,DIFF,CONST1,VMAX,VMIN,XDIM(3)
0204            COMMON /TSTEP/ ITGIV,TIME,DT(NTSTEP)
0205            COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)
0206            COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)
0207            COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)
0208            COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)
0209            COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)
0210            COMMON /SOLCNV/ VER(NPMT,IU)
0211            COMMON /SOLSTR/ STS(NPM,3,3)
0212            COMMON /SOLDIS/ DIS(NPMT,3)
0213            COMMON /BDVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)
0214            COMMON /BDSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)
0215            COMMON /BDDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)
0216            COMMON /DATE/ IDATE(3)
0217            DATA P3/'XXXX','XXXX'/
0218            DATA KFILE,LFILE,MFILE,NFILE/15,17,14,12/
0219        C
0220        C KFILE=OUTPUT   MOMENTUM, ENERGY, DENSITY AND STSRESS MATRICES
0221        C                AND ASSOCIATED FORCING VECTORS IN CELFE ZONE
0222        C LFILE=OUTPUT   DISPLACEMENT MATRIX OF CELFE ZONE
```

```
0223       C MFILE=INPUT     SOLUTIONS OF ENTIRE STRUCTURE IN DISPLACEMENTS AND
0224       C MFILE=INPUT     SOLUTIONS OF DENSITY, MOMENTUMS, ENERGY, STRESSES
0225       C                 IN CELFE ZONE
0226       C NFILE=INPUT     SOLUTIONS OF ENTIRE STRUCTURE IN DISPLACEMENTS AND
0227       C                 VELOCITIES
0228       C FILE 16 USED TO STORE CELFE VARIABLES AND RELATED QUANTITIES BETWEEN
0229       C      EXECUTIONS OF NASTRAN
0230            DATA NWD/2000/
0231        END
PE0232     STATEC PROC
0233       C
0234            COMMON /UNITCV/ IUNIT,FACTL(2),FACTV(2),FACTR(2),FACTE(2),
0235           1 FACTS(2),DIFFS(2)
0236            DIMENSION VER(NPH,5),P(1),INTF(1)
0237       C
0238       C
0239       C     TARGET  ---  BORON/EPOXY
0240       C
0241            DATA AT1,AT2,BT0,BT1,BT2,CT0,CT1,PHIT/1.8212,4.3509,0.3764,0.3287,
0242           1  1.0801,0.5531,0.6346,0.250/
0243       C
0244       C     PROJECTILE ---  SILASTIC
0245       C
0246            DATA AP1,AP2,BP0,BP1,BP2,CP0,CP1,PHIP/0.004794,0.04684,0.33969,
0247           1  0.02377,0.50767,0.4925,0.5721,0.3000/
0248        END

END EDP  ERRORS : NONE

@HDG,P     CELTRN
```

```
@FOR,S    CELTRN,CELTRN
HSA  E3 -01/09/78-23:43:25 (7,)

    SUBROUTINE CELTRN    ENTRY POINT 001003

    STORAGE USED: CODE(1) 001016; DATA(0) 004245; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003    SYSMTX  007641
     0004    DATAX   000011
     0005    CELZ1   000015
     0006    CELZ3   001140
     0007    ORTHPC  000060
     0010    MATRL   000005
     0011    NCASE   000003
     0012    GEOM1   000524
     0013    IMPCT1  006650
     0014    IMPCT3  000500
     0015    IMPCT4  000660
     0016    ELMLAG  000410
     0017    PARDOF  000002
     0020    CELZ2   000007
     0021    LAGRGN  000005
     0022    PROJMT  000047
     0023    NODEPJ  000003
     0024    SURFCE  000010
     0025    TSTEP   000312
     0026    SRFACE  000077
     0027    MATRL2  000230
     0030    STRESS  002530
     0031    IMPCT2  000230
     0032    DISPLC  001620
     0033    SOLCNV  000574
     0034    SOLSTR  001254
     0035    SOLDIS  000344
     0036    BDVER   001376
     0037    BDSTR   002541
     0040    BDDIS   000713
     0041    DATE    000003

    EXTERNAL REFERENCES (BLOCK, NAME)

     0042    NTRAN
     0043    OPNNAS
     0044    OPENTP
     0045    OPENF
     0046    INTVFC
     0047    FRTRAN
     0050    NFRR2$
     0051    NWDU$
     0052    NIO2$
     0053    NGIU$
```

```
0054   NIO3$
0055   NIO1$
0056   NWDC$
0057   NERR3$
```

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

```
0001   000013 100L      0000   004051 101F      0001   000045 150L      0000   004062 1800F     0000   004065 1810F
0000   004074 1820F     0000   004101 1825F     0000   004106 1900F     0000   004122 1910F     0000   004134 1911F
0000   004145 1912F     0001   000060 200L      0001   000070 207G      0001   000116 210L      0001   000125 211L
0001   000147 215L      0001   000253 225L      0001   000267 230L      0001   000334 252L      0001   000344 255L
0001   000217 256G      0001   000226 261G      0001   000243 267G      0000   004164 3000F     0001   000300 314G
0001   000307 317G      0001   000324 325G      0001   000414 360G      0001   000434 367G      0001   000443 377G
0001   000354 400L      0001   000467 405G      0001   000406 405L      0001   000411 410L      0001   000507 414G
0001   000523 427G      0001   000543 436G      0001   000520 450L      0001   000556 451G      0001   000576 460G
0001   000550 460L      0001   000604 480L      0001   000717 500L      0001   000626 501G      0001   000632 504G
0001   000647 511G      0001   000672 526G      0001   000701 531G      0001   000722 900L      0001   000732 906L
0001   000742 907L      0001   000752 908L      0001   000760 999L      0005   000001 A         0005 R 000002 ALPHA
0032   001254 BFS       0000 R 003736 BG        0000 R 003766 BI        0013   001370 C         0016   000000 CD
0014   000040 CFRC      0015   000110 CINT      0007   000055 CK        0016   000200 CL        0024   000002 CONST1
0016   000100 CS        0015   000044 CSPF      0015   000000 CSTF      0005 R 000013 DELT      0016   000250 DF
0024   000001 DIFF      C020   000004 DIMX      0020   000005 DIMY      0020   000006 DIMZ      0035 R 000000 DIS
0040   000347 DISBD     0032   000000 DISN      0032   000344 DISPRV     0025   000002 DT        0007   000033 E
0016   000310 ELXYZ     0014   000030 FJCB      0013   001140 FJN       0013   001254 FJPRV     0000 I 004030 I
0040 I 000003 IBDIS     0040 I 000000 IBDIST     0037   000011 IBDS      0037   000000 IBDST     0036   000005 IBDV
0036   000000 IBDVT     0004   000010 ICOND      0000 I 004034 ID        0041 I 000000 IDATE     0000 I 004040 IE
0000 I 004042 IFRR      0000 I 004036 II        0000 I 004050 IJ        0011 I 000000 INDEX     0000   004223 INJP$
0005 I 000011 INNER     0026   000065 INTF      0023   000002 INTFT     0005   000000 IORDC     0021   000002 IORDL
0010   000004 ISOP      0010 I 000003 ISOT      0005   000012 ITER      0025   000000 ITGIV     0000 I 003722 ITRL
0004   000007 IUNIT     0000 I 004035 J         0000 I 004037 K         0000 I 003731 KFILE     0012   000000 KODE
0000 I 004027 L         0000 I 004045 LCOL      0000 I 003732 LFILE     0007   000022 LI        0000 I 004044 LIE
0000 I 004047 LL        0000 I 004046 LSET      0017   000000 MAXDOF    0000 I 003733 MFILE     0011 I 000001 NASTRN
0005   000006 NBW       0021   000003 NBWT      0020   000003 NELG      0021   000001 NELT      0005   000010 NEPJ
0023   000000 NEPJT     0005   000003 NET       0020   000001 NETR      0000 I 003734 NFILE     0010   000000 NFRAC
0026   000000 NFS       0005   000014 NFST      0017   000001 NGO       0005   000005 NHBW      0021   000004 NHBWT
0017   000114 NOD       0027   000114 NODFRC    0020   000002 NPLG      0005   000007 NPPJ      0023   000001 NPPJT
0005   000004 NPS       0021 I 000000 NPST      0020   000000 NPTR      0000 I 004043 NR        0011   000002 NSLIDE
0003   007640 NSTAB     0000 I 004026 NSTORE    0000 I 003735 NWD       0014   000000 OM        0022   000044 PCK
0022   000022 PE        0016   000300 PM        0010   000002 PMIU      0007   000044 PNIU      0022   000033 PPNIU
0031   000000 PRSN      0031   000114 PRSPRV    0022   000011 PSLC      0022   000000 PSLT      0004   000006 PYLD
0000 R 003720 P3        0004   000001 RHOP      0004   000000 RHOT      0003 R 000000 S         0037   001265 SRD
0016   000210 SF        0007   000011 SLC       0007   000000 SLT       0024   000000 SMIU      0034 R 000000 STS
0030   000000 STSN      0030   001254 STSPRV    0025   000001 TIME      0010   000001 TMIU      0004   000005 TYLD
0004   000002 V         0036   000601 VBD       0000 R 004041 VEL       0033 R 000000 VER       0013   000344 VERN
0006 R 000344 VERPRV    0016   000340 VL        0024   000003 VMAX      0024   000004 VMIN      0016   000220 W
0000 R 000000 WORK      0032   000710 X         0024   000005 XDIM      0013 R 000000 XN        0006 R 000000 XPRV
0000 R 004031 X2        0027   000000 YIELD     0000 R 004032 Y2        0000 R 004033 Z2
```

```
00101      1*          SUBROUTINE CELTRN(KKK)                                          000000
00101      2*      C                                                                   000000
00101      3*      C*********************                                              000000
00101      4*      C                                                                   000000
```

```
00101      5*     C  INTERFACING CELFE WITH NASTRAN                                                    000000
00101      6*     C                                                                                    000000
00101      7*     C*******************                                                                 000000
00101      8*     C                                                                                    000000
00103      9*           INCLUDE PARAM1,LIST                                                            000000
00103      9*     PARAM1 PROC                                                                          000000
00103      9*     C                                                                                    000000
00104      9*           PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                                    000000
00105      9*           PARAMETER NPELG= 8,NPE= 8                                                      000000
00106      9*           PARAMETER NDFRS=53,NDINT=10                                                    000000
00106      9*     C                                                                                    000000
00107      9*           PARAMETER NRMAX=2000,NCMAX=2                                                   000000
00110      9*           PARAMETER NGPLG=2,NGP=2                                                        000000
00111      9*           PARAMETER NCNT=6, NFRC=NPM                                                     000000
00112      9*           PARAMETER IU=5,NDPJT= 10,NTSTEP=200                                            000000
00113      9*           PARAMETER NBV=NPM,NBS=NPM,N3D=NPM                                              000000
00113      9*      END                                                                                 000000
00114     10*           INCLUDE CELSTR,LIST                                                            000000
00114     10*     CELSTR PROC                                                                          000000
00115     10*           PARAMETER NWK=2000,NTRL=7                                                      000000
00116     10*           DIMENSION WORK(NWK),P3(2),ITRL(NTRL)                                           000000
00117     10*           COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB                                           000000
00120     10*           COMMON /DATAX/ RHOT,RHOP,V(3),TYLD,PYLD,IUNIT,ICOND                            000000
00121     10*           COMMON /CEL71/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,                 000000
00121     10*          1  ITER,DEL1,NFST                                                               000000
00122     10*           COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)                                   000000
00123     10*           COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)               000000
00124     10*           COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                                       000000
00125     10*           COMMON /NCASE/ INDEX,NASTRN,NSLIDE                                             000000
00126     10*           COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)                                      000000
00127     10*           COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),                  000000
00127     10*          1C(NPMT,NCNT,NCNT)                                                              000000
00130     10*           COMMON /IMPCT3/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)                      000000
00131     10*           COMMON /IMPCT4/ CSTF(NCNT,NCNT),CSPF(NCNT,NCNT),                               000000
00131     10*          1CINT(NDINT,NCNT,NCNT)                                                          000000
00132     10*           COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),                     000000
00132     10*          1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),                     000000
00132     10*          2VL(NPELG,IU)                                                                   000000
00133     10*           COMMON /PARDOF/ MAXDOF,NGD                                                     000000
00134     10*           COMMON /CELZ2/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ                              000000
00135     10*           COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                                     000000
00136     10*           COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3)                  000000
00137     10*           COMMON /NODEPJ/ NEPJT,NPPJT,INTFT                                              000000
00140     10*           COMMON /SURFCE/ SMIU,DIFF,CONST1,VMAX,VMIN,XDIM(3)                             000000
00141     10*           COMMON /TSTEP/ ITGIV,TIME,DT(NTSTEP)                                           000000
00142     10*           COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)                                          000000
00143     10*           COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)                                       000000
00144     10*           COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)                                  000000
00145     10*           COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)                                          000000
00146     10*           COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)              000000
00147     10*           COMMON /SOLCNV/ VFR(NPMT,IU)                                                   000000
00150     10*           COMMON /CLSTR/ STS(NPM,3,3)                                                    000000
00151     10*           COMMON /SOLDIS/ DIS(NPMT,3)                                                    000000
00152     10*           COMMON /BDVEL/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)                              000000
00153     10*           COMMON /BDSTL/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)                           000000
00154     10*           COMMON /BDDIS/ IBDIS(3),IBDIS(NBD,3),DISBD(NBD,3)                              000000
```

```
00155     10*          COMMON /DATE/ IDATE(3)                                    000000
00156     10*          DATA P3/'XXXX','XXXX'/                                    000000
00160     10*          DATA KFILE,LFILE,MFILE,NFILE/15,17,14,12/                 000000
00160     10*     C                                                              000000
00160     10*     C KFILE=OUTPUT   MOMENTUM, ENERGY, DENSITY AND STSRESS MATRICES 000000
00160     10*     C                AND ASSOCIATED FORCING VECTORS IN CELFE ZONE   000000
00160     10*     C LFILE=OUTPUT   DISPLACEMENT MATRIX OF CELFE ZONE              000000
00160     10*     C MFILE=INPUT    SOLUTIONS OF ENTIRE STRUCTURE IN DISPLACEMENTS AND 000000
00160     10*     C MFILE=INPUT    SOLUTIONS OF DENSITY, MOMENTUMS, ENERGY, STRESSES 000000
00160     10*     C                IN CELFE ZONE                                  000000
00160     10*     C NFILE=INPUT    SOLUTIONS OF ENTIRE STRUCTURE IN DISPLACEMENTS AND 000000
00160     10*     C                VELOCITIES                                     000000
00160     10*     C FILE 16 USED TO STORE CELFE VARIABLES AND RELATED QUANTITIES BETWEEN 000000
00160     10*     C     EXECUTIONS OF NASTRAN                                     000000
00165     10*          DATA NWD/2000/                                            000000
00165     10*      END                                                           000000
00167     11*          DIMENSION BG(24),BI(32)                                   000000
00170     12*          NSTORE=NRMAX*NCMAX+1 +9 +13 +NPMT*(3+IU) +48 +5 +3 +NPMT+NEMT*NPEL 000000
00170     13*         1G +NPMT*(3+IU+NCNT*NCNT)+2*NPM                            000000
00170     14*         2+NCNT*(NCNT+NCNT+NCNT*NDINT) +2 +7 +5 +39 +3 +8 +2+NTSTEP 000000
00170     15*         3+NDFRS+NDINT +NPMT+NFRC +NPM*(9+9) +NPM*2 +NPMT*12 +NPMT*IU 000000
00170     16*         4+NPM*9 +NPMT*3 +IU*(1+2*NBV) +9+NBS*18 +3+NBD*6           000000
00171     17*          GO TO (100,150,200,400),KKK                               000001
00171     18*     C                                                              000001
00171     19*     C  STORE DATA IN DISK FILE 16 AND OPEN FILE IN NASTRAN INPUTT2 000001
00171     20*     C    FORMAT FOR WRITE ELEMENT MATRICES OF CONSERVATION AND     000001
00171     21*     C    CONSTITUTIVE EQUATIONS                                    000001
00171     22*     C                                                              000001
00172     23*       100 CONTINUE                                                 000013
00173     24*          WRITE(6,101)NSTORE                                        000013
00176     25*       101 FORMAT(1X,'LENGTH OF LABLED COMMON STORED IN FILE 16=',I8) 000020
00177     26*          CALL NTRAN(16,10,1,NSTORE,IBDIST(1),L,22)                 000020
00200     27*          CALL OPNNAS(KFILE,WORK,NWD,IDATE,P3,1)                    000031
00201     28*          RETURN                                                    000041
00201     29*     C                                                              000041
00201     30*     C  OPEN FILE IN NASTRAN INPUTT2 FORMAT FOR WRITE ELEMENT MATRICES 000041
00201     31*     C    OF DISPLACEMENT VECTOR                                    000041
00201     32*     C                                                              000041
00202     33*       150 CONTINUE                                                 000045
00203     34*          CALL OPNNAS(LFILE,WORK,NWD,IDATE,P3,1)                    000045
00204     35*          RETURN                                                    000054
00204     36*     C                                                              000054
00204     37*     C  UPDATE AND WRITE NASTRAN BULK DATA IN FILE                  000054
00204     38*     C                                                              000054
00205     39*       200 CONTINUE                                                 000060
00206     40*          DO 220 I=1,NPST                                           000060
00211     41*          READ (18,1800)BG,X2,Y2,Z2,BI                              000070
00220     42*          GO TO (210,211),INNER                                     000106
00221     43*       210 X2=XPRV(I,1)                                             000116
00222     44*          Y2=XPRV(I,2)                                              000117
00223     45*          Z2=XPRV(I,3)                                              000121
00224     46*          GO TO 215                                                 000123
00224     47*     C                                                              000123
00224     48*     C UPDATE GRID COORDINATES FROM LAST TIME STEP                  000123
00224     49*     C                                                              000123
00225     50*       211 X2=ALPHA*XPRV(I,1)+(1.0-ALPHA)*XN(I,1)                   000125
```

```
00226     51*          Y2=ALPHA*XPRV(I,2)+(1.0-ALPHA)*XN(I,2)                                  000132
00227     52*          Z2=ALPHA*XPRV(I,3)+(1.0-ALPHA)*XN(I,3)                                  000140
00230     53*      215 CONTINUE                                                                000147
00230     54*        C                                                                         000147
00230     55*        C OUTPUT TO NASTRAN NEW GRID LOCATION                                     000147
00230     56*        C                                                                         000147
00231     57*          PUNCH 1800,BG,X2,Y2,Z2,BI                                               000147
00240     58*          WRITE(19,1800)BG,X2,Y2,Z2,BI                                            000164
00247     59*      220 CONTINUE                                                                000205
00251     60*          PUNCH 1810, DELT                                                        000205
00254     61*          ID=1                                                                    000212
00255     62*          DO 230 I=1,NPST                                                         000217
00260     63*          DO 230 J=1,3                                                            000226
00263     64*          IF (IBDIST(J) .EQ. 0) GO TO 225                                         000232
00265     65*          II=IBDIST(J)                                                            000234
00266     66*          DO 223 K=1,II                                                           000236
00271     67*          IE=IBDIS(K,J)                                                           000243
00272     68*          IF (IE .EQ. I) GO TO 230                                                000245
00274     69*      223 CONTINUE                                                                000253
00276     70*      225 CONTINUE                                                                000253
00276     71*        C                                                                         000253
00276     72*        C UPDATE VELOCITY FOR NASTRAN INITIAL CONDITION                           000253
00276     73*        C                                                                         000253
00277     74*          VEL=VERPRV(I,J)/VERPRV(I,5)                                             000253
00300     75*          PUNCH 1820, ID,I,J,DIS(I,J),VEL                                         000255
00307     76*      230 CONTINUE                                                                000273
00312     77*          ID=2                                                                    000273
00313     78*          DO 255 I=1,NPST                                                         000300
00316     79*          DO 255 J=1,3                                                            000307
00321     80*          IF (IBDIST(J) .EQ. 0) GO TO 252                                         000313
00323     81*          II=IBDIST(J)                                                            000315
00324     82*          DO 250 K=1,II                                                           000317
00327     83*          IE=IBDIS(K,J)                                                           000324
00330     84*          IF (IE .EQ. I) GO TO 255                                                000326
00332     85*      250 CONTINUE                                                                000334
00334     86*      252 CONTINUE                                                                000334
00335     87*          PUNCH 1825, ID,I,J,DIS(I,J)                                             000334
00343     88*      255 CONTINUE                                                                000350
00346     89*          RETURN                                                                  000350
00346     90*        C                                                                         000350
00346     91*        C  UPDATE FILE 16 AND READ SOLUTIONS FROM NASTRAN IN OUTPUT2 FORMAT       000350
00346     92*        C                                                                         000350
00347     93*      400 CONTINUE                                                                000354
00350     94*          CALL NTRAN(16,10,2,NSTORE  ,IBDIST(1),L,22)                 ********    000354
00351     95*          CALL OPENTP(MFILE,WORK,NWK,IERR,1)                                      000364
00352     96*          IF (IERR .NE. 0) GO TO 900                                              000373
00354     97*          GO TO (410,450,405),INDEX                                               000375
00355     98*      405 RETURN                                                                  000406
00356     99*      410 CONTINUE                                                                000414
00357    100*          DO 420 I=1,IU                                                           000414
00362    101*          CALL OPENF(IERR,ITRL)                                                   000414
00363    102*          IF (IERR .NE. 0) GO TO 900                                              000420
00365    103*          CALL INTVEC(*900,WORK,NWK,ITRL(3),1)                                    000422
00366    104*          DO 419 NR=1,NPMT                                                        000434
00371    105*          VER(NR,I)=WORK(NR)                                                      000434
00372    106*      419 CONTINUE                                                                000443
```

```
00374    107*      420 CONTINUE                                                  000443
00376    108*          DO 425 I=1,3                                              000443
00401    109*          LIE=1                                                     000443
00402    110*          IF(ISOT .GE. 0) LIE=I                                     000445
00404    111*          DO 425 J=LIE,3                                            000452
00407    112*          CALL OPENF(IERR,ITRL)                                     000467
00410    113*          IF(IERR .NE. 0) GO TO 900                                 000473
00412    114*          CALL INTVEC($900,WORK,NWK,ITRL(3),1)                      000475
00413    115*          DO 424 NR=1,NPM                                           000507
00416    116*          STS(NR,I,J)=WORK(NR)                                      000507
00417    117*      424 CONTINUE                                                  000516
00421    118*      425 CONTINUE                                                  000516
00424    119*          GO TO 460                                                 000516
00425    120*      450 CONTINUE                                                  000523
00426    121*          DO 455 I=1,3                                              000523
00431    122*          CALL OPENF(IERR,ITRL)                                     000523
00432    123*          IF (IERR .NE. 0) GO TO 900                                000527
00434    124*          CALL INTVEC($900,WORK,NWK,ITRL(3),1)                      000531
00435    125*          DO 454 NR=1,NPMT                                          000543
00440    126*          VER(NR,I)=WORK(NR)                                        000543
00441    127*      454 CONTINUE                                                  000550
00443    128*      455 CONTINUE                                                  000550
00445    129*      460 CONTINUE                                                  000550
00446    130*          IF (NASTRN .EQ. 1) GO TO 480                              000550
00450    131*          DO 465 I=1,3                                              000556
00453    132*          CALL OPENF(IERR,ITRL)                                     000556
00454    133*          IF (IERR .NE. 0) GO TO 900                                000562
00456    134*          CALL INTVEC($900,WORK,NWK,ITRL(3),1)                      000564
00457    135*          DO 464 NR=1,NPMT                                          000576
00462    136*          DIS(NR,I)=WORK(NR)                                        000576
00463    137*      464 CONTINUE                                                  000602
00465    138*      465 CONTINUE                                                  000602
00467    139*          GO TO 500                                                 000602
00470    140*      480 CONTINUE                                                  000604
00470    141*      C                                                             000604
00470    142*      C READ NASTRAN DISPLACEMENT AND VELOCITY VECTORS INTO S-ARRAY FOR 000604
00470    143*      C   LAST NASTRAN TIME STEP                                    000604
00470    144*      C                                                             000604
00471    145*          CALL OPENTP(NFILE,WORK,NWK,IERR,1)                        000604
00472    146*          IF (IERR .NE. 0) GO TO 906                                000612
00474    147*          CALL OPENF(IERR,ITRL)                                     000614
00475    148*          IF(IERR.NE.0) GO TO 907                                   000620
00477    149*          LCOL=0                                                    000622
00500    150*          DO 490 LSET=1,4                                           000632
00503    151*          DO 490 LL=1,2                                             000632
00506    152*          LCOL=LCOL+1                                               000632
00507    153*          CALL INTVEC($908,WORK,NWK,ITRL(3),1)                      000635
00510    154*          DO 485 NR=1,NRMAX                                         000647
00513    155*          S(NR,LL)=WORK(NR)                                         000647
00514    156*      485 CONTINUE                                                  000651
00516    157*          WRITE (6,3000) LCOL,NWD                                   000651
00522    158*      490 CONTINUE                                                  000664
00525    159*          DO 498 I=1,NPST                                           000664
00530    160*          DO 498 J=1,3                                              000701
00533    161*          IJ=6*(I-1)+J                                              000701
00534    162*          DIS(I,J)=S(IJ,1)                                          000705
```

```
      CELTRN                                                                    DATE 010978

00535     163*          498 CONTINUE                                                    000717
00540     164*          500 CONTINUE                                                    000717
00541     165*              RETURN                                                      000717
00542     166*          900 WRITE(6,1900) MFILE,IERR                                    000722
00546     167*              GO TO 999                                                   000730
00547     168*          906 WRITE(6,1910) NFILE,IERR                                    000732
00553     169*              GO TO 999                                                   000740
00554     170*          907 WRITE(6,1911) NFILE,IERR                                    000742
00560     171*              GO TO 999                                                   000750
00561     172*          908 WRITE(6,1912) NFILE                                         000752
00564     173*          999 CONTINUE                                                    000760
00565     174*              CALL ERTRAN(1)                                              000760
00566     175*         1800 FORMAT(24A1,3F8.3,32A1)                                     000762
00567     176*         1810 FORMAT ('TSTEP   1', 7X,'2',7X,F8.3,'1')                    000762
00570     177*         1820 FORMAT('TIC      ',3I8,1P2E8.2)                             000762
00571     178*         1825 FORMAT ('DAREA   ',3I8, 1PE8.2)                             000762
00572     179*         1900 FORMAT (1X,' B MATRIX FILE ERROR',5X,'FILE NO. ',I3,5X,     000762
00572     180*             1  'IERR = ',I4)                                             000762
00573     181*         1910 FORMAT (1X,'ERROR IN OPENTP, FILE NO. ',  I3,5X,'IERR = ',I4)  000762
00574     182*         1911 FORMAT (1X,'ERROR IN OPENF, FILE NO. ',  I3,5X,'IERR= ',I4)   000762
00575     183*         1912 FORMAT (1X,'ERROR IN INTVEC WHEN READING DISP. AND VELOC. '/  000762
00575     184*             1  5X,'VECTORS FROM FILE NO.',I4)                            000762
00576     185*         3000 FORMAT(1X,'COLUMN ',I3,' HAD ',I5,' WORDS READ,   ',I5,     000762
00576     186*             1' WORDS WERE SKIPPED IN THIS COLUMN.')                      000762
00577     187*              RETURN                                                      000762
00600     188*              END                                                         001015

       END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      CONTROL

@PRT,S    CONTROL
FURPUR 27R2A RL71-3 01/09/78 23:43:30
```

CONTROL DATE 010978

```
CH-LEEBIN202*LEE(0).CONTROL
      1     LIB SYS$*MSFC$.
      2     SEG CELFE
      3     IN CELFE,MATRX
      4     SEG C3VL*,(CELFE)
      5     IN C3VL
      6     SEG CELA*,C3VL
      7     IN CELA
      8     SEG CELD*,CELA
      9     IN CELD
     10     SEG C3JB*,CELD
     11     IN C3JB
     12     SEG FRCR*,C3JB
     13     IN FRCR
     14     SEG STFG*,FRCR
     15     IN STFG
     16     SEG STRN*,STFG
     17     IN STRN
     18     SEG SLID*,STRN
     19     IN SLID
     20     SEG OPENT*,SLID
     21     IN OPENT,INTVEC
     22     END

@HDG,P     C3J8
```

```
C3JB                                                                    DATE 010978

@FOR,S    C3JB,C3JB
HSA  E3 -01/09/78-23:43:30 (1,)

   SUBROUTINE CEL3JB    ENTRY POINT 000111

   STORAGE USED: CODE(1) 000132; DATA(0) 000033; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CELZ1  000015

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NERR2$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000062 100L      0001   000031 1076      0001   000036 50L       0001   000051 60L       0001   000054 80L
    0001   000056 85L       0003   000001 A         0003   000002 ALPHA     0003   000013 DELT      0000   000005 INJP$
    0003 I 000011 INNER     0003   000000 IORDC     0003   000012 ITER      0000 I 000000 N         0003   000006 NBW
    0003   000010 NEPJ      0003   000003 NET       0003   000014 NFST      0003   000005 NHBW      0003   000007 NPPJ
    0003 I 000004 NPS       0000 R 000001 X         0000 R 000002 Y         0000 R 000003 Z

00101      1*            SUBROUTINE CEL3JB(NPM,VERPRV,VERN,VER,FJPRV,FJ,KODE)                    000031
00103      2*            DIMENSION VERPRV(NPM,5),VERN(NPM,5),VER(NPM,5),                         000031
00103      3*           2FJPRV(1),FJ(1),KODE(1)                                                  000031
00104      4*            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,          000031
00104      5*           1  ITER,DELT,NFST                                                        000031
00104      6*     C                                                                             000031
00105      7*            FJCB(X,Y,Z)=X*Y/Z                                                       000031
00105      8*     C                                                                             000031
00106      9*            DO 100 N=1,NPS                                                          000031
00111     10*            IF(KODE(N) .NE. 0) GO TO 50                                             000031
00113     11*            FJ(N) =1.0                                                              000032
00114     12*            GO TO 100                                                               000034
00115     13*         50 X=FJPRV(N)                                                              000036
00116     14*            Y=VERPRV(N,5)                                                           000037
00117     15*            GO TO (60,80),INNER                                                     000041
00120     16*         60 Z=VERN(N,5)                                                             000051
00121     17*            GO  TO 85                                                               000052
00122     18*         80 Z= VER(N,5)                                                             000054
00123     19*         85 FJ(N) =FJCB(X,Y,Z)                                                      000056
00124     20*        100 CONTINUE                                                                000070
00126     21*            RETURN                                                                  000070
00127     22*            END                                                                     000131

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

C3J8 DATE 010978

@HDG,P C3VL

```
@FOR,S    C3VL,C3VL
HSA  E3 -01/09/78-23:43:31 (1,)

SUBROUTINE CEL3CV    ENTRY POINT 001461

STORAGE USED: CODE(1) 001526; DATA(0) 000101; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   SYSMTX 007641
 0004   CELZ1  000015
 0005   CELZ3  001140
 0006   ORTHPC 000060
 0007   MATRL  000005
 0010   NCASE  000003
 0011   GEOM1  000524
 0012   IMPCT1 006650
 0013   IMPCT3 000500
 0014   ELMLAG 000410
 0015   PARDOF 000002
 0016   STRESS 002530
 0017   STSELM 000110
 0020   DISPLC 001620
 0021   DISELM 000060
 0022   SOLCNV 000574
 0023   BDVER  001375
 0024   LAGRGN 000005

EXTERNAL REFERENCES (BLOCK, NAME)

 0025   ELCELV
 0026   ELMTRX
 0027   ELAGVA
 0030   BNDFQ
 0031   NERR2$
 0032   NERR3$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001   000275 110L    0001   000324 115L    0001   000333 118L    0001   000434 120L    0001   000004 142G
 0001   000005 145G    0001   000517 150L    0001   000523 155L    0001   000544 158L    0001   000607 165L
 0001   000616 170L    0001   000050 171G    0001   000642 171L    0001   000106 175G    0001   000672 180L
 0001   000730 188L    0001   001052 190L    0001   001135 193L    0001   001141 195L    0001   001162 197L
 0001   001230 200L    0001   000117 202G    0001   000141 214G    0001   000144 220G    0001   000205 231G
 0001   000225 236G    0001   000246 244G    0001   001234 250L    0001   000303 254G    0001   000320 263G
 0001   000331 271G    0001   001241 280L    0001   001254 293L    0001   001262 295L    0001   001330 300L
 0001   000421 301G    0001   001335 302L    0001   000445 314G    0001   000447 317G    0001   001403 320L
 0001   000471 331G    0001   000511 337G    0001   000554 354G    0001   000564 361G    0001   000632 376G
 0001   000647 402G    0001   000655 406G    0001   000013 41L     0001   000702 420G    0001   000710 424G
 0001   000735 436G    0001   000743 442G    0001   000744 445G    0001   001037 455G    0001   000021 46L
 0001   000040 47L     0001   001063 470G    0001   001065 473G    0001   000060 48L     0001   001107 505G
 0001   001127 513G    0001   001175 530G    0001   001205 535G    0001   001317 547G    0001   001322 574G
```

0001		001362	611G	0001		001374	615G	0001		001427	626G	0001		000131	68L	0001		000173	85L
0004		000001	A	0004	R	000002	ALPHA	0021	R	000030	BF	0020	R	001254	BFS	0012	R	001370	C
0014	R	000000	CD	0013	R	000040	CFRC	0006		000055	CK	0014	R	000200	CL	0014	R	000100	CS
0004	R	000013	DELT	0014	R	000250	DF	0020	R	000000	DISN	0020	R	000344	DISPRV	0021	R	000000	DL
0006		000033	E	0014	R	000310	ELXYZ	0013	R	000030	FJCB	0012	R	001140	FJN	0012	R	001254	FJPRV
0000	I	000000	I	0023	I	000005	IBDV	0023	I	000000	IBDVT	0000	I	000014	IE	0000	I	000013	II
0010	I	000000	INDEX	0000		000044	INJPS	0004	I	000011	INNER	0004	I	000000	IORDC	0024	I	000002	IORDL
0007		000004	ISOP	0007		000003	ISOT	0004	I	000012	ITER	0000	I	000001	J	0000	I	000007	K
0011	I	000000	KODE	0006	I	000022	LI	0000	I	000002	LIN	0015		000000	MAXDOF	0000	I	000003	MBW
0000	I	000004	MHBW	0000	I	000005	N	0010	I	000001	NASTRN	0004	I	000006	NBW	0024	I	000003	NBWT
0000	I	000011	NC	0024	I	000001	NELT	0004	I	000010	NEPJ	0004	I	000003	NET	0007		000000	NFRAC
0004		000014	NFST	0015		000001	NGD	0004	I	000005	NHBW	0024	I	000004	NHBWT	0000	I	000006	NI
0011	I	000114	NOD	0004		000007	NPPJ	0000	I	000012	NPQ	0004	I	000004	NPS	0024	I	000000	NPST
0000	I	000010	NR	0010		000002	NSLIDE	0003	I	007640	NSTAB	0013	R	000000	OM	0014	R	000300	PM
0007		000002	PMIU	0006		000044	PNIU	0003	R	000000	S	0014	R	000210	SF	0006		000011	SLC
0006		000000	SLT	0017	R	000000	STL	0016	R	000000	STSN	0016	R	001254	STSPRV	0007		000001	TMIU
0023	R	000601	VBD	0022	R	000000	VER	0012	R	000344	VERN	0005	R	000344	VERPRV	0014	R	000340	VL
0014	R	000220	W	0020	R	000710	X	0012	R	000000	XN	0005	R	000000	XPRV				

```
00101      1*          SUBROUTINE CEL3CV(L)                                                    000000
00101      2*     C                                                                            000000
00101      3*     C**************************                                                 000000
00101      4*     C                                                                            000000
00101      5*     C        ASSEMBLE SYSTEM MATRICES FOR CONSERVATION EQUATIONS                 000000
00101      6*     C                                                                            000000
00101      7*     C**************************                                                 000000
00101      8*     C                                                                            000000
00103      9*          INCLUDE PARAM1,LIST                                                     000000
00103      9*     PARAM1 PROC                                                                  000000
00103      9*     C                                                                            000000
00104      9*          PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                             000000
00105      9*          PARAMETER NPELG= 8,NPE= 8                                               000000
00106      9*          PARAMETER NDFRS=53,NDINT=10                                             000000
00106      9*     C                                                                            000000
00107      9*          PARAMETER NRMAX=2000,NCMAX=2                                            000000
00110      9*          PARAMETER NGPLG=2,NGP=2                                                 000000
00111      9*          PARAMETER NCNT=6, NFRC=NPM                                              000000
00112      9*          PARAMETER IU=5,NDPJT= 10,NTSTEP=200                                     000000
00113      9*          PARAMETER NBV=NPM,NBS=NPM,NBJ=NPM                                       000000
00113      9*     END                                                                          000000
00114     10*          INCLUDE CONSRV,LIST                                                     000000
00114     10*     CONSRV PROC                                                                  000000
00115     10*          COMMON /SYSMTX/ S(NRMAX,NCMAX),NSTAB                                    000000
00116     10*          COMMON /CEL21/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,          000000
00116     10*         1  ITER,DELT,NFST                                                        000000
00117     10*          COMMON /CEL23/ XPRV(NPMT,3), VERPRV(NPMT,IU)                            000000
00120     10*          COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)        000000
00121     10*          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                                000000
00122     10*          COMMON /NCASE/ INDEX,NASTRN,NSLIDE                                      000000
00123     10*          COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)                               000000
00124     10*          COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),           000000
00124     10*         1C(NPMT,NCNT,NCNT)                                                       000000
00125     10*          COMMON /IMPCT2/ OM(NPE,3),FJCB(NPE),CFRC(NPELG,NCNT,NCNT)               000000
```

```
00126     10*           COMMON /ELMLAG/ CD(NPELG,NPELG),CS(NPELG,NPELG),CL(NPELG),                000000
00126     10*          1SF(NPELG),W(3,NPELG),DF(3,NPELG),PM(NPELG),ELXYZ(NPELG,3),               000000
00126     10*          2VL(NPELG,IU)                                                             000000
00127     10*           COMMON /PARDOF/ MAXDOF,NGD                                               000000
00130     10*           COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)                            000000
00131     10*           COMMON /STSELM/ STL(NPE,3,3)                                             000000
00132     10*           COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)        000000
00133     10*           COMMON /DISELM/ DL(NPELG,3),BF(NPELG,3)                                  000000
00134     10*           COMMON /SOLCNV/ VER(NPMT,IU)                                             000000
00135     10*           COMMON /BDVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)                        000000
00135     10*      END                                                                           000000
00136     11*           COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                               000000
00136     12*      C                                                                             000000
00136     13*      C                                                                             000000
00137     14*           IF (NASTRN .NE. 0) GO TO 41                                              000000
00141     15*           DO 40 I=1,NRMAX                                                          000005
00144     16*           DO 40 J=1,NCMAX                                                          000005
00147     17*        40 S(I,J)=0.0                                                               000005
00152     18*        41 CONTINUE                                                                 000013
00153     19*           IF (INDEX .EQ. 1) GO TO 46                                               000013
00155     20*           LIN=1                                                                    000015
00156     21*           GO TO 170                                                                000017
00157     22*        46 CONTINUE                                                                 000021
00160     23*           LIN=NET+1                                                                000021
00161     24*           MBW=NBWT                                                                 000023
00162     25*           MHBW=NHBWT                                                               000025
00163     26*           IF (L .LE. 3) GO TO 47                                                   000027
00165     27*           MBW=NBW                                                                  000033
00166     28*           MHBW=NHBW                                                                000035
00167     29*        47 CONTINUE                                                                 000040
00170     30*           DO 165 N=1,NET                                                           000040
00173     31*           GO TO (48,85),INNER                                                      000050
00174     32*        48 DO  80 I=1,NPE                                                           000060
00177     33*           NI=NOD(N,I)                                                              000106
00200     34*           FJCB(I)=FJPRV(NI)                                                        000107
00201     35*           DO 50 J=1,3                                                              000117
00204     36*           ELXYZ(I,J)=XPRV(NI,J)                                                    000117
00205     37*        50 VL(I,J)=VERPRV(NI,J)/VERPRV(NI,5)                                        000120
00207     38*           IF(ITER  .GT. 1 .OR. N .GE. NEPJ) GO TO 68                               000124
00211     39*           VL(I,3)=0.0                                                              000126
00212     40*           VL(I,4)=0.0                                                              000127
00213     41*        68 DO 75 J=1,3                                                              000131
00216     42*           BF(I,J)=BFS(NI,J)                                                        000141
00217     43*           DO 75 K=1,3                                                              000144
00222     44*        75 STL(I,J,K)=STSPRV(NI,J,K)                                                000144
00225     45*        80 CONTINUE                                                                 000171
00227     46*           GO TO 110                                                                000171
00230     47*        85 DO 100 I=1,NPE                                                           000173
00233     48*           NI=NOD(N,I)                                                              000205
00234     49*           FJCB(I)=ALPHA*FJPRV(NI)+(1.0-ALPHA)*FJN(NI)                              000212
00235     50*           DO 90 J=1,3                                                              000225
00240     51*           ELXYZ(I,J)=ALPHA*XPRV(NI,J)+(1.0-ALPHA)*XN(NI,J)                         000225
00241     52*           VL(I,J)=ALPHA*VERPRV(NI,J)/VERPRV(NI,5)+(1.0-ALPHA)*                     000232
00241     53*          1  VERN(NI,J)/VERN(NI,5)                                                 000232
00242     54*           BF(I,J)=BFS(NI,J)                                                        000242
00243     55*           DO 90 K=1,3                                                              000246
```

```
00246     56*        90 STL(I,J,K)=ALPHA*STSPRV(NI,J,K)+(1.0-ALPHA)*STSN(NI,J,K)                    000246
00251     57*       100 CONTINUE                                                                    000275
00253     58*       110 DO 118 I=1,NPE                                                              000275
00256     59*           NI=NOD(N,I)                                                                 000306
00257     60*           IF(KODE(NI)) 112,112,115                                                    000310
00262     61*       112 DO 113 J=1,3                                                                000320
00265     62*       113 OM(I,J)=VL(I,J)                                                             000320
00267     63*           GO TO 118                                                                   000322
00270     64*       115 DO 116 J=1,3                                                                000324
00273     65*       116 OM(I,J)=0.0                                                                 000331
00275     66*       118 CONTINUE                                                                    000335
00277     67*           CALL ELCELV(NGP,NPE,ELXYZ,SF,DF,W,VL,OM,FJCB,BF,STL,CD,CS,CL,               000335
00277     68*          1  IU,DELT,L,IORDC)                                                          000335
00300     69*           DO 120 I=1,NPE                                                              000361
00303     70*           NI=NOD(N,I)                                                                 000421
00304     71*           PM(I)=VERPRV(NI,L)                                                          000422
00305     72*           IF(ITER .GT. 1 .OR. N .GE. NEPJ) GO TO 120                                  000426
00307     73*           IF(L .EQ. 3 .OR. L .EQ. 4)PM(I)=0.0                                         000430
00311     74*       120 CONTINUE                                                                    000436
00313     75*           DO 140 I=1,NPE                                                              000436
00316     76*           DO 140 J=1,NPE                                                              000447
00321     77*       140 CL(I)=CL(I)+CD(I,J)*PM(J)                                                   000447
00324     78*           IF (NASTRN .EQ. 0) GO TO 158                                                000460
00326     79*           IF (NSTAB .NE. 0) GO TO 155                                                 000462
00330     80*           DO 150 I=1,NPE                                                              000471
00333     81*           NI=NOD(N,I)                                                                 000474
00334     82*           IF (KODE(NI). .LT. 2) GO TO 150                                             000476
00336     83*           DO 148 J=1,NPE                                                              000511
00341     84*       148 CS(I,J)=0.0                                                                 000511
00343     85*           CS(I,I)=1.0                                                                 000512
00344     86*           CL(I)=VERPRV(NI,L)                                                          000514
00345     87*       150 CONTINUE                                                                    000523
00347     88*       155 CONTINUE                                                                    000523
00350     89*           CALL ELMTRX(CS,CL,NPS,NEMT,NPELG,NOD,N,CD,IBDVT,IBDV,VBD,NBV,L,1)           000523
00351     90*           GO TO 165                                                                   000542
00352     91*       158 CONTINUE                                                                    000544
00353     92*           DO 160 I=1,NPE                                                              000544
00356     93*           NR=NOD(N,I)                                                                 000554
00357     94*           S(NR,MBW)=S(NR,MBW)+CL(I)                                                   000557
00360     95*           DO 160 J=1,NPE                                                              000564
00363     96*           NC=NOD(N,J)-NR+MHBW                                                         000564
00364     97*       160 S(NR,NC)=S(NR,NC)+CS(I,J)                                                   000571
00367     98*       165 CONTINUE                                                                    000611
00371     99*           IF (LIN .GT. NELT) GO TO 250                                                000611
00373    100*       170 IF (L .GT. 3) GO TO 250                                                     000616
00375    101*           DO 200 N=LIN,NELT                                                           000621
00400    102*           GO TO (171,180),INNER                                                       000632
00401    103*       171 DO 175 I=1,NPELG                                                            000642
00404    104*           NI= NOD(N,I)                                                                000647
00405    105*           DO 175 J=1,3                                                                000650
00410    106*           ELXYZ(I,J)=X(NI,J)                                                          000655
00411    107*            DL(I,J)=DISPRV(NI,J)                                                       000656
00412    108*           BF(I,J)=BFS(NI,J)                                                           000660
00413    109*       175 CONTINUE                                                                    000670
00416    110*           GO TO 188                                                                   000670
00417    111*       180 DO 185 I=1,NPELG                                                            000672
```

```
00422    112*            NI=NOD(N,I)                                                    000702
00423    113*            DO 185 J=1,3                                                   000703
00426    114*            ELXYZ(I,J)=X(NI,J)                                             000710
00427    115*             DL(I,J)=ALPHA*DISPRV(NI,J)+1.0-ALPHA)*DISN(NI,J)              000711
00430    116*            BF(I,J)=BFS(NI,J)                                              000717
00431    117*        185 CONTINUE                                                       000730
00434    118*        188 CONTINUE                                                       000730
00435    119*            DO 189 I=1,NPELG                                               000730
00440    120*            NI=NOD(N,I)                                                    000735
00441    121*            DO 189 J=1,NCNT                                                000736
00444    122*            DO 189 K=1,NCNT                                                000744
00447    123*        189 CFRC(I,J,K)=C(NI,J,K)                                          000744
00453    124*            CALL ELAGVA(NGPLG,NPELG,ELXYZ,SF,DF,W,DL,    BF,CD,CS,CL,LI,    000762
00453    125*           1  CFRC,NCNT,DELT,L,IORDL)                                      000762
00454    126*            DO 190 I=1,NPELG                                               001005
00457    127*            NI=NOD(N,I)                                                    001037
00460    128*            PM(I)=VERPRV(NI,L)                                             001040
00461    129*            IF(ITER .GT. 1 .OR. N .GE. NEPJ) GO TO 190                     001044
00463    130*            IF(L .EQ. 3 ) PM(I)=0.0                                        001046
00465    131*        190 CONTINUE                                                       001054
00467    132*            DO 191 I=1,NPELG                                               001054
00472    133*            DO 191 J=1,NPELG                                               001065
00475    134*        191 CL(I)=CL(I)+CD(I,J)*PM(J)                                      001065
00500    135*            IF (NASTRN .EQ. 0) GO TO 197                                   001076
00502    136*            IF (NSTAB .NE. 0) GO TO 195                                    001100
00504    137*            DO 193 I=1,NPELG                                               001107
00507    138*            NI=NOD(N,I)                                                    001112
00510    139*            IF (KODE(NI) .LT. 2) GO TO 193                                 001114
00512    140*            DO 192 J=1,NPELG                                               001127
00515    141*        192 CS(I,J)=0.0                                                    001127
00517    142*            CS(I,I)=1.0                                                    001130
00520    143*            CL(I)=VERPRV(NI,L)                                             001132
00521    144*        193 CONTINUE                                                       001141
00523    145*        195 CONTINUE                                                       001141
00524    146*            CALL ELMTRX(CS,CL,NPST,NEMT,NPELG,NOD,N,CD,IBDVT,IBDV,VBD,NBV,L,I) 001141
00525    147*            GO TO 200                                                      001160
00526    148*        197 CONTINUE                                                       001162
00527    149*            DO 198 I=1,NPELG                                               001162
00532    150*            NR=NOD(N,I)                                                    001175
00533    151*            S(NR,NBWT) =S(NR,NBWT) +CL(I)                                  001200
00534    152*            DO 198 J=1,NPELG                                               001205
00537    153*            NC=NOD(N,J)-NR+NHBWT                                           001205
00540    154*        198 S(NR,NC)=S(NR,NC)+CS(I,J)                                      001212
00543    155*        200 CONTINUE                                                       001234
00545    156*        250 CONTINUE                                                       001234
00546    157*            IF (NASTRN .EQ. 0) GO TO 280                                   001234
00550    158*            RETURN                                                         001235
00551    159*        280 IF (L-4)292,293,293                                            001241
00554    160*        292 NPQ=NPST                                                       001244
00555    161*            MBW=NBWT                                                       001246
00556    162*            MHBW=NHBWT                                                     001250
00557    163*            GO TO 295                                                      001252
00560    164*        293 NPQ=NPS                                                        001254
00561    165*            MBW=NBW                                                        001255
00562    166*            MHBW=NHBW                                                      001257
00563    167*        295 CONTINUE                                                       001262
```

```
      C3VL                                                                DATE 010978

00564     168*          IF (NSTAB .NE. 0) GO TO 302                          001262
00566     169*          DO 300 I=1,NPQ                                       001307
00571     170*          IF (KODE(I) .LT. 2) GO TO 300                        001312
00573     171*          DO 298 J=1,MBW                                       001322
00576     172*      298 S(I,J)=0.0                                           001322
00600     173*          S(I,MHBW)=1.0                                        001323
00601     174*          S(I,MBW)=VERPRV(I,L)                                 001325
00602     175*      300 CONTINUE                                             001335
00604     176*      302 CONTINUE                                             001335
00605     177*          IF(IBDVT(L) .EQ. 0) GO TO320                         001335
00607     178*          II=IBDVT(L)                                          001337
00610     179*          DO 310 I=1,II                                        001352
00613     180*          IE=IBDV(I,L)                                         001362
00614     181*          DO 305 J=1,MBW                                       001363
00617     182*      305 S(IE,J)=0.0                                          001374
00621     183*          S(IE,MHBW)=1.0                                       001375
00622     184*      310 S(IE,MBW)=VBD(I,L)                                   001377
00624     185*      320 CALL BNDEQ(S,NRMAX,NCMAX,NPQ,MHBW)                   001403
00625     186*          DO 340 I=1,NPQ                                       001411
00630     187*      340 VER(I,L) =S(I,MBW)                                   001427
00632     188*          RETURN                                               001431
00633     189*          END                                                  001525

        END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P       ELAG
```

```
ELAG                                                                    DATE 010978

@FOR,S    ELAG,ELAG
HSA  E3 -01/09/78-23:43:36 (0,)

     SUBROUTINE ELAGVA    ENTRY POINT 000676

     STORAGE USED: CODE(1) 001016; DATA(0) 000356; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003    SFCALL
      0004    SFCALQ
      0005    SFCALC
      0006    MATMUL
      0007    INVDET
      0010    NERR2$
      0011    NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001    000127 113G     0001    000131 117G     0001    000146 126G     0001    000150 131G     0001    000152 134G
      0001    000326 160G     0001    000330 164G     0001    000341 172G     0001    000342 175G     0001    000360 203G
      0001    000366 207G     0001    000367 212G     0001    000416 221G     0001    000420 224G     0001    000421 227G
      0001    000456 237G     0001    000514 247G     0001    000543 254G     0001    000616 266G     0001    000622 272G
      0001    000170 55L      0001    000176 56L      0001    000204 57L      0001    000211 58L      0000 R  000003 C
      0000 R  000231 CFT      0000 R  000240 CNI      0000 R  000230 CNST     0000 R  000241 CI       0000 R  000242 C2
      0000 R  000227 DET      0000 R  000124 DS       0000 R  000232 FJ       0000 R  000140 GAUSS    0000 R  000233 GFOS
      0000 R  000200 GJ       0000 R  000211 GJINV    0000 R  000135 H        0000 I  000222 I        0000 I  000224 II
      0000 I  000236 IJ       0000    000256 INJP$    0000 I  000223 J        0000 I  000225 JJ       0000 I  000237 JK
      0000 I  000234 K        0000 I  000226 KK       0000 I  000235 LEE      0000 R  000160 WT       0000 R  000000 XYZ

00101     1*          SUBROUTINE ELAGVA(NGP,NN,ELXYZ,SF,DF,W,DIS,BF,CD,CS,                    000076
00101     2*         1CSL,LI,CFRC,IDIM,DT,L,IORD)                                             000076
00103     3*          DIMENSION ELXYZ(NN,3),SF(NN),DF(3,NN),W(3,NN),CD(NN,NN),CS(NN,NN)       000076
00103     4*         1  ,DIS(NN,3),CSL(NN),BF(NN,3),CFRC(NN,IDIM,IDIM)                        000076
00104     5*          DIMENSION XYZ(3),C(9,9),DS(3,3),LI(3,3),H(3)                            000076
00105     6*          DIMENSION GAUSS(4,4),WT(4,4),GJ(3,3),GJINV(3,3)                         000076
00106     7*          DATA GAUSS/4*0.,-.57735027,.57735027,2*0.,                              000076
00106     8*         1      -.77459667,0.,.77459667,0.,                                       000076
00106     9*         2           -.86113631,-.33998104,.33998104,.86113631/                   000076
00110    10*          DATA WT/2.,3*0.,2*1.,2*0.,.55555556,.88888889,.55555556,0.,             000076
00110    11*         1        .34785485,2*.65214515,.34785485/                                000076
00112    12*          DO 50 I=1,NN                                                            000076
00115    13*          CSL(I)=0.0                                                              000127
00116    14*          DO 50 J=1,NN                                                            000131
00121    15*          CS(I,J)=0.0                                                             000131
00122    16*       50 CD(I,J)=0.0                                                             000131
00125    17*          DO 150 II=1,NGP                                                         000152
00130    18*          DO 150 JJ=1,NGP                                                         000152
00133    19*          DO 150 KK=1,NGP                                                         000152
```

```
00136     20*            XYZ(1)=GAUSS(II,NGP)                                                  000152
00137     21*            XYZ(2)=GAUSS(JJ,NGP)                                                  000153
00140     22*            XYZ(3)=GAUSS(KK,NGP)                                                  000155
00141     23*            GO TO (55,56,57),IORD                                                 000157
00142     24*         55 CALL SFCALL(XYZ,SF,DF)                                                000170
00143     25*            GO TO 58                                                              000174
00144     26*         56 CALL SFCALQ(XYZ,SF,DF)                                                000176
00145     27*            GO TO 58                                                              000202
00146     28*         57 CALL SFCALC(XYZ,SF,DF)                                                000204
00147     29*         58 CONTINUE                                                              000211
00150     30*            CALL MATMUL(DF,ELXYZ,GJ,3,NN,3)                                       000211
00151     31*            CALL INVDET(GJ,GJINV,DET)                                             000220
00152     32*            CALL MATMUL(GJINV,DF,W,3,3,NN)                                        000225
00153     33*            CNST=DET*WT(II,NGP)*WT(JJ,NGP)*WT(KK,NGP)                             000235
00154     34*            CFT=CNST*DT                                                           000242
00155     35*            FJ=0.0                                                                000244
00156     36*            GFOS=0.0                                                              000245
00157     37*            DO 60 J=1,3                                                           000326
00162     38*            H(J)=0.0                                                              000326
00163     39*            DO 60 K=1,3                                                           000330
00166     40*         60 DS(J,K)=0.0                                                           000330
00171     41*            DO 65 I=1,IDIM                                                        000342
00174     42*            DO 65 J=1,IDIM                                                        000342
00177     43*         65 C(I,J)=0.0                                                            000342
00202     44*            DO 72 I=1,NN                                                          000360
00205     45*            FJ=FJ+SF(I)* BF(I,L)                                                  000360
00206     46*            DO 72 J=1,IDIM                                                        000367
00211     47*            DO 72 K=1,IDIM                                                        000367
00214     48*         72 C(J,K)=C(J,K)+CFRC(I,J,K)*SF(I)                                       000367
00220     49*            DO  75 I=1,NN                                                         000421
00223     50*            DO  75 J=1,3                                                          000421
00226     51*            DO  75 K=1,3                                                          000421
00231     52*         75 DS(J,K)=DS(J,K)+W(K,I)*DIS(I,J)                                       000421
00235     53*            LEE=LI(L,L)                                                           000440
00236     54*            DO 105 I=1,3                                                          000456
00241     55*            IJ=LI(I,I)                                                            000456
00242     56*            JK=LI(L,I)                                                            000457
00243     57*            H(I)=H(I)-0.50*G(JK,JK)*(DS(L,I)+DS(I,L))                             000461
00244     58*        105 GFOS=GFOS- G(LEE,IJ)*DS(I,I)                                          000471
00246     59*            DO 140 I=1,NN                                                         000500
00251     60*            CNI=CNST*SF(I)                                                        000514
00252     61*            CSL(I)=CSL(I)+CNI*DT*FJ+CFT*GFOS*W(L,I) +CFT*(H(1)*W(1,I)             000516
00252     62*           1 +H(2)*W(2,I)+H(3)*W(3,I))                                            000516
00253     63*            DO 140 J=1,NN                                                         000543
00256     64*        140 CD(I,J)=CD(I,J)+CNI*SF(J)                                             000543
00261     65*        150 CONTINUE                                                              000565
00265     66*            DO 200 I=1,NN                                                         000565
00270     67*            CSL(I)=2.50*CSL(I)                                                    000616
00271     68*            DO 200 J=1,NN                                                         000622
00274     69*            C1=CD(I,J)                                                            000622
00275     70*            C2=4.0*CD(I,J)                                                        000623
00276     71*            CD(I,J)=C2                                                            000625
00277     72*        200 CS(I,J)=C1                                                            000626
00302     73*            RETURN                                                                000641
00303     74*            END                                                                   001015
```

```
ELAG                                                    DATE 010978

        END OF COMPILATION:       NO  DIAGNOSTICS.

@HDG,P      ELCA
```

```
@FOR,S    ELCA,ELCA
HSA  E3 -01/09/78-23:43:39 (0,)

   SUBROUTINE ELCELA    ENTRY POINT 001111

   STORAGE USED: CODE(1) 001244; DATA(0) 000414; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

     0003    SFCALL
     0004    SFCALQ
     0005    SFCALC
     0006    MATMUL
     0007    INVDET
     0010    NERR2$
     0011    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000136 113G      0001   000140 117G      0001   000660 120L      0001   000155 126G      0001   000160 131G
     0001   000163 134G      0001   000347 160G      0001   000353 166G      0001   000366 176G      0001   000367 201G
     0001   000407 207G      0001   000416 213G      0001   000427 220G      0001   000465 227G      0001   000467 232G
     0001   000470 235G      0001   000522 246G      0001   000524 251G      0001   000525 254G      0001   000555 264G
     0001   000572 271G      0001   000621 300G      0001   000622 303G      0001   000652 314G      0001   000712 322G
     0001   000743 327G      0001   001032 342G      0001   001033 345G      0001   000203 55L       0001   000211 56L
     0001   000217 57L       0001   000224 58L       0000 R 000047 C         0000 R 000261 CFT       0000 R 000270 CNI
     0000 R 000260 CNST      0000 R 000271 C1        0000 R 000272 C2        0000 R 000257 DET       0000 R 000265 DIV
     0000 R 000025 DOM       0000 R 000036 DV        0000 R 000262 FJ        0000 R 000170 GAUSS     0000 R 000263 GFOS
     0000 R 000230 GJ        0000 R 000241 GJINV     0000 R 000017 H         0000 I 000252 I         0000 I 000254 II
     0000 I 000267 IJ        0000   000306 INJP$     0000 I 000253 J         0000 I 000255 JJ        0000 I 000264 K
     0000 I 000256 KK        0000 I 000266 LEE       0000 R 000022 OM        0000 R 000006 S         0000 R 000003 V
     0000 R 000210 WT        0000 R 000000 XYZ

00101      1*          SUBROUTINE ELCELA(NGP,NN,ELXYZ,SF,DF,W,VEL,OME,FJCB,STR,CD,CS,    000077
00101      2*         1CSL,LI,CFRC,IDIM,DI,L1,L2,IORD)                                  000077
00103      3*          DIMENSION ELXYZ(NN,3),SF(NN),DF(3,NN),W(3,NN),CD(NN,NN),CS(NN,NN) 000077
00103      4*         1,STR(NN,3,3),CSL(NN),FJCB(NN),VEL(NN,3),OME(NN,3)                000077
00103      5*         2  ,CFRC(NN,IDIM,IDIM)                                            000077
00104      6*          DIMENSION XYZ(3),V(3),S(3,3),H(3),OM(3),DOM(3,3),DV(3,3),LI(3,3) 000077
00104      7*         1  ,C(9,9)                                                        000077
00105      8*          DIMENSION GAUSS(4,4),WT(4,4),GJ(3,3),GJINV(3,3)                   000077
00106      9*          DATA GAUSS/4*0.,-.57735027,.57735027,2*0.,                        000077
00106     10*         1      -.77459667,0.,.77459667,0.,                                000077
00106     11*         2           -.86113631,-.33998104,.33998104,.86113631/            000077
00110     12*          DATA WT/2.,3*0.,2*1.,2*0.,.55555556,.88888889,.55555556,0.,       000077
00110     13*         1       .34785485,2*.65214515,.34785485/                          000077
00112     14*          DO 50 I=1,NN                                                      000077
00115     15*          CSL(I)=0.0                                                        000136
00116     16*          DO 50 J=1,NN                                                      000140
```

```
00121     17*          CS(I,J)=0.0                                                    000140
00122     18*       50 CD(I,J)=0.0                                                    000140
00125     19*          DO 150 II=1,NGP                                                000163
00130     20*          DO 150 JJ=1,NGP                                                000163
00133     21*          DO 150 KK=1,NGP                                                000163
00136     22*          XYZ(1)=GAUSS(II,NGP)                                           000163
00137     23*          XYZ(2)=GAUSS(JJ,NGP)                                           000164
00140     24*          XYZ(3)=GAUSS(KK,NGP)                                           000167
00141     25*          GO TO (55,56,57),IORD                                          000172
00142     26*       55 CALL SFCALL(XYZ,SF,DF)                                         000203
00143     27*          GO TO 58                                                       000207
00144     28*       56 CALL SFCALQ(XYZ,SF,DF)                                         000211
00145     29*          GO TO 58                                                       000215
00146     30*       57 CALL SFCALC(XYZ,SF,DF)                                         000217
00147     31*       58 CONTINUE                                                       000224
00150     32*          CALL MATMUL(DF,ELXYZ,GJ,3,NN,3)                                000224
00151     33*          CALL INVDET(GJ,GJINV,DET)                                      000233
00152     34*          CALL MATMUL(GJINV,DF,W,3,3,NN)                                 000240
00153     35*          CNST=DET*WT(II,NGP)*WT(JJ,NGP)*WT(KK,NGP)                      000250
00154     36*          CFT=CNST*DT                                                    000257
00155     37*          FJ=0.0                                                         000261
00156     38*          GFOS=0.0                                                       000262
00157     39*          DO 60 J=1,3                                                    000347
00162     40*          OM(J)=0.0                                                      000347
00163     41*          V(J)=0.0                                                       000347
00164     42*          H(J)=0.0                                                       000350
00165     43*          DO 60 K=1,3                                                    000353
00170     44*          DOM(J,K)=0.0                                                   000353
00171     45*          S(J,K)=0.0                                                     000353
00172     46*       60 DV(J,K)=0.0                                                    000354
00175     47*          DO 65 I=1,IDIM                                                 000367
00200     48*          DO 65 J=1,IDIM                                                 000367
00203     49*       65 C(I,J)=0.0                                                     000367
00206     50*          DO 70 I=1,NN                                                   000407
00211     51*          FJ=FJ+SF(I)*FJCB(I)                                            000407
00212     52*          DO 70 J=1,3                                                    000416
00215     53*          OM(J)=OM(J)+SF(I)*OME(I,J)                                     000416
00216     54*          V(J)=V(J)+SF(I)*VEL(I,J)                                       000421
00217     55*          DO 70 K=1,3                                                    000427
00222     56*       70 S(J,K)=S(J,K)+SF(I)*STR(I,J,K)                                 000427
00226     57*          DO 72 I=1,NN                                                   000470
00231     58*          DO 72 J=1,IDIM                                                 000470
00234     59*          DO 72 K=1,IDIM                                                 000470
00237     60*       72 C(J,K)=C(J,K)+CERC(I,J,K)*SF(I)                                000470
00243     61*          CFT=CFT*FJ                                                     000510
00244     62*          DIV=0.0                                                        000513
00245     63*          DO  75 I=1,NN                                                  000525
00250     64*          DO  75 J=1,3                                                   000525
00253     65*          DO  75 K=1,3                                                   000525
00256     66*          DOM(J,K)=DOM(J,K)+W(K,I)*OME(I,J)                              000525
00257     67*       75 DV(J,K)=DV(J,K)+W(K,I)*VEL(I,J)                                000530
00263     68*          DO 78 J=1,3                                                    000555
00266     69*       78 DIV=DIV+DOM(J,J)                                               000555
00270     70*          DO 100 J=1,3                                                   000572
00273     71*          GFOS=GFOS-0.50*S(L1,J)*(DV(J,L2)-DV(L2,J))+                    000572
00273     72*         1  0.50*S(J,L2)*(DV(L1,J)-DV(J,L1))                            000572
```

```
      ELCA                                                                    DATE 010978

00274     73*      100 CONTINUE                                                000607
00276     74*          LEE=LI(L1,L2)                                           000607
00277     75*          DO 105 I=1,3                                            000611
00302     76*          DO 105 J=1,3                                            000622
00305     77*          IJ=LI(I,J)                                              000622
00306     78*      105 GFOS=GFOS+0.50*C(LEE,IJ)*(DV(I,J)+DV(J,I))              000623
00311     79*          IF(L1 .NE. L2) GO TO 120                                000644
00313     80*          DO 110 I=1,3                                            000652
00316     81*      110 GFOS=GFOS-0.33333333*C(LEE,I)*DIV                       000652
00320     82*      120 CONTINUE                                                000660
00321     83*          DO 140 I=1,NN                                           000660
00324     84*          CNI=CNST*SF(I)*FJ                                       000723
00325     85*          CSL(I)=CSL(I)+CNI*DT*GFOS                               000727
00326     86*          DO 140 J=1,NN                                           000743
00331     87*          CS(I,J)=CS(I,J)+CFT*(W(1,I)*OM(1)+W(2,I)*OM(2)+W(3,I)*OM(3))*SF(J)  000743
00331     88*         1  +CNI*DT*DIV*SF(J)                                     000743
00332     89*      140 CD(I,J)=CD(I,J)+CNI*SF(J)                               000751
00335     90*      150 CONTINUE                                                001002
00341     91*          DO 200 I=1,NN                                           001002
00344     92*          DO 200 J=1,NN                                           001033
00347     93*          C1=CD(I,J)-0.33333333*CS(I,J)                           001033
00350     94*          C2=CD(I,J)+0.66666667*CS(I,J)                           001036
00351     95*          CD(I,J)=C2                                              001042
00352     96*      200 CS(I,J)=C1                                              001043
00355     97*          RETURN                                                  001054
00356     98*          END                                                     001243

      END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      ELCD
```

```
@FOR,S    ELCD,ELCD
HSA, E3 -01/09/78-23;43;42. (0,)

     SUBROUTINE ELCELD    ENTRY POINT 000527

     STORAGE USED: CODE(1) 000636; DATA(0) 000201; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003    SFCALL
      0004    SFCALQ
      0005    SFCALC
      0006    MATMUL
      0007    INVDET
      0010    NERR2$
      0011    NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001    000061 113G      0001    000063 117G      0001    000074 125G      0001    000077 130G      0001    000102 133G
      0001    000235 156G      0001    000256 164G      0001    000264 170G      0001    000332 201G      0001    000362 206G
      0001    000452 221G      0001    000453 224G      0001    000122 55L       0001    000130 56L       0001    000136 57L
      0001    000143 58L       0000 R  000104 CI        0000 R  000101 CNST      0000 R  000105 C1        0000 R  000106 C2
      0000 R  000100 DET       0000 R  000103 DIV       0000 R  000102 FJ        0000 R  000011 GAUSS     0000 R  000051 GJ
      0000 R  000062 GJINV     0000 I  000073 I         0000 I  000075 II        0000    000114 INJP$     0000 I  000074 J
      0000 I  000076 JJ        0000 I  000077 KK        0000 R  000006 OM        0000 R  000003 V         0000 R  000031 WT
      0000 R  000000 XYZ
```

```
00101      1*          SUBROUTINE ELCELD(NGP,NN,ELXYZ,SF,DF,W,VEL,OME,FJCB,CO,CS,                    000037
00101      2*         1CSL,DT,L,IORD)                                                                000037
00103      3*          DIMENSION ELXYZ(NN,3),SF(NN),DF(3,NN),W(3,NN),CO(NN,NN),CSL(NN),              000037
00103      4*         1  VEL(NN,1),OME(NN,1),FJCB(NN),CS(NN,NN)                                      000037
00104      5*          DIMENSION XYZ(3),V(3),OM(3)                                                   000037
00105      6*          DIMENSION GAUSS(4,4),WT(4,4),GJ(3,3),GJINV(3,3)                               000037
00106      7*          DATA GAUSS/4*0.,-.57735027,.57735027,2*0.,                                    000037
00106      8*         1      -.77459667,0.,.77459667,0.,                                            000037
00106      9*         2           -.86113631,-.33998104,.33998104,.86113631/                        000037
00110     10*          DATA WT/2.,3*0.,2*1.,2*0.,.55555556,.88888889,.55555556,0.,                   000037
00110     11*         1        .34785485,2*.65214515,.34785485/                                       000037
00112     12*          DO 50 I=1,NN                                                                  000037
00115     13*          CSL(I)=0.0                                                                    000061
00116     14*          DO 50 J=1,NN                                                                  000063
00121     15*       50 CO(I,J)=0.0                                                                   000063
00124     16*          DO 150 II=1,NGP                                                               000102
00127     17*          DO 150 JJ=1,NGP                                                               000102
00132     18*          DO 150 KK=1,NGP                                                               000102
00135     19*          XYZ(1)=GAUSS(II,NGP)                                                          000102
00136     20*          XYZ(2)=GAUSS(JJ,NGP)                                                          000103
00137     21*          XYZ(3)=GAUSS(KK,NGP)                                                          000106
```

```
ELCD                                                                    DATE 010978

00140     22*            GO TO (55,56,57),IORD                                000111
00141     23*       55   CALL SFCALL(XYZ,SF,DF)                               000122
00142     24*            GO TO 58                                             000126
00143     25*       56   CALL SFCALQ(XYZ,SF,DF)                               000130
00144     26*            GO TO 58                                             000134
00145     27*       57   CALL SFCALC(XYZ,SF,DF)                               000136
00146     28*       58   CONTINUE                                             000143
00147     29*            CALL MATMUL(DF,ELXYZ,GJ,3,NN,3)                      000143
00150     30*            CALL INVDET(GJ,GJINV,DET)                            000152
00151     31*            CALL MATMUL(GJINV,DF,W,3,3,NN)                       000157
00152     32*            CNST=DET*WT(II,NGP)*WT(JJ,NGP)*WT(KK,NGP)            000167
00153     33*            FJ=0.0                                               000176
00154     34*            DIV=0.0                                              000177
00155     35*            DO 60 I=1,3                                          000235
00160     36*            V(I)=0.0                                             000235
00161     37*       60   OM(I)=0.0                                            000235
00163     38*            DO 100 I=1,NN                                        000256
00166     39*            FJ=FJ+FJCB(I)*SF(I)                                  000256
00167     40*            DO 100 J=1,3                                         000264
00172     41*            V(J)=V(J)+VEL(I,J)*SF(I)                             000264
00173     42*            DIV=DIV+OME(I,J)*W(J,I)                              000267
00174     43*      100   OM(J)=OM(J)+OME(I,J)*SF(I)                           000273
00177     44*            CNST=CNST*FJ                                         000312
00200     45*            DO 120 I=1,NN                                        000332
00203     46*            CI=CNST*SF(I)                                        000344
00204     47*            CSL(I)=CSL(I)+CI* V(L) *DT                           000347
00205     48*            DO 120 J=1,NN                                        000362
00210     49*            CS(I,J)=CS(I,J)+CNST*(OM(1)*W(1,I)+OM(2)*W(2,I)+OM(3)*W(3,I))  000362
00210     50*          1   *DT*SF(J)+CI*DIV*SF(J)*DT                          000362
00211     51*      120   CD(I,J)=CD(I,J)+CI*SF(J)                            000371
00214     52*      150   CONTINUE                                             000422
00220     53*            DO 200 I=1,NN                                        000422
00223     54*            DO 200 J=1,NN                                        000453
00226     55*            C1=CD(I,J)-0.3333333*CS(I,J)                         000453
00227     56*            C2=CD(I,J)+0.6666667*CS(I,J)                         000456
00230     57*            CD(I,J)=C2                                           000462
00231     58*      200   CS(I,J)=C1                                           000463
00234     59*            RETURN                                               000474
00235     60*            END                                                  000635

     END OF COMPILATION:      NO  DIAGNOSTICS.

@HDG,P     ELCV
```

```
@FOR,S    ELCV,ELCV
HSA  E3 -01/09/78-23:43:45 (0,)

     SUBROUTINE ELCELV    ENTRY POINT 001031

     STORAGE USED: CODE(1) 001165; DATA(0) 000250; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   SFCALL
      0004   SFCALQ
      0005   SFCALC
      0006   MATMUL
      0007   INVDET
      0010   NERR2$
      0011   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000565 1001     0001   000075 113G     0001   000077 117G     0001   000114 126G     0001   000117 131G
      0001   000122 134G     0001   000264 160G     0001   000270 166G     0001   000313 176G     0001   000322 202G
      0001   000333 207G     0001   000374 220G     0001   000376 223G     0001   000377 226G     0001   000427 236G
      0001   000435 243G     0001   000462 253G     0001   000473 260G     0001   000527 266G     0001   000530 271G
      0001   000551 277G     0001   000553 302G     0001   000622 311G     0001   000645 315G     0001   000663 323G
      0001   000753 336G     0001   000754 341G     0001   000142 55L      0001   000150 56L      0001   000156 57L
      0001   000163 58L      0001   000500 90L      0000 R 000140 CFT      0000 R 000145 CNI      0000 R 000137 CNST
      0000 R 000146 C1       0000 R 000147 C2       0000 R 000136 DET      0000 R 000144 DIV      0000 R 000025 DOM
      0000 R 000036 DV       0000 R 000141 FJ       0000 R 000047 GAUSS    0000 R 000142 GFOS     0000 R 000107 GJ
      0000 R 000120 GJINV    0000 R 000017 H        0000 I 000131 I        0000 I 000133 II       0000   000160 INJP$
      0000 I 000132 J        0000 I 000134 JJ       0000 I 000143 K        0000 I 000135 KK       0000 R 000022 OM
      0000 R 000006 S        0000 R 000003 V        0000 R 000067 WT       0000 R 000000 XYZ

00101      1*          SUBROUTINE ELCELV(NGP,NN,ELXYZ,SF,DF,W,VEL,OME,FJCB,BF,STR,CD,          000044
00101      2*         1CS,CSL,IU,DT,L1,IORD)                                                  000044
00103      3*          DIMENSION ELXYZ(NN,3),SF(NN),DF(3,NN),W(3,NN),CD(NN,NN),CS(NN,NN)      000044
00103      4*         1,STR(NN,3,3),CSL(NN),BF(NN,3),VEL(NN,5),OME(NN,3),FJCB(NN)            000044
00104      5*          DIMENSION XYZ(3),V(3),S(3,3),H(3),OM(3),DOM(3,3),DV(3,3)               000044
00105      6*          DIMENSION GAUSS(4,4),WT(4,4),GJ(3,3),GJINV(3,3)                        000044
00106      7*          DATA GAUSS/4*0.,-.57735027,.57735027,2*0.,                             000044
00106      8*         1      -.77459667,0.,.77459667,0.,                                     000044
00106      9*         2           -.86113631,-.33998104,.33998104,.86113631/                 000044
00110     10*          DATA WT/2.,3*0.,2*1.,2*0.,.55555556,.88888889,.55555556,0.,            000044
00110     11*         1        .34785485,2*.65214515,.34785485/                              000044
00112     12*          DO 50 I=1,NN                                                           000044
00115     13*          CSL(I)=0.0                                                             000075
00116     14*          DO 50 J=1,NN                                                           000077
00121     15*          CS(I,J)=0.0                                                            000077
00122     16*       50 CD(I,J)=0.0                                                            000077
00125     17*          DO 150 II=1,NGP                                                        000122
```

```
00130    18*          DO 150 JJ=1,NGP                                    000122
00133    19*          DO 150 KK=1,NGP                                    000122
00136    20*          XYZ(1)=GAUSS(II,NGP)                               000122
00137    21*          XYZ(2)=GAUSS(JJ,NGP)                               000123
00140    22*          XYZ(3)=GAUSS(KK,NGP)                               000126
00141    23*          GO TO (55,56,57),IORD                              000131
00142    24*       55 CALL SFCALL(XYZ,SF,DF)                             000142
00143    25*          GO TO 58                                           000146
00144    26*       56 CALL SFCALQ(XYZ,SF,DF)                             000150
00145    27*          GO TO 58                                           000154
00146    28*       57 CALL SFCALC(XYZ,SF,DF)                             000156
00147    29*       58 CONTINUE                                           000163
00150    30*          CALL MATMUL(DF,ELXYZ,GJ,3,NN,3)                    000163
00151    31*          CALL INVDET(GJ,GJINV,DET)                          000172
00152    32*          CALL MATMUL(GJINV,DF,W,3,3,NN)                     000177
00153    33*          CNST=DET*WT(II,NGP)*WT(JJ,NGP)*WT(KK,NGP)          000207
00154    34*          CFT=CNST*DT                                        000216
00155    35*          FJ=0.0                                             000220
00156    36*          GFOS=0.0                                           000221
00157    37*          DO 60 J=1,3                                        000264
00162    38*          OM(J)=0.0                                          000264
00163    39*          V(J)=0.0                                           000264
00164    40*          H(J)=0.0                                           000265
00165    41*          DO 60 K=1,3                                        000270
00170    42*          DOM(J,K)=0.0                                       000270
00171    43*          S(J,K)=0.0                                         000270
00172    44*       60 DV(J,K)=0.0                                        000271
00175    45*          DO 70 I=1,NN                                       000313
00200    46*          FJ=FJ+SF(I)*FJCB(I)                                000313
00201    47*          DO 70 J=1,3                                        000322
00204    48*          OM(J)=OM(J)+SF(I)*OME(I,J)                         000322
00205    49*          V(J)=V(J)+SF(I)*VEL(I,J)                           000325
00206    50*          DO 70 K=1,3                                        000333
00211    51*       70 S(J,K)=S(J,K)+SF(I)*STR(I,J,K)                     000333
00215    52*          CFT=CFT*FJ                                         000362
00216    53*          DIV=0.0                                            000365
00217    54*          DO  75 I=1,NN                                      000377
00222    55*          DO  75 J=1,3                                       000377
00225    56*          DO  75 K=1,3                                       000377
00230    57*          DOM(J,K)=DOM(J,K)+W(K,I)*OME(I,J)                  000377
00231    58*       75 DV(J,K)=DV(J,K)+W(K,I)*VEL(I,J)                    000402
00235    59*          DO 78 J=1,3                                        000427
00240    60*       78 DIV=DIV+DOM(J,J)                                   000427
00242    61*          DO 81 J=1,3                                        000435
00245    62*       81 DIV=DIV-DV(J,J)                                    000435
00247    63*             IF(L1-4)85,90,100                               000440
00252    64*       85 DO 88 I=1,NN                                       000444
00255    65*       88 GFOS=GFOS+SF(I)*BF(I,L1)                           000462
00257    66*          DO 89 K=1,3                                        000473
00262    67*       89 H(K)=H(K)-S(L1,K)                                  000473
00264    68*          GO TO 100                                          000476
00265    69*       90 DO 95 I=1,NN                                       000500
00270    70*          DO 95 J=1,3                                        000530
00273    71*       95 GFOS=GFOS+SF(I)*BF(I,J)*VEL(I,J)                   000530
00276    72*          DO 96 J=1,3                                        000553
00301    73*          DO 96 K=1,3                                        000553
```

```
ELCV                                                                    DATE 010978

00304     74*          96 H(K)=H(K)-S(J,K)*V(J)                                    000553
00307     75*         100 CONTINUE                                                 000565
00310     76*             DO 120 I=1,NN                                            000565
00313     77*             CNI=SF(I)*CNST*FJ                                        000633
00314     78*             DO 110 J=1,3                                             000637
00317     79*         110 CSL(I)=CSL(I)+CFT*W(J,I)*H(J)                            000645
00321     80*             CSL(I)=CSL(I)+CNI*GFOS*DT                                000652
00322     81*             DO 120 J=1,NN                                            000663
00325     82*             CS(I,J)=CS(I,J)+CFT*(W(1,I)*OM(1)+W(2,I)*OM(2)+W(3,I)*OM(3))*  000663
00325     83*            1 SF(J)+CNI*DT*DIV*SF(J)                                  000663
00326     84*         120 CD(I,J)=CD(I,J)+CNI*SF(J)                                000671
00331     85*         150 CONTINUE                                                 000723
00335     86*             DO 200 I=1,NN                                            000723
00340     87*             DO 200 J=1,NN                                            000754
00343     88*             C1=CD(I,J)-0.33333333*CS(I,J)                            000754
00344     89*             C2=CD(I,J)+0.66666667*CS(I,J)                            000757
00345     90*             CD(I,J)=C2                                               000763
00346     91*         200 CS(I,J)=C1                                               000764
00351     92*         300 RETURN                                                   000775
00352     93*             END                                                      001164

        END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      ELGD
```

```
ELGD                                                                    DATE 010978

@FOR,S    ELGD;ELGD
HSA  E3 -01/09/78-23:43:48 (0,)

   SUBROUTINE ELDISA    ENTRY POINT 001041

   STORAGE USED: CODE(1) 001164; DATA(0) 000401; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SFCALL
    0004   SFCALQ
    0005   SFCALC
    0006   MATMUL
    0007   INVDET
    0010   NERR2$
    0011   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000145 113G      0001   000147 117G      0001   000165 126G      0001   000170 131G      0001   000173 134G
    0001   000374 161G      0001   000377 166G      0001   000410 174G      0001   000411 177G      0001   000430 205G
    0001   000443 212G      0001   000450 216G      0001   000513 226G      0001   000515 231G      0001   000516 234G
    0001   000552 244G      0001   000612 254G      0001   000643 261G      0001   000646 264G      0001   000753 301G
    0001   000760 305G      0001   000214 55L       0001   000222 56L       0001   000230 57L       0001   000235 58L
    0000 R 000006 C         0000 R 000234 CFT       0000 R 000244 CNI       0000 R 000233 CNST      0000 R 000245 C1
    0000 R 000246 C2        0000 R 000232 DET       0000 R 000127 DS        0000 R 000236 FJ        0000 R 000143 GAUSS
    0000 R 000237 GFOS      0000 R 000203 GJ        0000 R 000214 GJINV     0000 R 000140 H         0000 I 000225 I
    0000 I 000227 II        0000 I 000242 IJ        0000   000266 INJP$     0000 I 000226 J         0000 I 000230 JJ
    0000 I 000243 JK        0000 I 000240 K         0000 I 000231 KK        0000 I 000241 LEE       0000 R 000235 RHO
    0000 R 000003 V         0000 R 000163 WT        0000 R 000000 XYZ

00101     1*          SUBROUTINE ELDISA(NGP,NN,ELXYZ,SF,DF,W,DIS,VEL,BF,GD,GS,          000103
00101     2*         1CSL,LI,CFRC,IDIM,DT,L,IORD)                                       000103
00103     3*          DIMENSION ELXYZ(NN,3),SF(NN),DF(3,NN),W(3,NN),CD(NN,NN),CS(NN,NN)  000103
00103     4*         1  ,VEL(NN,5),CSL(NN),BF(NN,3),DIS(NN,3)                           000103
00103     5*         2  ,CFRC(NN,IDIM,IDIM)                                             000103
00104     6*          DIMENSION XYZ(3),V(3),C(9,9),LI(3,3),DS(3,3),H(3)                 000103
00105     7*          DIMENSION GAUSS(4,4),WT(4,4),GJ(3,3),GJINV(3,3)                   000103
00106     8*          DATA GAUSS/4*0.,-.57735027,.57735027,2*0.,                        000103
00106     9*         1       -.77459667,0.,.77459667,0.,                                000103
00106    10*         2            -.86113631,-.33998104,.33998104,.86113631/            000103
00110    11*          DATA WT/2.,3*0.,2*1.,2*0.,.55555556,.88888889,.55555556,0.,       000103
00110    12*         1        .34785485,2*.65214515,.34785485/                          000103
00112    13*          DO 50 I=1,NN                                                      000103
00115    14*          CSL(I)=0.0                                                        000145
00116    15*          DO 50 J=1,NN                                                      000147
00121    16*          CS(I,J)=0.0                                                       000147
00122    17*       50 CD(I,J)=0.0                                                       000147
00125    18*          DO 150 II=1,NGP                                                   000173
```

```
00130    19*          DO 150 JJ=1,NGP                                                000173
00133    20*          DO 150 KK=1,NGP                                                000173
00136    21*          XYZ(1)=GAUSS(II,NGP)                                           000173
00137    22*          XYZ(2)=GAUSS(JJ,NGP)                                           000175
00140    23*          XYZ(3)=GAUSS(KK,NGP)                                           000200
00141    24*          GO TO (55,56,57),IORD                                          000203
00142    25*       55 CALL SFCALL(XYZ,SF,DF)                                         000214
00143    26*          GO TO 58                                                       000220
00144    27*       56 CALL SFCALQ(XYZ,SF,DF)                                         000222
00145    28*          GO TO 58                                                       000226
00146    29*       57 CALL SFCALC(XYZ,SF,DF)                                         000230
00147    30*       58 CONTINUE                                                       000235
00150    31*          CALL MATMUL(DF,ELXYZ,GJ,3,NN,3)                                000235
00151    32*          CALL INVDET(GJ,GJINV,DET)                                      000244
00152    33*          CALL MATMUL(GJINV,DF,W,3,3,NN)                                 000251
00153    34*          CNST=DET*WT(II,NGP)*WT(JJ,NGP)*WT(KK,NGP)                      000261
00154    35*          CFT=CNST*DT                                                    000271
00155    36*          RHO=0.0                                                        000273
00156    37*          FJ=0.0                                                         000274
00157    38*          GFOS=0.0                                                       000275
00160    39*          DO 60 J=1,3                                                    000374
00163    40*          H(J)=0.0                                                       000374
00164    41*          V(J)=0.0                                                       000374
00165    42*          DO 60 K=1,3                                                    000377
00170    43*       60 DS(J,K)=0.0                                                    000377
00173    44*          DO 65 I=1,IDIM                                                 000411
00176    45*          DO 65 J=1,IDIM                                                 000411
00201    46*       65 C(I,J)=0.0                                                     000411
00204    47*          DO 70 I=1,NN                                                   000430
00207    48*          FJ=FJ+SF(I)*BF(I,L)                                            000430
00210    49*          RHO=RHO+SF(I)*VEL(I,5)                                         000433
00211    50*          DO 70 J=1,3                                                    000443
00214    51*          V(J)=V(J)+SF(I)*VEL(I,J)                                       000443
00215    52*          DO 70 K=1,3                                                    000450
00220    53*          DS(J,K)=DS(J,K)+W(K,I)*DIS(I,J)                                000450
00221    54*       70 CONTINUE                                                       000476
00225    55*          DO 72 I=1,NN                                                   000476
00230    56*          DO 72 J=1,IDIM                                                 000516
00233    57*          DO 72 K=1,IDIM                                                 000516
00236    58*       72 C(J,K)=C(J,K)+CFRC(I,J,K)*SF(I)                                000516
00242    59*          LEE=LI(L,L)                                                    000536
00243    60*          DO 105 I=1,3                                                   000552
00246    61*          IJ=LI(I,I)                                                     000552
00247    62*          JK=LI(L,I)                                                     000553
00250    63*          H(I)=H(I)-0.50*C(JK,JK)*DS(I,L)                                000555
00251    64*      105 GFOS=GFOS- C(LEE,IJ)*DS(I,I)                                   000563
00253    65*          DO 140 I=1,NN                                                  000572
00256    66*          CNI=CNST*SF(I)                                                 000612
00257    67*          CSL(I)=CSL(I)+CNI*(V(L)+0.50*DT*FJ)+0.50*CFT*GFOS*W(L,I)+0.50* 000614
00257    68*         1CFT*(H(1)*W(1,I)+H(2)*W(2,I)+H(3)*W(3,I))                      000614
00260    69*          DO 140 J=1,NN                                                  000646
00263    70*          DO 135 K=1,3                                                   000646
00266    71*          JK=LI(L,K)                                                     000646
00267    72*      135 CS(I,J)=CS(I,J)+0.50*C(JK,JK)*CFT*W(K,I)*W(K,J)                000647
00271    73*      140 CD(I,J)=CD(I,J)+CNI*SF(J)*RHO                                  000660
00274    74*      150 CONTINUE                                                       000722
```

```
      ELGO                                                                    DATE. 010978

00300      75*                DO 200 I=1,NN                                              000722
00303      76*                CSL(I)=5.0*DT*CSL(I)/6.0                                   000753
00304      77*                DO 200 J=1,NN                                              000760
00307      78*                C1=CD(I,J)+3.0*DT*CS(I,J)/28.0                             000760
00310      79*                C2=CD(I,J)-13.0*DT*CS(I,J)/42.0                            000764
00311      80*                CD(I,J)=C2                                                 000771
00312      81*            200 CS(I,J)=C1                                                 000772
00315      82*                RETURN                                                     001004
00316      83*                END                                                        001163

      END OF COMPILATION:      NO  DIAGNOSTICS.

@HOG,P      FRCR
```

```
 FRCR                                                                    DATE 010978

@FOR,S    FRCR,FRCR
HSA  E3 -01/09/78-23:43:51 (6,)

   SUBROUTINE FRCRIT    ENTRY POINT 001420

   STORAGE USED: CODE(1) 001457; DATA(0) 000156; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CELZ1  000015
    0004   NODEPJ 000003
    0005   MATRL  000005
    0006   ORTHPC 000060
    0007   PROJMT 000047

   EXTERNAL RFFERENCES (BLOCK, NAME)

    0010   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000510 105L      0001   000543 110L      0001   000123 117G      0001   000161 124G      0001   000572 125L
    0001   000162 127G      0001   000212 137G      0001   000605 140L      0001   000217 142G      0001   000220 145G
    0001   000702 145L      0001   000712 150L      0001   000240 154G      0001   000256 160G      0001   000257 163G
    0001   000302 177G      0001   000727 200L      0001   000335 204G      0001   000733 205L      0001   000361 216G
    0001   000174 22L       0001   001012 220L      0001   000434 240G      0001   000443 245G      0001   001044 245L
    0001   001045 246L      0001   000477 265G      0001   001147 276L      0001   000515 277G      0001   001202 280L
    0001   001251 285L      0001   001244 290L      0001   001341 292L      0001   001351 295L      0001   000351 30L
    0001   001366 300L      0001   000520 302G      0001   000620 341G      0001   000755 373G      0001   000774 376G
    0001   001023 411G      0001   001076 433G      0001   001105 440G      0001   000402 45L       0001   001136 457G
    0001   000403 46L       0001   001154 471G      0001   001157 474G      0001   001257 533G      0003   000001 A
    0003   000002 ALPHA     0006 R 000055 CK        0003   000013 DELT      0006   000033 E         0000 R 000046 F
    0000 I 000041 I         0000 I 000045 II        0000   000066 INJP$     0003 I 000011 INNER     0004 I 000002 INTFT
    0003   000000 IORDC     0005 I 000004 ISOP      0004 I 000003 ISOT      0003   000012 ITER      0000 I 000042 J
    0000 I 000043 K         0000 I 000047 KI        0000 I 000050 LEE       0006 I 000022 LI        0000 I 000056 LI1
    0000 I 000057 LI2       0000 I 000060 LJ        0000 I 000061 LK        0000 I 000024 L1        0000 I 000032 L2
    0000 I 000054 M         0000 I 000053 MLTMIN    0000 I 000003 MP        0000 I 000040 N         0003   000006 NBW
    0003   000010 NEPJ      0004   000000 NEPJT     0003   000003 NET       0005 I 000000 NFRAC     0003 I 000014 NFST
    0003   000005 NHBW      0000 I 000055 NI        0000 I 000062 NK        0003 I 000007 NPPJ      0004 I 000001 NPPJT
    0003 I 000004 NPS       0007 R 000044 PCK       0007   000022 PE        0005   000002 PMIU      0006   000044 PNIU
    0007   000033 PPNIU     0007 R 000011 PSLC      0007 R 000000 PSLT      0000 R 000010 RATIO     0000 R 000052 RM
    0000 R 000016 RMIN      0000 R 000051 RN        0000 R 000000 S         0006 R 000011 SLC       0006 R 000000 SLT
    0000 R 000044 SUM       0005   000001 TMIU

00101      1*          SUBROUTINE FRCRIT(NPMT,NPM,STS,INTF,C,CSTF,CSPF,NCNT,NFRC,                    000112
00101      2*         1  NDFRC,NFS,EF,PNF,CKF,KODE)                                                  000112
00101      3*     C                                                                                  000112
00101      4*     C                                                                                  000112
00101      5*     C  TEST THE FAILURE OF COMPOSITES BASED ON CHAMIS' CRITERION                       000112
```

```
00101     6*     C                                                                    000112
00101     7*     C                                                                    000112
00103     8*           DIMENSION STS(NPM,3,3),C(NPMT,NCNT,NCNT),PNF(NFRC,3,3),        000112
00103     9*          1 CSTF(NCNT,NCNT),NDFRC(NFRC),EF(NFRC,3,3),INTF(1)              000112
00103    10*          2  ,KODE(1),NFS(1),CKF(NFRC,3)                                 000112
00103    11*          3  ,CSPF(NCNT,NCNT)                                            000112
00104    12*           DIMENSION S(3),  MP(5),RATIO(6),RMIN(6),L1(6),L2(6)           000112
00105    13*           COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER, 000112
00105    14*          1  ITER,DELT,NFST                                              000112
00106    15*           COMMON /NODEPJ/ NFPJT,NPPJT,INTFT                             000112
00107    16*           COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                      000112
00110    17*           COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3) 000112
00111    18*           COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3) 000112
00112    19*           DATA MP/1,2,3,1,2/                                            000112
00114    20*           DATA LI/1,6,5,6,2,4,5,4,3/                                    000112
00116    21*           DO 22 N=1,NPS                                                 000112
00121    22*           IF (N .GE. NPPJ .AND. N .LE. NPPJT) GO TO 22                  000131
00123    23*           DO 20 I=1,NCNT                                                000162
00126    24*           DO 20 J=1,NCNT                                                000162
00131    25*        20 C(N,I,J)=CSTF(I,J)                                            000162
00134    26*        22 CONTINUE                                                      000175
00136    27*           DO 24 N=NPPJ,NPPJT                                            000175
00141    28*           DO 24 I=1,NCNT                                                000220
00144    29*           DO 24 J=1,NCNT                                                000220
00147    30*        24 C(N,I,J)=CSPF(I,J)                                            000220
00153    31*           DO 26 K=1,INTFT                                               000240
00156    32*           N=INTF(K)                                                     000240
00157    33*           DO 26 I=1,NCNT                                                000241
00162    34*           DO 26 J=1,NCNT                                                000257
00165    35*           IF (KODE(N) .EQ. 0) C(N,I,J)=CSPF(I,J)                        000257
00167    36*        26 CONTINUE                                                      000272
00173    37*           NFRAC=0                                                       000272
00174    38*           IF (ISOT .EQ. 0) GO TO 205                                    000273
00176    39*           DO 200 N=1,NPS                                                000302
00201    40*           IF(N .GE. NPPJ .AND. N .LE. NPPJT) GO TO 200                  000313
00203    41*           DO 30 I=1,INTFT                                               000335
00206    42*           IF (KODE(N) .NE. 0 .AND. N .EQ. INTF(I)) GO TO 30             000343
00210    43*           IF(N .EQ. INTF(I)) GO TO 200                                  000346
00212    44*        30 CONTINUE                                                      000353
00214    45*           SUM=0.0                                                       000353
00215    46*           DO 50 I=1,3                                                   000361
00220    47*           IF(STS(N,I,I) .EQ. 0.0) GO TO 45                              000361
00222    48*           IF(STS(N,I,I) .LT. 0.0) S(I)=SLC(I,I)                         000363
00224    49*           IF(STS(N,I,I) .GT. 0.0) S(I)=SLT(I,I)                         000370
00226    50*           RATIO(I)=STS(N,I,I)/S(I)                                      000375
00227    51*           GO TO 46                                                      000400
00230    52*        45 RATIO(I)=0.0                                                  000402
00231    53*        46 CONTINUE                                                      000403
00232    54*           II=I+3                                                        000403
00233    55*           J=MP(I+1)                                                     000406
00234    56*           K=MP(I+2)                                                     000410
00235    57*        50 RATIO(II)=STS(N,J,K)/SLT(J,K)                                 000422
00237    58*           DO 60 I=1,NCNT                                                000434
00242    59*        60 SUM=SUM+RATIO(I)**2                                           000434
00244    60*           DO 65 I=1,3                                                   000443
00247    61*           II=MP(I+1)                                                    000443
```

```
00250     62*        65 SUM=SUM-CK(I)*RATIO(I)*RATIO(II)                                000445
00252     63*           F=1.0-SUM                                                       000453
00253     64*           IF(F) 100,100,200                                               000455
00256     65*       100 NFRAC=NFRAC+1                                                   000457
00257     66*           NDFRC(NFRAC)=N                                                  000464
00260     67*           IF (INNER .EQ. 1) GO TO 105                                     000466
00262     68*           IF (KODE(N) .NE. 2) KODE(N)=0                                   000470
00264     69*           DO 102 I=1,NFST                                                 000477
00267     70*           KI=NFS(I)                                                       000477
00270     71*           IF (N .EQ. KI) KODE(N)=1                                        000500
00272     72*       102 CONTINUE                                                        000510
00274     73*       105 CONTINUE                                                        000510
00275     74*           II=1                                                            000510
00276     75*           DO 140 I=1,3                                                    000520
00301     76*           DO 140 J=1,3                                                    000520
00304     77*           IF (II .GT. 6) GO TO 140                                        000520
00306     78*           LEE=LI(I,J)                                                     000524
00307     79*           IF(LEE .GT. 1) GO TO 110                                        000527
00311     80*           RMIN(II)=ABS(RATIO(LEE))                                        000532
00312     81*           L1(II)=I                                                        000535
00313     82*           L2(II)=J                                                        000537
00314     83*           GO TO 140                                                       000541
00315     84*       110 CONTINUE                                                        000543
00316     85*           RN=RMIN(II)+1.E-4                                               000544
00317     86*           RM=RMIN(II)-1.E-4                                               000547
00320     87*           IF(ABS(RATIO(LEE)).GT. RN) GO TO 140                            000552
00322     88*           IF(ABS(RATIO(LEE)).GE. RM) GO TO 125                            000556
00324     89*           RMIN(II)=ABS(RATIO(LEE))                                        000562
00325     90*           L1(II)=I                                                        000564
00326     91*           L2(II)=J                                                        000566
00327     92*           GO TO 140                                                       000570
00330     93*       125 II=II+1                                                         000572
00331     94*           RMIN(II)=ABS(RATIO(LEE))                                        000576
00332     95*           L1(II)=I                                                        000600
00333     96*           L2(II)=J                                                        000602
00334     97*       140 CONTINUE                                                        000612
00337     98*           MLTMIN=II                                                       000612
00340     99*           DO 180 M=1,MLTMIN                                               000614
00343    100*           NI=NFRAC                                                        000620
00344    101*           LI1=L1(M)                                                       000621
00345    102*           LI2=L2(M)                                                       000623
00346    103*           EF(NI,LI1,LI2)=0.0                                              000633
00347    104*           IF(LI1 .NE. LI2) GO TO 145                                      000634
00351    105*           LJ=MP(LI1+1)                                                    000637
00352    106*           LK=MP(LI1+2)                                                    000655
00353    107*           PNF(NI,LI1,LJ)=0.0                                              000672
00354    108*           PNF(NI,LI1,LK)=0.0                                              000673
00355    109*           CKF(NI,LI1)=0.0                                                 000675
00356    110*           CKF(NI,LK) =0.0                                                 000676
00357    111*           GO TO 150                                                       000700
00360    112*       145 EF(NI,LI2,LI1)=0.0                                              000702
00361    113*       150 CONTINUE                                                        000712
00362    114*           C(N,LI1,LI2)=0.0                                                000723
00363    115*       180 CONTINUE                                                        000733
00365    116*       200 CONTINUE                                                        000733
00367    117*       205 CONTINUE                                                        000733
```

```
00370   118*          IF (ISOP .EQ. 0) RETURN                                  000733
00372   119*          DO 300 N=1,NPPJT                                         000755
00375   120*          DO 210 I=1,INTFT                                         000774
00400   121*          IF (KODE(N) .EQ. 0 .AND. N .EQ. INTF(I)) GO TO 220       000774
00402   122*          IF (N .GE. NPPJ ) GO TO 220                              001003
00404   123*      210 CONTINUE                                                 001010
00406   124*          GO TO 300                                                001010
00407   125*      220 SUM=0.0                                                  001012
00410   126*          DO 250 I=1,3                                             001012
00413   127*          IF(STS(N,I,I) .EQ. 0.0) GO TO 245                        001023
00415   128*          IF(STS(N,I,I) .LT. 0.0) S(I)=PSLC(I,I)                   001025
00417   129*          IF(STS(N,I,I) .GT. 0.0) S(I)=PSLT(I,I)                   001032
00421   130*          RATIO(I)=STS(N,I,I)/S(I)                                 001037
00422   131*          GO TO 246                                                001042
00423   132*      245 RATIO(I)=0.0                                             001044
00424   133*      246 CONTINUE                                                 001045
00425   134*          II=I+3                                                   001045
00426   135*          J=MP(I+1)                                                001050
00427   136*          K=MP(I+2)                                                001052
00430   137*      250 RATIO(II)=STS(N,J,K)/PSLI(J,K)                           001064
00432   138*          DO 260 I=1,NCNT                                          001076
00435   139*      260 SUM=SUM+RATIO(I)**2                                      001076
00437   140*          DO 265 I=1,3                                             001105
00442   141*          II=MP(I+1)                                               001105
00443   142*      265 SUM=SUM-PCK(I)*RATIO(I)*RATIO(II)                        001107
00445   143*          F=1.0-SUM                                                001115
00446   144*          IF(F) 270,270,300                                        001117
00451   145*      270 NFRAC=NFRAC+1                                            001121
00452   146*          NOFRC(NFRAC)=N                                           001126
00453   147*          IF (INNER .EQ. 1) GO TO 276                              001130
00455   148*          KODE(N)=0                                                001132
00456   149*          DO 275 K=1,NFST                                          001136
00461   150*          NK=NFS(K)                                                001136
00462   151*          IF (N .EQ. NK) KODE(N)=1                                 001137
00464   152*      275 CONTINUE                                                 001147
00466   153*      276 CONTINUE                                                 001147
00467   154*          II=1                                                     001147
00470   155*          DO 290 I=1,3                                             001157
00473   156*          DO 290 J=1,3                                             001157
00476   157*          IF (II .GT. 6) GO TO 290                                 001157
00500   158*          LEE=LI(I,J)                                              001163
00501   159*          IF(LEE .GT. 1) GO TO 280                                 001166
00503   160*          RMIN(II)=ABS(RATIO(LEE))                                 001171
00504   161*          L1(II)=I                                                 001174
00505   162*          L2(II)=J                                                 001176
00506   163*          GO TO 290                                                001200
00507   164*      280 CONTINUE                                                 001202
00510   165*          RN=RMIN(II)+1.E-4                                        001203
00511   166*          RM=RMIN(II)-1.E-4                                        001206
00512   167*          IF(ABS(RATIO(LEE)).GT. RN) GO TO 290                     001211
00514   168*          IF(ABS(RATIO(LEE)).GE. RM) GO TO 285                     001215
00516   169*          RMIN(II)=ABS(RATIO(LEE))                                 001221
00517   170*          L1(II)=I                                                 001223
00520   171*          L2(II)=J                                                 001225
00521   172*          GO TO 290                                                001227
00522   173*      285 II=II+1                                                  001231
```

```
00523    174*            RMIN(II)=ABS(RATIO(LEE))                                001235
00524    175*            L1(II)=I                                                001237
00525    176*            L2(II)=J                                                001241
00526    177*      290   CONTINUE                                                001251
00531    178*            MLTMIN=II                                               001251
00532    179*            DO 298 M=1,MLTMIN                                       001253
00535    180*            NI=NFRAC                                                001257
00536    181*            LI1=L1(M)                                               001260
00537    182*            LI2=L2(M)                                               001262
00540    183*            EF(NI,LI1,LI2)=0.0                                      001272
00541    184*            IF(LI1 .NE. LI2) GO TO 292                              001273
00543    185*            LJ=MP(LI1+1)                                            001276
00544    186*            LK=MP(LI1+2)                                            001314
00545    187*            PNF(NI,LI1,LJ)=0.0                                      001331
00546    188*            PNF(NI,LI1,LK)=0.0                                      001332
00547    189*            CKF(NI,LI1)=0.0                                         001334
00550    190*            CKF(NI,LK) =0.0                                         001335
00551    191*            GO TO 295                                               001337
00552    192*        292 EF(NI,LI2,LI1)=0.0                                      001341
00553    193*        295 CONTINUE                                                001351
00554    194*            C(N,LI1,LI2)=0.0                                        001362
00555    195*        298 CONTINUE                                                001371
00557    196*        300 CONTINUE                                                001371
00561    197*            RETURN                                                  001371
00562    198*            END                                                     001456

      END OF COMPILATION;       NO  DIAGNOSTICS.

@HDG,P      GEOM
```

```
@FOR,S    GEOM,GEOM
HSA  E3 -01/09/78-23:43:55 (3,)

   SUBROUTINE GEOMTR    ENTRY POINT 000376

   STORAGE USED: CODE(1) 000412; DATA(0) 000125; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   SRFACE 000077
    0004   GEOM1  000524
    0005   CELZ1  000015
    0006   LAGRGN 000005
    0007   CELZ2  000007
    0010   NODEPJ 000003

   EXTERNAL REFERENCES (BLOCK, NAME)

    0011   NPRT$
    0012   NIO2$
    0013   NRDC$
    0014   NIO1$
    0015   NIO3$
    0016   NEPR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000026 1000F    0000   000027 1110F    0000   000035 1150F    0000   000036 1200F    0001   000024 127G
    0000   000043 1300F    0000   000046 1305F    0001   000031 133G     0000   000051 1350F    0000   000060 1400F
    0001   000042 142G     0000   000066 1420F    0001   000076 153G     0001   000103 157G     0001   000114 166G
    0001   000150 177G     0001   000161 205G     0001   000176 215G     0001   000130 215L     0001   000216 224G
    0001   000217 227G     0001   000220 232G     0001   000250 244G     0001   000251 247G     0001   000252 252G
    0001   000310 270G     0001   000325 300G     0001   000336 306G     0001   000353 316G     0005   000001 A
    0005   000002 ALPHA    0005   000013 DELT     0007   000004 DIMX     0007   000005 DIMY     0007   000006 DIMZ
    0000 R 000000 G        0000 I 000021 I        0000   000100 INJP$    0005   000011 INNER    0003 I 000065 INTF
    0010 I 000002 INTFT    0005   000000 IORDC    0006   000002 IORDL    0005   000012 ITER     0000 I 000023 J
    0000 I 000024 K        0004 I 000000 KODE     0000 I 000020 N        0005   000006 NBW      0006   000003 NBWT
    0007   000003 NELG     0006 I 000001 NELT     0005   000010 NEPJ     0010   000000 NEPJT    0005 I 000003 NET
    0007   000001 NETR     0003 I 000000 NFS      0005 I 000014 NFST     0005 I 000005 NHBW     0006 I 000004 NHBWT
    0000 I 000022 NN       0004 I 000114 NOD      0007   000002 NPLG     0005   000007 NPPJ     0010   000001 NPPJT
    0005   000004 NPS      0006 I 000000 NPST     0007   000000 NPTR     0000 I 000025 NW

00101      1*          SUBROUTINE GEOMTR(X)                                                    000013
00101      2*      C                                                                           000013
00101      3*      C                                                                           000013
00101      4*      C  GENERATE GEOMETRIC INFORMATION FROM INPUT DATA                           000013
00101      5*      C                                                                           000013
00101      6*      C                                                                           000013
00103      7*          INCLUDE PARAM1,LIST                                                     000013
```

```
00103     7*     PARAM1 PROC                                                                  000013
00103     7*     C                                                                            000013
00104     7*            PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                           000013
00105     7*            PARAMETER NPELG= 8,NPE= 8                                             000013
00106     7*            PARAMETER NDFRS=53,NDINT=10                                           000013
00106     7*     C                                                                            000013
00107     7*            PARAMETER NRMAX=2000,NCMAX=2                                          000013
00110     7*            PARAMETER NGPLG=2,NGP=2                                               000013
00111     7*            PARAMETER NCNT=6, NFRC=NPM                                            000013
00112     7*            PARAMETER IU=5,NDPJT= 10,NTSTEP=200                                   000013
00113     7*            PARAMETER NBV=NPM,NBS=NPM,N3D=NPM                                     000013
00113     7*      END                                                                         000013
00114     8*            INCLUDE GENMSH,LIST                                                   000013
00114     9*     C                                                                            000013
00114    10*     C                                                                            000013
00114    10*     GENMSH PROC                                                                  000013
00115    10*            COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)                                 000013
00116    10*            COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)                             000013
00117    10*            COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,        000013
00117    10*           1  ITER,DELT,NFST                                                      000013
00120    10*            COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                            000013
00121    10*            COMMON /CELZ2/ NPTR,NETR,NPLG,NELG,DIMX,DIMY,DIMZ                     000013
00122    10*            COMMON /NODEPJ/ NEPJT,NPPJT,INTFT                                     000013
00123    10*            DIMENSION X(NPMT,3),G(16)                                             000013
00123    10*      END                                                                         000013
00124    11*            PRINT 1110                                                            000013
00126    12*            DO 210 N=1,NET                                                        000024
00131    13*            READ 1000, (NOD(N,I),I=1,NPE)                                         000024
00137    14*        210 PRINT 1150, N,(NOD(N,I),I=1,NPE)                                      000034
00147    15*            NN=NELT+1                                                             000055
00150    16*            IF (NN .GT. NELT) GO TO 215                                           000060
00152    17*            DO 211 N=NN,NELT                                                      000063
00155    18*            READ 1000, (NOD(N,I),I=1,NPELG)                                       000076
00163    19*        211 PRINT 1150, N,(NOD(N,I),I=1,NPELG)                                    000106
00173    20*        215 CONTINUE                                                              000130
00174    21*            PRINT 1200                                                            000130
00176    22*            DO 220 N=1,NPST                                                       000133
00201    23*            READ 1300, G,KODE(N),(X(N,I),I=1,3)                                   000150
00211    24*        220 PRINT 1305, G,KODE(N),(X(N,I),I=1,3)                                  000164
00222    25*            NHBW=0                                                                000211
00223    26*            DO 250 I=1,NET                                                        000220
00226    27*            DO 250 J=1,NPE                                                        000220
00231    28*            DO 250 K=1,NPE                                                        000220
00234    29*            NW=IABS(NOD(I,J)-NOD(I,K))+1                                          000220
00235    30*        250 IF (NHBW .LT. NW) NHBW=NW                                             000224
00242    31*            NHBWT=NHBW                                                            000242
00243    32*            DO 252 I=NN,NELT                                                      000252
00246    33*            DO 252 J=1,NPELG                                                      000252
00251    34*            DO 252 K=1,NPELG                                                      000252
00254    35*            NW=IABS(NOD(I,J)-NOD(I,K))+1                                          000252
00255    36*        252 IF (NHBWT .LT. NW) NHBWT=NW                                           000256
00262    37*            PRINT 1350, NHBW,NHBWT                                                000274
00266    38*            READ 1000, (NFS(I),I=1,NFST)                                          000302
00274    39*            PRINT 1400                                                            000313
00276    40*            PRINT 1000, (NFS(I),I=1,NFST)                                         000317
00304    41*            READ 1000, (INTF(I),I=1,INTFT)                                        000330
```

```
00312              PRINT 1420                                                000341
00314              PRINT 1000, (INTF(I),I=1,INTFT)                            000345
00322         1000 FORMAT (20I4)                                              000356
00323         1110 FORMAT (' ELEMENT NO. AND NODE NUMBERS'/)                  000356
00324         1150 FORMAT(33I4)                                               000356
00325         1200 FORMAT (//' GRID COORDINATES'/)                            000356
00326         1300 FORMAT(16A1,I8,3F8.3)                                      000356
00327         1305 FORMAT (1X,16A1,I8,3F8.3)                                  000356
00330         1350 FORMAT (/// 6X,'NHBW =',I5/5X,'NHBWT =',I5///)             000356
00331         1400 FORMAT (//' NODES ON FREE SURFACE'/)                       000356
00332         1420 FORMAT (//' NODES ON IMPACT SURFACE'/)                     000356
00333              RETURN                                                     000356
00334              END                                                        000411

      END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      INTVEC
```

```
@FOR,S    INTVEC,INTVEC
HSA  E3 -01/09/78-23:43:58 (0,)

     SUBROUTINE INTVEC    ENTRY POINT 000705

     STORAGE USED: CODE(1) 000731; DATA(0) 000166; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   READF
      0004   NWDU$
      0005   NIO2$
      0006   NSTOP$
      0007   NERR4$
      0010   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000022 10L      0001   000377 100L     0001   000403 110L     0001   000426 120L     0001   000451 130L
      0001   000061 131G     0001   000479 140L     0001   000517 150L     0001   000520 160L     0001   000260 164G
      0001   000530 170L     0001   000551 190L     0001   000554 200L     0001   000332 207G     0001   000566 210L
      0001   000575 220L     0001   000365 224G     0001   000604 225L     0001   000614 230L     0001   000402 234G
      0001   000623 240L     0001   000633 250L     0001   000643 260L     0001   000652 270L     0001   000066 30L
      0000   000026 300F     0001   000547 306G     0000   000033 310F     0000   000042 320F     0000   000050 325F
      0000   000061 330F     0000   000067 340F     0000   000076 350F     0000   000110 360F     0000   000120 370F
      0001   000661 400L     0001   000263 45L      0001   000266 50L      0001   000303 55L      0001   000337 80L
      0001   000350 90L      0000 D 000132 D        0000 I 000014 I        0000 I 000011 IERR     0000 I 000010 IFL
      0000 I 000024 II       0000   000140 INJP$    0000 I 000007 IPAD     0000 I 000023 IS       0000 I 000012 IWD
      0000 I 000132 IX       0000 I 000134 IY       0000 I 000013 K        0000 I 000020 KK       0000 I 000017 L
      0000 I 000021 LAST     0000 I 000025 M        0000 I 000015 MTYPE    0000 I 000016 MWD      0000 I 000005 NOUT
      0000 I 000022 NS       0000 I 000006 NULL     0000 I 000001 PTYPE    0000 R 000134 R        0000 I 000000 WD

00101      1*          SUBROUTINE INTVEC(*,A,NWD,N,ITYPE)                                          000006
00103      2*          INTEGER A(1),WD,IX(2),PTYPE(4),IY                                           000006
00103      3*     C  *****                    *****                                               000006
00103      4*     C        INTERPRETER VECTOR                                                     000006
00103      5*     C  *****                    *****                                               000006
00104      6*          DOUBLE PRECISION  D                                                        000006
00105      7*          EQUIVALENCE (D,IX),(R,IY)                                                  000006
00106      8*          DATA PTYPE /1,2,2,4/                                                       000006
00110      9*          DATA NOUT/6/                                                               000006
00112     10*          DATA NULL,IPAD/4HNULL,4HPAD /                                              000006
00112     11*     C                                                                               000006
00112     12*     C                                                                               000006
00115     13*          WD=NWD                                                                     000006
00116     14*          CALL READF($200,$10,A,WD,IFL,IERR)                                         000010
00117     15*          GO TO 210                                                                  000020
00120     16*       10 IF(ITYPE.LE.0.OR.ITYPE.GE.5) GO TO 220                                     000027
00122     17*          IWD=PTYPE(ITYPE)                                                           000035
```

```
00123    18*            IF(IWD*N+IWD.GE.NWD) GO TO 225                                  000041
00125    19*            IF(A(2).NE.NULL) GO TO 30                                       000046
00127    20*            K=N*IWD                                                         000051
00130    21*            DO 20 I=1,K                                                     000054
00133    22*         20 A(I)=0                                                          000061
00135    23*            RETURN                                                          000062
00136    24*      30    MTYPE=A(1)                                                      000066
00137    25*            IF(MTYPE.LE.0.OR.MTYPE.GE.5) GO TO 230                          000067
00141    26*            MWD=PTYPE(MTYPE)                                                000103
00142    27*            IF(MTYPE.LE.2.AND.ITYPE.GE.3) GO TO 240                         000106
00144    28*            IF(ITYPE.LE.2.AND.MTYPE.GE.3) GO TO 240                         000130
00144    29*      C **** TO SET TYPE CHANGE  SINGLE TO DOUBLE OR DOUBLE TO SINGLE        000130
00146    30*            IF(MTYPE.EQ.1.AND.ITYPE.EQ.2) ASSIGN 110 TO L                   000150
00150    31*            IF(MTYPE.EQ.2.AND.ITYPE.EQ.1) ASSIGN 120 TO L                   000166
00152    32*            IF(MTYPE.EQ.3.AND.ITYPE.EQ.4) ASSIGN 130 TO L                   000204
00154    33*            IF(MTYPE.EQ.4.AND.ITYPE.EQ.3) ASSIGN 140 TO L                   000222
00156    34*            KK=0                                                            000240
00157    35*            LAST=0                                                          000241
00160    36*            K=NWD-WD                                                        000242
00161    37*            IF(K.EQ.0) GO TO 45                                             000245
00163    38*            DO 40 I=WD,1,-1                                                 000247
00166    39*         40 A(K+I)=A(I)                                                     000260
00170    40*         45 K=K+2                                                           000263
00170    41*      C *** TO CATCH INDEX OF STRING HEADER                                 000263
00170    42*      C                                                                     000263
00171    43*         50 IF(K.GT.NWD) GO TO 170                                          000266
00173    44*            IF(A(K).NE.IPAD) GO TO 55                                       000271
00175    45*            K=K+1                                                           000276
00176    46*            GO TO 50                                                        000301
00177    47*         55 NS=FLD(0,20,A(K))                                               000303
00200    48*            IS=FLD(20,16,A(K))                                              000307
00200    49*      C                                                                     000307
00201    50*            II=IS-LAST-1                                                    000313
00202    51*            IF(II) >50,80,60                                                000316
00205    52*         60 M=II*IWD                                                        000320
00206    53*            DO 70 I=1,M                                                     000325
00211    54*         70 A(KK+I)=0                                                       000332
00213    55*            KK=KK+M                                                         000333
00214    56*         80 IF(NS) 260,260,85                                               000337
00217    57*         85 M=IWD*NS                                                        000341
00220    58*            IF(MTYPE.EQ.ITYPE) GO TO 90                                     000343
00222    59*            GO TO 100                                                       000346
00223    60*         90 DO 95 I=1,M                                                     000350
00226    61*         95 A(KK+I)=A(K+I)                                                  000365
00230    62*            KK=KK+M                                                         000367
00231    63*            K=K+M                                                           000372
00232    64*            GO TO 160                                                       000375
00233    65*        100 DO 150 I=1,NS                                                   000377
00236    66*            GO TO L,(110,120,130,140)                                       000402
00237    67*        110 K=K+1                                                           000403
00240    68*            IY=A(K)                                                         000411
00241    69*            D=R                                                             000413
00242    70*            A(KK+1)=IX(1)                                                   000415
00243    71*            A(KK+2)=IX(2)                                                   000417
00244    72*            KK=KK+2                                                         000421
00245    73*            GO TO 150                                                       000424
```

```
00246     74*      120  K=K+2                                                    000426
00247     75*           IX(1)=A(K-1)                                            000432
00250     76*           IX(2)=A(K)                                              000434
00251     77*           R=0                                                     000436
00252     78*           KK=KK+1                                                 000440
00253     79*           A(KK)=IY                                                000445
00254     80*           GO TO 150                                               000447
00255     81*      130  K=K+1                                                   000451
00256     82*           IY=A(K)                                                 000457
00257     83*           D=R                                                     000461
00260     84*           A(KK+1)=IX(1)                                           000463
00261     85*           A(KK+2)=IX(2)                                           000465
00262     86*           KK=KK+2                                                 000467
00263     87*           GO TO 110                                               000472
00264     88*      140  K=K+2                                                   000474
00265     89*           IX(1)=A(K-1)                                            000500
00266     90*           IX(2)=A(K)                                              000502
00267     91*           R=D                                                     000504
00270     92*           KK=KK+1                                                 000506
00271     93*           A(KK)=IY                                                000513
00272     94*           GO TO 120                                               000515
00273     95*      150  CONTINUE                                               000520
00275     96*      160  K=K+1                                                   000520
00276     97*           LAST=KK/IWD                                             000522
00277     98*           GO TO 50                                                000526
00300     99*      170  M=KK-IWD                                               000530
00301    100*           M=(N-M)*IWD                                             000532
00302    101*           IF(M) 270,190,175                                       000535
00305    102*      175  DO 180 I=1,M                                           000542
00310    103*      180  A(KK+I)=0                                              000547
00312    104*      190  CONTINUE                                               000551
00313    105*           RETURN                                                  000551
00313    106*      C *****              *****                                   000551
00313    107*      C         ERROR OCCURED                                      000551
00313    108*      C *****              *****                                   000551
00313    109*      C                                                            000551
00314    110*      200  WRITE(NOUT,300) IERR                                   000554
00317    111*           RETURN 1                                                000561
00320    112*      210  WRITE(NOUT,310) NWD                                    000566
00323    113*           GO TO 400                                               000573
00324    114*      220  WRITE(NOUT,320) ITYPE                                  000575
00327    115*           GO TO 400                                               000602
00330    116*      225  WRITE(NOUT,325) N,NWD                                  000604
00334    117*           GO TO 400                                               000612
00335    118*      230  WRITE(NOUT,330) MTYPE                                  000614
00340    119*           GO TO 400                                               000621
00341    120*      240  WRITE(NOUT,340) ITYPE,MTYPE                            000623
00345    121*           GO TO 400                                               000631
00346    122*      250  WRITE(NOUT,350) IS,LAST                                000633
00352    123*           GO TO 400                                               000641
00353    124*      260  WRITE(NOUT,360) NS                                     000643
00356    125*           GO TO 400                                               000650
00357    126*      270  WRITE(NOUT,370) N,KK                                   000652
00363    127*      400  STOP                                                    000661
00364    128*      300  FORMAT('0 INTVEC ENCOUNTERED EOF')                     000730
00365    129*      310  FORMAT('0 INTVEC INSUFFICIENT CORE   NWD=',I5)         000730
```

```
      INTVEC                                                                    DATE 010978

00366   130*      320  FORMAT('0 INTVEC   ITYPE  ABNORMAL ',I5)                          000730
00367   131*      325  FORMAT('0 INTVEC   AREA LITTLE  N=',I5,'     AREA=',I5)         000730
00370   132*      330  FORMAT('0 INTVEC   MTYPE  ABNORMAL ',I5)                          000730
00371   133*      340  FORMAT('0 INTVEC   ITYPE=',I3,'   MTYP=',I3)                     000730
00372   134*      350  FORMAT('0 INTVEC  MATRIX INDEX ABNORMAL  IS=',I5,'   LAST',I5)   000730
00373   135*      360  FORMAT('0 INTVEC  MATRIX INDEX ABNORMAL  NS=',I5)               000730
00374   136*      370  FORMAT('0 INTVEC  MATRIX SIZE DIFFERS   N=',I5,'     KK=',I5)   000730
00375   137*           END                                                            000730

          END OF COMPILATION:        NO  DIAGNOSTICS.

   @HDG,P      INVD
```

```
INVO                                                                DATE 010978

@FOR,S   INVO,INVO
HSA  E3 -01/09/78-23:44:01 (0,)

     SUBROUTINE INVOET    ENTRY POINT 000136

     STORAGE USED: CODE(1) 000157; DATA(0) 000016; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0000 R 000000 C1        0000 R 000001 C2        0000 R 000002 C3        0000   000003 INJP$


00101      1*            SUBROUTINE INVOET(A,B,DET)                          INV0010   000001
00103      2*            DIMENSION A(3,3),B(3,3)                             INV0020   000001
00104      3*            G(Z1,Z2,Z3,Z4)=Z1*Z2-Z3*Z4                          INV0030   000001
00105      4*            F(Z1,Z2,Z3,Z4)=G(Z1,Z2,Z3,Z4)/DET                   INV0040   000001
00106      5*            C1=G(A(2,2),A(3,3),A(2,3),A(3,2))                   INV0050   000001
00107      6*            C2=G(A(2,3),A(3,1),A(2,1),A(3,3))                   INV0060   000007
00110      7*            C3=G(A(2,1),A(3,2),A(2,2),A(3,1))                   INV0070   000015
00111      8*            DET=A(1,1)*C1+A(1,2)*C2+A(1,3)*C3                   INV0080   000023
00112      9*            B(1,1)=F(A(2,2),A(3,3),A(3,2),A(2,3))               INV0090   000032
00113     10*            B(1,2)=-F(A(1,2),A(3,3),A(1,3),A(3,2))              INV0100   000035
00114     11*            B(1,3)=F(A(1,2),A(2,3),A(1,3),A(2,2))               INV0110   000044
00115     12*            B(2,1)=-F(A(2,1),A(3,3),A(2,3),A(3,1))              INV0120   000053
00116     13*            B(2,2)=F(A(1,1),A(3,3),A(3,1),A(1,3))               INV0130   000060
00117     14*            B(2,3)=-F(A(1,1),A(2,3),A(1,3),A(2,1))              INV0140   000067
00120     15*            B(3,1)=F(A(2,1),A(3,2),A(3,1),A(2,2))               INV0150   000076
00121     16*            B(3,2)=-F(A(1,1),A(3,2),A(1,2),A(3,1))              INV0160   000105
00122     17*            B(3,3)=F(A(1,1),A(2,2),A(2,1),A(1,2))               INV0170   000114
00123     18*            RETURN                                              INV0180   000123
00124     19*            END                                                 INV0190   000156

        END OF COMPILATION:        NO DIAGNOSTICS.

@HDG,P      MATM
```

```
MATM                                                                    DATE 010978

@FOR,S    MATM,MATM
HSA  E3 -01/09/78-23:44:03 (0,)

   SUBROUTINE MATMUL    ENTRY POINT 000121

     STORAGE USED: CODE(1) 000141; DATA(0) 000042; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

       0003   NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000052 105G      0001  000055 110G      0001  000057 114G      0000 I 000000 I       0000  000004 INJP$
      0000 I 000001 J         0000 I 000002 K
```

A-80

```
00101     1*          SUBROUTINE MATMUL(A,B,C,M,N,L)                    MUL0010   000055
00103     2*          DIMENSION A(M,N),B(N,L),C(M,L)                    MUL0020   000055
00104     3*          DO 10 I=1,M                                       MUL0030   000055
00107     4*          DO 10 J=1,L                                       MUL0040   000055
00112     5*          C(I,J)=0.                                         MUL0050   000055
00113     6*          DO 10 K=1,N                                       MUL0060   000057
00116     7*       10 C(I,J)=C(I,J)+A(I,K)*B(K,J)                       MUL0070   000057
00122     8*          RETURN                                            MUL0080   000075
00123     9*          END                                               MUL0090   000140

        END OF COMPILATION:       NO  DIAGNOSTICS.

@HDG,P      MATRX
```

```
@FOR,S    MATRX,MATRX
HSA  E3 -01/09/78-23:44:04 (1,)


     SUBROUTINE OPNNAS    ENTRY POINT 001107
                MATRX     ENTRY POINT 001120
                COLUMN    ENTRY POINT 001135
                ELEMNT    ENTRY POINT 001142
                WEOF      ENTRY POINT 001151

     STORAGE USED: CODE(1) 001154; DATA(0) 000143; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

          0003   NWDU$
          0004   NIO2$
          0005   NIO1$
          0006   NIO3$
          0007   NSTOP$
          0010   NERR6$
          0011   NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

          0000   000047 1001F    0001   000377 105L    0001   000413 107L    0001   000413 110L    0001   000434 120L
          0001   000451 130L     0001   000033 135G    0001   000475 150L    0000   000054 151F    0001   000076 156G
          0001   000564 195L     0001   000571 200L    0001   000144 205G    0001   000621 205L    0001   000640 210L
          0001   000707 240L     0001   000712 250L    0000   000066 251F    0001   000275 257G    0001   000741 310L
          0001   000752 320L     0001   000771 330L    0001   001006 340L    0001   000551 375G    0001   000656 424G
          0001   001020 5000L    0001   001044 520G    0001   001052 6000L   0000   000077 6001F   0001   000122 8L
          0000   000046 8000F    0001   000364 90L     0000 I 000033 I       0000 I 000001 IDHDR   0000 I 000020 IEOF
          0000 I 000021 IEOR     0000 I 000034 IFT     0000 I 000045 II      0000 I 000041 IKL     0000   000111 INJP$
          0000 I 000040 IRET     0000 I 000023 ITRL    0000 I 000032 IWD     0000 I 000035 JJ      0000 I 000022 KEY
          0000 I 000037 KSV      0000 I 000044 L       0000 I 000036 LAST    0000 I 000043 M       0000 I 000000 MAX
          0000 I 000042 MM       0000 I 000017 NN      0000 I 000014 NOUT    0000 I 000015 NULL    0000 I 000016 NUNIT
          0000 I 000013 P        0000 I 000012 PAD     0000 I 000011 SS      0000 I 000010 SYSBUF
```

```
00101      1*          SUBROUTINE OPNNAS(NNNNN,W,NWD,IDATE,LABEL,ILB)                    8/76-RS.   000000
00101      2*      C   NNNNN     OUTPUT NUMBER                                           8/76-RS.   000000
00101      3*      C   W(1) - W(NWD)   WORKING BUFFER                                    8/76-RS.   000000
00101      4*      C   IDATE(1) - IDATE(3)   DATE                                        8/76-RS.   000000
00101      5*      C   LABEL(1) - LABEL(2)   LABEL                                       8/76-RS.   000000
00101      6*      C   NAME(1) - NAME(2) MATRIX NAME, THIS NAME IS USED FOR DATABLOCK NAME 8/76-RS.   000000
00101      7*      C   NCOL  - - - - - - NUMBER OF COLUMNS                               8/76-RS.   000000
00101      8*      C   NROW  - - - - - - NUMBER OF ROWS                                  8/76-RS.   000000
00101      9*      C   NFORM - - - - - - FORM OF MATRIX  1,SQUARE  2,RECTANGULAR  3,DIAGON 8/76-RS.   000000
00101     10*      C                      4,LOWER TRIANG  5,UPPER TRIANG  6,SYMMETRIC    8/76-RS.   000000
00101     11*      C                      7,ROW VECTOR  8,IDENTITY                       8/76-RS.   000000
00101     12*      C   NTYPE  - - - - - TYPE  1,SINGLE  2,DOUBLE  3,COMPLEX SINGLE       8/76-RS.   000000
00101     13*      C                      4,COMPLEX DOUBLE                               8/76-RS.   000000
```

```
00103    14*          INTEGER W(1),IDHOR(7),IDATE(3),LABEL(2)                    8/76-RS.   000000
00104    15*          INTEGER SYSBUF,SS,PAD,P                                    8/76-RS.   000000
00105    16*          DATA IDHDR/4HNAST,4HRAN ,4HFORT,4H TAP,4HE ID,4H COD,4HE ~ /  8/76-RS.   000000
00107    17*          DATA NOUT/6/                                               8/76-RS.   000000
00111    18*          DATA SYSBUF/871/                                           8/76-RS.   000000
00113    19*          DATA PAD/4HPAD /                                           8/76-RS.   000000
00115    20*          DATA NULL/4HNULL/                                          8/76-RS.   000000
00117    21*    8000  FORMAT(22A6)                                                         000000
00120    22*          NUNIT=NNNNN                                                8/76-RS.   000000
00121    23*          NN=NWD-8                                                   8/76-RS.   000001
00122    24*          SS=SYSBUF-5                                                8/76-RS.   000004
00123    25*          IEOF=0                                                     8/76-RS.   000007
00124    26*          IEOR=0                                                     8/76-RS.   000010
00125    27*          IF(ILB.EQ.0) GO TO 8                                       8/76-RS.   000011
00127    28*          KEY=3                                                      8/76-RS.   000013
00130    29*          WRITE(NUNIT,8000) KEY                                         8/7     000015
00133    30*          WRITE(NUNIT,8000) IDATE                                       8/7     000023
00141    31*          KEY=7                                                      8/76-RS.   000036
00142    32*          WRITE(NUNIT,8000) KEY                                         8/7     000040
00145    33*          WRITE(NUNIT,8000) IDHDR                                       8/7     000046
00150    34*          KEY=2                                                      8/76-RS.   000056
00151    35*          WRITE(NUNIT,8000) KEY                                         8/7     000060
00154    36*          WRITE(NUNIT,8000) LABEL                                       8/7     000066
00162    37*          IEOR=IEOR-1                                                8/76-RS.   000101
00163    38*          WRITE(NUNIT,8000) IEOR                                        8/7     000104
00166    39*          WRITE(NUNIT,8000) IEOF                                        8/7     000112
00171    40*          IEOR=0                                                     8/76-RS.   000120
00172    41*     8    CONTINUE                                                   8/76-RS.   000122
00173    42*          RETURN                                                     8/76-RS.   000122
00173    43*     C                                                               8/76-RS.   000122
00173    44*     C                                                               8/76-RS.   000122
00174    45*          ENTRY MATRX(NAME,NCOL,NROW,NFORM,NTYPE)                    8/76-RS.   000124
00176    46*          INTEGER NAME(2),ITRL(7)                                    8/76-RS.   000124
00177    47*          KEY=2                                                      8/76-RS.   000124
00200    48*          WRITE(NUNIT,8000) KEY                                         8/7     000126
00203    49*          WRITE(NUNIT,8000) NAME                                        8/7     000134
00211    50*          IEOR=IEOR-1                                                8/76-RS.   000147
00212    51*          WRITE(NUNIT,8000) IEOR                                        8/7     000152
00215    52*          KEY=7                                                      8/76-RS.   000160
00216    53*          WRITE(NUNIT,8000) KEY                                         8/7     000162
00221    54*          ITRL(1)=101                                                8/76-RS.   000170
00222    55*          ITRL(2)=NCOL                                               8/76-RS.   000172
00223    56*          ITRL(3)=NROW                                               8/76-RS.   000174
00224    57*          ITRL(4)=NFORM                                              8/76-RS.   000176
00225    58*          ITRL(5)=NTYPE                                              8/76-RS.   000200
00226    59*          IWD=2                                                      8/76-RS.   000202
00227    60*          IF(ITRL(5).EQ.1) IWD=1                                     8/76-RS.   000204
00231    61*          IF(ITRL(5).EQ.4) IWD=4                                     8/76-RS.   000211
00233    62*          MAX=IWD*NROW+2                                             8/76-RS.   000215
00234    63*          ITRL(6)=MAX                                                8/76-RS.   000222
00235    64*          ITRL(7)=0                                                  8/76-RS.   000223
00236    65*          WRITE(NUNIT,8000) ITRL                                        8/7     000224
00241    66*          WRITE(NOUT,1001) (ITRL(I),I=1,7)                           8/76-RS.   000234
00244    67*  1001    FORMAT('0MATRIX TRAILER ',7I8)                             8/76-RS.   000244
00245    69*          IEOR=IEOR-1                                                8/76-RS.   000244
00246    69*          WRITE(NUNIT,8000) IEOR                                        8/7     000247
```

```
00251    70*          KEY=2                                              8/76-RS.   000255
00252    71*          WRITE(NUNIT,8000) KEY                                 8/7     000257
00255    72*          WRITE(NUNIT,8000) NAME                                8/7     000265
00263    73*          IEOR=IEOR-1                                        8/76-RS.   000300
00264    74*          WRITE(NUNIT,8000) IEOR                                8/7     000303
00267    75*          KEY=0                                              8/76-RS.   000311
00270    76*          IFT=1                                              8/76-RS.   000312
00271    77*          P=4                                                8/76-RS.   000314
00272    78*          RETURN                                             8/76-RS.   000316
00272    79*     C                                                       8/76-RS.   000316
00272    80*     C                                                       8/76-RS.   000316
00273    81*          ENTRY COLUMN(ICOL)                                 8/76-RS.   000321
00275    82*          IF(IFT.EQ.1.AND.ICOL.GT.1) GO TO 110               8/76-RS.   000321
00277    83*          IF(IFT.EQ.1) GO TO 107                             8/76-RS.   000336
00301    84*          IF(JJ.GT.0) GO TO 90                               8/76-RS.   000341
00303    85*          P=P-2                                              8/76-RS.   000344
00304    86*          KEY=KEY-1                                          8/76-RS.   000347
00305    87*          IF(LAST.EQ.-3) GO TO 105                           8/76-RS.   000352
00307    88*          IFT=IFT-1                                          8/76-RS.   000355
00310    89*          LAST=-2                                            8/76-RS.   000360
00311    90*          GO TO 107                                          8/76-RS.   000362
00312    91*     90   FLD(0,20,W(KSV))=JJ                               8/76-RS.   000364
00313    92*          ASSIGN 105 TO IRET                                 8/76-RS.   000373
00314    93*          GO TO 5000                                         8/76-RS.   000375
00315    94*     105  P=P+1                                              8/76-RS.   000377
00316    95*          IEOR=IEOR-1                                        8/76-RS.   000401
00317    96*          WRITE(NUNIT,8000) IEOR                                8/7     000404
00322    97*     107  CONTINUE                                           8/76-RS.   000413
00323    98*     110  IKL=ICOL-IFT                                       8/76-RS.   000413
00324    99*          IF(IKL) 150,130,115                                8/76-RS.   000415
00327   100*     115  KEY=2                                              8/76-RS.   000417
00330   101*          P=P+2                                              8/76-RS.   000421
00331   102*          W(1)=ITRL(5)                                       8/76-RS.   000424
00332   103*          W(2)=NULL                                          8/76-RS.   000426
00333   104*          ASSIGN 120 TO IRET                                 8/76-RS.   000430
00334   105*          GO TO 5000                                         8/76-RS.   000432
00335   106*     120  IEOR=IEOR-1                                        8/76-RS.   000434
00336   107*          WRITE(NUNIT,8000) IEOR                                8/7     000436
00341   108*          IFT=IFT+1                                          8/76-RS.   000444
00342   109*          GO TO 110                                          8/76-RS.   000447
00343   110*     130  KEY=KEY+1                                          8/76-RS.   000451
00344   111*          IFT=IFT+1                                          8/76-RS.   000453
00345   112*          P=P+2                                              8/76-RS.   000456
00346   113*          W(KEY)=ITRL(5)                                     8/76-RS.   000461
00347   114*          LAST=-1                                            8/76-RS.   000466
00350   115*          JJ=0                                               8/76-RS.   000470
00351   116*          RETURN                                             8/76-RS.   000471
00352   117*     150  WRITE(NOUT,151) ICOL,IFT                          8/76-RS.   000475
00356   118*     151  FORMAT('0 COLUMN TERMINATES AS COLUMN INDEX DECREASED',2I7)  8/76-RS.   000503
00357   119*          STOP                                               8/76-RS.   000503
00357   120*     C                                                       8/76-RS.   000503
00357   121*     C                                                       8/76-RS.   000503
00360   122*          ENTRY ELEMNT(IROW,A)                               8/76-RS.   000505
00362   123*          INTEGER A(4)                                       8/76-RS.   000505
00363   124*          MM=0                                               8/76-RS.   000505
00364   125*          IF(IROW-LAST.GT.1) MM=1                            8/76-RS.   000506
```

```
00366   126*          M=MOD(P+1,SS)                                    8/76-RS.  000515
00367   127*          IF(M.EQ.0.OR.M.EQ.1) GO TO 195                   8/76-RS.  000522
00371   128*          M=SS-M                                           8/76-RS.  000535
00372   129*          IF(M.GE.IWD+MM) GO TO 200                        8/76-RS.  000540
00374   130*          DO 190 L=1,M                                     8/76-RS.  000545
00377   131*          KEY=KEY+1                                        8/76-RS.  000551
00400   132*    190   W(KEY)=PAD                                      8/76-RS.  000553
00402   133*          P=P+M                                            8/76-RS.  000560
00403   134*    195   LAST=-1                                         8/76-RS.  000564
00404   135*          P=P+2                                            8/76-RS.  000565
00405   136*    200   II=IROW-LAST                                    8/76-RS.  000571
00406   137*          IF(II.LE.0) GO TO 250                            8/76-RS.  000573
00410   138*          JJ=JJ+1                                          8/76-RS.  000575
00411   139*          IF(II.EQ.1) GO TO 210                            8/76-RS.  000600
00413   140*          IF(JJ.EQ.1) GO TO 205                            8/76-RS.  000606
00415   141*          FLD(0,20,W(KSV))=JJ-1                            8/76-RS.  000610
00416   142*          JJ=1                                             8/76-RS.  000616
00417   143*    205   KEY=KEY+1                                       8/76-RS.  000621
00420   144*          KSV=KEY                                          8/76-RS.  000623
00421   145*          FLD(20,16,W(KEY))=IROW                           8/76-RS.  000624
00422   146*          P=P+1                                            8/76-RS.  000634
00423   147*    210   DO 220 L=1,IWD                                   8/76-RS.  000640
00426   148*    220   W(KEY+L)=A(L)                                    8/76-RS.  000656
00430   149*          KEY=KEY+IWD                                      8/76-RS.  000660
00431   150*          P=P+IWD                                          8/76-RS.  000663
00432   151*          LAST=IROW                                        8/76-RS.  000666
00433   152*          IF(KEY.LT.NN) GO TO 240                          8/76-RS.  000670
00435   153*          FLD(0,20,W(KSV))=JJ                              8/76-RS.  000673
00436   154*          LAST=-3                                          8/76-RS.  000700
00437   155*          JJ=0                                             8/76-RS.  000702
00440   156*          ASSIGN 240 TO IRET                               8/76-RS.  000703
00441   157*          GO TO 5000                                       8/76-RS.  000705
00442   158*    240   CONTINUE                                        8/76-RS.  000707
00443   159*          RETURN                                           8/76-RS.  000707
00444   160*    250   WRITE(NOUT,251) IROW,LAST                       8/76-RS.  000712
00450   161*    251   FORMAT('0 ELEMNT TERMINATES AS ROW INDEX DECREASED',2I7)  8/76-RS  000720
00451   162*          STOP                                             8/76-RS.  000720
00451   163*    C                                                      8/76-RS.  000720
00451   164*    C                                                      8/76-RS.  000720
00452   165*          ENTRY WEOF                                       8/76-RS.  000722
00453   166*          IF(KEY.EQ.0) GO TO 320                           8/76-RS.  000722
00455   167*          FLD(0,20,W(KSV))=JJ                              8/76-RS.  000724
00456   168*          JJ=0                                             8/76-RS.  000734
00457   169*          ASSIGN 310 TO IRET                               8/76-RS.  000735
00460   170*          GO TO 5000                                       8/76-RS.  000737
00461   171*    310   IEOR=IEOR-1                                     8/76-RS.  000741
00462   172*          WRITE(NUNIT,8000) IEOR                             8/7   000743
00465   173*    320   IKL=ITRL(2)-IFT                                  8/76-RS.  000752
00466   174*          IF(IKL.LE.0) GO TO 340                           8/76-RS.  000754
00470   175*          KEY=2                                            8/76-RS.  000757
00471   176*          W(1)=ITRL(5)                                     8/76-RS.  000761
00472   177*          W(2)=NULL                                        8/76-RS.  000763
00473   178*          ASSIGN 330 TO IRET                               8/76-RS.  000765
00474   179*          GO TO 5000                                       8/76-RS.  000767
00475   180*    330   IEOR=IEOR-1                                     8/76-RS.  000771
00476   181*          WRITE(NUNIT,8000) IEOR                             8/7   000773
```

```
00501    182*            IFT=IFT+1                                          8/76-RS.    001001
00502    183*            GO TO 320                                          8/76-RS.    001004
00503    184*      340   WRITE(NUNIT,8000) IEOF                                 8/7     001006
00506    185*            IEOR=0                                             8/76-RS.    001013
00507    186*            RETURN                                             8/76-RS.    001014
00507    187*     C                                                         8/76-RS.    001014
00507    188*     C                                                         8/76-RS.    001014
00510    189*     5000   CONTINUE                                           8/76-RS.    001020
00511    190*            IF(KEY.GT.NN+8) GO TO 6000                         8/76-RS.    001020
00513    191*            WRITE(NUNIT,8000) KEY                                  8/7     001024
00516    192*            WRITE(NUNIT,8000) (W(I),I=1,KEY)                       8/7     001034
00524    193*            KEY=3                                              8/76-RS.    001047
00525    194*            GO TO IRET,   (105,120,240,310,330)                8/76-RS.    001050
00526    195*      6000  WRITE(NOUT,6001) KEY                                8/76-RS.    001052
00531    196*            RETURN 0                                           8/76-RS.    001057
00532    197*      6001  FORMAT('0 WORK AREA IS TOO SMALL  KEY= ',I6)        8/76-RS.    001153
00533    198*            END                                                8/76-RS.    001153

        END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      MATS
```

```
@FOR,S    MATS,MATS
HSA  E3  -01/09/78-23:44:07  (1,)

     SUBROUTINE ELMATS    ENTRY POINT 000231

     STORAGE USED: CODE(1) 000312; DATA(0) 000064; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   PARDOF 000002

     EXTERNAL REFERENCES (BLOCK, NAME)

       0004   OUTMAT
       0005   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000065 112G      0001   000074 116G      0001   000113 124G      0001   000121 125L      0001   000160 143G
       0001   000131 145L      0000 I 000003 I         0000 I 000004 IE        0000 I 000002 II        0000   000017 INJP$
       0000 I 000005 J         0000 I 000007 K         0000 I 000010 L         0003   000000 MAXDOF    0000 I 000000 NAME
       0003   000001 NGD       0000 I 000006 NJ

00101     1*          SUBROUTINE ELMATS(CS,CL,NUMNP,NUMEL,NPE,NODE,NN,CD,IBDT,IBD,SBD,      000036
00101     2*         1NBS,L1,L2)                                                            000036
00103     3*          DIMENSION NAME(2),CS(NPE,NPE),CL(NPE),NODE(NUMEL,NPE)                 000036
00103     4*         1  ,CD(NPE,NPE)                                                        000036
00104     5*          DIMENSION IBDT(3,3),IBD(NBS,3,3),SBD(NBS,3,3)                         000036
00105     6*          COMMON /PARDOF/ MAXDOF,NGD                                            000036
00106     7*          IF (IBDT(L1,L2) .EQ. 0) GO TO 145                                     000036
00110     8*          II=IBDT(L1,L2)                                                        000040
00111     9*          DO 140 I=1,II                                                         000055
00114    10*          IE=IBD(I,L1,L2)                                                       000065
00115    11*          DO 125 J=1,NPE                                                        000074
00120    12*          NJ=NODE(NN,J)                                                         000101
00121    13*          IF (NJ .NE. IE) GO TO 125                                             000103
00123    14*          DO 120 K=1,NPE                                                        000113
00126    15*      120 CS(J,K)=0.0                                                           000113
00130    16*          CS(J,J)=1.0                                                           000114
00131    17*          CL(J)=SBD(I,L1,L2)                                                    000116
00132    18*      125 CONTINUE                                                              000131
00134    19*      140 CONTINUE                                                              000131
00136    20*      145 CONTINUE                                                              000131
00137    21*          NAME(1)=4HCS                                                          000131
00140    22*          NAME(2)=4H                                                            000132
00141    23*          CALL OUTMAT(NAME,CS,NUMNP,1,NODE,NUMEL,NPE,NN,L,1)                    000134
00142    24*          DO 150 I=1,NPE                                                        000150
00145    25*      150 CD(I,1)=CL(I)                                                         000160
00147    26*          NAME(1)=4HCL                                                          000162
```

```
      MATS                                                                    DATE 010978

00150     27*           NAME(2)=4H                                                  000164
00151     28*           CALL OUTMAT(NAME,CO,1,1,NODE,NUMEL,NPE,NN,L,1)              000166
00152     29*           RETURN                                                      000202
00153     30*           END                                                         000311

          END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      MATX
```

```
@FOR,S    MATX,MATX
HSA  E3 -01/09/78-23:44:09 (1,)

     SUBROUTINE ELMTRX    ENTRY POINT 000231

     STORAGE USED: CODE(1) 000320; DATA(0) 000057; BLANK COMMON(2) 000000

      COMMON BLOCKS:

     0003   PARDOF 000002

     EXTERNAL REFERENCES (BLOCK, NAME)

      0004   OUTMAT
      0005   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000057 112G      0001   000066 116G      0001   000105 124G      0001   000113 125L      0001   000162 145G
      0001   000123 145L      0000 I 000003 I         0000 I 000004 IE        0000 I 000002 II        0000   000016 INJP$
      0000 I 000005 J         0000 I 000007 K         0003   000000 MAXDOF    0000 I 000000 NAME      0003   000001 NGD
      0000 I 000006 NJ

00101      1*          SUBROUTINE ELMTRX(CS,CL,NUMNP,NUMEL,NPE,NODE,NN,CD,IBDT,IBD,VBD,    000016
00101      2*         1  NBM,L,IPARAM)                                                     000016
00103      3*          DIMENSION NAME(2),CS(NPE,NPE),CL(NPE),NODE(NUMEL,NPE)              000016
00103      4*         1  ,CD(NPE,NPE)                                                     000016
00104      5*          DIMENSION IBD(NBM,1),VBD(NBM,1),IBDT(1)                            000016
00105      6*          COMMON /PARDOF/ MAXDOF,NGD                                         000016
00106      7*          IF (IBDT(L) .EQ. 0) GO TO 145                                      000016
00110      8*          II=IBDT(L)                                                         000022
00111      9*          DO 140 I=1,II                                                      000047
00114     10*          IE=IBD(I,L)                                                        000057
00115     11*          DO 125 J=1,NPE                                                     000066
00120     12*          NJ=NODE(NN,J)                                                      000073
00121     13*          IF (NJ .NE. IE) GO TO 125                                          000075
00123     14*          DO 120 K=1,NPE                                                     000105
00126     15*      120 CS(J,K)=0.0                                                        000105
00130     16*          CS(J,J)=1.0                                                        000106
00131     17*          CL(J)=VBD(I,L)                                                     000110
00132     18*      125 CONTINUE                                                           000123
00134     19*      140 CONTINUE                                                           000123
00136     20*      145 CONTINUE                                                           000123
00137     21*          NAME(1)=4HCS                                                       000123
00140     22*          NAME(2)=4H                                                         000124
00141     23*          CALL OUTMAT(NAME,CS,NUMNP,1,NODE,NUMEL,NPE,NN,L,IPARAM)            000126
00142     24*          IF (IPARAM .GT. 1) RETURN                                          000142
00144     25*          DO 150 I=1,NPE                                                     000162
00147     26*      150 CD(I,1)=CL(I)                                                      000162
```

```
      MATX                                                                    DATE 010978

00151     27*           NAME(1)=4HCL                                                000164
00152     28*           NAME(2)=4H                                                  000166
00153     29*           CALL OUTMAT(NAME,CD,1,1,NODE,NUMEL,NPE,NN,L,1)              000170
00154     30*           RETURN                                                      000204
00155     31*           END                                                         000317

      END OF COMPILATION:  NO DIAGNOSTICS.

$HDG,P      MESHUP
```

```
@FOR,S    MESHUP,MESHUP
HSA  E3  -01/09/78-23:44:11 (1,)

     SUBROUTINE MESHUP    ENTRY POINT 000470

     STORAGE USED: CODE(1) 000520; DATA(0) 000217; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

       0003   NPRT$
       0004   NIO2$
       0005   NSTOP$
       0006   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000435 100L      0001   000102 11L       0000   000131 1200F     0001   000051 122G      0001   000070 127G
       0001   000115 145G      0001   000103 15L       0001   000151 152G      0001   000175 163G      0001   000200 166G
       0001   000241 176G      0001   000106 18L       0001   000260 210G      0001   000277 222G      0001   000321 233G
       0001   000156 35L       0001   000170 40L       0001   000372 66L       0001   000420 70L       0001   000426 80L
       0000 R 000124 DF0       0000 R 000130 DFX       0000 R 000127 DFY       0000 R 000115 DFZ       0000 R 000125 DF1
       0000 R 000126 DX        0000 I 000113 I         0000   000144 INJP$     0000 I 000114 J         0000 I 000121 K
       0000 I 000122 M         0000 I 000110 MJ        0000 I 000000 MOVE1     0000 I 000014 MOVE2     0000 I 000030 MOVE3
       0000 I 000044 MOVE4     0000 I 000060 MOVE5     0000 I 000074 MOVE6     0000 I 000116 N         0000 I 000120 ND
       0000 I 000123 NJ        0000 I 000117 NT

00101      1*          SUBROUTINE MESHUP(NPMT,NEMT,NPELG,NPE,IU,IORDC,NET,NPST,DELT,                    000051
00101      2*         1  KODE,NOD,DIS,X,XPRV,VERPRV)                                                    000051
00101      3*     C                                                                                    000051
00101      4*     C*************************                                                         000051
00101      5*     C                                                                                    000051
00101      6*     C       UPDATE MESH                                                                  000051
00101      7*     C                                                                                    000051
00101      8*     C*************************                                                         000051
00101      9*     C                                                                                    000051
00103     10*          DIMENSION KODE(NPMT),NOD(NEMT,NPELG),DIS(NPMT,3),X(NPMT,3),                     000051
00103     11*         1  XPRV(NPMT,3),VERPRV(NPMT,IU)                                                  000051
00104     12*          DIMENSION MOVE1(4,3),MOVE2(4,3),MOVE3(4,3),MOVE4(4,3),MOVE5(4,3),               000051
00104     13*         1  MOVE6(4,3),MJ(3)                                                              000051
00105     14*          DATA MOVE1/1,4,5,8,1,7,13,19,1,10,21,30/                                        000051
00107     15*          DATA MOVE2/1,2,5,6,1,3,13,15,1,4,21,24/                                         000051
00111     16*          DATA MOVE3/1,2,3,4,1,3,5,7,1,4,7,10/                                            000051
00113     17*          DATA MOVE4/2,3,6,7,3,5,15,17,4,7,24,27/                                         000051
00115     18*          DATA MOVE5/4,3,8,7,7,5,19,17,10,7,30,27/                                        000051
00117     19*          DATA MOVE6/5,6,7,8,13,15,17,19,21,24,27,30/                                     000051
00121     20*          DO 18 I=1,NPST                                                                  000051
00124     21*          IF (KODE(I) .EQ. 0) GO TO 18                                                    000062
00126     22*          DO 15 J=1,3                                                                     000070
00131     23*          DFZ=DIS(I,J)*VERPRV(I,J)                                                        000070
```

```
00132    24*          IF (DFZ) 10,11,15                                          000072
00135    25*       10 DIS(I,J)=DELT*VERPRV(I,J)/VERPRV(I,5)                      000074
00136    26*          GO TO 15                                                   000100
00137    27*       11 DIS(I,J)=0.0                                               000102
00140    28*       15 CONTINUE                                                   000115
00142    29*       18 CONTINUE                                                   000115
00144    30*          DO 40 I=1,NPST                                             000115
00147    31*          IF (KODE(I) .EQ. 0) GO TO 35                               000131
00151    32*          DO 30 J=1,3                                                000151
00154    33*       30 X(I,J)=XPRV(I,J)+DIS(I,J)                                  000151
00156    34*          GO TO 40                                                   000154
00157    35*       35 X(I,J)=XPRV(I,J)                                           000156
00160    36*       40 CONTINUE                                                   000200
00162    37*          DO 90 N=1,NET                                              000200
00165    38*          DO 80 I=1,NPE                                              000200
00170    39*          NT=I                                                       000200
00171    40*          ND=NOD(N,NT)                                               000206
00172    41*          IF (KODE(ND) .EQ. 0) GO TO 80                              000210
00174    42*          MJ(1)=0                                                    000214
00175    43*          DO 58 J=1,4                                                000241
00200    44*          IF (NT .EQ. MOVE1(J,IORDC)) MJ(1)=MOVE4(J,IORDC)           000241
00202    45*          IF (NT .EQ. MOVE4(J,IORDC)) MJ(1)=MOVE1(J,IORDC)           000245
00204    46*       58 CONTINUE                                                   000254
00206    47*          MJ(2)=0                                                    000254
00207    48*          DO 60 J=1,4                                                000260
00212    49*          IF (NT .EQ. MOVE2(J,IORDC)) MJ(2)=MOVE5(J,IORDC)           000260
00214    50*          IF (NT .EQ. MOVE5(J,IORDC)) MJ(2)=MOVE2(J,IORDC)           000264
00216    51*       60 CONTINUE                                                   000273
00220    52*          MJ(3)=0                                                    000273
00221    53*          DO 62 J=1,4                                                000277
00224    54*          IF (NT .EQ. MOVE3(J,IORDC)) MJ(3)=MOVE6(J,IORDC)           000277
00226    55*          IF (NT .EQ. MOVE6(J,IORDC)) MJ(3)=MOVE3(J,IORDC)           000303
00230    56*       62 CONTINUE                                                   000321
00232    57*          DO 70 K=1,3                                                000321
00235    58*          M=MJ(K)                                                    000321
00236    59*          IF (M .EQ. 0) GO TO 70                                     000323
00240    60*          NJ=NOD(N,M)                                                000325
00241    61*          DF0=XPRV(ND,K)-X(NJ,K)                                     000340
00242    62*          DF1=X(ND,K)-X(NJ,K)                                        000343
00243    63*          DFZ=DF0*DF1                                                000346
00244    64*          IF (DFZ) 65,65,66                                          000350
00247    65*       65 DX=DELT*VERPRV(ND,K)/VERPRV(ND,5)                          000352
00250    66*          DIS(ND,K)=DX                                               000356
00251    67*          DX=ABS(DX)                                                 000357
00252    68*          IF (DX .GE. ABS(DF0)) GO TO 100                            000361
00254    69*          X(ND,K)=XPRV(ND,K)+DIS(ND,K)                               000365
00255    70*          GO TO 70                                                   000370
00256    71*       66 DFY=ABS(DF1)-ABS(DF0)                                      000372
00257    72*          IF (DFY) 70,70,68                                          000375
00262    73*       68 DFX=ABS(XPRV(ND,K)-X(ND,K))                                000377
00263    74*          DX=DELT*VERPRV(ND,K)/VERPRV(ND,5)                          000403
00264    75*          IF (ABS(DX) .GE. DFX) GO TO 70                             000407
00266    76*          DIS(ND,K)=DX                                               000413
00267    77*          X(ND,K)=XPRV(ND,K)+DX                                      000415
00270    78*       70 CONTINUE                                                   000431
00272    79*       80 CONTINUE                                                   000431
```

```
      MESHUP                                                                    DATE 010978

00274      80*        90 CONTINUE                                                      000431
00276      81*           RETURN                                                        000431
00277      82*       100 PRINT 1200                                                   000435
00301      83*           STOP                                                         000440
00302      84*      1200 FORMAT (2X,'********** REDUCE THE TIME STEP, MESH TO COARSE')000517
00303      85*           END                                                          000517

          END OF COMPILATION:      NO  DIAGNOSTICS.

@HDG,P     NTABS
```

```
@ASM,S    NTABS,NTABS
ASM 14R1 RL72R1 01/09/78 23:44:14 (1,)

    1.     .02  000000  00 00 01 000000        NSTAB 47,1,1,1,1 6 7 5 0
                000001  00 00 00 000000
                000002  00 00 00 000000
                000003  00 00 00 000000
                000004  00 00 00 000000
                000005  00 40 00 000000
                000006  00 44 00 000000
                000007  00 41 00 000000
                000010  00 00 00 000000
                000011  00 00 00 000000
                000012  00 00 00 000000
                000013  00 00 00 000000
                000014  00 00 00 000000
                000015  00 00 00 000000
                000016  00 00 00 000000
                000017  00 00 00 000000
                000020  00 00 00 000000
                000021  00 00 00 000000
                000022  00 00 00 000000
                000023  00 00 00 000000
                000024  00 00 00 000000
                000025  00 00 00 000000
                000026  00 00 00 000000
                000027  00 00 00 000000
                000030  00 00 00 000000
                000031  00 00 00 000000
                000032  00 00 00 000000
                000033  00 00 00 000000
                000034  00 00 00 000000
                000035  00 00 00 000000
                000036  00 00 00 000000
                000037  00 00 00 000000
                000040  00 00 00 000000
                000041  00 00 00 000000
                000042  00 00 00 000000
                000043  00 00 00 000000
                000044  00 00 00 000000
                000045  00 00 00 000000
                000046  00 00 00 000000
                000047  00 00 00 000000
                000050  00 00 00 000000
                000051  00 00 00 000000
                000052  00 00 00 000000
                000053  00 00 00 000000
                000054  00 00 00 000000
                000055  00 00 00 000000
                000056  00 00 00 000000
    2.                                    END
                                           END.          ASSEMBLER IMAGE
NO END IMAGE

END ASM. ERRORS : NONE
```

```
NTABS                                                          DATE 010978

@HDG,P      OPENT
```

```
aFOR,S    OPENT,OPENT
HSA. E3 -01/09/78-23:44:18 (1,1)

    SUBROUTINE OPENTP     ENTRY POINT 000766
               OPENF      ENTRY POINT 001054
               READF      ENTRY POINT 001142
               SKIPF      ENTRY POINT 001242

    STORAGE USED: CODE(1) 001326; DATA(0) 000103; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   NRDU$
     0004   NIO2$
     0005   NIO1$
     0006   NWDU$
     0007   NBSP$
     0010   NERR4$
     0011   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000433 10L       0001   000627 100L      0001   000152 1001L     0001   000155 1002L     0001   000160 1003L
     0001   000163 1004L     0001   000166 1005L     0001   000372 1011L     0001   000375 1012L     0001   000400 1013L
     0001   000403 1014L     0001   000575 1020L     0001   000600 1021L     0001   000603 1022L     0001   000606 1023L
     0001   000611 1024L     0001   000717 1031L     0001   000722 1032L     0001   000727 1034L     0001   000037 123G
     0001   000063 136G      0001   000107 151G      0001   000123 160G      0001   000644 200L      0000   000012 2001F
     0000   000017 2002F     0000   000025 2011F     0000   000033 2012F     0000   000040 2021F     0000   000045 2022F
     0000   000056 2031F     0001   000663 220L      0001   000673 230L      0001   000240 233G      0001   000275 253G
     0001   000310 261G      0001   000341 276G      0001   000366 312G      0001   000461 352G      0001   000512 362G
     0001   000710 472G      0001   000544 55L       0001   000553 60L       0001   000563 70L       0001   000150 8L
     0000   000011 8000F     0001   000624 999L      0000 I 000004 I         0000 I 000006 II        0000   000065 INJP$
     0000 I 000002 IWD       0000 I 000010 IX        0000 I 000003 K         0000 I 000007 L         0000 I 000005 LAST
     0000 I 000001 NOUT      0000 I 000000 UNIT

00101      1*            SUBROUTINE OPENTP(NNN,N,NWD,IERR,ILB)                                    000000
00101      2*      C      NNN    NUMBER OF UNIT                                                   000000
00103      3*            INTEGER UNIT,N(1)                                                       000000
00104      4*            DATA NOUT/6/                                                            000000
00104      5*      C    OPENTP   FIRST CALL TO PROCESS NASTRAN TAPE MADE BY OUTPUT2 ---- TAPE HEADER 000000
00104      6*      C    OPENF    EVERY DATA-BLOCK HAS TWO RECORDS.   1.NAME   2.TRAILER            000000
00104      7*      C    READF    TRUE NASTRAN DATA-BLOCK BEGINS FROM THIRD RECORD                   000000
00104      8*      C      A       IS ADRESS OF UESR CORE TO READ-IN                                000000
00104      9*      C      WD      NUMBER OF WORDS TO READ (DESIRE  INPUT) (RESULT  OUTPUT)         000000
00104     10*      C      FLAG    NUMBER OF WORDS LEFT IN THE CURRENT RECORD, (OUTPUT)             000000
00106     11*            IERR=0                                                                   000000
00107     12*            UNIT=NNN                                                                 000000
00110     13*      8000  FORMAT(22A6)                                                             000002
00111     14*            IWD=NWD                                                                  000002
```

```
00112    15*          IF(ILB.EQ.0)  GO TO 8                                    000004
00114    16*          READ(UNIT,8000)  K                                       000006
00117    17*          IF(K.NE.3) GO TO 1001                                    000014
00121    18*          READ(UNIT,8000) (N(I),I=1,K)                             000023
00127    19*          READ(UNIT,8000)  K                                       000042
00132    20*          IF(K.NE.7) GO TO 1002                                    000050
00134    21*          READ(UNIT,8000) (N(I),I=4,10)                            000053
00142    22*          READ(UNIT,8000)  K                                       000066
00145    23*          IF(K.NE.2) GO TO 1003                                    000074
00147    24*          READ(UNIT,8000) (N(I),I=11,12)                           000077
00155    25*          WRITE(NOUT,2001)  UNIT,(N(I),I=1,12)                     000112
00164    26*    2001  FORMAT('0 UNIT=',I3,5X,3I4,9A4)                          000126
00165    27*          READ(UNIT,8000)  K                                       000126
00170    28*          IF(K.NE.-1) GO TO 1004                                   000134
00172    29*          READ(UNIT,8000)  K                                       000137
00175    30*          IF(K.NE.0)  GO TO 1005                                   000145
00177    31*        8 CONTINUE                                                 000150
00200    32*          LAST=0                                                   000150
00201    33*          GO TO 999                                                000150
00202    34*     1001 IERR=IERR-1                                              000152
00203    35*     1002 IERR=IERR-1                                              000155
00204    36*     1003 IERR=IERR-1                                              000160
00205    37*     1004 IERR=IERR-1                                              000163
00206    38*     1005 IERR=IERR-1                                              000166
00207    39*          WRITE(NOUT,2002)  IERR                                   000170
00212    40*     2002 FORMAT('0  OPENTP ERROR  IERR=',I6)                      000176
00213    41*          GO TO 999                                                000176
00213    42*      C                                                            000176
00214    43*          ENTRY OPENF(IERR,ITRL)                                   000177
00216    44*          DIMENSION ITRL(1)                                        000177
00217    45*          IERR=-10                                                 000177
00220    46*          READ(UNIT,8000)  K                                       000201
00220    47*      C   READ AGAIN IF EOF IS FOUND                               000201
00223    48*          IF(K.EQ.0) READ (UNIT,8000) K                            000207
00227    49*          IF(K.NE.2) GO TO 1011                                    000217
00231    50*          READ(UNIT,8000) (N(I),I=1,2)                             000226
00237    51*          READ(UNIT,8000)  K                                       000243
00242    52*          IF(K.NE.-1) GO TO 1012                                   000251
00244    53*          READ(UNIT,8000)  K                                       000254
00247    54*          IF(K.NE.7)  GO TO 1013                                   000262
00251    55*          READ(UNIT,8000) (N(I),I=3,9)                             000265
00257    56*          WRITE(NOUT,2011) (N(I),I=1,9)                            000300
00265    57*    2011  FORMAT('0',4X,2A4,'    TRAILER=',7I8)                    000313
00266    58*          READ(UNIT,8000)  K                                       000313
00271    59*          READ(UNIT,8000) K                                        000321
00274    60*          READ(UNIT,8000) (N(I),I=1,K)                             000327
00302    61*          READ(UNIT,8000) K                                        000344
00305    62*          IF(K.NE.-3)  GO TO 1014                                  000352
00307    63*          LAST=K                                                   000355
00310    64*          IERR=0                                                   000357
00311    65*          DO 5 I=1,7                                               000366
00314    66*        5 ITRL(I)=N(I+2)                                           000366
00316    67*          GO TO 999                                                000370
00317    68*     1011 IERR=IERR-1                                              000372
00320    69*     1012 IERR=IERR-1                                              000375
00321    70*     1013 IERR=IERR-1                                              000400
```

```
00322     71*     1014 IERR=IERR-1                                                    000403
00323     72*          WRITE(NOUT,2012)  IERR                                         000405
00326     73*     2012 FORMAT('0  OPENF ERROR  IERR=',I6)                             000413
00327     74*          GO TO 999                                                      000413
00327     75*      C                                                                  000413
00330     76*          ENTRY READF(*,*,A,WD,FLAG,IERR)                                000414
00332     77*          INTEGER A(1),WD,FLAG                                           000414
00333     78*          IERR=-20                                                       000414
00334     79*          II=0                                                           000416
00335     80*          L=WD                                                           000417
00336     81*          READ(UNIT,8000)  K                                             000421
00341     82*          IF(K) 1020,100,10                                              000427
00344     83*       10 IF(K.GT.L)     L=K                                            000433
00346     84*          IF(IWD.LT.K)  GO TO 1021                                       000443
00350     85*          READ(UNIT,8000)  (N(I),I=1,K)                                  000447
00356     86*          L=L-K                                                          000464
00357     87*          IF(II+K.GT.WD) K=WD-II                                         000467
00361     88*          DO 40 I=1,K                                                    000477
00364     89*       40 A(II+I)=N(I)                                                   000512
00366     90*          II=II+K                                                        000514
00367     91*          READ(UNIT,8000)  K                                             000517
00372     92*          IF(K)  50,1022,60                                              000525
00375     93*       50 IF(K.NE.LAST-1)  GO TO 1023                                    000530
00377     94*          LAST=K                                                         000534
00400     95*          FLAG=1                                                         000536
00401     96*          IF(L)  1024,70,55                                              000540
00404     97*       55 WD=II                                                          000544
00405     98*          IERR=0                                                         000545
00406     99*          RETURN 2                                                       000546
00407    100*       60 IF(L)  1024,65,10                                              000553
00412    101*       65 FLAG=0                                                         000556
00413    102*          BACKSPACE UNIT                                                 000557
00414    103*       70 IF(WD.GT.II) GO TO 55                                          000563
00416    104*          WD=II                                                          000566
00417    105*          IERR=0                                                         000570
00420    106*          RETURN                                                         000571
00421    107*     1020 IERR=IERR-1                                                    000575
00422    108*     1021 IERR=IERR-1                                                    000600
00423    109*     1022 IERR=IERR-1                                                    000603
00424    110*     1023 IERR=IERR-1                                                    000606
00425    111*     1024 IERR=IERR-1                                                    000611
00426    112*     1025 IERR=IERR-1                                                    000613
00427    113*          WRITE(NOUT,2021)  IERR                                         000615
00432    114*     2021 FORMAT('0  READF ERROR  IERR=',I6)                             000624
00433    115*      999 RETURN                                                         000624
00434    116*      100 CONTINUE                                                       000627
00435    117*          WRITE(NOUT,2022)  UNIT                                         000627
00440    118*     2022 FORMAT('0  READF ENCOUNTERED EOF MARK WHILE READING',I4)       000634
00441    119*          RETURN 1                                                       000634
00441    120*      C                                                                  000634
00441    121*      C                                                                  000634
00442    122*          ENTRY SKIPF(IERR)                                              000640
00444    123*          IERR=-30                                                       000640
00445    124*          IX=0                                                           000642
00446    125*      200 READ(UNIT,8000) K                                              000644
00451    126*          IF(K)  210,220,230                                             000651
```

```
      OPENT                                                                DATE 010978

00454    127*        210 IF(LAST-1.NE.5)  GO TO 1031                          000654
00456    128*            LAST=K                                               000660
00457    129*            GO TO 200                                            000661
00460    130*        220 IF(IX.EQ.1) GO TO 1032                               000663
00462    131*            LAST=0                                               000665
00463    132*            IERR=0                                               000666
00464    133*            RETURN                                               000667
00465    134*        230 CONTINUE                                             000673
00466    135*            IF(K.GT.IWD)  GO TO 1034                             000673
00470    136*            READ(UNIT,8000) (N(I),I=1,K)                         000676
00476    137*            IX=1                                                 000713
00477    138*            GO TO 200                                            000715
00500    139*       1031 IERR=IERR-1                                          000717
00501    140*       1032 IERR=IERR-1                                          000722
00502    141*       1033 IERR=IERR-1                                          000724
00503    142*       1034 IERR=IERR-1                                          000727
00504    143*            WRITE(NOUT,2031) IERR                                000731
00507    144*       2031 FORMAT('0  SKIPF ERROR   IERR=',I5)                  000737
00510    145*            RETURN                                               000737
00511    146*            END                                                  001325

        END OF COMPILATION:        NO  DIAGNOSTICS.

@HDG,P      OUTM
```

```
@FOR,S    OUTM,OUTM
HSA  E3 -01/09/78-23:44:21 (0,)

     SUBROUTINE OUTMAT    ENTRY POINT 000372

     STORAGE USED: CODE(1) 000456; DATA(0) 000104; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   LAGRGN 000005
      0004   PARDOF 000002

     EXTERNAL REFERENCES (BLOCK, NAME)

      0005   SORT
      0006   MATRX
      0007   COLUMN
      0010   ELEMNT
      0011   WEOF
      0012   NERR2$
      0013   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000024 100L      0001   000135 120L      0001   000054 122G      0001   000066 125G      0001   000112 135G
      0001   000147 150L      0001   000165 154G      0001   000240 167G      0001   000257 173G      0001   000213 200L
      0001   000303 203G      0001   000344 220G      0001   000331 220L      0000 D 000041 CDP       0000 I 000040 DOF
      0000 I 000044 I         0000   000056 INJP$     0003   000002 IORDL     0000 I 000046 J         0000 I 000045 K
      0000 I 000050 LEE       0000 I 000047 LI        0000 I 000052 LL        0004 I 000000 MAXDOF    0003   000003 NBWT
      0000 I 000043 NCOLMN    0003   000001 NELT      0004   000001 NGD       0003   000004 NHBWT     0000 I 000051 NJ
      0003 I 000000 NPST      0000 I 000000 NY

00101      1*          SUBROUTINE OUTMAT(NAME,CD,NUMNP, NFORM,NODE,NUMEL,NPE,NN,L,IPARAM)          000005
00103      2*          DIMENSION NAME(1),CD(NPE,1),NODE(NUMEL,NPE)                                 000005
00104      3*          DIMENSION NY(32)                                                            000005
00104      4*     C                                                                                000005
00105      5*          INTEGER DOF                                                                 000005
00105      6*     C                                                                                000005
00106      7*          COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                                  000005
00107      8*          COMMON /PARDOF/ MAXDOF,NGD                                                  000005
00107      9*     C                                                                                000005
00110     10*          DATA DOF/6/                                                                 000005
00110     11*     C                                                                                000005
00112     12*          DOUBLE PRECISION CDP                                                        000005
00112     13*     C                                                                                000005
00113     14*          CALL SORT(NODE,NY,NN,NPE, NUMEL)                                            000005
00114     15*          GO TO (100,200),IPARAM                                                      000014
00115     16*      100 NCOLMN=NPST                                                                 000024
00116     17*          IF (NUMNP .EQ. 1) GO TO 150                                                 000025
```

```
OUTH                                                                    DATE 010978

00120    18*          CALL MATRX(NAME,NCOLMN, NCOLMN,NFORM,1)                    000030
00121    19*          DO 120 I=1,NUMNP                                           000045
00124    20*          DO 120 K=1,NPE                                             000066
00127    21*          IF (I .NE. NODE(NN,K) .AND. I .NE. NUMNP) GO TO 120        000077
00131    22*          CALL COLUMN(I)                                             000102
00132    23*          IF (I .NE. NODE(NN,K)) GO TO 120                           000105
00134    24*          DO 110 J=1,NPE                                             000112
00137    25*          LI=NY(J)                                                   000112
00140    26*          CALL ELEMNT(NODE(NN,LI),CD(LI,K))                          000113
00141    27*      110 CONTINUE                                                   000141
00143    28*      120 CONTINUE                                                   000141
00146    29*          CALL WEOF                                                  000141
00147    30*          RETURN                                                     000143
00150    31*      150 CONTINUE                                                   000147
00151    32*          CALL MATRX(NAME,1,NCOLMN,NFORM,1)                          000147
00152    33*          CALL COLUMN(1)                                             000155
00153    34*          DO 160 J=1,NPE                                             000165
00156    35*          LI=NY(J)                                                   000165
00157    36*          CALL ELEMNT(NODE(NN,LI),CD(LI,1))                          000166
00160    37*      160 CONTINUE                                                   000205
00162    38*          CALL WEOF                                                  000205
00163    39*          RETURN                                                     000207
00164    40*      200 NCOLMN=MAXDOF                                              000213
00165    41*          CALL MATRX(NAME,NCOLMN, NCOLMN,NFORM,2)                    000214
00166    42*          DO 220 I=1,NUMNP                                           000240
00171    43*          LI=DOF*(I-1)+L                                             000246
00172    44*          DO 220 K=1,NPE                                             000257
00175    45*          IF (I .NE. NODE(NN,K) .AND. I .NE. NUMNP) GO TO 220        000270
00177    46*          CALL COLUMN(LI)                                            000273
00200    47*          IF (I .NE. NODE(NN,K)) GO TO 220                           000276
00202    48*          DO 210 J=1,NPE                                             000303
00205    49*          LEE=NY(J)                                                  000303
00206    50*          NJ=DOF*(NODE(NN,LEE)-1)+L                                  000314
00207    51*          CDP=DBLE(CD(LEE,K))                                        000321
00210    52*          CALL ELEMNT(NJ,CDP)                                        000323
00211    53*      210 CONTINUE                                                   000335
00213    54*      220 CONTINUE                                                   000335
00216    55*          LL=LI+1                                                    000335
00217    56*          DO 230 I=LL,MAXDOF                                         000340
00222    57*          CALL COLUMN(I)                                             000344
00223    58*      230 CONTINUE                                                   000351
00225    59*          CALL WEOF                                                  000351
00226    60*          RETURN                                                     000353
00227    61*          END                                                        000455

        END OF COMPILATION:       NO  DIAGNOSTICS.

@HDG,P      SFCL
```

```
SFCL                                                                    DATE 010978

@FOR,S    SFCL,SFCL
HSA  E3 -01/09/78-23:44:22 (0,)

      SUBROUTINE SFCALL    ENTRY POINT 000066

      STORAGE USED: CODE(1) 000101; DATA(0) 000055; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

       0003   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000012 1100      0000 I 000030 I        0000   000041 INJP$     0000 R 000000 XNODE     0000 R 000031 XNP1
       0000 R 000032 YNP1       0000 R 000033 ZNP1

00101     1*          SUBROUTINE SFCALL(XYZ,SF,DF)                                                    000012
00101     2*     C    3-D LINEAR ISOPARAMETRIC ELEMENT                                                000012
00101     3*     C    THIS SUBROUTINE EVALUATES THE SHAPE FUNCTIONS AND THEIR FIRST                   000012
00101     4*     C    DERIVATIVES AT THE GAUSSIAN POINT XYZ                                           000012
00103     5*          DIMENSION XNODE(8,3),XYZ(3),SF(8),DF(3,8)                                       000012
00104     6*          DATA XNODE/-1.0,2*1.0,2*-1.0,2*1.0,3*-1.0,2*1.0,2*-1.0,                         000012
00104     7*         1    2*1.0,4*-1.0,4*1.0/                                                         000012
00106     8*          FCK(A,B,C)=0.125*A*B*C                                                          000012
00107     9*          DO 20 I=1,8                                                                     000012
00112    10*          XNP1=XYZ(1)*XNODE(I,1)+1.0                                                      000012
00113    11*          YNP1=XYZ(2)*XNODE(I,2)+1.0                                                      000015
00114    12*          ZNP1=XYZ(3)*XNODE(I,3)+1.0                                                      000021
00115    13*          SF(I)=FCK(XNP1,YNP1,ZNP1)                                                       000025
00116    14*          DF(1,I)=FCK(XNODE(I,1),YNP1,ZNP1)                                               000031
00117    15*          DF(2,I)=FCK(XNP1,XNODE(I,2),ZNP1)                                               000036
00120    16*       20 DF(3,I)=FCK(XNP1,YNP1,XNODE(I,3))                                               000043
00122    17*          RETURN                                                                          000051
00123    18*          END                                                                             000100

      END OF COMPILATION:     NO DIAGNOSTICS.

@HDG,P      SFCQ
```

```
@FOR,S    SFCQ,SFCQ
HSA  E3 -01/09/78-23:44:24 (0,)

     SUBROUTINE SFCALQ    ENTRY POINT 000267

     STORAGE USED: CODE(1) 000311; DATA(0) 000233; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

       0003   NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000013 114G      0001   000077 130G      0001   000164 15L       0001   000234 20L       0000 R 000142 CST
       0000 R 000150 ETA       0000 I 000153 I         0000 I 000152 IED       0000 I 000144 II        0000   000170 INJP$
       0000 I 000154 I1        0000 I 000143 J         0000 I 000145 JJ        0000 I 000146 KK        0000 I 000074 NM
       0000 I 000134 NP        0000 R 000147 XI        0000 R 000000 XNODE     0000 R 000160 XNP1      0000 R 000155 XP
       0000 R 000161 YNP1      0000 R 000156 YP        0000 R 000151 ZETA      0000 R 000162 ZNP1      0000 R 000157 ZP

A-102

00101      1*          SUBROUTINE SFCALQ(XYZ,SF,DF)                                                    000013
00101      2*      C 3-D QUADRATIC ISOPARAMETRIC ELEMENT                                               000013
00101      3*      C   EVALUATE SHAPE FUNCTIONS AND FIRST DERIVATIVES AT THE                           000013
00101      4*      C   GAUSSIAN POINT XYZ                                                              000013
00103      5*          DIMENSION XNODE(20,3),NM(8,4),NP(6),XYZ(3),SF(20),DF(3,20)                      000013
00104      6*          DATA XNODE/-1.,0.,3*1.,0.,3*-1.,2*1.,2*-1.,0.,3*1.,0.,2*-1.,                    000013
00104      7*         1  3*-1.,0.,3*1.,0.,2*-1.,2*1.,3*-1.,0.,3*1.,0.,8*-1.,4*0.,8*1./                000013
00106      8*          DATA NM/2,6,18,14,4*0,8,20,16,4,4*0,9,10,11,12,4*0,                             000013
00106      9*         1      1,3,5,7,13,15,17,19/,NP/1,2,3,1,2,3/                                      000013
00111     10*          FCM(A,B,C)=0.25*(1.-A*A)*B*C                                                    000013
00112     11*          FCK(A,B,C,D)=0.125*A*B*C*(CST+D)                                                000013
00113     12*          DO 30 J=1,4                                                                     000013
00116     13*          II=NP(J)                                                                        000025
00117     14*          JJ=NP(J+1)                                                                      000032
00120     15*          KK=NP(J+2)                                                                      000037
00121     16*          XI=XYZ(II)                                                                      000044
00122     17*          ETA=XYZ(JJ)                                                                     000053
00123     18*          ZETA=XYZ(KK)                                                                    000055
00124     19*          IED=4                                                                           000057
00125     20*          IF (J .EQ. 4)  IED=8                                                            000066
00127     21*          DO 20 I=1,IED                                                                   000072
00132     22*          I1=NM(I,J)                                                                      000077
00133     23*          XP=XNODE(I1,II)                                                                 000122
00134     24*          YP=XNODE(I1,JJ)                                                                 000124
00135     25*          ZP=XNODE(I1,KK)                                                                 000126
00136     26*          XNP1=XP*XI+1.                                                      SFC0490      000130
00137     27*          YNP1=YP*ETA+1.                                                     SFC0500      000133
00140     28*          ZNP1=ZP*ZETA+1.                                                    SFC0510      000136
00141     29*          IF (J .EQ. 4)  GO TO 15                                                         000141
00143     30*          SF(I1)=FCM(XI,YNP1,ZNP1)                                                        000143
```

```
SFCQ                                                                    DATE 010978

00144      31*            DF(II,I1)=-0.5*XI*YNP1*ZNP1                              000146
00145      32*            DF(JJ,I1)=FCM(XI,YP,ZNP1)                                000152
00146      33*            DF(KK,I1)=FCM(XI,ZP,YNP1)                                000156
00147      34*            GO TO 20                                                 000162
00150      35*         15 CST=XNP1+YNP1+ZNP1-5.                                    000164
00151      36*            SF(I1)=FCK(XNP1,YNP1,ZNP1,0.)                            000175
00152      37*            DF(1,I1)=FCK(XP,YNP1,ZNP1,XNP1)                          000203
00153      38*            DF(2,I1)=FCK(YP,ZNP1,XNP1,YNP1)                          000213
00154      39*            DF(3,I1)=FCK(ZP,XNP1,YNP1,ZNP1)                          000223
00155      40*         20 CONTINUE                                      SFC0610    000237
00157      41*         30 CONTINUE                                      SFC0790    000237
00161      42*            RETURN                                                   000237
00162      43*            END                                                      000310

        END OF COMPILATION:       NO DIAGNOSTICS.

@HDG,P      SFCU
```

```
@FOR,S    SFCU,SFCU
HSA  E3 -01/09/78-23:44:26 (0,)

   SUBROUTINE SFCALC    ENTRY POINT 000307

   STORAGE USED: CODE(1) 000326; DATA(0) 000301; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000027 117G      0001   000106 130G      0001   000206 15L      0001   000257 20L      0000 R 000207 CST
    0000 R 000215 ETA       0000 I 000217 I         0000 I 000211 II        0000   000237 INJP$    0000 I 000220 I1
    0000 I 000210 J         0000 I 000212 JJ        0000 I 000213 KK        0000 I 000140 NM        0000 I 000200 NP
    0000 R 000206 TEMP      0000 R 000227 TEMP1     0000 R 000214 XI        0000 R 000000 XNODE     0000 R 000224 XNP1
    0000 R 000221 XP        0000 R 000225 YNP1      0000 R 000222 YP        0000 R 000216 ZETA      0000 R 000226 ZNP1
    0000 R 000223 ZP

00101      1*            SUBROUTINE SFCALC(XYZ,SF,DF)                                              000011
00101      2*        C   3-D CUBIC ISOPARAMETRIC ELEMENT                                           000011
00101      3*        C   EVALUATE SHAPE FUNCTIONS AND FIRST DERIVATIVES AT THE                     000011
00101      4*        C   GAUSSIAN POINT XYZ                                                        000011
00103      5*            DIMENSION XNODE(32,3),NM(8,4),NP(6),XYZ(3),SF(32),DF(3,32)                000011
00104      6*            DATA XNODE/-1.,-.333333333,.333333333,4*1.,.333333333,          SFC0080   000011
00104      7*           1          -.333333333,4*-1.,2*1.,2*-1.,2*1.,2*-1.,-.333333333,  SFC0090   000011
00104      8*           2          .333333333,4*1.,.333333333,-.333333333,3*-1.,                     000011
00104      9*           3          4*-1.,-.333333333,.333333333,4*1.,.333333333,                     000011
00104     10*           4          -.333333333,2*-1.,2*1.,2*-1.,2*1.,4*-1.,-.333333333,  SFC0120   000011
00104     11*           5          .333333333,4*1.,.333333333,-.333333333,                           000011
00104     12*           6          12*-1.,4*-.333333333,4*.333333333,12*1./               SFC       000011
00106     13*            DATA NM/2,9,29,22,3,8,28,23,12,32,25,5,11,31,26,6,13,14,15,16,             000011
00106     14*           1      17,18,19,20,1,4,7,10,21,24,27,30/,NP/1,2,3,1,2,3/                    000011
00111     15*            FUNC1(A,B)=0.140625*A*B                                                   000011
00112     16*            FUNC2(A,B,C)=TEMP*(CST*A+2.*B*C)                                SFC0190   000011
00113     17*            FUNC4(A,B)=(1.-A*A)*(9.*B-8.)                                   SFC0210   000011
00114     18*            FUNC5(A,B)=TEMP*(9.*A*(1.-3.*B*B)-B-B)                          SFC0220   000011
00115     19*            CST=XYZ(1)**2+XYZ(2)**2+XYZ(3)**2-2.11111111                              000011
00116     20*            DO 30 J=1,4                                                               000027
00121     21*            II=NP(J)                                                                  000035
00122     22*            JJ=NP(J+1)                                                                000042
00123     23*            KK=NP(J+2)                                                                000047
00124     24*            XI=XYZ(II)                                                                000054
00125     25*            ETA=XYZ(JJ)                                                               000056
00126     26*            ZETA=XYZ(KK)                                                              000060
00127     27*            DO 20 I=1,8                                                     SFC0440   000073
00132     28*            II=NM(I,J)                                                                000106
00133     29*            XP=XNODE(II,II)                                                           000107
```

```
SFCU                                                                    DATE 010978

00134    30*            YP=XNODE(I1,JJ)                                        000113
00135    31*            ZP=XNODE(I1,KK)                                        000117
00136    32*            XNP1=XP*XI+1.                          SFC0490         000123
00137    33*            YNP1=YP*ETA+1.                         SFC0500         000126
00140    34*            ZNP1=ZP*ZETA+1.                        SFC0510         000131
00141    35*            IF (J .EQ. 4)  GO TO 15                                000134
00143    36*            TEMP=FUNC1(ZNP1,YNP1)                  SFC0700         000136
00144    37*            TEMP1=FUNC4(XI,XNP1)                   SFC0710         000141
00145    38*            SF(I1)=TEMP*TEMP1                      SFC0720         000145
00146    39*            DF(II,I1)=FUNC5(XP,XI)                                 000151
00147    40*            DF(JJ,I1)=TEMP1*FUNC1(YP,ZNP1)                         000172
00150    41*            DF(KK,I1)=TEMP1*FUNC1(ZP,YNP1)                         000177
00151    42*            GO TO 20                                               000204
00152    43*         15 TEMP=FUNC1(YNP1,ZNP1)                  SFC0330         000206
00153    44*            SF(I1)=TEMP*XNP1*CST                   SFC0340         000216
00154    45*            DF(1,I1)=FUNC2(XP,XI,XNP1)             SFC0350         000221
00155    46*            TEMP=FUNC1(ZNP1,XNP1)                  SFC0370         000230
00156    47*            DF(2,I1)=FUNC2(YP,ETA,YNP1)            SFC0380         000234
00157    48*            TEMP=FUNC1(XNP1,YNP1)                  SFC0400         000243
00160    49*            DF(3,I1)=FUNC2(ZP,ZETA,ZNP1)           SFC0410         000247
00161    50*         20 CONTINUE                               SFC0610         000262
00163    51*         30 CONTINUE                               SFC0790         000262
00165    52*            RETURN                                                 000262
00166    53*            END                                                    000325

      END OF COMPILATION:       NO  DIAGNOSTICS.

$HDG,P      SLID
```

```
@FOR,S    SLID,SLID
HSA  E3 -01/09/78-23:44:28 (8,)

   SUBROUTINE SLTEST    ENTRY POINT 003016

   STORAGE USED: CODE(1) 003041; DATA(0) 000544; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CELZ1  000015
    0004   CELZ3  001140
    0005   MATRL  000005
    0006   NCASE  000003
    0007   GEOM1  000524
    0010   LAGRGN 000005
    0011   NODEPJ 000003
    0012   SRFACE 000077
    0013   SURFCE 000010
    0014   STRESS 002530
    0015   DISPLC 001620
    0016   IMPCT1 006650
    0017   IMPCT2 000230
    0020   SOLCNV 000574
    0021   SOLSTR 001254
    0022   SOLDIS 000344
    0023   BDVER  001375
    0024   BDSTR  002541
    0025   BDDIS  000713
    0026   MATRL2 000230

   EXTERNAL REFERENCES (BLOCK, NAME)

    0027   NPRT$
    0030   NI02$
    0031   SQRT
    0032   NSTOP$
    0033   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  001643 1014G    0001  001655 1021G    0001  000331 104L    0001  001711 1042G    0001  001734 1055G
    0001  001752 1062G    0001  001762 1070G    0001  000357 110L    0000  000413 1100F    0001  000452 111L
    0001  002014 1111G    0001  002034 1117G    0001  000461 112L    0001  002047 1127G    0001  000467 113L
    0001  002071 1141G    0001  002104 1147G    0001  000703 115L    0001  002117 1155G    0001  002136 1167G
    0001  002157 1175G    0001  000705 118L    0000  000422 1200F    0000  000431 1201F    0000  000441 1202F
    0001  002211 1211G    0001  002214 1214G    0001  002232 1222G    0001  002260 1237G    0001  002263 1242G
    0001  002303 1254G    0001  002322 1266G    0001  001141 130L    0001  002341 1300G    0001  001143 131L
    0001  001165 132L    0001  002452 1337G    0001  001202 135L    0001  002553 1406G    0001  002570 1420G
    0001  002601 1431G    0001  001337 144L    0001  002620 1440G    0001  001346 145L    0001  002651 1454G
    0001  002657 1457G    0001  002706 1472G    0001  001374 148L    0001  001376 150L    0001  002734 1510G
    0001  002745 1517G    0001  001434 152L    0001  001463 155L    0001  001530 156L    0001  001536 160L
    0001  001543 162L    0001  000056 163G    0001  001626 165L    0001  001670 168L    0001  000056 170G
```

SLID DATE 010978

0001		001701	171L	0001		001725	173L	0001		001777	174L	0001		002002	175L	0001		002043	182L
0001		002045	185L	0001		002064	200L	0001		000232	241G	0001		000233	244G	0001		000333	277G
0001		002256	300L	0001		000347	307G	0001		002400	310L	0001		002424	314L	0001		002450	317L
0001		002474	319L	0001		002503	320L	0001		002516	321L	0001		002525	322L	0001		000367	323G
0001		002527	323L	0001		002543	325L	0001		002561	326L	0001		002563	327L	0001		002573	330L
0001		000442	334G	0001		002724	380L	0001		002757	397L	0001		000174	40L	0001		002761	410L
0001		000725	427G	0001		000753	437G	0001		001033	460G	0001		001045	465G	0001		000176	5DL
0001		002764	500L	0001		001102	507G	0001		001147	534G	0001		001152	540G	0001		001156	544G
0001		001207	565G	0001		001230	575G	0001		001243	603G	0001		001265	611G	0001		001436	677G
0001		001452	705G	0001		001467	717G	0001		001476	722G	0001		001512	732G	0001		001515	735G
0001		001553	756G	0001		001565	763G	0001		001604	774G	0003		000001	A	0000	R	000272	ABN
0000	R	000273	ABT	0003		000002	ALPHA	0000	R	000267	APB1	0000	R	000270	APB2	0000	R	000271	APB3
0015		001254	BFS	0016		001370	C	0000	R	000274	CN1	0000	R	000275	CN2	0000	R	000276	CN3
0000	R	000277	CT1	0000	R	000300	CT2	0000	R	000301	CT3	0003	R	000013	DELT	0000	R	000406	DF1
0000	R	000410	DF2	0000	R	000412	DF3	0000	R	000243	DIFF	0013	R	000001	DIFFPR	0000	R	000230	DIM1
0000	R	000231	DIM2	0000	R	000232	DIM3	0022		000000	DIS	0025	R	000347	DISBD	0015		000000	DISN
0015		000344	DISPRV	0000	R	000236	DMIN	0000	R	000233	DXMIN1	0000	R	000234	DXMIN2	0000	R	000235	DXMIN3
0000	R	000344	DX1	0000	R	000345	DX2	0000	R	000343	DX3	0016		001140	FJN	0016		001254	FJPRV
0000	R	000325	FNML	0000	R	000326	FTGN	0000	R	000320	F1	0000	R	000321	F2	0000	I	000213	I
0025	I	000003	IBDIS	0025	I	000000	IBDIST	0024	I	000011	IBDS	0024	I	000000	IBDST	0023	I	000005	IBDV
0023	I	000000	IBDVT	0000	I	000244	IDEX	0000	I	000362	IE	0000	I	000361	II	0000	I	000334	IN
0006	I	000000	INDEX	0000	I	000360	INDG	0000	I	000363	INEL	0000		000475	INJPS	0003		000011	INNER
0000	I	000000	INT	0000	I	000337	INTEL	0000	I	000041	INTELM	0012	I	000065	INTF	0011	I	000002	INTFT
0000	I	000340	INTND	0000	I	000351	INT1	0000	I	000355	INT2	0000	I	000350	IN1	0000	I	000354	IN2
0000	I	000377	IOMIT	0003	I	000000	IORDC	0010		000002	IORDL	0000	I	000331	IPLANE	0005		000004	ISOP
0005		000003	ISOT	0003	I	000012	ITER	0000	I	000214	I1	0000	I	000215	I2	0000	I	000216	I3
0000	I	000217	I4	0000	I	000220	I5	0000	I	000221	I6	0000	I	000241	J	0000	I	000252	JA
0000	I	000253	JB	0000	I	000246	J3	0000	I	000247	J6	0000	I	000341	K	0007	I	000000	KODE
0000	I	000364	KPE	0000	I	000374	LIAO	0000	I	000373	LIN	0000	I	000376	LK	0000	I	000365	M
0000	I	000366	MJ	0000	I	000402	MJ1	0000	I	000403	MJ2	0000	I	000404	MJ3	0000	I	000371	MK
0000	I	000375	MN	0000	I	000102	MOVE1	0000	I	000116	MOVE2	0000	I	000132	MOVE3	0000	I	000146	MOVE4
0000	I	000162	MOVE5	0000	I	000176	MOVE6	0000	I	000212	N	0006		000001	NASTRN	0003		000006	NBW
0010		000003	NBWT	0000	I	000313	NCRIT	0000	I	000401	ND	0000	I	000001	NDEG	0000	I	000370	NDIFEL
0010		000001	NELT	0003	I	000010	NEPJ	0011	I	000000	NEPJT	0003	I	000003	NET	0000	I	000356	NF
0000	I	000357	NFDG	0000	I	000367	NFEL	0005	I	000000	NFRAC	0012	I	000000	NFS	0003	I	000014	NFST
0000	I	000347	NFS1	0000	I	000353	NFS2	0000	I	000346	NF1	0000	I	000352	NF2	0003		000005	NHBW
0010		000004	NHBWT	0000	I	000250	NI	0000	I	000372	NJ	0000	I	000254	NJA	0000	I	000255	NJB
0000	I	000251	NJC	0000	I	000222	NJ1	0000	I	000223	NJ2	0000	I	000224	NJ3	0000	I	000225	NJ4
0000	I	000226	NJ5	0000	I	000227	NJ6	0000	I	000342	NK	0007	I	000114	NOD	0026	I	000114	NODFRC
0003	I	000007	NPPJ	0011	I	000001	NPPJT	0003	I	000004	NPS	0010		000000	NPST	0006	I	000002	NSLIDE
0000	I	000400	NT	0000	I	000405	NTX	0000	I	000407	NTY	0000	I	000411	NTZ	0000	I	000245	NVS
0000	I	000332	NXMIN1	0000	I	000333	NXMIN2	0000	I	000316	N1	0000	I	000317	N2	0000	I	000322	N3
0005		000002	PMIU	0017		000000	PRSN	0017		000114	PRSPRV	0000	R	000327	RT	0000	R	000312	RV
0024	R	001265	SBD	0013	R	000000	SMIU	0021	R	000000	STS	0014		000000	STSN	0014		001254	STSPRV
0000	R	000330	SUM	0005		000001	TMIU	0023	R	000601	VBD	0000	R	000314	VDIFF1	0000	R	000315	VDIFF2
0020	R	000000	VFR	0016		000344	VERN	0004	R	000344	VERPRV	0000	R	000237	VMAX	0013	R	000003	VMAXPR
0000	R	000240	VMIN	0013	R	000004	VMINPR	0000	R	000311	VN	0000	R	000242	VNOD	0000	R	000305	VN1
0000	R	000306	VN2	0000	R	000307	VN3	0013	R	000002	VO	0000	R	000310	VT	0000	R	000302	VT1
0000	R	000303	VT2	0000	R	000304	VT3	0000	R	000323	V1	0000	R	000324	V2	0015	R	000710	X
0000	R	000256	XA1	0000	R	000257	XA2	0000	R	000260	XA3	0000	R	000261	XB1	0000	R	000262	XB2
0000	R	000263	XB3	0000	R	000264	XC1	0000	R	000265	XC2	0000	R	000266	XC3	0013	R	000005	XDIM
0016	R	000000	XN	0004	R	000000	XPRV	0000	R	000335	XSLIP1	0000	R	000336	XSLIP2	0026		000000	YIELD

```
00101     1*          SUBROUTINE SLTEST                                                    000026
00101     2*     C                                                                         000026
00101     3*     C*********************                                                  000026
00101     4*     C                                                                         000026
00101     5*     C     TEST OF WHETHER THE PROJECTILE REBOUNDS, SLIDES OF PENETRATES       000026
00101     6*     C                                                                         000026
00101     7*     C         NSLIDE = 0 ----- NORMAL IMPACT                                  000026
00101     8*     C                  = 1 ----- OBLIQUE IMPACT                               000026
00101     9*     C                                                                         000026
00101    10*     C*********************                                                  000026
00101    11*     C                                                                         000026
00103    12*          INCLUDE PARAM1,LIST                                                  000026
00103    12*     PARAM1 PROC                                                               000026
00103    12*     C                                                                         000026
00104    12*          PARAMETER NPMT= 76,NEMT= 33,NPM= 76,NEM= 33                          000026
00105    12*          PARAMETER NPELG= 8,NPE= 8                                            000026
00106    12*          PARAMETER NDFRS=53,NDINT=10                                          000026
00106    12*     C                                                                         000026
00107    12*          PARAMETER NRMAX=2000,NCMAX=2                                         000026
00110    12*          PARAMETER NGPLG=2,NGP=2                                              000026
00111    12*          PARAMETER NCNT=6, NFRC=NPM                                           000026
00112    12*          PARAMETER IU=5,NDPJT= 10,NTSTEP=200                                  000026
00113    12*          PARAMETER NBV=NPM,NBS=NPM,NBD=NPM                                    000026
00113    12*      END                                                                      000026
00114    13*          INCLUDE TESTNG,LIST                                                  000026
00114    13*     TESTNG PROC                                                               000026
00115    13*          PARAMETER NDIM=NEM                                                   000026
00116    13*          DIMENSION NDEG(32),INTELM(NDIM)                                      000026
00116    13*     C                                                                         000026
00117    13*          COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NPPJ,NEPJ,INNER,       000026
00117    13*         1  ITER,DELT,NFST                                                     000026
00120    13*          COMMON /CELZ3/ XPRV(NPMT,3), VERPRV(NPMT,IU)                         000026
00121    13*          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                             000026
00122    13*          COMMON /NCASE/ INDEX,NASTRN,NSLIDE                                   000026
00123    13*          COMMON /GEOM1/ KODE(NPMT),NOD(NEMT,NPELG)                            000026
00124    13*          COMMON /LAGRGN/ NPST,NELT,IORDL,NBWT,NHBWT                           000026
00125    13*          COMMON /NODEPJ/ NEPJT,NPPJT,INTFT                                    000026
00126    13*          COMMON/SRFACE/ NFS(NDFRS),INTF(NDINT)                                000026
00127    13*          COMMON /SURFCE/ SMIU,DIFFPR,V0,VMAXPR,VMINPR,XDIM(3)                 000026
00130    13*          COMMON /STRESS/ STSN(NPM,3,3),STSPRV(NPM,3,3)                        000026
00131    13*          COMMON /DISPLC/ DISN(NPMT,3),DISPRV(NPMT,3),X(NPMT,3),BFS(NPMT,3)    000026
00132    13*          COMMON /IMPCT1/ XN(NPMT,3),VERN(NPMT,IU),FJN(NPM),FJPRV(NPM),        000026
00132    13*         1  C(NPMT,NCNT,NCNT)                                                 000026
00133    13*          COMMON /IMPCT2/ PRSN(NPM),PRSPRV(NPM)                                000026
00134    13*          COMMON /SOLCNV/ VER(NPMT,IU)                                         000026
00135    13*          COMMON /SOLSTR/ STS(NPM,3,3)                                         000026
00136    13*          COMMON /SOLDIS/ DIS(NPMT,3)                                          000026
00137    13*          COMMON /BDVER/ IBDVT(IU),IBDV(NBV,IU),VBD(NBV,IU)                    000026
00140    13*          COMMON /BDSTR/ IBDST(3,3),IBDS(NBS,3,3),SBD(NBS,3,3)                 000026
00141    13*          COMMON /BDDIS/ IBDIST(3),IBDIS(NBD,3),DISBD(NBD,3)                   000026
00142    13*          COMMON /MATRL2/ YIELD(NPMT),NODFRC(NFRC)                             000026
00142    13*      END                                                                      000026
00143    14*          DIMENSION MOVE1(4,3),MOVE2(4,3),MOVE3(4,3),MOVE4(4,3),MOVE5(4,3),    000026
00143    15*         1  MOVE6(4,3)                                                        000026
00144    16*          DATA MOVE1/1,4,5,8,1,7,13,19,1,10,21,30/                             000026
00146    17*          DATA MOVE2/1,2,5,6,1,3,13,15,1,4,21,24/                              000026
```

```
00150     18*           DATA MOVE3/1,2,3,4,1,3,5,7,1,4,7,10/                                  000026
00152     19*           DATA MOVE4/2,3,6,7,3,5,15,17,4,7,24,27/                               000026
00154     20*           DATA MOVE5/4,3,8,7,7,5,19,17,10,7,30,27/                              000026
00156     21*           DATA MOVE6/5,6,7,8,13,15,17,19,21,24,27,30/                           000026
00156     22*      C                                                                          000026
00156     23*      C                                                                          000026
00160     24*           IF (ITER .GT. 1) GO TO 50                                             000026
00162     25*           DO 40 N=1,NET                                                         000035
00165     26*           IF (N .GE. NEPJ .AND. N .LE. NEPJT) GO TO 40                          000035
00167     27*           DO 35 I=1,4                                                           000056
00172     28*           I1=MOVE1(I,IORDC)                                                     000056
00173     29*           I2=MOVE2(I,IORDC)                                                     000061
00174     30*           I3=MOVE3(I,IORDC)                                                     000065
00175     31*           I4=MOVE4(I,IORDC)                                                     000071
00176     32*           I5=MOVE5(I,IORDC)                                                     000075
00177     33*           I6=MOVE6(I,IORDC)                                                     000101
00200     34*           NJ1=NOD(N,I1)                                                         000105
00201     35*           NJ2=NOD(N,I2)                                                         000110
00202     36*           NJ3=NOD(N,I3)                                                         000113
00203     37*           NJ4=NOD(N,I4)                                                         000117
00204     38*           NJ5=NOD(N,I5)                                                         000123
00205     39*           NJ6=NOD(N,I6)                                                         000127
00206     40*           DIM1=ABS(XPRV(NJ1,1)-XPRV(NJ4,1))                                     000133
00207     41*           DIM2=ABS(XPRV(NJ2,2)-XPRV(NJ5,2))                                     000137
00210     42*           DIM3=ABS(XPRV(NJ3,3)-XPRV(NJ6,3))                                     000143
00211     43*           IF (DIM1 .LE. XDIM(1)) XDIM(1)=DIM1                                   000147
00213     44*           IF (DIM2 .LE. XDIM(2)) XDIM(2)=DIM2                                   000155
00215     45*           IF (DIM3 .LE. XDIM(3)) XDIM(3)=DIM3                                   000163
00217     46*        35 CONTINUE                                                              000176
00221     47*        40 CONTINUE                                                              000176
00223     48*        50 CONTINUE                                                              000176
00224     49*           DXMIN1=XDIM(1)/10.                                                    000176
00225     50*           DXMIN2=XDIM(2)/10.                                                    000200
00226     51*           DXMIN3=XDIM(3)/10.                                                    000203
00227     52*           DMIN=DXMIN1                                                           000206
00230     53*           IF (DMIN .GE. DXMIN2) DMIN=DXMIN2                                     000207
00232     54*           IF (DMIN .GE. DXMIN3) DMIN=DXMIN3                                     000215
00234     55*           IF (NFRAC .NE. 0) GO TO 110                                           000223
00234     56*      C                                                                          000223
00234     57*      C  TEST THE REBOUND CASE                                                   000223
00234     58*      C                                                                          000223
00236     59*           VMAX=0.0                                                              000225
00237     60*           VMIN=0.0                                                              000226
00240     61*           DO 100 I=1,NPS                                                        000233
00243     62*           DO 100 J=1,3                                                          000233
00246     63*           VNOD=ABS(VER(I,J))                                                    000233
00247     64*           IF (VNOD .GE. VMAX) VMAX=VNOD                                         000234
00251     65*           IF (VNOD .LE. VMIN) VMIN=VNOD                                         000242
00253     66*       100 CONTINUE                                                              000254
00256     67*           IF (VMAX .GE. VMAXPR .OR. VMIN .LE. VMINPR) GO TO 104                 000254
00260     68*           DIFF=VMAX-VMIN                                                        000272
00261     69*           IF (DIFF .GE. DIFFPR) GO TO 104                                       000275
00263     70*           DIFFPR=DIFF                                                           000300
00264     71*           VMAXPR=VMAX                                                           000302
00265     72*           VMINPR=VMIN                                                           000304
00266     73*           IF (DIFF .GT. 0.05) GO TO 104                                         000306
```

```
00270     74*            IF (DIFF .LE. 0.05) IDEX=2                                    000311
00272     75*            IF (DIFF .LE. 0.01) IDEX=3                                    000317
00274     76*            RETURN                                                        000325
00275     77*        104 CONTINUE                                                      000333
00276     78*            DO 105 J=1,INTFT                                              000333
00301     79*            I=INTF(J)                                                     000333
00302     80*            IF(VER(I,3) .LT. 0.0) GO TO 110                               000336
00304     81*        105 CONTINUE                                                      000347
00306     82*            DO 107 I=NPPJ,NPPJT                                           000347
00311     83*        107 KODE(I)=4                                                     000347
00313     84*            PRINT 1100                                                    000351
00315     85*            GO TO 200                                                     000355
00316     86*        110 CONTINUE                                                      000357
00316     87*        C                                                                 000357
00316     88*        C  TEST THE SLIDING CASE                                          000357
00316     89*        C                                                                 000357
00317     90*            IF (NSLIDE .EQ. 0) GO TO 300                                  000357
00321     91*            NVS=0                                                         000362
00322     92*            DO 118 N=1,NET                                                000367
00325     93*            IF (N .GE. NEPJ .AND. N .LE. NEPJT) GO TO 118                 000367
00327     94*            J3=MOVE3(1,IORDC)                                             000405
00330     95*            J6=MOVE6(1,IORDC)                                             000415
00331     96*            NJ3=NOD(N,J3)                                                 000425
00332     97*            NJ6=NOD(N,J6)                                                 000430
00333     98*            DO 115 I=1,INTFT                                              000442
00336     99*            NI=INTF(I)                                                    000442
00337    100*            IF(NI .EQ. NJ3) GO TO 111                                     000443
00341    101*            IF(NI .EQ. NJ6) GO TO 112                                     000445
00343    102*            GO TO 115                                                     000450
00344    103*        111 NJO=NJ3                                                       000452
00345    104*            JA=J3+IORDC                                                   000453
00346    105*            JB=J3+3*IORDC                                                 000455
00347    106*            GO TO 113                                                     000457
00350    107*        112 NJO=NJ6                                                       000461
00351    108*            JA=J6+IORDC                                                   000462
00352    109*            JB=J6+3*IORDC                                                 000464
00353    110*        113 NJA=NOD(N,JA)                                                 000467
00354    111*            NJB=NOD(N,JB)                                                 000473
00355    112*            XA1=X(NJA,1)-X(NJO,1)                                         000500
00356    113*            XA2=X(NJA,2)-X(NJO,2)                                         000505
00357    114*            XA3=X(NJA,3)-X(NJO,3)                                         000510
00360    115*            XB1=X(NJB,1)-X(NJO,1)                                         000513
00361    116*            XB2=X(NJB,2)-X(NJO,2)                                         000517
00362    117*            XB3=X(NJB,3)-X(NJO,3)                                         000522
00363    118*            XC1=XA2*XB3-XA3*XB2                                           000525
00364    119*            XC2=XA3*XB1-XA1*XB3                                           000531
00365    120*            XC3=XA1*XB2-XA2*XB1                                           000535
00366    121*            APB1=XA1+XB1                                                  000542
00367    122*            APB2=XA2+XB2                                                  000545
00370    123*            APB3=XA3+XB3                                                  000550
00371    124*            ABN=SQRT(XC1**2+XC2**2+XC3**2)                                000552
00372    125*            ABT=SQRT(APB1**2+APB2**2+APB3**2)                             000564
00373    126*            CN1=XC1/ABN                                                   000577
00374    127*            CN2=XC2/ABN                                                   000602
00375    128*            CN3=XC3/ABN                                                   000605
00376    129*            CT1=APB1/ABT                                                  000610
```

```
00377   130*          CT2=APB2/ABT                                          000613
00400   131*          CT3=APB3/ABT                                          000616
00401   132*          VT1=VER(NJO,1)/CT1                                    000621
00402   133*          VT2=VER(NJO,2)/CT2                                    000624
00403   134*          VT3=VER(NJO,3)/CT3                                    000627
00404   135*          VN1=VER(NJO,1)/CN1                                    000632
00405   136*          VN2=VER(NJO,2)/CN2                                    000635
00406   137*          VN3=VER(NJO,3)/CN3                                    000640
00407   138*          VT=SQRT(VT1**2+VT2**2+VT3**2)                         000643
00410   139*          VN=SQRT(VN1**2+VN2**2+VN3**2)                         000655
00411   140*          RV=VT/VN-0.01                                         000667
00412   141*          IF(RV .GT. SMIU) NVS=NVS+1                            000673
00414   142*      115 CONTINUE                                              000706
00416   143*      118 CONTINUE                                              000706
00420   144*          NCRIT=3*INTFT/4                                       000706
00421   145*          IF (NVS .LE. NCRIT) GO TO 300                         000712
00423   146*          VDIFF1=0.0                                            000715
00424   147*          VDIFF2=0.0                                            000716
00425   148*          N1=INTF(1)                                            000717
00426   149*          DO 120 I=2,INTFT                                      000721
00431   150*          N2=INTF(I)                                            000725
00432   151*          F1=VER(N1,1)*VER(N2,1)                                000730
00433   152*          F2=VER(N1,2)*VER(N2,2)                                000733
00434   153*          IF(F1 .LT. 0.0 .AND. F2 .LT. 0.0) GO TO 300           000736
00436   154*          DO 120 J=1,INTFT                                      000753
00441   155*          N3=INTF(J)                                            000753
00442   156*          V1=ABS(VER(N2,1)-VER(N3,1))/V0                        000756
00443   157*          V2=ABS(VER(N2,2)-VER(N3,2))/V0                        000763
00444   158*          IF(V1 .GE. VDIFF1) VDIFF1=V1                          000770
00446   159*          IF(V2 .GE. VDIFF2) VDIFF2=V2                          000776
00450   160*          IF(VDIFF1 .GE. 0.01 .AND. VDIFF2 .GE. 0.01) GO TO 300 001004
00452   161*      120 CONTINUE                                              001026
00455   162*          FNML=0.0                                              001026
00456   163*          FTGN=0.0                                              001027
00457   164*          DO 125 I=1,INTFT                                      001033
00462   165*          NI=INTF(I)                                            001033
00463   166*          FNML=FNML+STS(NI,3,3)                                 001036
00466   167*          DO 122 J=1,2                                          001045
00467   168*      122 FTGN=FTGN+STS(NI,J,3)                                 001045
00471   169*      125 CONTINUE                                              001052
00473   170*          FNML=ABS(FNML)                                        001052
00474   171*          FTGN=ABS(FTGN)                                        001054
00475   172*          RT=FTGN/FNML                                          001056
00476   173*          SUM=SMIU+0.01                                         001060
00477   174*          IF (RT .LE. SUM) GO TO 300                            001063
00501   175*          PRINT 1200                                            001066
00503   176*          IPLANE=4*IORDC                                        001072
00504   177*          NXMIN1=0                                              001075
00505   178*          NXMIN2=0                                              001076
00506   179*          DO 126 I=1,INTFT                                      001102
00511   180*          IN=INTF(I)                                            001102
00512   181*          XSLIP1=VER(IN,1)*DELT                                 001104
00513   182*          XSLIP2=VER(IN,2)*DELT                                 001107
00514   183*          IF (XSLIP1 .GE. DXMIN1) DXMIN1=XSLIP1                 001112
00516   184*          IF (XSLIP2 .GE. DXMIN2) DXMIN2=XSLIP2                 001120
00520   185*      126 CONTINUE                                              001130
```

```
00522   186*          IF (NSLIDE) 127,130,131                                  001130
00525   187*      127 PRINT 1201                                               001133
00527   188*          GO TO 200                                                001137
00530   189*      130 NSLIDE=1                                                 001141
00531   190*      131 CONTINUE                                                 001143
00531   191*        C                                                          001143
00531   192*        C  UPDATE MESH ON SLIDING BAND                             001143
00531   193*        C                                                          001143
00532   194*          INTEL=0                                                  001143
00533   195*          DO 135 N=1,NET                                           001147
00536   196*          INTNO=0                                                  001147
00537   197*          DO 134 I=1,NPE                                           001152
00542   198*          NI=NOD(N,I)                                              001152
00543   199*          DO 132 K=1,INTFT                                         001156
00546   200*          NK=INTF(K)                                               001156
00547   201*          IF (NI .NE. NK) GO TO 132                                001157
00551   202*          INTNO=INTNO+1                                            001161
00552   203*      132 CONTINUE                                                 001170
00554   204*      134 CONTINUE                                                 001170
00556   205*          IF (INTNO .NE. IPLANE) GO TO 135                         001170
00560   206*          INTEL=INTEL+1                                            001173
00561   207*          INTELM(INTEL)=N                                          001177
00562   208*      135 CONTINUE                                                 001207
00564   209*          DO 137 J=1,INTFT                                         001207
00567   210*          I=INTF(J)                                                001207
00570   211*          KODE(I)=-2                                               001211
00571   212*          X(I,1)=X(I,1)+VER(I,1)*DELT                              001213
00572   213*      137 X(I,2)=X(I,2)+VER(I,2)*DELT                              001217
00574   214*          DO 138 I=NPPJ,NPPJT                                      001230
00577   215*          X(I,1)=X(I,1)+VER(I,1)*DELT                              001230
00600   216*      138 X(I,2)=X(I,2)+VER(I,2)*DELT                              001233
00602   217*          DO 150 I=1,NFST                                          001243
00605   218*          NI=NFS(I)                                                001243
00606   219*          IF (NI .GE. NPPJ .AND. NI .LE. NPPJT) GO TO 150          001245
00610   220*          DO 148 J=1,INTFT                                         001265
00613   221*          INT=INTF(J)                                              001265
00614   222*          IF (NI .EQ. INT) GO TO 148                               001267
00616   223*          DX3 =ABS(X(NI,3)-X(INT,3))                               001271
00617   224*          IF (DX3 .GE. DXMIN3) GO TO 148                           001277
00621   225*          DX1 =ABS(X(NI,1)-X(INT,1))                               001302
00622   226*          DX2 =ABS(X(NI,2)-X(INT,2))                               001306
00623   227*          IF (DX1 .GE. DXMIN1) GO TO 145                           001312
00625   228*          DXMIN1=DX1                                               001315
00626   229*          NXMIN1=NXMIN1+1                                          001317
00627   230*          NF1=I                                                    001322
00630   231*          NFS1=NI                                                  001324
00631   232*          IN1=J                                                    001327
00632   233*          INT1=INT                                                 001331
00633   234*          IF (KODE(NFS1) .NE. 2) GO TO 145                         001333
00635   235*      144 PRINT 1202                                               001337
00637   236*          NSLIDE=2                                                 001342
00640   237*          GO TO 500                                                001344
00641   238*      145 CONTINUE                                                 001346
00642   239*          IF (DX2 .GE. DXMIN2) GO TO 148                           001346
00644   240*          DXMIN2=DX2                                               001351
00645   241*          NXMIN2=NXMIN2+1                                          001353
```

```
00646   242*            NF2=I                                                   001356
00647   243*            NFS2=NI                                                 001360
00650   244*            IN2=J                                                   001363
00651   245*            INT2=INT                                                001365
00652   246*            IF (KODE(NFS2) .NE. 2) GO TO 148                        001367
00654   247*            GO TO 144                                               001372
00655   248*        148 CONTINUE                                                001377
00657   249*        150 CONTINUE                                                001377
00661   250*            IF (NXMIN1 .EQ. 0 .AND. NXMIN2 .EQ. 0) GO TO 185        001377
00663   251*            NF=NF1                                                  001407
00664   252*            IN=IN1                                                  001411
00665   253*            NFDG=NFS1                                               001413
00666   254*            INDG=INT1                                               001415
00667   255*            IF (DXMIN1 .LE. DXMIN2) GO TO 152                       001417
00671   256*            NF=NF2                                                  001423
00672   257*            IN=IN2                                                  001425
00673   258*            NFDG=NFS2                                               001427
00674   259*            INDG=INT2                                               001431
00675   260*        152 CONTINUE                                                001436
00676   261*            DO 155 J=1,2                                            001436
00701   262*            II=IBDIST(J)                                            001442
00702   263*            IF (II .EQ. 0) GO TO 155                                001444
00704   264*            DO 153 I=1,II                                           001446
00707   265*            IE=IBDIS(I,J)                                           001452
00710   266*            IF (IE .EQ. NFDG) NSLIDE=-1                             001453
00712   267*        153 CONTINUE                                                001467
00714   268*        155 CONTINUE                                                001467
00716   269*            DO 160 N=1,NET                                          001467
00721   270*            DO 160 I=1,NPE                                          001476
00724   271*            NI=NOD(N,I)                                             001476
00725   272*            IF (NI .NE. INDG) GO TO 160                             001500
00727   273*            INEL=N                                                  001502
00730   274*            KPE=I                                                   001504
00731   275*            DO 156 M=1,NET                                          001515
00734   276*            DO 156 J=1,NPE                                          001515
00737   277*            MJ=NOD(M,J)                                             001515
00740   278*            IF (MJ .NE. NFDG) GO TO 156                             001517
00742   279*            IF (J .NE. KPE) GO TO 156                               001521
00744   280*            NFEL=M                                                  001524
00745   281*            GO TO 162                                               001526
00746   282*        156 CONTINUE                                                001543
00751   283*        160 CONTINUE                                                001543
00754   284*        162 NDIFEL=M-N                                              001543
00755   285*            DO 170 I=1,INTEL                                        001545
00760   286*            N=INTELM(I)                                             001553
00761   287*            M=N+NDIFEL                                              001555
00762   288*            DO 168 K=1,NPE                                          001560
00765   289*            NK=NOD(N,K)                                             001565
00766   290*            MK=NOD(M,K)                                             001567
00767   291*            IF (KODE(NK) .NE. -2) GO TO 168                         001572
00771   292*            X(MK,1)=X(NK,1)                                         001575
00772   293*            X(MK,2)=X(NK,2)                                         001577
00773   294*            DO 165 J=1,INTFT                                        001604
00776   295*            NJ=INTF(J)                                              001604
00777   296*            IF (NK .NE. NJ .OR. KODE(NJ) .NE. -2) GO TO 165         001605
01001   297*            INTF(J)=MK                                              001621
```

```
01002  298*            KODE(MK)=-5                                                  001623
01003  299*      165 CONTINUE                                                       001634
01005  300*      168 CONTINUE                                                       001634
01007  301*      170 CONTINUE                                                       001634
01011  302*          M=NEPJ-1                                                       001634
01012  303*          LIN=0                                                          001637
01013  304*          DO 172 I=1,INTEL                                               001643
01016  305*          N=INTELM(I)                                                    001643
01017  306*          M=M+1                                                          001644
01020  307*          DO 171 K=1,NPE                                                 001650
01023  308*          MK=NOD(M,K)                                                    001655
01024  309*          IF (KODE(MK) .NE. -2) GO TO 171                                001656
01026  310*          LIN=LIN+1                                                      001662
01027  311*          NK=NOD(N,K)                                                    001665
01030  312*          X(MK,1)=X(NK,1)                                                001667
01031  313*          X(MK,2)=X(NK,2)                                                001672
01032  314*          KODE(MK)=1                                                     001674
01033  315*          NDEG(LIN)=MK                                                   001676
01034  316*      171 CONTINUE                                                       001705
01036  317*      172 CONTINUE                                                       001705
01040  318*          LIAO=0                                                         001705
01041  319*          DO 173 I=1,NFST                                                001711
01044  320*          II=NFS(I)                                                      001711
01045  321*          IF (KODE(II) .NE. -5) GO TO 173                                001712
01047  322*          LIAO=LIAO+1                                                    001716
01050  323*          NFS(I)=NDEG(LIAO)                                              001721
01051  324*      173 CONTINUE                                                       001726
01053  325*          M=NEPJ-1                                                       001726
01054  326*          DO 178 I=1,INTEL                                               001734
01057  327*          M=M+1                                                          001734
01060  328*          N=INTELM(I)                                                    001737
01061  329*          DO 175 J=1,NPE                                                 001741
01064  330*          MJ=NOD(M,J)                                                    001752
01065  331*          IF (MJ .GE. NPPJ) GO TO 175                                    001753
01067  332*          DO 174 K=1,NPE                                                 001762
01072  333*          NK=NOD(N,K)                                                    001762
01073  334*          IF(MJ .NE. NK) GO TO 174                                       001764
01075  335*          MN=N+NOFFEL                                                    001766
01076  336*          LK=NOD(MN,K)                                                   001770
01077  337*          NOD(M,J)=LK                                                    001775
01100  338*      174 CONTINUE                                                       002006
01102  339*      175 CONTINUE                                                       002006
01104  340*      178 CONTINUE                                                       002006
01106  341*          IF (NSLIDE .GE. 0) GO TO 185                                   002006
01110  342*          DO 182 J=1,2                                                   002014
01113  343*          IF(IBDIST(J) .EQ. 0) GO TO 182                                 002023
01115  344*          II=IBDIST(J)                                                   002025
01116  345*          DO 180 I=1,II                                                  002027
01121  346*          NI=IBDIS(I,J)                                                  002034
01122  347*      180 X(NI,J)=XPRV(NI,J)                                             002037
01124  348*      182 CONTINUE                                                       002047
01126  349*      185 DO 190 I=1,NPS                                                 002047
01131  350*          IF (KODE(I) .LT. 2) KODE(I)=1                                  002047
01133  351*      190 CONTINUE                                                       002056
01135  352*          INDEX=1                                                        002056
01136  353*          RETURN                                                         002060
```

```
01137   354*      200 CONTINUE                                                  002064
01137   355*        C                                                           002064
01137   356*        C  UPDATE DATA DUE TO REBOUND OR SLIDING                    002064
01137   357*        C                                                           002064
01140   358*            DO 215 I=1,NPS                                          002064
01143   359*          215 KODE(NI)=3                                            002071
01145   360*            IOMIT=NPPJT-NPPJ+1                                      002073
01146   361*            DO 218 J=1,3                                            002077
01151   362*            M=IBDIST(J)                                             002107
01152   363*            IBDIST(J)=M+IOMIT                                       002111
01153   364*            N=NPPJ-1                                                002113
01154   365*            DO 218 I=1,IOMIT                                        002117
01157   366*            N=N+1                                                   002117
01160   367*            M=M+1                                                   002121
01161   368*            IBDIS(M,J)=N                                            002126
01162   369*            DISBD(M,J)=0.0                                          002127
01163   370*          218 CONTINUE                                              002136
01166   371*            DO 220 J=1,IU                                           002136
01171   372*            M=IBDVT(J)                                              002147
01172   373*            IBDVT(J)=M+IOMIT                                        002151
01173   374*            N=NPPJ-1                                                002153
01174   375*            DO 220 I=1,IOMIT                                        002157
01177   376*            N=N+1                                                   002157
01200   377*            M=M+1                                                   002161
01201   378*            IBDV(M,J)=N                                             002164
01202   379*            VBD(M,J)=0.0                                            002167
01203   380*            IF (J .EQ. 5) VBD(M,J)=VERPRV(N,J)                      002170
01205   381*          220 CONTINUE                                              002214
01210   382*            DO 225 J=1,3                                            002214
01213   383*            DO 225 K=1,3                                            002214
01216   384*            M=IBDST(J,K)                                            002222
01217   385*            IBDST(J,K)=M+IOMIT                                      002224
01220   386*            N=NPPJ-1                                                002226
01221   387*            DO 225 I=1,IOMIT                                        002232
01224   388*            N=N+1                                                   002232
01225   389*            M=M+1                                                   002234
01226   390*            IBDS(M,J,K)=N                                           002241
01227   391*            SBD(M,J,K)=0.0                                          002242
01230   392*          225 CONTINUE                                              002252
01234   393*            RETURN                                                  002252
01235   394*          300 CONTINUE                                              002263
01235   395*        C                                                           002263
01235   396*        C  TEST THE PENETRATION CASE                                002263
01235   397*        C                                                           002263
01236   398*            DO 350 N=1,NET                                          002263
01241   399*            DO 330 I=1,NPE                                          002263
01244   400*            NT=I                                                    002263
01245   401*            ND=NOD(N,NT)                                            002267
01246   402*            IF (KODE(ND) .EQ. 0) GO TO 330                          002272
01250   403*            IF (KODE(ND) .EQ. 4) GO TO 330                          002274
01252   404*            MJ1=0                                                   002277
01253   405*            DO 302 J=1,4                                            002303
01256   406*            IF (NT .EQ. MOVE1(J,IORDC)) MJ1=MOVE4(J,IORDC)          002303
01260   407*            IF (NT .EQ. MOVE4(J,IORDC)) MJ1=MOVE1(J,IORDC)          002307
01262   408*          302 CONTINUE                                              002316
01264   409*            MJ2=0                                                   002316
```

```
01265   410*          DO 303 J=1,4                                                  002322
01270   411*          IF (NT .EQ. MOVE2(J,IORDC)) MJ2=MOVE5(J,IORDC)                002322
01272   412*          IF (NT .EQ. MOVE5(J,IORDC)) MJ2=MOVE2(J,IORDC)                002326
01274   413*      303 CONTINUE                                                      002335
01276   414*          MJ3=0                                                         002335
01277   415*          DO 304 J=1,4                                                  002341
01302   416*          IF (NT .EQ. MOVE3(J,IORDC)) MJ3=MOVE6(J,IORDC)                002341
01304   417*          IF (NT .EQ. MOVE6(J,IORDC)) MJ3=MOVE3(J,IORDC)                002345
01306   418*      304 CONTINUE                                                      002354
01310   419*          NTX=0                                                         002354
01311   420*          IF (MJ1 .EQ. 0) GO TO 310                                     002355
01313   421*          NJ1=NOD(N,MJ1)                                                002357
01314   422*          DF1=ABS(XN(ND,1)-XN(NJ1,1))                                   002365
01315   423*          IF (DF1 .LT. DMIN) NTX=1                                      002371
01317   424*      310 NTY=0                                                         002400
01320   425*          IF (MJ2 .EQ. 0) GO TO 314                                     002401
01322   426*          NJ2=NOD(N,MJ2)                                                002403
01323   427*          DF2=ABS(XN(ND,2)-XN(NJ2,2))                                   002411
01324   428*          IF (DF2 .LT. DMIN) NTY=1                                      002415
01326   429*      314 NTZ=0                                                         002424
01327   430*          IF (MJ3 .EQ. 0) GO TO 317                                     002425
01331   431*          NJ3=NOD(N,MJ3)                                                002427
01332   432*          DF3=ABS(XN(ND,3)-XN(NJ3,3))                                   002435
01333   433*          IF (DF3 .LT. DMIN) NTZ=1                                      002441
01335   434*      317 CONTINUE                                                      002452
01336   435*          DO 318 K=1,NFST                                               002452
01341   436*          NK=NFS(K)                                                     002452
01342   437*          IF (NK .EQ. ND .AND. KODE(NK) .NE. 2) GO TO 319               002455
01344   438*      318 CONTINUE                                                      002472
01346   439*          GO TO 330                                                     002472
01347   440*      319 CONTINUE                                                      002474
01350   441*          IF (NTX .NE. 0) GO TO 320                                     002474
01352   442*          IF (NTY .NE. 0) GO TO 321                                     002475
01354   443*          IF (NTZ .NE. 0) GO TO 322                                     002477
01356   444*          GO TO 330                                                     002501
01357   445*      320 IF (DF1 .GT. DF2) GO TO 321                                   002503
01361   446*          IF (DF1 .GT. DF3) GO TO 322                                   002506
01363   447*          NJ=NJ1                                                        002512
01364   448*          GO TO 323                                                     002514
01365   449*      321 IF (DF2 .GT. DF3) GO TO 322                                   002516
01367   450*          NJ=NJ2                                                        002521
01370   451*          GO TO 323                                                     002523
01371   452*      322 NJ=NJ3                                                        002525
01372   453*      323 CONTINUE                                                      002527
01373   454*          IF (KODE(NJ) .EQ. -1) GO TO 330                               002530
01375   455*          IF (KODE(NJ) .NE. 2) GO TO 325                                002533
01377   456*          KODE(NJ)=4                                                    002536
01400   457*          KODE(ND)=4                                                    002540
01401   458*          GO TO 327                                                     002541
01402   459*      325 CONTINUE                                                      002543
01403   460*          KODE(NJ)=-1                                                   002544
01404   461*          KODE(ND)=4                                                    002546
01405   462*          DO 326 K=1,NFST                                               002553
01410   463*          NK=NFS(K)                                                     002553
01411   464*          IF (NK .NE. ND) GO TO 326                                     002554
01413   465*          NFS(K)=NJ                                                     002556
```

```
01414    466*        326 CONTINUE                                                  002563
01416    467*        327 CONTINUE                                                  002563
01417    468*            DO 328 K=1,3                                              002563
01422    469*        328 X(ND,K)=XPRV(ND,K)                                        002570
01424    470*        330 CONTINUE                                                  002601
01426    471*        350 CONTINUE                                                  002601
01430    472*            DO 380 I=1,NPS                                            002601
01433    473*            IF (KODE(I) .EQ. -1) KODE(I)=1                            002604
01435    474*            IF (KODE(I) .NE. 4) GO TO 380                             002611
01437    475*            DO 355 J=1,IU                                             002620
01442    476*            M=IBDVT(J)                                                002620
01443    477*            IBDVT(J)=M+1                                              002622
01444    478*            MJ=IBDVT(J)                                               002624
01445    479*            IBDV(MJ,J)=I                                              002630
01446    480*            VBD(MJ,J)=0.0                                             002632
01447    481*            IF (J .EQ. 5) VBD(MJ,J)=VERPRV(I,J)                       002634
01451    482*        355 CONTINUE                                                  002651
01453    483*            DO 358 J=1,3                                              002651
01456    484*            DO 358 K=1,3                                              002657
01461    485*            M=IBDST(J,K)                                              002657
01462    486*            IBDST(J,K)=M+1                                            002661
01463    487*            MJ=IBDST(J,K)                                             002663
01464    488*            IBDS(MJ,J,K)=I                                            002671
01465    489*            SBD(MJ,J,K)=0.0                                           002673
01466    490*        358 CONTINUE                                                  002706
01471    491*            DO 360 J=1,3                                              002706
01474    492*            M=IBDIST(J)                                               002706
01475    493*            IBDIST(J)=M+1                                             002710
01476    494*            MJ=IBDIST(J)                                              002712
01477    495*            IBDIS(MJ,J)=I                                             002713
01500    496*             DISBD(MJ,J)=0.0                                          002720
01501    497*        360 CONTINUE                                                  002725
01503    498*        380 CONTINUE                                                  002725
01505    499*            IF (NFRAC .EQ. 0) GO TO 410                               002725
01507    500*            DO 397 I=1,NFRAC                                          002727
01512    501*            NI=NODFRC(I)                                              002734
01513    502*            IF (KODE(NI) .EQ. 4) GO TO 397                            002736
01515    503*            KODE(NI)=0                                                002741
01516    504*            DO 394 J=1,NFRT                                           002745
01521    505*            NJ=NFS(J)                                                 002745
01522    506*            IF (NI .EQ. NJ) KODE(NI)=1                                002746
01524    507*        394 CONTINUE                                                  002761
01526    508*        397 CONTINUE                                                  002761
01530    509*        410 RETURN                                                    002761
01531    510*        500 CONTINUE                                                  002764
01532    511*            STOP                                                      002764
01533    512*       1100 FORMAT (2X,'********** PROJECTILE REFLECTED'//)           003040
01534    513*       1200 FORMAT (2X,'********** PROJECTILE SLIDES'//)              003040
01535    514*       1201 FORMAT (2X,'***** PROJECTILE REACHES THE BOUNDARY.'//)    003040
01536    515*       1202 FORMAT (2X,'***** PROJECTILE SLIDES OUT OF IMPACT ZONE,'/2X,003040
01536    516*          1' REGENERATING MESH IS NEEDED TO PROCEED THE CALCULATIONS.') 003040
01537    517*            END                                                       003040

        END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
SLID                                                    DATE 010978
@HDG,P    SORT
```

```
SORT                                                                    DATE 010978

@FOR,S    SORT,SORT
HSA  E3 -01/09/78-23:44:36 (0,)

     SUBROUTINE SORT      ENTRY POINT 000140

     STORAGE USED: CODE(1) 000155; DATA(0) 000070; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000114 100L      0001   000025 106G      0001   000044 117G      0001   000056 123G      0000 I 000042 I
      0000   000047 INJP$     0000 I 000044 J         0000 I 000040 K         0000 I 000041 M         0000 I 000043 N
      0000 R 000045 TEMP      0000 I 000000 X

00101      1*          SUBROUTINE SORT (NODE,NY,NN,NPE,MM)                                          000025
00103      2*          DIMENSION X(32),NY(32),NODE(MM,NPE)                                          000025
00104      3*          INTEGER X                                                                    000025
00104      4*    C  PUT EXTERNAL NUMBERS IN TEMP ARRAY X                                            000025
00105      5*          DO 10 K=1,NPE                                                                000025
00110      6*          X(K)=NODE(NN,K)                                                              000025
00111      7*          NY(K)=K                                                                      000027
00112      8*       10 CONTINUE                                                                     000033
00114      9*          K=NPE                                                                        000033
00115     10*          M=NPE-1                                                                      000035
00116     11*          DO 100 I=1,M                                                                 000037
00121     12*          N=I+1                                                                        000044
00122     13*          DO 100 J=N,K                                                                 000047
00125     14*          IF(X(I).LT.X(J)) GO TO 100                                                   000056
00127     15*          TEMP=X(I)                                                                    000061
00130     16*          X(I)=X(J)                                                                    000064
00131     17*          X(J)=TEMP                                                                    000066
00132     18*          TEMP=NY(I)                                                                   000076
00133     19*          NY(I)=NY(J)                                                                  000101
00134     20*          NY(J)=TEMP                                                                   000103
00135     21*      100 CONTINUE                                                                     000121
00140     22*          RETURN                                                                       000121
00141     23*          END                                                                          000154

       END OF COMPILATION:      NO DIAGNOSTICS.

@HDG,P      STFG
```

```
@FOR,S    STFG,STFG
HSA  E3 -01/09/78-23:44:37 (1,).

     SUBROUTINE STIFFG    ENTRY POINT 001144

     STORAGE USED: CODE(1) 001262; DATA(0) 000076; BLANK COMMON(2) 000000

     COMMON BLOCKS:

      0003   ORTHPC 000060
      0004   PROJMT 000047
      0005   MATRL  000005

     EXTERNAL REFERENCES (BLOCK, NAME)

      0006   STIFF
      0007   NROC$
      0010   NIO1$
      0011   NIO2$
      0012   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000053 10L     0001   000222 100L    0001   000145 122G    0001   000146 125G    0001   000162 133G
      0001   000163 136G    0000   000011 1400F   0001   000164 141G    0001   000204 150G    0001   000205 153G
      0001   000206 156G    0001   000227 167G    0001   000230 171G    0001   000245 200G    0001   000503 200L
      0001   000246 202G    0001   000263 211G    0001   000264 213G    0001   000301 222G    0001   000302 224G
      0001   000641 230L    0001   000314 232G    0001   000315 235G    0001   000646 240L    0001   000426 247G
      0001   001120 250L    0001   000433 252G    0001   000434 255G    0001   000457 264G    0001   000464 267G
      0001   000465 272G    0001   000552 305G    0001   000553 310G    0001   000567 316G    0001   000570 321G
      0001   000571 324G    0001   000606 333G    0001   000631 341G    0001   000632 344G    0001   000653 356G
      0001   000654 360G    0001   000671 367G    0001   000672 371G    0001   000707 400G    0001   000710 402G
      0001   000725 411G    0001   000726 413G    0001   000740 421G    0001   000741 424G    0001   001030 436G
      0001   001035 441G    0001   001036 444G    0001   001056 453G    0001   001105 461G    0001   001106 464G
      0003 R 000055 CK      0000 R 000002 CNT1    0000 R 000007 CNT2    0003 R 000033 E       0000 R 000000 FACT1
      0000 R 000001 FACT2   0000 I 000003 I       0000 I 000006 IN      0000   000020 INJP$   0005 I 000004 ISOP
      0005 I 000003 ISOT    0000 I 000004 J       0000 I 000010 K       0003   000022 LI      0000 I 000005 N
      0005   000000 NFRAC   0004 R 000044 PCK     0004 R 000022 PE      0005 R 000002 PMIU    0003 R 000044 PNIU
      0004 R 000033 PPNIU   0004 R 000011 PSLC    0004 R 000000 PSLT    0003 R 000011 SLC     0003 R 000000 SLT
      0005 R 000001 TMIU

00101    1*          SUBROUTINE STIFFG(NPM,C,NP,NPT,INTF,INT,NPS,CF,CFP,ND,CINT,    000040
00101    2*         1  KODE,IUNIT)                                                   000040
00101    3*     C                                                                   000040
00101    4*     C                                                                   000040
00101    5*     C    GENERATE STIFFNESS COEFFICIENTS FOR ISOTROPIC OR ORTHOTROPIC   000040
00101    6*     C  MATERIALS                                                        000040
00101    7*     C                                                                   000040
00101    8*     C                                                                   000040
```

```
00103     9*          DIMENSION CF(ND,ND),      C(NPM,ND,ND),INTF(1),CINT(INT,ND,ND)                  000040
00103    10*         1  ,KODE(1),CFP(ND,ND)                                                           000040
00104    11*          COMMON /ORTHPC/ SLT(3,3),SLC(3,3),LI(3,3),E(3,3),PNIU(3,3),CK(3)                 000040
00105    12*          COMMON /PROJMT/ PSLT(3,3),PSLC(3,3),PE(3,3),PPNIU(3,3),PCK(3)                    000040
00106    13*          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                                         000040
00107    14*          FACT1=1.0                                                                        000040
00110    15*          FACT2=1.0                                                                        000042
00111    16*          IF (IUNIT .EQ. 1) GO TO 10                                                       000043
00113    17*          FACT1=6.895/100000.0                                                             000046
00114    18*          FACT2=6.895/100.0                                                                000050
00115    19*       10 CONTINUE                                                                         000053
00116    20*          IF(ISOT .NE. 0) GO TO 100                                                        000053
00120    21*          CNT1=2.0*TMIU                                                                    000054
00121    22*          DO 18 I=1,ND                                                                     000146
00124    23*          DO 18 J=1,ND                                                                     000146
00127    24*       18 CF(I,J)=CNT1                                                                     000146
00132    25*          DO 20 N=1,NPS                                                                    000164
00135    26*          DO 20 I=1,ND                                                                     000164
00140    27*          DO 20 J=1,ND                                                                     000164
00143    28*       20 C(N,I,J)=CNT1                                                                    000164
00147    29*          DO 30 IN=1,INT                                                                   000206
00152    30*          DO 30 I=1,ND                                                                     000206
00155    31*          DO 30 J=1,ND                                                                     000206
00160    32*       30 CINT(IN,I,J)=CNT1                                                                000206
00164    33*          GO TO 200                                                                        000220
00165    34*      100 READ 1400, ((SLT(I,J),J=1,3),I=1,3)                                              000222
00176    35*          READ 1400, ((SLC(I,J),J=1,3),I=1,3)                                              000237
00207    36*          READ 1400,   ((E(I,J),J=1,3),I=1,3)                                              000255
00220    37*          READ 1400,((PNIU(I,J),J=1,3),I=1,3)                                              000273
00231    38*          DO 105 I=1,3                                                                     000315
00234    39*          DO 105 J=1,3                                                                     000315
00237    40*          SLT(I,J)=SLT(I,J)*FACT1                                                          000315
00240    41*          SLC(I,J)=SLC(I,J)*FACT1                                                          000317
00241    42*          E(I,J)=E(I,J)*FACT2                                                              000322
00242    43*      105 CONTINUE                                                                         000332
00245    44*          CALL STIFF(CF,ND,E,PNIU,CK)                                                      000332
00246    45*          DO 110  N=1,NPS                                                                  000341
00251    46*          DO 110  I=1,ND                                                                   000434
00254    47*          DO 110  J=1,ND                                                                   000434
00257    48*      110 C(N,I,J)=CF(I,J)                                                                 000434
00263    49*          DO 115 IN=1,INT                                                                  000465
00266    50*          DO 115 I=1,ND                                                                    000465
00271    51*          DO 115 J=1,ND                                                                    000465
00274    52*      115 CINT(IN,I,J)=CF(I,J)                                                             000465
00300    53*      200 CONTINUE                                                                         000503
00301    54*          IF (ISOP .NE. 0) GO TO 240                                                       000503
00303    55*          CNT2=2.0*PMIU                                                                    000504
00304    56*          DO 208 I=1,ND                                                                    000553
00307    57*          DO 208 J=1,ND                                                                    000553
00312    58*      208 CFP(I,J)=CNT2                                                                    000553
00315    59*          DO 210 N=NP,NPT                                                                  000571
00320    60*          DO 210 I=1,ND                                                                    000571
00323    61*          DO 210 J=1,ND                                                                    000571
00326    62*      210 C(N,I,J)=CNT2                                                                    000571
00332    63*          DO 230 K=1,INT                                                                   000606
00335    64*          N=INTF(K)                                                                        000606
```

```
      STIFG                                                                    DATE 010978

00336     65*            IF(KODE(N) .NE. 0) GO TO 230                          000607
00340     66*            DO 220 I=1,ND                                         000632
00343     67*            DO 220 J=1,ND                                         000632
00346     68*        220 C(N,I,J)=CNT2                                         000632
00351     69*        230 CONTINUE                                              000642
00353     70*            RETURN                                                000642
00354     71*        240 READ 1400,((PSLT(I,J),J=1,3),I=1,3)                   000646
00365     72*            READ 1400,((PSLC(I,J),J=1,3),I=1,3)                   000663
00376     73*            READ 1400,  ((PE(I,J),J=1,3),I=1,3)                   000701
00407     74*            READ 1400,((PPNIU(I,J),J=1,3),I=1,3)                  000717
00420     75*            DO 241 I=1,3                                          000741
00423     76*            DO 241 J=1,3                                          000741
00426     77*            PSLT(I,J)=PSLT(I,J)*FACT1                             000741
00427     78*            PSLC(I,J)=PSLC(I,J)*FACT1                             000743
00430     79*            PE(I,J)=PE(I,J)*FACT2                                 000746
00431     80*        241 CONTINUE                                              000756
00434     81*            CALL STIFF(CFP,ND,PE,PPNIU,PCK)                       000756
00435     82*            DO 242  N=NP,NPT                                      000765
00440     83*            DO 242   I=1,ND                                       001036
00443     84*            DO 242   J=1,ND                                       001036
00446     85*        242 C(N,I,J)=CFP(I,J)                                     001036
00452     86*            DO 250 K=1,INT                                        001056
00455     87*            N=INTF(K)                                             001056
00456     88*            IF(KODE(N) .NE. 0) GO TO 250                          001057
00460     89*            DO 245 I=1,ND                                         001106
00463     90*            DO 245 J=1,ND                                         001106
00466     91*        245 C(N,I,J)=CFP(I,J)                                     001106
00471     92*        250 CONTINUE                                              001121
00473     93*       1400 FORMAT (8F10.2)                                       001121
00474     94*            RETURN                                                001121
00475     95*            END                                                   001261

        END OF COMPILATION:      NO  DIAGNOSTICS.

aHDG,P    STIF
```

```
aFOR,S    STIF,STIF
HSA  E3 -01/09/78-23:44:40 (0,)

     SUBROUTINE STIFF    ENTRY POINT 000327

     STORAGE USED; CODE(1) 000360; DATA(0) 000073; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

       0003   SQRT
       0004   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000055 111G      0001   000056 114G      0001   000114 123G      0001   000214 142G      0000 R 000013 A1
       0000 R 000014 A2        0000 R 000015 A3        0000 R 000007 H         0000 I 000005 I         0000   000023 INJP$
       0000 I 000010 I1        0000 I 000011 I2        0000 I 000006 J         0000 I 000012 K         0000 I 000000 MP
```

```
00101      1*          SUBROUTINE STIFF(CF,ND,E,PNIU,CK)                                  000056
00103      2*          DIMENSION CF(ND,ND),MP(5),E(3,3),PNIU(3,3),CK(3)                   000056
00104      3*          DATA MP/1,2,3,1,2/                                                 000056
00104      4*     C                                                                       000056
00106      5*          FNCT1(A,B,X,Y)=(1.0-A*B)*X/Y                                       000056
00107      6*          FNCT2(A,B,C,X,Y)=(A+B*C)*X/Y                                       000056
00107      7*     C                                                                       000056
00110      8*          DO 20 I=1,ND                                                       000056
00113      9*          DO 20 J=1,ND                                                       000056
00116     10*       20 CF(I,J)=0.0                                                        000056
00121     11*          H=1.0-PNIU(1,2)*PNIU(2,1)-PNIU(2,3)*PNIU(3,2)-PNIU(3,1)*PNIU(1,3)  000063
00121     12*         1  -2.0*PNIU(1,2)*PNIU(2,3)*PNIU(3,1)                              000063
00122     13*          DO 25 I=1,3                                                        000114
00125     14*          J=I+3                                                              000114
00126     15*          I1=MP(I+1)                                                         000117
00127     16*          I2=MP(I+2)                                                         000121
00130     17*          CF(I,I)=FNCT1(PNIU(I1,I2),PNIU(I2,I1),E(I,I),H)                    000123
00131     18*       25 CF(J,J)=E(I1,I2)                                                   000142
00133     19*          CF(1,2)=FNCT2(PNIU(2,1),PNIU(2,3),PNIU(3,1),E(1,1),H)              000154
00134     20*          CF(1,3)=FNCT2(PNIU(3,1),PNIU(2,1),PNIU(3,2),E(1,1),H)              000162
00135     21*          CF(2,3)=FNCT2(PNIU(3,2),PNIU(1,2),PNIU(3,1),E(2,2),H)              000171
00136     22*          CF(2,1)=CF(1,2)                                                    000177
00137     23*          CF(3,1)=CF(1,3)                                                    000201
00140     24*          CF(3,2)=CF(2,3)                                                    000203
00141     25*          DO 30 I=1,3                                                        000214
00144     26*          J=MP(I+1)                                                          000214
00145     27*          K=MP(I+2)                                                          000230
00146     28*          A1=(1.0+4.0*PNIU(I,J)-PNIU(I,K))*E(J,J)                            000241
00147     29*          A2=(1.0-PNIU(J,K))*E(I,I)                                          000247
00150     30*          A3=SQRT(E(I,I)*E(J,J)*(2.0+PNIU(I,J)+PNIU(I,K))*(2.0+PNIU(J,I)     000253
00150     31*         1  +PNIU(J,K)))                                                     000253
```

```
      STIF                                                                DATE 010978

   00151     32*        30 CK(I)=(A1+A2)/A3                                    000273
   00153     33*           RETURN                                              000300
   00154     34*           END                                                 000357

          END OF COMPILATION:        NO  DIAGNOSTICS.

   @HDG,P      STRN
```

```
 @FOR,S    STRN,STRN
 HSA  E3 -01/09/78-23:44:43 (3,)

      SUBROUTINE STRENG    ENTRY POINT 000440

      STORAGE USED: CODE(1) 000466; DATA(0) 000067; BLANK COMMON(2) 000000

       COMMON BLOCKS:

       0003   CELZ1  000015
       0004   NODEPJ 000003
       0005   MATRL  000005

      EXTERNAL REFERENCES (BLOCK, NAME)

       0006   SQRT
       0007   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000224 100L      0001   000073 117G      0001   000122 125G      0001   000123 130G      0001   000150 143G
       0001   000210 156G      0001   000211 161G      0001   000273 175G      0001   000230 200L      0001   000302 202G
       0001   000303 205G      0001   000330 220G      0001   000367 233G      0001   000370 236G      0001   000341 405L
       0001   000400 420L      0001   000163 58L       0001   000221 80L       0003   000001 A         0003   000002 ALPHA
       0000 R 000012 BETA      0003   000013 DELT      0000 R 000005 FJ2       0000 I 000006 I         0000   000016 INJP$
       0003 I 000011 INNER     0004 I 000002 INTFT     0003   000000 IORDC     0005 I 000004 ISOP      0005 I 000003 ISOT
       0003   000012 ITER      0000 I 000007 J         0000 I 000002 K         0000 I 000000 L         0000 I 000001 LP
       0000 I 000010 M         0000 I 000003 N         0003   000006 NBW       0003   000010 NEPJ      0004   000000 NEPJT
       0003   000003 NET       0005   000000 NFRAC     0003 I 000014 NFST      0003   000005 NHBW      0003 I 000007 NN
       0004 I 000001 NP        0003 I 000004 NPS       0005   000002 PMIU      0000 R 000011 RATIO     0005   000001 TMIU
       0000 R 000004 Y2

 00101      1*          SUBROUTINE STRENG(NPM,STS,Y,NFS,INTF,KODE)                                    000047
 00101      2*     C                                                                                  000047
 00101      3*     C                                                                                  000047
 00101      4*     C  VON MISES CRITERION FOR ISOTROPIC MATERIALS                                     000047
 00101      5*     C                                                                                  000047
 00101      6*     C                                                                                  000047
 00103      7*          DIMENSION STS(NPM,3,3),Y(1),INTF(1),KODE(1)                                   000047
 00103      8*         1  ,NFS(1)                                                                     000047
 00103      9*     C                                                                                  000047
 00104     10*          COMMON /CELZ1/ IORDC,A,ALPHA,NET,NPS,NHBW,NBW,NN,NEPJ,INNER,                  000047
 00104     11*         1  ITER,DELT,NFST                                                              000047
 00105     12*          COMMON /NODEPJ/ NEPJT,NP,INTFT                                                000047
 00106     13*          COMMON /MATRL/ NFRAC,TMIU,PMIU,ISOT,ISOP                                      000047
 00106     14*     C                                                                                  000047
 00106     15*     C                                                                                  000047
 00107     16*          IF (ISOP .NE. 0) RETURN                                                       000047
 00111     17*          IF (ISOT) 20,100,20                                                           000062
```

```
00114     18*          20 L=NN                                                  000064
00115     19*             LP=NP                                                 000066
00116     20*             DO 80 K=1,INTFT                                       000073
00121     21*             N=INTF(K)                                             000073
00122     22*             Y2=0.33333333*Y(N)*Y(N)                               000074
00123     23*             FJ2=0.0                                               000102
00124     24*             DO 50 I=1,3                                           000123
00127     25*             DO 50 J=1,3                                           000123
00132     26*          50     FJ2 = FJ2+0.50*STS(N,I,J)*STS(N,J,I)              000123
00135     27*             IF  (FJ2  .LE. Y2)  GO TO 80                          000137
00137     28*             IF (INNER .EQ. 1) GO TO 58                            000142
00141     29*             KODE(N)=0                                             000144
00142     30*             DO 55 I=1,NFST                                        000150
00145     31*             M=NFS(I)                                              000150
00146     32*             IF (N .EQ. M) KODE(N)=1                               000151
00150     33*          55 CONTINUE                                              000163
00152     34*          58 CONTINUE                                              000163
00153     35*             RATIO=SQRT(Y2/FJ2)                                    000166
00154     36*             BETA=1.0-RATIO                                        000176
00155     37*             DO 60 I=1,3                                           000200
00160     38*             DO 60 J=1,3                                           000211
00163     39*          60 STS(N,I,J)=(1.0-BETA)*STS(N,I,J)                      000211
00166     40*          80 CONTINUE                                              000222
00170     41*             GO TO 200                                             000222
00171     42*         100 L=1                                                   000224
00172     43*             LP=NPS                                                000225
00173     44*         200 CONTINUE                                              000230
00174     45*             DO 420 N=L,LP                                         000230
00177     46*             Y2=0.33333333*Y(N)*Y(N)                               000273
00200     47*             FJ2=0.0                                               000277
00201     48*             DO 400 I=1,3                                          000303
00204     49*             DO 400 J=1,3                                          000303
00207     50*         400     FJ2 = FJ2+0.50*STS(N,I,J)*STS(N,J,I)              000303
00212     51*             IF  (FJ2  .LE. Y2)  GO TO 420                         000317
00214     52*             IF (INNER .EQ. 1) GO TO 405                           000322
00216     53*             KODE(N)=0                                             000324
00217     54*             DO 402 I=1,NFST                                       000330
00222     55*             M=NFS(I)                                              000330
00224     56*             IF (N .EQ. M) KODE(N)=1                               000331
00225     57*         402 CONTINUE                                              000341
00227     58*         405 CONTINUE                                              000341
00230     59*             RATIO=SQRT(Y2/FJ2)                                    000345
00231     60*             BETA=1.0-RATIO                                        000355
00232     61*             DO 410 I=1,3                                          000357
00235     62*             DO 410 J=1,3                                          000370
00240     63*         410 STS(N,I,J)=(1.0-BETA)*STS(N,I,J)                      000370
00243     64*         420 CONTINUE                                              000412
00245     65*             RETURN                                                000412
00246     66*             END                                                   000465

      END OF COMPILATION:        NO  DIAGNOSTICS.

$HDG,P      SYM
```

Appendix B

NASTRAN Routines Changed

Table	Contents
B1	XSEMO7
B2	XBSBD
B3	LINK7
B4	INPUTT2
B5	OUTPUT2

```
SCHMTZBIN209*OBJ(0),XSEM07
      1      C-*********1*********2*********3*********4*********5*********6*********7**XSEM    2
      2      C THE PURPOSE OF THIS ROUTINE IS TO GET THE NEXT MODULE TO BE EXECUTED  XSEM    3
      3      C FROM THE OSCAR FILE, INITIALIZE THE MODULE AND CALL IT IF IT IS IN    XSEM    4
      4      C THIS LINK OR CALL THE LINK IN WHICH THE MODULE RESIDES.                XSEM    5
      5      C                                                                       XSEM    6
      6            INTEGER ANDF ,DATABF,ERRFLG,FIST  ,FISTNM,FSTRST,OPNTR ,ORF       XSEM    7
     7     1       ,PARML,PARAM ,PARMN ,POOL  ,REW   ,RSHIFT,SCRCHM,SCRTCH          XSEM    8
      8     2       ,VPS  ,VPARML,TYPECD,VPSX  ,WORDB ,WORDE ,PROGMI                 XSEM    9
      9     3       ,THISLK,EXIT,CORSZ,SYSBUF                                       XSEM   10
     10      C                                                                       XSEM   11
     11           DIMENSION SCRTCH(2),WORDB(4),WORDE(2),NUMBR(10)                    XSEM   12
     12           COMMON/SEM   /MASK  ,MASK2 ,MASK3 ,PROGMI(15)                      XSEM   13
     13           1                                                                  XSEM   14
     14           F       /SYSTEM/SYSBUF,XX(37),NBPC,NBPW,NCPW                       XSEM   15
     15           G                                                                  XSEM   16
     16           H       /XLINK /LXLINK,MAXLNK,MXLINK(1)                            XSEM   17
     17           1                                                                  XSEM   18
     18           2       /XFIST /FIST(1)                                            XSEM   19
     19           3                                                                  XSEM   20
     20           4       /XPFIST/FSTRST                                             XSEM   21
     21           5                                                                  XSEM   22
     22           6       /OSCENT/INOSCR(200)                                        XSEM   23
     23           7                                                                  XSEM   24
     24           8       /ESFA  /DATABF(1)                                          XSEM   25
     25           9                                                                  XSEM   26
     26           A       /      /PARAM(100)                                         XSEM   27
     27           B                                                                  XSEM   28
     28           C       /XVPS  /VPS(1)                                             XSEM   29
     29           D                                                                  XSEM   30
     30           E       /MSGX  /NMSG                                               XSEM   31
     31      C                                                                       XSEM   32
     32           DATA REW     /     1/,NOREW/     0/,POOL /4HPOOL/                  XSEM   33
     33     3,    SCRTCH /4HSCRA,4HTCH0/                                           XSEM   34
     34     4,   NUMBR/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9 /                   XSEM   35
     35     5,   WORDE     /4HBEGN,4HEND /                                         XSEM   36
     36     6,   IBLNK/4H    /                                                     XSEM   37
     37     6,   MODX   /  200/                                                    XSEM   38
     38     7,    EXIT/4HEXIT/                                                     XSEM   39
     39          DATA THISLK/7/                                                    XSEM   40
     40           DATA BCDZRO /1H0/                                                XSEM   41
     41      C*****                                                                 XSEM   42
     42      C INITIALIZE MACHINE DEPENDENT CONSTANTS                               XSEM   43
     43           CALL SETC(15)
     44           CALL BTSTRP                                                      XSEM   44
     45      C*****                                                                 XSEM   45
     46      C CALL BGNSYS TO INITIATE PROCESSING                                   XSEM   46
     47      C*****                                                                 XSEM   47
     48           KSCR= LSHIFT(1,NBPW-4*NBPC)                                      XSEM   48
     49           CALL BGNSYS                                                      XSEM   49
     50           IBUF1 = CORSZ(DATABF,PARAM)-SYSBUF                               XSEM   50
     51           GO TO 30                                                         XSEM   51
     52      C*****                                                                 XSEM   52
     53      C-RETURN HERE AFTER MODULE HAS EXECUTED                                XSEM   53
     54      C*****                                                                 XSEM   54
     55        10 IF(TYPECD .GT. 2 .AND. INOSCR(4) .NE. EXIT)  GO TO 20             XSEM   55
```

```
      CALL SSWTCH (6,L)                                                 XSEM  56
      WORDB(4) = WORDE(2)                                               XSEM  57
      CALL CONMSG(WORDB,4,L)                                            XSEM  58
   20 IF(NMSG .GT. 0) CALL MSGWRT                                       XSEM  59
      CALL OPEN($270,POOL,DATABF(IBUF1),2)                              XSEM  60
C*****                                                                  XSEM  61
C READ THE OSCAR ENTRY                                                  XSEM  62
C*****                                                                  XSEM  63
   30 CALL READ($280,$40,POOL,INOSCR,200,1,ERRFLG)                      XSEM  64
      GO TO 290                                                         XSEM  65
   40 IF (INOSCR(6))50,30,30                                            XSEM  66
C*****                                                                  XSEM  67
C TRY AGAIN IF EXECUTE FLAG IS OFF                                      XSEM  68
C*****                                                                  XSEM  69
   50 CALL CLOSE(POOL,2)                                                XSEM  70
      TYPECD= ANDF(INOSCR(3),MASK)                                      XSEM  71
C*****                                                                  XSEM  72
C NOW DETERMINE TYPE OR OSCAR FORMAT                                    XSEM  73
C*****                                                                  XSEM  74
      IF(TYPECD .GT. 2) GO TO 200                                       XSEM  75
C*****                                                                  XSEM  76
C*****                                                                  XSEM  77
C NOW PROCESSING TYPE 0 AND F                                           XSEM  78
C*****                                                                  XSEM  79
   60 MODNO= INOSCR(2)                                                  XSEM  80
      FIST(2)= FSTRST                                                   XSEM  81
      OPNTR = 7                                                         XSEM  82
      ASSIGN 110 TO MM                                                  XSEM  83
      FISTNM=101                                                        XSEM  84
C*****                                                                  XSEM  85
C PROCESS FILES IN OSCAR ENTRY.                                         XSEM  86
C*****                                                                  XSEM  87
   70 J=INOSCR(OPNTR)                                                   XSEM  88
      OPNTR=OPNTR+1                                                     XSEM  89
      IF(J.EQ.0) GO TO 100                                              XSEM  90
      DO 90 I=1,J                                                       XSEM  91
      CALL GNFIST(INOSCR(OPNTR),FISTNM,MODNO)                           XSEM  92
      IF(MODNO) 60,260,80                                               XSEM  93
   80 OPNTR= OPNTR+ 3                                                   XSEM  94
   90 FISTNM=FISTNM+1                                                   XSEM  95
  100 GO TO MM,(110,120)                                                XSEM  96
C*****                                                                  XSEM  97
C SETUP TO PROCESS OUTPUT FILES                                         XSEM  98
C*****                                                                  XSEM  99
  110 IF(TYPECD.EQ.2) GO TO 120                                         XSEM 100
      ASSIGN 120 TO MM                                                  XSEM 101
      FISTNM=201                                                        XSEM 102
      GO TO 70                                                          XSEM 103
C*****                                                                  XSEM 104
C PROCESS SCRATCH FILES                                                 XSEM 105
C*****                                                                  XSEM 106
  120 J1= INOSCR(OPNTR)                                                 XSEM 107
      IF(J1.EQ.0) GO TO 140                                             XSEM 108
      FISTNM=-301                                                       XSEM 109
      SCRCHM=LSHIFT(RSHIFT(BCDZRO,NBPW-NBPC),NBPW-4*NBPC)               XSEM 110
      DO 130 J=1,J1                                                     XSEM 111
```

```
112             SCRCHM= SCRCHM+KSCR                                     XSEM 112
113             SCRTCH(2)=ORF(ANDF(SCRTCH(2),MASK2),SCRCHM)             XSEM 113
114             CALL GNFIST(SCRTCH,FISTNM,HODNO)                        XSEM 114
115             IF(HODNO)60,260,130                                     XSEM 115
116         130 FISTNM=FISTNM+1                                         XSEM 116
117         140 OPNTR=OPNTR+1                                           XSEM 117
118       C*****                                                        XSEM 118
119       C NOW PROCESS PARAMETER LIST IN OSCAR                         XSEM 119
120       C  PARMN = NO. OF PARAMETERS TO PROCESS                       XSEM 120
121       C*****                                                        XSEM 121
122             PARMN=INOSCR(OPNTR)                                     XSEM 122
123             IF(PARMN.EQ.0) GO TO 200                                XSEM 123
124             II=1                                                    XSEM 124
125             OPNTR=OPNTR+1                                           XSEM 125
126             DO 190 J2=1,PARMN                                       XSEM 126
127             IF(INOSCR(OPNTR))170,150,150                            XSEM 127
128       C*****                                                        XSEM 128
129       C NOW PROCESS CONSTANT PARAMETER                              XSEM 129
130       C*****                                                        XSEM 130
131         150 PARML=INOSCR(OPNTR)                                     XSEM 131
132             OPNTR=OPNTR+1                                           XSEM 132
133             DO 160 J3=1,PARML                                       XSEM 133
134             PARAM(II)=INOSCR(OPNTR)                                 XSEM 134
135             II=II+1                                                 XSEM 135
136         160 OPNTR=OPNTR+1                                           XSEM 136
137             GO TO 190                                               XSEM 137
138       C*****                                                        XSEM 138
139       C MOVE VARIABLE INTO COMMON VIA VPS TABLE                     XSEM 139
140       C*****                                                        XSEM 140
141         170 VPSX=ANDF(INOSCR(OPNTR),MASK3)                          XSEM 141
142             OPNTR=OPNTR+1                                           XSEM 142
143             VPARML=VPS(VPSX-1)                                      XSEM 143
144             DO 180 J5=1,VPARML                                      XSEM 144
145             PARAM(II)=VPS(VPSX)                                     XSEM 145
146             II=II+1                                                 XSEM 146
147         180 VPSX=VPSX+1                                             XSEM 147
148         190 CONTINUE                                                XSEM 148
149       C*****                                                        XSEM 149
150       C USE LINK SPECIFICATION TABLE (MXLINK) TO DETERMINE IF MODULE RESIDES  XSEM 150
151       C IN THIS LINK.                                               XSEM 151
152       C MODX = INDEX INTO MXLINK                                    XSEM 152
153       C*****                                                        XSEM 153
154         200 MODX = RSHIFT(INOSCR(3),16)                             XSEM 154
155             IF(ANDF(MXLINK(MODX),LSHIFT(1,THISLK-1)) .NE. 0)GO TO 280  XSEM 155
156       C MODULE IS NOT IN THIS LINK - DETERMINE CLOSEST LINK WHERE IT RESIDES.  XSEM 156
157             J = 2*MAXLNK+1                                          XSEM 157
158             DO 210 I=1,MAXLNK                                       XSEM 158
159             IF(ANDF(MXLINK(MODX),LSHIFT(1,I-1)).EQ. 0)GO TO 210     XSEM 159
160       C MODULE IS IN LINK I - SEE IF LINK I IS CLOSER THAN LINK J.  XSEM 160
161             IF(MIND(IABS(THISLK-J),IABS(THISLK-I)) .EQ. IABS(THISLK-J))  XSEM 161
162            1  GO TO 220                                             XSEM 162
163       C LINK I IS CLOSER THAN J - MAKE LINK J CLOSEST LINK FOUND SO FAR.  XSEM 163
164             J = I                                                   XSEM 164
165         210 CONTINUE                                                XSEM 165
166             I = 2*MAXLNK                                            XSEM 166
167         220 IF(J .GT. MAXLNK) GO TO 940                             XSEM 167
```

```
      C CALL ENDSYS TO GET LINK J                                       XSEM 168
      C CHECK FOR LINK I BEING SAME DISTANCE AS J FROM THISLK, IF IT IS GO TO XSEM 169
      C LINK I SINCE LINK TAPE SHOULD BE MOVED IN FORWARD DIRECTION IF POSSIBLXSEM 170
            IF(THISLK-J.EQ.I-THISLK) J=I                                XSEM 171
            NAME = PROGMI(J)                                            XSEM 172
            IF(THISLK - J) 230,940,240                                  XSEM 173
      C LINK J IS AFTER THIS LINK                                       XSEM 174
      230   CALL ENDSYS(NAME,NOREW)                                     XSEM 175
            STOP                                                        XSEM 176
      C*****                                                            XSEM 177
      C LINK J IS BEFORE THIS LINK                                      XSEM 178
      240   CALL ENDSYS(NAME,REW)                                       XSEM 179
            STOP                                                        XSEM 180
      C*****                                                            XSEM 181
      C MODULE IS IN THIS LINK                                          XSEM 182
      C PRINT TIME MODULE BEGAN EXECUTION IF FUNCTIONAL MODULE          XSEM 183
      C*****                                                            XSEM 184
      250   IF(TYPECD .GT. 2 .AND. (NOSCR(4) .NE. EXIT)) GO TO 1000     XSEM 185
            WORDB(1) = IBLNK                                            XSEM 186
            WORDB(2) = INOSCR(4)                                        XSEM 187
            WORDB(3) = INOSCR(5)                                        XSEM 188
            WORDB(4) = WORDE(1)                                         XSEM 189
      C                                                                 XSEM 190
      C     EXTRACT DMAP SEQUENCE NUMBER                                XSEM 191
      C                                                                 XSEM 192
            IDIN = ANDF(INOSCR(6),MASK)                                 XSEM 193
            DO 251  I =1,4                                              XSEM 194
            ICHR = IDIN-(IDIN/10)*10 +1                                 XSEM 195
            L = NBPW-NBPC                                               XSEM 196
            WORDB(I) = ORF(RSHIFT(WORDB(I),NBPC),LSHIFT(RSHIFT(NUMBR(ICHR),L),XSEM 197
           1  L))                                                       XSEM 198
            IDIN = IDIN/10                                              XSEM 199
            IF(IDIN .EQ. 0)  GO TO 252                                  XSEM 200
      251   CONTINUE                                                    XSEM 201
      252   CONTINUE                                                    XSEM 202
            CALL SSWTCH(5,L)                                            XSEM 203
            CALL CONMSG(WORDB,4,L)                                      XSEM 204
            GO TO 1000                                                  XSEM 205
      C*****                                                            XSEM 206
      C                    E R R O R   M E S S A G E S                  XSEM 207
      C*****                                                            XSEM 208
      C MODULE REQUIREMENTS EXCEED AVAILABLE FILES.                     XSEM 209
        260 INOSCR(6) = ANDF(INOSCR(6),MASK)                            XSEM 210
            CALL MESAGE(-18,INOSCR(6),INOSCR(4))                        XSEM 211
      C                                                                 XSEM 212
      C UNEXPECTED ALTERNATE RETURN TAKEN WHILE ATTEMPTING TO OPEN POOL TAPE.XSEM 213
        270 CONTINUE                                                    XSEM 214
      C                                                                 XSEM 215
      C OSCAR FILE POSITIONED INCORRECTLY - HIT EOF.                    XSEM 216
      280   CONTINUE                                                    XSEM 217
      C                                                                 XSEM 218
      C OSCAR RECORD TOO LARGE FOR /OSCENT/                             XSEM 219
        290 CONTINUE                                                    XSEM 220
      C                                                                 XSEM 221
      C LINK SPECIFICATIONS INCORRECT FOR THIS MODULE.                  XSEM 222
      940   CALL MESAGE(-61,0,0)                                        XSEM 223
```

```
      C*******************************************************************  XSEM 222
      C*******************************************************************  XSEM 223
      C EXECUTE MODULE                                                       XSEM 224
 1000 IF(MODX.GT.  35) GO TO 1010                                            XSEM 225
           I = MODX -   2                                                    XSEM 226
           GO TO( 2003                                                       XSEM 227
          1,    940,2005,2006,2007,2008,2009,2010,2011,2012,2013,2014, 940   XSEM 228
          2,    940, 940,2018,2019, 940, 940, 940,2023, 940, 940, 940,2027   XSEM 229
          3,   2028,2029,2030, 940, 940,2033),I                              XSEM 230
 1010 IF(MODX.GT.50) GO TO 1020
           I=MODX-40
           GO TO (2041
          1, 940
          1, 940
          1, 940
          1, 940
          1, 2046
          1, 940
          1, 940
          1, 940
          1, 2050
          1),I
 1020 IF(MODX .GT.  59) GO TO 1030                                           XSEM 233
           I = MODX -  53                                                    XSEM 234
           GO TO( 2054                                                       XSEM 235
          1, 2055                                                            XSEM 236
          1,  940                                                            XSEM 237
          1, 2057                                                            XSEM 238
          1, 2058                                                            XSEM 239
          1, 2059                                                            XSEM 240
          1),I                                                               XSEM 241
 1030 IF(MODX.GT.  70) GO TO 1040                                            XSEM 242
           I=MODX-63
           GO TO (2064
          1, 940
          1, 940
          1, 2067
          1, 2068                                                            XSEM 245
          1, 2069                                                            XSEM 246
          1, 2070                                                            XSEM 247
          1),I                                                               XSEM 248
 1040 IF(MODX.GT. 103) GO TO 1050                                            XSEM 249
           I = MODX -  99                                                    XSEM 250
           GO TO( 2100                                                       XSEM 251
          1, 2101                                                            XSEM 252
          1),I                                                               XSEM 253
 1050 IF(MODX .GT. 120) GO TO  940                                           XSEM 254
           I = MODX - 115                                                    XSEM 255
           GO TO( 2116                                                       XSEM 256
          1, 2117                                                            XSEM 257
          1, 2118                                                            XSEM 258
          1,  940                                                            XSEM 259
          1, 2120                                                            XSEM 260
          1),I                                                               XSEM 261
 2003 CALL XCHK                                                              XSEM 262
      GO TO 10                                                               XSEM 263
```

```
 280        2005 CALL XCEI                                                XSEM 264
 281             GO TO 10                                                 XSEM 265
 282        2006 CALL XCEI                                                XSEM 266
 283             GO TO 10                                                 XSEM 267
 284        2007 CALL XCEI                                                XSEM 268
 285             GO TO 10                                                 XSEM 269
 286        2008 CALL XSAVE                                               XSEM 270
 287             GO TO 10                                                 XSEM 271
 288        2009 CALL XPURGE                                              XSEM 272
 289             GO TO 10                                                 XSEM 273
 290        2010 CALL XEQUIV                                              XSEM 274
 291             GO TO 10                                                 XSEM 275
 292        2011 CALL XCEI                                                XSEM 276
 293             GO TO 10                                                 XSEM 277
 294        2012 CALL XCEI                                                XSEM 278
 295             GO TO 10                                                 XSEM 279
 296        2013 CALL DADD                                                XSEM 280
 297             GO TO 10                                                 XSEM 281
 298        2014 CALL DADD5                                               XSEM 282
 299             GO TO 10                                                 XSEM 283
 300        2018 CALL CYCT1                                               XSEM 284
 301             GO TO 10                                                 XSEM 285
 302        2019 CALL CYCT2                                               XSEM 286
 303             GO TO 10                                                 XSEM 287
 304        2023 CALL DDCOMP                                              XSEM 288
 305             GO TO 10                                                 XSEM 289
 306        2027 CALL DUMOD1                                              XSEM 290
 307             GO TO 10                                                 XSEM 291
 308        2028 CALL DUMOD2                                              XSEM 292
 309             GO TO 10                                                 XSEM 293
 310        2029 CALL DUMOD3                                              XSEM 294
 311             GO TO 10                                                 XSEM 295
 312        2030 CALL DUMOD4                                              XSEM 296
 313             GO TO 10                                                 XSEM 297
 314        2033 CALL DFBS                                                XSEM 298
 315             GO TO 10                                                 XSEM 299
 316        2041 CALL GPCYC                                               XSEM 300
 317             GO TO 10                                                 XSEM 301
 318        2046 CALL INPTT2
 319             GO TO 10
 320        2050 CALL MATPRN
 321             GO TO 10
 322        2054 CALL MERGE1                                              XSEM 302
 323             GO TO 10                                                 XSEM 303
 324        2055 CALL MODA                                                XSEM 304
 325             GO TO 10                                                 XSEM 305
 326        2057 CALL MODB                                                XSEM 306
 327             GO TO 10                                                 XSEM 307
 328        2058 CALL MODC                                                XSEM 308
 329             GO TO 10                                                 XSEM 309
 330        2059 CALL OMPYAD                                              XSEM 310
 331             GO TO 10                                                 XSEM 311
 332        2064 CALL OUTPT2
 333             GO TO 10
 334        2067 CALL QPARAM                                              XSEM 312
 335             GO TO 10                                                 XSEM 313
```

```
 336        2068 CALL PARAML                                          XSEM 314
 337             GO TO 10                                             XSEM 315
 338        2069 CALL QPARMR                                          XSEM 316
 339             GO TO 10                                             XSEM 317
 340        2070 CALL PARTNI                                          XSEM 318
 341             GO TO 10                                             XSEM 319
 342        2100 CALL SMPYAD                                          XSEM 320
 343             GO TO 10                                             XSEM 321
 344        2101 CALL SOLVE                                           XSEM 322
 345             GO TO 10                                             XSEM 323
 346        2116 CALL DTRANP                                          XSEM 324
 347             GO TO 10                                             XSEM 325
 348        2117 CALL DUMERG                                          XSEM 326
 349             GO TO 10                                             XSEM 327
 350        2118 CALL DUPART                                          XSEM 328
 351             GO TO 10                                             XSEM 329
 352        2120 CALL XEC                                             XSEM 330
 353             GO TO 10                                             XSEM 331
 354             END                                                  XSEM 332
```

DPRT,S OBJ,XBSBD

```
SCHHTZBIN209*08J(0),XBSBD
      C     BLOCK DATA XBSBD                                                00000010
            BLOCK DATA                                                      00000020
      C*****************************************************************  00000030
      C           L I N K    S P E C I F I C A T I O N    T A B L E         00000040
      C A LINK TABLE ENTRY CONTAINS AN EXECUTABLE DMAP INSTRUCTION NAME, ITS 00000050
      C CORRESPONDING SUBROUTINE ENTRY POINT NAME AND MACHINE DEPENDENT      00000060
      C LINK FLAG WORDS.                                                    00000070
      C EACH BIT IN THE LINK FLAG SPECIFIES A LINK NUMBER. BIT 1(RIGHT MOST) 00000080
      C SPECIFIES LINK ONE, BIT 2 SPECIFIES LINK 2, ETC.                    00000090
      C BIT ON SPECIFIES MODULE IS IN THAT LINK, BIT OFF MEANS IT IS NOT.    00000100
      C EXAMPLE - SUPPOSE MODULE X IS IN LINKS 2,4 AND 5. ITS LINK FLAG=32(8),00000110
      C                                                                     00000120
      C     L E V E L  5  L I N K  S P E C  T A B L E                       00000130
      C                                                                     00000140
      C LLINK = LENGTH OF LINK TABLE.                                       00000150
      C*****************************************************************  00000160
      C                                                                     00000170
      C SET SENSE SWITCH 8 TO GENERATE ALL FORTRAN CODE BELOW.              00000180
      C                                                                     00000190
      C*****************************************************************  00000200
      C     LINK SPEC. TABLE FOR SUBROUTINE XBSBD
            DIMENSION LINK (1144)
           1,        LINK1( 144)
           1,        LINK2( 144)
           1,        LINK3( 144)
           1,        LINK4( 144)
           1,        LINK5( 144)
           1,        LINK6( 144)
           1,        LINK7( 144)
           1,        LINK8( 136)
            COMMON/XLKSPC/LLINK,LINK
            EQUIVALENCE (LINK(   1),LINK1(1))
           1,          (LINK( 145),LINK2(1))
           1,          (LINK( 289),LINK3(1))
           1,          (LINK( 433),LINK4(1))
           1,          (LINK( 577),LINK5(1))
           1,          (LINK( 721),LINK6(1))
           1,          (LINK( 865),LINK7(1))
           1,          (LINK(1009),LINK8(1))
            DATA LLINK/1144/
            DATA LINK1/
           1 4HCHKP,4HNT  ,  4HXCHK,4H    ,  16383,  16383,  16383,  16383
           1,4HREPT,4H    ,  4HXCEI,4H    ,  16383,  16383,  16383,  16383
           1,4HJUMP,4H    ,  4HXCEI,4H    ,  16383,  16383,  16383,  16383
           1,4HCOND,4H    ,  4HXCEI,4H    ,  16383,  16383,  16383,  16383
           1,4HSAVE,4H    ,  4HXSAV,4HE   ,  16382,  16382,  16382,  16382
           1,4HPURG,4HE   ,  4HXPUR,4HGE  ,  16383,  16383,  16383,  16383
           1,4HEQUI,4HV   ,  4HXEQU,4HIV  ,  16383,  16383,  16383,  16383
           1,4HEND ,4H    ,  4HXCEI,4H    ,  16383,  16383,  16383,  16383
           1,4HEXIT,4H    ,  4HXCEI,4H    ,  16383,  16383,  16383,  16383
           1,4HADD ,4H    ,  4HDADD,4H    ,     72,     72,     72,     72
           1,4HADD5,4H    ,  4HDADD,4HS   ,     64,     64,     64,     64
           1,4HBMG ,4H    ,  4HBMG ,4H    ,    512,    512,    512,    512
           1,4HCASE,4H    ,  4HCASE,4H    ,    512,    512,    512,    512
           1,4HCEAD,4H    ,  4HCEAD,4H    ,   1024,   1024,   1024,   1024
```

```
      1,4HCYCT,4H1   ,  4HCYCT,4H1   ,   64,   64,   64,   64
      1,4HCYCT,4H2   ,  4HCYCT,4H2   ,   64,   64,   64,   64
      1,4HDDR ,4H    ,  4HDDR ,4H    ,  128,  128,  128,  128
      1,4HDDR1,4H    ,  4HDDR1,4H    , 2048, 2048, 2048, 2048
      1/
     DATA LINK2/
      1 4HDDR2,4H    ,  4HDDR2,4H    , 2048, 2048, 2048, 2048
      1,4HDDCO,4HMP  ,  4HDDCO,4HMP  ,   64,   64,   64,   64
      1,4HDPD ,4H    ,  4HDPD ,4H    ,   32,   32,   32,   32
      1,4HDSMG,4H1   ,  4HDSMG,4H1   , 4096, 4096, 4096, 4096
      1,4HDSMG,4H2   ,  4HDSMG,4H2   ,    8,    8,    8,    8
      1,4HDUMM,4HOD1 ,  4HDUMO,4HD1  ,   64,   64,   64,   64
      1,4HDUMM,4HOD2 ,  4HDUMO,4HD2  ,   64,   64,   64,   64
      1,4HDUMM,4HOD3 ,  4HDUMO,4HD3  ,   64,   64,   64,   64
      1,4HDUMM,4HOD4 ,  4HDUMO,4HD4  ,   64,   64,   64,   64
      1,4HEMA ,4H    ,  4HEMA ,4H    ,  128,  128,  128,  128
      1,4HEMG ,4H    ,  4HEMG ,4H    ,  128,  128,  128,  128
      1,4HFBS ,4H    ,  4HFBS ,4H    ,   64,   64,   64,   64
      1,4HFRRD,4H    ,  4HFRRD,4H    ,  512,  512,  512,  512
      1,4HGKAD,4H    ,  4HGKAD,4H    ,  512,  512,  512,  512
      1,4HGKAM,4H    ,  4HGKAM,4H    ,  512,  512,  512,  512
      1,4HGP1 ,4H    ,  4HGP1 ,4H    ,    2,    2,    2,    2
      1,4HGP2 ,4H    ,  4HGP2 ,4H    ,    2,    2,    2,    2
      1,4HGP3 ,4H    ,  4HGP3 ,4H    ,    2,    2,    2,    2
      1/
     DATA LINK3/
      1 4HGP4 ,4H    ,  4HGP4 ,4H    ,    8,    8,    8,    8
      1,4HGPCY,4HC   ,  4HGPCY,4HC   ,   64,   64,   64,   64
      1,4HGPSP,4H    ,  4HGPSP,4H    ,    8,    8,    8,    8
      1,4HGPWG,4H    ,  4HGPWG,4H    ,    8,    8,    8,    8
      1,4HINPU,4HT   ,  4HINPU,4HT   ,    2,    2,    2,    2
      1,4HINPU,4HTT1 ,  4HINPT,4HT1  ,    2,    2,    2,    2
      1,4HINPU,4HTT2 ,  4HINPT,4HT2  ,   64,   64,   64,   64
      1,4HINPU,4HTT3 ,  4HINPT,4HT3  ,    2,    2,    2,    2
      1,4HINPU,4HTT4 ,  4HINPT,4HT4  ,    2,    2,    2,    2
      1,4HMATG,4HPR  ,  4HMATG,4HPR  ,  128,  128,  128,  128
      1,4HMATP,4HRN  ,  4HMATP,4HRN  ,   64,   64,   64,   64
      1,4HMATP,4HRT  ,  4HPRTI,4HNT  ,  128,  128,  128,  128
      1,4HMCE1,4H    ,  4HMCE1,4H    ,    8,    8,    8,    8
      1,4HMCE2,4H    ,  4HMCE2,4H    ,    8,    8,    8,    8
      1,4HMERG,4HE   ,  4HMERG,4HE1  ,   64,   64,   64,   64
      1,4HMODA,4H    ,  4HMODA,4H    ,   64,   64,   64,   64
      1,4HMODA,4HCC  ,  4HMODA,4HCC  , 2048, 2048, 2048, 2048
      1,4HMODB,4H    ,  4HMODB,4H    ,   64,   64,   64,   64
      1/
     DATA LINK4/
      1 4HMODC,4H    ,  4HMODC,4H    ,   64,   64,   64,   64
      1,4HNPYA,4HD   ,  4HDMPY,4HAD  ,   64,   64,   64,   64
      1,4HMTRX,4HIN  ,  4HMTRX,4HIN  ,  512,  512,  512,  512
      1,4HOFP ,4H    ,  4HOFP ,4H    , 8192, 8192, 8192, 8192
      1,4HOUTP,4HUT  ,  4HOUTP,4HT   , 8192, 8192, 8192, 8192
      1,4HOUTP,4HUT1 ,  4HOUTP,4HT1  , 8192, 8192, 8192, 8192
      1,4HOUTP,4HUT2 ,  4HOUTP,4HT2  ,   64,   64,   64,   64
      1,4HOUTP,4HUT3 ,  4HOUTP,4HT3  , 8192, 8192, 8192, 8192
      1,4HOUTP,4HUT4 ,  4HOUTP,4HT4  , 8192, 8192, 8192, 8192
      1,4HPARA,4HM   ,  4HQPAR,4HAM  , 16382, 16382, 16382, 16382
```

```
1,4HPARA,4HML  ,  4HPARA,4HML  ,  16382,  16382,  16382,  16382
1,4HPARA,4HMR  ,  4HQPAR,4HMR  ,  16382,  16382,  16382,  16382
1,4HPART,4HN   ,  4HPART,4HN1  ,     64,     64,     64,     64
1,4HPART,4HVEC ,  4HSUBV,4HEC  ,    256,    256,    256,    256
1,4HPLA1,4H    ,  4HPLA1,4H    ,      4,      4,      4,      4
1,4HPLA2,4H    ,  4HPLA2,4H    ,   4096,   4096,   4096,   4096
1,4HPLA3,4H    ,  4HPLA3,4H    ,   4096,   4096,   4096,   4096
1,4HPLA4,4H    ,  4HPLA4,4H    ,   4096,   4096,   4096,   4096
1/
DATA LINK5/
1 4HPLOT,4H    ,  4HDPLO,4HT   ,      2,      2,      2,      2
1,4HPLTS,4HET  ,  4HDPLT,4HST  ,      2,      2,      2,      2
1,4HPLTT,4HRAN ,  4HPLTT,4HRA  ,    128,    128,    128,    128
1,4HPRTM,4HSG  ,  4HPRTM,4HSG  ,      2,      2,      2,      2
1,4HPRTP,4HARM ,  4HPRTP,4HRM  ,    128,    128,    128,    128
1,4HRAND,4HOM  ,  4HRAND,4HOM  ,   8192,   8192,   8192,   8192
1,4HRMG ,4H    ,  4HRMG ,4H    ,     16,     16,     16,     16
1,4HRBMG,4H1   ,  4HRBMG,4H1   ,      8,      8,      8,      8
1,4HRBMG,4H2   ,  4HRBMG,4H2   ,      8,      8,      8,      8
1,4HRBMG,4H3   ,  4HRBMG,4H3   ,      8,      8,      8,      8
1,4HRBMG,4H4   ,  4HRBMG,4H4   ,      8,      8,      8,      8
1,4HREAD,4H    ,  4HREIG,4H    ,     32,     32,     32,     32
1,4HSCE1,4H    ,  4HSCE1,4H    ,      8,      8,      8,      8
1,4HSDR1,4H    ,  4HSDR1,4H    ,   2048,   2048,   2048,   2048
1,4HSDR2,4H    ,  4HSDR2,4H    ,   4096,   4096,   4096,   4096
1,4HSDR3,4H    ,  4HSDR3,4H    ,   8192,   8192,   8192,   8192
1,4HSDRH,4HT   ,  4HSDRH,4HT   ,   4096,   4096,   4096,   4096
1,4HSEEM,4HAT  ,  4HSEEM,4HAT  ,      2,      2,      2,      2
1/
DATA LINK6/
1 4HSETV,4HAL  ,  4HSETV,4HAL  ,      2,      2,      2,      2
1,4HSMA1,4H    ,  4HSMA1,4H    ,      4,      4,      4,      4
1,4HSMA2,4H    ,  4HSMA2,4H    ,      4,      4,      4,      4
1,4HSMA3,4H    ,  4HSMA3,4H    ,      8,      8,      8,      8
1,4HSMP1,4H    ,  4HSMP1,4H    ,      8,      8,      8,      8
1,4HSMP2,4H    ,  4HSMP2,4H    ,      8,      8,      8,      8
1,4HSMPY,4HAD  ,  4HSMPY,4HAD  ,     64,     64,     64,     64
1,4HSOLV,4HE   ,  4HSOLV,4HE   ,     64,     64,     64,     64
1,4HSSG1,4H    ,  4HSSG1,4H    ,     16,     16,     16,     16
1,4HSSG2,4H    ,  4HSSG2,4H    ,     16,     16,     16,     16
1,4HSSG3,4H    ,  4HSSG3,4H    ,     16,     16,     16,     16
1,4HSSG4,4H    ,  4HSSG4,4H    ,     16,     16,     16,     16
1,4HSSGH,4HT   ,  4HSSGH,4HT   ,     16,     16,     16,     16
1,4HTA1 ,4H    ,  4HTA1 ,4H    ,      2,      2,      2,      2
1,4HTABP,4HREP ,  4HTABP,4HRE  ,    256,    256,    256,    256
1,4HTABP,4HCH  ,  4HTABP,4HCH  ,    256,    256,    256,    256
1,4HTABP,4HRT  ,  4HTABF,4HMT  ,    128,    128,    128,    128
1,4HSDRX,4H    ,  4HSDRX,4H    ,   8192,   8192,   8192,   8192
1/
DATA LINK7/
1 4HTABP,4HT   ,  4HTABP,4HT   ,    128,    128,    128,    128
1,4HTIME,4HTEST,  4HTIMT,4HST  ,    256,    256,    256,    256
1,4HTRD ,4H    ,  4HTRD ,4H    ,   1024,   1024,   1024,   1024
1,4HTRHT,4H    ,  4HTRHT,4H    ,   1024,   1024,   1024,   1024
1,4HTRLG,4H    ,  4HTRLG,4H    ,     16,     16,     16,     16
1,4HTRNS,4HP   ,  4HDTRA,4HNP  ,     64,     64,     64,     64
```

```
  168          1,4HUMER,4HGE   ,  4HDUME,4HRG   ,     64,     64,     64,     64
  169          1,4HUPAR,4HTN   ,  4HDUPA,4HRT   ,     64,     64,     64,     64
  170          1,4HVDR ,4H     ,  4HVDR ,4H     ,   2048,   2048,   2048,   2048
  171          1,4HVEC ,4H     ,  4HVEC ,4H     ,     64,     64,     64,     64
  172          1,4HXYPL,4HOT   ,  4HXYPL,4HOT   ,      2,      2,      2,      2
  173          1,4HXYPR,4HNPLT ,  4HXYPR,4HPT   ,   8192,   8192,   8192,   8192
  174          1,4HXYTR,4HAN   ,  4HXYTR,4HAN   ,   8192,   8192,   8192,   8192
  175          1,4HAPD ,4H     ,  4HAPD ,4H     ,    128,    128,    128,    128
  176          1,4HGI  ,4H     ,  4HGI  ,4H     ,    128,    128,    128,    128
  177          1,4HAMG ,4H     ,  4HAMG ,4H     ,    128,    128,    128,    128
  178          1,4HAMP ,4H     ,  4HAMP ,4H     ,    128,    128,    128,    128
  179          1,4HFA1 ,4H     ,  4HFA1 ,4H     ,    128,    128,    128,    128
  180          1/
  181          DATA LINK8/
  182          1 4HFA2 ,4H     ,  4HFA2 ,4H     ,    128,    128,    128,    128
  183          1,4HOPTP,4HR1   ,  4HOPTP,4HR1   ,   4224,   4224,   4224,   4224
  184          1,4HOPTP,4HR2   ,  4HOPTP,4HR2   ,   4224,   4224,   4224,   4224
  185          1,4HDSCH,4HK    ,  4HDSCH,4HK    ,    128,    128,    128,    128
  186          1,4HCOMB,4H1    ,  4HCOMB,4H1    ,    256,    256,    256,    256
  187          1,4HCOMB,4H2    ,  4HCOMB,4H2    ,    256,    256,    256,    256
  188          1,4HEXIO,4H     ,  4HEXIO,4H     ,    256,    256,    256,    256
  189          1,4HRCOV,4HR    ,  4HRCOV,4HR    ,    256,    256,    256,    256
  190          1,4HRCOV,4HR3   ,  4HRCOV,4HR3   ,    256,    256,    256,    256
  191          1,4HREDU,4HGE   ,  4HREDU,4HGE   ,    256,    256,    256,    256
  192          1,4HSGEN,4H     ,  4HSGEN,4H     ,    256,    256,    256,    256
  193          1,4HSOFI,4H     ,  4HSOFI,4H     ,    256,    256,    256,    256
  194          1,4HSOFO,4H     ,  4HSOFO,4H     ,    256,    256,    256,    256
  195          1,4HSOFU,4HT    ,  4HSOFU,4HT    ,    256,    256,    256,    256
  196          1,4HSUBP,4HH1   ,  4HSUBP,4HH1   ,    256,    256,    256,    256
  197          1,4HPLTH,4HRG   ,  4HPLTH,4HRG   ,    256,    256,    256,    256
  198          1,4HCURV,1H     ,  4HCURV,1H     ,   8192,8192,8192,8192
  199          1/
  200          END                                                        00001880
```

```
$PRT,S OBJ,LINK7
```

```
$CHMTZBIN209*OBJ(0),LINK7
      1        LIB--OBJ,LEG*UR,MSG*LOCAL1B
      2         TYPE  SETAFCM
      3        SEG-MAIN
      4        IN YTRACE
      5        IN-FREAD,GOPEN,SDEESP,FBSXEX
      6        IN SEMDBD,XSEM07,NTAB$,NTRANS
      7        IN-GNFIST
      8        IN UNPACK,&NTPK,BLDPK,PACK,WRTTRL,BITPOS,ADDX,MPYADX,FBSX,DCOMPX,CDCMPX,SFACT
      9        IN-TMTOGO,GFBSX,ZBLPKX,ZNTPKX,UNPAKX,PACKX,OPNCOR
     10        IN CORSZ,DESCRP,XCEITB,STIME,STAPID,TIME,XLINK,SEM
     11        IN-QREADX,TYPE,XHDMSK,OGLNT,GINOX,TWO,NAMES,OUTPUT,XVPS,XPFIST
     12        IN XXFIAT,XFIST,XFIAT,MSGX
     13        IN GINO,XGINO,GINOIO,FNAME,PEXIT,SSWTCH,UTIL,OPEN,CLOSE,READ,PPDUMP
     14        IN MAPFNS,MESAGE
     15        IN QWRITX,FWDREC,WRITE
     16        IN REWIND
     17        IN-BCKREC
     18        IN PATX,PARMEG
     19        IN XDPL,OSCENT,XNSTRN
     20        IN SYSTEM
     21        IN-CNO8BD,CONDAD,CONDAS
     22        SEG A1*
     23        IN-PAGE
     24       SEG TABPRT*,(A1)
     25        IN TABPRT,MATDUM,MATPRN
     26        SEG TABPRX*,(TABPRT)
     27        IN-TABPRX
     28        SEG INPTT2*,(A1)
     29        IN-INPTT2
     30        SEG INP2XX*,(INPTT2)
     31        IN INP2XX
     32        SEG OUT2XX*,(A1)
     33        IN OUTPT2
     34        SEG OXXXXX*,(OUT2XX)
     35        IN-OUT2XX
     36         SEG MSGWRT*,(A1)
     37         IN MSGWRT,USRMSG
     38         SEG ENDSYS*,(A1)
     39         IN-BTSTRP,ENDSYS,SEARCH
     40         IN DEFCOR,DFCOR
     41        SEG-ENDSSS*,(ENDSYS)
     42        IN ENDSSS
     43        SEG-QPARAM*,(A1)
     44        IN QPARAM,FNDPAR,QPARMR,PARAML
     45        SEG PARMLX*,(QPARAM)
     46        IN PARMLX
     47        SEG-XSAVE*,(A1)
     48        IN XSAVE
     49        SEG SKPFIL*,(A1)
     50        IN SKPFIL,EOF
     51        SEG SCEI*,(SKPFIL)
     52        IN XCEI
     53        SEG XCHK*,(SKPFIL)
     54        IN XCHK,XEOT,RWUNLD
     55        SEG XSFA*,(SKPFIL)
```

```
      IN XPOLCK,RPDABD,XPURGE
      IN XSFA,XCLEAN,XPUNP,XDPH,XSOSGN
      IN XSFA1
      SEG ESFA*,(XSFA1)
      IN ESFA
      SEG DMPYAD*,A1
      IN DMPYAD,DMPYX
      SEG SOLVE*,A1
      IN SOLVE
      SEG DDCOMP*,A1
      IN DDCOMP
      SEG DFBS*,A1
      IN DFBS
      SEG SMPYAD*,A1
      IN SMPYAD
      SEG MATRX*,(DMPYAD,SOLVE,DDCOMP,DFBS,SMPYAD)
      IN DMPY,MPYAD,MPYQ,FILSW1,MPYADZ,SSG2B
      SEG MPYA1D*,(MATRX)
      IN MPYA1D,MPYA2D
      SEG CYCT2*,(MATRX)
      IN CYCT2,CYCT2A,CYCT2B
      SEG CYCT2X*,(CYCT2)
      IN CYCT2X,SSGB2
      SEG CYCT1*,(MATRX)
      IN CYCT1
      SEG CYCT1Z*,(CYCT1)
      IN CYCT1Z
      SEG GPCYC*,MATRX
      IN GPCYC,BISRCH
      IN PRELOC
      SEG GPCYCX*,(GPCYC)
      IN GPCYCX
      SEG SOLV2X*,MATRX
      IN FBS,FBSSP,FBSDP
      SEG SOLVXX*,(SOLV2X)
      IN SOLV2X,DFBS1X
      SEG SOLV4X*,MATRX
      IN GFBS
      SEG SOLXXX*,(SOLV4X)
      IN SOLV4X,DFBS2X
      SEG DADDA*,MATRX
      IN SADDX
      IN ADD,DADD,SADD,DADD5
      SEG DADDAX*,(DADDA)
      IN MPYA3D,DADDA
      SEG GENX*,MATRX
      IN GENVEC,FINDC,TIMEEQ
      SEG SOLV3X*,(GENX)
      IN ONETWO,TRANSP,DECOMP,DLOOP
      SEG SOLXX1*,(SOLV3X)
      IN SOLV3X,DECP2X
      SEG SOLV5X*,(GENX)
      IN CXLOOP
      IN COM12,CDCOMP,GTRNSP
      SEG SOLXX2*,(SOLV5X)
      IN SOLV5X,DECP3X
```

```
   112        SEG SPSDC*,MATRX
   113        IN RSPSDC,RSPL00
   114        SEG DPSDC*,MATRX
   115        IN SDCOMP,LOOP
   116        SEG SOLV1X*,(SPSDC,DPSDC)
   117        IN SOLV1X,DECP1X
   118        SEG CSPSDC*,MATRX
   119        IN CSPSDC,CSPL00
   120        SEG SOLV6X*,(CSPSDC)
   121        IN SOLV6X,DECP4X
   122        SEG CXFBS*,MATRX
   123        IN CXFBS
   124        SEG SOLV7X*,(CXFBS)
   125        IN SOLV7X,DFBS3X
   126        SEG PRTMRG*,MATRX
   127        IN PARTN2,PARTN3
   128        IN PRTMRG
   129        SEG PART1*,(PRTMRG)
   130        IN PARTN1
   131        SEG PTMGZZ*,(PRTMRG)
   132        IN MERGE1
   133        SEG PTMGZC*,(PTMGZZ)
   134        IN PTMGZZ
   135        SEG DTRANX*,MATRX
   136        IN TRNSPX
   137        IN TRNSP,DTRANP
   138        SEG DTRAXX*,(DTRANX)
   139        IN DTRANX
   140        SEG MRGPRT*,MATRX
   141        IN CALCV,RULER
   142        SEG SDRB1*,(MRGPRT)
   143        IN DUMERG,MERGE,SDR1B
   144        SEG SDRB1X*,(SDRB1)
   145        IN SDRB1
   146        SEG UPART*,(MRGPRT)
   147        IN DUPART,UPART,PARTN.
   148        SEG UPARTX*,(UPART)
   149        IN UPARTX
   150        SEG VFCXXX*,MATRX
   151        IN VEC
   152        SEG VECXXX*,(VFCXXX)
   153        IN VECXXX
   154        SEG DUMOD1*,MATRX
   155        IN DUMOD1
   156        SEG DUM1XX*,(DUMOD1)
   157        IN DUM1XX
   158        SEG DUMOD2*,MATRX
   159        IN DUMOD2
   160        SEG DUM2XX*,(DUMOD2)
   161        IN DUM2XX
   162        SEG DUMOD3*,MATRX
   163        IN DUMOD3
   164        SEG DUM3XX*,(DUMOD3)
   165        IN DUM3XX
   166        SEG DUMOD4*,MATRX
   167        IN DUMOD4
```

```
      SEG DUM4XX*,(DUMOD4)
      IN-DUM4XX
      SEG MODA*,MATRX
      IN-MODA
      SEG MODXXX
        IN-MODAXX
      SEG MODB*,MATRX
      IN-MODB
      SEG MODC*,MATRX
      IN-MODC
      END
```

```
LOCK,OBJ(1),INPTT2
      1          SUBROUTINE INPTT2                                          INPT   1
      2      C                                                              INPT   2
      3      C   READ DATA BLOCK(S) FROM A FORTRAN UNIT.                    INPT   3
      4      C                                                              INPT   4
      5      C                    CALL TO THIS MODULE IS                    INPT   5
      6      C                                                              INPT   6
      7      C                      INPUTT2 /01,02,03,04,05/V,N,P1/V,N,P2/V,N,P3 $   INPT   7
      8      C                                                              INPT   8
      9      C                    P1=0 , NO ACTION TAKEN BEFORE READ        INPT   9
     10      C                    P1=+N, SKIP FORWARD N DATA BLOCKS BEFORE READ   INPT  10
     11      C                    P1=-1, FORTRAN TAPE IS REWOUND BEFORE READ     INPT  11
     12      C                    P1=-3, THE NAMES OF ALL DATA BLOCKS ON FORTRAN TAPE  INPT  12
     13      C                           ARE PRINTED AND READ OCCURS AT BEGINNING OF  INPT  13
     14      C                           TAPE                                 INPT  14
     15      C                    P1=-5, SEARCH FORTRAN TAPE FOR FIRST VERSION OF DATA  INPT  15
     16      C                           BLOCKS REQUESTED.                    INPT  16
     17      C                           IF ANY ARE NOT FOUND, A FATAL TERMINATION  INPT  17
     18      C                           OCCURS.                              INPT  18
     19      C                    P1=-6, SEARCH FORTRAN TAPE FOR FINAL VERSION OF DATA  INPT  19
     20      C                           BLOCKS REQUESTED.                    INPT  20
     21      C                           IF ANY ARE NOT FOUND, A FATAL TERMINATION  INPT  21
     22      C                           OCCURS.                              INPT  22
     23      C                    P1=-7, SEARCH FORTRAN TAPE FOR FIRST VERSION OF DATA  INPT  23
     24      C                           BLOCKS REQUESTED.                    INPT  24
     25      C                           IF ANY ARE NOT FOUND, A WARNING OCCURS.  INPT  25
     26      C                    P1=-8, SEARCH FORTRAN TAPE FOR FINAL VERSION OF DATA  INPT  26
     27      C                           BLOCKS REQUESTED.                    INPT  27
     28      C                           IF ANY ARE NOT FOUND, A WARNING OCCURS.  INPT  28
     29      C                    THE HPL DEFAULT VALUE FOR P1 IS  0 .         INPT  29
     30      C                                                                 INPT  30
     31      C                    P2 IS THE FORTRAN UNIT FROM WHICH THE DATA BLOCK(S)  INPT  31
     32      C                           WILL BE READ.                         INPT  32
     33      C                    THE HPL DEFAULT VALUE FOR P2 IS 0 .          INPT  33
     34      C                                                                 INPT  34
     35      C                    P3= TAPE ID CODE FOR FORTRAN TAPE, AN ALPHANUMERIC  INPT  35
     36      C                    VARIABLE WHOSE VALUE MUST MATCH A CORRESPONDING  INPT  36
     37      C                    VALUE ON THE FORTRAN TAPE.                   INPT  37
     38      C                    THIS CHECK IS DEPENDENT ON THE VALUE OF      INPT  38
     39      C                    P1 AS FOLLOWS..                              INPT  39
     40      C                          *P1*          *TAPE ID CHECKED*       INPT  40
     41      C                          +N            NO                      INPT  41
     42      C                           0            NO                      INPT  42
     43      C                          -1            YES                     INPT  43
     44      C                          -3            YES (WARNING CHECK)     INPT  44
     45      C                          -5            YES                     INPT  45
     46      C                          -6            YES                     INPT  46
     47      C                          -7            YES                     INPT  47
     48      C                          -8            YES                     INPT  48
     49      C                    THE HPL DEFAULT VALUE FOR P3 IS XXXXXXXX .   INPT  49
     50      C                                                                 INPT  50
     51      C                                                                 INPT  51
     52            INTEGER TRL(7),CORSZ,OUBUF,NAME(2),NONE(2),SUBNAM(2)        INPT  52
     53            INTEGER OUT(5),OUTPUT,P1,ZERO,NAMEX(2),P2                   INPT  53
     54            INTEGER P3,IDHDR(7),IDHDRX(7),P3X(2),NT(5,3),DX(3)          INPT  54
     55            INTEGER TAPCOD(2)                                           INPT  55
     56      C     INTEGER ENDREC,ENDFIL                                       INPT  56
```

```
      INTEGER RET                                                       INPT  57
C                                                                       INPT  58
      COMMON /     / P1,P2,P3(2)                                        INPT  59
     *      /SYSTEM/ NB,NOUT,JUNK(6),NLPP                              INPT  60
     *      /INP2XX/ X(2)                                              INPT  61
C                                                                       INPT  62
      DATA NONE,SUBNAM/4H (NO,4HNE) ,4HINPU,4HT2  /                     INPT  63
      DATA OUT/201,202,203,204,205/                                     INPT  64
      DATA ZERO,MONE,MTWO,MTRE,MFOR/0,-1,-2,-3,-4/                      INPT  65
     *     ,     MFIV,MSIX,METE/  -5,-6,-87                           INPT  66
      DATA IDHDR/4HNAST,4HRAN ,4HFORT,4H TAP,4HE ID,4H COD,4HE - /      INPT  67
      DATA XNULL/4HNULL/                                                8/74-GC.
C                                                                       INPT  68
C                                                                       INPT  69
C                                                                       INPT  70
C                                                                       INPT  71
      LCOR=CORSZ(P1,X) - NB                                             INPT  72
      IF(LCOR.LE.0) CALL MESAGE(-8,LCOR,SUBNAM)                         INPT  73
      OUBUF=LCOR+1                                                      INPT  74
      TAPCOD(1) = P3(1)                                                 INPT  75
      TAPCOD(2) = P3(2)                                                 INPT  76
      IN=P2                                                             INPT  77
      IF(P1.LT.METE .OR. P1.EQ.MTWO .OR. P1.EQ.MFOR) GO TO 9908         INPT  78
C                                                                       INPT  79
      IF(P1.LT.MFOR) GO TO 3000                                         INPT  80
      IF(P1.EQ.MTRE) GO TO 2000                                         INPT  81
      IF(P1.LE.ZERO) GO TO 190                                          INPT  82
C                                                                       INPT  83
      I=1                                                               INPT  84
  110 CONTINUE                                                          INPT  85
 8000 FORMAT(22A6)
      READ(IN,8000) KEY                                                 INPT  86
      KEYX=2                                                            INPT  87
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT  88
      READ(IN,8000) NAMEX                                               INPT  89
      READ(IN,8000) KEY                                                 INPT  90
      IF(KEY.GE.0) GO TO 9919                                           INPT  91
      ASSIGN 120 TO RET                                                 INPT  92
      NSKIP=1                                                           INPT  93
      GO TO 5000                                                        INPT  94
  120 CONTINUE                                                          INPT  95
      I=I+1                                                             INPT  96
      IF(I.LE.P1) GO TO 110                                             INPT  97
      GO TO 195                                                         INPT  98
C                                                                       INPT  99
C                                                                       INPT 100
C     ------------- OPEN FORTRAN TAPE TO READ WITHOUT REWIND.          INPT 101
C                                                                       INPT 102
  190 CONTINUE                                                          INPT 103
      IF(P1.NE.MONE) GO TO 195                                          INPT 104
      REWIND IN                                                         INPT 105
      READ(IN,8000) KEY                                                 INPT 106
      KEYX=3                                                            INPT 107
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 108
      READ(IN,8000) DX                                                  INPT 109
      READ(IN,8000) KEY                                                 INPT 110
      KEYX=7                                                            INPT 111
```

```
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 112
      READ(IN,8000) IDHDRX                                                    INPT 113
      DO 192 KF=1,7                                                     INPT 114
      IF(IDHDRX(KF).NE.IDHDR(KF)) GO TO 9918                            INPT 115
  192 CONTINUE                                                          INPT 116
      READ(IN,8000) KEY                                                       INPT 117
      KEYX=2                                                            INPT 118
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 119
      READ(IN,8000) P3X                                                       INPT 120
      READ(IN,8000) KEY                                                       INPT 121
      IF(KEY.GE.0) GO TO 9919                                           INPT 122
      IF(P3X(1).NE.P3(1) .OR. P3X(2).NE.P3(2)) GO TO 9910               INPT 123
      ASSIGN 193 TO RET                                                 INPT 124
      NSKIP=1                                                           INPT 125
      GO TO 5000                                                        INPT 126
  193 CONTINUE                                                          INPT 127
C                                                                       INPT 128
  195 DO 1000 I=1,5                                                     INPT 129
C                                                                       INPT 130
      OUTPUT=OUT(I)                                                     INPT 131
      CALL FNAME(OUTPUT,NAME)                                           INPT 132
      IF(NAME(1).EQ.NONE(1) .AND. NAME(2).EQ.NONE(2)) GO TO 1000        INPT 133
C                                                                       INPT 134
C                     PASS FILE NAME HEADER RECORD.                     INPT 135
C                                                                       INPT 136
      READ(IN,8000) KEY                                                       INPT 137
      KEYX=2                                                            INPT 138
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 139
      READ(IN,8000) NAMEX                                                     INPT 140
      READ(IN,8000) KEY                                                       INPT 141
      IF(KEY.GE.0) GO TO 9919                                           INPT 142
C                                                                       INPT 143
C                     READ TRAILER RECORD.                              INPT 144
C                                                                       INPT 145
      READ(IN,8000) KEY                                                       INPT 146
      KEYX=7                                                            INPT 147
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 148
      READ(IN,8000) TRL                                                       INPT 149
      READ(IN,8000) KEY                                                       INPT 150
      IF(KEY.GE.0) GO TO 9919                                           INPT 151
C                                                                       INPT 152
C                     OPEN OUTPUT DATA BLOCK TO WRITE WITH REWIND.      INPT 153
C                                                                       INPT 154
C     CALL OPEN(S9902,OUTPUT,X(OUBUF),1)                                INPT 155
      CALL OPEN(S9902,OUTPUT,X(OUBUF),1)                                INPT 156
C                                                                       INPT 157
C                     COPY CONTENTS OF FORTRAN TAPE ONTO OUTPUT         INPT 158
C                     DATA BLOCK.                                       INPT 159
C                                                                       INPT 160
C                                                                       9/75-NUK
C     NEXT 5 CARDS WERE REMOVED TO ALLOW HEADER-RECORD OF MORE THAN     9/75-NUK
C     TWO WORDS  (EXAMPLES...  USET,PPT)                                9/75-NUK
C                                                                       9/75-NUK
C     KEYX=2                                                            9/75-NUK
C     READ (IN,8000) KEY                                                      9/75-NUK
C     IF (KEY.NE.KEYX)  GO TO 9918                                      9/75-NUK
C     READ (IN,8000) (X(L),L=1,2)                                             9/75-NUK
```

```
      C     CALL WRITE (OUTPUT2,NAME,2,0)                                  9775=NUK
            NULLCK=0                                                       8774=GC.
      C                                                                    INPT 166
          200 CONTINUE                                                     INPT 167
            READ(IN,8000) KEY                                                INPT 168
            IF(KEY) 250,300,210                                            INPT 169
          210 CONTINUE                                                     INPT 170
            IF(KEY.GT.LCOR) GO TO 9917                                     INPT 171
            READ(IN,8000) (X(L),L=1,KEY)                                     INPT 172
            IF (X(2).EQ.XNULL) GO TO 240                                   8774=GC.
            CALL WRITE(OUTPUT,X,KEY,0)                                     INPT 173
            GO TO 200                                                      INPT 174
          240 NULLCK=1                                                     8774=GC.
            GO TO 260                                                      8774=GC.
      C                                                                    INPT 175
          250 CONTINUE                                                     INPT 176
            IF (NULLCK.NE.1) GO TO 260                                     8774=GC.
            NULLCK=0                                                       8774=GC.
            GO TO 200                                                      8774=GC.
          260 CONTINUE                                                     8774=GC.
            CALL WRITE(OUTPUT,X,0,1)                                       INPT 177
            GO TO 200                                                      INPT 178
      C                                                                    INPT 179
      C                     CLOSE OUTPUT DATA BLOCK WITH REWIND AND EOF.   INPT 180
      C                                                                    INPT 181
          300 CALL CLOSE(OUTPUT,1)                                         INPT 182
      C                                                                    INPT 183
      C                     WRITE TRAILER.                                 INPT 184
      C                                                                    INPT 185
            TRL(1)=OUTPUT                                                  INPT 186
            CALL WRTTRL(TRL)                                               INPT 187
            WRITE(NOUT,400) NAME,IN,NAMEX                                  INPT 188
          400 FORMAT(35H0*** USER INFORMATION MESSAGE 4105,                INPT 189
           * 5X,11HDATA BLOCK ,2A4,29H RETRIEVED FROM FORTRAN TAPE ,I2 /    INPT 190
           *      52H NAME OF DATA BLOCK WHEN PLACED ON FORTRAN TAPE WAS ,  INPT 191
           *      2A4,1H.)                                                  INPT 192
      C                                                                    INPT 193
         1000 CONTINUE                                                     INPT 194
      C                                                                    INPT 195
      C                     CLOSE FORTRAN TAPE WITHOUT REWIND.             INPT 196
      C                                                                    INPT 197
            RETURN                                                         INPT 198
      C                                                                    INPT 199
      C                     OBTAIN LIST OF DATA BLOCKS ON FORTRAN TAPE.    INPT 200
      C                                                                    INPT 201
         2000 CONTINUE                                                     INPT 202
            REWIND IN                                                      INPT 203
            READ(IN,8006) KEY                                                INPT 204
            KEYX=3                                                         INPT 205
            IF(KEY.NE.KEYX) GO TO 9918                                     INPT 206
            READ(IN,8000) DX                                                 INPT 207
            READ(IN,8000) KEY                                                INPT 208
            KEYX=7                                                         INPT 209
            IF(KEY.NE.KEYX) GO TO 9918                                     INPT 210
            READ(IN,8000) IDHDRX                                             INPT 211
            DO 2005 KF=1,7                                                 INPT 212
            IF(IDHDRX(KF).NE.IDHDR(KF)) GO TO 9917                         INPT 213
```

```
      2005 CONTINUE                                                     INPT 214
           READ(IN,8000) KEY                                                 INPT 215
           KEYX=2                                                       INPT 216
           IF(KEY.NE.KEYX) GO TO 9918                                   INPT 217
           READ(IN,8000) P3X                                                 INPT 218
           READ(IN,8000) KEY                                                 INPT 219
           IF(KEY.GE.0) GO TO 9919                                      INPT 220
           IF(P3X(1).NE.P3(1) .OR. P3X(2).NE.P3(2)) GO TO 9914          INPT 221
      2006 CONTINUE                                                     INPT 222
           ASSIGN 2008 TO RET                                           INPT 223
           NSKIP=1                                                      INPT 224
           GO TO 5000                                                   INPT 225
      2008 CONTINUE                                                     INPT 226
           KF=0                                                         INPT 227
      2007 CALL PAGE(1                                                  INPT 228
           WRITE(NOUT,2010) IN                                          INPT 229
 2010 FORMAT (1H0,50X,I4,14H FILE CONTENTS/46X,4HFILE,I8X,4HNAME/1H0)   8/74*GC.
      2020 CONTINUE                                                     INPT 231
           READ(IN,8000) KEY                                                 INPT 232
           IF(KEY.EQ.0) GO TO 2050                                      INPT 233
C          KEYX=2                                                       INPT 234
C          IF(KEY.NE.KEYX) GO TO 9918                                   INPT 235
           READ(IN,8000) NAMEX                                               INPT 236
C          READ(IN,8000) KEY                                                 INPT 237
C          IF(KEY.GE.0) GO TO 9919                                      INPT 238
           ASSIGN 2025 TO RET                                           INPT 239
           NSKIP=1                                                      INPT 240
           GO TO 5000                                                   INPT 241
      2025 CONTINUE                                                     INPT 242
           KF=KF+1                                                      INPT 243
           WRITE(NOUT,2030) KF,NAMEX                                    INPT 244
 2030 FORMAT(45X,I5,(8X,2A4)                                            INPT 245
           IF(MOD(KF,NLPP)) 2020,2007,2020.                             INPT 246
      2050 CONTINUE                                                     INPT 247
           REWIND IN                                                    INPT 248
           ASSIGN 2055 TO RET                                           INPT 249
           NSKIP=1                                                      INPT 250
           GO TO 5000                                                   INPT 251
      2055 CONTINUE                                                     INPT 252
           GO TO 195                                                    INPT 253
C                                                                       INPT 254
C                                                                       INPT 255
C                                                                       INPT 256
C*****************************************************************    INPT 257
C                                                                       INPT 258
C                          SEARCH MODE                                  INPT 259
C                                                                       INPT 260
 3000 CONTINUE                                                          INPT 261
C                                                                       INPT 262
C                    EXAMINE OUTPUT REQUESTS AND FILL NAME TABLE.       INPT 263
C                                                                       INPT 264
           NNT=0                                                        INPT 265
           DO 3050 I=1,5                                                INPT 266
           OUTPUT=OUT(I)                                                INPT 267
           CALL FNAME(OUTPUT,NAME)                                      INPT 268
           IF(NAME(1).EQ.NONE(1) .AND. NAME(2).EQ.NONE(2)) GO TO 3020   INPT 269
           NT(I,1)=0                                                    INPT 270
```

```
      NT(1,2)=NAME(1)                                                   INPT 271
      NT(1,3)=NAME(2)                                                   INPT 272
      NNT=NNT+1                                                         INPT 273
      GO TO 3050                                                        INPT 274
 3020 NT(1,1)=-1                                                        INPT 275
      NT(1,2)=NONE(1)                                                   INPT 276
      NT(1,3)=NONE(2)                                                   INPT 277
 3050 CONTINUE                                                          INPT 278
C                                                                       INPT 279
      IF(NNT.GT.0) GO TO 3070                                           INPT 280
      WRITE(NOUT,3060)                                                  INPT 281
 3060 FORMAT(3(/),6H0***,' USER WARNING MESSAGE 4,37,                   INPT 282
     *       56H  ALL OUTPUT DATA BLOCKS FOR INPUTT2 ARE PURGED.)       INPT 283
      RETURN                                                            INPT 284
C                                                                       INPT 285
C                        CHECK TAPE ID LABEL.                           INPT 286
C                                                                       INPT 287
 3070 CONTINUE                                                          INPT 288
      REWIND IN                                                         INPT 289
      READ(IN,8000) KEY                                                 INPT 290
      KEYX=3                                                            INPT 291
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 292
      READ(IN,8000) DX                                                  INPT 293
      READ(IN,8000) KEY                                                 INPT 294
      KEYX=7                                                            INPT 295
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 296
      READ(IN,8000) IDHDRX                                              INPT 297
      DO 3080 KF=1,7                                                    INPT 298
      IF(IDHDRX(KF).NE.IDHDR(KF)) GO TO 9913                            INPT 299
 3080 CONTINUE                                                          INPT 300
      READ(IN,8000) KEY                                                 INPT 301
      KEYX=2                                                            INPT 302
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 303
      READ(IN,8000) P3X                                                 INPT 304
      READ(IN,8000) KEY                                                 INPT 305
      IF(KEY.GE.0) GO TO 9919                                           INPT 306
      IF(P3X(1).NE.P3(1) .OR. P3X(2).NE.P3(2)) GO TO 9910               INPT 307
      ASSIGN 3085 TO RET                                                INPT 308
      NSKIP=1                                                           INPT 309
      GO TO 5000                                                        INPT 310
 3085 CONTINUE                                                          INPT 311
C                                                                       INPT 312
C                                                                       INPT 313
C                        BEGIN SEARCH OF TAPE.                          INPT 314
C                                                                       INPT 315
      KF=0                                                              INPT 316
 3110 CONTINUE                                                          INPT 317
      READ(IN,8000) KEY                                                 INPT 318
      IF(KEY.EQ.0) GO TO 3500                                           INPT 319
C     KEYX=2                                                            INPT 320
C     IF(KEY.NE.KEYX) GO TO 9918                                        INPT 321
      READ(IN,8000) NAMEX                                               INPT 322
      READ(IN,8000) KEY                                                 INPT 323
      IF(KEY.GE.0) GO TO 9919                                           INPT 324
      KF=KF+1                                                           INPT 325
C                                                                       INPT 326
      DO 3200 I=1,5                                                     INPT 327
```

```
      NAME(1)=NT(I,2)                                                   INPT 328
      NAME(2)=NT(I,3)                                                   INPT 329
      IF(NT(I,1).LT.0) GO TO 3200                                       INPT 330
      IF(NAME(1).NE.NAMEX(1) .OR. NAME(2).NE.NAMEX(2)) GO TO 3200       INPT 331
      NT(I,1)=NT(I,1)+1                                                 INPT 332
      IF(NT(I,1).EQ.1 .OR. PI.EQ.MSIX .OR. PI.EQ.METE) GO TO 3150       INPT 333
      WRITE(NOUT,3140) NAME,KF                                          INPT 334
 3140 FORMAT(31H0*** USER WARNING MESSAGE 4,38,                         INPT 335
     *    5X,11HDATA BLOCK ,2A4,21H (DATA BLOCK COUNT = ,I5,           INPT 336
     *      5H)  HAS PREVIOUSLY BEEN RETRIEVED FROM FORTRAN TAPE ,I2,  INPT 337
     *      21H AND WILL BE IGNORED.)                                  INPT 338
      GO TO 3205                                                        INPT 339
 3150 CONTINUE                                                          INPT 340
      READ(IN,8000) KEY                                                 INPT 341
      KEYX=7                                                            INPT 342
      IF(KEY.NE.KEYX) GO TO 9918                                        INPT 343
      READ(IN,8000) TRL                                                 INPT 344
      READ(IN,8000) KEY                                                 INPT 345
      IF(KEY.GE.0) GO TO 9919                                           INPT 346
      OUTPUT=OUT(I)                                                     INPT 347
C     CALL OPEN(*9902,OUTPUT,X(OUBUF),1)                                INPT 348
      CALL OPEN(*9902,OUTPUT,X(OUBUF),1)                                INPT 349
      NULLCK=0                                                          8/74-GC.
 3160 CONTINUE                                                          INPT 350
      READ(IN,8000) KEY                                                 INPT 351
      IF(KEY) 3170,3180,3165                                            INPT 352
 3165 CONTINUE                                                          INPT 353
      IF(KEY.GT.LCOR) GO TO 9917                                        INPT 354
      READ(IN,8000) (X(L),L=1,KEY)                                      INPT 355
      IF (X(2).EQ.XNULL) GO TO 3168                                     8/74-GC.
      CALL WRITE(OUTPUT,X,KEY,0)                                        INPT 356
      GO TO 3160                                                        INPT 357
 3168 NULLCK=1                                                          8/74-GC.
      GO TO 3175                                                        8/74-GC.
 3170 CONTINUE                                                          INPT 358
      IF (NULLCK.NE.1) GO TO 3175                                       8/74-GC.
      NULLCK=0                                                          8/74-GC.
      GO TO 3160                                                        8/74-GC.
 3175 CONTINUE                                                          8/74-GC.
      CALL WRITE(OUTPUT,X,0,1)                                          INPT 359
      GO TO 3160                                                        INPT 360
 3180 CALL CLOSE(OUTPUT,1)                                              INPT 361
      TRL(1)=OUTPUT                                                     INPT 362
      CALL WRTTRL(TRL)                                                  INPT 363
      WRITE(NOUT,3185) NAME,IN,KF                                       INPT 364
 3185 FORMAT(35H0*** USER INFORMATION MESSAGE 4,39,                     INPT 365
     *  5X,11HDATA BLOCK ,2A4,29H RETRIEVED FROM FORTRAN TAPE ,I2,      INPT 366
     *    21H (DATA BLOCK COUNT = ,I5,1H)    )                        INPT 367
      IF(NT(I,1).GT.1) GO TO 3190                                       INPT 368
      NNT=NNT-1                                                         INPT 369
      GO TO 3210                                                        INPT 370
 3190 WRITE(NOUT,3195)                                                  INPT 371
 3195 FORMAT(31H0*** USER WARNING MESSAGE 9,40,                         INPT 372
     *  58H SECONDARY VERSION OF DATA BLOCK HAS REPLACED EARLIER ONE.)  INPT 373
      GO TO 3210                                                        INPT 374
 3200 CONTINUE                                                          INPT 375
C                                                                       INPT 376
```

```
 3205 CONTINUE                                                          INPT 377
      ASSIGN 3207 TO RET                                                INPT 378
      NSKIP=1                                                           INPT 379
      GO TO 5000                                                        INPT 380
 3207 CONTINUE                                                          INPT 381
 3210 IF(NNT.GT.0 .OR. PI.EQ.HSIK .OR. PI.EQ.HETE) GO TO 3110           INPT 382
      GO TO 3900                                                        INPT 383
C                                                                       INPT 384
 3500 IF(NNT.LE.0) GO TO 3900                                           INPT 385
      IF(PI.EQ.HFIV .OR. PI.EQ.HSIX) GO TO 9916                         INPT 386
      WRITE(NOUT,3510)                                                  INPT 387
 3510 FORMAT(31H0*** USER WARNING MESSAGE 9141,                         INPT 388
     * 51H ONE OR MORE DATA BLOCKS NOT FOUND ON FORTRAN TAPE,  )        INPT 389
      DO 3530 I=1,6                                                     INPT 390
      IF(NT(1,I ),NE.0) GO TO 3530                                      INPT 391
      WRITE(NOUT,3520) NT(1,2),NT(1,3)                                  INPT 392
 3520 FORMAT(20X,21HNAME OF DATA BLOCK = ,2A4)                          INPT 393
 3530 CONTINUE                                                          INPT 394
C                                                                       INPT 395
 3900 CONTINUE                                                          INPT 396
      ASSIGN 3905 TO RET                                                INPT 397
      NSKIP=1                                                           INPT 398
      GO TO 5000                                                        INPT 399
 3905 CONTINUE                                                          INPT 400
      RETURN                                                            INPT 401
C                                                                       INPT 402
C                    SIMULATION OF SKPFIL(IN,NSKIP)                     INPT 403
C                                                                       INPT 404
 5000 CONTINUE                                                          INPT 405
      IF(NSKIP) 5100,5050,5200                                          INPT 406
 5050 GO TO RET,( 120, 193,2008,2025,2055,3086,3207,3905)               INPT 407
 5100 REWIND IN                                                         INPT 408
C NSKIP=COMPLEMENT OF NSKIP,                                            INPT 409
 5200 DO 5290 NS=1,NSKIP                                                INPT 410
 5205 READ(IN,8000) KWORD                                               INPT 411
      IF(KWORD) 5220,5230,5210                                          INPT 412
 5210 CONTINUE                                                          INPT 413
      IF(KWORD.GT.LCOR) GO TO 9917                                      INPT 414
      READ(IN,8000) (X(L),L=1,KWORD)                                    INPT 415
      GO TO 5205                                                        INPT 416
 5220 NREC=-KWORD                                                       INPT 417
      GO TO 5205                                                        INPT 418
 5230 CONTINUE                                                          INPT 419
 5290 CONTINUE                                                          INPT 420
      GO TO 5050                                                        INPT 421
C                                                                       INPT 422
C ---------- ---------- ---------- ---------- ---------- ---------- -- INPT 423
C ---------- ---------- ---------- ---------- ---------- ---------- -- INPT 424
C**********                                                             INPT 425
C                                                                       INPT 426
C                         ERRORS                                        INPT 427
C ---------- ---------- ---------- ---------- ---------- ---------- -- INPT 428
C ---------- ---------- ---------- ---------- ---------- ---------- -- INPT 429
 9902 WRITE(NOUT,9952) OUTPUT                                           INPT 440
 9952 FORMAT(31H0*** SYSTEM FATAL MESSAGE 9108,  ,                      INPT 441
     *51HSUBROUTINE INPTT2 UNABLE TO OPEN OUTPUT DATA BLOCK ,I5)         INPT 442
      GO TO 9995                                                        INPT 443
```

```
      C                                                                 INPT 490
 9908 WRITE(NOUT,9958) P1                                               INPT 491
 9958 FORMAT(83HO*** USER FATAL MESSAGE 4113, MODULE INPUTT2 - ILLEGAL VINPT 492
     *ALUE FOR FIRST PARAMETER = ,I20)                                   INPT 493
      GO TO 9995                                                        INPT 494
      C                                                                 INPT 495
 9910 WRITE(NOUT,9960) P3X(1),P3X(2) , P3(1),P3(2)                      INPT 496
 9960 FORMAT(51HO*** USER  FATAL  MESSAGE 4136, USER TAPE ID CODE =,    INPT 497
     * 2A4,  47H= DOES NOT MATCH THIRD INPUTT2 DMAP PARAMETER =,        INPT 498
     * 2A4,   2H=,)                                                     INPT 499
      GO TO 9995                                                        INPT 500
      C                                                                 INPT 525
 9913 WRITE(NOUT,9963) (IDHDRX(KF),KF=1,7)                              INPT 526
 9963 FORMAT(74HO*** USER FATAL MESSAGE 4134, MODULE INPUTT2 - ILLEGAL TINPT 527
     *APE CODE HEADER = ,7A4)                                           INPT 528
      GO TO 9995                                                        INPT 529
      C                                                                 INPT 530
 9914 WRITE(NOUT,9964) P3X(1),P3X(2) , P3(1),P3(2)                      INPT 531
 9964 FORMAT(51HO*** USER WARNING MESSAGE 4135, USER TAPE ID CODE =,    INPT 532
     * 2A4,  47H= DOES NOT MATCH THIRD INPUTT2 DMAP PARAMETER =,        INPT 533
     * 2A4,   2H=,)                                                     INPT 534
      GO TO 2006                                                        INPT 535
      C                                                                 INPT 547
 9916 WRITE(NOUT,9966)                                                  INPT 548
 9966 FORMAT(29HO*** USER FATAL MESSAGE 4142,                           INPT 549
     * 5X,51HONE OF MORE DATA BLOCKS NOT FOUND ON FORTRAN TAPE.   )     INPT 550
      GO TO 9990                                                        INPT 551
      C                                                                 INPT 552
 9917 WRITE(NOUT,9967) LCOR,KEY                                         INPT 553
 9967 FORMAT(87HO*** USER FATAL MESSAGE 2187, INSUFFICIENT WORKING CORE INPT 554
     *TO HOLD FORTRAN LOGICAL RECORD.   /                               INPT 555
     *       5X,25HLENGTH OF WORKING CORE = ,I10,1H,,5X,                INPT 556
     *         35HLENGTH OF FORTRAN LOGICAL RECORD = ,I10,1H.)          INPT 557
      GO TO 9995                                                        INPT 558
      C                                                                 INPT 559
 9918 WRITE(NOUT,9968) KEY,KEYX                                         INPT 560
 9968 FORMAT(56HO*** SYSTEM FATAL MESSAGE 2190, ILLEGAL VALUE FOR KEY = INPT 561
     *      ,I10,1H,,10X,17HEXPECTED VALUE = ,I10,1H. )                INPT 562
      GO TO 9995                                                        INPT 563
      C                                                                 INPT 564
 9919 WRITE(NOUT,9969) KEY                                              INPT 565
 9969 FORMAT(56HO*** SYSTEM FATAL MESSAGE 2190, ILLEGAL VALUE FOR KEY = INPT 566
     *      ,I10,1H.)                                                  INPT 567
      GO TO 9995                                                        INPT 568
      C                                                                 INPT 569
      C                                                                 INPT 570
 9990 DO 9992 I=1,5                                                     INPT 571
      IF(NT(I,1).NE.0) GO TO 9992                                       INPT 572
      WRITE(NOUT,9991) NT(I,2),NT(I,3)                                  INPT 573
 9991 FORMAT(20X,21HNAME OF DATA BLOCK = ,2A4)                          INPT 574
 9992 CONTINUE                                                          INPT 575
      GO TO 9995                                                        INPT 576
      C                                                                 INPT 577
      C                                                                 INPT 578
 9995 WRITE(NOUT,9996)                                                  INPT 579
 9996 FORMAT(17HO*** FATAL ERROR.)                                      INPT 580
      CALL MESAGE(-61,LCOR,SUBNAM)                                      INPT 581
```

```
513          RETURN
514    C
515          END                                          INPT 583
                                                        INPT 584
```

```
SCHMTZBIN209*OBJ(0):OUTPT2
      1              SUBROUTINE OUTPT2                                                OUTP    1
      2      C
      3      C       COPY DATA BLOCK(S) ONTO FORTRAN UNIT.                           OUTP    3
      4      C       MODIFIED TO HANDLE NULL COLUMNS AND ALLOW USER TAPE POSITIONING  8/74-GC.
      5      C       IN OUTPT2                                                       8/74-GC.
      6      C                                                                       OUTP    4
      7      C                       CALL TO THIS MODULE IS                          OUTP    5
      8      C                                                                       OUTP    6
      9      C                       OUTPUT2  IN1,IN2,IN3,IN4,IN5//V,N,P1/V,N,P2/V,N,P3 $OUTP    7
     10      C                                                                       OUTP    8
     11      C                       P1=0 , NO ACTION TAKEN BEFORE WRITE             OUTP    9
     12      C                       P1=+N, SKIP FORWARD N DATA BLOCKS BEFORE WRITE  OUTP   10
     13      C                       P1=-1, FORTRAN TAPE IS REWOUND BEFORE WRITE     OUTP   11
     14      C                       P1=-2, FORTRAN TAPE IS REWOUND AND ADVANCED TO  8/74-GC.
     15      C                              BEGINNING OF FIRST DATA BLOCK            8/74-GC.
     16      C                       P1=-3, THE NAMES OF ALL DATA BLOCKS ON FORTRAN TAPE OUTP   12
     17      C                              ARE PRINTED AND WRITE OCCURS AT THE END OF OUTP   13
     18      C                              TAPE                                     OUTP   14
     19      C                       P1=-9, WRITE A NULL FILE, ENDFILE AND           OUTP   15
     20      C                              REWIND FORTRAN TAPE.                     OUTP   16
     21      C                       THE MPL DEFAULT VALUE FOR P1 IS  0 .            OUTP   17
     22      C                                                                       OUTP   18
     23      C                       P2 IS THE FORTRAN UNIT NO. ON WHICH THE DATA    OUTP   19
     24      C                              BLOCKS WILL BE WRITTEN.                  OUTP   20
     25      C                       THE MPL DEFAULT VALUE FOR P2 IS  0 .            OUTP   21
     26      C                                                                       OUTP   22
     27      C                       P3= TAPE ID CODE FOR FORTRAN TAPE, AN ALPHANUMERIC OUTP 23
     28      C                       VARIABLE WHOSE VALUE WILL BE WRITTEN ON A FORTRAN OUTP  24
     29      C                       TAPE.                                           OUTP   25
     30      C                       THE WRITING OF THIS ITEM IS DEPENDENT ON THE    OUTP   26
     31      C                       VALUE OF P1 AS FOLLOWS..                        OUTP   27
     32      C                              *P1*          *TAPE ID WRITTEN*          OUTP   28
     33      C                               +N                NO                    OUTP   29
     34      C                               0                 NO                    OUTP   30
     35      C                               -1               YES                    OUTP   31
     36      C                               3          NO (WARNING CHECK)          OUTP   32
     37      C                       THE MPL DEFAULT VALUE FOR P3 IS XXXXXXXX .      OUTP   33
     38      C                                                                       OUTP   34
     39      C                                                                       OUTP   35
     40              INTEGER OUT,TRL(7),CORSZ,NAME(2),NONE(2),SUBNAM(2)              OUTP   36
     41              INTEGER IN(5),P1,ZERO,NAMEX(2),P2,P3,IDHDR(7),IDHDRX(7)         OUTP   37
     42              INTEGER P3X(2),D,DX(3)                                          OUTP   38
     43              INTEGER TAPCOD(2)                                               OUTP   39
     44              INTEGER ENDREC,ENDFIL                                           OUTP   40
     45              INTEGER RET                                                     OUTP   41
     46      C                                                                       OUTP   42
     47              COMMON /      / P1,P2,P3(2)                                     OUTP   43
     48      *               /SYSTEM/ NB,NOUT,JUNK(6),NLPP,JNK2(5),D(3)              OUTP   44
     49      *              /OUT2XX/ X(2)                                            OUTP   45
     50      C                                                                       OUTP   46
     51              DATA NONE,SUBNAM/4H (NO,4HNE) ,4HOUTP,4HT2  /                   OUTP   47
     52              DATA IN/101,102,103,104,105/                                    OUTP   48
     53              DATA ZERO,MONE,MTWO,MTRE,MNIN/0,-1,-2,-3,-9/                    OUTP   49
     54              DATA IDHDR/4HNAST,4HRAN ,4HFORT,4H TAP,4HE ID,4H COD,4HE  /     OUTP   50
     55              DATA XNULL/4HNULL/                                              8/74-GC.
```

```
    56       C                                                                  OUTP  51
    57       C                                                                  OUTP  52
    58       C                                                                  OUTP  53
    59       C                                                                  OUTP  54
    60  8000 FORMAT(22A6)                                                       OUTP  55
    61       ENDFIL=0                                                           OUTP  56
    62       ENDREC=0                                                           OUTP  57
    63       LCOR=CORSZ(P1,X) - NB                                              9/75-NUK
    64       C                                                                  9/75-NUK
    65       C     TO CONTROL FORTRAN DATA SIZE                                 9/75-NUK
    66       C                                                                  9/75-NUK
    67       IF (LCOR.GT.2000)  LCOR=2000                                       OUTP  58
    68       LCOR2=LCOR/2                                                       OUTP  59
    69       IF(LCOR.LE.0) CALL MESAGE(-8,LCOR,SUBNAM)                          OUTP  60
    70       INBUF=LCOR+1                                                       OUTP  61
    71       TAPCOD(1) = P3(1)                                                  OUTP  62
    72       TAPCOD(2) = P3(2)                                                  OUTP  63
    73       OUT=P2                                                             OUTP  64
    74       IF(P1.EQ.MNIN) GO TO 1500                                          8/74-GC.
    75       IF (P1.LT.MTRE) GO TO 9905                                         8/74-GC.
    76       C                                                                  OUTP  68
    77       IF (P1.EQ.MTWO .OR. P1.EQ.MTRE) GO TO 2000                         OUTP  69
    78       IF(P1.LE.ZERO) GO TO 190                                           OUTP  70
    79       C                                                                  OUTP  71
    80       I=1                                                                OUTP  72
    81   120 CONTINUE                                                           OUTP  73
    82       READ(OUT,8000) KEY                                                 OUTP  74
    83       KEYX=2                                                             OUTP  75
    84       IF(KEY.NE.KEYX) GO TO 9918                                         OUTP  76
    85       READ(OUT,8000) NAMEX                                               OUTP  77
    86       READ(OUT,8000) KEY                                                 OUTP  78
    87       IF(KEY.GE.0) GO TO 9919                                            OUTP  79
    88       ASSIGN 125 TO RET                                                  OUTP  80
    89       NSKIP=1                                                            OUTP  81
    90       GO TO 5000                                                         OUTP  82
    91   125 CONTINUE                                                           OUTP  83
    92       I=I+1                                                              OUTP  84
    93       IF(I.LE.P1) GO TO 120                                              OUTP  85
    94       C                                                                  OUTP  86
    95   190 CONTINUE                                                           OUTP  87
    96       IF(P1.NE.MONE) GO TO 195                                           OUTP  88
    97       REWIND OUT                                                         OUTP  89
    98       KEY=3                                                              OUTP  90
    99       WRITE(OUT,8000) KEY                                                OUTP  91
   100       WRITE(OUT,8000) D                                                  OUTP  92
   101       KEY=7                                                              OUTP  93
   102       WRITE(OUT,8000) KEY                                                OUTP  94
   103       WRITE(OUT,8000) IDHDR                                              OUTP  95
   104       KEY=2                                                              OUTP  96
   105       WRITE(OUT,8000) KEY
   106       WRITE(OUT,8000) P3
   107       ENDREC=ENDREC-1
   108       WRITE(OUT,8000) ENDREC
   109       WRITE(OUT,8000) ENDFIL
   110       ENDREC=0
   111       C
```

```
112      195 CONTINUE                                                    OUTP  97
113          DO 1000 I=1,5                                               OUTP  98
114      C                                                               OUTP  99
115          INPUT=IN(I)                                                 OUTP 100
116          CALL FNAME(INPUT,NAME)                                      OUTP 101
117          IF(NAME(1).EQ.NONE(1) .AND. NAME(2).EQ.NONE(2)) GO TO 1000  OUTP 102
118          TRL(1)=INPUT                                                OUTP 103
119          CALL RDTRL(TRL)                                             OUTP 104
120          IF(TRL(1).LE.0) GO TO 1000                                  OUTP 105
121      C                                                               OUTP 106
122      C        OPEN INPUT DATA BLOCK TO READ WITH REWIND.             OUTP 107
123      C                                                               OUTP 108
124      C   CALL OPEN(*9901,INPUT,X(INBUF),0)                          OUTP 109
125          CALL OPEN(*9901,INPUT,X(INBUF),0)                           OUTP 110
126          KEY=2                                                       OUTP 111
127          WRITE(OUT,8000) KEY                                             OUTP 112
128          WRITE(OUT,8000) NAME                                        OUTP 113
129          ENDREC=ENDREC-1                                             OUTP 114
130          WRITE(OUT,8000) ENDREC                                      OUTP 115
131          KEY=7                                                       OUTP 116
132          WRITE(OUT,8000) KEY                                         OUTP 117
133          WRITE(OUT,8000) TRL                                             OUTP 118
134          ENDREC=ENDREC-1                                             OUTP 119
135          WRITE(OUT,8000) ENDREC                                          OUTP 120
136      C                                                               OUTP 121
137      C              COPY CONTENTS OF INPUT DATA BLOCK ONTO FILE.     OUTP 122
138      C                                                               OUTP 123
139      C 200 CALL READ(*300,*250,INPUT,X(1),LCOR2,0,NF)                OUTP 124
140        200 CALL READ(*300,*250,INPUT,X(1),LCOR2,0,NF)                OUTP 125
141          WRITE(OUT,8000) LCOR2                                           OUTP 126
142          WRITE(OUT,8000) (X(L),L=1,LCOR2)                            OUTP 127
143          GO TO 200                                                   OUTP 128
144      C                                                               OUTP 129
145      250 CONTINUE                                                    OUTP 130
146          IF (NF.EQ.0) GO TO 260                                      8/74-GC.
147          WRITE(OUT,8000) NF                                              OUTP 131
148          WRITE(OUT,8000) (X(L),L=1,NF)                               OUTP 132
149          XX1=X(1)                                                    8/74-GC.
150      255 CONTINUE                                                    8/74-GC.
151          ENDREC=ENDREC-1                                             OUTP 133
152          WRITE(OUT,8000) ENDREC                                          OUTP 134
153          GO TO 200                                                   OUTP 135
154      260 NFZ=2                                                       8/74-GC.
155          WRITE (OUT,8000) NFZ                                            8/74-GC.
156          X(1)=XX1                                                    8/74-GC.
157          X(2)=XNULL                                                  8/74-GC.
158          WRITE (OUT,8000) (X(L),L=1,NFZ)                                 8/74-GC.
159          GO TO 255                                                   8/74-GC.
160      C                                                               OUTP 136
161      C              CLOSE INPUT DATA BLOCK WITH REWIND.              OUTP 137
162      C                                                               OUTP 138
163        300 CALL CLOSE(INPUT,1)                                       OUTP 139
164      C                                                               OUTP 140
165          WRITE(OUT,8000) ENDFIL                                          OUTP 141
166          ENDREC=0                                                    OUTP 142
167          WRITE(NOUT,350) NAME,OUT,TRL                                OUTP 143
```

```
  350 FORMAT(35HO***  USER INFORMATION MESSAGE 4114,/                    OUTP 144
     *       12H DATA BLOCK ,2A4,25H WRITTEN ON FORTRAN UNIT ,I2,         OUTP 145
     *       8H, TRL = /1X,7I18 )                                        OUTP 146
C                                                                        OUTP 147
 1000 CONTINUE                                                           OUTP 148
C                                                                        OUTP 149
C                    CLOSE FORTRAN TAPE WITHOUT REWIND.                  OUTP 150
C                                                                        OUTP 151
      WRITE (OUT,8000) ENDFIL                                            9/75-NUK
      ENDREC=0                                                           9/75-NUK
      BACKSPACE OUT                                                      9/75-NUK
      RETURN                                                             OUTP 155
C                                                                        OUTP 156
C                         FINAL CALL TO OUTPUT2.                         OUTP 157
C                                                                        OUTP 158
 1500 CONTINUE                                                           OUTP 159
      WRITE(OUT,8000) ENDFIL                                             OUTP 160
      ENDREC=0                                                           OUTP 161
      END FILE OUT                                                       OUTP 162
      REWIND OUT                                                         OUTP 163
      RETURN                                                             OUTP 164
C                                                                        OUTP 165
C                    OBTAIN LIST OF DATA BLOCKS ON FORTRAN TAPE.         OUTP 166
C                                                                        OUTP 167
 2000 CONTINUE                                                           OUTP 168
      REWIND OUT                                                         OUTP 169
      READ(OUT,8000) KEY                                                 OUTP 170
      KEYX=3                                                             OUTP 171
      IF(KEY.NE.KEYX) GO TO 9918                                         OUTP 172
      READ(OUT,8000)   DX                                                OUTP 173
      READ(OUT,8000) KEY                                                 OUTP 174
      KEYX=7                                                             OUTP 175
      IF(KEY.NE.KEYX) GO TO 9918                                         OUTP 176
      READ(OUT,8000) IDHDRX                                              OUTP 177
      DO 2005 KF=1,7                                                     OUTP 178
      IF(IDHDRX(KF).NE.IDHDR(KF)) GO TO 9913                             OUTP 179
 2005 CONTINUE                                                           OUTP 180
      READ(OUT,8000) KEY                                                 OUTP 181
      KEYX=2                                                             OUTP 182
      IF(KEY.NE.KEYX) GO TO 9918                                         OUTP 183
      READ(OUT,8000) P3X                                                 OUTP 184
      IF(P3X(1).NE.P3(1) .OR. P3X(2).NE.P3(2)) GO TO 9914                OUTP 185
 2006 CONTINUE                                                           OUTP 186
      ASSIGN 2008 TO RET                                                 OUTP 187
      NSKIP=1                                                            OUTP 188
      GO TO 5000                                                         OUTP 189
 2008 CONTINUE                                                           OUTP 190
      IF (P1.EQ.MTWO) RETURN                                             8/74-GC.
      KF=0                                                               OUTP 191
 2007 CALL PAGE1                                                         OUTP 192
      WRITE(NOUT,2010) OUT                                               OUTP 193
 2010 FORMAT (1HO,50X,I4,14H FILE CONTENTS/46X,4HFILE,18X,4HNAME/1HO)    8/74-GC.
 2020 CONTINUE                                                           OUTP 195
      READ(OUT,8000) KEY                                                 OUTP 196
      IF(KEY) 9915,2050,2023                                             OUTP 197
 2023 CONTINUE                                                           OUTP 198
```

```
      READ(OUT,8000) NAMEX                                                    OUTP 199
      ASSIGN 2025 TO RET                                                      OUTP 200
      NSKIP=1                                                                 OUTP 201
      GO TO 5000                                                              OUTP 202
2025  CONTINUE                                                                OUTP 203
      KF=KF+1                                                                 OUTP 204
      WRITE(NOUT,2030) KF,NAMEX                                               OUTP 205
2030  FORMAT(45X,I5,18X,2A4)                                                  OUTP 206
      IF(MOD(KF,NLPP)) 2020,2007,2020                                         OUTP 207
2050  CONTINUE                                                                OUTP 208
      BACKSPACE OUT                                                           8/74-GC.
      GO TO 195                                                               OUTP 213
C                                                                             OUTP 214
C                     SIMULATION OF SKPFIL(OUT,NSKIP) .                       OUTP 215
C                                                                             OUTP 216
5000  CONTINUE                                                                OUTP 217
      IF(NSKIP) 5100,5050,5200                                                OUTP 218
5050  GO TO RET, (125,2008,2025)                                              8/74-GC.
5100  REWIND OUT                                                              OUTP 220
C NSKIP=COMPLEMENT OF NSKIP.                                                  OUTP 221
5200  DO 5290 NS=1,NSKIP                                                      OUTP 222
5205  READ(OUT,8000) KWORD                                                    OUTP 223
      IF(KWORD) 5220,5230,5210                                                OUTP 224
5210  CONTINUE                                                                OUTP 225
      IF(KWORD.GT.LCOR) GO TO 9917                                            OUTP 226
      READ(OUT,8000) (X(L),L=1,KWORD)                                         OUTP 227
      GO TO 5205                                                              OUTP 228
5220  NREC=-KWORD                                                             OUTP 229
      GO TO 5205                                                              OUTP 230
5230  CONTINUE                                                                OUTP 231
5290  CONTINUE                                                                OUTP 232
      GO TO 5050                                                              OUTP 233
C                                                                             OUTP 234
C                                                                             OUTP 235
C*************************************************************************** OUTP 236
C                                                                             OUTP 237
C-------------------------ERRORS------------------------------------------- OUTP 238
C                                                                             OUTP 239
9901  WRITE(NOUT,9951) INPUT                                                  OUTP 240
9951  FORMAT(31H0*** SYSTEM FATAL MESSAGE 4116, ,                             OUTP 241
     *52HSUBROUTINE OUTPT2 UNABLE TO OPEN INPUT DATA BLOCK - ,I5)              OUTP 242
      GO TO 9995                                                              OUTP 243
C-------------------------------------------------------------------------- OUTP 244
9905  WRITE(NOUT,9955) P1                                                     OUTP 258
9955  FORMAT(83H0***-USER FATAL MESSAGE 4120, MODULE OUTPUT2 -- ILLEGAL VOUTP 259
     *ALUE FOR FIRST PARAMETER = ,I20)                                         OUTP 260
      GO TO 9995                                                              OUTP 261
C                                                                             OUTP 262
9913  WRITE(NOUT,9963) (IDHDRX(KF),KF=1,7)                                    OUTP 287
9963  FORMAT(74H0*** USER FATAL MESSAGE 4130, MODULE OUTPUT2 - ILLEGAL TOUTP 288
     *APE CODE HEADER = ,7A4)                                                 OUTP 289
      GO TO 9995                                                              OUTP 290
C                                                                             OUTP 291
9914  WRITE(NOUT,9964) P3X(1),P3X(2) , P3(1),P3(2)                            OUTP 292
9964  FORMAT(54H0*** USER WARNING MESSAGE 4131, FORTRAN TAPE ID CODE -, OUTP 293
     * 2A4,  47H- DOES NOT MATCH THIRD OUTPUT2 DMAP PARAMETER -,              OUTP 294
```

```
     *2A4,  2H=.)                                                       OUTP 295
           GO TO 2006                                                   OUTP 296
     C                                                                  OUTP 297
      9915 WRITE(NOUT,9965)                                             OUTP 298
      9965 FORMAT(59H0***SYSTEM FATAL MESSAGE 4115, MODULE OUTPUT2 - SHORT OUTP 299
           *EC.)                                                        OUTP 300
           GO TO 9995                                                   OUTP 301
     C                                                                  OUTP 302
      9917 WRITE(NOUT,9967) LCOR,KEY                                    OUTP 303
      9967 FORMAT(87H0*** USER FATAL MESSAGE 2187, INSUFFICIENT WORKING CORE OUTP 304
     *TO HOLD FORTRAN LOGICAL RECORD./                                  OUTP 305
     *       5X,25HLENGTH OF WORKING CORE = ,I10,1H,,5X,                OUTP 306
     *       35HLENGTH OF FORTRAN LOGICAL RECORD = ,I10,1H.)            OUTP 307
           GO TO 9995                                                   OUTP 308
     C                                                                  OUTP 309
      9918 WRITE(NOUT,9968) KEY,KEYX                                    OUTP 310
      9968 FORMAT(56H0*** SYSTEM FATAL MESSAGE 2190, ILLEGAL VALUE FOR KEY = OUTP 311
     *       ,I10,1H,,10X,17HEXPECTED VALUE = ,I10,1H. )                OUTP 312
           GO TO 9995                                                   OUTP 313
     C                                                                  OUTP 314
      9919 WRITE(NOUT,9969) KEY                                         OUTP 315
      9969 FORMAT(56H0*** SYSTEM FATAL MESSAGE 2190, ILLEGAL VALUE FOR KEY =
     *       ,I10,1H.)                                                  OUTP 316
           GO TO 9995                                                   OUTP 317
     C                                                                  OUTP 318
      9995 WRITE(NOUT,9996)                                             OUTP 319
      9996 FORMAT(17H0*** FATAL ERROR.)                                 OUTP 320
           CALL MESAGE(-61,LCOR,SUBNAM)                                 OUTP 321
           RETURN
     C
           END
```

Appendix C

Data File Listings

Table	Description
C-1	RUN1 File
C-2	START File
C-3	NAS1 File
C-4	LOOP File
C-5	PCH2 File

Table C-1

RUN1 FILE

```
@DATA,L RUN1.
DATA 7 RL70-5 01/09-23:42:57
      1.      @ASG,A START.
      2.      @ASG,A LEE.
      3.      @ASG,A NAS1.
      4.      @ASG,A PCH2.
      5.      @ASG,A LOOP.
      6.      @ASG,A NASTRAN.
      7.      @ASG,A OBJ.
      8.      @USE TPF$.,LEE.
      9.      @ADD,P START.
     10.      @CPDMPH,DO
     11.       PRT 15,30
     12.       PRT 17,30
     13.      @ADD,P LOOP.
     14.      @ADD,P LOOP.
     15.      @ADD,P LOOP.
     16.      @ADD,P LOOP.
     17.      @ADD,P LOOP.
     18.      @PMD,E
     19.      @FIN
END DATA.
```

Table C-2

START FILE

```
      SYM
@DATA,L START.
DATA 8R1 SL73R1 10/31/78 21:19:15 (0)
    1.      @ASG,T PLT1,F///2000
    2.      @USE 12.,PLT1.
    3.      @ASG,T INPT,F///2000
    4.      @ASG,T INP1,F///2000
    5.      @ASG,T INP2,F///2000
    6.      @ASG,T INP3,F///2000
    7.      @USE 14.,INPT.
    8.      @USE 15.,INP1.
    9.      @USE 16.,INP2.
   10.      @USE 17.,INP3.
   11.      @ASG,A PCH2.
   12.      @USE 18.,PCH2.
   13.      @ASG,T PCH1,F///1000
   14.      @ASG,C PCH3,F///1000
   15.      @USE 19.,PCH3.
   16.      @BRKPT PUNCH$/PCH1
   17.      @XQT CELFE
   18.      +1      CHANGE THIS CARD TO 0 FOR CLEFE RUN ONLY
   19.       0
   20.        HIGH VELOCITY IMPACT --- DEMONSTRATION PROBLEM
   21.        1   1  51    4   1   0   2   8   1   0   0
   22.        4.CO       0.       0.
   23.             0.0         0.0 -7.5600+3  2.0000+0  2.0000+0  8.2000-2
   24.       7.3CU0-2  5.1200-2        0.0  2.7027+3        0.0  6.0000+3  1.0000+0
   25.       76  33  13   3  16   3  62  30  77  34  76  33 648 103
   26.       20  20   0  15  15  10  10  10  10  10  10  10  10  10  20  20   0
   27.        1   4  10   7   2   5  11   8
   28.        2   5  11   8   3   6  12   9
   29.        3   6  12   9  13  14  16  15
   30.        4  17  20  10   5  18  21  11
   31.        5  18  21  11   6  19  22  12
   32.       10  20  23  26  11  21  24  27
   33.       11  21  24  27  12  22  25  28
   34.        7  10  26  29   8  11  27  30
   35.        8  11  27  30   9  12  28  31
   36.       17  32  35  20  18  33  36  21
   37.       18  33  36  21  19  34  37  22
   38.       20  35  38  23  21  36  39  24
   39.       21  36  39  24  22  37  40  25
   40.       26  23  38  41  27  24  39  42
   41.       27  24  39  42  28  25  40  43
   42.       29  26  41  44  30  27  42  45
   43.       30  27  42  45  31  28  43  46
   44.       32  47  50  35  33  48  51  36
   45.       33  48  51  36  34  49  52  37
   46.       35  50  53  38  36  51  54  39
   47.       36  51  54  39  37  52  55  40
   48.       41  38  53  56  42  39  54  57
   49.       42  39  54  57  43  40  55  58
   50.       44  41  56  59  45  42  57  60
   51.       45  42  57  60  46  43  58  61
   52.       47  62  65  50  48  63  66  51
   53.       48  63  66  51  49  64  67  52
   54.       50  65  68  53  51  66  69  54
```

(Continued)

Table C-2 (Continued)

SYM											
55.		51	66	69	54	52	67	70	55		
56.		56	53	68	71	57	54	69	72		
57.		57	54	69	72	58	55	70	73		
58.		59	56	71	74	60	57	72	75		
59.		60	57	72	75	61	58	73	76		
60.	GRID	1					20.		0.		0.
61.	GRID	2					10.		0.		0.041
62.	GRID	3					10.		0.		0.082
63.	GRID	4					20.15		0.		0.
64.	GRID	5					10.15		0.		0.041
65.	GRID	6					10.15		0.		0.082
66.	GRID	7					20.		0.15		0.
67.	GRID	8					10.		0.15		0.041
68.	GRID	9					10.		0.15		0.082
69.	GRID	10					20.15		0.15		0.
70.	GRID	11					10.15		0.15		0.041
71.	GRID	12					10.15		0.15		0.082
72.	GRID	13					10.		0.		0.232
73.	GRID	14					10.15		0.		0.232
74.	GRID	15					10.		0.15		0.232
75.	GRID	16					10.15		0.15		0.232
76.	GRID	17					20.35		0.		0.
77.	GRID	18					10.35		0.		0.041
78.	GRID	19					10.35		0.		0.082
79.	GRID	20					20.35		0.15		0.
80.	GRID	21					10.35		0.15		0.041
81.	GRID	22					10.35		0.15		0.082
82.	GRID	23					20.35		0.35		0.
83.	GRID	24					10.35		0.35		0.041
84.	GRID	25					10.35		0.35		0.082
85.	GRID	26					20.15		0.35		0.
86.	GRID	27					10.15		0.35		0.041
87.	GRID	28					10.15		0.35		0.082
88.	GRID	29					20.		0.35		0.
89.	GRID	30					10.		0.35		0.041
90.	GRID	31					10.		0.35		0.082
91.	GRID	32					20.5		0.		0.
92.	GRID	33					10.5		0.		0.041
93.	GRID	34					10.5		0.		0.082
94.	GRID	35					20.5		0.35		0.
95.	GRID	36					10.5		0.35		0.041
96.	GRID	37					10.5		0.35		0.082
97.	GRID	38					20.5		0.5		0.
98.	GRID	39					10.5		0.5		0.041
99.	GRID	40					10.5		0.5		0.082
100.	GRID	41					20.35		0.5		0.
101.	GRID	42					10.35		0.5		0.041
102.	GRID	43					10.35		0.5		0.082
103.	GRID	44					20.		0.5		0.
104.	GRID	45					10.		0.5		0.041
105.	GRID	46					10.		0.5		0.082
106.	GRID	47					21.		0.		0.
107.	GRID	48					11.		0.		0.041
108.	GRID	49					11.		0.		0.082
109.	GRID	50					21.		0.5		0.
110.	GRID	51					11.		0.5		0.041

Table C-2 (Concluded)

```
SYM
111.      GRID    52                  11.       0.5      0.082
112.      GRID    53                  21.       1.       0.
113.      GRID    54                  11.       1.       0.041
114.      GRID    55                  11.       1.       0.082
115.      GRID    56                  20.5      1.       0.
116.      GRID    57                  10.5      1.       0.041
117.      GRID    58                  10.5      1.       0.082
118.      GRID    59                  20.       1.       0.
119.      GRID    60                  10.       1.       0.041
120.      GRID    61                  10.       1.       0.082
121.      GRID    62                  22.       0.       0.
122.      GRID    63                  22.       0.       .041
123.      GRID    64                  22.       0.       .082
124.      GRID    65                  22.       1.       0.
125.      GRID    66                  22.       1.       .041
126.      GRID    67                  22.       1.       .082
127.      GRID    68                  22.       2.       0.
128.      GRID    69                  22.       2.       .041
129.      GRID    70                  22.       2.       .082
130.      GRID    71                  21.       2.       0.
131.      GRID    72                  21.       2.       .041
132.      GRID    73                  21.       2.       .082
133.      GRID    74                  20.       2.       0.
134.      GRID    75                  20.       2.       .041
135.      GRID    76                  20.       2.       .082
136.        1   4   6   7   9  10  12  13  14  15  16  17  19  20  22  23  25  26  28  29
137.       31  32  34  35  37  38  40  41  43  44  46  47  49  50  52  53  55  56  58  59
138.       61  62  64  65  67  68  70  71  73  74  76
139.        3   6   9  12
140.      8.1000+3  9.1000+3  8.9000+3  9.1000+3  0.1990+6  9.1000+3  8.9000+3  9.1000+3
141.      8.1000+3
142.     17.9000+3  9.1000+3  8.9000+3  9.1000+3  0.2320+6  9.1000+3  8.9000+3  9.1000+3
143.     17.9000+3
144.      3.1500+6  0.7800+6  0.6000+6  0.7800+6 29.2000+6  0.7800+6  0.6000+6  0.7800+6
145.      3.1500+6
146.        0.00      0.17      0.53      0.17      0.00      0.17      0.53      0.17
147.        0.00
148.        1   2   3   7   8   9  13  15  29  30  31  44  45  46  59  60  61  74  75  76
149.        1   2   3   4   5   6  13  14  17  18  19  32  33  34  47  48  49  62  63  64
150.       62  63  64  65  66  67  68  69  70  71  72  73  74  75  76
151.       62  63  64  65  66  67  68  69  70  71  72  73  74  75  76
152.       62  64  65  67  68  70  71  73  74  76
153.       62  64  65  67  68  70  71  73  74  76
154.       62  64  65  67  68  70  71  73  74  76
155.       62  64  65  67  68  70  71  73  74  76
156.       62  64  65  67  68  70  71  73  74  76
157.       62  64  65  67  68  70  71  73  74  76
158.       62  64  65  67  68  70  71  73  74  76
159.       62  64  65  67  68  70  71  73  74  76
160.       62  64  65  67  68  70  71  73  74  76
161.        1   2   3   7   8   9  13  15  29  30  31  44  45  46  59  60  61  74  75  76
162.        1   2   3   4   5   6  13  14  17  18  19  32  33  34  47  48  49  62  63  64
163.        0.001     0.001     0.002     0.002     0.002     0.002     0.005     0.005
164.      $BRKPT PUNCH=
165.      $FREE,6 18.
166.      $FREE,8 19.
167.      $COPY PCH3.,PCH2.
END DATA.
```

Table C-3

NAS1 FILE

```
$DATA,L NAS1.,NAS1.
$DATA 7 RL70-5-12/29-17:05:36
     1.       $HDG,N
     2.       $XQT *NASTRAN,LINK1
     3.       NASTRAN BUFFSIZE=871
     4.       ID NASTRAN,CELFE
     5.       APP DISP
     6.       SOL 9,0
     7.       TIME 60
     8.       DIAG 8,21
     9.       ALTER 2
    10.       $
    11.       $ ASSEMBLE AND SOLVE 11 PRIMARY VARIABLES
    12.       PARAM    //C,N,NOP/V,N,TRUE=-1 $
    13.       PARAM    //C,N,NOP/V,N,FALSE=+1 $
    14.       PARAM    //C,N,ADD/V,N,NUEL/V,Y,NUMNP $
    15.       OUTPUT2  ,,,,//C,N,-1/C,N,14 $
    16.       INPUTT2  /,,,,1/C,N,-1/C,N,15 $
    17.       $
    18.       LABEL OUTER $
    19.       PARAM    //C,N,SUB/V,Y,NOEQU/V,Y,NOEQU $
    20.       COND     BOTTOM1,NOEQU $
    21.       INPUTT2  /A1,B1,,,/C,N,0/C,N,15 $
    22.       ADD      A1,/AX $
    23.       ADD      B1,/BX $
    24.       $
    25.       LABEL INNER $
    26.       PARAM    //C,N,SUB/V,Y,NUMNP/V,Y,NUMNP $
    27.       COND     BOTTOM2,NUMNP $
    28.       EQUIV    AX,AT/FALSE $
    29.       EQUIV    BX,BT/FALSE $
    30.       INPUTT2  /A,B,,,/V,Y,BLOK/C,N,15 $
    31.       ADD      A,AX/AT $
    32.       EQUIV    AT,AX/TRUE $
    33.       ADD      B,BX/BT $
    34.       EQUIV    BT,BX/TRUE $
    35.       REPT INNER,31 $
    36.       $
    37.       LABEL    BOTTOM2 $
    38.       MATPRN   AT,BT,,,// $
    39.       SOLVE    AT,BT/X/V,Y,SYM/C,N,1/C,N,1/C,N,1 $
    40.       MATPRN   X,,,,// $
    41.       OUTPUT2  X,,,,//C,N,0/C,N,14 $
    42.       PARAM    //C,N,SUB/V,Y,NUMNP/V,N,NUEL $
    43.       REPT OUTER,11 $
    44.       $
    45.       LABEL  BOTTOM1 $
    46.       OUTPUT2, ,,,,//C,N,-9/C,N,14 $
    47.       $
    48.       ALTER 30
    49.       $
    50.       $ ASSEMBLE STIFFNESS MATRIX KGG
    51.       PARAM    //C,N,SUB/V,Y,NUMNP/V,N,NUEL $
    52.       INPUTT2  /,,,,/C,N,-1/C,N,17 $
    53.       LABEL KLOOP $
    54.       PARAM    //C,N,SUB/V,Y,NUMNP/V,Y,NUMNP $
```

(Continued)

Table C-3 (Continued)

```
55.     COND      KEND,NUMNP $
56.     EQUIV     KGGX,KGGG/FALSE $
57.     INPUTT2   /K1,K2,K3,,/V,Y,BLOK/C,N,17 $
58.     ADDS      K1,K2,K3,KGGX,/KGGG $
59.     EQUIV     KGGG,KGGX/TRUE $
60.     REPT KLOOP,32 $
61.     LABEL     KEND $
62.     MATPRN    KGGX,,,,,//-1 $
63.     $
64.     ALTER 141
65.     $
66.     $ OUTPUT DISP AND VELOCITY TO CELFE
67.     UMERGE    USETD,UDVT,/UVN/C,N,N/C,N,D/C,N,S $
68.     MATPRN    UVN,,,,//-1 $
69.     OUTPUT2   UVN,,,,//C,N,-1/C,N,12 $
70.     $
71.     ENDALTER
72.     CEND
73.     TITLE= 5.0 X 5.0 X 0.0820 INCH THICK PLATE
74.     SUBTITLE= ASSEMBLE AND SOLVE PRIMARY VARIABLES
75.     LABEL= ONE TIME STEP TRANSIENT ANALYSIS
76.     IC=1
77.     MPC=1
78.     SPC=1
79.     TSTEP=1
80.     DLOAD=1
81.     OUTPUT
82.     SET 1=62 THRU 76,206,231,236
83.     DISP=1
84.     VELO=1
85.     BEGIN BULK
86.     $ PCH1 CONTAINS DAREA, GRID AND TSTEP CARDS FROM CELFE, CHANGE EACH DT
87.     MADD,P PCH1,
88.     PARAM   BLOK    0
89.     PARAM   SYM     0
90.     PARAM   NOEQU   11
91.     PARAM   NUMNP   32
92.     CELAS2  1001    1.+3    236     1
93.     CELAS2  1002    1.+3    236     2
94.     CELAS2  1003    1.+3    236     3
95.     GRDSET                                                          456
96.     CORD2R          1       0       0.0     0.0     0.0     0.0     0.0     1.0+222
97.     +222            1.0     0.0     0.0
98.     PQUAD2          201     123     .082
99.     CQUAD2          203     201     175     204     210     172
100.    CQUAD2          204     201     204     205     211     210
101.    CQUAD2          205     201     205     206     212     211
102.    CQUAD2          208     201     172     210     216     169
103.    CQUAD2          209     201     210     211     217     216
104.    CQUAD2          210     201     211     212     218     217
105.    CQUAD2          211     201     163     166     220     219
106.    CQUAD2          212     201     166     169     221     220
107.    CQUAD2          213     201     169     216     222     221
108.    CQUAD2          214     201     216     217     223     222
109.    CQUAD2          215     201     217     218     224     223
110.    CQUAD2          216     201     219     220     226     225
```

Table C-3 (Continued)

```
111.    CQUAD2      217     201     220     221     227     226
112.    CQUAD2      218     201     221     222     228     227
113.    CQUAD2      219     201     222     223     229     228
114.    CQUAD2      220     201     223     224     230     229
115.    CQUAD2      221     201     225     226     232     231
116.    CQUAD2      222     201     226     227     233     232
117.    CQUAD2      223     201     227     228     234     233
118.    CQUAD2      224     201     228     229     235     234
119.    CQUAD2      225     201     229     230     236     235
120.    GRID   163  1       2.      .0      .041    0       6
121.    GRID   166  1       2.      1.      .041    0       6
122.    GRID   169  1       2.0     2.0     .041    0       6
123.    GRID   172  1       1.      2.      .041    0       6
124.    GRID   175  1       .0      2.0     .041    0       6
125.    GRID        204     1       .000    3.000   .041    0       6
126.    GRID        205     1       .000    4.000   .041    0       6
127.    GRID        206     1       .000    4.375   .041    0       6
128.    GRID        210     1       1.000   3.000   .041    0       6
129.    GRID        211     1       1.000   4.000   .041    0       6
130.    GRID        212     1       1.000   4.375   .041    0       6
131.    GRID        216     1       2.000   3.000   .041    0       6
132.    GRID        217     1       2.000   4.000   .041    0       6
133.    GRID        218     1       2.000   4.375   .041    0       6
134.    GRID        219     1       3.000   .000    .041    0       6
135.    GRID        220     1       3.000   1.000   .041    0       6
136.    GRID        221     1       3.000   2.000   .041    0       6
137.    GRID        222     1       3.000   3.000   .041    0       6
138.    GRID        223     1       3.000   4.000   .041    0       6
139.    GRID        224     1       3.000   4.375   .041    0       6
140.    GRID        225     1       4.000   .000    .041    0       6
141.    GRID        226     1       4.000   1.000   .041    0       6
142.    GRID        227     1       4.000   2.000   .041    0       6
143.    GRID        228     1       4.000   3.000   .041    0       6
144.    GRID        229     1       4.000   4.000   .041    0       6
145.    GRID        230     1       4.000   4.375   .041    0       6
146.    GRID        231     1       4.875   .000    .041    0       6
147.    GRID        232     1       4.875   1.000   .041    0       6
148.    GRID        233     1       4.875   2.000   .041    0       6
149.    GRID        234     1       4.875   3.000   .041    0       6
150.    GRID        235     1       4.875   4.000   .041    0       6
151.    GRID        236     1       4.875   4.375   .041    0       6
152.    MAT2   123  3.16+06 0.537+060.0     29.29+060.0     0.78+06 0.073
153.    CBAR   1    1       62      163     0.0     1.0     0.0     1
154.    CBAR   2    1       163     64      0.0     1.0     0.0     1
155.    CBAR   3    1       65      166     0.0     1.0     0.0     1
156.    CBAR   4    1       166     67      0.0     1.0     0.0     1
157.    CBAR   5    1       68      169     0.0     1.0     0.0     1
158.    CBAR   6    1       169     70      0.0     1.0     0.0     1
159.    CBAR   7    1       71      172     0.0     1.0     0.0     1
160.    CBAR   8    1       172     73      0.0     1.0     0.0     1
161.    CBAR   9    1       74      175     0.0     1.0     0.0     1
162.    CBAR   10   1       175     76      0.0     1.0     0.0     1
163.    PBAR   1    2       10.     10.     10.     20.
164.    MAT1   2    3.15+06         0.17    0.073
165.    MPC    1    163     1       1.0     63      1       -1.0
166.    MPC    1    163     2       1.0     63      2       -1.0
```

Table C-3 (Concluded)

```
  167.      MPC     1       163     3       1.0     63      3       -1.0
  168.      MPC     1       166     1       1.0     66      1       -1.0
  169.      MPC     1       166     2       1.0     66      2       -1.0
  170.      MPC     1       166     3       1.0     66      3       -1.0
  171.      MPC     1       169     1       1.0     69      1       -1.0
  172.      MPC     1       169     2       1.0     69      2       -1.0
  173.      MPC     1       169     3       1.0     69      3       -1.0
  174.      MPC     1       172     1       1.0     72      1       -1.0
  175.      MPC     1       172     2       1.0     72      2       -1.0
  176.      MPC     1       172     3       1.0     72      3       -1.0
  177.      MPC     1       175     1       1.0     75      1       -1.0
  178.      MPC     1       175     2       1.0     75      2       -1.0
  179.      MPC     1       175     3       1.0     75      3       -1.0
  180.      SPC1    1       15      1       2       3       7       8       9
  181.      SPC1    1       15      13      15      29      30      31      44
  182.      SPC1    1       15      45      46      59      60      61
  183.      SPC1    1       1       74      75      76
  184.      SPC1    1       24      1       2       3       4       5       6
  185.      SPC1    1       24      13      14      17      18      19      32
  186.      SPC1    1       24      33      34      47      48      49
  187.      SPC1    1       2       62      63      64
  188.      TLOAD1  1       2                       10
  189.      TABLED1 10                                                      +TD1
  190.      +TD1    0.      1.      1000.   1.      ENDT
  191.      ENDDATA
  192.      $ADD SP *NASTRAN *CONTRL
END DATA.
```

Table C-4

LOOP FILE

```
SYM

@DATA,L LOOP.
DATA 8R1 SL73R1 10/31/78 21:19:16 (0)
     1.     @ADD,P NAS1.
     2.     @BRKPT PUNCH$/PCH1
     3.     @ASG,A PCH2.
     4.     @ASG,A PCH3.
     5.     @USE 18.,PCH2.
     6.     @USE 19.,PCH3.
     7.     @XQT CELFE
     8.     +1
     9.     +1
    10.     @BRKPT PUNCH$
    11.     @FREE,B 18.
    12.     @FREE,B 19.
    13.     @COPY PCH3.,PCH2.
    14.     @ADD,P NAS1.
    15.     @BRKPT PUNCH$/PCH1
    16.     @ASG,A PCH2.
    17.     @ASG,A PCH3.
    18.     @USE 18.,PCH2.
    19.     @USE 19.,PCH3.
    20.     @XQT CELFE
    21.     +1
    22.     -1
    23.     @BRKPT PUNCH$
    24.     @FREE,B 18.
    25.     @FREE,B 19.
    26.     @COPY PCH3.,PCH2.
END DATA.
```

Table C-5

PCH2 FILE

```
SYM
@DATA,L PCH2.
DATA 8R1 SL73R1 10/31/78 21:19:18 (0)
     1.    GRID     1                .000     .000     .000
     2.    GRID     2                .000     .000     .041
     3.    GRID     3                .000     .000     .082
     4.    GRID     4                .150     .000     .000
     5.    GRID     5                .150     .000     .041
     6.    GRID     6                .150     .000     .082
     7.    GRID     7                .000     .150     .000
     8.    GRID     8                .000     .150     .041
     9.    GRID     9                .000     .150     .082
    10.    GRID    10                .150     .150     .000
    11.    GRID    11                .150     .150     .041
    12.    GRID    12                .150     .150     .082
    13.    GRID    13                .000     .000     .232
    14.    GRID    14                .150     .000     .232
    15.    GRID    15                .000     .150     .232
    16.    GRID    16                .150     .150     .232
    17.    GRID    17                .350     .000     .000
    18.    GRID    18                .350     .000     .041
    19.    GRID    19                .350     .000     .082
    20.    GRID    20                .350     .150     .000
    21.    GRID    21                .350     .150     .041
    22.    GRID    22                .350     .150     .082
    23.    GRID    23                .350     .350     .000
    24.    GRID    24                .350     .350     .041
    25.    GRID    25                .350     .350     .082
    26.    GRID    26                .150     .350     .000
    27.    GRID    27                .150     .350     .041
    28.    GRID    28                .150     .350     .082
    29.    GRID    29                .000     .350     .000
    30.    GRID    30                .000     .350     .041
    31.    GRID    31                .000     .350     .082
    32.    GRID    32                .500     .000     .000
    33.    GRID    33                .500     .000     .041
    34.    GRID    34                .500     .000     .082
    35.    GRID    35                .500     .350     .000
    36.    GRID    36                .500     .350     .041
    37.    GRID    37                .500     .350     .082
    38.    GRID    38                .500     .500     .000
    39.    GRID    39                .500     .500     .041
    40.    GRID    40                .500     .500     .082
    41.    GRID    41                .350     .500     .000
    42.    GRID    42                .350     .500     .041
    43.    GRID    43                .350     .500     .082
    44.    GRID    44                .000     .500     .000
    45.    GRID    45                .000     .500     .041
    46.    GRID    46                .000     .500     .082
    47.    GRID    47               1.000     .000     .000
    48.    GRID    48               1.000     .000     .041
    49.    GRID    49               1.000     .000     .082
    50.    GRID    50               1.000     .500     .000
    51.    GRID    51               1.000     .500     .041
    52.    GRID    52               1.000     .500     .082
    53.    GRID    53               1.000    1.000     .000
    54.    GRID    54               1.000    1.000     .041
```

(Continued)

Table C-5 (Concluded)

	SYM							
	55.	GRID	55			1.000	1.000	.082
	56.	GRID	56			.500	1.000	.000
	57.	GRID	57			.500	1.000	.041
	58.	GRID	58			.500	1.000	.082
	59.	GRID	59			.000	1.000	.000
	60.	GRID	60			.000	1.000	.041
	61.	GRID	61			.000	1.000	.082
	62.	GRID		62	1	2.000	.000	.000
	63.	GRID	63	1		2.000	.000	.041
	64.	GRID		64	1	2.000	.000	.082
	65.	GRID	65	1		2.000	1.000	.000
	66.	GRID		66	1	2.000	1.000	.041
	67.	GRID	67	1		2.000	1.000	.082
	68.	GRID		68	1	2.000	2.000	.000
	69.	GRID	69	1		2.000	2.000	.041
	70.	GRID		70	1	2.000	2.000	.082
	71.	GRID	71	1		1.000	2.000	.000
	72.	GRID		72	1	1.000	2.000	.041
	73.	GRID	73	1		1.000	2.000	.082
	74.	GRID		74	1	.000	2.000	.000
	75.	GRID	75	1		.000	2.000	.041
	76.	GRID		76	1	.000	2.000	.082

END DATA.