

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

AFFDL-TR-66-207
VOLUME II — SUPPLEMENT II

**FORMAT II.—SECOND VERSION OF FORTRAN
MATRIX ABSTRACTION TECHNIQUE**

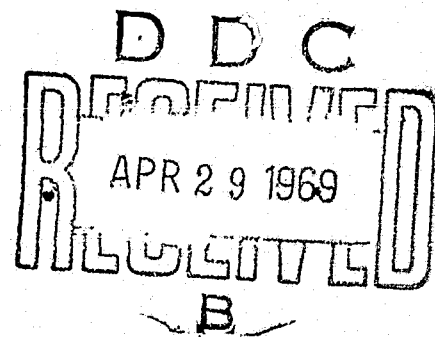
**VOLUME II.—SUPPLEMENT II. DESCRIPTION OF
DIGITAL COMPUTER PROGRAM SYSTEM/360**

C. G. HOOKS

*International Business Machines (IBM), Federal
Systems Division*

TECHNICAL REPORT AFFDL-TR-66-207, VOLUME II
SUPPLEMENT II

FEBRUARY 1969



This document has been approved for public
release and sale; its distribution is unlimited.

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

Reproduced by the
CLEARINGHOUSE
for Federal Scientific & Technical
Information Springfield Va. 22151

82

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This document has been approved for public release and sale;
its distribution is unlimited.

ACCESSION for		
CPSTI	WHITE SECTION	☒
DDC	BUFF SECTION	☐
UNANNOUNCED		☐
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODES		
DIST.	AVAIL. and/or	SPECIAL
1		

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

**FORMAT II.—SECOND VERSION OF FORTRAN
MATRIX ABSTRACTION TECHNIQUE**

**VOLUME II.—SUPPLEMENT II. DESCRIPTION OF
DIGITAL COMPUTER PROGRAM SYSTEM/360**

C. G. HOOKS

FEBRUARY 1969

**This document has been approved for public
release and sale; its distribution is unlimited.**

FOREWORD

This report was prepared by International Business Machines (IBM), Federal Systems Division, Huntsville, Alabama, under NASA Contract No. NAS8-14000 as documentation for IBM Instrument Unit (IU) Structural studies.

This report was coordinated with the Air Force Flight Dynamics Laboratory with Mr. J.R. Johnson, FDTR, Project Engineer, for AFFDL release.

The work reported herein was conducted during the period of January, 1968 through September, 1968. This report was released by the author for publication as an AFFDL Technical report in October, 1968. This documentation is the final report on the conversion of FORMAT II (Fortran Matrix Abstraction Technique-Second Version) for System/360 Model 75 use.

Test on the program conversion was performed to the extent of the NASA Contract No. NAS8-14000 requirements. Hence, the suitability of the program for general use cannot be guaranteed, nor can the program be assumed to be error free. IBM does not warrant use of this program.

This report has been reviewed and is approved.



FRANCIS J. IANTKE, JR.
Chief, Theoretical Mechanics Branch
Structures Division

ABSTRACT

A detailed description of the conversion effort for implementing FORMAT II on the System/360 using direct access I/O is presented. The conversion, coding, changes, and System/360 Job Control Language are presented in detail. A timing comparison with the FORMAT II 7094 timer is also presented. The data input of FORMAT II is unchanged except that SAVE statements are not usable at present. The output is unchanged except that some trace statements to show time of execution for each phase have been added. ()

TABLE OF CONTENTS

	<u>Page</u>
Conversion of FORMAT II to System/360	1
Coding Changes to Convert FORMAT II to System/360	6
System/360 JCL for Executing FORMAT II	44
Timing for Runs on the System/360 for FORMAT II	49
Revised Program Listing	50
Appendix-Illustrative Example	51
References	74

Conversion of FORMAT II to System/360

The FORMAT II Program (Reference (1)) has been converted to use direct access I/O for scratch units in place of sequential access I/O as the program was originally written.

The change to direct access was necessary since there were not enough tape drives available on the present system in Huntsville to allow for more than 5 scratch data sets.*

The conversion of the sequential I/O statements for scratch data sets to direct access I/O had to be done keeping the following three problems in mind:

1. The associated variable (position pointer for next record to be processed after each READ, WRITE, or FIND) would have to be available to all routines.
2. Once a particular set of data (matrix header) was located, the next set of records (matrix values) would have to read in a sequential manner if the present logic of the program was to remain unchanged.
3. In the case of output and input tapes, there would have to be two types of I/O statements, one for direct access and one for sequential, for each READ or WRITE.

The above three problems were solved in two ways. First, all of the scratch I/O statements of the FORMAT II routines were converted to

*Scratch data sets refer to all I/O except the normal system input and output data sets.

subroutine CALL's. Second, all of the scratch I/O statements of the routines in the USERXX (user oriented) packages were converted to direct access.

The use of subroutine CALL'S was chosen for the FORMAT II routines since a subroutine could be written which had multiple entries and thus all of the I/O could be combined in one routine. The use of one subroutine eliminated the problem of having to make the associated variable available to all routines. The problem of the number of I/O statements needed to be written for both direct access and sequential I/O was reduced since many of the original I/O statements used the same type of list (arguments). Thus, the number of different forms of entries to be written was far less than the number of original I/O statements. The problem of reading in a sequential manner was solved by using the previous value of the associated variable available after each I/O operation to read or write the next record.

The following is a sample of the change to the scratch I/O statements of the FORMAT II routines.

Original I/O Statement

READ (NINST) NUM,(WORK(I),I=1,NUM)

Subroutine Call

CALL RDO (NINST,NUM,WORK)

In RDO the READ was coded

ENTRY RDO (N,I1,A)

READ (N'NU(N)) I1,(A(I),I=1,I1)

where

A is dimensioned and NU is an array in the COMMON 'IOUNIT' to the associated variable for the FORTRAN data sets defined in the MAIN routine.

A subroutine calling sequence was generated in a similar manner for each different I/O list encountered. Additional arguments were required in the case of two dimensioned arrays. The name of the I/O routine is 'BACKSP' which is also the entry for back-spacing a record.

All of the associated variables were placed in a COMMON labeled 'IOUNIT' in order to communicate with the MAIN routine where the direct access I/O is initialized.

The provision for input and output tapes can be made by adding an array in COMMON to pass the data set type from the preprocessor routine MRES to the BACKSP routine and then coding the proper decision logic (for tape or disk) and the required sequential I/O statements. The provision for sequential I/O would only be necessary down through routine RD9 (see listing of BACKSP). Disk space could be cataloged and used in place of input and output tapes where possible and thus eliminate the need for distinguishing between tapes and disks. The scratch I/O statements of USERXX routines were just converted to direct access and the COMMON 'IOUNIT' was added since there was no need for input or output tapes and the writing of future user routines would be easier.

As a by-product of the use of direct access, a table was provided which keeps up with where each of the matrices is written for the execution phase. The table then eliminates the need for searching for a matrix since the FIND statement can be used directly to start at the beginning of a matrix. The routines MAIN, EUTL1, EUTL2, EUTL3, EUTL5, EUTL6, and MATR were the only routines changed for the above feature. The tables for keeping up with the matrices have a limit of 30 and are stored in the COMMON 'LOCATE'.

Three additional printouts were added in order to monitor execution and to aid in allocating scratch space on direct access.

The first printout is to monitor the completion of the PREP and EXEQ routines which are the main routines for the preprocessor and execution phases respectively. The second printout is of the time taken in the execution of each abstraction instruction. The abstraction instructions must be numbered. The third printout is a print at the end of each problem to show the maximum number of records written on each scratch data set.

The CLOCK routine referenced is a MAP360 assembly language routine which reads the internal timer (core locations 50, 51, and 52) on the System/360 and returns the time in seconds. If the clock routine is not available, then the following should be inserted:

```
FUNCTION CLOCK(X)
  CLOCK = 0.
  RETURN
END
```

The time for each run will vary under MVT since the time is actual and not task time but will still serve as an indicator in predicting run times.

Even if a CLOCK routine is not available, the print at the end of the execution of each abstraction instruction still serves as an indication of how far a problem has progressed in case the run time is exceeded.

Coding Changes to Convert FORMAT II to System/360

The following is a listing of all of the changes incorporated in the FORMAT II program.*

The updating of the decks was performed by use of an update routine in which the following control cards were used:

./CONTROL-----SEARCH=NAME

where NAME is the deck to be altered

./ALTER N1,N2

where the new cards following the alter card are to be inserted in place of the cards in the original deck with sequence numbers N1 through N2. If only N1 is present, the new cards are added before N1; and, if no cards follow the ./ALTER before the next ./ALTER or ./CONTROL is encountered, then there is a deletion.

*The entire subroutine 'BACKSP' and subroutines 'US05' through 'US09' are listed in the alter cards.

FORMAT II ALTER CARDS

./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MAIN	FMII0001
./ ALTER 1	FMII0002
COMMON NPIT, NPOT, KONST, NWORK, WORK(13000)	FMII0003
COMMON/OUNIT/ N1,N2,N3,N4,N5,N6,N7,N8,N9,NA,NB,NC	FMII0004
COMMON/LOCATE/LONAME(12,30,7),LOPUS(12,30),LOEND(12),	FMII0005
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0006
INTEGER NU(12), MWORK(1)	FMII0007
EQUIVALENCE (NU(1), N1)	FMII0008
./ ALTER 2	FMII0009
C	FMII0010
DEFINE FILE 1(500,200,U,N1), 2(2500,200,U,N2), 3(2500,200,U,N3),	FMII0011
1 4(2500,200,U,N4), 7(2500,200,U,N7), 8(2500,200,U,N8),	FMII0012
2 9(3500,200,U,N9),	FMII0013
3 10(2500,200,U,NA),11(2500,200,U,NB),12(2500,200,U,NC)	FMII0014
T1 = CLOCK(T1)	FMII0015
N1 = 1	FMII0016
N2 = 1	FMII0017
N3 = 1	FMII0018
N4 = 1	FMII0019
N5 = 1	FMII0020
N6 = 1	FMII0021
N7 = 1	FMII0022
N8 = 1	FMII0023
N9 = 1	FMII0024
NA = 1	FMII0025
NB = 1	FMII0026
NC = 1	FMII0027
DO 1 I = 1,2917	FMII0028
1 LONAME(I,1,1) = 0	FMII0029
DO 6 I = 1,13000	FMII0030
6 WORK(I) = 0.0	FMII0031
WRITE (6,20)	FMII0032
20 FORMAT ('CORE ZEROED' /)	FMII0033
MAXDTA = 30	FMII0034
NPIT = 5	FMII0035
NPOT = 6	FMII0036
KONST = 2000	FMII0037
NWORK = 13000	FMII0038
C	FMII0039
J = 1	FMII0040
DO 2 I = 1,12	FMII0041
IF (I.EQ. 5) GO TO 2	FMII0042
IF (I.EQ. 6) GO TO 2	FMII0043
N = I	FMII0044
CALL WDI (N,N,N,J,N)	FMII0045
WRITE (NPOT,3) N, NU(N)	FMII0046
3 FORMAT (14, 'INITIALIZED. ASSOCIATED VARIABLE = ', 14)	FMII0047
CALL REWIND (N)	FMII0048
2 CONTINUE	FMII0049
TIME = CLOCK(TIME)	FMII0050
C	FMII0051

T2 = CLOCK(T2) - T1	FMIIC052
WRITE (6,7) T2	FMIIC053
7 FORMAT (' T2 = ',F8.3)	FMIIC054
./ ALTER 3,3	FMIIC055
TIME = CLOCK(TIME) - TIME	FMIIC056
WRITE (6,21) NINST, IPRINT, GO	FMIIC057
21 FORMAT (' 1PREP RETURNED' /	FMIIC058
1 'ONINST = ', I8 /	FMIIC059
2 'OIPRINT = ', I8 /	FMIIC060
3 'OGO = ', L8)	FMIIC061
WRITE(6,30) TIME	FMIIC062
30 FORMAT (' PHASE1 TIME = ',F8.3)	FMIIC063
IF(.NOT.GO) GO TO 34	FMIIC064
TIME = CLOCK(TIME)	FMIIC065
./ ALTER 5	FMIIC066
TIME = CLOCK(TIME) - TIME	FMIIC067
WRITE (6,22) NINST, IPRINT	FMIIC068
22 FORMAT (' 1EXEQ RETURNED' /	FMIIC069
1 'ONINST = ', I8 /	FMIIC070
2 'OIPRINT = ', I8)	FMIIC071
WRITE(6,33) TIME	FMIIC072
33 FORMAT(' PHASE2 TIME = ',F8.3)	FMIIC073
WRITE(6,31) LOMAX	FMIIC074
WRITE(6,32) LONUMX	FMIIC075
31 FORMAT (' LOMAX ',I2I8)	FMIIC076
32 FORMAT (' LONUMX',I2I8)	FMIIC077
TIME = CLOCK(TIME)	FMIIC078
./ ALTER 6,6	FMIIC079
34 TOTIME = CLOCK(TOTIME) - T1	FMIIC080
WRITE (6,35) TOTIME	FMIIC081
35 FORMAT (' TOTAL TIME IS ',F8.3)	FMIIC082
STOP	FMIIC083
END	FMIIC084
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=BACKSP,LABEL=98900001,NUCOMP	FMIIC085
SUBROUTINE BACKSP (N)	FMIIC086
COMMON/IOUN!T/ NU(12)	FMIIC087
REAL A(1), B(1)	FMIIC088
INTEGER JJJ(ID,IE,IF)	FMIIC089
INTEGER II(1),III(IA,1),IIJ(IB,1)	FMIIC090
INTEGER IIK(IC,1)	FMIIC091
C	FMIIC092
FIND (N' NU(N)-1)	FMIIC093
RETURN	FMIIC094
C	FMIIC095
ENTRY REWND (N)	FMIIC096
FIND (N' 1)	FMIIC097
RETURN	FMIIC098
C	FMIIC099
ENTRY RDO (N, I1, A)	FMIIC100
READ (N'NU(N), ERR=10) I1, (A(I), I = 1,I1)	FMIIC101
RETURN	FMIIC102
C	FMIIC103
ENTRY RDI (N, I1, I2, I3, A)	FMIIC104
READ (N' NU(N), ERR = 10) I1, I2, I3, (A(I), I = 1,I3)	FMIIC105
RETURN	FMIIC106
C	FMIIC107
ENTRY WDI (N, I1, I2, I3, B)	FMIIC108

	IF (I3) 11, 11, 12	FMII0109
11	I3M = -I3	FMII0110
	WRITE (N' NU(N)) I1, I2, I3M, (B(I), I = 1, I3M)	FMII0111
	RETURN	FMII0112
12	WRITE (N' NU(N)) I1, I2, I3, (B(I), I = 1, I3)	FMII0113
	RETURN	FMII0114
C	ENTRY RD2(N, I1, I2, I3, I4, I5)	FMII0115
	READ(N' NU(N)) I1, I2, I3, I4, I5	FMII0116
	RETURN	FMII0117
C	ENTRY WD2 (N, I1, I2, I3, I4, I5)	FMII0118
	WRITE (N' NU(N)) I1, I2, I3, I4, I5	FMII0119
	RETURN	FMII0120
C	ENTRY RD3 (N, I1)	FMII0121
	READ (N' NU(N), ERR = 10) I1	FMII0122
	RETURN	FMII0123
C	ENTRY RD4 (N, I1, I2, I3, A, I4, I5)	FMII0124
	READ (N' NU(N), ERR = 10) I1, I2, I3, (A(I-2), I=3, I3), I4, I5	FMII0125
	RETURN	FMII0126
C	ENTRY WD4 (N, I1, I2, I3, B, I4, I5)	FMII0127
	WRITE (N' NU(N)) I1, I2, I3, (B(I-2), I=3, I3), I4, I5	FMII0128
	RETURN	FMII0129
C	ENTRY RD5 (N, I1, I2)	FMII0130
	READ (N' NU(N), ERR = 10) I1, I2	FMII0131
	RETURN	FMII0132
C	ENTRY RD6 (N, I1, A, I2)	FMII0133
	READ (N' NU(N), ERR = 10) I1, (A(I), I = 1, I2)	FMII0134
	RETURN	FMII0135
C	ENTRY WD6 (N, I1, B, I2)	FMII0136
	IF (I2) 13, 13, 14	FMII0137
13	I2M = -I2	FMII0138
	WRITE(N' NU(N)) I1, (B(I), I = 1, I2M)	FMII0139
	RETURN	FMII0140
14	WRITE(N' NU(N)) I1, (B(I), I = 1, I2)	FMII0141
	RETURN	FMII0142
C	ENTRY RD7 (N, A, I1)	FMII0143
	READ (N' NU(N), ERR = 10) (A(I), I = 1, I1)	FMII0144
	RETURN	FMII0145
C	ENTRY WD7 (N, B, I1)	FMII0146
	WRITE (N' NU(N)) (B(I), I = 1, I1)	FMII0147
	RETURN	FMII0148
C	ENTRY RD8 (N, A, B, I1)	FMII0149
	READ (N' NU(N), ERR = 10) (A(I), B(I), I = 1, I1)	FMII0150
	RETURN	FMII0151
C	ENTRY RD9 (N, I1, I2, I3, I4)	FMII0152
	READ (N' NU(N), ERR = 10) I1, I2, I3, I4	FMII0153
		FMII0154
		FMII0155
		FMII0156
		FMII0157
		FMII0158
		FMII0159
		FMII0160
		FMII0161
		FMII0162
		FMII0163
		FMII0164
		FMII0165

C	RETURN	FMIIC166
		FMIIC167
	ENTRY WD9 (N, I1, I2, I3, I4)	FMIIC168
	WRITE (N'NU(N)) I1, I2, I3, I4	FMIIC169
	RETURN	FMIIC170
C		FMIIC171
	ENTRY WD10 (N, IA, I1, I2, I11, I3)	FMIIC172
	WRITE (N'NU(N)) (I11(K, I2), K=1, I1), I3	FMIIC173
	RETURN	FMIIC174
C		FMIIC175
	ENTRY WD11(N, I1, I2, I3, I4, I1)	FMIIC176
	WRITE (N'NU(N)) I1, I2, I3, (I1(I), I=1, I4)	FMIIC177
	RETURN	FMIIC178
C		FMIIC179
	ENTRY RD12 (N, IA, I1, I2, I11)	FMIIC180
	READ (N'NU(N)) (I11(I, J), I=1, I1), J=1, I2)	FMIIC181
	RETURN	FMIIC182
C		FMIIC183
	ENTRY WD12 (N, IA, I1, I2, I11)	FMIIC184
	WRITE (N'NU(N)) (I11(I, J), I=1, I1), J=1, I2)	FMIIC185
	RETURN	FMIIC186
C		FMIIC187
	ENTRY RD13(N, I1, I2, I3, I4, I1)	FMIIC188
	READ (N'NU(N)) I1, I2, I3, (I1(I), I=1, I4)	FMIIC189
	RETURN	FMIIC190
C		FMIIC191
	ENTRY RD14 (N, IA, I1, I2, I11, I3, I4)	FMIIC192
	READ (N'NU(N)) (I11(J, I2), J=1, I1), I3, I4	FMIIC193
	RETURN	FMIIC194
C		FMIIC195
	ENTRY RD15(N, I1, I2, I3)	FMIIC196
	READ (N'NU(N)) I1, I2, I3	FMIIC197
	RETURN	FMIIC198
C		FMIIC199
	ENTRY WD15 (N, I1, I2, I3)	FMIIC200
	WRITE (N'NU(N)) I1, I2, I3	FMIIC201
	RETURN	FMIIC202
C		FMIIC203
	ENTRY WD16 (N, IA, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I11, I11)	FMIIC204
	WRITE (N'NU(N)) I1, I2, I3, I4, I5, I6, I7,	FMIIC205
	1 ((I11(I, I8), I=1, I9), J=1, I10), (I1(I), I=1, I11)	FMIIC206
	RETURN	FMIIC207
C		FMIIC208
	ENTRY RD17(N, I1, I2, I3, I1)	FMIIC209
	READ (N'NU(N)) I1, I2, I3, (I1(I), I=1, I3)	FMIIC210
	RETURN	FMIIC211
C		FMIIC212
	ENTRY WD18(N, I1, I2, I3, I4, I5, I6, I7)	FMIIC213
	WRITE (N'NU(N)) I1, I2, I3, I4, I5, I6, I7	FMIIC214
	RETURN	FMIIC215
C		FMIIC216
	ENTRY WD19 (N, IA, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I11, I12, I13,	FMIIC217
	1 I14, I15, I16)	FMIIC218
	WRITE (N'NU(N)) I1, I2, I3, I4, I5, I6, I7, ((I11(I, I8), I=1, I9), K=1, I10),	FMIIC219
	1 I11, I12, I13, I14, I15, I16	FMIIC220
	RETURN	FMIIC221
C		FMIIC222

	ENTRY RD20 (N,IA,I1,I2,III)	FMII0223
	READ (N*NU(N)) I1, (III(I,I2),I=1,I1)	FMII0224
	RETURN	FMII0225
C		FMII0226
	ENTRY WD20 (N,IA,I1,I2,III)	FMII0227
	WRITE (N*NU(N)) I1, (III(I,I2),I=1,I1)	FMII0228
	RETURN	FMII0229
C		FMII0230
	ENTRY WD21 (N,I1,I2,I3,I4,I5,I6,I7,I8)	FMII0231
	WRITE(N*NU(N)) I1,I2,I3,I4,I5,I6,I7,I8	FMII0232
	RETURN	FMII0233
C		FMII0234
	ENTRY WD22 (N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,III,I9,I10,IIJ,I11, 1 I12)	FMII0235
	1 WRITE(N*NU(N)) I1,I2,I3,I4,I5,I6, ((III(I,J),I=1,I7),J=1,I8), 1 ((IIJ(I,J),I=1,I9),J=1,I10), I11,I12	FMII0237
	RETURN	FMII0238
C		FMII0239
		FMII0240
	ENTRY WD23 (N,IA,I1,I2,I3,I4,I5,I6,I7,I8,III,I9,II)	FMII0241
	WRITE (N*NU(N)) I1,I2,I3,I4,I5,I6, ((III(I,J),I=1,I7),J=1,I8), 1 (II(I),I=1,I9)	FMII0243
	RETURN	FMII0244
C		FMII0245
	ENTRY WD24 (N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,I9,III,I10,I11,IIJ)	FMII0246
	1 WRITE (N*NU(N)) I1,I2,I3,I4,I5,I6,((III(I,J),I=1,I7),J=1,I8,I9), 1 (IIJ(I,I11), I=1,I10)	FMII0247
	RETURN	FMII0248
C		FMII0249
		FMII0250
	ENTRY RD25(N,C,I1,I2,I3,I4,I5)	FMII0251
	READ(N*NU(N))C,I1,I2,I3,I4,I5	FMII0252
	RETURN	FMII0253
C		FMII0254
	ENTRY RD26(N,I1,I2,I3,I4,I5,I6,II)	FMII0255
	READ(N*NU(N))I1,I2,I3,I4,I5,I6,(II(J),J=1,I6)	FMII0256
	RETURN	FMII0257
C		FMII0258
	ENTRY RD27(N,I1,II)	FMII0259
	READ (N*NU(N)) (II(J),J=1,I1)	FMII0260
	RETURN	FMII0261
C		FMII0262
	ENTRY WD27(N,I1,II)	FMII0263
	WRITE(N*NU(N))(II(J),J=1,I1)	FMII0264
	RETURN	FMII0265
C		FMII0266
	ENTRY RD28(N,I1,I2,I3,II)	FMII0267
	READ(N*NU(N))I1,I2,I3, (II(K),K=1,I3)	FMII0268
	RETURN	FMII0269
C		FMII0270
	ENTRY WD28(N,I1,I2,I3,II)	FMII0271
	WRITE(N*NU(N))I1,I2,I3, (II(J),J=1,I3)	FMII0272
	RETURN	FMII0273
C		FMII0274
	ENTRY RD29 (N,IA,I1,I2,III)	FMII0275
	READ(N*NU(N))(III(I,I2),I=1,I1)	FMII0276
	RETURN	FMII0277
C		FMII0278
	ENTRY WD30 (N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,I9,III,I10,I11,IIJ,	FMII0279

1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,J),I=1,I7),J=I8,I9),	I12,I11	FMI10280
1	(IIJ(I,I11),I=1,I10),(II(I),I=1,I12)		FMI10281
1	RETURN		FMI10282
C	ENTRY WD31(N,IA,IB,IC,I1,I2,I3,I4,I5,I6,I7,I8,III,I9,I10,IIJ,		FMI10283
1	I11,I12,IIK,I13,II)		FMI10284
1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,I8),I=1,I7),		FMI10285
1	(IIJ(I,I10),I=1,I9),(IIK(I,I12),I=1,I11),		FMI10286
2	(II(I),I=1,I13)		FMI10287
1	RETURN		FMI10288
C	ENTRY WD32(N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,I9,III,I10,I11,I12,		FMI10289
1	IIJ,I13,I14)		FMI10290
1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,J),I=1,I7),J=I8,I9),		FMI10291
1	(IIJ(I,J),I=1,I10),J=I11,I12),I13,I14		FMI10292
1	RETURN		FMI10293
C	ENTRY WD33 (N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,I9,III,I10,I11,I12,		FMI10294
1	IIJ)		FMI10295
1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,J),I=1,I7),J=I8,I9),		FMI10296
1	(IIJ(I,J),I=1,I10),J=I11,I12)		FMI10297
1	RETURN		FMI10298
C	ENTRY WD34(N,IA,I1,I2,I3,I4,I5,I6,I7,I8,I9,III)		FMI10299
1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,J),I=1,I7),J=I8,I9)		FMI10300
1	RETURN		FMI10301
C	ENTRY RD35 (N,IA,I1,I2,I3,I4,III)		FMI10302
1	READ (N'NU(N)) ((III(I,J), I=I1,I2),J=I3,I4)		FMI10303
1	RETURN		FMI10304
C	ENTRY RD36 (N,IA,I1,I2,I3,I4,I5,I6,I7,III)		FMI10305
1	READ (N'NU(N)) I1,I2,I3, (III(I,J),I=I4,I5),J=I6,I7)		FMI10306
1	RETURN		FMI10307
C	ENTRY WD37 (N,I1,I2,I3,I4,A,I5,I6,I7,II)		FMI10308
1	WRITE (N'NU(N)) I1,I2,I3, (A(I+I4),II(I),I=I5,I6,I7)		FMI10309
1	RETURN		FMI10310
C	ENTRY WD38 (N,ID,IE,IF,I1,I2,I3,JJJ)		FMI10311
1	WRITE (N'NU(N)) I1, (JJJ(I,I2,I3),I=1,I1)		FMI10312
1	RETURN		FMI10313
C	ENTRY WD39(N,IA,IB,I1,I2,I3,I4,I5,I6,I7,I8,I9,III,I10,I11,I12,		FMI10314
1	IIJ,I13)		FMI10315
1	WRITE (N'NU(N)) I1,I2,I3,I4,I5,I6, (III(I,J),I=1,I7),J=I8,I9),		FMI10316
1	(IIJ(I,J),I=1,I10),J=I11,I12),I13		FMI10317
1	RETURN		FMI10318
C	ENTRY RD40 (N,I1,I2,I3,I4,I5,A,I6,I7,B)		FMI10319
1	READ (N'NU(N)) I1,I2,I3,(A(I),I=I4,I5),(B(J),J=I6,I7)		FMI10320
1	RETURN		FMI10321
C	ENTRY WD41 (N,I1,I2,I3,I4,I5,I6,I7,I8,II,I9)		FMI10322
1	WRITE(N'NU(N)) I1,I2,I3,I4,I5,I6,(II(I),I=I7,I8),I9		FMI10323
1	RETURN		FMI10324

C	FMII0337
C	FMII0338
C	FMII0339
C	FMII0340
10 WRITE (6, 6) N	FMII0341
6 FORMAT ('110 ERROR ON UNIT', I9)	FMII0342
STOP 16	FMII0343
END	FMII0344
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PUTL1	FMII0345
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PUTL2	FMII0346
./ ALTER 11,11	FMII0347
IF(IARRAY(K) .NE. KARRAY(K)) RETURN	FMII0348
10 CONTINUE	FMII0349
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PUTL3	FMII0350
./ ALTER 17,17	FMII0351
IF(IARRAY(ID) .EQ. LINK(I)) GO TO 20	FMII0352
10 CONTINUE	FMII0353
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PUTL4	FMII0354
./ ALTER 25,25	FMII0355
IF (CARD(IK) .EQ. ALPHA(I)) GO TO 150	FMII0356
100 CONTINUE	FMII0357
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PREP	FMII0358
./ ALTER 22,22	FMII0359
IF(EQUAL) GO TO (100, 200, 300, 400, 500) , K	FMII0360
40 CONTINUE	FMII0361
./ ALTER 53,54	FMII0362
CALL REWIND(NPREP)	FMII0363
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MRES	FMII0364
./ ALTER 18,18	FMII0365
NUMR = 10	FMII0366
./ ALTER 29,29	FMII0367
KONFIG (2, 2) = IOUTIL	FMII0368
./ ALTER 34,34	FMII0369
KONFIG (2, 3) = IOUTIL	FMII0370
./ ALTER 39,39	FMII0371
KONFIG (2, 4) = IOUTIL	FMII0372
./ ALTER 44,44	FMII0373
KONFIG (2, 5) = IOUTIL	FMII0374
./ ALTER 63	FMII0375
KONFIG (1, 9) = 11	FMII0376
KONFIG (2, 9) = IOUTIL	FMII0377
KONFIG (3, 9) = NDISK	FMII0378
KONFIG (4, 9) = NA	FMII0379
KONFIG (5, 9) = INF	FMII0380
KONFIG (1, 10) = 12	FMII0381
KONFIG (2, 10) = IOUTIL	FMII0382
KONFIG (3, 10) = NDISK	FMII0383
KONFIG (4, 10) = NB	FMII0384
KONFIG (5, 10) = INF	FMII0385
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MRES1	FMII0386
./ ALTER 23,23	FMII0387
10 READ (NPIT,15,END=700) CARD	FMII0388
./ ALTER 39,39	FMII0389
IF(EQUAL) GO TO (70,61),K	FMII0390
59 CONTINUE	FMII0391
./ ALTER 57,57	FMII0392
IF(EQUAL) GO TO (10, 10, 150) , I	FMII0393

80 CONTINUE	FMII0394
./ ALTER 75,75	FMII0395
IF(EQUAL) GO TO (250, 275) , K	FMII0396
210 CONTINUE	FMII0397
./ ALTER 105,105	FMII0398
IF(KONFIG(1,J) .EQ. NUM) GO TO 430	FMII0399
420 CONTINUE	FMII0400
./ ALTER 142	FMII0401
700 IERROR = 3	FMII0402
RETURN	FMII0403
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MRES11	FMII0404
./ ALTER 22,22	FMII0405
IF(CARD(I) .EQ. COMMA) GO TO (100,200,300,400,100), ICOUNT	FMII0406
60 CONTINUE	FMII0407
./ ALTER 40,40	FMII0408
IF(EQUAL) GO TO 230	FMII0409
215 CONTINUE	FMII0410
./ ALTER 48,48	FMII0411
IF(EQUAL) GO TO 330	FMII0412
310 CONTINUE	FMII0413
./ ALTER 56,56	FMII0414
IF(CHN(J) .EQ. CARD(LI)) GO TO 420	FMII0415
410 CONTINUE	FMII0416
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MRES2	FMII0417
./ ALTER 46,46	FMII0418
CALL REWND(NPREP)	FMII0419
./ ALTER 63,63	FMII0420
CALL WD20(NPREP,5,IFIVE,IX,KONFIG)	FMII0421
./ ALTER 76,76	FMII0422
CALL WD20(NPREP,5,IFIVE,IX,KONFIG)	FMII0423
./ ALTER 87,87	FMII0424
CALL WD20(NPREP,5,IFIVE,IX,KONFIG)	FMII0425
./ ALTER 89,90	FMII0426
CALL REWND(NDATA)	FMII0427
CALL REWND(NINST)	FMII0428
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PROB	FMII0429
./ ALTER 43,43	FMII0430
IF(CARD(KOMMA) .EQ. COMMA) GO TO 60	FMII0431
50 CONTINUE	FMII0432
./ ALTER 98,98	FMII0433
IF(CARD(LEFT) .EQ. PARENL) GO TO 350	FMII0434
330 CONTINUE	FMII0435
./ ALTER 106,106	FMII0436
IF(CARD(J) .EQ. PARENR) LIMIT = J	FMII0437
360 CONTINUE	FMII0438
./ ALTER 156,156	FMII0439
IF(CARD(KOMMA) .EQ. COMMA) GO TO 630	FMII0440
610 CONTINUE	FMII0441
./ ALTER 211,211	FMII0442
IF(EQUAL) GO TO 830	FMII0443
810 CONTINUE	FMII0444
./ ALTER 213,213	FMII0445
820 CALL WD38(NPREP,7,500,2,ISEVEN,I,J,NAME)	FMII0446
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST	FMII0447
./ ALTER 56,56	FMII0448
IF(EQUAL) GO TO 200	FMII0449
150 CONTINUE	FMII0450

./ ALTER 102,102	FMI10451
IF(CARD(I) .EQ. EQUALS) GO TO 2000	FMI10452
260 CONTINUE	FMI10453
./ ALTER 107,107	FMI10454
IF(CARD(K) .EQ. PAREN1) GO TO 350	FMI10455
300 CONTINUE	FMI10456
./ ALTER 145,145	FMI10457
IF(CARD(IDT) .EQ. DOT) GO TO 2030	FMI10458
2010 CONTINUE	FMI10459
./ ALTER 151,151	FMI10460
IF(CARD(IDT) .EQ. DOT) GO TO 2050	FMI10461
2040 CONTINUE	FMI10462
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST01	FMI10463
./ ALTER 23,23	FMI10464
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMI10465
50 CONTINUE	FMI10466
./ ALTER 52,52	FMI10467
CALL WD41 (NPREP,ISTNO,NOPC,IONE,IZERO,IONE,ITOT,1,7,MATRIX,INTG)	FMI10468
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST02	FMI10469
./ ALTER 22,22	FMI10470
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMI10471
50 CONTINUE	FMI10472
./ ALTER 57,57	FMI10473
IF(CARD(IPT) .EQ. COMMA) GO TO 1150	FMI10474
1100 CONTINUE	FMI10475
./ ALTER 69,72	FMI10476
CALL WD22 (NPREP,7,7,ISTNO,NOPC,NUMIN,IZERO,NUMSC,ITOT,7,NUMIN,	FMI10477
1 MATRIX,6,2,ROWCOL,IPSPEC,FNUMB)	FMI10478
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST03	FMI10479
./ ALTER 18,18	FMI10480
IF(CARD(K) .EQ. PAREN1) GO TO 50	FMI10481
30 CONTINUE	FMI10482
./ ALTER 27,27	FMI10483
IF(CARD(K) .EQ. COMMA) GO TO 250	FMI10484
200 CONTINUE	FMI10485
./ ALTER 36,37	FMI10486
CALL WD23 (NPREP,7,ISTNO,NOPC,NUMIN,IZERO,ISIX,ITOT,7,NUMIN,	FMI10487
1 MATRIX,6,NAME)	FMI10488
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INSTFP	FMI10489
./ ALTER 17,17	FMI10490
IF(ARRAY(I) .EQ. E) GO TO 100	FMI10491
50 CONTINUE	FMI10492
./ ALTER 28,28	FMI10493
IF(ARRAY(I) .EQ. DOT) GO TO 200	FMI10494
150 CONTINUE	FMI10495
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST10	FMI10496
./ ALTER 28,28	FMI10497
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMI10498
50 CONTINUE	FMI10499
./ ALTER 51,52	FMI10500
CALL WD24 (NPREP,7,7,ISTNO,NOPC,ITWO,IONE,IZERO,ITOT,7,2,3,	FMI10501
1 MATRIX,7,1,MATRIX)	FMI10502
./ ALTER 54,56	FMI10503
2050 CALL WD30 (NPREP,7,7,ISTNO,NOPC,ITHREE,IONE,ITWO,ITOT,7,2,4,	FMI10504
1 MATRIX,7,1,MATRIX,2,INTG)	FMI10505
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST16	FMI10506
./ ALTER 24,24	FMI10507

IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0508
50 CONTINUE	FMII0509
./ ALTER 37,41	FMII0510
2000 CALL WD31 (NPREP,7,7,7,ISTNO,NOPC,ITWO,IONE,ITWO,ITOT,7,2,MATRIX,	FMII0511
1 7,3,MATRIX,7,1,MATRIX,2,INTG)	FMII0512
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST33	FMII0513
./ ALTER 26,26	FMII0514
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0515
50 CONTINUE	FMII0516
./ ALTER 51,53	FMII0517
CALL WD32 (NPREP,7,7,ISTNO,NOPC,NUMIN,ITWO,ITWO,ITOT,7,3,N,	FMII0518
1 MATRIX,7,1,2,MATRIX,CUTOFF,K)	FMII0519
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST20	FMII0520
./ ALTER 22,22	FMII0521
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0522
50 CONTINUE	FMII0523
./ ALTER 30,31	FMII0524
CALL WD24 (NPREP,7,7,ISTNO,NOPC,IONE,IONE,IZERO,ITOT,7,2,2,MATRIX,	FMII0525
1 7,1,MATRIX)	FMII0526
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST31	FMII0527
./ ALTER 20,20	FMII0528
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0529
50 CONTINUE	FMII0530
./ ALTER 30,31	FMII0531
2000 CALL WD24 (NPREP,7,7,ISTNO,NOPC,ITWO,IONE,IZERO,ITOT,7,2,3,MATRIX,	FMII0532
1 7,1,MATRIX)	FMII0533
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST32	FMII0534
./ ALTER 21,21	FMII0535
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0535
50 CONTINUE	FMII0537
./ ALTER 33,35	FMII0538
2000 CALL WD39 (NPREP,7,7,ISTNO,NOPC,ITWO,IONE,IONE,ITOT,7,2,3,MATRIX,	FMII0539
1 7,1,1,MATRIX,INTG)	FMII0540
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST41	FMII0541
./ ALTER 22,22	FMII0542
IF(CARD(IPT) .EQ. SYMBOL(IC)) GO TO 75	FMII0543
50 CONTINUE	FMII0544
./ ALTER 35,37	FMII0545
2000 CALL WD39 (NPREP,7,7,ISTNO,NOPC,IONE,ITWO,IONE,ITOT,7,3,3,MATRIX,	FMII0546
1 7,1,2,MATRIX,INTG)	FMII0547
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INST90	FMII0548
./ ALTER 14,14	FMII0549
IF(CARD(M) .EQ. ESIGN) GO TO 50	FMII0550
30 CONTINUE	FMII0551
./ ALTER 18,18	FMII0552
IF(CARD(K) .EQ. COMMA) GO TO 90	FMII0553
70 CONTINUE	FMII0554
./ ALTER 35,35	FMII0555
IF(CARD(K) .EQ. COMMA) GO TO 200	FMII0556
170 CONTINUE	FMII0557
./ ALTER 55,57	FMII0558
CALL WD33(NPREP,7,7,ISTNO,NOPC,NUMIN,NUMOT,IZERO,ITOT,7,J,LIMIT2,	FMII0559
1 MATRIX,7,1,NUMOT,MATRIX)	FMII0560
./ ALTER 59,60	FMII0561
250 CALL WD34 (NPREP,7,ISTNO,NOPC,NUMIN,IZERO,IZERO,ITOT,7,1,NUMIN,	FMII0562
1 MATRIX)	FMII0563
./ ALTER 62,63	FMII0564

260 CALL WD34 (NPREP,7,ISTNO,NOPC,IZERO,NUMOT,IZERO,ITOT,	FMII0565
1 7,1,NUMOT,MATRIX)	FMII0566
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MATR	FMII0567
./ ALTER 26	FMII0568
COMMON/IOUNIT/NU(12)	FMII0569
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0570
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0571
./ ALTER 27	FMII0572
DO 1 IXY = 1,12800	FMII0573
1 WORK(IXY) = 0.	FMII0574
./ ALTER 43,43	FMII0575
DO 1005 K=1,MAXDTA	FMII0576
LOPOS(NDATA,K) = 0	FMII0577
DO 1005 L=1,7	FMII0578
1005 LONAME(NDATA,K,L)=0	FMII0579
LOEND(NDATA)=0	FMII0580
LOMAX(NDATA)=0	FMII0581
LONUMX(NDATA)=0	FMII0582
WRITE (NDATA*NU(NDATA)) MNUS10,IZERO,NINE,(IZERO,K=1,9)	FMII0583
./ ALTER 177,178	FMII0584
230 CONTINUE	FMII0585
DO 1001 K= 1, MAXDTA	FMII0586
IF (LOPOS(NDATA,K)) 1001,1002,1001	FMII0587
1001 CONTINUE	FMII0588
WRITE(NPOT,1003) NDATA	FMII0589
1003 FORMAT ('1 MATR- 30 MATRICES ALREADY ON UNIT',I8)	FMII0590
STOP	FMII0591
1002 LOPOS(NDATA,K)=NU(NDATA)	FMII0592
LONAME(NDATA,K,7) = IONE	FMII0593
DO 1004 L=1,6	FMII0594
1004 LONAME(NDATA,K,L)=NAME(L)	FMII0595
LONUMX(NDATA)= MAXO(LONUMX(NDATA),K)	FMII0596
WRITE (NDATA*NU(NDATA)) MINUS1,IZERO,NINE,(NAME(K),K=1,6),	FMII0597
1 IONE,IMAX,JMAX	FMII0598
WRITE (NPREP*NU(NPREP)) (NAME(IX),IX=1,6), IONE,IMAX,JMAX	FMII0599
./ ALTER 216,216	FMII0600
370 WRITE (NDATA*NU(NDATA)) MINUS2,IZERO,IONE,IZERO	FMII0601
./ ALTER 261,261	FMII0602
IF(NEWNAM(1) .EQ. IALPHA(IX)) GO TO 55	FMII0603
670 CONTINUE	FMII0604
./ ALTER 290,292	FMII0605
710 CONTINUE	FMII0606
LOEND(NDATA)=NU(NDATA)	FMII0607
LOMAX(NDATA)= MAXO(NU(NDATA),LOMAX(NDATA))	FMII0608
WRITE(NDATA*NU(NDATA)) MNUS20,IZERO,IONE,IZERO	FMII0609
CALL REWND(NDATA)	FMII0610
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MATR1	FMII0611
./ ALTER 10,10	FMII0612
CALL WD37 (NDATA,J,IONE,NUMBER,2,WORK,1,LENWRK,3,IWORK)	FMII0613
./ ALTER 40,40	FMII0614
CALL WD1 (NDATA,J,IZERO,IMAX,WORK)	FMII0615
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC	FMII0616
./ ALTER 5,5	FMII0617
COMMON NPIT, NPOT, KONST, NWORK, WORK(13000)	FMII0618
./ ALTER 7	FMII0619
DO 1 I = 201,13000	FMII0620
1 WORK(I) = 0.	FMII0621

./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC1	FMII0622
./ ALTER 65,65	FMII0623
CALL RD25(NPREP,CRAP,NSET(I),IFCT(I),IDEV(I),ICHAN(I),ICAP(I))	FMII0624
./ ALTER 68,68	FMII0625
CALL REWND(K1)	FMII0626
./ ALTER 69	FMII0627
100 CONTINUE	FMII0628
./ ALTER 71,71	FMII0629
IF(NDATA .EQ. NSET(I)) GO TO 115	FMII0630
114 CONTINUE	FMII0631
./ ALTER 94,94	FMII0632
110 CALL RD20 (NPREP,7,K1,J,MASTIN)	FMII0633
./ ALTER 97,97	FMII0634
130 CALL RD20 (NPREP,7,K1,J,MASTIO)	FMII0635
./ ALTER 135,135	FMII0636
CALL REWND(NPREP1)	FMII0637
./ ALTER 154,154	FMII0638
CALL RD26(NPREP,ISTAT(I),INSNUM(I),IN,IO,IS,ITOT,ITEMP)	FMII0639
./ ALTER 200,200	FMII0640
IF(NAMOD(K,J) .NE. NAMOD(K,I)) GO TO 300	FMII0641
290 CONTINUE	FMII0642
./ ALTER 215,215	FMII0643
IF(MOD(INSNUM(I),100) .EQ. JSVE) GO TO 330	FMII0644
320 CONTINUE	FMII0645
./ ALTER 216,216	FMII0646
GO TO 440	FMII0647
./ ALTER 237,237	FMII0648
IF(EQUAL) GO TO 380	FMII0649
360 CONTINUE	FMII0650
./ ALTER 253,253	FMII0651
IF(IEND .EQ. ISTAT(I)) GO TO 396	FMII0652
395 CONTINUE	FMII0653
./ ALTER 264,265	FMII0654
CALL WD27 (NPREP1,KK,ISAVM)	FMII0655
CALL WD27 (NPREP1,KK,ISAVT)	FMII0656
./ ALTER 287,289	FMII0657
CALL REWND(NTAPE)	FMII0658
CALL RD13(NTAPE,K1,K1,NUM,7,ITEMP)	FMII0659
CALL REWND(NTAPE)	FMII0660
./ ALTER 291,291	FMII0661
IF(EQUAL) GO TO 470	FMII0662
450 CONTINUE	FMII0663
./ ALTER 318,318	FMII0664
CALL WD27(NPREP1,K,JPRINT)	FMII0665
./ ALTER 323,323	FMII0666
IF(NUMOD(J) .EQ. I) GO TO 550	FMII0667
540 CONTINUE	FMII0668
./ ALTER 326,326	FMII0669
CALL WD10 (NPREP1,7,6,J,NAMOD,NSAVE)	FMII0670
./ ALTER 341,341	FMII0671
IF(NUMOD(J) .EQ. I) GO TO 590	FMII0672
580 CONTINUE	FMII0673
./ ALTER 349,349	FMII0674
CALL WD27(NPREP1,IONLY,INONLY)	FMII0675
./ ALTER 351,351	FMII0676
CALL WD27(NPREP1,IOTPUT,IOPUT)	FMII0677
./ ALTER 408,408	FMII0678

CALL WD27(NPREP1,IONLY,MANY)	FMII0679
./ ALTER 410,411	FMII0680
CALL WD27(NPREP1,IONLY,LOCAT)	FMII0681
CALL WD27(NPREP1,LENSC,ISCALR)	FMII0682
./ ALTER 422,422	FMII0683
CALL WD27(NPREP1,K,ITEMP)	FMII0684
./ ALTER 446,446	FMII0685
CALL WD11(NPREP1,I,LPL,JPL,IPL,INCOMP)	FMII0686
./ ALTER 454,454	FMII0687
CALL WD2(NPREP1,NSET(J),IFCT(J),IDEV(J),ICHAN(J),ICAP(J))	FMII0688
./ ALTER 460,462	FMII0689
CALL WD12(NPREP1,7,7,MATTOT,NAMOD)	FMII0690
CALL REWND(NPREP1)	FMII0691
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC11	FMII0692
./ ALTER 6,6	FMII0693
CALL REWND(NSET)	FMII0694
./ ALTER 9,9	FMII0695
CALL WD28(NSET,KOD(I),IZERO,NUM,NAME)	FMII0696
./ ALTER 11,12	FMII0697
CALL BACKSP(NSET)	FMII0698
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC12	FMII0699
./ ALTER 48,48	FMII0700
IF(NAME .EQ. NAMIN(I)) GO TO 180	FMII0701
170 CONTINUE	FMII0702
./ ALTER 62,62	FMII0703
GO TO 240	FMII0704
./ ALTER 70,70	FMII0705
IF(NAME .EQ. NAMIN(I)) GO TO (270,280), I	FMII0706
260 CONTINUE	FMII0707
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC13	FMII0708
./ ALTER 12,12	FMII0709
GO TO 300	FMII0710
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC2	FMII0711
./ ALTER 52,53	FMII0712
CALL RD27(NPREP1,NUMSAV,ISAVM)	FMII0713
CALL RD27(NPREP1,NUMSAV,ISAVT)	FMII0714
./ ALTER 55,55	FMII0715
CALL RD27(NPREP1,NPRINT,JPRINT)	FMII0716
./ ALTER 58,58	FMII0717
113 CALL RD29(NPREP1,7,7,J,NSAVE)	FMII0718
./ ALTER 60,60	FMII0719
CALL RD27(NPREP1,IONLY,INONLY)	FMII0720
./ ALTER 63,63	FMII0721
IF(INONLY(J) .EQ. ISAVM(I)) ISAVM(I) = -ISAVM(I)	FMII0722
90 CONTINUE	FMII0723
./ ALTER 68,68	FMII0724
CALL RD27(NPREP1,IOTPUT,IOPUT)	FMII0725
./ ALTER 70,72	FMII0726
CALL RD27(NPREP1,IONLY,MANY)	FMII0727
CALL RD27(NPREP1,IONLY,LOCAT)	FMII0728
CALL RD27(NPREP1,LENSC,ISCALR)	FMII0729
./ ALTER 74,74	FMII0730
CALL RD27(NPREP1,NUMIMP,JIN)	FMII0731
./ ALTER 92,92	FMII0732
180 CALL RD14(NPREP,7,7,I,IGEN,KK,KK)	FMII0733
./ ALTER 124,124	FMII0734
IF(EQUAL) GO TO 200	FMII0735

190 CONTINUE	FMII0736
./ ALTER 144,144	FMII0737
IF(K .EQ. INONLY(J)) GO TO 218	FMII0738
217 CONTINUE	FMII0739
./ ALTER 165,165	FMII0740
CALL REWND(NDATA)	FMII0741
./ ALTER 206,206	FMII0742
IF(INONLY(L) .EQ. K) GO TO 317	FMII0743
315 CONTINUE	FMII0744
./ ALTER 230,230	FMII0745
IF(MAYBE(I) .NE. 0) GO TO 261	FMII0746
335 CONTINUE	FMII0747
./ ALTER 233,233	FMII0748
340 CALL REWND(NTAPE)	FMII0749
./ ALTER 257,257	FMII0750
IF(IABS(INSMAT(J+1)) .EQ. I) GO TO 430	FMII0751
420 CONTINUE	FMII0752
./ ALTER 340,340	FMII0753
IF(NUM .EQ. ISTAT(J)) GO TO 680	FMII0754
650 CONTINUE	FMII0755
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC21	FMII0756
./ ALTER 8,8	FMII0757
100 CALL RD15(NDATA,J,KODE,NUM)	FMII0758
./ ALTER 10,11	FMII0759
CALL BACKSP(NDATA)	FMII0760
CALL RD40 (NDATA,J,KODE,NUM,1,7,NPART,1,2,IDIM)	FMII0761
./ ALTER 13,13	FMII0762
IF(NPART(I) .NE. NAME(I)) GO TO 100	FMII0763
110 CONTINUE	FMII0764
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC22	FMII0765
./ ALTER 20,20	FMII0766
110 CALL RD28(NTAPE,J,KODE,NUM,ITEMP)	FMII0767
./ ALTER 25,25	FMII0768
IF(ITEMP(I) .GE. IROW) GO TO 160	FMII0769
140 CONTINUE	FMII0770
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC23	FMII0771
./ ALTER 9,9	FMII0772
100 CALL RD15(NTAPE,J,KODE,NUM)	FMII0773
./ ALTER 12,13	FMII0774
CALL BACKSP(NTAPE)	FMII0775
CALL RD13 (NTAPE,J,KODE,NUM,9,MHEAD)	FMII0776
./ ALTER 15,15	FMII0777
110 CALL REWND(NTAPE)	FMII0778
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC24	FMII0779
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC3	FMII0780
./ ALTER 67,67	FMII0781
IF(ISCAL(K3) .EQ. ISTAT(K4)) GO TO 60	FMII0782
50 CONTINUE	FMII0783
./ ALTER 88,88	FMII0784
IF(I .EQ. ISAVM(J)) GO TO 170	FMII0785
120 CONTINUE	FMII0786
./ ALTER 90,90	FMII0787
IF(NUMOD(K) .EQ. I) GO TO 150	FMII0788
140 CONTINUE	FMII0789
./ ALTER 96,96	FMII0790
IF(NUMOD(L) .EQ. I) I HOLD = L	FMII0791
160 CONTINUE	FMII0792

./ ALTER 116,116	FMII0793
CALL RD17(NPREP1,J,K1,K2,INCOMP)	FMII0794
./ ALTER 133,133	FMII0795
IF(INCOMP(L) .EQ. J-1) ICNT = ICNT + 1	FMII0796
200 CONTINUE	FMII0797
./ ALTER 206,206	FMII0798
IF(NUMOD(K1) .EQ. L) GO TO 310	FMII0799
305 CONTINUE	FMII0800
./ ALTER 209,209	FMII0801
IF(K3 .GT. IFLOC(K2) .AND. K3 .LT. IFEND(K2)) GO TO 330	FMII0802
315 CONTINUE	FMII0803
./ ALTER 241,241	FMII0804
IF(I .EQ. NUMOD(L)) GO TO 363	FMII0805
361 CONTINUE	FMII0806
./ ALTER 251,251	FMII0807
IF(ISAVM(J) .EQ. I) GO TO 560	FMII0808
370 CONTINUE	FMII0809
./ ALTER 260,260	FMII0810
IF(LPURGE(MAT) .GT. 0) GO TO 380	FMII0811
373 CONTINUE	FMII0812
./ ALTER 310,310	FMII0813
IF(M .EQ. 0) GO TO 420	FMII0814
410 CONTINUE	FMII0815
./ ALTER 345,345	FMII0816
IF(INCOMP(L) .EQ. IASGN(NTAPE,LL)) KBAD = KBAD + 1	FMII0817
450 CONTINUE	FMII0818
./ ALTER 387,387	FMII0819
IF(INCOMP(J) .EQ. IASGN(ISET,L)) IBAD = IBAD + 1	FMII0820
481 CONTINUE	FMII0821
./ ALTER 416,416	FMII0822
IF(IFULL(L) .EQ. 0) GO TO 530	FMII0823
520 CONTINUE	FMII0824
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC31	FMII0825
./ ALTER 7,7	FMII0826
IF(MINSCR(J) .GT. NWAIT) NWAIT = MINSCR(J)	FMII0827
100 CONTINUE	FMII0828
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ALOC4	FMII0829
./ ALTER 64,64	FMII0830
100 CALL RD2(NPREP1,NSET(I),IFCT(I),IDEV(I),ICHAN(I),ICAP(I))	FMII0831
./ ALTER 69,69	FMII0832
105 CALL WD9 (NINST,N,IZERO,IHEAD,NSET(I))	FMII0833
./ ALTER 73,74	FMII0834
CALL RD12 (NPREP1,7,7,MATTOT,NAMOD)	FMII0835
CALL REWND(NPREP1)	FMII0836
./ ALTER 83,83	FMII0837
IF(JASGN(NTAPE,J) .EQ. K) GO TO 120	FMII0838
110 CONTINUE	FMII0839
./ ALTER 88,88	FMII0840
IF(NUMOD(L) .EQ. JASGN(NTAPE,J-1)) GO TO 150	FMII0841
140 CONTINUE	FMII0842
./ ALTER 108,108	FMII0843
IF(MTAPE(L) .EQ. I) GO TO 210	FMII0844
200 CONTINUE	FMII0845
./ ALTER 114,114	FMII0846
IF(K1 .EQ. IWIND(I,K)) GO TO 220	FMII0847
215 CONTINUE	FMII0848
	FMII0849

./ ALTER 134,137	FMIIC850
270 CALL WD19 (NINST,7,N,IZERO,LCOPY,IONE,IONE,IZERO,IZERO,MAT,7,2,	FMIIC851
1 NSAVE,JIN(NTAPE),JIND(NTAPE),JINCH(NTAPE),NSET(NTAPE1),	FMIIC852
2 IDEV(NTAPE1),ICHAN(NTAPE1))	FMIIC853
CALL WD9 (NINST,K,IZERO,LWIND,NSET(NTAPE1))	FMIIC854
./ ALTER 151,151	FMIIC855
286 CALL WD9 (NINST,K2,IZERO,IHEAD,NSET(NUMSC))	FMIIC856
./ ALTER 154,154	FMIIC857
CALL WD15 (NINST,KT,ISTAT(I),NOP)	FMIIC858
./ ALTER 158,158	FMIIC859
IF(MAT .EQ. IABS(ISAVM(J)) .AND. ISAVT(J) .GT. 0) GO TO 305	FMIIC860
300 CONTINUE	FMIIC861
./ ALTER 175,175	FMIIC862
IF(MAT .EQ. IABS(ISAVM(K))) GO TO 312	FMIIC863
311 CONTINUE	FMIIC864
./ ALTER 185,186	FMIIC865
315 CALL WD16 (NINST,7,KQ,IZERO,LCOPY,IONE,IONE,IZERO,IZERO,MAT,7,2,	FMIIC866
1 NSAVE,6,ITEMP)	FMIIC867
./ ALTER 203,203	FMIIC868
CALL WD9 (NINST,K1,IZERO,IHEAD,NSET(NTAPE))	FMIIC869
./ ALTER 241,241	FMIIC870
IF(ISAVM(L) .EQ. MAT) GO TO 360	FMIIC871
350 CONTINUE	FMIIC872
./ ALTER 286,286	FMIIC873
450 CALL WD27 (NINST,N,ITEMP)	FMIIC874
./ ALTER 297,299	FMIIC875
CALL WD19 (NINST,7,N,IZERO,LCOPY,IONE,IONE,IZERO,IZERO,MAT,7,2,	FMIIC876
1 NSAVE,JIN(NTAPE),JIND(NTAPE),JINCH(NTAPE),NSET(NTAPE1),	FMIIC877
2 IDEV(NTAPE1),ICHAN(NTAPE1))	FMIIC878
./ ALTER 306,306	FMIIC879
CALL WD9 (NINST,N,IZERO,LWIND,NSET(L))	FMIIC880
./ ALTER 317,317	FMIIC881
CALL WD9 (NINST,N,IZERO,LWIND,NSET(NTAPE))	FMIIC882
./ ALTER 321,321	FMIIC883
CALL WD15 (NINST, N,IZERO,IRETRN)	FMIIC884
./ ALTER 323,325	FMIIC885
520 CALL WD2 (NINST,NSET(I),IFCT(I),IDEV(I),ICHAN(I),ICAP(I))	FMIIC886
CALL REWIND(NINST)	FMIIC887
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=LOGC	FMIIC888
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=LOGC1	FMIIC889
./ ALTER 13,13	FMIIC890
COMMON NPIT,NPOT,KONST,NWORK,USER(60),ITEMP(7)	FMIIC891
./ ALTER 24,24	FMIIC892
120 CALL RDO (NINST,N,ITEMP)	FMIIC893
./ ALTER 34,34	FMIIC894
IF(NOPC(I) .EQ. IOPC) GO TO 160	FMIIC895
150 CONTINUE	FMIIC896
./ ALTER 136,136	FMIIC897
4210 CALL RD35 (NINST,NUMR,I,I,1,5,KONFIG)	FMIIC898
./ ALTER 175,176	FMIIC899
CALL REWIND(NTAPE)	FMIIC900
CALL RD36 (NTAPE,7,K1,K1,N,1,7,1,1,NAME)	FMIIC901
./ ALTER 183,183	FMIIC902
CALL REWIND(NINST)	FMIIC903
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SPCL	FMIIC904
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EXEQ	FMIIC905
./ ALTER 3,3	FMIIC906

COMMON NPIT, NPOT, KONST, NWORK, WORK(13000)	FMII0907
./ ALTER 17	FMII0908
TIME = CLOCK(TIME)	FMII0909
./ ALTER 18,18	FMII0910
50 CALL RDO (NINST, NUM, WORK)	FMII0911
IF (ISTNO) 555, 555, 556	FMII0912
556 TIMEP = TIME	FMII0913
TIME = CLOCK(TIME)	FMII0914
TIMEP = TIME - TIMEP	FMII0915
II = ISTNO - 1	FMII0916
WRITE(6, 557) II, TIMEP	FMII0917
557 FORMAT(' ISTNO = ',I3,' TIME = ',F8.3,' SECONDS')	FMII0918
555 CONTINUE	FMII0919
./ ALTER 199,199	FMII0920
3300 CALL REWND (NUMIN)	FMII0921
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL1	FMII0922
./ ALTER 6,6	FMII0923
COMMON/ IOUNIT/NU(12)	FMII0924
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0925
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0926
CALL REWND(NSET)	FMII0927
./ ALTER 9,9	FMII0928
CALL WDI(NSET,IHEAD(I),IZERO,-NUM,IZERO)	FMII0929
./ ALTER 11,12	FMII0930
CALL BACKSP(NSET)	FMII0931
LOEND(NSET)=NU(NSET)	FMII0932
DO 16 I= 1,MAXDTA	FMII0933
16 LOPOS(NSET,I)=IZERO	FMII0934
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL2	FMII0935
./ ALTER 5,7	FMII0936
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0937
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0938
FIND(NSET,LOEND(NSET))	FMII0939
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL3	FMII0940
./ ALTER 6	FMII0941
COMMON/ IOUNIT/NU(12)	FMII0942
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0943
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0944
./ ALTER 8,11	FMII0945
K=NU(NSET)	FMII0946
101 CONTINUE	FMII0947
DO 2 I= 1,MAXDTA	FMII0948
IF(LOPOS(NSET,I)) 3,3,4	FMII0949
4 IF(LOPOS(NSET,I).LT.K) GO TO 2	FMII0950
./ ALTER 14,15	FMII0951
IF(INAME(J,IFIND).NE.LONAME(NSET,I,J)) GO TO 40	FMII0952
30 CONTINUE	FMII0953
IF(IABS(INAME(7,IFIND)).EQ.IABS(LONAME(NSET,I,7))) GO TO 100	FMII0954
./ ALTER 17,18	FMII0955
2 CONTINUE	FMII0956
3 CONTINUE	FMII0957
./ ALTER 21,22	FMII0958
K=0	FMII0959
GO TO 101	FMII0960
./ ALTER 24,24	FMII0961
CALL REWND(NSET)	FMII0962
./ ALTER 27	FMII0963

FIND(NSET*LOPOS(NSET,I))	FMII0964
CALL RD4 (NSET,J,KODE,NUM,NAME,IMAX,JMAX)	FMII0965
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL4	FMII0966
./ ALTER 13,13	FMII0967
100 CALL RD1 (NSET1, J, CRAP, NUM, WORK)	FMII0968
./ ALTER 15,15	FMII0969
CALL WD1 (NSET2, J, CRAP, NUM, WORK)	FMII0970
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL5	FMII0971
./ ALTER 6	FMII0972
COMMON/IOUNIT/NU(12)	FMII0973
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0974
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0975
DO 16 I= 1,MAXDTA	FMII0976
IF(LOPOS(NSET,I).EQ. IZERO)GO TO 7	FMII0977
16 CONTINUE	FMII0978
WRITE (6,19)NSET	FMII0979
19 FORMAT ('NO SPACE IN UNIT'16'SEE EUTL5')	FMII0980
STOP	FMII0981
7 DO 8 J=1,7	FMII0982
8 LONAME(NSET,I,J)=NAME(J)	FMII0983
LONUMX(NSET) = MAXO(LONUMX(NSET),I)	FMII0984
./ ALTER 7,7	FMII0985
LOPOS(NSET,I)=NU(NSET)	FMII0986
CALL WD4(NSET,IHEAD,IZERO,NINE,NAME,IMAX,JMAX)	FMII0987
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL6	FMII0988
./ ALTER 6,9	FMII0989
COMMON/IOUNIT/NU(12)	FMII0990
COMMON/LOCATE/LONAME(12,30,7),LOPOS(12,30),LOEND(12),	FMII0991
1 LOMAX(12),LONUMX(12),MAXDTA	FMII0992
CALL WD1(NSET,ITRIL,KODE,IONE,IZERO)	FMII0993
CALL WD1(NSET,JTRIL,IZERO,IONE,IZERO)	FMII0994
CALL BACKSP(NSET)	FMII0995
LOEND(NSET)=NU(NSET)	FMII0996
LOMAX(NSET) = MAXO(LOEND(NSET),LOMAX(NSET))	FMII0997
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL7	FMII0998
./ ALTER 14,14	FMII0999
100 CALL RD1 (NSET, ICOL, KODE, NUM, ITEMP)	FMII1000
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL8	FMII1001
./ ALTER 24,24	FMII1002
IF(ITEMP(I) .EQ. 0) J = J + 1	FMII1003
100 CONTINUE	FMII1004
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EUTL9	FMII1005
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PRNT	FMII1006
./ ALTER 48,48	FMII1007
IF(.NOT. FOUND(J)) GO TO 180	FMII1008
175 CONTINUE	FMII1009
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PRNT11	FMII1010
./ ALTER 47,47	FMII1011
70 CALL RD1 (NSET, J, KODE, NUMBER, WORK)	FMII1012
./ ALTER 87,87	FMII1013
LIMIT = LINES - LNCTR	FMII1014
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=IFCN	FMII1015
./ ALTER 15,15	FMII1016
140 CALL RD3 (NSET1, J)	FMII1017
./ ALTER 18,18	FMII1018
150 CALL RD5 (INST, NUM, ISTNO)	FMII1019
./ ALTER 20,20	FMII1020

CALL BACKSP (INST)	FMII1021
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADD	FMII1022
./ ALTER 17,17	FMII1023
50 CALL REWND (NSET)	FMII1024
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADDA	FMII1025
./ ALTER 39,39	FMII1026
COPY(I) = INSPEC(1,I) .EQ. IOSPEC(1,1) .OR. COPY12	FMII1027
./ ALTER 101,101	FMII1028
IF(INSPEC(1,NUM) .EQ. IOSPEC(1,1)) GO TO 320	FMII1029
./ ALTER 120,120	FMII1030
IF(.NOT. FOUND(I)) GO TO 376	FMII1031
375 CONTINUE	FMII1032
./ ALTER 127,127	FMII1033
IF(INSPEC(2,J) .EQ. NTAPE) ISTRAT = ISTRAT + 1	FMII1034
410 CONTINUE	FMII1035
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADDA1	FMII1036
./ ALTER 11,11	FMII1037
CALL RD1 (NSET, JCOL, KODE, NUM, TEMP)	FMII1038
./ ALTER 23,24	FMII1039
130 CALL WD1 (NSET1, JCOL, KODE, NUM, BULK(1,I))	FMII1040
150 CALL RD1 (NSET, JCOL, KODE, NUM, TEMP)	FMII1041
./ ALTER 26,26	FMII1042
CALL WD1 (NSET1, JCOL, KODE, NUM, TEMP)	FMII1043
./ ALTER 29,29	FMII1044
CALL REWND (NSET1)	FMII1045
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADDA1	FMII1046
./ ALTER 31,31	FMII1047
CALL RD1 (NSET, ICOL(I), KODE(I), NUM, TEMP(1,I))	FMII1048
./ ALTER 101,101	FMII1049
340 CALL WD1 (IOSPEC, ICOL(L1), KODE(L1), NUM, TEMP(1,L1))	FMII1050
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADD2	FMII1051
./ ALTER 42,42	FMII1052
120 CALL RD1 (NSET1, ICOL, KODE, NUM, TEMP)	FMII1053
./ ALTER 84,84	FMII1054
250 CALL BACKSP (NSET1)	FMII1055
./ ALTER 88,88	FMII1056
260 CALL RD1 (NSET2, ICOL, KODE, NUM, TEMP)	FMII1057
./ ALTER 123,123	FMII1058
420 CALL WD1 (IOSPEC, ICOL, KODE, NUM, TEMP)	FMII1059
./ ALTER 128,128	FMII1060
450 CALL BACKSP (NSET2)	FMII1061
./ ALTER 141,141	FMII1062
480 CALL WD1 (IOSPEC, ICOL, KODE, NUM, BULK(1,I))	FMII1063
./ ALTER 152,152	FMII1064
600 CALL RD1 (NSET, ICOL, KODE, NUM, TEMP)	FMII1065
./ ALTER 161,161	FMII1066
630 CALL WD1 (IOSPEC, ICOL, KODE, NUM, TEMP)	FMII1067
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MADD3	FMII1068
./ ALTER 46,46	FMII1069
120 CALL RD1 (INSPEC, JCOL, KODE1, NUM, TEMP)	FMII1070
./ ALTER 61,61	FMII1071
240 CALL WD1 (IOSPEC, ICOL(I), K1, NUM1,BULK(1,I))	FMII1072
./ ALTER 75,75	FMII1073
291 CALL WD1 (IOSPEC, JCOL, KODE1, NUM, TEMP)	FMII1074
./ ALTER 105,105	FMII1075
385 CALL WD1 (IOSPEC, JCOL, KODE, NUM, TEMP)	FMII1076
./ ALTER 112,112	FMII1077

CALL WD1 (IOSPEC, JCOL, KODE, NUM, BULK(1,IPOS))	FMII1078
./ ALTER 155,155	FMII1079
710 CALL WD1 (IOSPEC, ICOL(I), KODE, NUM, BULK(1,I))	FMII1080
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EMPTY	FMII1081
./ ALTER 15,15	FMII1082
50 CALL REWND (NSET)	FMII1083
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EMPTYA	FMII1084
./ ALTER 125,125	FMII1085
IF(.NOT. FOUND(I)) GO TO 377	FMII1086
375 CONTINUE	FMII1087
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EMPTYA1	FMII1088
./ ALTER 11,11	FMII1089
CALL RD1 (NSET, JCOL, KODE, NUM, TEMP)	FMII1090
./ ALTER 23,24	FMII1091
130 CALL WD1 (NSET1, JCOL, KODE, NUM, BULK(1,I))	FMII1092
150 CALL RD1 (NSET, JCOL, KODE, NUM, TEMP)	FMII1093
./ ALTER 26,26	FMII1094
CALL WD1 (NSET1, JCOL, KODE, NUM, TEMP)	FMII1095
./ ALTER 29,29	FMII1096
CALL REWND (NSET1)	FMII1097
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EMPTY1	FMII1098
./ ALTER 43,43	FMII1099
85 CALL RD1 (NSET2, JCOL, KODE1, NUM, TEMP)	FMII1100
./ ALTER 50,50	FMII1101
CALL RD1 (NSET1, ICOL(II), KODE, NUM1, BULK(1,II))	FMII1102
./ ALTER 55,55	FMII1103
IF(KODE .EQ. 0) NUMB(II) = -NUMB(II)	FMII1104
100 CONTINUE	FMII1105
./ ALTER 64,64	FMII1106
200 CALL RD1 (NSET2, JCOL, KODE1, NUM, TEMP)	FMII1107
./ ALTER 182,182	FMII1108
870 CALL WD1 (IOSPEC, JCOL, KODE, NUM, TEMP)	FMII1109
./ ALTER 184,184	FMII1110
900 CALL WD1 (IOSPEC, JCOL, KODE, NUM, BULK(1,IPOS))	FMII1111
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TMPY	FMII1112
./ ALTER 118,118	FMII1113
320 CALL REWND (NSETB1)	FMII1114
./ ALTER 120,120	FMII1115
330 CALL RD1 (NSETB, J, KODE, NUM, WORKR)	FMII1116
./ ALTER 122,122	FMII1117
CALL WD1 (NSETB1, J, KODE, NUM, WORKR)	FMII1118
./ ALTER 125,126	FMII1119
CALL REWND (NSETB1)	FMII1120
./ ALTER 143,144	FMII1121
355 CALL RD1 (NSETA, J, KODE, NUM, WORKR)	FMII1122
CALL WD1 (NSETA1, J, KODE, NUM, WORKR)	FMII1123
./ ALTER 146,146	FMII1124
360 CALL WD1 (NSETA2, J, KODE, NUM, WORKR)	FMII1125
./ ALTER 148,149	FMII1126
CALL REWND (NSETA1)	FMII1127
./ ALTER 151,152	FMII1128
CALL REWND (NSETA2)	FMII1129
./ ALTER 183,183	FMII1130
CALL REWND (NSETA)	FMII1131
./ ALTER 195,195	FMII1132
IF(.NOT. FOUND(I)) GO TO 1030	FMII1133
1020 CONTINUE	FMII1134

./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TMPY1	FMII1135
./ ALTER 11,11	FMII1136
20 CALL RD1 (NSETB, J, KODE, NUM, ITRASH)	FMII1137
./ ALTER 31,31	FMII1138
100 CALL BACKSP (NSETB)	FMII1139
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TMPY2	FMII1140
./ ALTER 14,14	FMII1141
20 CALL RD1 (NSETA, JCOL, KODE, NUM, A)	FMII1142
./ ALTER 59,59	FMII1143
190 CALL WD1 (NSETC, JCOL, KODE, NUMC, C(1,J))	FMII1144
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TMPY3	FMII1145
./ ALTER 14,14	FMII1146
10 CALL RD1 (NSETA, J, KODE, NUM, IB)	FMII1147
./ ALTER 33,33	FMII1148
100 CALL BACKSP (NSETA)	FMII1149
./ ALTER 36,36	FMII1150
CALL REWND (NSETA1)	FMII1151
./ ALTER 40,40	FMII1152
CALL REWND (NSETA2)	FMII1153
./ ALTER 53,53	FMII1154
CALL WD1 (NSETA1, J, KODE, NUM, IA(IX))	FMII1155
./ ALTER 55,55	FMII1156
150 CALL WD1 (NSETA2, J, KODE, NUM, IA(IX))	FMII1157
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TMPY4	FMII1158
./ ALTER 12,12	FMII1159
CALL RD1 (NSETB, JCOL, KODE, NUM, B)	FMII1160
./ ALTER 50,50	FMII1161
150 CALL WD1 (NSETC, JCOL, KODE, NUM, C)	FMII1162
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MULT	FMII1163
./ ALTER 117,117	FMII1164
320 CALL REWND (NSETB1)	FMII1165
./ ALTER 119,119	FMII1166
330 CALL RD1 (NSETB, J, KODE, NUM, WORKR)	FMII1167
./ ALTER 121,121	FMII1168
CALL WD1 (NSETB1, J, KODE, NUM, WORKR)	FMII1169
./ ALTER 124,125	FMII1170
CALL REWND (NSETB1)	FMII1171
./ ALTER 142,143	FMII1172
355 CALL RD1 (NSETA, J, KODE, NUM, WORKR)	FMII1173
CALL WD1 (NSETA1, J, KODE, NUM, WORKR)	FMII1174
./ ALTER 145,145	FMII1175
360 CALL WD1 (NSETA2, J, KODE, NUM, WORKR)	FMII1176
./ ALTER 147,148	FMII1177
CALL REWND (NSETA1)	FMII1178
./ ALTER 150,151	FMII1179
CALL REWND (NSETA2)	FMII1180
./ ALTER 182,182	FMII1181
CALL REWND (NSETA)	FMII1182
./ ALTER 194,194	FMII1183
IF (.NOT. FOUND(I)) GO TO 1030	FMII1184
1020 CONTINUE	FMII1185
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MULT1	FMII1186
./ ALTER 11,11	FMII1187
20 CALL RD1 (NSETB, J, KODE, NUM, ITRASH)	FMII1188
./ ALTER 31,31	FMII1189
100 CALL BACKSP (NSETB)	FMII1190
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MULT2	FMII1191

./ ALTER 14,14	FMII1192
20 CALL RD1 (NSETA, JCOL, KODE, NUM, A)	FMII1193
./ ALTER 26,26	FMII1194
IF (IB(1) .EQ. JCOL) GO TO 60	FMII1195
50 CONTINUE	FMII1196
./ ALTER 55,55	FMII1197
190 CALL WD1 (NSETC, JCOL, KODE, NUMC, C(1,J))	FMII1198
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MULT3	FMII1199
./ ALTER 14,14	FMII1200
10 CALL RD1 (NSETA, J, KODE, NUM, IB)	FMII1201
./ ALTER 33,33	FMII1202
100 CALL BACKSP (NSETA)	FMII1203
./ ALTER 36,36	FMII1204
CALL REWND (NSETA1)	FMII1205
./ ALTER 40,40	FMII1206
CALL REWND (NSETA2)	FMII1207
./ ALTER 53,53	FMII1208
CALL WD1 (NSETA1, J, KODE, NUM, IA(IX))	FMII1209
./ ALTER 55,55	FMII1210
150 CALL WD1 (NSETA2, J, KODE, NUM, IA(IX))	FMII1211
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=MULT4	FMII1212
./ ALTER 12,12	FMII1213
CALL RD1 (NSETB, JCOL, KODE, NUM, B)	FMII1214
./ ALTER 50,50	FMII1215
150 CALL WD1 (NSETC, JCOL, KODE, NUM, C)	FMII1216
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SMPY	FMII1217
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SMPYA	FMII1218
./ ALTER 71,71	FMII1219
IF(NUM .GT. NEND) GO TO 330	FMII1220
320 CONTINUE	FMII1221
./ ALTER 76,76	FMII1222
CALL REWND (NSET)	FMII1223
./ ALTER 88,88	FMII1224
350 CALL WD1 (NSET, IWORK(J), KODE, IWORK(K), IWORK(NUM))	FMII1225
./ ALTER 90,90	FMII1226
370 CALL RD1 (NSET1, JCOL, KODE, NUM, IWORK)	FMII1227
./ ALTER 92,92	FMII1228
CALL WD1 (NSET, JCOL, KODE, NUM, IWORK)	FMII1229
./ ALTER 95,95	FMII1230
CALL REWND (NSET)	FMII1231
./ ALTER 106,106	FMII1232
IF(.NOT. FOUND(1)) GO TO 420	FMII1233
440 CONTINUE	FMII1234
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SMPYA1	FMII1235
./ ALTER 5,5	FMII1236
CALL RD1 (NSET, ICOL, KODE, NUMB, WORK)	FMII1237
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SMPY1	FMII1238
./ ALTER 24,24	FMII1239
CALL WD1 (IOSPEC, ICOL(1), KODE, NUMB(1), WORK(NUM))	FMII1240
./ ALTER 31,31	FMII1241
CALL RD1 (INSPEC, JCOL, KODE, NUM, WORK)	FMII1242
./ ALTER 36,36	FMII1243
310 CALL WD1 (IOSPEC, JCOL, KODE, NUM, WORK)	FMII1244
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=POWR	FMII1245
./ ALTER 46,46	FMII1246
CALL BACKSP (NSET1)	FMII1247
./ ALTER 49,49	FMII1248

CALL REWND (NSET)	FMII1249
./ ALTER 54,54	FMII1250
110 CALL RD1 (NSET1, J, KODE, NUM, WORKR)	FMII1251
./ ALTER 63,63	FMII1252
150 CALL WD1 (NSET0, J, KODE, NUM, WORKR)	FMII1253
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRAN	FMII1254
./ ALTER 66,66	FMII1255
CALL REWND (NSETX)	FMII1256
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRAN1	FMII1257
./ ALTER 9,9	FMII1258
10 CALL RD1 (NSET1, J, KODE, NUM, ITEMP)	FMII1259
./ ALTER 24,24	FMII1260
100 CALL BACKSP (NSET1)	FMII1261
./ ALTER 26,27	FMII1262
CALL REWND (NSET1)	FMII1263
CALL REWND (NSET2)	FMII1264
./ ALTER 40,40	FMII1265
CALL WD1 (NSET1, J, KODE, NUM, IWORKR(IX))	FMII1266
./ ALTER 42,42	FMII1267
190 CALL WD1 (NSET2, J, KODE, NUM, IWORKR(IX))	FMII1268
./ ALTER 44,45	FMII1269
250 CALL RD1 (NSET1, J, KODE, NUM, ITEMP)	FMII1270
CALL WD1 (NSET1, J, KODE, NUM, ITEMP)	FMII1271
./ ALTER 47,47	FMII1272
290 CALL WD1 (NSET2, J, KODE, NUM, ITEMP)	FMII1273
./ ALTER 50,53	FMII1274
500 CALL REWND (NSET1)	FMII1275
CALL REWND (NSET2)	FMII1276
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRAN2	FMII1277
./ ALTER 13,13	FMII1278
20 CALL RD1 (NSET, J, KODE, NUM, TEMP)	FMII1279
./ ALTER 45,45	FMII1280
230 CALL WD1 (NSET0, J, KODE, NUM, WORKR(1,IR))	FMII1281
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRAN3	FMII1282
./ ALTER 41,41	FMII1283
190 CALL WD1 (NSET0, J, KODE, NUM, TEMP)	FMII1284
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INVT	FMII1285
./ ALTER 51,51	FMII1286
CALL REWND (MSCR1)	FMII1287
./ ALTER 62,62	FMII1288
CALL RD1 (INSET1, KCO, KCOM, NUM, WKR)	FMII1289
./ ALTER 65,65	FMII1290
CALL WD6 (MSCR1, IEQ, WKR, IMX)	FMII1291
./ ALTER 77,78	FMII1292
CALL REWND (MSCR1)	FMII1293
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INVT1	FMII1294
./ ALTER 58,58	FMII1295
CALL REWND (KTAPE)	FMII1296
./ ALTER 63,63	FMII1297
CALL REWND (MATXIN)	FMII1298
./ ALTER 83,83	FMII1299
137 CALL RD6 (MATXIN, KOLL, COL(1,J), IMAX)	FMII1300
./ ALTER 98,99	FMII1301
CALL REWND (MATXIN)	FMII1302
CALL REWND (MATXOT)	FMII1303
./ ALTER 107,107	FMII1304
210 CALL RD6 (MATXIN, JJM, COL(1,J), IMAX)	FMII1305

./ ALTER 132,132	FMII1306
250 CALL WD6 (MATXOT, KOLL, COL(1,N), IMAX)	FMII1307
./ ALTER 135,137	FMII1308
CALL REWND (MATXIN)	FMII1309
CALL REWND (MATXOT)	FMII1310
./ ALTER 147,147	FMII1311
CALL WD6 (KTAPE, KOLL, COL(1,1), IMAX)	FMII1312
./ ALTER 150,151	FMII1313
CALL REWND (KTAPE)	FMII1314
./ ALTER 196,196	FMII1315
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1316
./ ALTER 253,255	FMII1317
1160 CALL BACKSP (MATXIN)	FMII1318
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1319
CALL WD6 (NSCR, KOLL, COL(1,LOOP2), IMAX)	FMII1320
./ ALTER 266,268	FMII1321
1170 CALL REWND (MATXIN)	FMII1322
CALL REWND (NSCR)	FMII1323
./ ALTER 291,292	FMII1324
1903 CALL REWND (NSCR)	FMII1325
./ ALTER 305,305	FMII1326
CALL RD6 (NSCR, KOLL, TREJ, IMAX)	FMII1327
./ ALTER 310,311	FMII1328
1910 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1329
1920 CALL REWND (NSCR)	FMII1330
./ ALTER 315,315	FMII1331
CALL RD6 (MATXIN, KOLL, TREJ, IMAX)	FMII1332
./ ALTER 320,320	FMII1333
1930 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1334
./ ALTER 322,323	FMII1335
1940 CALL REWND (MATXIN)	FMII1336
./ ALTER 327,327	FMII1337
CALL REWND (MATXOT)	FMII1338
./ ALTER 329,329	FMII1339
CALL RD6 (KTAPE, KOLL, TREJ, IMAX)	FMII1340
./ ALTER 331,333	FMII1341
1950 CALL WD6 (NSCR, KOLL, TREJ, IMAX)	FMII1342
CALL REWND (NSCR)	FMII1343
./ ALTER 340,340	FMII1344
CALL REWND (NSCR)	FMII1345
./ ALTER 353,353	FMII1346
CALL RD6 (KTAPE, KOLL, COL(1,1), IMAX)	FMII1347
./ ALTER 360,360	FMII1348
CALL WD1 (MOT1, IFO, KOMP, INUM, COL(1,2))	FMII1349
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=INVT11	FMII1350
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SEQE	FMII1351
./ ALTER 55,55	FMII1352
CALL REWND (MSCR1)	FMII1353
./ ALTER 68,68	FMII1354
CALL RD1 (INSET1, KCO, KCOM, NUM, WKR)	FMII1355
./ ALTER 74,74	FMII1356
CALL WD6 (MSCR1, IEQ, KCDF, -IMX)	FMII1357
./ ALTER 77,77	FMII1358
CALL WD6 (MSCR1, IEQ, WKR, IMX)	FMII1359
./ ALTER 89,90	FMII1360
CALL REWND (MSCR1)	FMII1361
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SEQE1	FMII1362

./ ALTER 66,66	FMII1363
CALL REWND (MATXIN)	FMII1364
./ ALTER 86,86	FMII1365
137 CALL RD6 (MATXIN, KOLL, COL(1,J), IMAX)	FMII1366
./ ALTER 101,102	FMII1367
CALL REWND (MATXIN)	FMII1368
CALL REWND (MATXOT)	FMII1369
./ ALTER 110,110	FMII1370
210 CALL RD6 (MATXIN, JJM, COL(1,J), IMAX)	FMII1371
./ ALTER 134,134	FMII1372
250 CALL WD6 (MATXOT, KOLL, COL(1,N), IMAX)	FMII1373
./ ALTER 137,139	FMII1374
CALL REWND (MATXIN)	FMII1375
CALL REWND (MATXOT)	FMII1376
./ ALTER 167,167	FMII1377
340 CALL RD6 (KTAPE, KOLL, COL(1,J), IMAX)	FMII1378
./ ALTER 173,173	FMII1379
380 CALL WD6 (MATXOT, KOL2, COL(1,N), IMAX)	FMII1380
./ ALTER 176,179	FMII1381
CALL REWND (NSCR)	FMII1382
CALL REWND (KTAPE)	FMII1383
CALL REWND (MATXOT)	FMII1384
./ ALTER 227,227	FMII1385
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1386
./ ALTER 284,286	FMII1387
1160 CALL BACKSP (MATXIN)	FMII1388
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1389
CALL WD6 (NSCR, KOLL, COL(1,LOOP2), IMAX)	FMII1390
./ ALTER 297,299	FMII1391
1170 CALL REWND (MATXIN)	FMII1392
CALL REWND (NSCR)	FMII1393
./ ALTER 322,323	FMII1394
1903 CALL REWND (NSCR)	FMII1395
./ ALTER 336,336	FMII1396
CALL RD6 (NSCR, KOLL, TREJ, IMAX)	FMII1397
./ ALTER 341,342	FMII1398
1910 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1399
1920 CALL REWND (NSCR)	FMII1400
./ ALTER 346,346	FMII1401
CALL RD6 (MATXIN, KOLL, TREJ, IMAX)	FMII1402
./ ALTER 351,351	FMII1403
1930 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1404
./ ALTER 353,354	FMII1405
1940 CALL REWND (MATXIN)	FMII1406
./ ALTER 358,358	FMII1407
CALL REWND (MATXOT)	FMII1408
./ ALTER 360,360	FMII1409
CALL RD6 (KTAPE, KOLL, TREJ, IMAX)	FMII1410
./ ALTER 362,364	FMII1411
1950 CALL WD6 (NSCR, KOLL, TREJ, IMAX)	FMII1412
CALL REWND (NSCR)	FMII1413
./ ALTER 371,371	FMII1414
CALL REWND (NSCR)	FMII1415
./ ALTER 384,384	FMII1416
CALL RD6 (KTAPE, KOLL, COL(1,1), IMAX)	FMII1417
./ ALTER 391,391	FMII1418
CALL WD1 (MOT1, IFO, KOMP, INUM, COL(1,2))	FMII1419

./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SEQ11	FMII1420
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SEQ1	FMII1421
./ ALTER 18,18	FMII1422
CALL REWND (MSCR)	FMII1423
./ ALTER 28,28	FMII1424
CALL RD1 (MSIN1, KCOL, KCOM, NUM, WORKR)	FMII1425
./ ALTER 35,36	FMII1426
10 CALL WD7 (MSCR, WORKR, INUM)	FMII1427
CALL REWND (MSCR)	FMII1428
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SEQ11,	*FMII1429
./ NAME=SEQ11	FMII1430
./ ALTER 17,17	FMII1431
CALL REWND (MTEMP)	FMII1432
./ ALTER 44,44	FMII1433
CALL RD1 (MSIN2, KCOL, KCOM, NUM, COL)	FMII1434
./ ALTER 80,80	FMII1435
CALL RD8 (NSCR1, COL, ISEQ, NUM)	FMII1436
./ ALTER 85,85	FMII1437
CALL RD7 (NSCR1, COL, IMAX)	FMII1438
./ ALTER 117,117	FMII1439
CALL REWND (NSCR1)	FMII1440
./ ALTER 121,121	FMII1441
1900 CALL WD7 (MTEMP, COLX(1,IRITE), IMAX)	FMII1442
./ ALTER 124,125	FMII1443
CALL REWND (MTEMP)	FMII1444
./ ALTER 129,130	FMII1445
CALL RD7 (MTEMP, COL, IMAX)	FMII1446
2100 CALL WD1 (MSOT, IOUT, KOMP, IMAX, COL)	FMII1447
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=STRC	FMII1448
./ ALTER 58,58	FMII1449
CALL REWND (MSCR1)	FMII1450
./ ALTER 71,71	FMII1451
CALL RD1 (INSET1, KCO, KCOM, NUM, WKR)	FMII1452
./ ALTER 74,74	FMII1453
CALL WD6 (MSCR1, IEQ, WKR, IMX)	FMII1454
./ ALTER 86,87	FMII1455
CALL REWND (MSCR1)	FMII1456
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=STRC1	FMII1457
./ ALTER 73,73	FMII1458
CALL RD9 (IWAT, KCO, KCOM, NUM, WAIT(IW))	FMII1459
./ ALTER 94,94	FMII1460
CALL REWND (MATXIN)	FMII1461
./ ALTER 114,114	FMII1462
137 CALL RD6 (MATXIN, KOLL, COL(1,J), IMAX)	FMII1463
./ ALTER 129,130	FMII1464
CALL REWND (MATXIN)	FMII1465
CALL REWND (MATXOT)	FMII1466
./ ALTER 138,138	FMII1467
210 CALL RD6 (MATXIN, JJM, COL(1,J), IMAX)	FMII1468
./ ALTER 164,164	FMII1469
250 CALL WD6 (MATXOT, KOLL, COL(1,N), IMAX)	FMII1470
./ ALTER 167,169	FMII1471
CALL REWND (MATXIN)	FMII1472
CALL REWND (MATXOT)	FMII1473
./ ALTER 197,197	FMII1474
340 CALL RD6 (KTAPE, KOLL, COL(1,J), IMAX)	FMII1475
./ ALTER 203,203	FMII1476

380 CALL WD6 (MATXOT, KOL2, COL(1,N), IMAX)	FMII1477
./ ALTER 206,209	FMII1478
CALL REWND (NSCR)	FMII1479
CALL REWND (KTAPE)	FMII1480
CALL REWND (MATXOT)	FMII1481
./ ALTER 258,258	FMII1482
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1483
./ ALTER 314,316	FMII1484
1160 CALL BACKSP (MATXIN)	FMII1485
CALL RD6 (MATXIN, KOLL, COL(1,LOOP2), IMAX)	FMII1486
CALL WD6 (NSCR, KOLL, COL(1,LOOP2), IMAX)	FMII1487
./ ALTER 326,328	FMII1488
1170 CALL REWND (MATXIN)	FMII1489
CALL REWND (NSCR)	FMII1490
./ ALTER 355,356	FMII1491
1903 CALL REWND (NSCR)	FMII1492
./ ALTER 369,369	FMII1493
CALL RD6 (NSCR, KOLL, TREJ, IMAX)	FMII1494
./ ALTER 374,375	FMII1495
1910 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1496
1920 CALL REWND (NSCR)	FMII1497
./ ALTER 379,379	FMII1498
CALL RD6 (MATXIN, KOLL, TREJ, IMAX)	FMII1499
./ ALTER 384,384	FMII1500
1930 CALL WD6 (MATXOT, KOLL, TREJ, IMAX)	FMII1501
./ ALTER 386,387	FMII1502
1940 CALL REWND (MATXIN)	FMII1503
./ ALTER 391,391	FMII1504
CALL REWND (MATXOT)	FMII1505
./ ALTER 393,393	FMII1506
CALL RD6 (KTAPE, KOLL, TREJ, IMAX)	FMII1507
./ ALTER 395,397	FMII1508
1950 CALL WD6 (NSCR, KOLL, TREJ, IMAX)	FMII1509
CALL REWND (NSCR)	FMII1510
./ ALTER 404,404	FMII1511
CALL REWND (NSCR)	FMII1512
./ ALTER 422,423	FMII1513
CALL REWND (NSCR)	FMII1514
./ ALTER 433,434	FMII1515
CALL RD6 (MATXIN, KOLL, COL(1,1), IMAX)	FMII1516
2500 CALL WD6 (NSCR, KOLL, COL(1,1), IMAX)	FMII1517
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=STRC11	FMII1518
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=STRC12	FMII1519
./ ALTER 49,49	FMII1520
CALL RD6 (MFX, IB5, COL2, IMAX)	FMII1521
./ ALTER 60,60	FMII1522
CALL WD1 (MOT1, IFX, KOMP, INUM, COL1)	FMII1523
./ ALTER 66,66	FMII1524
CALL RD6 (MFO, KOLL, COL2, IMAX)	FMII1525
./ ALTER 76,76	FMII1526
CALL WD1 (MOT2, IFO, KOMP, INUM, COL1)	FMII1527
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EIGN	FMII1528
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EIGN1	FMII1529
./ ALTER 40,40	FMII1530
IEQ = IEQ	FMII1531
./ ALTER 94,94	FMII1532
CALL RD1 (NSET1, KCO, KCOM, NUM, VALU)	FMII1533

•/ ALTER 270,270	FMII1534
CALL WD1 (MOT1, IFO, KOMP, NEIG, VALU)	FMII1535
•/ ALTER 382,382	FMII1536
530 CALL WD1 (MOT2, IFO, KOMP, N, V)	FMII1537
•/ ALTER 424,424	FMII1538
CALL WD1 (MOT2, IFO, KOMP, N, V)	FMII1539
•/ ALTER 426,426	FMII1540
IF(N.EQ. 2) V(I) = AV2	FMII1541
660 CONTINUE	FMII1542
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ADJN	FMII1543
•/ ALTER 64,64	FMII1544
CALL BACKSP (NTAPE1)	FMII1545
•/ ALTER 71,71	FMII1546
IF(ERROR) GO TO 400	FMII1547
230 CONTINUE	FMII1548
•/ ALTER 80,80	FMII1549
250 CALL RD1 (NSET1, J, KODE, NUM, WORKR)	FMII1550
•/ ALTER 82,82	FMII1551
CALL WD1 (NSET3, J, KODE, NUM, WORKR)	FMII1552
•/ ALTER 84,84	FMII1553
260 CALL RD1 (NSET2, J, KODE, NUM, WORKR)	FMII1554
•/ ALTER 87,87	FMII1555
CALL WD1 (NSET3, J1,KODE, NUM, WORKR)	FMII1556
•/ ALTER 91,91	FMII1557
300 CALL RD1 (NSET1, J, KODE, NUM, WORKR)	FMII1558
•/ ALTER 96,96	FMII1559
CALL WD1 (NSET3, J, KODE, NUM, WORKR)	FMII1560
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ENVR	FMII1561
•/ ALTER 17,17	FMII1562
100 CALL RD1 (INSPEC, J, KODE, NUM, WORKR)	FMII1563
•/ ALTER 29,29	FMII1564
130 CALL RD1 (INSPEC, J, KODE, NUM, WORKR)	FMII1565
•/ ALTER 37,37	FMII1566
IF(WORKR(I) .LT. WORKR(K)) WORKR(K) = WORKR(I)	FMII1567
150 CONTINUE	FMII1568
•/ ALTER 66,66	FMII1569
CALL WD1 (IOSPEC, J, KODE, NUM, WORKR(K))	FMII1570
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ENVC	FMII1571
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ENVC1	FMII1572
•/ ALTER 15,15	FMII1573
20 CALL RD1 (NSET1, J, KODE, NUM, WORK)	FMII1574
•/ ALTER 46,46	FMII1575
CALL WD2 (NSET2, L, IZERO, ITWO, TMAX(L), TMIN(L))	FMII1576
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DIAG	FMII1577
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DIAG1	FMII1578
•/ ALTER 12,12	FMII1579
CALL RD1 (NSET1, J, KODE, NUM, WORK)	FMII1580
•/ ALTER 38,38	FMII1581
160 CALL WD2 (NSET2, JWORK(K+1), IONE, ITWO, BWORK(K), JWORK(K+1))	FMII1582
•/ ALTER 46,46	FMII1583
CALL RD1 (NSET1, J, KODE, NUM, WORK)	FMII1584
•/ ALTER 57,57	FMII1585
CALL RD1 (NSET1, J, KODE, NUM, WORK)	FMII1586
•/ ALTER 65,65	FMII1587
290 CALL WD2 (NSET2, JWORK(K+1), IONE, ITWO, BWORK(K), JWORK(K+1))	FMII1588
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=NAME	FMII1589
•/ ALTER 33,33	FMII1590

CALL REWIND (NSETS)	FMII1591
./ ALTER 42,42	FMII1592
CALL REWIND (NSETS)	FMII1593
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=NAME1	FMII1594
./ ALTER 14,14	FMII1595
CALL RDL (NSETI, J, KODE, NUM, TEMP)	FMII1596
./ ALTER 49,49	FMII1597
180 CALL WDL (NSETS, J, KODE, NUM, WORKR(K))	FMII1598
./ ALTER 51,51	FMII1599
200 CALL RDL (NSETI, J, KODE, NUM, TEMP)	FMII1600
./ ALTER 57,57	FMII1601
215 CALL WDL (NSETS, J, KODE, NUM, TEMP)	FMII1602
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=US01	FMII1603
./ ALTER 11	FMII1604
COMMON/IOUNIT/ NU(12)	FMII1605
./ ALTER 33,35	FMII1606
FIND (IST1'1)	FMII1607
FIND (IST2'1)	FMII1608
FIND (IST3'1)	FMII1609
./ ALTER 122,124	FMII1610
1000 FIND (IST1'1)	FMII1611
FIND (IST2'1)	FMII1612
FIND (IST3'1)	FMII1613
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SRT	FMII1614
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=VECT	FMII1615
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=COORD	FMII1616
./ ALTER 6	FMII1617
COMMON/IOUNIT/ NU(12)	FMII1618
./ ALTER 127,127	FMII1619
WRITE (ITAPE'NU(ITAPE)) KNT, (ND(I),(ADC(I,J),J=1,3),I=1,KNT)	FMII1620
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=VECRD	FMII1621
./ ALTER 5	FMII1622
COMMON/IOUNIT/ NU(12)	FMII1623
./ ALTER 118,118	FMII1624
WRITE (ITAPE'NU(ITAPE)) NVEC, NAME, (AK(J), J = 1,3)	FMII1625
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=BARRO	FMII1626
./ ALTER 6	FMII1627
COMMON/IOUNIT/ NU(12)	FMII1628
./ ALTER 147,149	FMII1629
WRITE (IST3'NU(IST3)) MD,K,(AK(J),J=1,10),((D(I,J),I=1,MD),J=1,MD)	FMII1630
1, (NAME(J),J=1,K)	FMII1631
WRITE (IST1'NU(IST1)) MT,(ND(J),(DFOR(I,J),I=1,3), J=1,MT)	FMII1632
./ ALTER 152,152	FMII1633
WRITE (IST1'NU(IST1)) MT,(ND(J),(DFOR(I,J),I=1,3), J=N1,N2)	FMII1634
./ ALTER 162,162	FMII1635
WRITE (IST2'NU(IST2)) MR,(ND(J),(DFOR(I,J),I=1,3), J=N1,N2)	FMII1636
./ ALTER 165,165	FMII1637
WRITE (IST2'NU(IST2)) MR,(ND(J),(DFOR(I,J),I=1,3), J=N1,N2)	FMII1638
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PANRD	FMII1639
./ ALTER 6	FMII1640
COMMON/IOUNIT/ NU(12)	FMII1641
./ ALTER 188,188	FMII1642
WRITE (IST3'NU(IST3)) (AK(J),J=1,35),(NID(J,2),J=1,4)	FMII1643
./ ALTER 191,191	FMII1644
420 WRITE (IST1'NU(IST1)) L, NID(I,1),(D(J,I),J=1,3)	FMII1645
./ ALTER 234,237	FMII1646
550 FIND (IST2'1)	FMII1647

FIND (IST3'1)	FMII1648
•/ ALTER 242,243	FMII1649
555 FIND (IST1'1)	FMII1650
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PSORT	FMII1651
•/ ALTER 5	FMII1652
COMMON/IOUNIT/ NU(12)	FMII1653
•/ ALTER 31,31	FMII1654
FIND (IT'1)	FMII1655
•/ ALTER 42,42	FMII1656
FIND (IST2'1)	FMII1657
•/ ALTER 45,46	FMII1658
FIND (IST1'1)	FMII1659
•/ ALTER 70,71	FMII1660
130 FIND (IST2'1)	FMII1661
•/ ALTER 74,74	FMII1662
FIND (IST1'1)	FMII1663
•/ ALTER 84,84	FMII1664
160 FIND (IST2'1)	FMII1665
•/ ALTER 88,89	FMII1666
FIND (IST1'1)	FMII1667
•/ ALTER 96,96	FMII1668
FIND (IST1'1)	FMII1669
•/ ALTER 114,117	FMII1670
FIND (IT1'1)	FMII1671
FIND (IT2'1)	FMII1672
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRD	FMII1673
•/ ALTER 5	FMII1674
COMMON/IOUNIT/ NU(12)	FMII1675
•/ ALTER 6,6	FMII1676
10 READ (ITAPE'NU(ITAPE)) N,(ID(J),(D(I,J),I=1,3),J=1,N)	FMII1677
•/ ALTER 8,8	FMII1678
FIND (ITAPE'NU(ITAPE)-1)	FMII1679
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=TRDW	FMII1680
•/ ALTER 5	FMII1681
COMMON/IOUNIT/ NU(12)	FMII1682
•/ ALTER 7,7	FMII1683
100 READ (ITR'NU(ITR)) M,(NB(J),(B(K,J),K=1,3),J=1,M)	FMII1684
•/ ALTER 9,9	FMII1685
WRITE (ITW'NU(ITW)) M,(NB(J),(B(K,J),K=1,3),J=1,M)	FMII1686
•/ ALTER 12,12	FMII1687
200 READ (ITR'NU(ITR)) K1,K2,K3,M,(NB(J),(B(L,J),L=1,3),J=1,M)	FMII1688
•/ ALTER 14,14	FMII1689
WRITE (ITW'NU(ITW)) K1,K2,K3,M,(NB(J),(B(L,J),L=1,3),J=1,M)	FMII1690
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=WRT	FMII1691
•/ ALTER 6	FMII1692
COMMON/IOUNIT/ NU(12)	FMII1693
•/ ALTER 22,22	FMII1694
WRITE (ITAPE'NU(ITAPE)) J1,J2,NRC,L3,(ND(J),(A(LJ,J),LJ=1,3),	FMII1695
1 J=K1,L2)	FMII1696
•/ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=SORT	FMII1697
•/ ALTER 7	FMII1698
COMMON/IOUNIT/ NU(12)	FMII1699
•/ ALTER 16,16	FMII1700
50 READ (ITR'NU(ITR)) NS,NR,NREC,KR,(NB(J),(B(K,J),K=1,3),J=1,KR)	FMII1701
•/ ALTER 37,37	FMII1702
WRITE (ITW'NU(ITW)) NS,NR,NREC,KR,(NB(J),(B(K,J),K=1,3),J=1,KR)	FMII1703
•/ ALTER 39,39	FMII1704

READ (ITR'NU(ITR)) NS,NR,NRRC,KR,(NB(J),(B(K,J),K=1,3),J=1,KR)	FMII1705
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=ELIM	FMII1706
./ ALTER 5	FMII1707
COMMON/IOUNIT/ NU(12)	FMII1708
./ ALTER 34,34	FMII1709
60 READ (IST1'NU(IST1))MS,MR,MREC,MF,(IPF(J),(PF(I,J),I=1,3),J=MS,MR)	FMII1710
./ ALTER 44,44	FMII1711
90 READ (IST2'NU(IST2))MS,MR,MREC,MO,(IPO(J),(PO(I,J),I=1,3),J=MS,MR)	FMII1712
./ ALTER 246,246	FMII1713
READ (IST1'NU(IST1)) MS,MR,MREC,MF	FMII1714
./ ALTER 265,265	FMII1715
READ (IST2'NU(IST2)) MS,MR,MREC,MO	FMII1716
./ ALTER 292,293	FMII1717
820 FIND (IST1'1)	FMII1718
FIND (IST2'1)	FMII1719
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=PFO	FMII1720
./ ALTER 6	FMII1721
COMMON/IOUNIT/ NU(12)	FMII1722
./ ALTER 42,42	FMII1723
30 READ (IT'NU(IT)) MS,MR,MREC,KR,(NP(J),(DP(I,J),I=1,3),J=MS,MR)	FMII1724
./ ALTER 62,62	FMII1725
WRITE (NSET'NU(NSET)) IROW,KODE,M,(DP(I,J),JT(J),J=1,K)	FMII1726
./ ALTER 65,65	FMII1727
WRITE (NSET'NU(NSET)) IROW,KODE,K,DP(I,K)	FMII1728
./ ALTER 68,68	FMII1729
70 FIND (IT'1)	FMII1730
./ ALTER 82,82	FMII1731
WRITE (NSET'NU(NSET)) I, KODE, M, P(I), I	FMII1732
./ ALTER 95,96	FMII1733
READ (IST3'NU(IST3))MD,K,(AK(J),J=1,10),((DK(J,L),J=1,MD),L=1,MD),	FMII1734
1 (LK(J), J=1,K)	FMII1735
./ ALTER 104,104	FMII1736
WRITE (IST2'NU(IST2)) IGBR,KODE,M,J,P(1),IBCOL,ID	FMII1737
./ ALTER 107,107	FMII1738
WRITE (IST2'NU(IST2)) IGBR,KODE,M,J,P(1),IXCOL,ID	FMII1739
./ ALTER 126,126	FMII1740
280 WRITE (IST1'NU(IST1)) LK(L),KODE,M,K,(DK(J,L),LK(J),J=2,MD)	FMII1741
./ ALTER 129,129	FMII1742
WRITE (NSET'NU(NSET)) ICOL,KODE,M,P(1),IBCOL,P(1),IXCOL	FMII1743
./ ALTER 136,136	FMII1744
290 WRITE (NSET'NU(NSET)) ICOL,KODE,M,P(1),IXCOL	FMII1745
./ ALTER 156,156	FMII1746
WRITE (IST2'NU(IST2)) IGBR,KODE,M,J,P(1),N1,DK(1,1),N2,ID	FMII1747
./ ALTER 163,163	FMII1748
WRITE (IST2'NU(IST2)) IGBR,KODE,M,J,AK(10),N1,ID	FMII1749
./ ALTER 168,168	FMII1750
WRITE (IST2'NU(IST2)) IGBR, KODE, M, J, P(1),N2,ID	FMII1751
./ ALTER 175,175	FMII1752
READ (IST3'NU(IST3)) (AK(J),J=1,35), (LK(J),J=31,34)	FMII1753
./ ALTER 196,197	FMII1754
WRITE (IST2'NU(IST2))AK(16),AK(12),AK(13),AK(15),(AK(J),J=21,24)	FMII1755
350 WRITE (NSET'NU(NSET)) ICOL,KODE,M,(AK(J),LK(J),J=31,34),	FMII1756
1 AK(16),IXCOL	FMII1757
./ ALTER 201,204	FMII1758
FIND (IST1'1)	FMII1759
FIND (IST2'1)	FMII1760
./ ALTER 214,214	FMII1761

410 WRITE (NSET*NU(NSET)) IXCOL,KODE,M,P(IXCOL),IXCOL	FMII1762
./ ALTER 216,217	FMII1763
READ (IST1*NU(IST1)) IXCOL,KODE,M,K,(AK(J),LK(J),J=1,K)	FMII1764
420 WRITE (NSET*NU(NSET)) IXCOL,KODE,M,(AK(J),LK(J),J=1,K)	FMII1765
./ ALTER 222,222	FMII1766
430 WRITE (NSET*NU(NSET)) IXCOL,KODE,M,P(IXCOL),IXCOL	FMII1767
./ ALTER 244,244	FMII1768
460 WRITE (NSET*NU(NSET)) I,KODE,M,P(I)	FMII1769
./ ALTER 251,251	FMII1770
470 WRITE (NSET*NU(NSET)) J,KODE,M,P(I)	FMII1771
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=EXT	FMII1772
./ ALTER 6	FMII1773
COMMON/IOUNIT/ NU(12)	FMII1774
./ ALTER 46,46	FMII1775
WRITE (NSET*NU(NSET)) K,KODE,J,ET(IXL),I,ET(IXL),IXL	FMII1776
./ ALTER 59,59	FMII1777
WRITE (NSET*NU(NSET)) J,KODE,J,P(1)	FMII1778
./ ALTER 63,63	FMII1779
510 WRITE (NSET*NU(NSET)) I,KODE,J,P(1),I	FMII1780
./ ALTER 72,72	FMII1781
READ (IST2*NU(IST2)) K1,K2,K3,K4,(AK(J),LK(J),J=1,K4),IP(I)	FMII1782
./ ALTER 74,74	FMII1783
WRITE (NSET*NU(NSET)) K1,K2,K3,(AK(J),LK(J),J=1,K4)	FMII1784
./ ALTER 104,104	FMII1785
READ (IST2*NU(IST2)) (AK(J),J=1,4),(AK(J),J=11,14)	FMII1786
./ ALTER 118,118	FMII1787
WRITE (NSET*NU(NSET)) ICOL,KODE,N,(AK(J),J=1,4)	FMII1788
./ ALTER 120,120	FMII1789
545 WRITE (NSET*NU(NSET)) ICOL,KODE,M,(AK(J),LK(J),J=1,4)	FMII1790
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=USO2	FMII1791
./ ALTER 11	FMII1792
COMMON/IOUNIT/NU(12)	FMII1793
./ ALTER 42,45	FMII1794
FIND (NTAPE1*1)	FMII1795
FIND (NTAPE2*1)	FMII1796
FIND (NTAPE3*1)	FMII1797
FIND (NTAPE4*1)	FMII1798
./ ALTER 99,99	FMII1799
3 NPAGE,ERROR)	FMII1800
CALL DMG3 (NODE,LOAD,NREAC,LEM,MUMENT,JS,NCO,NDIR,JTAPE2,	FMII1801
1 WORK(22),WORK(38),WORK(56),WORK(JE4),NAMOUT,	FMII1802
2 NTAPE2,NROCT,NPAGE,ERROR,NUMOT)	FMII1803
CALL DMG4 (NODE,LOAD,NREAC,LEM,MUMENT,JS,NCO,NDIR,JTAPE4,	FMII1804
1 WORK(1), WORK(5), WORK(9),	FMII1805
2 WORK(13), WORK(44), WORK(50), WORK(JE1),	FMII1806
3 WORK(JE2), WORK(JE3),NAMOUT,NTAPE1,NTAPE2,	FMII1807
4 NTAPE3,NROCT,IPRINT,NPAGE,ERROR)	FMII1808
IF(ERROR) RETURN	FMII1809
CALL DMG5 (NODE,LOAD,NREAC,LEM,MUMENT,JS,NCO,NDIR,KS,MEMO,	FMII1810
1 WORK(1),WORK(5),WORK(9),	FMII1811
2 WORK(13), WORK(22), WORK(34), WORK(38),	FMII1812
3 WORK(44), WORK(50), WORK(56), WORK(62),	FMII1813
4 WORK(206), WORK(350), WORK(494), WORK(638),	FMII1814
5 WORK(JE12), WORK(JE13), WORK(JE14),NAMOUT,	FMII1815
6 NTAPE1,NTAPE2,NTAPE3,JTAPE1,NROCT)	FMII1816
CALL DMG6 (NODE, LOAD, NREAC,LEM,MUMENT,JS,NCO,NDIR,KS,MEMO,	FMII1817
1 WORK(1), WORK(5),WORK(9),	FMII1818

2	WORK(13), WORK(22), WORK(34), WORK(38),	FMII1819
3	WORK(44), WORK(50), WORK(56), WORK(62),	FMII1820
4	WORK(206), WORK(350), WORK(494), WORK(638),	FMII1821
5	WORK(JE12), WORK(JE13), WORK(JE14), NAMOUT,	FMII1822
6	NTAPE1, NTAPE2, NTAPE3, NTAPE4, JTAPE3)	FMII1823
	CALL DMMG7 (NODE, LOAD, NREAC, LEM, MOMENT, JS, NCO, NDIR, KS, MEMO,	FMII1824
1	WORK(1), WORK(145), WORK(149), WORK(JE20),	FMII1825
2	WORK(JE21), WORK(JE22), NAMOUT, NTAPE1, NTAPE2,	FMII1826
3	NTAPE3, NTAPE4, JTAPE5)	FMII1827
	JES1 = 78	FMII1828
	JES2 = JES1 + NREAC	FMII1829
	JES3 = JES2 + NREAC	FMII1830
	JES4 = JES3 + NREAC	FMII1831
	JES5 = JES4 + NREAC * JS	FMII1832
	JES6 = JES5 + 72	FMII1833
	JES7 = JES6 + 72	FMII1834
	CALL DMMG8 (NODE, LOAD, NREAC, LEM, MOMENT, JS, NCO, NDIR,	FMII1835
1	WORK(1), WORK(9), WORK(28), WORK(46),	FMII1836
2	WORK(JES1), WORK(JES2), WORK(JES3), WORK(JES4),	FMII1837
3	WORK(JES5), WORK(JES6), WORK(JES7), NAMOUT(1,1),	FMII1838
4	WORK(58), WORK(69), JTAPE6, JTAPE7, JTAPE8,	FMII1839
5	NTAPE1, NTAPE2, NTAPE3, NTAPE4, WORK(1))	FMII1840
	RETURN	FMII1841
	END	FMII1842
	./ CONTROL COMPRESS=IN, NEWCOMP=OUT, TEXT=SYSPRINT, SEARCH=GRAFO	FMII1843
	./ ALTER 8	FMII1844
	COMMON/IOUNIT/NU(12)	FMII1845
	./ ALTER 25,27	FMII1846
	WRITE(JTAPEX*NU(JTAPEX)) KOLUMN, KOMPR, NVAR, (STIF(JM), METRO(JM),	FMII1847
1	JM=1, JVAR)	FMII1848
	GO TO 140	FMII1849
	130 WRITE(JTAPEX*NU(JTAPEX)) KOLUMN, KOMPR, NCO, (CU(JM), JM=1, NCO)	FMII1850
	./ CONTROL COMPRESS=IN, NEWCOMP=OUT, TEXT=SYSPRINT, SEARCH=MMULT	FMII1851
	./ ALTER 9	FMII1852
	COMMON/IOUNIT/NU(12)	FMII1853
	./ ALTER 19,21	FMII1854
	WRITE(NTAPE4*NU(NTAPE4)) KODE, MOMENT, LAMDAI, LAMDAJ, IP, IQ, IR, IS	FMII1855
	DO 120 I=1, LAMDAJ	FMII1856
	120 WRITE(NTAPE4*NU(NTAPE4)) (ALAT(I, J), J=1, LAMDAJ)	FMII1857
	./ CONTROL COMPRESS=IN, NEWCOMP=OUT, TEXT=SYSPRINT, SEARCH=LAMBOA	FMII1858
	./ CONTROL COMPRESS=IN, NEWCOMP=OUT, TEXT=SYSPRINT, SEARCH=ANALS	FMII1859
	./ ALTER 10	FMII1860
	COMMON/IOUNIT/NU(12)	FMII1861
	./ ALTER 47,48	FMII1862
	117 WRITE(NTAPE1*NU(NTAPE1)) KODE, ID, X, Y, Z, (MS1(I), I=1, 6),	FMII1863
1	(NP1(I), I=1, 6), NTY, JNTM, IFLAG, JNTN, JREV, JDIR	FMII1864
	./ ALTER 164,164	FMII1865
	WRITE(NTAPE1*NU(NTAPE1)) KODE, IDEL, IP, IQ, IR, IS, AT, PI, SI, BPI, E, G,	FMII1866
1	PR, ALPHA	FMII1867
	./ ALTER 224,225	FMII1868
	240 FIND(NTAPE1*1)	FMII1869
	./ CONTROL COMPRESS=IN, NEWCOMP=OUT, TEXT=SYSPRINT, SEARCH=DMMG1	FMII1870
	./ ALTER 11	FMII1871
	COMMON/IOUNIT/NU(12)	FMII1872
	./ ALTER 20,21	FMII1873
	120 READ(NTAPE1*NU(NTAPE1)) KODE, ID, XX, YY, ZZ, (NS1(I), I=1, 6),	FMII1874
1	(NP1(I), I=1, 6), NTY, JNTM, IFLAG, JNTN, JREV, JDIR	FMII1875

./ ALTER 138,145	FMI11876
400 WRITE(NTAPE2'NU(NTAPE2)) DIR(I,1),DIR(I,2),DIR(I,3)	FMI11877
DO 410 I=1,LOAD	FMI11878
410 WRITE(NTAPE2'NU(NTAPE2)) NODAL(I),(DAOL(I,J),J=1,JS)	FMI11879
WRITE(NTAPE2'NU(NTAPE2)) (X(I),Y(I),Z(I),I=1,NODE)	FMI11880
DO 420 I=1,NREAC	FMI11881
420 WRITE(NTAPE2'NU(NTAPE2)) NODAR(I),(REAC(I,J),J=1,JS)	FMI11882
FIND(NTAPE2'1)	FMI11883
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG2	FMI11884
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG3	FMI11885
./ ALTER 8	FMI11886
COMMON/IOUNIT/NU(12)	FMI11887
./ ALTER 33,33	FMI11888
90 READ(NTAPE2'NU(NTAPE2)) DEM,DEM,DEM	FMI11889
./ ALTER 41,41	FMI11890
READ(NTAPE2'NU(NTAPE2)) KODAL,(DC(J),J=1,JS)	FMI11891
./ ALTER 57,57	FMI11892
WRITE(JTAPE2'NU(JTAPE2)) KOLUMN,KOMPR,NVAR,(H(I),LOC(I),I=1,JM)	FMI11893
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG4	FMI11894
./ ALTER 11,11	FMI11895
COMMON/IOUNIT/NU(12)	FMI11896
READ(NTAPE2'NU(NTAPE2)) (X(I),Y(I),Z(I),I=1,NODE)	FMI11897
./ ALTER 28,28	FMI11898
READ(NTAPE1'NU(NTAPE1)) KODE,IDEL,IP,IQ,IR,IS,AT,PI,SI,BPI,E,G,	FMI11899
1 PR,ALPHA	FMI11900
./ ALTER 154,154	FMI11901
215 WRITE(NTAPE3'NU(NTAPE3)) (EL(I,J),J=1,3)	FMI11902
./ ALTER 164,164	FMI11903
WRITE(NTAPE3'NU(NTAPE3)) (CD1(I),CD2(I),CD3(I),I=1,4),X21,Y21,Z21	FMI11904
./ ALTER 179,179	FMI11905
WRITE(JTAPE4'NU(JTAPE4)) KOLUMN,KOMPR,NVAR,CT,JES	FMI11906
./ ALTER 197,197	FMI11907
WRITE(JTAPE4'NU(JTAPE4)) KOLUMN,KOMPR,NVAR,CT,JES,CT,JES1	FMI11908
./ ALTER 204,208	FMI11909
250 FIND (NTAPE1'1)	FMI11910
FIND (NTAPE3'1)	FMI11911
260 READ (NTAPE1'NU(NTAPE1)) KODE,ID,XX,YY,ZZ,(NS1(I),I=1,6),(NP1(I),	FMI11912
1 I=1,6),NTY,JNTM,JREV,JDIR	FMI11913
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG5	FMI11914
./ ALTER 14	FMI11915
COMMON/IOUNIT/NU(12)	FMI11916
./ ALTER 17,17	FMI11917
100 READ(NTAPE1'NU(NTAPE1)) KODE,IDEL,IP,IQ,IR,IS,AT,PI,SI,BPI,E,G,	FMI11918
1 PR,ALPHA	FMI11919
./ ALTER 21,22	FMI11920
110 READ(NTAPE3'NU(NTAPE3)) (EL(I,J),J=1,3)	FMI11921
READ(NTAPE3'NU(NTAPE3)) (CD1(I),CD2(I),CD3(I),I=1,4),X21,Y21,Z21,	FMI11922
./ ALTER 249,252	FMI11923
480 FIND(NTAPE1'1)	FMI11924
FIND(NTAPE3'1)	FMI11925
490 READ(NTAPE1'NU(NTAPE1)) KODE,ID,XX,YY,ZZ,(NS1(I),I=1,6),	FMI11926
1 (NP1(I),I=1,6),NTY,JNTM,IFLAG,JNTN,JREV,JDIR	FMI11927
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG6	FMI11928
./ ALTER 12,12	FMI11929
COMMON/IOUNIT/NU(12)	FMI11930
FIND(NTAPE4'1)	FMI11931
./ ALTER 16,16	FMI11932

100 READ(NTAPE1'NU(NTAPE1))	KODE, IDEL, IP, IQ, IR, IS, AT, PI, SI, BPI, E, G,	FMII1933	
1	PR, ALPHA	FMII1934	
./ ALTER 19,20		FMII1935	
110 READ(NTAPE3'NU(NTAPE3))	(EL(I,J),J=1,3)	FMII1936	
	READ(NTAPE3'NU(NTAPE3))	(CD1(I),CD2(I),CD3(I),I=1,4),X21,Y21,Z21,	FMII1937
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMG6A		FMII1938	
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMG6B		FMII1939	
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMG6C		FMII1940	
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG7		FMII1941	
./ ALTER 11,13		FMII1942	
	COMMON/IOUNIT/NU(12)	FMII1943	
	FIND(NTAPE1'1)	FMII1944	
	FIND(NTAPE3'1)	FMII1945	
	FIND(NTAPE4'1)	FMII1946	
./ ALTER 19,28		FMII1947	
	READ(NTAPE4'NU(NTAPE4))	KODE, MOMENT, LAMDAI, LAMDAJ, IP, IQ, IR, IS	FMII1948
	WRITE(NTAPE1'NU(NTAPE1))	KODE, MOMENT, LAMDAI, LAMDAJ, IP, IQ, IR, IS	FMII1949
	DO 110 I=1, LAMDAJ		FMII1950
	READ (NTAPE4'NU(NTAPE4))	(DNSK(I,J),J=1, LAMDAJ)	FMII1951
110 WRITE(NTAPE1'NU(NTAPE1))	(DNSK(I,J),J=1, LAMDAJ)		FMII1952
120 CONTINUE			FMII1953
	FIND(NTAPE1'1)		FMII1954
	FIND(NTAPE4'1)		FMII1955
./ ALTER 41,49			FMII1956
150 READ(NTAPE1'NU(NTAPE1))	KODE, MOMENT, LAMDAI, LAMDAJ, (IGON(L),L=1,4)		FMII1957
	DO 155 I=1, LAMDAJ		FMII1958
155 READ(NTAPE1'NU(NTAPE1))	(DNSK(I,J),J=1, LAMDAJ)		FMII1959
	FIND (NTAPE4'1)		FMII1960
	GO TO 170		FMII1961
160 READ (NTAPE4'NU(NTAPE4))	KODE, MOMENT, LAMDAI, LAMDAJ, (IGON(L),L=1,4)		FMII1962
	DO 165 I=1, LAMDAJ		FMII1963
165 READ (NTAPE4'NU(NTAPE4))	(DNSK(I,J),J=1, LAMDAJ)		FMII1964
	FIND (NTAPE1'1)		FMII1965
./ ALTER 127,129			FMII1966
280 FIND (NTAPE1'1)			FMII1967
	FIND (NTAPE4'1)		FMII1968
	FIND (NTAPE3'1)		FMII1969
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=DMMG8			FMII1970
./ ALTER 11,12			FMII1971
	COMMON/IOUNIT/NU(12)		FMII1972
	DO 100 I=1,NREAC		FMII1973
	READ (NTAPE2'NU(NTAPE2))	NDAR(I),(REAC(I,J),J=1,JS)	FMII1974
./ ALTER 16,16			FMII1975
	FIND (NTAPE2'1)		FMII1976
./ ALTER 41,43			FMII1977
	WRITE (JTAPE6'NU(JTAPE6))	KDIAG, KOMPR, NVAR, REONE, KDIAG	FMII1978
	WRITE (NTAPE1'NU(NTAPE1))	KDIAG, KOMPR, NVAR, ZERO, KDIAG	FMII1979
	WRITE (NTAPE2'NU(NTAPE2))	KDIAG, KOMPR, NVAR, REONE, KDIAG	FMII1980
./ ALTER 65,67			FMII1981
	WRITE (JTAPE6'NU(JTAPE6))	KDIAG, KOMPR, NVAR, REONE, KDIAG	FMII1982
	WRITE (NTAPE1'NU(NTAPE1))	KDIAG, KOMPR, NVAR, ZERO, KDIAG	FMII1983
	WRITE (NTAPE2'NU(NTAPE2))	KDIAG, KOMPR, NVAR, REONE, KDIAG	FMII1984
./ ALTER 139,141			FMII1985
	WRITE (JTAPE6'NU(JTAPE6))	KDIAG, KOMPR, NVAR, REONE, KDIAG	FMII1986
	WRITE (NTAPE1'NU(NTAPE1))	KDIAG, KOMPR, NVAR, ONE, KDIAG	FMII1987
	WRITE (NTAPE2'NU(NTAPE2))	KDIAG, KOMPR, NVAR, ERO, KDIAG	FMII1988
./ ALTER 208,210			FMII1989

```

WRITE(JTAPE6'NU(JTAPF6)) KDIAG,KOMPR,JVAR,(XL(I,K),NS1(K),K=1,3) FMII1990
WRITE(NTAPE1'NU(NTAPE1)) KDIAG,KOMPR,NVAR,ONE,KDIAG FMII1991
WRITE(NTAPE2'NU(NTAPE2)) KDIAG,KOMPR,NVAR,ERO,KDIAG FMII1992
./ ALTER 226,227 FMII1993
  FIND(NTAPE1'1) FMII1994
  FIND(NTAPE2'1) FMII1995
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=US03 FMII1996
./ CONTROL COMPRESS=IN,NEWCOMP=OUT,TEXT=SYSPRINT,SEARCH=US04 FMII1997
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=US05,LABEL=95000000,NOCOMP FMII1998
  SUBROUTINE US05 (A,B,C,D,E,F,G,H,I,J,K,L,M,N) FMII1999
    LOGICAL K FMII2000
    COMMON NPIT,NPOT FMII2001
    K = .TRUE. FMII2002
    WRITE (NPOT,10) FMII2003
  10 FORMAT (1H0,56HERROR *** THE DUMMY SUBROUTINE FOR US05 HAS BEEN ENFMII2004
    ITERED ) FMII2005
    RETURN FMII2006
  END FMII2007
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=US06,LABEL=96000000,NOCOMP FMII2008
  SUBROUTINE US06 (A,B,C,D,E,F,G,H,I,J,K,L,M,N) FMII2009
    LOGICAL K FMII2010
    COMMON NPIT,NPOT FMII2011
    K = .TRUE. FMII2012
    WRITE (NPOT,10) FMII2013
  10 FORMAT (1H0,56HERROR *** THE DUMMY SUBROUTINE FOR US06 HAS BEEN ENFMII2014
    ITERED ) FMII2015
    RETURN FMII2016
  END FMII2017
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=US07,LABEL=97000000,NOCOMP FMII2018
  SUBROUTINE US07 (A,B,C,D,E,F,G,H,I,J,K,L,M,N) FMII2019
    LOGICAL K FMII2020
    COMMON NPIT,NPOT FMII2021
    K = .TRUE. FMII2022
    WRITE (NPOT,10) FMII2023
  10 FORMAT (1H0,56HERKOR *** THE DUMMY SUBROUTINE FOR US07 HAS BEEN ENFMII2024
    ITERED ) FMII2025
    RETURN FMII2026
  END FMII2027
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=US08,LABEL=98000000,NOCOMP FMII2028
  SUBROUTINE US08 (A,B,C,D,E,F,G,H,I,J,K,L,M,N) FMII2029
    LOGICAL K FMII2030
    COMMON NPIT,NPOT FMII2031
    K = .TRUE. FMII2032
    WRITE (NPOT,10) FMII2033
  10 FORMAT (1H0,56HERROR *** THE DUMMY SUBROUTINE FOR US08 HAS BEEN ENFMII2034
    ITERED ) FMII2035
    RETURN FMII2036
  END FMII2037
./ CONTROL NEWCOMP=OUT,TEXT=SYSPRINT,NAME=US09,LABEL=99000000,NOCOMP FMII2038
  SUBROUTINE US09 (A,B,C,D,E,F,G,H,I,J,K,L,M,N) FMII2039
    LOGICAL K FMII2040
    COMMON NPIT,NPOT FMII2041
    K = .TRUE. FMII2042
    WRITE (NPOT,10) FMII2043
  10 FORMAT (1H0,56HERROR *** THE DUMMY SUBROUTINE FOR US09 HAS BEEN ENFMII2044
    ITERED ) FMII2045

```


-43-

RETURN
END

FMII2046
FMII2047

System/360 JCL for Executing FORMAT II

The JCL has been set up for 10 utility data sets (1, 2, 3, 4, 7, 8, 9, 10, 11, and 12). Data sets 5 and 6 are designated for the normal system input and output.

The number of data sets and their numeric values are fixed in routine 'MRES' (statement NUMR=10) and 'MAIN' (statements DEFINE FILE----and DO 2 I=1, 12 for initialization).

The space for the data sets is fixed in routine 'MAIN' (statements DEFINE FILE----).

The SEP parameter was used to put each of the data sets on separate units to cut down arm motion. Channel separation is not necessary since the data sets are unbuffered for direct access I/O.

The initialization for direct access scratch space takes approximately 10 minutes for the present allocation. Thus, the existing ability to stack jobs in FORMAT II should be used where possible.

For general runs on small problems (less than 100 equations and few abstraction instructions), the amount of space for the data sets should be reduced to cut down on the overhead of initialization for direct access on each run.

Another solution to cutting down the overhead for initialization is to save the scratch data sets on disk; and therefore, there is no initialization on each run after the first.

The compiled routines for FORMAT II were put on a tape to facilitate access by the engineering groups.

FORMAT II DECK SET UP

//	(JOB CARD)	FMII0001
//STEP1 EXEC	FORTGLG, PARM.LKED='MAP,LIST,OVLY',	\$FMII0002
//	PARM.GO='DUMP=ERROR', REGION.GO=208K, TIME.GO=25	FMII0003
//LKED.SYSLMOD DD	SPACE=(3520,(200,20,1))	FMII0004
//LKED.SYSUT1 DD	SPACE=(3520,(250,20))	FMII0005
//LKED.PHASE1 DD	DSNAME=FORMATII, UNIT=9-TRACK, LABEL=(1,SL),	\$FMII0006
//	DISP=(OLD,KEEP), VOLUME=(,RETAIN,SER=XXXXXX)	FMII0007
//LKED.SYSIN DD *		FMII0008
INCLUDE PHASE1		FMII0009
OVERLAY FMT000		FMII0010
INSERT PREP,PUTL1,PUTL2,PUTL3,PUTL4		FMII0011
OVERLAY FMT100		FMII0012
INSERT MRES,MRES1,MRES2,MRES11		FMII0013
OVERLAY FMT100		FMII0014
INSERT PROB		FMII0015
OVERLAY FMT100		FMII0016
INSERT INST		FMII0017
OVERLAY FMT110		FMII0018
INSERT INST01,INST02,INST03,INSTFP,INST10,INST16,INST33		FMII0019
OVERLAY FMT110		FMII0020
INSERT INST20,INST31,INST32,INST41,INST90		FMII0021
OVERLAY FMT100		FMII0022
INSERT MATR,MATR1		FMII0023
OVERLAY FMT100		FMII0024
INSERT ALOC		FMII0025
OVERLAY FMT120		FMII0026
INSERT ALOC1,ALOC11,ALOC12,ALOC13		FMII0027
OVERLAY FMT120		FMII0028
INSERT ALOC2,ALOC21,ALOC22,ALOC23,ALOC24		FMII0029
OVERLAY FMT120		FMII0030
INSERT ALOC3,ALOC31		FMII0031
OVERLAY FMT120		FMII0032
INSERT ALOC4		FMII0033
OVERLAY FMT100		FMII0034
INSERT SPCL		FMII0035
OVERLAY FMT100		FMII0036
INSERT LOGC,LOGC1		FMII0037
OVERLAY FMT000		FMII0038
INSERT EXEQ,EUTL1,EUTL2,EUTL3,EUTL4,EUTL5,EUTL6,EUTL7		FMII0039
INSERT EUTL8,EUTL9		FMII0040
OVERLAY FMT200		FMII0041
OVERLAY FMT210		FMII0042
INSERT PRNT,PRNT11		FMII0043
OVERLAY FMT210		FMII0044
INSERT IFCN		FMII0045
OVERLAY FMT210		FMII0046
INSERT MADD,MADDA,MADDA1,MADD1,MADD2,MADD3		FMII0047
OVERLAY FMT210		FMII0048
INSERT EMPY,EMPYA,EMPYA1,EMPY1		FMII0049
OVERLAY FMT210		FMII0050
INSERT TMPY,TMPY1,TMPY2,TMPY3,TMPY4		FMII0051

OVERLAY FMT210
INSERT MULT,MULT1,MULT2,MULT3,MULT4
OVERLAY FMT210
INSERT SMPY,SMPYA,SMPYA1,SMPY1
OVERLAY FMT210
INSERT POWR
OVERLAY FMT210
INSERT TRAN,TRAN1,TRAN2,TRAN3
OVERLAY FMT210
INSERT INVT,INVT1,INVT11
OVERLAY FMT200
INSERT SEQE,SEQE1,SEQE11
OVERLAY FMT200
OVERLAY FMT220
INSERT SEQ1,SEQ11
OVERLAY FMT220
INSERT STRC,STRC1,STRC11,STRC12
OVERLAY FMT220
INSERT EIGN,EIGN1
OVERLAY FMT220
INSERT ADJN
OVERLAY FMT220
INSERT ENVR
OVERLAY FMT220
INSERT ENVC,ENVC1
OVERLAY FMT220
INSERT DIAG,DIAG1
OVERLAY FMT220
INSERT NAME,NAME1
OVERLAY FMT220
INSERT USC1,SRT,VECT
OVERLAY B
INSERT COORD
OVERLAY B
INSERT VECRD
OVERLAY B
INSERT BAPRD
OVERLAY B
INSERT PANRD
OVERLAY B
INSERT PSORT,TRD,TRDW,WRT,SORT
OVERLAY B
INSERT ELIM
OVERLAY B
INSERT PFC
OVERLAY B
INSERT EXT
OVERLAY FMT220
INSERT USC2,GRAFO,MMULT,LAMBDA
OVERLAY BETA
INSERT ANALS
OVERLAY BETA
INSERT DMMG1
OVERLAY BETA
INSERT DMMG2
OVERLAY BETA
INSERT DMMG3

FMI10052
FMI10053
FMI10054
FMI10055
FMI10056
FMI10057
FMI10058
FMI10059
FMI10060
FMI10061
FMI10062
FMI10063
FMI10064
FMI10065
FMI10066
FMI10067
FMI10068
FMI10069
FMI10070
FMI10071
FMI10072
FMI10073
FMI10074
FMI10075
FMI10076
FMI10077
FMI10078
FMI10079
FMI10080
FMI10081
FMI10082
FMI10083
FMI10084
FMI10085
FMI10086
FMI10087
FMI10088
FMI10089
FMI10090
FMI10091
FMI10092
FMI10093
FMI10094
FMI10095
FMI10096
FMI10097
FMI10098
FMI10099
FMI10100
FMI10101
FMI10102
FMI10103
FMI10104
FMI10105
FMI10106
FMI10107
FMI10108

OVERLAY BETA	FMIIC109
INSERT DMMG4	FMIIC110
OVERLAY BETA	FMIIC111
INSERT DMMG5	FMIIC112
OVERLAY BETA	FMIIC113
INSERT DMMG6	FMIIC114
OVERLAY GAMMA	FMIIC115
INSERT DMG6A	FMIIC116
OVERLAY GAMMA	FMIIC117
INSERT DMG6B	FMIIC118
OVERLAY GAMMA	FMIIC119
INSERT DMG6C	FMIIC120
OVERLAY BETA	FMIIC121
INSERT DMMG7	FMIIC122
OVERLAY BETA	FMIIC123
INSERT DMMG8	FMIIC124
OVERLAY FMT220	FMIIC125
INSERT US03	FMIIC126
OVERLAY FMT220	FMIIC127
INSERT US04	FMIIC128
OVERLAY FMT220	FMIIC129
INSERT USC5	FMIIC130
OVERLAY FMT220	FMIIC131
INSERT USC6	FMIIC132
OVERLAY FMT220	FMIIC133
INSERT US07	FMIIC134
OVERLAY FMT220	FMIIC135
INSERT USC8	FMIIC136
OVERLAY FMT220	FMIIC137
INSERT US09	FMIIC138
/*	FMIIC139
//GO.FT06F001 DD SPACE=(3520,(200,50))	FMIIC140
//GO.SNAPDUMP DD UNIT=(DISK,4)	FMIIC141
//GO.FT01F001 DD DSNAME=&FT01,	\$FMIIC142
// UNIT=DISK,SPACE=(808,(500)),	\$FMIIC143
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC144
//GO.FT02F001 DD DSNAME=&FT02,	\$FMIIC145
// UNIT=(DISK,SEP=FT01F001),	\$FMIIC146
// SPACE=(808,(2500)),	\$FMIIC147
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC148
//GO.FT03F001 DD DSNAME=&FT03,	\$FMIIC149
// UNIT=(DISK,SEP=(FT01F001,FT02F001)),	\$FMIIC150
// SPACE=(808,(2500)),	\$FMIIC151
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC152
//GO.FT04F001 DD DSNAME=&FT04,	\$FMIIC153
// UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001)),	\$FMIIC154
// SPACE=(808,(2500)),	\$FMIIC155
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC156
//GO.FT07F001 DD DSNAME=&FT07,	\$FMIIC157
// UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001,FT04F001)),	\$FMIIC158
// SPACE=(808,(2500)),	\$FMIIC159
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC160
//GO.FT08F001 DD DSNAME=&FT08,	\$FMIIC161
// UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001,FT04F001,	\$FMIIC162
// FT07F001)),	\$FMIIC163
// SPACE=(808,(2500)),	\$FMIIC164
// DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIC165

//GO.FT09F001	DD DSNAME=&FT09,	\$FMII0166
//	UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001,FT04F001,	\$FMII0167
//	FT07F001,FT08F001)),	\$FMII0168
//	SPACE=(808,(3500)),	\$FMII0169
//	DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIO170
//GO.FT10F001	DD DSNAME=&FT10,	\$FMII0171
//	UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001,FT04F001,	\$FMII0172
//	FT07F001,FT08F001,FT09F001)),	\$FMII0173
//	SPACE=(808,(2500)),	\$FMII0174
//	DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIO175
//GO.FT11F001	DD DSNAME=&FT11,	\$FMII0176
//	UNIT=(DISK,SEP=(FT01F001,FT02F001,FT03F001,FT04F001,	\$FMII0177
//	FT07F001,FT08F001,FT09F001,FT10F001)),	\$FMII0178
//	SPACE=(808,(2500)),	\$FMII0179
//	DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIO180
//GO.FT12F001	DD DSNAME=&FT12,	\$FMII0181
//	UNIT=DISK,	\$FMII0182
//	SPACE=(808,(2500)),	\$FMII0183
//	DCB=(RECFM=V,LRECL=804,BLKSIZE=808,BUFNO=1)	FMIIO184
//GO.SYSIN DD *	FORMAT II DATA FOLLOWS	FMIIO185
/*		FMIIO186

Timing Runs of the System/360 for FORMAT II

System/360 Execution Times

PROBLEM	EXECUTION TIME	
	PREP	EXEQ
Example 1	29.2 sec.	240.4 sec.
Example 2	20.1 sec.	191.8 sec.

System/360 timings are with MVT and would be shorter with single job processing.

Timing Comparison of System/360 and 7094

	System/360	7094
Example 1	4.5 min.	8.7 min.
Example 2	3.3 min.	6.2 min.

- (1) System/360 runs were made with a Model 75.
7094 times were obtained from Reference (1), page 101.
- (2) Initialization of direct access scratch space takes approximately ten minutes for the present allocation and is not included in the System/360 timings. For suggestions in reducing or eliminating this initialization time see the section of this volume entitled System/360 JCL for Executing FORMAT II.

REVISED PROGRAM LISTING

The listing and punched deck of the revised FORMAT II source deck may be obtained by using the following program to list the FORMAT II source tape.

```
// EXEC FORTGCLG
//SYSIN DD *
```

```
    INTEGER IA(20)
    DATA I0001/'0001'/
10  READ (9,4) IA
    4  FORMAT(20A4)
    IF (IA(20)-I0001) 11,20,11
11  WRITE (6,6) IA
    6  FORMAT (1X,20A4)
    WRITE (7,4) IA
    GO TO 10
20  WRITE (6,7)
    7  FORMAT ('1')
    GO TO 11
END
```

```
/*
//GO.FT09F001 DD DSNAME=SOURCE,DISP=(OLD,KEEP),UNIT=9-TRACK,
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=3520),
//              VOL=SER=FORMAT,LABEL=(1,NL)
//GO.FT07F001  (PUNCHED OUTPUT DATA SET)
```

\$
\$

Appendix-Illustrative Example

This appendix includes sample output data from the System/360 for the spaceframe-force analysis problem Example 1. Output from the 7094 for the same problem is contained in Appendix I, Volume I.

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX DELTA PAGE 1

CUTOFF = 0.0

SIZE 12 8Y 12

COL	ROW	ROW	ROW	ROW	ROW			
1	1	0.326516E-05	2	-0.384261E-06	3	-0.116114E-04	4	-0.225424E-04
	5	0.817885E-06	6	-0.374219E-06	7	-0.224866E-04	8	-0.504109E-05
	9	-0.712028E-06	10	0.294725E-07	11	-0.127307E-04	12	0.388970E-05
2	1	-0.388287E-06	2	0.326122E-05	3	0.447094E-06	4	-0.223880E-04
	5	0.503031E-05	6	0.115431E-04	7	-0.224423E-04	8	-0.806435E-06
	9	-0.269756E-07	10	0.709690E-06	11	-0.383301E-05	12	0.126770E-04
3	1	-0.116501E-04	2	0.322063E-06	3	0.112457E-02	4	0.307416E-03
	5	0.325766E-03	6	0.888333E-03	7	0.307489E-03	8	0.391867E-03
	9	0.648203E-04	10	0.505732E-04	11	0.831515E-03	12	0.661505E-03
4	1	-0.225599E-04	2	-0.224951E-04	3	0.307066E-03	4	0.136383E-02
	5	-0.182660E-03	6	-0.306139E-03	7	0.136156E-02	8	0.182792E-03
	9	0.208401E-04	10	-0.208093E-04	11	0.526122E-03	12	0.525385E-03
5	1	0.815631E-06	2	0.503816E-05	3	0.325949E-03	4	-0.182589E-03
	5	0.174638E-03	6	0.391731E-03	7	-0.182610E-03	8	0.972338E-04
	9	0.178911E-04	10	0.231555E-04	11	0.190347E-03	12	0.343760E-03
6	1	-0.347888E-06	2	0.116193E-04	3	0.888512E-03	4	-0.306728E-03
	5	0.391754E-03	6	0.112379E-02	7	-0.306778E-03	8	0.325795E-03
	9	0.505799E-04	10	0.647937E-04	11	0.661631E-03	12	0.830947E-03

52

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

PAGE 2

MATRIX DELTA

CUTOFF = 0.0

SIZE 12 BY 12

COL	ROW	ROW	ROW	ROW
7	1	2	3	4
5	5	6	7	8
9	9	10	11	12
COL	8	2	3	4
5	5	6	7	8
9	9	10	11	12
COL	9	2	3	4
5	5	6	7	8
9	9	10	11	12
COL	10	2	3	4
5	5	6	7	8
9	9	10	11	12
COL	11	2	3	4
5	5	6	7	8
9	9	10	11	12
COL	12	2	3	4
5	5	6	7	8
9	9	10	11	12
ISTNO	16	TIME		
ISTNO	17	TIME		
ISTNO	18	TIME		
ISTNO	19	TIME		
ISTNO	20	TIME		
ISTNO	21	TIME		
ISTNO	22	TIME		
ISTNO	23	TIME		
ISTNO	24	TIME		
ISTNO	25	TIME		
ISTNO	26	TIME		
ISTNO	27	TIME		

-53-

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX DEFLN PAGE 1

CUTOFF = 0.0

SIZE 4 BY 12

	COND.	COND.	COND.	COND.				
VECTOR 1	1	-0.804969E-03	2	-0.498552E-02	3	-0.579049E-02	4	-0.383447E-03
VECTOR 2	1	0.810962E-03	2	-0.498282E-02	3	-0.417186E-02	4	0.383402E-03
VECTOR 3	1	0.438104E 00	2	0.855501E-01	3	0.523655E 00	4	-0.135130E-01
VECTOR 4	1	0.140447E-03	2	0.362614E 00	3	0.362755E 00	4	0.385990E-01
VECTOR 5	1	0.158753E 00	2	-0.446222E-01	3	0.114131E 00	4	0.901282E-02
VECTOR 6	1	0.438209E 00	2	-0.852420E-01	3	0.352967E 00	4	-0.135135E-01
VECTOR 7	1	-0.304908E-03	2	0.362680E 00	3	0.362376E 00	4	-0.385979E-01
VECTOR 8	1	0.158734E 00	2	0.446660E-01	3	0.203400E 00	4	0.901290E-02
VECTOR 9	1	0.269627E-01	2	0.723357E-02	3	0.341963E-01	4	0.209773E-02
VECTOR 10	1	0.269664E-01	2	-0.722331E-02	3	0.197431E-01	4	0.209771E-02

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

PAGE 2

MATRIX DEFN

CUTOFF = 0.0

SIZE 4 BY 12

COND.

COND.

COND.

COND.

VECTOR 11 1 0.326180E 00 2 0.138328E 00 3 0.464508E 00 4 0.581721E-01

VECTOR 12 1 0.326259E 00 2 -0.138087E 00 3 0.188172E 00 4 0.581717E-01

ISTNO = 28 TIME = 2.383 SECONDS
 ISTNO = 29 TIME = 2.984 SECONDS
 ISTNO = 30 TIME = 3.016 SECONDS
 ISTNO = 31 TIME = 2.035 SECONDS
 ISTNO = 32 TIME = 8.082 SECONDS
 ISTNO = 33 TIME = 8.234 SECONDS
 ISTNO = 34 TIME = 1.363 SECONDS
 ISTNO = 35 TIME = 1.820 SECONDS
 ISTNO = 36 TIME = 1.332 SECONDS
 ISTNO = 37 TIME = 1.750 SECONDS

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX REACT PAGE 1

CUTOFF = 0.0

SIZE 4 BY 12

	COND.	COND.	COND.	COND.					
ROW	1	1	-0.101377E 03	2	-0.173964E 03	3	-0.275341E 03	4	0.354661E 01
ROW	2	1	-0.196770E 03	2	-0.287964E 03	3	-0.484734E 03	4	0.142490E 01
ROW	3	1	-0.361664E 03	2	-0.567680E 03	3	-0.929345E 03	4	-0.712455E 02
ROW	4	1	0.101370E 03	2	-0.173959E 03	3	-0.725892E 02	4	-0.354657E 01
ROW	5	1	-0.196754E 03	2	0.287964E 03	3	0.912099E 02	4	0.142489E 01
ROW	6	1	-0.361681E 03	2	0.567687E 03	3	0.206007E 03	4	-0.712454E 02
ROW	7	1	-0.242330E 03	2	-0.874451E 02	3	-0.329775E 03	4	0.854900E 00
ROW	8	1	-0.106445E 01	2	0.182608E 02	3	0.171964E 02	4	0.285700E 02
ROW	9	1	-0.297161E 02	2	0.677813E 03	3	0.648098E 03	4	-0.113995E 01
ROW	10	1	-0.242339E 03	2	0.874448E 02	3	-0.154895E 03	4	0.854983E 00

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX REACT PAGE 2

CUTOFF = 0.0 SIZE 4 BY 12

		COND.		COND.		COND.				
ROW		11	1	0.107217E 01	2	0.182442E 02	3	0.193164E 02	4	-0.285699E 02
ROW		12	1	-0.297420E 02	2	-0.677813E 03	3	-0.707556E 03	4	-0.113989E 01

MATRIX BAR PAGE 1

COND.	COND.	COND.	COND.
COND.	COND.	COND.	COND.

ROW	1	1	-0.211557E 03	2	-0.321128E 03	3	-0.532686E 03	4	-0.598992E 01
ROW	2	1	-0.211557E 03	2	-0.321128E 03	3	-0.532686E 03	4	-0.598992E 01
ROW	3	1	-0.211540E 03	2	0.321145E 03	3	0.109606E 03	4	-0.598993E 01
ROW	4	1	-0.211540E 03	2	0.321145E 03	3	0.109606E 03	4	-0.598993E 01
ROW	5	1	0.289574E 03	2	0.430979E 03	3	0.720554E 03	4	0.260195E 01
ROW	6	1	0.289574E 03	2	0.430979E 03	3	0.720554E 03	4	0.260195E 01
ROW	7	1	-0.225845E 01	2	-0.235081E 02	3	-0.257666E 02	4	0.635300E 01
ROW	8	1	0.110409E-03	2	-0.160358E-02	3	-0.149318E-02	4	0.856676E-03
ROW	9	1	0.903381E 02	2	0.940323E 03	3	0.103066E 04	4	-0.254119E 03
ROW	10	1	0.140965E 02	2	0.278290E 02	3	0.419256E 02	4	0.459682E 01

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAR PAGE 2

CUTOFF = 0.0

SIZE 4 BY 120

	COND.	COND.	COND.	COND.
ROW 11	1	-0.361664E 03	2	-0.567681E 03
			3	-0.929346E 03
			4	-0.712455E 02
ROW 12	1	0.202198E 03	2	0.545479E 03
			3	0.747676E 03
			4	0.112627E 03
ROW 14	1	0.289552E 03	2	-0.430989E 03
			3	-0.141438E 03
			4	0.260196E 01
ROW 15	1	0.289552E 03	2	-0.430989E 03
			3	-0.141438E 03
			4	0.260196E 01
ROW 16	1	-0.225942E 01	2	0.234960E 02
			3	0.212365E 02
			4	0.635298E 01
ROW 17	1	0.342049E-03	2	0.795663E-03
			3	0.113771E-02
			4	0.856675E-03
ROW 18	1	0.903773E 02	2	-0.939838E 03
			3	-0.849460E 03
			4	-0.254118E 03
ROW 19	1	-0.140953E 02	2	0.278398E 02
			3	0.137445E 02
			4	-0.459683E 01
ROW 20	1	0.361681E 03	2	-0.567688E 03
			3	-0.206007E 03
			4	0.712454E 02
ROW 21	1	-0.202133E 03	2	0.545906E 03
			3	0.343773E 03
			4	-0.112628E 03

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

		MATRIX BAR				PAGE			
		CUTOFF = 0.0				SIZE 4 BY 120			
		COND.		COND.		COND.		COND.	
ROW	23	1	0.289571E 03	2	0.442639E 03	3	0.732211E 03	4	0.260195E 01
ROW	24	1	0.150003E 03	2	0.162757E 03	3	0.312759E 03	4	-0.104904E-04
ROW	25	1	-0.116480E 02	2	0.292716E 02	3	0.176236E 02	4	-0.404330E 02
ROW	26	1	-0.200803E 03	2	0.394041E 03	3	0.193237E 03	4	-0.420564E 03
ROW	27	1	0.321562E 02	2	-0.191392E 03	3	-0.159235E 03	4	0.328094E 03
ROW	28	1	0.381623E 01	2	-0.157624E 02	3	-0.119461E 02	4	-0.563142E 01
ROW	29	1	-0.763329E 02	2	0.241643E 03	3	0.165310E 03	4	0.112628E 03
ROW	30	1	-0.824094E-02	2	-0.736044E 02	3	-0.736127E 02	4	-0.381469E-03
ROW	31	1	-0.528261E 02	2	0.988169E 01	3	-0.429445E 02	4	-0.786781E-04
ROW	32	1	0.289555E 03	2	-0.442649E 03	3	-0.153095E 03	4	0.260197E 01

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAR PAGE 4

CUTOFF = 0.0

SIZE 4 BY 120

	COND.		COND.		COND.		COND.		COND.
ROW	33	1	0.149997E 03	2	-0.162757E 03	3	-0.127593E 02	4	0.104904E-04
ROW	34	1	-0.116552E 02	2	-0.292425E 02	3	-0.408978E 02	4	-0.404329E 02
ROW	35	1	-0.200868E 03	2	-0.393542E 03	3	-0.594411E 03	4	-0.480563E 03
ROW	36	1	0.322361E 02	2	0.191309E 03	3	0.223545E 03	4	0.328094E 03
ROW	37	1	-0.382081E 01	2	-0.157836E 02	3	-0.196044E 02	4	0.563135E 01
ROW	38	1	0.764079E 02	2	0.242067E 03	3	0.318475E 03	4	-0.112627E 03
ROW	39	1	-0.824094E-02	2	-0.736044E 02	3	-0.736127E 02	4	-0.381469E-03
ROW	40	1	0.528315E 02	2	0.985840E 01	3	0.626899E 02	4	-0.625848E-04
ROW	41	1	0.297159E 02	2	-0.677813E 03	3	-0.648098E 03	4	0.113995E 01
ROW	42	1	0.297159E 02	2	-0.677813E 03	3	-0.648098E 03	4	0.113995E 01

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAR

PAGE 5

CUTOFF = 0.0

SIZE 4 BY 120

	COND.	COND.	COND.	COND.	COND.
ROW 43	1	-0.106422E 01	2	0.182613E 02	3 0.171970E 02 4 0.285700E 02
ROW 44	1	-0.125496E-03	2	0.138454E-02	3 0.125905E-02 4 0.252258E-03
ROW 45	1	0.106421E 02	2	-0.182612E 03	3 -0.171969E 03 4 -0.285700E 03
ROW 46	1	0.242330E 03	2	0.874460E 02	3 0.329776E 03 4 -0.854908E 00
ROW 47	1	-0.299950E-02	2	-0.285053E-02	3 -0.585003E-02 4 0.350720E-04
ROW 48	1	0.242330E 04	2	0.874457E 03	3 0.329775E 04 4 -0.854904E 01
ROW 50	1	0.297421E 02	2	0.677813E 03	3 0.707556E 03 4 0.113989E 01
ROW 51	1	0.297421E 02	2	0.677813E 03	3 0.707556E 03 4 0.113989E 01
ROW 52	1	-0.107233E 01	2	-0.182446E 02	3 -0.193169E 02 4 0.285699E 02
ROW 53	1	-0.100762E-03	2	-0.967037E-03	3 -0.106780E-02 4 0.252590E-03

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAR

PAGE 6

CUTOFF = 0.0

SIZE 4 BY 120

	COND.		COND.		COND.		COND.		
ROW	54	1	0.107232E 02	2	0.182445E 03	3	0.193168E 03	4	-0.285699E 03
ROW	55	1	-0.242340E 03	2	0.874451E 02	3	-0.154894E 03	4	0.854978E 00
ROW	56	1	0.299955E-02	2	-0.285046E-02	3	0.149092E-03	4	-0.350731E-04
ROW	57	1	-0.242339E 04	2	0.874448E 03	3	-0.154894E 04	4	0.854975E 01
ROW	59	1	-0.102433E 03	2	-0.363705E 03	3	-0.466137E 03	4	-0.260195E 01
ROW	60	1	0.371364E 02	2	-0.838223E 02	3	-0.466859E 02	4	0.762939E-05
ROW	61	1	0.515519E 00	2	-0.184975E 03	3	-0.184460E 03	4	-0.266256E 02
ROW	62	1	0.836946E 01	2	-0.188509E 04	3	-0.187672E 04	4	-0.351965E 03
ROW	63	1	-0.194092E 01	2	0.181442E 04	3	0.181248E 04	4	0.180547E 03
ROW	64	1	-0.158665E 03	2	-0.720856E 02	3	-0.230751E 03	4	0.427411E 00

PAGE 7

CUTOFF = 0.0

MATRIX BAR

SIZE	4	RY	120
------	---	----	-----

COND.	COND.	COND.	COND.
COND.	COND.	COND.	COND.

ROW	COND.	COND.	COND.
65	1	0.317330F 04	2
	2	0.123279F 04	3
	3	0.440610E 04	4
	4	-0.854858F 01	

ROW	1	2	3	4
66	-0.118256E-02	-0.208920E 03	-0.208921E 03	-0.356674E-03

R	O	W
67	I	C.458813E-02
	2	0.344406E 02
	3	0.344452F 02
	4	-J.II8017E-J3

RNW	68	1	-0.102416E 03	2		0.363715F 03	3		0.261299E 03	4		-0.260197E 01
-----	----	---	---------------	---	--	--------------	---	--	--------------	---	--	---------------

ROW								
69	1	C.371416E 02	2	0.838223E 02	3	0.120964F 03	4	-0.762939F-05

ROW	70	1	0.522408E 00	2	0.184956E 03	3	0.185479E 03	4	-0.266257E 02
-----	----	---	--------------	---	--------------	---	--------------	---	---------------

[illegible]

ROW	72	1	-0.201935E 01	2	-0.181442E 04	3	-0.181644E 04	4	0.180547E 03
-----	----	---	---------------	---	---------------	---	---------------	---	--------------

[illegible][illegible]

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

		MATRIX		BAR		SIZE		4 BY		120		PAGE	
		CUTOFF = 0.0										8	
COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.
ROW	75	1	-0.118256E-02	2	-0.208920E 03	3	-0.208921E 03	4	-0.356674E-03				
ROW	76	1	0.297546E-03	2	0.344487E 02	3	0.344490E 02	4	-0.255704E-03				
ROW	77	1	0.102813E 02	2	0.566297E 02	3	0.669111E 02	4	0.102283E 02				
ROW	78	1	-0.268848E 03	2	-0.362446E 03	3	-0.631294E 03	4	0.502421E 01				
ROW	79	1	0.102743E 02	2	-0.566605E 02	3	-0.463863E 02	4	0.102282E 02				
ROW	80	1	-0.268850E 03	2	0.362435E 03	3	0.935857E 02	4	0.502438E 01				
ROW	81	1	-0.461841E 02	2	-0.664204E 02	3	-0.112605E 03	4	-0.563142E 01				
ROW	82	1	0.232944E 03	2	0.148660E 03	3	0.381604E 03	4	-0.427420E 00				
ROW	83	1	-0.461790E 02	2	0.564416E 02	3	0.202626E 02	4	-0.563135E 01				
ROW	84	1	0.232947E 03	2	-0.148660E 03	3	0.842864E 02	4	-0.427514E 00				

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

CUTOFF = 0.0				MATRIX BAP		SIZE 4 BY 120		PAGE 9
COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	
ROW 85	1	0.938938F 01	2	-0.527795E 02	3	-0.433902E 02	4	0.467859F 02
ROW 86	1	0.939579E 01	2	0.527385F 02	3	0.621344E 02	4	0.467859F 02
ROW 87	1	-0.190735E-03	2	-0.130377F 02	3	-0.130379F 02	4	0.470877E-05
ROW 88	1	-0.132869F 03	2	-0.195577E 03	3	-0.328446E 03	4	0.786781F-04
ROW 89	1	-0.132863E 03	2	0.195553E 03	3	0.626899E 02	4	-0.625849F-04
ROW 90	1	0.358486E-02	2	-0.116601F 02	3	-0.116565F 02	4	0.476837E-05
ROW 91	1	-0.802521E 02	2	0.174895F 03	3	0.946426F 02	4	0.226445E 03
ROW 92	1	-0.801446F 02	2	-0.174907E 03	3	-0.255052E 03	4	0.226445E 03
ROW 93	1	-0.399590F-02	2	0.253048F 02	3	0.253008E 02	4	-0.549316E-03
ROW 94	1	0.883522E 02	2	0.129271E 03	3	0.217624E 03	4	-0.404330F 02

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAR PAGE 10

CUTOFF = 0.0

	COND.		COND.		COND.	SIZE	4	RY	120	COND.
ROW	95	1	0.883448E 02	2	-0.129243F 03	3	-0.408978E 02	4	-0.404329E 02	
ROW	96	1	0.177383F-03	2	0.658009E 00	3	0.658186E 00	4	-0.470876E-05	
ROW	97	1	-0.528259E 02	2	0.988180E 01	3	-0.429443E 02	4	-0.786781F-04	
ROW	98	1	-0.528313E 02	2	-0.985847F 01	3	-0.626899E 02	4	0.625848E-04	
ROW	99	1	0.266266E-02	2	0.127567E 02	3	0.127593F 02	4	-0.104904E-04	
ROW	100	1	0.321562E 02	2	-0.191392E 03	3	-0.159235E 03	4	0.328094E 03	
ROW	101	1	0.322361F 02	2	0.191309F 03	3	0.223545F 03	4	0.328094E 03	
ROW	102	1	-0.823975E-02	2	-0.736045E 02	3	-0.736128E 02	4	-0.381470E-03	
ROW	103	1	0.975310E 02	2	0.527726E 02	3	0.150304E 03	4	0.580020E 02	
ROW	104	1	0.975246E 02	2	-0.527450F 02	3	0.447796E 02	4	0.580021E 02	

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX BAP

PAGE 11

CUTOFF = 0.0

	CUTOFF =	0.0	COND.	COND.	COND.	COND.	SIZE	4 BY	120
ROW	105	1	0.162601E-03	2	0.229631F 01	3	0.229648F 01	4	-0.124574E-04
ROW	106	1	0.458479E-02	2	0.344407E 02	3	0.344453E 02	4	-0.118017E-03
ROW	107	1	-0.293255E-03	2	-0.344487E 02	3	-0.344490F 02	4	0.255704E-03
ROW	108	1	-0.407219E-03	2	0.113491E 03	3	0.113491E 03	4	-0.228882F-04
ROW	109	1	-0.227941E 01	2	-0.170247E 04	3	-0.170475E 04	4	-0.662668E 02
ROW	110	1	-0.229162E 01	2	0.170226E 04	3	0.169997E 04	4	-0.662675E 02
ROW	111	1	-0.110779F-01	2	0.391643F 03	3	0.391632E 03	4	-0.427246F-03
ROW	112	1	0.562240E 02	2	-0.129267E 03	3	-0.730426E 02	4	-0.266256F 02
ROW	113	1	0.562314F 02	2	0.129247E 03	3	0.185479E 03	4	-0.266257E 02
ROW	114	1	-0.162601E-03	2	-0.229631E 01	3	-0.229647E 01	4	0.124574E-04

MATRIX BAR

PAGE 12

-69-

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

		CUTOFF = 0.0		MATRIX PANEL		SIZE		BY		PAGE	
COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.	COND.
ROW	1	1	-0.644779E-02	2	-0.105518E 03	3	-0.105525E 03	4	0.887941E-04		
ROW	2	1	-0.859705E-02	2	-0.140691E 03	3	-0.140699E 03	4	0.118392E-03		
ROW	3	1	-0.644779E-02	2	-0.105518E 03	3	-0.105525E 03	4	0.887941E-04		
ROW	4	1	-0.859705E-02	2	-0.140691E 03	3	-0.140699E 03	4	0.118392E-03		
ROW	5	1	0.739861E-02	2	0.258514E 03	3	0.258521E 03	4	-0.610352E-04		
ROW	6	1	0.986480E-02	2	0.344685E 03	3	0.344694E 03	4	-0.813802E-04		
ROW	7	1	0.739860E-02	2	0.258514E 03	3	0.258521E 03	4	-0.610351E-04		
ROW	8	1	0.986480E-02	2	0.344685E 03	3	0.344694E 03	4	-0.813802E-04		
ROW	9	1	-0.139569E 03	2	-0.279883E 03	3	-0.419452E 03	4	-0.260196E 01		
ROW	10	1	-0.279138E 03	2	-0.559766E 03	3	-0.838904E 03	4	-0.520392E 01		

EXAMPLE 1 - SPACE FRAME - FORCE ANALYSIS

MATRIX FLOW PAGE 1

CUTOFF = 0.0

SIZE 4 BY 16

	COND.		COND.		COND.		COND.
ROW	1	1	-0.214926E-03	2	-0.351727E 01	3	-0.351749E 01 4 0.295980E-05
ROW	2	1	-0.214926E-03	2	-0.351727E 01	3	-0.351749E 01 4 0.295980E-05
ROW	3	1	-0.214926E-03	2	-0.351727E 01	3	-0.351749E 01 4 0.295980E-05
ROW	4	1	-0.214926E-03	2	-0.351727E 01	3	-0.351749E 01 4 0.295980E-05
ROW	5	1	0.246620E-03	2	0.861713E 01	3	0.861737E 01 4 -0.203451E-05
ROW	6	1	0.246620E-03	2	0.861713E 01	3	0.861736E 01 4 -0.203450E-05
ROW	7	1	0.246620E-03	2	0.861713E 01	3	0.861736E 01 4 -0.203450E-05
ROW	8	1	0.246620E-03	2	0.861713E 01	3	0.861736E 01 4 -0.203450E-05
ROW	9	1	-0.697845E 01	2	-0.139942E 02	3	-0.209726E 02 4 -0.130098E 00
ROW	10	1	-0.697845E 01	2	-0.139942E 02	3	-0.209726E 02 4 -0.130098E 00

MATRIX FLOW

PAGE 2

CUTOFF = 0.0

SIZE	4	8Y	16
------	---	----	----

COND.	COND.	COND.
COND.	COND.	COND.

ROW	1	2	3	4
11	-0.697845E 01	-0.139942E 02	-0.209726E 02	-0.130098F 00

ROW	1	2	3	4
12	-0.697845E 01	-0.139942F 02	-0.209726E 02	-0.130098E 00

ROW	1	2	3	4
13	-0.697790E 01	0.139947E 02	0.701677E 01	-0.130098E 00

ROW	1	2	3	4
14	-0.697790E 01	0.139947E 02	0.701677F 01	-0.130098E 00

ROW	1	2	3	4
15	-0.697790E 01	0.139947E 02	0.701677E 01	-0.130098E 00

16	1	-0.697790E 01	2	0.139947E 02	3	0.701677E 01	4	-0.130098E 00
----	---	---------------	---	--------------	---	--------------	---	---------------

REFERENCES

1. "FORMAT II - Second Version of Fortran Matrix Abstraction Technique." AFFDL-TR-66-207, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio

J. Pickard, Volume I. "Engineering User Report." January 1967.

J.P. Cogan, Jr., Volume II. "Description of Digital Computer Program." March 1967.

R.C. Morris, Volume III. "A User-Coded Matrix Generator for the Force Method." March 1967.

J.E. Serpanos, Volume IV. "A User-Coded Matrix Generator for the Displacement Method." March 1967.

2. FORTRAN IV Language, IBM System/360, Form C28-6515-6.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) International Business Machines (IBM) Federal System Division		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP N/A	
3. REPORT TITLE FORMAT II-SECOND VERSION OF FORTRAN MATRIX ABSTRACTION TECHNIQUE VOLUME II SUPPLEMENT II. Description of Digital Computer Program System/360			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Technical Report			
5. AUTHOR(S) (First name, middle initial, last name) Hooks, C.G.			
6. REPORT DATE February 1969		7a. TOTAL NO. OF PAGES 74	7b. NO. OF REFS 2
8a. CONTRACT OR GRANT NO. NAS8-14000 b. PROJECT NO. 1467 c. Task No. 146705 d.		9a. ORIGINATOR'S REPORT NUMBER(S) 68-261-0001 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFFDL-TR-66-207 Volume II Supplement II	
10. DISTRIBUTION STATEMENT Distribution of this document is unlimited.			
11. SUPPLEMENTARY NOTES None		12. SPONSORING MILITARY ACTIVITY Air Force Flight Dynamics Laboratory Air Force Systems Command Wright-Patterson AFB, Ohio	
13. ABSTRACT A detailed description of the conversion effort for implementing FORMAT II on the System/360 using direct access I/O is presented. The conversion, coding, changes, and System/360 Job Control Language are presented in detail. A timing comparison with the FORMAT II 7094 timer is also presented. The data input of FORMAT II is unchanged except that SAVE statements are not usable at present. The output is unchanged except that some trace statements to show time of execution for each phase have been added.			

DD FORM 1473
1 NOV 65

UNCLASSIFIED

Security Classification

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
1. Structural analysis 2. Matrix methods 3. Matrix abstraction 4. Digital computer methods						

Security Classification