```
*       INDIANA UNIVERSITY FASTRAN (FAST FORTRAN II) COMPILER                FT000000
*                                                                            FT000010
```

```
        SPACE  5                                                             FT000020
*       F A S T R A N  (F A S T  F O R T R A N  I I)  C O M P I L E R         FT000030
```

```
      SPACE   5                                                          FT000040
*     AUTHORS...                                                         FT000050
*               STANLEY HAGSTROM                                         FT000060
*               FRANKLIN PROSSER                                         FT000070
*               STEPHEN YOUNG                                            FT000080
```

```
      SPACE   3                                                          FT000090
*     RESEARCH COMPUTING CENTER,  INDIANA UNIVERSITY                     FT000100
      REM                                                                FT000110
*     BLOOMINGTON, INDIANA                                               FT000120
      REM                                                                FT000130
*     1964                                                               FT000140
```

```
      SPACE   8                                                          FT000150
*     COPYRIGHT  INDIANA UNIVERSITY FOUNDATION  1964                     FT000160
```

```
      EJECT                                                          FT000170
      PCC       ON                                                   FT000180
      COUNT     15000                                                FT000190
      LBL       1FSTR                                                FT000200
      NOCRS                                                          FT000210
      TITLE                                                          FT000220
      PCC       OFF                                                  FT000230
```

```
      SPACE  9                                                    FT000270
*       *   *   *   *   *   *   *   *   *   *   *   *             FT000280
      REM                                                         FT000290
*              F A S T R A N  C O M P I L E R                     FT000300
      REM                                                         FT000310
*     R E S E A R C H  C O M P U T I N G  C E N T E R             FT000320
      REM                                                         FT000330
*           I N D I A N A  U N I V E R S I T Y                    FT000340
      REM                                                         FT000350
*              B L O O M I N G T O N ,  I N D I A N A            FT000360
```

```
      SPACE   3                                                          FT000370
*        *   *   *   *   *   *   *   *   *   *   *   *                  FT000380
      REM                                                               FT000390
*                G E N E R A L   C O M M E N T A R Y                    FT000400
      REM                                                               FT000410
*        *   *   *   *   *   *   *   *   *   *   *   *                  FT000420
```

```
      TTL I N T R O D U C T I O N -- GENERAL COMMENTARY                  FT000430
*     FASTRAN IS A FORTRAN II COMPILER WRITTEN FOR THE IBM              FT000440
*     709/7090/7094 SERIES OF COMPUTERS.  FASTRAN STANDS FOR FAST       FT000450
*     FORTRAN.  IT WAS DESIGNED AND WRITTEN AT INDIANA UNIVERSITY       FT000460
*     IN LATE 1963 AND EARLY 1964 BY STANLEY HAGSTROM, FRANKLIN         FT000470
*     PROSSER, AND STEPHEN YOUNG.  THE PROJECT WAS UNDERTAKEN FOR TWO   FT000480
*     REASONS.  FIRST, WE WERE DISSATISFIED WITH THE COMPILATION SPEED  FT000490
*     OF IBM'S FORTRAN II COMPILER.  SECOND, WE WISHED TO GAIN SOME     FT000500
*     EXPERIENCE IN COMPILER-WRITING TECHNIQUES.  THE ORIGINAL GOAL     FT000510
*     WAS TO PRODUCE A COMPILER WHICH WOULD COMPILE                     FT000520
*     PROGRAMS QUICKLY, PROBABLY AT THE EXPENSE OF OBJECT PROGRAM       FT000530
*     LENGTH AND EFFICIENCY.                                            FT000540
*     THE GOAL OF FAST COMPILATION WAS READILY ACHIEVED.  HOWEVER,      FT000550
*     TO OUR PLEASANT SURPRISE, THE RESULTING OBJECT PROGRAMS WERE      FT000560
*     IN GENERAL SHORTER THAN IBM'S, AND THE EXECUTION TIMES DID        FT000570
*     NOT COMPARE TOO UNFAVORABLY, EVEN IN THE EARLY VERSIONS.          FT000580
      REM                                                               FT000590
*     IN A SENSE, FASTRAN IS AN OUTGROWTH OF AN EARLIER EFFORT AT       FT000600
*     INDIANA UNIVERSITY BY DAVID KURN AND ONE OF THE AUTHORS (SH)      FT000610
*     TO PRODUCE A LOAD-AND-GO COMPILER (CALLED FLAT AND BASED ON A     FT000620
*     GENERALIZATION OF THE FORTRAN IV LANGUAGE) DURING THE SUMMER OF   FT000630
*     1962.  THIS PROJECT, ALTHOUGH EVENTUALLY ABANDONED, SHOWED        FT000640
*     CLEARLY HOW TO ACHIEVE HIGH COMPILE SPEEDS WITH REASONABLE        FT000650
*     OBJECT CODE EFFICIENCY.  MANY OF THE TECHNIQUES USED IN FLAT      FT000660
*     HAVE ALSO BEEN USED IN FASTRAN, INCLUDING THE METHOD OF SCANNING  FT000670
*     STATEMENTS AND PROCESSING ARITHMETIC EXPRESSIONS.  IN FACT,       FT000680
*     INITIALLY SOME SECTIONS OF THE FLAT CODE WERE TAKEN OVER BODILY   FT000690
*     TO FASTRAN.  IT IS DOUBTFUL IF WITHOUT THIS EARLIER EXPERIENCE    FT000700
*     ON FLAT WE COULD HAVE AVOIDED THE MANY PITFALLS AWAITING THE      FT000710
*     COMPILER WRITER AND HAVE SO EASILY MET OUR ORIGINAL GOALS.        FT000720
*     IT IS ESTIMATED THAT FROM START TO FINISH REQUIRED ABOUT NINE     FT000730
*     MAN-MONTHS OF EFFORT FOR THE FIRST VERSION OF FASTRAN, WITH       FT000740
*     MAYBE THREE MAN-MONTHS MORE REQUIRED FOR REFINEMENTS.             FT000750
      REM                                                               FT000760
*     FASTRAN IS A TWO-PASS IN-CORE COMPILER THAT USES A SINGLE SCRATCH  FT000770
*     TAPE IN THE COMPILATION OF LARGE PROGRAMS.  FASTRAN ACCEPTS THE   FT000780
*     COMPLETE FORTRAN II LANGUAGE, INCLUDING HOLLERITH LITERALS AND    FT000790
*     DOUBLE PRECISION AND COMPLEX ARITHMETIC.  ALSO, A FLOATING        FT000800
*     ROUNDING FEATURE IS IMPLEMENTED, AND PAGE SPACING, EJECTING, AND  FT000810
*     SUBTITLING IS PERMITTED.  FASTRAN MAY OPERATE IN A MONITOR        FT000820
*     SYSTEM SUCH AS FMS, OR IT MAY BE EXECUTED AS AN FMS JOB, WITH     FT000830
*     SOURCE PROGRAMS AS THE DATA.  (IN THIS LAST MODE, CLEARLY NO      FT000840
*     OBJECT PROGRAM EXECUTION IS POSSIBLE FOLLOWING COMPILATION.)      FT000850
      REM                                                               FT000860
*     OUR AIM HAS BEEN TO MAINTAIN COMPATIBILITY WITH IBM'S FMS-II      FT000870
*     VERSION OF FORTRAN II.  FASTRAN PRODUCES AN OBJECT PROGRAM        FT000880
*     BINARY DECK WHICH IS COMPLETELY ACCEPTABLE TO THE BSS LOADER.     FT000890
*     THE STANDARD FMS LIBRARY IS UTILIZED, AND FASTRAN'S CALLS TO      FT000900
*     THESE LIBRARY ROUTINES ARE IDENTICAL TO THOSE OF FORTRAN II.      FT000910
*     THE FASTRAN AND FORTRAN II BINARY DECKS ARE COMPLETELY INTER-     FT000920
*     CHANGABLE AS FAR AS SYSTEM FUNCTIONS ARE CONCERNED.  FORTRAN II   FT000930
*     SOURCE LANGUAGE DEBUGGING MAY BE USED WITH FASTRAN-COMPILED       FT000940
*     PROGRAMS.  TO DO THIS, A SYMBOL TABLE MUST BE REQUESTED, JUST     FT000950
*     AS IN IBM'S FORTRAN II.                                           FT000960
      REM                                                               FT000970
*     FASTRAN IS WRITTEN TO OPERATE USING THE INDIANA UNIVERSITY        FT000980
*     INPUT/OUTPUT SYSTEM, IOS.  THIS SYSTEM AUTOMATICALLY PROVIDES     FT000990
*     BLOCKING AND BUFFERING FOR ALL INPUT/OUTPUT.  A BRIEF DESCRIPTION FT001000
*     OF IOS, AND A DISCUSSION OF FASTRAN'S INPUT/OUTPUT REQUIREMENTS   FT001010
*     ARE PROVIDED IN LATER PAGES.  WITH THIS INFORMATION, THE TASK     FT001020
```

```
*       OF IMPLEMENTING FASTRAN WITH ANOTHER I/O SYSTEM IS GENERALLY      FT001030
*       NOT DIFFICULT.  NOTICE THAT WITHOUT BLOCKING AND BUFFERING OF     FT001040
*       TAPES, FASTRAN WILL BE TAPE-BOUND, AND THE COMPILE SPEEDS WILL    FT001050
*       SUFFER.                                                           FT001060
        REM                                                               FT001070
*       IF FASTRAN IS USED WITH AN ADEQUATE INPUT/OUTPUT SYSTEM (SUCH     FT001080
*       AS IOS), COMPILATION SPEEDS ARE OBTAINED THAT ARE 10 TO 100       FT001090
*       TIMES FASTER THAN THE IBM FORTRAN II UNDER FMS II.  AN 'AVERAGE'  FT001100
*       COMPILE TIME FACTOR IS ABOUT 13, BUT VERY SHORT PROGRAMS          FT001110
*       RESULT IN QUITE LARGE TIME FACTORS, AND IN GENERAL THE SHORTER    FT001120
*       THE PROGRAM, THE LARGER THE RATIO OF FORTRAN II VS FASTRAN        FT001130
*       COMPILE TIMES.  THIS TREND IS DUE MAINLY TO THE SUBSTANTIAL       FT001140
*       OVERHEAD TIME IN IBM'S COMPILER.  NEVERTHELESS, FOR LARGE         FT001150
*       PROGRAMS, THE TIME FACTOR APPEARS TO APPROACH 10, SO FASTRAN      FT001160
*       MAY BE CONSERVATIVELY SAID TO RUN AT A BASIC SPEED ABOUT TEN      FT001170
*       TIMES FASTER THAN THE IBM COMPILER.                               FT001180
        REM                                                               FT001190
*       OUR EXPERIENCE SHOWS THAT THE OBJECT CODE PRODUCED BY FASTRAN     FT001200
*       AVERAGES ABOUT TEN PERCENT SHORTER THAN THAT PRODUCED BY IBM'S    FT001210
*       FORTRAN II.  THIS FIGURE IS BASED ON A CONSIDERATION OF NUMBER    FT001220
*       OF INSTRUCTIONS, WORKING CELLS, AND CONSTANTS PRODUCED.           FT001230
        REM                                                               FT001240
*       THE EXECUTION TIME OF FASTRAN-PRODUCED OBJECT PROGRAMS IS         FT001250
*       GENERALLY SLIGHTLY LONGER THAN FOR IBM'S VERSIONS.  THIS IS       FT001260
*       ATTRIBUTABLE MAINLY TO THE LACK OF INDEXING OPTIMIZATION IN THIS  FT001270
*       VERSION OF FASTRAN.  THE MOST USEFUL FORMS OF INDEXING IMPROVE-   FT001280
*       MENT ARE FORTUNATELY RATHER EASY TO IMPLEMENT IN FASTRAN, AND     FT001290
*       THIS IS CURRENTLY BEING DONE.  ODDLY ENOUGH, WE ANTICIPATE THAT   FT001300
*       SINCE IMPROVED INDEXING WILL FURTHER SHORTEN THE OBJECT CODE,     FT001310
*       FASTRAN COMPILATIONS SHOULD SPEED UP A BIT...                     FT001320
        REM                                                               FT001330
*       FASTRAN IS AN IN-CORE COMPILER.  IT CAN REMAIN IN CORE THROUGH    FT001340
*       THE COMPILATION OF A NUMBER OF CONSECUTIVE FORTRAN PROGRAMS.      FT001350
*       THIS ALLOWS A VERY SMALL SYSTEM OVERHEAD TIME.  THERE ARE NO      FT001360
*       EXTRANEOUS PASSES OVER THE MATERIAL TO BE COMPILED.  THERE        FT001370
*       IS ONE PASS OVER THE SOURCE DECK, AND A SECOND AND FINAL PASS     FT001380
*       OVER THE INTERMEDIATE MATERIAL TO PRODUCE THE BINARY DECK.        FT001390
*       SHORT AND MEDIUM-LENGTH PROGRAMS ARE COMPILED IN CORE, AND        FT001400
*       NO SCRATCH TAPE IS USED (I.E. AN INTERNAL FILE IS MAINTAINED).    FT001410
*       ON LONGER PROGRAMS, A SINGLE INTERMEDIATE TAPE IS WRITTEN,        FT001420
*       REWOUND, AND THEN READ SEQUENTIALLY.  THERE IS NO DELAY DURING    FT001430
*       THE REWIND, SINCE THE INTERNAL FILE IS PROCESSED AT THIS TIME.    FT001440
*       THE COMPILATION DOES NOT PROCEED THROUGH A SYMBOLIC ASSEMBLY      FT001450
*       PHASE, THUS ELIMINATING ADDITIONAL TIME-CONSUMING OPERATIONS      FT001460
*       (THIS IS OF COURSE DONE AT THE QUESTIONABLE EXPENSE OF AN         FT001470
*       ASSEMBLY LISTING OPTION).                                         FT001480
*       IT IS PERHAPS APPROPRIATE AT THIS POINT TO STRESS THAT FASTRAN     FT001490
*       IS BASICALLY A ONE-PASS COMPILER, THE SECOND PASS BEING REQUIRED  FT001500
*       SOLELY BY THE BSS BINARY DECK FORMAT (I.E. PROGRAM CARD FIRST     FT001510
*       AND NO PROVISION FOR CHAINED BACKFILLS, ETC.).                    FT001520
        REM                                                               FT001530
*       ALL SOURCE PROGRAM SYMBOLS AND LABELS ARE PLACED IN A SYMBOL      FT001540
*       TABLE (SYMTAB) WHOSE SIZE IS 4096 WORDS.  THE METHOD OF           FT001550
*       ASSIGNING A SYMBOL OR LABEL TO A LOCATION IN SYMTAB IS SUCH THAT  FT001560
*       NO LENGTHY SEARCHES OR SORTS ARE NECESSARY.  A SCRAMBLED SORT     FT001570
*       TECHNIQUE IS USED, WHEREBY THE BITS IN A BCD SYMBOL OR LABEL      FT001580
*       ARE CONCENTRATED INTO A 12-BIT FIELD (2**12=4096).  THIS IS       FT001590
*       DONE BY MULTIPLYING THE BCD SYMBOLIC NAME BY A SIMPLE STANDARD    FT001600
*       INTEGER, AND EXTRACTING A TWELVE-BIT FIELD FROM THE RESULT.       FT001610
*       THE POINTER THUS OBTAINED IS NORMALLY THE SYMTAB LOCATION         FT001620
```

```
*       OF THE SYMBOL OR LABEL.  IF HOWEVER THE LOCATION IS ALREADY      FT001630
*       OCCUPIED BY A DIFFERENT SYMBOL (UNLIKELY IN MOST PROGRAMS), THEN FT001640
*       THE SYMBOL IS ASSIGNED TO THE NEXT AVAILABLE LOCATION.  TO       FT001650
*       LOCATE A SYMBOL IN SYMTAB, THE SAME COMPUTATION IS PERFORMED.    FT001660
*       IF THE SYMBOL IS NOT AT THE COMPUTED LOCATION, A SEQUENTIAL      FT001670
*       SEARCH IS PERFORMED UNTIL EITHER THE ITEM IS LOCATED, OR AN      FT001680
*       EMPTY CELL IS ENCOUNTERED.  (THE EMPTY CELL MEANS OF COURSE      FT001690
*       THAT THE SYMBOL IS NOT IN THE TABLE, AND IT MAY BE INSERTED      FT001700
*       IF DESIRED IN THE LOCATED EMPTY CELL).                           FT001710
*       THE TECHNIQUE IS VERY RAPID FOR SPARSELY-FILLED SYMBOL TABLES.   FT001720
*       IT BECOMES LESS EFFICIENT AS THE TABLE FILLS UP, SINCE MORE AND  FT001730
*       LONGER SEQUENTIAL SEARCHES WILL BE PERFORMED.  HOWEVER, A VERY   FT001740
*       LARGE PROGRAM IS REQUIRED TO FILL THE TABLE MORE THAN SPARSELY.  FT001750
        REM                                                              FT001760
*       EACH SYMTAB WORD HAS A COUNTERPART IN ANOTHER 4096-WORD TABLE,   FT001770
*       EQUIV.  EQUIV CONSISTS PRIMARILY OF FLAG BITS DESCRIBING THE     FT001780
*       CORRESPONDING SYMTAB ENTRY, AND AN ADDRESS POINTER WHICH MAY BE  FT001790
*       USED TO DEFINE THE VALUE OF THE ENTITY, OR TO POINT INTO A       FT001800
*       GENERAL POOL AREA WHERE MOST ADDITIONAL INFORMATION IS STORED.   FT001810
        REM                                                              FT001820
*       THE FORMAT OF THE EQUIV TABLE ENTRY IS DESCRIBED NOW IN GENERAL  FT001830
*       TERMS.  THE ACTUAL BIT CONFIGURATIONS FOR THE VARIOUS FLAGS AND  FT001840
*       FIELDS ARE FOUND IN THE REAR OF THE LISTING, UNDER 'ABSOLUTE     FT001850
*       SYMBOL DEFINITIONS'.                                             FT001860
        REM                                                              FT001870
*       THE PREFIX OF AN EQUIV WORD HOLDS THE MODE OF THE ITEM.  THE     FT001880
*       MODE IS REPRESENTED BY A THREE-BIT FIELD, AND NOT AS INDIVIDUAL  FT001890
*       BITS.  THE FOUR MODES ARE MLABL (LABEL ON EXECUTABLE STATEMENT), FT001900
*       MSTRG (FORMAT STRING LABEL), MINTG (INTEGER VARIABLE OR FUNCTION FT001910
*       NAME), AND MREAL (REAL VARIABLE OR FUNCTION NAME).               FT001920
        REM                                                              FT001930
*       BITS 3-14 HOLD BIT FLAGS DESCRIBING THE STATUS OF THE SYMTAB     FT001940
*       ITEM.  NORMALLY EACH BIT STANDS FOR A SINGLE ITEM OF INFORMATION.FT001950
*       HOWEVER, SOME BITS ARE USED FOR TWO OR MORE TASKS, IF NO CONFU-  FT001960
*       SION CAN RESULT.  THE BIT FLAGS AND THEIR INTERPRETATIONS ARE... FT001970
        REM                                                              FT001980
*       BVARB... SYMBOL HAS APPEARED AS A VARIABLE IN AN EXECUTABLE      FT001990
*         STATEMENT, AND THUS MUST BE DEFINED IN THE PROGRAM.            FT002000
*       BLHSX... SYMBOL OR LABEL HAS BEEN DEFINED (HAS A 'LEFT-HAND      FT002010
*         VALUE').                                                       FT002020
*       BDOUB... SYMBOL WAS USED AT SOME POINT AS A DOUBLE PRECISION     FT002030
*         OR A COMPLEX VARIABLE, AND THUS MUST HAVE DOUBLE STORAGE.      FT002040
*       BARGT... SYMBOL IS A FORMAL SUBPROGRAM PARAMETER.                FT002050
*       BARRY... SYMBOL APPEARS IN A DIMENSION STATEMENT.  ADDRESS FIELD FT002060
*         OF EQUIV WORD POINTS TO THE COMMON/DIMENSION/EQUIVALENCE       FT002070
*         INFORMATION IN THE POOL AREA.                                  FT002080
*       BCOMN... SYMBOL APPEARS IN A COMMON STATEMENT.  ADDRESS FIELD    FT002090
*         OF EQUIV WORD POINTS TO POOL AREA.                             FT002100
*       BEQUV... SYMBOL APPEARS IN ONE OR MORE EQUIVALENCE STATEMENTS.   FT002110
*         ADDRESS FIELD OF EQUIV WORD POINTS TO POOL AREA.               FT002120
*       BASSN... SYMBOL APPEARS AS THE VARIABLE IN AN ASSIGN OR          FT002130
*         ASSIGNED GO TO STATEMENT.  ADDRESS FIELD OF EQUIV WORD POINTS  FT002140
*         TO ASSIGN LISTS IN POOL AREA.                                  FT002150
*       BINTF... SYMBOL IS USED AS AN ARITHMETIC STATEMENT FUNCTION      FT002160
*         NAME.                                                          FT002170
*       BEXTF... SYMBOL IS USED AS AN EXTERNAL NAME.                     FT002180
*       BLIBF... EXTERNAL SYMBOL NAME IS CURRENTLY OF LIBRARY FUNCTION   FT002190
*         TYPE, I.E. ENDS IN 'F'.                                        FT002200
*       BPATH... LABEL IS IN THE PATH OF FLOW OF THE PROGRAM, I.E. THERE FT002210
*         IS A WAY FOR THE PROGRAM FLOW TO REACH THIS STATEMENT.         FT002220
```

```
      REM                                                                FT002230
*     BITS 15-20 HOLD THE SUBPROGRAM ARGUMENT NUMBER, IF BARGT IS ON.    FT002240
*     A MAXIMUM OF 64 FORMAL PARAMETERS CAN BE HANDLED.                  FT002250
      REM                                                                FT002260
*     BITS 21-35, THE ADDRESS FIELD, HOLD A SYMBOL OR LABEL DEFINITION,  FT002270
*     OR IF CERTAIN EQUIV FLAGS ARE ON, POINT TO FURTHER INFORMATION     FT002280
*     IN THE POOL AREA OR ELSEWHERE.                                     FT002290
      REM                                                                FT002300
*     ASIDE FROM SYMTAB AND EQUIV, THERE ARE VERY FEW FIXED-LENGTH       FT002310
*     TABLES IN FASTRAN.  THIS ALLOWS MAXIMUM FLEXIBILITY IN THE USE     FT002320
*     OF AVAILABLE STORAGE.  CONSEQUENTLY, THE LIKELIHOOD OF             FT002330
*     'OVERFLOW' OF THE COMPILER IS QUITE REMOTE.                        FT002340
      REM                                                                FT002350
*     A BRIEF SURVEY OF THE COMPILATION PROCESS IS GIVEN HERE.  MORE     FT002360
*     DETAILED INFORMATION IS TO BE FOUND IN DESCRIPTIONS OF THE         FT002370
*     INDIVIDUAL SECTIONS IN THE BODY OF THE LISTING.  THE COMPILATION   FT002380
*     IS LOGICALLY DIVIDED INTO FIVE SECTIONS... INITIALIZATION,         FT002390
*     PASS 1, POST PASS 1, PASS 2, AND POST PASS 2.                      FT002400
      REM                                                                FT002410
*     INITIALIZATION IS VERY SHORT, BEING CONCERNED WITH SETTING         FT002420
*     VARIOUS CELLS TO THEIR INITIAL VALUES.                             FT002430
      REM                                                                FT002440
*     PASS 1 IS THE PRIMARY PASS OVER THE SOURCE DECK.  THE SOURCE       FT002450
*     PROGRAM IS LISTED, THE STATEMENTS ARE SCANNED AND INTERPRETED,     FT002460
*     THE SYMBOL TABLE IS FORMED, VARIOUS LISTS ARE GENERATED, AND MANY  FT002470
*     TYPES OF ERRORS ARE DETECTED.  INTERMEDIATE OBJECT CODE IS         FT002480
*     PREPARED, TO BE PROCESSED IN PASS 2 WHEN THE SCAN OF THE SOURCE    FT002490
*     DECK IS COMPLETE.                                                  FT002500
      REM                                                                FT002510
*     POST PASS 1 PROCESSES THE SYMBOL TABLE, DETECTS ALL REMAINING      FT002520
*     ERRORS (ERRORS INVOLVING RELATIONS BETWEEN THE VARIOUS PROGRAM     FT002530
*     ITEMS), COMPUTES THE OBJECT PROGRAM LENGTH AND PUTS OUT THE        FT002540
*     BINARY DECK PROGRAM CARD, DEFINES ALL VARIABLES.  IF THIS PROGRAM  FT002550
*     IS A SUBROUTINE OR FUNCTION, ITS PROLOGUE IS FORMED HERE.          FT002560
*     IF AN OBJECT PROGRAM SYMBOL TABLE HAS BEEN REQUESTED, IT IS        FT002570
*     FORMED AND PUT OUT PRIOR TO THE PROGRAM CARD.                      FT002580
*     POST PASS 1 IS ENTERED ONLY IF PASS 1 WAS FREE FROM ERRORS.        FT002590
      REM                                                                FT002600
*     PASS 2 PROCESSES THE INTERMEDIATE OBJECT CODE, PRODUCING THE       FT002610
*     THE FINAL CODE AND THE REMAINDER OF THE BINARY DECK.               FT002620
      REM                                                                FT002630
*     POST PASS 2 PUTS OUT THE OBJECT PROGRAM STORAGE MAP, AND           FT002640
*     TERMINATES THE COMPILATION.  PASS 2 AND POST PASS 2 ARE            FT002650
*     ENTERED ONLY FOR ERROR-FREE COMPILATIONS.                          FT002660
```

```
      SPACE   2                                                          FT002670
*     A FEW WORDS ON FASTRAN'S ERROR DETECTION PHILOSOPHY ARE IN ORDER.FT002680
*     MOST ERRORS, INCLUDING TYPOGRAPHICAL AND SYNTACTICAL ERRORS, ARE FT002690
*     CAUGHT DURING PASS 1, THE PRIMARY PASS OVER THE SYMBOLIC DECK.   FT002700
*     WE FEEL THAT DIAGNOSTIC MESSAGES ARE MOST INFORMATIVE WHEN THEY  FT002710
*     ARE ISSUED ALONG WITH THE ERRONEOUS STATEMENT.  THEREFORE,       FT002720
*     MOST ERROR MESSAGES IN FASTRAN WILL APPEAR IN THE LISTING OF THE FT002730
*     SOURCE PROGRAM IMMEDIATELY FOLLOWING THE STATEMENT IN ERROR.     FT002740
*     THERE ARE SOME ERRORS, NAMELY INVOLVING INTER-STATEMENT          FT002750
*     RELATIONS, THAT CANNOT BE DETECTED UNTIL THE ENTIRE DECK HAS     FT002760
*     BEEN READ.  THESE THEN ARE GIVEN IMMEDIATELY FOLLOWING THE END   FT002770
*     CARD.  ALL ERROR MESSAGES ARE SIGNALLED BY AN OBVIOUS SERIES     FT002780
*     OF ASTERISKS AT THE LEFT SIDE OF THE PAGE.                       FT002790
      REM                                                              FT002800
*     THE DIAGNOSTIC MESSAGES ARE AS INFORMATIVE AS POSSIBLE WITHOUT   FT002810
*     UNDULY SACRIFICING COMPILATION SPEED, AND WITHOUT UNDULY CHANCINGFT002820
*     THE CONFUSION OF THE PROGRAMMER. IN RESERVED WORD (FIXED FORMAT) FT002830
*     STATEMENTS, ONE CAN BE RATHER SPECIFIC WITHOUT RISK OF CONFUSING FT002840
*     THE SUBMITTER OF THE PROGRAM.  HOWEVER, SPECIFIC DIAGNOSTICS ARE FT002850
*     IN GENERAL AVOIDED IN ARITHMETIC STATEMENTS, SINCE WE FEEL THAT  FT002860
*     TOO OFTEN A SPECIFIC MESSAGE DOES NOT IN FACT IDENTIFY THE       FT002870
*     PROGRAMMER'S DIFFICULTY, BUT INSTEAD CAUSES HIM MOMENTARY        FT002880
*     PUZZLEMENT.  THEREFORE, A COMMON ARITHMETIC STATEMENT ERROR      FT002890
*     MESSAGE IS SIMPLY 'INVALID FORM OF STATEMENT'.                   FT002900
      REM                                                              FT002910
*     IN PASSING WE NOTE THAT FASTRAN DETECTS MANY TYPES OF ERRORS THATFT002920
*     IBM'S FORTRAN II WILL NOT CATCH.  AMONG THESE ARE...             FT002930
*     CONSISTENCY ERRORS IN ASSIGN/ASSIGNED-GO-TO USAGE, UNDEFINED     FT002940
*     VARIABLES, RELATIVE CONSTANT ERRORS, ERRORS IN THE CONSISTENCY   FT002950
*     OF ARITHMETIC STATEMENT FUNCTION USAGES, AND NONSENSE SUCH AS    FT002960
*     '75    GO TO 75'.                                                FT002970
      REM                                                              FT002980
*     ALSO, A NUMBER OF VALID SITUATIONS INVOLVING FORTRAN II 'JOHNNY- FT002990
*     COME-LATELY' FEATURES, SUCH AS HOLLERITH LITERALS AND DOUBLE PRE-FT003000
*     CISION AND COMPLEX ARITHMETIC, ARE HANDLED CORRECTLY AND         FT003010
*     CONSISTENTLY IN FASTRAN, WHILE FORTRAN II COMPILES ERRONEOUSLY   FT003020
*     OR INCONSISTENTLY.  (DETAILS WILL BE SUPPLIED UPON REQUEST).     FT003030
```

```
*       FASTRAN COMPILER CORE STORAGE LAYOUT                            FT003060


        SPACE   3                                                       FT003070
*       THE LAYOUT OF CORE STORAGE IN THE FASTRAN COMPILER IS AS FOLLOWS.FT003080
*       THE CORE LOCATIONS OF THE VARIOUS SECTIONS ARE OF COURSE        FT003090
*       SUBJECT TO CHANGE.  MOST LOCATIONS ARE NOT IMPORTANT, AND ARE   FT003100
*       NOT SHOWN.                                                      FT003110
```

```
        SPACE   3                                                       FT003120
00000   REM     MONITOR CELLS, ETC                                      FT003130
00144   REM     INPUT/OUTPUT SYSTEM (INDIANA UNIVERSITY IOS)            FT003140
10000   REM     BEGINNING OF FASTRAN COMPILER                           FT003150
        REM     INITIALIZATION OF COMPILER                              FT003160
        REM         ONE-TIME INITIALIZATION WHEN COMPILER ENTERS CORE    FT003170
        REM         INITIALIZATION FOR EACH COMPILATION                 FT003180
        REM     INITIAL CARD PROCESSING                                 FT003190
        REM         BURSTING OF CARD                                    FT003200
        REM         STATEMENT LABEL AND MODE                            FT003210
        REM         DETERMINATION OF STATEMENT TYPE                     FT003220
        REM     STATEMENT TERMINATION ROUTINE                           FT003230
        REM     PASS 1 STATEMENT PROCESSORS                             FT003240
        REM     ARITHMETIC STATEMENT PROCESSING                         FT003250
        REM         COMPILER                                            FT003260
```

```
*       INDIANA UNIVERSITY FASTRAN (FAST FORTRAN II) COMPILER                FT000000
*                                                                            FT000010
        SPACE   5                                                            FT000020
*       F A S T R A N  (F A S T  F O R T R A N  I I)  C O M P I L E R        FT000030
        SPACE   5                                                            FT000040
*       AUTHORS...                                                           FT000050
*               STANLEY HAGSTROM                                             FT000060
*               FRANKLIN PROSSER                                             FT000070
*               STEPHEN YOUNG                                                FT000080
        SPACE   3                                                            FT000090
*       RESEARCH COMPUTING CENTER,  INDIANA UNIVERSITY                       FT000100
        REM                                                                  FT000110
*       BLOOMINGTON, INDIANA                                                 FT000120
        REM                                                                  FT000130
*       1964                                                                 FT000140
        SPACE   8                                                            FT000150
*       COPYRIGHT  INDIANA UNIVERSITY FOUNDATION  1964                       FT000160
```

```
EJECT                                                   FT000170
PCC     ON                                              FT000180
COUNT   15000                                           FT000190
LBL     1FSTR                                           FT000200
NOCRS                                                   FT000210
TITLE                                                   FT000220
PCC     OFF                                             FT000230
```

```
      TTL I N T R O D U C T I O N                                    FT000240
*                                                                    FT000250
      REM                                                            FT000260
      SPACE   9                                                      FT000270
*        *   *   *   *   *   *   *   *   *   *   *                   FT000280
      REM                                                            FT000290
*                F A S T R A N   C O M P I L E R                     FT000300
      REM                                                            FT000310
*     R E S E A R C H   C O M P U T I N G   C E N T E R              FT000320
      REM                                                            FT000330
*           I N D I A N A   U N I V E R S I T Y                      FT000340
      REM                                                            FT000350
*        B L O O M I N G T O N ,   I N D I A N A                     FT000360
      SPACE   3                                                      FT000370
*        *   *   *   *   *   *   *   *   *   *   *                   FT000380
      REM                                                            FT000390
*           G E N E R A L   C O M M E N T A R Y                      FT000400
      REM                                                            FT000410
*        *   *   *   *   *   *   *   *   *   *   *                   FT000420
```

```
      TTL I N T R O D U C T I O N -- GENERAL COMMENTARY                  FT000430
*     FASTRAN IS A FORTRAN II COMPILER WRITTEN FOR THE IBM              FT000440
*     709/7090/7094 SERIES OF COMPUTERS.  FASTRAN STANDS FOR FAST       FT000450
*     FORTRAN.  IT WAS DESIGNED AND WRITTEN AT INDIANA UNIVERSITY       FT000460
*     IN LATE 1963 AND EARLY 1964 BY STANLEY HAGSTROM, FRANKLIN         FT000470
*     PROSSER, AND STEPHEN YOUNG.  THE PROJECT WAS UNDERTAKEN FOR TWO   FT000480
*     REASONS.  FIRST, WE WERE DISSATISFIED WITH THE COMPILATION SPEED  FT000490
*     OF IBM'S FORTRAN II COMPILER.  SECOND, WE WISHED TO GAIN SOME     FT000500
*     EXPERIENCE IN COMPILER-WRITING TECHNIQUES.  THE ORIGINAL GOAL     FT000510
*     WAS TO PRODUCE A COMPILER WHICH WOULD COMPILE                     FT000520
*     PROGRAMS QUICKLY, PROBABLY AT THE EXPENSE OF OBJECT PROGRAM       FT000530
*     LENGTH AND EFFICIENCY.                                            FT000540
*     THE GOAL OF FAST COMPILATION WAS READILY ACHIEVED.  HOWEVER,      FT000550
*     TO OUR PLEASANT SURPRISE, THE RESULTING OBJECT PROGRAMS WERE      FT000560
*     IN GENERAL SHORTER THAN IBM'S, AND THE EXECUTION TIMES DID        FT000570
*     NOT COMPARE TOO UNFAVORABLY, EVEN IN THE EARLY VERSIONS.          FT000580
      REM                                                               FT000590
*     IN A SENSE, FASTRAN IS AN OUTGROWTH OF AN EARLIER EFFORT AT       FT000600
*     INDIANA UNIVERSITY BY DAVID KURN AND ONE OF THE AUTHORS (SH)      FT000610
*     TO PRODUCE A LOAD-AND-GO COMPILER (CALLED FLAT AND BASED ON A     FT000620
*     GENERALIZATION OF THE FORTRAN IV LANGUAGE) DURING THE SUMMER OF   FT000630
*     1962.  THIS PROJECT, ALTHOUGH EVENTUALLY ABANDONED, SHOWED        FT000640
*     CLEARLY HOW TO ACHIEVE HIGH COMPILE SPEEDS WITH REASONABLE        FT000650
*     OBJECT CODE EFFICIENCY.  MANY OF THE TECHNIQUES USED IN FLAT      FT000660
*     HAVE ALSO BEEN USED IN FASTRAN, INCLUDING THE METHOD OF SCANNING  FT000670
*     STATEMENTS AND PROCESSING ARITHMETIC EXPRESSIONS.  IN FACT,       FT000680
*     INITIALLY SOME SECTIONS OF THE FLAT CODE WERE TAKEN OVER BODILY   FT000690
*     TO FASTRAN.  IT IS DOUBTFUL IF WITHOUT THIS EARLIER EXPERIENCE    FT000700
*     ON FLAT WE COULD HAVE AVOIDED THE MANY PITFALLS AWAITING THE      FT000710
*     COMPILER WRITER AND HAVE SO EASILY MET OUR ORIGINAL GOALS.        FT000720
*     IT IS ESTIMATED THAT FROM START TO FINISH REQUIRED ABOUT NINE     FT000730
*     MAN-MONTHS OF EFFORT FOR THE FIRST VERSION OF FASTRAN, WITH       FT000740
*     MAYBE THREE MAN-MONTHS MORE REQUIRED FOR REFINEMENTS.             FT000750
      REM                                                               FT000760
*     FASTRAN IS A TWO-PASS IN-CORE COMPILER THAT USES A SINGLE SCRATCHFT000770
*     TAPE IN THE COMPILATION OF LARGE PROGRAMS.  FASTRAN ACCEPTS THE   FT000780
*     COMPLETE FORTRAN II LANGUAGE, INCLUDING HOLLERITH LITERALS AND    FT000790
*     DOUBLE PRECISION AND COMPLEX ARITHMETIC.  ALSO, A FLOATING        FT000800
*     ROUNDING FEATURE IS IMPLEMENTED, AND PAGE SPACING, EJECTING, AND  FT000810
*     SUBTITLING IS PERMITTED.  FASTRAN MAY OPERATE IN A MONITOR        FT000820
*     SYSTEM SUCH AS FMS, OR IT MAY BE EXECUTED AS AN FMS JOB, WITH     FT000830
*     SOURCE PROGRAMS AS THE DATA.  (IN THIS LAST MODE, CLEARLY NO      FT000840
*     OBJECT PROGRAM EXECUTION IS POSSIBLE FOLLOWING COMPILATION.)      FT000850
      REM                                                               FT000860
*     OUR AIM HAS BEEN TO MAINTAIN COMPATIBILITY WITH IBM'S FMS-II      FT000870
*     VERSION OF FORTRAN II.  FASTRAN PRODUCES AN OBJECT PROGRAM        FT000880
*     BINARY DECK WHICH IS COMPLETELY ACCEPTABLE TO THE BSS LOADER.     FT000890
*     THE STANDARD FMS LIBRARY IS UTILIZED, AND FASTRAN'S CALLS TO      FT000900
*     THESE LIBRARY ROUTINES ARE IDENTICAL TO THOSE OF FORTRAN II.      FT000910
*     THE FASTRAN AND FORTRAN II BINARY DECKS ARE COMPLETELY INTER-     FT000920
*     CHANGABLE AS FAR AS SYSTEM FUNCTIONS ARE CONCERNED.  FORTRAN II   FT000930
*     SOURCE LANGUAGE DEBUGGING MAY BE USED WITH FASTRAN-COMPILED       FT000940
*     PROGRAMS.  TO DO THIS, A SYMBOL TABLE MUST BE REQUESTED, JUST     FT000950
*     AS IN IBM'S FORTRAN II.                                           FT000960
      REM                                                               FT000970
*     FASTRAN IS WRITTEN TO OPERATE USING THE INDIANA UNIVERSITY        FT000980
*     INPUT/OUTPUT SYSTEM, IOS.  THIS SYSTEM AUTOMATICALLY PROVIDES     FT000990
*     BLOCKING AND BUFFERING FOR ALL INPUT/OUTPUT.  A BRIEF DESCRIPTIONFT001000
*     OF IOS, AND A DISCUSSION OF FASTRAN'S INPUT/OUTPUT REQUIREMENTS   FT001010
*     ARE PROVIDED IN LATER PAGES.  WITH THIS INFORMATION, THE TASK     FT001020
```

```
*       OF IMPLEMENTING FASTRAN WITH ANOTHER I/O SYSTEM IS GENERALLY      FT001030
*       NOT DIFFICULT.  NOTICE THAT WITHOUT BLOCKING AND BUFFERING OF     FT001040
*       TAPES, FASTRAN WILL BE TAPE-BOUND, AND THE COMPILE SPEEDS WILL    FT001050
*       SUFFER.                                                           FT001060
        REM                                                               FT001070
*       IF FASTRAN IS USED WITH AN ADEQUATE INPUT/OUTPUT SYSTEM (SUCH     FT001080
*       AS IOS), COMPILATION SPEEDS ARE OBTAINED THAT ARE 10 TO 100       FT001090
*       TIMES FASTER THAN THE IBM FORTRAN II UNDER FMS II.  AN 'AVERAGE'  FT001100
*       COMPILE TIME FACTOR IS ABOUT 13, BUT VERY SHORT PROGRAMS          FT001110
*       RESULT IN QUITE LARGE TIME FACTORS, AND IN GENERAL THE SHORTER    FT001120
*       THE PROGRAM, THE LARGER THE RATIO OF FORTRAN II VS FASTRAN        FT001130
*       COMPILE TIMES.  THIS TREND IS DUE MAINLY TO THE SUBSTANTIAL       FT001140
*       OVERHEAD TIME IN IBM'S COMPILER.  NEVERTHELESS, FOR LARGE         FT001150
*       PROGRAMS, THE TIME FACTOR APPEARS TO APPROACH 10, SO FASTRAN      FT001160
*       MAY BE CONSERVATIVELY SAID TO RUN AT A BASIC SPEED ABOUT TEN      FT001170
*       TIMES FASTER THAN THE IBM COMPILER.                               FT001180
        REM                                                               FT001190
*       OUR EXPERIENCE SHOWS THAT THE OBJECT CODE PRODUCED BY FASTRAN     FT001200
*       AVERAGES ABOUT TEN PERCENT SHORTER THAN THAT PRODUCED BY IBM'S    FT001210
*       FORTRAN II.  THIS FIGURE IS BASED ON A CONSIDERATION OF NUMBER    FT001220
*       OF INSTRUCTIONS, WORKING CELLS, AND CONSTANTS PRODUCED.           FT001230
        REM                                                               FT001240
*       THE EXECUTION TIME OF FASTRAN-PRODUCED OBJECT PROGRAMS IS         FT001250
*       GENERALLY SLIGHTLY LONGER THAN FOR IBM'S VERSIONS.  THIS IS       FT001260
*       ATTRIBUTABLE MAINLY TO THE LACK OF INDEXING OPTIMIZATION IN THIS  FT001270
*       VERSION OF FASTRAN.  THE MOST USEFUL FORMS OF INDEXING IMPROVE-   FT001280
*       MENT ARE FORTUNATELY RATHER EASY TO IMPLEMENT IN FASTRAN, AND     FT001290
*       THIS IS CURRENTLY BEING DONE.  ODDLY ENOUGH, WE ANTICIPATE THAT   FT001300
*       SINCE IMPROVED INDEXING WILL FURTHER SHORTEN THE OBJECT CODE,      FT001310
*       FASTRAN COMPILATIONS SHOULD SPEED UP A BIT...                     FT001320
        REM                                                               FT001330
*       FASTRAN IS AN IN-CORE COMPILER.  IT CAN REMAIN IN CORE THROUGH    FT001340
*       THE COMPILATION OF A NUMBER OF CONSECUTIVE FORTRAN PROGRAMS.      FT001350
*       THIS ALLOWS A VERY SMALL SYSTEM OVERHEAD TIME.  THERE ARE NO      FT001360
*       EXTRANEOUS PASSES OVER THE MATERIAL TO BE COMPILED.  THERE        FT001370
*       IS ONE PASS OVER THE SOURCE DECK, AND A SECOND AND FINAL PASS     FT001380
*       OVER THE INTERMEDIATE MATERIAL TO PRODUCE THE BINARY DECK.        FT001390
*       SHORT AND MEDIUM-LENGTH PROGRAMS ARE COMPILED IN CORE, AND        FT001400
*       NO SCRATCH TAPE IS USED (I.E. AN INTERNAL FILE IS MAINTAINED).    FT001410
*       ON LONGER PROGRAMS, A SINGLE INTERMEDIATE TAPE IS WRITTEN,        FT001420
*       REWOUND, AND THEN READ SEQUENTIALLY.  THERE IS NO DELAY DURING    FT001430
*       THE REWIND, SINCE THE INTERNAL FILE IS PROCESSED AT THIS TIME.    FT001440
*       THE COMPILATION DOES NOT PROCEED THROUGH A SYMBOLIC ASSEMBLY      FT001450
*       PHASE, THUS ELIMINATING ADDITIONAL TIME-CONSUMING OPERATIONS      FT001460
*       (THIS IS OF COURSE DONE AT THE QUESTIONABLE EXPENSE OF AN         FT001470
*       ASSEMBLY LISTING OPTION).                                         FT001480
*       IT IS PERHAPS APPROPRIATE AT THIS POINT TO STRESS THAT FASTRAN     FT001490
*       IS BASICALLY A ONE-PASS COMPILER, THE SECOND PASS BEING REQUIRED  FT001500
*       SOLELY BY THE BSS BINARY DECK FORMAT (I.E. PROGRAM CARD FIRST     FT001510
*       AND NO PROVISION FOR CHAINED BACKFILLS, ETC.).                    FT001520
        REM                                                               FT001530
*       ALL SOURCE PROGRAM SYMBOLS AND LABELS ARE PLACED IN A SYMBOL      FT001540
*       TABLE (SYMTAB) WHOSE SIZE IS 4096 WORDS.  THE METHOD OF           FT001550
*       ASSIGNING A SYMBOL OR LABEL TO A LOCATION IN SYMTAB IS SUCH THAT  FT001560
*       NO LENGTHY SEARCHES OR SORTS ARE NECESSARY.  A SCRAMBLED SORT     FT001570
*       TECHNIQUE IS USED, WHEREBY THE BITS IN A BCD SYMBOL OR LABEL      FT001580
*       ARE CONCENTRATED INTO A 12-BIT FIELD (2**12=4096).  THIS IS       FT001590
*       DONE BY MULTIPLYING THE BCD SYMBOLIC NAME BY A SIMPLE STANDARD    FT001600
*       INTEGER, AND EXTRACTING A TWELVE-BIT FIELD FROM THE RESULT.       FT001610
*       THE POINTER THUS OBTAINED IS NORMALLY THE SYMTAB LOCATION         FT001620
```

```
*       OF THE SYMBOL OR LABEL.  IF HOWEVER THE LOCATION IS ALREADY      FT001630
*       OCCUPIED BY A DIFFERENT SYMBOL (UNLIKELY IN MOST PROGRAMS), THEN FT001640
*       THE SYMBOL IS ASSIGNED TO THE NEXT AVAILABLE LOCATION.  TO       FT001650
*       LOCATE A SYMBOL IN SYMTAB, THE SAME COMPUTATION IS PERFORMED.    FT001660
*       IF THE SYMBOL IS NOT AT THE COMPUTED LOCATION, A SEQUENTIAL      FT001670
*       SEARCH IS PERFORMED UNTIL EITHER THE ITEM IS LOCATED, OR AN      FT001680
*       EMPTY CELL IS ENCOUNTERED.  (THE EMPTY CELL MEANS OF COURSE      FT001690
*       THAT THE SYMBOL IS NOT IN THE TABLE, AND IT MAY BE INSERTED      FT001700
*       IF DESIRED IN THE LOCATED EMPTY CELL).                           FT001710
*       THE TECHNIQUE IS VERY RAPID FOR SPARSELY-FILLED SYMBOL TABLES.   FT001720
*       IT BECOMES LESS EFFICIENT AS THE TABLE FILLS UP, SINCE MORE AND  FT001730
*       LONGER SEQUENTIAL SEARCHES WILL BE PERFORMED.  HOWEVER, A VERY   FT001740
*       LARGE PROGRAM IS REQUIRED TO FILL THE TABLE MORE THAN SPARSELY.  FT001750
        REM                                                              FT001760
*       EACH SYMTAB WORD HAS A COUNTERPART IN ANOTHER 4096-WORD TABLE,   FT001770
*       EQUIV.  EQUIV CONSISTS PRIMARILY OF FLAG BITS DESCRIBING THE     FT001780
*       CORRESPONDING SYMTAB ENTRY, AND AN ADDRESS POINTER WHICH MAY BE  FT001790
*       USED TO DEFINE THE VALUE OF THE ENTITY, OR TO POINT INTO A       FT001800
*       GENERAL POOL AREA WHERE MOST ADDITIONAL INFORMATION IS STORED.   FT001810
        REM                                                              FT001820
*       THE FORMAT OF THE EQUIV TABLE ENTRY IS DESCRIBED NOW IN GENERAL  FT001830
*       TERMS.  THE ACTUAL BIT CONFIGURATIONS FOR THE VARIOUS FLAGS AND  FT001840
*       FIELDS ARE FOUND IN THE REAR OF THE LISTING, UNDER 'ABSOLUTE     FT001850
*       SYMBOL DEFINITIONS'.                                             FT001860
        REM                                                              FT001870
*       THE PREFIX OF AN EQUIV WORD HOLDS THE MODE OF THE ITEM.  THE     FT001880
*       MODE IS REPRESENTED BY A THREE-BIT FIELD, AND NOT AS INDIVIDUAL  FT001890
*       BITS.  THE FOUR MODES ARE MLABL (LABEL ON EXECUTABLE STATEMENT), FT001900
*       MSTRG (FORMAT STRING LABEL), MINTG (INTEGER VARIABLE OR FUNCTION FT001910
*       NAME), AND MREAL (REAL VARIABLE OR FUNCTION NAME).               FT001920
        REM                                                              FT001930
*       BITS 3-14 HOLD BIT FLAGS DESCRIBING THE STATUS OF THE SYMTAB     FT001940
*       ITEM.  NORMALLY EACH BIT STANDS FOR A SINGLE ITEM OF INFORMATION.FT001950
*       HOWEVER, SOME BITS ARE USED FOR TWO OR MORE TASKS, IF NO CONFU-  FT001960
*       SION CAN RESULT.  THE BIT FLAGS AND THEIR INTERPRETATIONS ARE... FT001970
        REM                                                              FT001980
*       BVARB... SYMBOL HAS APPEARED AS A VARIABLE IN AN EXECUTABLE      FT001990
*         STATEMENT, AND THUS MUST BE DEFINED IN THE PROGRAM.            FT002000
*       BLHSX... SYMBOL OR LABEL HAS BEEN DEFINED (HAS A 'LEFT-HAND      FT002010
*         VALUE').                                                       FT002020
*       BDOUB... SYMBOL WAS USED AT SOME POINT AS A DOUBLE PRECISION     FT002030
*         OR A COMPLEX VARIABLE, AND THUS MUST HAVE DOUBLE STORAGE.      FT002040
*       BARGT... SYMBOL IS A FORMAL SUBPROGRAM PARAMETER.                FT002050
*       BARRY... SYMBOL APPEARS IN A DIMENSION STATEMENT.  ADDRESS FIELD FT002060
*         OF EQUIV WORD POINTS TO THE COMMON/DIMENSION/EQUIVALENCE       FT002070
*         INFORMATION IN THE POOL AREA.                                  FT002080
*       BCOMN... SYMBOL APPEARS IN A COMMON STATEMENT.  ADDRESS FIELD    FT002090
*         OF EQUIV WORD POINTS TO POOL AREA.                             FT002100
*       BEQUV... SYMBOL APPEARS IN ONE OR MORE EQUIVALENCE STATEMENTS.   FT002110
*         ADDRESS FIELD OF EQUIV WORD POINTS TO POOL AREA.               FT002120
*       BASSN... SYMBOL APPEARS AS THE VARIABLE IN AN ASSIGN OR          FT002130
*         ASSIGNED GO TO STATEMENT.  ADDRESS FIELD OF EQUIV WORD POINTS  FT002140
*         TO ASSIGN LISTS IN POOL AREA.                                  FT002150
*       BINTF... SYMBOL IS USED AS AN ARITHMETIC STATEMENT FUNCTION      FT002160
*         NAME.                                                          FT002170
*       BEXTF... SYMBOL IS USED AS AN EXTERNAL NAME.                     FT002180
*       BLIBF... EXTERNAL SYMBOL NAME IS CURRENTLY OF LIBRARY FUNCTION   FT002190
*         TYPE, I.E. ENDS IN 'F'.                                        FT002200
*       BPATH... LABEL IS IN THE PATH OF FLOW OF THE PROGRAM, I.E. THERE FT002210
*         IS A WAY FOR THE PROGRAM FLOW TO REACH THIS STATEMENT.         FT002220
```

```
      REM                                                                FT002230
*     BITS 15-20 HOLD THE SUBPROGRAM ARGUMENT NUMBER, IF BARGT IS ON.    FT002240
*     A MAXIMUM OF 64 FORMAL PARAMETERS CAN BE HANDLED.                  FT002250
      REM                                                                FT002260
*     BITS 21-35, THE ADDRESS FIELD, HOLD A SYMBOL OR LABEL DEFINITION,  FT002270
*     OR IF CERTAIN EQUIV FLAGS ARE ON, POINT TO FURTHER INFORMATION     FT002280
*     IN THE POOL AREA OR ELSEWHERE.                                     FT002290
      REM                                                                FT002300
*     ASIDE FROM SYMTAB AND EQUIV, THERE ARE VERY FEW FIXED-LENGTH       FT002310
*     TABLES IN FASTRAN.  THIS ALLOWS MAXIMUM FLEXIBILITY IN THE USE     FT002320
*     OF AVAILABLE STORAGE.  CONSEQUENTLY, THE LIKELIHOOD OF             FT002330
*     'OVERFLOW' OF THE COMPILER IS QUITE REMOTE.                        FT002340
      REM                                                                FT002350
*     A BRIEF SURVEY OF THE COMPILATION PROCESS IS GIVEN HERE.  MORE     FT002360
*     DETAILED INFORMATION IS TO BE FOUND IN DESCRIPTIONS OF THE         FT002370
*     INDIVIDUAL SECTIONS IN THE BODY OF THE LISTING.  THE COMPILATION   FT002380
*     IS LOGICALLY DIVIDED INTO FIVE SECTIONS... INITIALIZATION,         FT002390
*     PASS 1, POST PASS 1, PASS 2, AND POST PASS 2.                      FT002400
      REM                                                                FT002410
*     INITIALIZATION IS VERY SHORT, BEING CONCERNED WITH SETTING         FT002420
*     VARIOUS CELLS TO THEIR INITIAL VALUES.                             FT002430
      REM                                                                FT002440
*     PASS 1 IS THE PRIMARY PASS OVER THE SOURCE DECK.  THE SOURCE       FT002450
*     PROGRAM IS LISTED, THE STATEMENTS ARE SCANNED AND INTERPRETED,     FT002460
*     THE SYMBOL TABLE IS FORMED, VARIOUS LISTS ARE GENERATED, AND MANY  FT002470
*     TYPES OF ERRORS ARE DETECTED.  INTERMEDIATE OBJECT CODE IS         FT002480
*     PREPARED, TO BE PROCESSED IN PASS 2 WHEN THE SCAN OF THE SOURCE    FT002490
*     DECK IS COMPLETE.                                                  FT002500
      REM                                                                FT002510
*     POST PASS 1 PROCESSES THE SYMBOL TABLE, DETECTS ALL REMAINING      FT002520
*     ERRORS (ERRORS INVOLVING RELATIONS BETWEEN THE VARIOUS PROGRAM     FT002530
*     ITEMS), COMPUTES THE OBJECT PROGRAM LENGTH AND PUTS OUT THE        FT002540
*     BINARY DECK PROGRAM CARD, DEFINES ALL VARIABLES.  IF THIS PROGRAM  FT002550
*     IS A SUBROUTINE OR FUNCTION, ITS PROLOGUE IS FORMED HERE.          FT002560
*     IF AN OBJECT PROGRAM SYMBOL TABLE HAS BEEN REQUESTED, IT IS        FT002570
*     FORMED AND PUT OUT PRIOR TO THE PROGRAM CARD.                      FT002580
*     POST PASS 1 IS ENTERED ONLY IF PASS 1 WAS FREE FROM ERRORS.        FT002590
      REM                                                                FT002600
*     PASS 2 PROCESSES THE INTERMEDIATE OBJECT CODE, PRODUCING THE       FT002610
*     THE FINAL CODE AND THE REMAINDER OF THE BINARY DECK.               FT002620
      REM                                                                FT002630
*     POST PASS 2 PUTS OUT THE OBJECT PROGRAM STORAGE MAP, AND           FT002640
*     TERMINATES THE COMPILATION.  PASS 2 AND POST PASS 2 ARE            FT002650
*     ENTERED ONLY FOR ERROR-FREE COMPILATIONS.                          FT002660
      SPACE  2                                                           FT002670
*     A FEW WORDS ON FASTRAN'S ERROR DETECTION PHILOSOPHY ARE IN ORDER.  FT002680
*     MOST ERRORS, INCLUDING TYPOGRAPHICAL AND SYNTACTICAL ERRORS, ARE   FT002690
*     CAUGHT DURING PASS 1, THE PRIMARY PASS OVER THE SYMBOLIC DECK.     FT002700
*     WE FEEL THAT DIAGNOSTIC MESSAGES ARE MOST INFORMATIVE WHEN THEY    FT002710
*     ARE ISSUED ALONG WITH THE ERRONEOUS STATEMENT.  THEREFORE,         FT002720
*     MOST ERROR MESSAGES IN FASTRAN WILL APPEAR IN THE LISTING OF THE   FT002730
*     SOURCE PROGRAM IMMEDIATELY FOLLOWING THE STATEMENT IN ERROR.       FT002740
*     THERE ARE SOME ERRORS, NAMELY INVOLVING INTER-STATEMENT            FT002750
*     RELATIONS, THAT CANNOT BE DETECTED UNTIL THE ENTIRE DECK HAS       FT002760
*     BEEN READ.  THESE THEN ARE GIVEN IMMEDIATELY FOLLOWING THE END     FT002770
*     CARD.  ALL ERROR MESSAGES ARE SIGNALLED BY AN OBVIOUS SERIES       FT002780
*     OF ASTERISKS AT THE LEFT SIDE OF THE PAGE.                         FT002790
      REM                                                                FT002800
*     THE DIAGNOSTIC MESSAGES ARE AS INFORMATIVE AS POSSIBLE WITHOUT     FT002810
*     UNDULY SACRIFICING COMPILATION SPEED, AND WITHOUT UNDULY CHANCING  FT002820
```

```
*       THE CONFUSION OF THE PROGRAMMER. IN RESERVED WORD (FIXED FORMAT) FT002830
*       STATEMENTS, ONE CAN BE RATHER SPECIFIC WITHOUT RISK OF CONFUSING FT002840
*       THE SUBMITTER OF THE PROGRAM.  HOWEVER, SPECIFIC DIAGNOSTICS ARE FT002850
*       IN GENERAL AVOIDED IN ARITHMETIC STATEMENTS, SINCE WE FEEL THAT  FT002860
*       TOO OFTEN A SPECIFIC MESSAGE DOES NOT IN FACT IDENTIFY THE       FT002870
*       PROGRAMMER'S DIFFICULTY, BUT INSTEAD CAUSES HIM MOMENTARY        FT002880
*       PUZZLEMENT.  THEREFORE, A COMMON ARITHMETIC STATEMENT ERROR      FT002890
*       MESSAGE IS SIMPLY 'INVALID FORM OF STATEMENT'.                   FT002900
        REM                                                              FT002910
*       IN PASSING WE NOTE THAT FASTRAN DETECTS MANY TYPES OF ERRORS THATFT002920
*       IBM'S FORTRAN II WILL NOT CATCH.  AMONG THESE ARE...             FT002930
*       CONSISTENCY ERRORS IN ASSIGN/ASSIGNED-GO-TO USAGE, UNDEFINED     FT002940
*       VARIABLES, RELATIVE CONSTANT ERRORS, ERRORS IN THE CONSISTENCY   FT002950
*       OF ARITHMETIC STATEMENT FUNCTION USAGES, AND NONSENSE SUCH AS    FT002960
*       '75   GO TO 75'.                                                 FT002970
        REM                                                              FT002980
*       ALSO, A NUMBER OF VALID SITUATIONS INVOLVING FORTRAN II 'JOHNNY- FT002990
*       COME-LATELY' FEATURES, SUCH AS HOLLERITH LITERALS AND DOUBLE PRE-FT003000
*       CISION AND COMPLEX ARITHMETIC, ARE HANDLED CORRECTLY AND         FT003010
*       CONSISTENTLY IN FASTRAN, WHILE FORTRAN II COMPILES ERRONEOUSLY   FT003020
*       OR INCONSISTENTLY.  (DETAILS WILL BE SUPPLIED UPON REQUEST).     FT003030
        SPACE   2                                                        FT003040
```

```
*       FASTRAN COMPILER CORE STORAGE LAYOUT                                FT003060
        SPACE   3                                                           FT003070
*       THE LAYOUT OF CORE STORAGE IN THE FASTRAN COMPILER IS AS FOLLOWS.   FT003080
*       THE CORE LOCATIONS OF THE VARIOUS SECTIONS ARE OF COURSE            FT003090
*       SUBJECT TO CHANGE.  MOST LOCATIONS ARE NOT IMPORTANT, AND ARE       FT003100
*       NOT SHOWN.                                                          FT003110
        SPACE   3                                                           FT003120
00000   REM     MONITOR CELLS, ETC                                          FT003130
00144   REM     INPUT/OUTPUT SYSTEM (INDIANA UNIVERSITY IOS)                FT003140
10000   REM     BEGINNING OF FASTRAN COMPILER                               FT003150
        REM     INITIALIZATION OF COMPILER                                  FT003160
        REM         ONE-TIME INITIALIZATION WHEN COMPILER ENTERS CORE        FT003170
        REM         INITIALIZATION FOR EACH COMPILATION                     FT003180
        REM     INITIAL CARD PROCESSING                                     FT003190
        REM         BURSTING OF CARD                                        FT003200
        REM         STATEMENT LABEL AND MODE                                FT003210
        REM         DETERMINATION OF STATEMENT TYPE                         FT003220
        REM     STATEMENT TERMINATION ROUTINE                               FT003230
        REM     PASS 1 STATEMENT PROCESSORS                                 FT003240
        REM     ARITHMETIC STATEMENT PROCESSING                             FT003250
        REM         COMPILER                                                FT003260
        REM         SCANNER                                                 FT003270
        REM         SUBSCRIPT SCANNER AND COMPILER                          FT003280
        REM         CODE KEY-WORD GENERATOR                                 FT003290
        REM         OBJECT PROGRAM CODE GENERATORS                          FT003300
        REM         ARITHMETIC STATEMENT UTILITY ROUTINES                   FT003310
        REM     PASS 1 UTILITY ROUTINES                                     FT003320
        REM     POST PASS 1                                                 FT003330
        REM         ASSIGN VARIABLE AND LABEL CHECKING                      FT003340
        REM         EQUIVALENCE VARIABLE STORAGE ASSIGNMENT                 FT003350
        REM         LENGTH OF PROLOGUE                                      FT003360
        REM         PRIMARY SYMBOL TABLE PASS                               FT003370
        REM         COMMON VARIABLE STORAGE ASSIGNMENT                      FT003380
        REM         LENGTH OF OBJECT PROGRAM                                FT003390
        REM         OUTPUT OF ERRORS FOUND IN SYMBOL TABLE PASS             FT003400
        REM         OBJECT PROGRAM SYMBOL TABLE OUTPUT, IF REQUESTED        FT003410
        REM         BINARY DECK LABEL ROUTINE                               FT003420
        REM         PROGRAM CARD OUTPUT                                     FT003430
        REM         TRANSFER VECTOR AND PROLOGUE OUTPUT ROUTINES            FT003440
        REM     POST PASS 1 UTILITY ROUTINES                                FT003450
        REM     PASS 2...  OUTPUT REMAINDER OF BINARY DECK                  FT003460
        REM         TEXT                                                    FT003470
        REM         CONSTANTS                                               FT003480
        REM         FORMAT STATEMENTS                                       FT003490
        REM     POST PASS 2... OBJECT PROGRAM MAP                           FT003500
        REM     COMPILER TERMINATION ROUTINE                                FT003510
        REM     UTILITY ROUTINES                                            FT003520
        REM     ERROR ROUTINE AND DIAGNOSTIC MESSAGES                       FT003530
        REM     COMPILER TABLES                                             FT003540
        REM     FLAGS AND WORKING CELLS                                     FT003550
        REM     (END OF FASTRAN SYMBOLIC DECK)                              FT003560
        REM     LITERALS                                                    FT003570
        REM     ARRAY 'COLUMN'  (666 CELLS LONG)                            FT003580
57775   REM     POOL (ABOUT 5200 CELLS LONG, BACKWARD TO END OF COLUMN)     FT003590
67776   REM     EQUIV  (4096 CELLS LONG, BACKWARD IN CORE)                  FT003600
77777   REM     SYMTAB (4096 CELLS LONG, BACKWARD IN CORE)                  FT003610
```

```
      TTL I N T R O D U C T I O N -- FASTRAN INPUT/OUTPUT REQUIREMENTS FT003630
*           REMARKS CONCERNING FASTRAN INPUT/OUTPUT REQUIREMENTS   FT003640
      SPACE  3                                                     FT003650
*     THE FASTRAN COMPILER UTILIZES A GENERALIZED INPUT/OUTPUT SYSTEM FT003660
*     DEVELOPED AT INDIANA UNIVERSITY ... IOS.  THIS I/O SYSTEM    FT003670
*     RESIDES IN LOW-CORE AND CONSISTS OF A SERIES OF ROUTINES FOR THE FT003680
*     BUFFERED READING AND WRITING OF BLOCKED TAPE RECORDS, ALONG WITH FT003690
*     NORMAL NON-DATA OPERATIONS (WEF, REW, ETC.).  IN ADDITION, IOS FT003700
*     HANDLES CERTAIN SPECIAL FUNCTIONS SUCH AS ON-LINE PRINTING,  FT003710
*     READING OF 716 CLOCK AND LOADING OF SYSTEM TAPE RECORDS.  THE FT003720
*     DATA CHANNEL TRAP FEATURE IS USED BY IOS TO CONTROL CHANNEL  FT003730
*     ACTIVITY.                                                    FT003740
      SPACE  2                                                     FT003750
*     CALLS TO IOS ARE EFFECTED BY I/O MACROS (EG.- BUFFER, READ,  FT003760
*     WRITE, BACKR, WEOF, ETC.).  THESE MACROS ARE BUILT INTO      FT003770
*     THE MACHINE LANGUAGE ASSEMBLER EMPLOYED AT INDIANA UNIVERSITY FT003780
*     (A SPECIALIZED VERSION OF IBM'S FAP ASSEMBLER).              FT003790
      REM                                                          FT003800
*     IN ORDER THAT ANOTHER INSTALLATION CAN ASSEMBLE FASTRAN, THIS FT003810
*     VERSION OF FASTRAN HAS BEEN MODIFIED.  ALL IOS MACRO CALLS   FT003820
*     HAVE BEEN EXPANDED IN-LINE, AND THE MACRO NAMES NULLIFIED    FT003830
*     (E.G. BY 'READ OPSYN NULL').  THE PERTINENT TRANSFER ADDRESSES FT003840
*     TO THE IOS ROUTINES ARE DECLARED IN BOOL STATEMENTS (E.G.    FT003850
*     'READ BOOL 242').  THIS MATERIAL IS LOCATED AFTER THE MACRO  FT003860
*     DEFINITION SECTION OF THE COMPILER.  ALL CALLS TO IOS ROUTINES FT003870
*     ARE BRACKETED IN THE LISTING BY CARDS HAVING 'IOSXX' AND     FT003880
*     'ENDXX' IN THE LOCATION FIELDS (XX IS A NUMBER FROM 01 TO 99). FT003890
*     THESE SEQUENCES MAY EASILY BE FOUND BY LOCATING THE IOSXX    FT003900
*     ENTRIES IN THE SYMBOLIC REFERENCE TABLE.                     FT003910
```

```
      EJECT                                                              FT003920
*     THE FOLLOWING IS A BRIEF SUMMARY OF THE EXPANSION OF I/O MACROS,   FT003930
      REM                                                                FT003940
*        ALL I/O MACROS ARE OF THE GENERAL FORM                          FT003950
      REM                                                                FT003960
*           SYMBOL NAME    A(1),A(2), ... A(N-1),A(N)                   FT003970
      SPACE  2                                                           FT003980
*        THE MACHINE LANGUAGE EXPANSION OF THIS MACRO IS,                FT003990
      REM                                                                FT004000
*           SYMBOL STL     IOEX                                         FT004010
*                  TXL     NAME,,(NORMAL RETURN)                        FT004020
*                  XYZ     A(1),,A(2)                                   FT004030
*                     .                                                 FT004040
*                     .                                                 FT004050
*                     .                                                 FT004060
*                  XYZ     A(N),,0 OR A(N-1),,A(N)                      FT004070
      REM                                                                FT004080
*        WHERE,                                                          FT004090
*           X = 0 IFF NOT LAST ARGUMENT PAIR                            FT004100
*           Y = 0 IFF ADDRESS ARGUMENT NOT INDIRECTLY ADDRESSED         FT004110
*           Z = 0 IFF DECREMENT ARGUMENT NOT INDIRECTLY ADDRESSED       FT004120
*           (NORMAL RETURN) = LOCATION + 1 OF LAST WORD OF MACRO        FT004130
*                             EXPANSION                                 FT004140
      SPACE  2                                                           FT004150
*        UNNECESSARY ARGUMENTS INCLUDED IN AN IOS MACRO ARE IGNORED      FT004160
*        ... WILL NOT CAUSE ASSEMBLY OR EXECUTION ERRORS.                FT004170
      REM                                                                FT004180
*        FIRST BLANK IN ARGUMENT STRING TERMINATES THE VARIABLE FIELD.   FT004190
      REM                                                                FT004200
*        AN ASTERISK (*) AS THE FIRST CHARACTER OF AN ARGUMENT           FT004210
*        SPECIFIES INDIRECT ADDRESSING (ALL INDEX REGISTERS ASSUMED TO   FT004220
*        CONTAIN ZERO).                                                  FT004230
      REM                                                                FT004240
*        ETC CARD(S) FOR CONTAINING LONG MACRO ARGUMENT LISTS (CARD      FT004250
*        PRECEDING ETC CARD MUST HAVE COMMA AS LAST NON-BLANK CHAR.).    FT004260
```

```
        EJECT                                                                   FT004270
*       THE FOLLOWING IS A BRIEF DESCRIPTION OF THE VARIOUS IOS MACROS          FT004280
*       USED BY FASTRAN,                                                        FT004290
        SPACE   2                                                               FT004300
*          BUFFER   LOC(1),NO(1),SIZE(1),TAPE(1),...                            FT004310
*             TO ASSIGN BUFFER SPACE TO VARIOUS TAPES.  BUFFERS MUST BE         FT004320
*             AT LEAST TWO WORDS LONGER THAN LARGEST PHYSICAL RECORD TO         FT004330
*             BE READ (OR WRITTEN) WITH THESE BUFFERS.  BUFFERS CAN             FT004340
*             BE RE-DEFINED AT ANY TIME.  TWO BUFFERS PER TAPE IS               FT004350
*             GENERALLY OPTIMUM.                                                FT004360
        REM                                                                     FT004370
*          XMODER   LOC                                                         FT004380
*             CONTROL TO LOC IF READ GIVEN IN WRONG MODE.  NO DATA              FT004390
*             TRANSMISSION OCCURS.                                              FT004400
        REM                                                                     FT004410
*          XRDN     LOC                                                         FT004420
*             CONTROL TO LOC IF READ OR WRITE REDUNDANCY.                       FT004430
        REM                                                                     FT004440
*          XEOF     LOC                                                         FT004450
*             CONTROL TO LOC IF END-OF-FILE READ.                               FT004460
        REM                                                                     FT004470
*          READ     TAPE,MODE,LOC(1),CNT(1),...LOC(N),CNT(N)                    FT004480
*             TO SCATTER READ ONE LOGICAL RECORD FROM TAPE.  THIS IS A          FT004490
*             TRANSMIT TYPE READ IN THAT DATA IS MOVED FROM THE BUFFER          FT004500
*             TO THE LOCATIONS SPECIFIED IN THE READ MACRO.                     FT004510
        REM                                                                     FT004520
*          LREAD    TAPE,MODE,L(LOC),L(COUNT)                                   FT004530
*             SET DECREMENT OF LOC TO LOCATION OF FIRST WORD OF THE             FT004540
*             NEXT LOGICAL RECORD.  THIS IS A NON-TRANSMIT TYPE READ IN         FT004550
*             THAT NO DATA IS MOVED FROM THE BUFFER.  DECREMENT OF              FT004560
*             LOCATION COUNT WILL CONTAIN NUMBER OF WORDS IN THE RECORD         FT004570
*             JUST LOCATED.                                                     FT004580
        REM                                                                     FT004590
*          WRITE    TAPE,MODE,LOC(1),CNT(1),...LOC(N),CNT(N)                    FT004600
*             TO GATHER WRITE ONE LOGICAL RECORD ONTO TAPE.  TRANSMIT           FT004610
*             TYPE WRITE.                                                       FT004620
        REM                                                                     FT004630
*          BACKR    TAPE,COUNT                                                  FT004640
*             TO BACKSPACE COUNT LOGICAL RECORDS ON TAPE.                       FT004650
        REM                                                                     FT004660
*          WEOF     TAPE                                                        FT004670
*             TO WRITE AN END-OF-FILE ON TAPE (BUFFER(S) FLUSHED                FT004680
*             AUTOMATICALLY).                                                   FT004690
        REM                                                                     FT004700
*          REWIND   TAPE                                                        FT004710
*             TO REWIND TAPE.                                                   FT004720
        REM                                                                     FT004730
*          PRINT    LOC(1),CNT(1),...LOC(N),CNT(N)                              FT004740
*             EACH PAIR OF PARAMETERS REPRESENTS A LINE TO BE PRINTED           FT004750
*             ON THE ON-LINE PRINTER (LOCATION AND COUNT).  FIRST               FT004760
*             CHARACTER USED AS SPACE CONTROL (A LA FORTRAN II).                FT004770
        REM                                                                     FT004780
*          RCLOCK   LOC                                                         FT004790
*             TO READ 716 PRINTER CLOCK INTO STORAGE, CONVERT TIME TO           FT004800
*             SIX BCD CHARACTERS AND STORE THIS IN LOC AND LOGICAL              FT004810
*             ACCUMULATOR (LOC IS AN OPTIONAL ARGUMENT).  SENSE EXIT 10         FT004820
*             AND ECHO IMPULSES AT THE 716 ARE USED TO OBTAIN THIS CLOCK        FT004830
*             READING.                                                          FT004840
```

```
      EJECT                                                                  FT004850
      SPACE  2                                                               FT004860
*     THE IOBASE PSEUDO-OPERATION APPEARING RIGHT AFTER THE MACRO            FT004870
*     DEFINITIONS IN THE ASSEMBLY LISTING OF FASTRAN IS REQUIRED BY          FT004880
*     THE ASSEMBLER TO PIN-POINT THE LOCATION OF A CRITICAL TRANSFER         FT004890
*     VECTOR LOCATED IN IOS.                                                 FT004900
      SPACE  2                                                               FT004910
*     FASTRAN REQUIRES FOUR TAPES DURING NORMAL OPERATION AND BUFFER         FT004920
*     SPACE FOR THESE TAPES IS DEFINED WITHIN FASTRAN.  ALL TAPES HAVE       FT004930
*     LOGICAL TAPE NUMBERS AND IOS IS RESPONSIBLE FOR ESTABLISHING           FT004940
*     THE LOGICAL/PHYSICAL TAPE EQUIVALENCES.  THE FOLLOWING TABLE           FT004950
*     SUMMARIZES THE TAPES USED BY FASTRAN,                                  FT004960
      SPACE  2                                                               FT004970
*        SYMBOLIC  LOGICAL  PHYSICAL   SYMBOLIC                              FT004980
*         NAME     NUMBER   NUMBER    BUFFER NAME   FUNCTION                 FT004990
      REM                                                                    FT005000
*        MINTAP     5        A2        BUFMIT        SYSTEM INPUT TAPE       FT005010
      REM                                                                    FT005020
*        MLSTAP     6        A3        BUFMLT        SYSTEM (BCD)            FT005030
*                                                     OUTPUT TAPE             FT005040
      REM                                                                    FT005050
*        SYSBIN     7        B4        BUFBIN        SYSTEM (BINARY)         FT005060
*                                                     LOAD TAPE               FT005070
*                                                     (CONTAINS OBJECT        FT005080
*                                                     DECK AT END OF          FT005090
*                                                     COMPILATION)            FT005100
      REM                                                                    FT005110
*        SYSUT1     2        B2        BUFUT1        SYSTEM UTILITY          FT005120
*                                                     TAPE (CONTAINS          FT005130
*                                                     C I T OVERFLOW          FT005140
*                                                     FROM PASS 1)            FT005150
      SPACE  2                                                               FT005160
*     NOTE-                                                                  FT005170
*     IF NOT RUN UNDER THE MONITOR, SYSBIN = MLSTAP = LOGICAL TAPE           FT005180
*     NUMBER 9 = PHYSICAL TAPE A5.  THAT IS, THE BCD LIST OUTPUT AND         FT005190
*     BINARY OBJECT DECK ARE PLACED ON TAPE 9 (= A5).                        FT005200
```

```
      EJECT                                                              FT005210
*     IF AN INSTALLATION IS INTERESTED IN USING A SYSTEM OTHER THAN      FT005220
*     IOS (EG. IOCS) TO PERFORM THE I/O REQUIREMENTS OF THE COMPILER     FT005230
*     THE FOLLOWING STEPS COULD BE TAKEN,                                FT005240
      REM                                                                FT005250
*        1. LOCATE ALL OF THE IOS MACROS IN FASTRAN.  THIS CAN BE        FT005260
*           DONE BY USING THE SYMBOLIC REFERENCE TABLE ... ALL IOS       FT005270
*           MACROS ARE NAMED IOSXX (XX = 01, 02, ...).                   FT005280
      REM                                                                FT005290
*        2. DELETE THOSE MACROS WHICH DO NOT APPLY TO THE NEW SYSTEM     FT005300
*           ... FOR EXAMPLE, THE XEOF FUNCTION MAY BE HANDLED BY EACH    FT005310
*           CALL TO THE READ ROUTINE IN THE NEW SYSTEM.                  FT005320
      REM                                                                FT005330
*        3. REPLACE ALL OTHER MACROS WITH CALLS TO THE NEW SYSTEM TO      FT005340
*           PERFORM A GIVEN FUNCTION ...                                 FT005350
      REM                                                                FT005360
*        NOTE-                                                          FT005370
*        THE FIRST CARD OF EACH FASTRAN COMPILATION AND THE FIRST CARD  FT005380
*        OF EACH STATEMENT THAT FOLLOWS A COMMENTS CARD, AN ASTERISK    FT005390
*        (*) CARD OR A SUBTITLING CARD IS READ WITH A NORMAL (TRANSMIT-FT005400
*        TYPE) READ.  ALL OTHER SOURCE CARDS ARE READ WITH A LOCATE     FT005410
*        READ MACRO.  THIS, OF COURSE, IS DESIGNED TO ELIMINATE THE     FT005420
*        BUFFER-TO-WORK AREA MOVE REQUIRED BY THE TRANSMIT-TYPE READ.   FT005430
      SPACE  3                                                           FT005440
*     A COPY OF AN IOS USER'S MANUAL WILL BE SENT UPON REQUEST.          FT005450
```

```
*       *       *       *       *       *       *       *       *FT005480
*       FASTRAN/FORTRAN II INCOMPATIBILITIES.                    FT005490
*       *       *       *       *       *       *       *       *FT005500
        SPACE   4                                                FT005510
*       THERE ARE SOME FEATURES OF FASTRAN WHICH ARE INCOMPATIBLE WITH   FT005520
*       (OR DIFFERENT FROM) THE IBM FORTRAN II.  THESE ARE LISTED        FT005530
*       AND DISCUSSED BELOW.  MOST ARE RATHER INCONSEQUENTIAL.           FT005540
*       HOWEVER, SOME ARE OF CONSIDERABLE SIGNIFICANCE, AND THE          FT005550
*       INCOMPATIBILITY WAS INTENTIONALLY CREATED.  THE LIST BELOW       FT005560
*       IS ROUGHLY IN THE ORDER OF DECREASING IMPORTANCE.                FT005570
        REM                                                      FT005580
        REM                                                      FT005590
        REM                                                      FT005600
****    THERE ARE NO RELATIVE CONSTANTS AT ALL IN FASTRAN.  ALL VARIABLESFT005610
*       HAVE A STORAGE ASSIGNMENT, AND STORAGE IS ALWAYS CORRECT AND     FT005620
*       UP-TO-DATE.  THIS IS A HIGHLY DESIRABLE FEATURE, IN OUR OPINION, FT005630
*       AS RELATIVE CONSTANTS ARE A CONSIDERABLE NUISANCE TO THE         FT005640
*       PROGRAMMER.  SOME RELCON 'ERRORS' IN FORTRAN II CAN BE SO        FT005650
*       WELL DISGUISED THAT MANY HOURS OR DAYS ARE SPENT IN DEBUGGING    FT005660
*       THE 'INCORRECT' PROGRAM.  ONE SIGNIFICANT RESULT OF DISPENSING   FT005670
*       WITH THE RELCON PROBLEM IS THAT IN FASTRAN, A DO-TYPE INDEX      FT005680
*       IN AN I/O LIST MAY NOT BE THE SAME VARIABLE NAME AS A            FT005690
*       CURRENT DO-STATEMENT INDEX, SINCE THE SAME STORAGE LOCATION      FT005700
*       IS ASSIGNED TO BOTH.  THIS IS CAUGHT IN FASTRAN AS AN ERROR...   FT005710
*       NO INCORRECT CODE IS PRODUCED.                                   FT005720
        REM                                                      FT005730
****    FIXED POINT EXPRESSIONS INVOLVING A SUCCESSION OF MULTIPLY       FT005740
*       AND DIVIDE OPERATIONS (E.G.  I*J/K OR I*J/K*L) ARE NOT           FT005750
*       REARRANGED, AS IS DONE IN 7090 FORTRAN II.  IN FASTRAN,          FT005760
*       THE CODE GENERATED IS EFFECTIVELY THAT DUE TO A LEFT-TO-RIGHT    FT005770
*       SCAN OF THE EXPRESSION.                                          FT005780
        REM                                                      FT005790
****    UNDEFINED VARIABLES AND STATEMENT LABELS IN THE SOURCE           FT005800
*       PROGRAM CAUSE AN ERROR MESSAGE.  A VARIABLE USED IN THE PROGRAM  FT005810
*       IS UNDEFINED IF IT DOES NOT HAVE ONE OR MORE OF THE FOLLOWING    FT005820
*       PROPERTIES...                                                    FT005830
*       IN A COMMON STATEMENT, EQUIVALENCED TO A COMMON VARIABLE,        FT005840
*       IN AN INPUT LIST, ON THE LEFT-HAND SIDE OF AN EQUAL SIGN,        FT005850
*       IN A SUBROUTINE OR FUNCTION STATEMENT ARGUMENT LIST, IN THE      FT005860
*       ARGUMENT LIST OF A CALL STATEMENT OR A CALL TO AN EXTERNAL       FT005870
*       FUNCTION (FORTRAN OR LIBRARY), USED AS A DO-INDEX OR AN          FT005880
*       INDEX IN AN INPUT/OUTPUT LIST, USED AS AN ASSIGN VARIABLE.       FT005890
*       STATEMENT LABELS ARE UNDEFINED IF THEY DO NOT APPEAR IN THE      FT005900
*       LABEL FIELD OF AN EXECUTABLE STATEMENT. (ALSO, OF COURSE ALL     FT005910
*       FORMAT STATEMENT LABELS MUST BE DEFINED).                        FT005920
        REM                                                      FT005930
***     NO LIST OR LIST8 FEATURE IS IMPLEMENTED.                 FT005940
        REM                                                      FT005950
***     IMPLIED MULTIPLICATION IS NOT RECOGNIZED IN FASTRAN.     FT005960
        REM                                                      FT005970
***     ALL FORMS OF LITERALS WORK.  TERMINAL BLANKS ARE OKAY.   FT005980
*       LITERALS IN BOOLEAN STATEMENTS WORK IF THE CHARACTER COUNT IS    FT005990
*       IN OCTAL. (LITERALS IN BOOLEAN STATEMENTS DO NOT WORK AT ALL     FT006000
*       IN IBM FORTRAN II).                                              FT006010
        REM                                                      FT006020
***     ERROR MESSAGES ARE ISSUED IN-LINE WHENEVER POSSIBLE.  I.E.       FT006030
*       THEY ARE ISSUED IMMEDIATELY WHEN THE ERROR IS DIAGNOSED.         FT006040
        REM                                                      FT006050
***     STATEMENTS LIKE  A=0, A=0.0, I=0, ETC. CAUSE STZ INSTRUCTIONS    FT006060
```

```
*       TO BE GENERATED.                                                FT006070
        REM                                                             FT006080
***     SOURCE PROGRAM DOUBLE PRECISION FLOATING POINT CONSTANTS MAY BE  FT006090
*       MORE ACCURATE THAN IN FORTRAN II (BY ONE BINARY BIT), DUE TO     FT006100
*       JUDICIOUS ROUNDING IN THE FASTRAN COMPILER NUMBER PROCESSOR.     FT006110
        REM                                                             FT006120
***     THE COLLECTION OF ASSIGN AND ASSIGNED GO TO STATEMENTS IN THE    FT006130
*       SOURCE PROGRAM IS CHECKED FOR CONSISTENCY.  ASSIGN STATEMENT     FT006140
*       VARIABLES MUST APPEAR IN AN ASSIGNED GO TO STATEMENT, AND        FT006150
*       VICE VERSA.  THE STATEMENT LABEL IN EACH ASSIGN STATEMENT        FT006160
*       MUST APPEAR IN THE LABEL LIST OF AN ASSIGNED GO TO FOR THE       FT006170
*       SAME ASSIGN VARIABLE.                                           FT006180
        REM                                                             FT006190
***     THE VALUE OF A VARIABLE SUBSCRIPT, EXCLUSIVE OF ITS ADDEND,      FT006200
*       NEED NOT BE LESS THAN OR EQUAL TO THE CORRESPONDING ARRAY        FT006210
*       DIMENSION, AS IS REQUIRED IN FORTRAN II FOR CORRECT CODE.        FT006220
        REM                                                             FT006230
**      F-CARDS MUST PRECEDE IN THE SOURCE PROGRAM THE USAGES OF THEIR   FT006240
*       EXTERNAL NAMES.                                                 FT006250
        REM                                                             FT006260
**      THE STANDARD ERROR OPTION IS NOT AVAILABLE IN FASTRAN COMPILED   FT006270
*       PROGRAMS.                                                       FT006280
        REM                                                             FT006290
**      NO CHECKING FOR VALIDITY OF FORMAT STATEMENTS IS DONE IN FASTRAN.FT006300
        REM                                                             FT006310
**      SINCE HPR MAY NOT SAFELY BE USED TO HALT THE COMPUTER WHEN       FT006320
*       THE DATA CHANNEL TRAP IS BEING USED, HTR MUST BE USED INSTEAD IN FT006330
*       OBJECT PROGRAM CODE FOR THE 'STOP N' AND 'PAUSE N' STATEMENTS.   FT006340
*       THUS THE STORAGE REGISTER ADDRESS IS UNAVAILABLE FOR THE         FT006350
*       CONSTANT N.  THEREFORE, FASTRAN PLACES THE CONSTANT N (OR ZERO)  FT006360
*       IN THE ADDRESS OF BOTH THE ACCUMULATOR AND THE MQ.               FT006370
        REM                                                             FT006380
**      THE COMPILER USES THE FACT THAT THE DOUBLE PRECISION ADD,        FT006390
*       SUBTRACT, MULTIPLY, AND DIVIDE LIBRARY ROUTINES LEAVE THE        FT006400
*       RESULT IN BOTH THE PSEUDO ACCUMULATOR (OR PSEUDO MQ) AND         FT006410
*       ALSO IN THE ACTUAL AC+MQ.  (THIS IS EASILY ALTERED, IF AN        FT006420
*       INSTALLATION CANNOT ACCEPT THIS OBJECT PROGRAM CODE SHORTCUT.)   FT006430
        REM                                                             FT006440
**      SUCH INCONSISTENCIES AS '5  GO TO 5'  OR  '5  IF (A) 5,6,7'      FT006450
*       ARE CAUGHT AS ERRORS IN FASTRAN.  IN OTHER WORDS, A STATEMENT    FT006460
*       MAY NOT REFER TO ITSELF.                                        FT006470
        REM                                                             FT006480
**      UNDER FASTRAN, THERE IS NO SCAN FOR MONITOR * CONTROL CARDS      FT006490
*       WITHIN A FORTRAN PROGRAM, AND THUS THEY ARE NOT LISTED           FT006500
*       ON-LINE OR INTERPRETED.  (* CARDS ARE LEGAL IN A FASTRAN         FT006510
*       PROGRAM... THEY ARE TREATED AS COMMENTS).                       FT006520
        REM                                                             FT006530
**      COMMENT AND ASTERISK CARDS MAY NOT IMMEDIATELY PRECEDE A         FT006540
*       STATEMENT CONTINUATION CARD.  BLANK CARDS ARE ACCEPTABLE         FT006550
*       ANYWHERE.                                                       FT006560
        REM                                                             FT006570
*       THE DOUBLE PRECISION AND COMPLEX BUILT-IN FUNCTION 'FIXF'        FT006580
*       IS NOT ALLOWED (ERROR MESSAGE IF USED).  THIS IS BECAUSE THE     FT006590
*       IBM FORTRAN II COMPILER GENERATES INCORRECT CODE FOR THIS        FT006600
*       FUNCTION.                                                       FT006610
        REM                                                             FT006620
*       THE LIBRARY FUNCTION 'XLOC' IS BUILT-IN IN FASTRAN.              FT006630
        REM                                                             FT006640
*       THE USE OF A PHYSICAL TAPE DESIGNATION SUCH AS A4 IN             FT006650
*       'CALL CHAIN (3,A4) IS NOT ALLOWED.  (HOWEVER, THE MONITOR        FT006660
```

```
*       CONTROL CARD '* CHAIN (3,A4)' IS PROCESSED BY THE MONITOR,      FT006670
*       AND THUS WOULD BE ACCEPTABLE.                                   FT006680
        REM                                                             FT006690
*       THE MONITOR END CARD OPTIONS MAY APPEAR ON THE END CARD, BUT    FT006700
*       THEY ARE IGNORED BY FASTRAN.                                    FT006710
        REM                                                             FT006720
*       FREQUENCY STATEMENTS ARE IGNORED.                               FT006730
        REM                                                             FT006740
*       A FUNCTION STATEMENT MUST HAVE ARGUMENTS, AND THE FUNCTION      FT006750
*       NAME MUST BE DEFINED WITHIN THE FUNCTION SUBPROGRAM.            FT006760
        REM                                                             FT006770
*       FASTRAN ALLOWS STATEMENT LABELS FROM 1 THROUGH 99999.           FT006780
        REM                                                             FT006790
*       THE READ DRUM AND WRITE DRUM STATEMENTS ARE NOT ACTIVATED       FT006800
*       IN THIS VERSION OF FASTRAN.                                     FT006810
```

```
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   FT006840
*       ADDITIONAL STATEMENTS ALLOWED IN FASTRAN                        FT006850
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   FT006860
        SPACE   3                                                       FT006870
*       IN ADDITION TO THE INCOMPATIBILITIES, THERE ARE SOME EXTENTIONS FT006880
*       TO THE LANGUAGE AVAILABLE FOR USE.                              FT006890
        SPACE   2                                                       FT006900
****    THE FASTRAN STATEMENT 'FLOATING ROUND' WILL CAUSE ROUNDING      FT006910
*       OF ALL SUBSEQUENT FASTRAN-COMPILED FLOATING POINT ARITHMETIC.   FT006920
*       NOTE THAT THIS IMPROVES ACCURACY, AND INCREASES THE LENGTH OF   FT006930
*       THE OBJECT PROGRAM.  THE FEATURE MAY BE TURNED OFF BY ISSUING   FT006940
*       A 'FLOATING ROUND OFF' STATEMENT.                               FT006950
        REM                                                             FT006960
****    A PAGE SUBTITLING OPTION IS ALLOWED.  A CARD WITH A $-SIGN      FT006970
*       IN COLUMN 1 INDICATES A PAGE SUBTITLE.  THIS CARD CAUSES        FT006980
*       A PAGE EJECT, AND SUBTITLING COMMENCES ON THE NEXT PAGE, COLUMNS FT006990
*       2-72 BEING USED AS THE SUBTITLE.  THIS SUBTITLE IS IN EFFECT    FT007000
*       UNTIL SUPERSEDED BY ANOTHER $-CARD.  TO SUSPEND SUBTITLING,     FT007010
*       USE A BLANK $-CARD.  (NOTE THAT THE MAIN PAGE TITLE MAY NOT     FT007020
*       BE ALTERED).                                                    FT007030
        REM                                                             FT007040
****    THE CHARACTER E IN COLUMN 1 OR THE WORD EJECT IN COLUMNS 1-5    FT007050
*       CAUSES A PAGE EJECT BEFORE PRINTING THE NEXT LINE OF THE        FT007060
*       SOURCE PROGRAM LISTING.  CONSECUTIVE E OR EJECT CARDS GIVE ONLY FT007070
*       ONE EJECT.                                                      FT007080
        REM                                                             FT007090
****    SPACING IN THE SOURCE PROGRAM LISTING MAY BE EFFECTED BY A      FT007100
*       CARD HAVING AN S IN COLUMN 1 OR THE WORD SPACE IN COLUMNS 1-5.  FT007110
*       AN INTEGER MUST APPEAR BEGINNING IN COLUMN 7, DENOTING THE      FT007120
*       NUMBER OF LINES TO BE SKIPPED.  IF THE SPACING CAUSES A PAGE    FT007130
```

```
*       EJECT, SPACING IS NOT CONTINUED ON THE NEW PAGE.  TO SKIP ONE     FT007140
*       LINE, USE A COMPLETELY BLANK CARD.                                FT007150
*       SPACE CARDS IMMEDIATELY FOLLOWING AN EJECT CARD OR A SUBTITLE     FT007160
*       CARD ARE IGNORED.                                                 FT007170
        REM                                                               FT007180
****    AN OBJECT PROGRAM SYMBOL TABLE MAY BE REQUESTED BY INCLUDING A    FT007190
*       FASTRAN STATEMENT 'SYMBOL TABLE' ANYWHERE IN THE SOURCE DECK.     FT007200
*       THIS STATEMENT TURNS ON THE SYMBOL TABLE BIT (BIT 31) IN THE      FT007210
*       FMS MONITOR FLAG BOX AT LOCATION 18.  FASTRAN LATER INTERROGATES  FT007220
*       THIS BIT TO DETERMINE IF A SYMBOL TABLE SHOULD BE OUTPUT.  NOTE   FT007230
*       THAT THIS IS ALSO AN FMS MONITOR FUNCTION (*  SYMBOL TABLE).      FT007240
        REM                                                               FT007250
****    AN ON-LINE LISTING MAY BE OBTAINED BY ISSUING A FASTRAN STATEMENT FT007260
*       'ONLINE'.  OUTPUT WILL THEN COMMENCE BOTH ON-LINE AND OFF-LINE    FT007270
*       AND CONTINUE UNTIL THE END OF THE COMPILATION OR UNTIL THE        FT007280
*       ON-LINE FEATURE IS TERMINATED BY AN 'OFFLINE' FASTRAN STATEMENT.  FT007290
*       NOTE THAT ANY NORMAL ON-LINE COMMENTARY (SUCH AS 'SOURCE PROGRAM  FT007300
*       ERROR') WILL APPEAR INTERSPERSED AMONG THE LINES OF THE SOURCE    FT007310
*       PROGRAM LISTING.                                                  FT007320
```

```
      TTL I N T R O D U C T I O N -- OBJECT PROGRAM STORAGE LAYOUT      FT007340
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  FT007350
*     OBJECT PROGRAM STORAGE LAYOUT                                    FT007360
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  FT007370
      SPACE  3                                                         FT007380
*     THE DETAILED ORDER OF THE VARIOUS SECTIONS OF THE OBJECT PROGRAM  FT007390
*     RESULTING FROM A FASTRAN COMPILATION IS AS FOLLOWS (BEGINNING AT  FT007400
*     LOCATION ZERO)...                                                FT007410
      REM                                                              FT007420
*        TRANSFER VECTOR                                               FT007430
*           LIBRARY REFERENCES                                         FT007440
*           NON-LIBRARY REFERENCES                                     FT007450
*        PROLOGUE (FOR SUBPROGRAMS ONLY)                               FT007460
*           INDEX REGISTER RESTORING INSTRUCTIONS                      FT007470
*           FINAL SUBPROGRAM EXIT                                      FT007480
*           INDEX REGISTER SAVING INSTRUCTIONS                         FT007490
*              (THIS IS THE SUBPROGRAM'S ENTRY POINT)                  FT007500
*           BODY OF PROLOGUE (SETTING OF ADDRESS IN TEXT)              FT007510
*           TRANSFER AROUND ARITH. STMNT. FCNS. IF THERE ARE ANY       FT007520
*        ARITHMETIC STATEMENT FUNCTIONS                                FT007530
*           (IN ORDER OF THEIR APPEARANCE IN THE SOURCE PROGRAM)       FT007540
*        FLOATING POINT TRAP INITIALIZATION (FOR MAIN PROGRAM ONLY)    FT007550
*           CAL    (FPT)                                               FT007560
*           SLW    8                                                   FT007570
*           STZ    77462                                               FT007580
*        TEXT (CODE GENERATED FOR EXECUTABLE STATEMENTS)               FT007590
*        CONSTANT STORAGE                                              FT007600
*        STRING (FORMAT) STORAGE                                       FT007610
*        WORKING STORAGE                                               FT007620
*        LOW-CORE (NON-COMMON) VARIABLE STORAGE                        FT007630
*           NON-DIMENSIONED AND DIMENSIONED VARIABLES INTERMIXED       FT007640
      REM      *                                                       FT007650
      REM      *                                                       FT007660
      REM      *                                                       FT007670
*        V A S T   E M P T I N E S S                                   FT007680
      REM      *                                                       FT007690
      REM      *                                                       FT007700
      REM      *                                                       FT007710
*        COMMON STORAGE                                                FT007720
```

```
        TTL F M S I I  E D I T O R  C O N T R O L  C A R D  N O.  1      FT007760
*       THE FOLLOWING 'INSTRUCTIONS' WILL CAUSE GENERATION OF THE PROGRAMFT007770
*       CARD REQUIRED BY THE FMS-II EDITOR TO EDIT FASTRAN INTO THE     FT007780
*       IOS/FMS-II SYSTEM.                                              FT007790
        REM                                                             FT007800
*       FASTRAN IS TOO BIG TO BE ACCEPTED AS ONE RECORD BY THE          FT007810
*       EDITOR.  THEREFORE THERE IS ANOTHER CONTROL CARD GENERATED      FT007820
*       SOMEWHERE BEFORE LOCATION 30000.                                FT007830
        REM                                                             FT007840
*       THESE TWO CARDS MUST BE PULLED OFF IF FASTRAN IS NOT BEING      FT007850
*       RUN UNDER THIS MONITOR SYSTEM.                                  FT007860
        SPACE   3                                                       FT007870
        ABS                                                             FT007880
        9LP     5                                                       FT007890
        IORT    PRGORG,,BEGRC2-PRGORG                                   FT007900
        TXI     BEGINY,,30          ENTRY PT.,,PROGRAM NUMBER           FT007910
        ABS                                                             FT007920
```

```
        TTL     M A C R O   D E F I N I T I O N S                    FT007930
                                                                     FT007940
*
        REM                                                          FT007950
                                                                     FT007960
        SPACE   9
*       *   *   *   *   *   *   *   *   *   *   *                    FT007970
                                                                     FT007980
        REM
*               F A S T R A N   C O M P I L E R                      FT007990
                                                                     FT008000
        REM
*       R E S E A R C H   C O M P U T I N G   C E N T E R            FT008010
                                                                     FT008020
        REM
*               I N D I A N A   U N I V E R S I T Y                  FT008030
                                                                     FT008040
        REM
*                 B L O O M I N G T O N ,   I N D I A N A            FT008050
                                                                     FT008060
        SPACE   3
*       *   *   *   *   *   *   *   *   *   *   *                    FT008070
                                                                     FT008080
        REM
*               M A C R O   D E F I N I T I O N S                    FT008090
                                                                     FT008100
        REM
*       *   *   *   *   *   *   *   *   *   *   *                    FT008110
```

```
        EJECT                                                                   FT008120
*       MACROS USED TO FORM AND COMMUNICATE OBJECT-TIME INSTRUCTION             FT008130
*       INFORMATION TO CITBLD.                                                  FT008140
        REM                                                                     FT008150
CITBLD  MACRO   A                                                               FT008160
        TSX     CITBLD,4,A                                                      FT008170
CITBLD  END                                                                     FT008180
        SPACE   2                                                               FT008190
CODEN   MACRO   A               MACRO FOR OUTPUTTING TYPE N INSTRUCTIONS        FT008200
        CAL     A                                                               FT008210
        TSX     CITBLD,4,N                                                      FT008220
CODEN   END                                                                     FT008230
        SPACE   2                                                               FT008240
CODEP   MACRO   A               MACRO FOR OUTPUTTING TYPE P INSTRUCTIONS        FT008250
        CAL     A                                                               FT008260
        ACL     PCOUNT                                                          FT008270
        TSX     CITBLD,4,P                                                      FT008280
CODEP   END                                                                     FT008290
        SPACE   2                                                               FT008300
CODETV  MACRO   A               MACRO TO OUTPUT A CALL TO AN EXTERNAL           FT008310
        TSX     OUTPTV,4        SUBROUTINE NAME.                                FT008320
        PZE     A                                                               FT008330
CODETV  END                                                                     FT008340
        SPACE   2                                                               FT008350
CODEC   MACRO   A,B,C                                                           FT008360
        CAL     A                                                               FT008370
        IFF     1,C,XRFLAG                                                      FT008380
        ORS     XRFLAG                                                          FT008390
        TSX     CITBLD,4,B                                                      FT008400
CODEC   END                                                                     FT008410
        SPACE   2                                                               FT008420
CODEO   MACRO   A,B,C                                                           FT008430
        ORA     A                                                               FT008440
        IFF     1,C,XRFLAG                                                      FT008450
        ORS     XRFLAG                                                          FT008460
        TSX     CITBLD,4,B                                                      FT008470
CODEO   END                                                                     FT008480
        SPACE   2                                                               FT008490
CODECO  MACRO   A,B,C,D                                                         FT008500
        CAL     A                                                               FT008510
        ORA     B                                                               FT008520
        IFF     1,D,XRFLAG                                                      FT008530
        ORS     XRFLAG                                                          FT008540
        TSX     CITBLD,4,C                                                      FT008550
CODECO  END                                                                     FT008560
        SPACE   2                                                               FT008570
*       MACRO FOR COMMUNICATING WITH THE GENERAL CODE ROUTINE.                  FT008580
        REM                                                                     FT008590
CODE    MACRO   OP,A                                                            FT008600
        TSX     CODE,4                                                          FT008610
        PZE     OP,,A                                                           FT008620
CODE    END                                                                     FT008630
        SPACE   2                                                               FT008640
*       SPECIAL PURPOSE MACROS TO COMMUNICATE WITH ROUTINES                     FT008650
*       WHICH WILL USE THE GENERAL CODE ROUTINE.                                FT008660
*       THESE MACROS CAUSE THE GENERATION OF TWO OBJECT-TIME                    FT008670
*       INSTRUCTIONS.  THEY ARE USED BY THE DOUBLE PRECISION AND                FT008680
*       COMPLEX ARITHMETIC SECTIONS OF THE COMPILER.                            FT008690
        REM                                                                     FT008700
CODLD   MACRO   A               MACRO TO LOAD THE AC AND MQ WITH OPERAND        FT008710
```

```
        TSX     CODLD,4                                                 FT008720
        PZE     CLA,,A                                                  FT008730
CODLD   END                                                             FT008740
        SPACE   2                                                       FT008750
CODLAC  MACRO   A               TO LOAD THE PSEUDO-AC (AND AC+MQ)       FT008760
        TSX     CODLAC,4                                                FT008770
        PZE     CLA,,A                                                  FT008780
CODLAC  END                                                             FT008790
        SPACE   2                                                       FT008800
CODSTO  MACRO   A               TO STORE THE AC+MQ                      FT008810
        TSX     CODSTO,4                                                FT008820
        PZE     STO,,A                                                  FT008830
CODSTO  END                                                             FT008840
        SPACE   2                                                       FT008850
CODSTR  MACRO   A               DP AND CA LIBRARY ROUTINE CALLING SEQUENCE FT008860
        TSX     CODSTR,4                                                FT008870
        PZE     STR,,A                                                  FT008880
CODSTR  END                                                             FT008890
        SPACE   2                                                       FT008900
CODSTZ  MACRO   A               TO ZERO OUT A DP OR CA OPERAND          FT008910
        TSX     CODSTZ,4                                                FT008920
        PZE     STZ,,A                                                  FT008930
CODSTZ  END                                                             FT008940
        SPACE   2                                                       FT008950
SETUP   MACRO   A               TO INITIALIZE A DP OR CA LIBRARY ROUTINE FT008960
        TSX     SETUP,4                                                 FT008970
        PZE     ('A')                                                   FT008980
SETUP   END                                                             FT008990
        SPACE   2                                                       FT009000
CODSUB  MACRO   A               TO LOAD AC+MQ WITH NEGATIVE OF OPERAND  FT009010
        TSX     CODSUB,4                                                FT009020
        PZE     CLS,,A                                                  FT009030
CODSUB  END                                                             FT009040
        SPACE   2                                                       FT009050
CODLMQ  MACRO   A               TO LOAD THE PSEUDO-MQ (AND AC+MQ)       FT009060
        TSX     CODLMQ,4                                                FT009070
        PZE     CLA,,A                                                  FT009080
CODLMQ  END                                                             FT009090
        SPACE   2                                                       FT009100
*       MACRO TO RECORD THE MODE AND REGISTER LOCATION OF A TEMPORARY   FT009110
*       RESULT DURING GENERATION OF ARITHMETIC CODE.                    FT009120
*       FIRST ARGUMENT IS THE MODE (REAL, INTEGR). SECOND ARGUMENT      FT009130
*       IS THE LOCATION (AC, MQ, PAC, PMQ).  IF THIRD ARGUMENT IS       FT009140
*       PRESENT, FLOATING ROUNDING IS PERFORMED BY ENTERING THE         FT009150
*       ROUNDX ROUTINE PRIOR TO ENTERING THE RESULT ROUTINE.            FT009160
        REM                                                             FT009170
RESULT  MACRO   A,B,C                                                   FT009180
        IFF     1,C                                                     FT009190
        TSX     RESULT,4,A*8+B                                          FT009200
        IFF     0,C                                                     FT009210
        TSX     ROUNDX,4,A*8+B                                          FT009220
RESULT  END                                                             FT009230
        SPACE   2                                                       FT009240
*       DEBUGGING MACRO.  PRINTS TYPE OF STATEMENT PROCESSOR IDENTIFIED. FT009250
        REM                                                             FT009260
XREMK   MACRO   A                                                       FT009270
        TRA     *+11                                                    FT009280
        BCI     , 'A'                                                   FT009290
        TSX     LIST,4                                                  FT009300
        PZE     *-11,,10                                                FT009310
```

```
XREMK   END                                                             FT009320
        SPACE   2                                                       FT009330
*       DEBUGGING MACRO.  DISABLED IN THIS VERSION.  TO MAKE THIS       FT009340
*       MACRO FUNCTIONAL, REPLACE IT WITH A 'REMARK OPSYN XREMK' CARD.  FT009350
        REM                                                             FT009360
REMARK  MACRO   A                                                       FT009370
        EQU     *                                                       FT009380
REMARK  END                                                             FT009390
        SPACE   2                                                       FT009400
*       MACRO TO FORM THE PROCESSOR KEY WORD.  GENERATED IN THE         FT009410
*       RESERVED WORD TEST.  B, C, AND D ARE ONE-BIT FLAGS.             FT009420
*       A GIVES THE ENTRY POINT TO THE PROCESSOR.                       FT009430
        REM                                                             FT009440
SPRSW   MACRO   A,B,C,D,E                                               FT009450
        CAL     *+2                                                     FT009460
        TRA     PROCX1                                                  FT009470
        VFD     1/B,1/C,1/D,015/E,3/,15/A                               FT009480
SPRSW   END                                                             FT009490
        SPACE   2                                                       FT009500
*       MACRO TO RECORD A SOURCE PROGRAM ERROR.  COMMUNICATION WITH THE FT009510
*       ERROR ROUTINE IS VIA AN STR TRAP.  ADDRESS OF THE STR HAS NAME  FT009520
*       OF THE ERROR MESSAGE, AND DECREMENT HAS THE POINT TO WHICH      FT009530
*       CONTROL IS TO BE RETURNED WHEN THE ERROR ROUTINE IS FINISHED.   FT009540
        REM                                                             FT009550
ERROR   MACRO   A,B                                                     FT009560
        STR     FORM'A',,B                                              FT009570
ERROR   END                                                             FT009580
        SPACE   2                                                       FT009590
*       MACRO TO RECORD A MACHINE ERROR.  USES THE ERROR ROUTINE.       FT009600
        REM                                                             FT009610
MACER   MACRO                                                           FT009620
        STR     MCR                                                     FT009630
MACER   END                                                             FT009640
        SPACE   2                                                       FT009650
*       MACRO TO TEST THE NOCODE FLAG.  NOCODE IS TURNED ON BY THE      FT009660
*       ERROR ROUTINE TO INDICATE A SOURCE PROGRAM ERROR.  IF AN ERROR  FT009670
*       HAS BEEN ENCOUNTERED, CONTROL LEAVES THE CURRENT PROCESSOR      FT009680
*       AND GOES TO SKEND.                                              FT009690
        REM                                                             FT009700
NOCODE  MACRO                                                           FT009710
        ZET     NOCODE                                                  FT009720
        TRA     SKEND                                                   FT009730
NOCODE  END                                                             FT009740
        SPACE   2                                                       FT009750
*       POST PASS 1 ERROR MACRO USED TO ACCUMULATE SYMBOLS IN ERROR     FT009760
*       FOR LATER PRINTING.                                             FT009770
        REM                                                             FT009780
ERRORY  MACRO   FLAG                                                    FT009790
        TSX     ERRORY,4,FLAG                                           FT009800
ERRORY  END                                                             FT009810
        SPACE   2                                                       FT009820
*       MACRO TO SAVE INDEX REGISTERS.                                  FT009830
        REM                                                             FT009840
SAVE    MACRO   Q                                                       FT009850
        SXA     Q,4                                                     FT009860
        SXA     Q+1,2                                                   FT009870
        SXA     Q+2,1                                                   FT009880
SAVE    END                                                             FT009890
        SPACE   2                                                       FT009900
*       MACRO TO RESTORE INDEX REGISTERS                                FT009910
```

```
        REM                                                                 FT009920
RESTOR  MACRO                                                               FT009930
        AXT     **,4                                                        FT009940
        AXT     **,2                                                        FT009950
        AXT     **,1                                                        FT009960
RESTOR  END                                                                 FT009970
        SPACE   2                                                           FT009980
*       MACRO TO COMMUNICATE WITH THE SKIP ROUTINE.  ARGUMENT IS A          FT009990
*       SYMBOL REPRESENTING ONE BCD CHARACTER (SIX BITS).  RETURN FROM      FT010000
*       SKIP IS TO 1,4 IF NEXT CHARACTER IN COLUMN IS NOT EQUAL TO          FT010010
*       THE ARGUMENT, AND TO 2,4 IF THE NEXT CHARACTER IS EQUAL TO          FT010020
*       THE  ARGUMENT.                                                      FT010030
        REM                                                                 FT010040
SKIP    MACRO   A                                                           FT010050
        TSX     SKIP,4,A                                                    FT010060
SKIP    END                                                                 FT010070
        SPACE   2                                                           FT010080
*       MACRO   TO COMMUNICATE WITH THE SKIP1 ROUTINE. SAME AS SKIP         FT010090
*       MACRO EXCEPT RETURNS ARE REVERSED.                                  FT010100
        REM                                                                 FT010110
SKIP1   MACRO   A                                                           FT010120
        TSX     SKIP1,4,A                                                   FT010130
SKIP1   END                                                                 FT010140
        SPACE   2                                                           FT010150
*       MACRO USED TO BUILD THE OPERATOR RANK TABLE.                        FT010160
        REM                                                                 FT010170
RANK    MACRO   OP,RANK,OPKEY,TYPE,ENTRY                                    FT010180
OP      EQU     *-RANKTB+1                                                  FT010190
OPKEY   VFD     7/OP,7/RANK,1/TYPE,2/1,4/,15/ENTRY                          FT010200
RANK    END                                                                 FT010210
        SPACE   2                                                           FT010220
*       MACROS USED TO BUILD THE GENERATOR KEY TABLES.                      FT010230
        REM                                                                 FT010240
KEY     MACRO   A,B,OP,D,E,F                                                FT010250
        VFD     7/OP,3/A,2/B,3/D,2/E,4/,15/F                                FT010260
KEY     END                                                                 FT010270
        SPACE   2                                                           FT010280
KEY1    MACRO   OP,M,SUB                                                    FT010290
        KEY     0,0,OP,M,VS,SUB*1                                           FT010300
        KEY     0,0,OP,M,AC,SUB*2                                           FT010310
        KEY     0,0,OP,M,MQ,SUB*3                                           FT010320
KEY1    END                                                                 FT010330
        SPACE   2                                                           FT010340
KEY2    MACRO   M1,OP,M2,SUB                                                FT010350
        KEY     M1,VS,OP,M2,VS,SUB*1                                        FT010360
        KEY     M1,VS,OP,M2,AC,SUB*2                                        FT010370
        KEY     M1,VS,OP,M2,MQ,SUB*3                                        FT010380
        KEY     M1,AC,OP,M2,VS,SUB*4                                        FT010390
        KEY     M1,MQ,OP,M2,VS,SUB*5                                        FT010400
KEY2    END                                                                 FT010410
        SPACE   2                                                           FT010420
KEY3    MACRO   M1,OP,M2,SUB                                                FT010430
        KEY     M1,VS,OP,M2,VS,SUB*1                                        FT010440
        KEY     M1,VS,OP,M2,AC,SUB*2                                        FT010450
        KEY     M1,AC,OP,M2,VS,SUB*4                                        FT010460
KEY3    END                                                                 FT010470
        SPACE   2                                                           FT010480
KEY4    MACRO   OP,M,SUB                                                    FT010490
        KEY     0,0,OP,M,VS,SUB*1                                           FT010500
        KEY     0,0,OP,M,AC,SUB*2                                           FT010510
```

```
KEY4    END                                                     FT010520
        SPACE   2                                               FT010530
*       MACRO   USED TO BUILD THE OPEN FUNCTION KEY TABLE.      FT010540
        REM                                                     FT010550
TAB3    MACRO   A                                               FT010560
A       EQU     *-LIBTAB+64                                     FT010570
        VFD     7/A,011/BINTF+BOPNF                             FT010580
        BCI     1,A                                             FT010590
TAB3    END                                                     FT010600
        SPACE   2                                               FT010610
```

```
        TTL     I O S  R O U T I N E  L O C A T I O N S,  E T C.        FT010620
*       THE FOLLOWING MATERIAL IS NEEDED TO IMPLEMENT THIS VERSION OF   FT010630
*       FASTRAN, WHICH DOES NOT DEPEND ON INDIANA UNIVERSITY'S VERSION  FT010640
*       OF THE FAP ASSEMBLER FOR THE MEANING OF THE IOS CALLS.          FT010650
*       SEE THE INTRODUCTION FOR MORE DETAILS ON THE INPUT/OUTPUT       FT010660
*       SCHEME IN FASTRAN.                                              FT010670
        SPACE   3                                                       FT010680
*       DECLARE ALL IU-FAP IOS OPERATIONS TO BE NULL                    FT010690
        REM                                                             FT010700
IOBASE  OPSYN   NULL                                                    FT010710
READ    OPSYN   NULL                                                    FT010720
WRITE   OPSYN   NULL                                                    FT010730
REWIND  OPSYN   NULL                                                    FT010740
WEOF    OPSYN   NULL                                                    FT010750
BACKR   OPSYN   NULL                                                    FT010760
PRINT   OPSYN   NULL                                                    FT010770
RCLOCK  OPSYN   NULL                                                    FT010780
XEOF    OPSYN   NULL                                                    FT010790
XRDN    OPSYN   NULL                                                    FT010800
XMODER  OPSYN   NULL                                                    FT010810
BUFFER  OPSYN   NULL                                                    FT010820
CLOSE   OPSYN   NULL                                                    FT010830
LOAD    OPSYN   NULL                                                    FT010840
LREAD   OPSYN   NULL                                                    FT010850
        SPACE   3                                                       FT010860
*       DEFINE THE LOCATIONS OF THE ROUTINES IN IOS                     FT010870
        REM                                                             FT010880
IOEX    BOOL    241                                                     FT010890
READ    BOOL    242                                                     FT010900
WRITE   BOOL    243                                                     FT010910
REWIND  BOOL    244                                                     FT010920
WEOF    BOOL    246                                                     FT010930
BACKR   BOOL    247                                                     FT010940
PRINT   BOOL    253                                                     FT010950
RCLOCK  BOOL    256                                                     FT010960
XEOF    BOOL    257                                                     FT010970
XRDN    BOOL    260                                                     FT010980
XMODER  BOOL    262                                                     FT010990
BUFFER  BOOL    263                                                     FT011000
CLOSE   BOOL    264                                                     FT011010
LOAD    BOOL    266                                                     FT011020
LREAD   BOOL    303                                                     FT011030
        SPACE   3                                                       FT011040
*       THE FOLLOWING IOS CALL IS NEEDED IN THE IU VERSION OF FASTRAN   FT011050
*       TO LOCATE THE ORIGIN OF IOS MACROS FOR FAP.                     FT011060
        REM                                                             FT011070
        IOBASE  161             BASE FOR ALL IOS MACROS                 FT011080
```

```
                TTL     C O M P I L E R   O R I G I N ,   E T C.                FT011090
PRGORG          BOOL    10000           ABSOLUTE ORIGIN OF FASTRAN COMPILER     FT011100
                ORG     PRGORG                                                  FT011110
                REM                                                             FT011120
                REM                                                             FT011130
*               DEFINE TSX INSTRUCTION TO ALLOW A DECREMENT                     FT011140
TSX             OPD     007415160002                                            FT011150
```

```
        TTL     M A N U A L   E N T R Y   F O R   E R R O R   S T O P          FT011160
*       IN CASE OF AN UNEXPECTED STOP DURING DEBUGGING OF THE COMPILER,        FT011170
*       THE OPERATOR MAY TRANSFER MANUALLY TO LOCATION 10000 OCTAL.            FT011180
*       THE SYMBOL TABLE WILL BE DUMPED ONTO THE OUTPUT TAPE, IF A             FT011190
*       SYMBOL TABLE LIST WAS REQUESTED (IF LSTSYM IS ON).  THEN AN            FT011200
*       ON-LINE MESSAGE WILL TELL THE OPERATOR TO TAKE A CORE DUMP.            FT011210
        SPACE   3                                                              FT011220
WHAMMY  ZET     LSTSYM          LIST SYMBOL TABLE ONLY IF REQUESTED            FT011230
        TSX     STDUMP,4                                                       FT011240
IOS01   EQU     *                                                              FT011250
        WEOF    *MLSTAP             ENTRY IN CASE OF EXECUTION HALT TO         FT011260
        STL     IOEX                                                           FT011270
        BRA     WEOF,,*+2                                                      FT011280
        MTW     MLSTAP                                                         FT011290
END01   REM                                                                    FT011300
IOS02   EQU     *                                                              FT011310
        BACKR   *MLSTAP,1            BACK OVER THE E-O-F                       FT011320
        STL     IOEX                                                           FT011330
        BRA     BACKR,,*+2                                                     FT011340
        MTW     MLSTAP,,1                                                      FT011350
END02   REM                                                                    FT011360
IOS03   EQU     *                                                              FT011370
        PRINT   DUMPRQ,4,DUMPRQ+4,1,DUMPRQ+4,1,DUMPRQ+4,1                      FT011380
        STL     IOEX                                                           FT011390
        BRA     PRINT,,*+5                                                     FT011400
        PZE     DUMPRQ,,4                                                      FT011410
        PZE     DUMPRQ+4,,1                                                    FT011420
        PZE     DUMPRQ+4,,1                                                    FT011430
        MZE     DUMPRQ+4,,1                                                    FT011440
END03   REM                                                                    FT011450
        HTR     *                                                              FT011460
DUMPRQ  BCI     4,0NOW TAKE CORE DUMP                                          FT011470
        BCI     1,0                                                            FT011480
```

```
      TTL   I N I T I A L I Z A T I O N                          FT011490
*                                                                FT011500
      REM                                                        FT011510
      SPACE  9                                                   FT011520
*       *  *  *  *  *  *  *  *  *  *  *  *                       FT011530
      REM                                                        FT011540
*           F A S T R A N  C O M P I L E R                       FT011550
      REM                                                        FT011560
*     R E S E A R C H  C O M P U T I N G  C E N T E R            FT011570
      REM                                                        FT011580
*           I N D I A N A  U N I V E R S I T Y                   FT011590
      REM                                                        FT011600
*        B L O O M I N G T O N ,  I N D I A N A                  FT011610
      SPACE  3                                                   FT011620
*       *  *  *  *  *  *  *  *  *  *  *  *                       FT011630
      REM                                                        FT011640
*             I N I T I A L I Z A T I O N                        FT011650
      REM                                                        FT011660
*       *  *  *  *  *  *  *  *  *  *  *  *                       FT011670
```

```
        TTL     I N I T I A L I Z A T I O N                                 FT011680
*       THE FOLLOWING ARE THE FIRST INSTRUCTIONS EXECUTED AFTER FASTRAN     FT011690
*       IS LOADED AND ARE EXECUTED ONE TIME ONLY.  THEY HAVE THE            FT011700
*       FOLLOWING FOUR FUNCTIONS,                                           FT011710
        SPACE   2                                                           FT011720
*          1. SORT GENERATOR KEY TABLES FOR,                                FT011730
*                A. REAL AND INTEGER                                       FT011740
*                B. BOOLEAN                                                FT011750
*                C. DOUBLE PRECISION                                       FT011760
*                D. COMPLEX                                                FT011770
        REM                                                                 FT011780
*          2. DEFINE BUFFERS FOR,                                           FT011790
*                A. MINTAP = SYSTEM INPUT TAPE                             FT011800
*                B. MLSTAP = SYSTEM OUTPUT TAPE                            FT011810
*                C. SYSBIN = SYSTEM (BINARY) LOAD TAPE                     FT011820
*                D. SYSUT1 = SYSTEM SCRATCH TAPE NO. 1                     FT011830
        REM                                                                 FT011840
*          3. DEFINE MODE ERROR AND END-OF-FILE RETURNS                     FT011850
        REM                                                                 FT011860
*          4. SET LOCATIO 2 = TRA   ERROR AND SET LOCATION 8 FOR            FT011870
*             FLOATING POINT TRAP.                                          FT011880
        SPACE   2                                                           FT011890
*       ON COMPLETION OF THIS PRE-INITIALIZATION SECTION, CONTROL IS        FT011900
*       GIVEN TO THE MAIN INITIALIZATION SECTION TO INITIALIZE FOR          FT011910
*       THIS COMPILATION.                                                   FT011920
        SPACE   3                                                           FT011930
        REM                                                                 FT011940
        REM                                                                 FT011950
        REM                                                                 FT011960
        REM                                                                 FT011970
IOS04   EQU     *                                                           FT011980
BEGINY  LOAD    ,,31                LOAD NEXT SYSTEM RECORD (PART 2)        FT011990
BEGINY  STL     IOEX                                                        FT012000
        BRA     LOAD,,*+3                                                   FT012010
        PZE                                                                 FT012020
        MZE     31                                                          FT012030
END04   REM                                                                 FT012040
BEGINZ  CAL     =6                  SET MLSTAP TO A3                        FT012050
        SLW     MLSTAP                  SYSBIN TO B4                        FT012060
        CAL     =7                   IF THIS IS MONITORED RUN               FT012070
        SLW     SYSBIN                                                      FT012080
IOS05   EQU     *                                                           FT012090
        REWIND  SYSTP1              START A1 REWINDING                      FT012100
        STL     IOEX                                                        FT012110
        BRA     REWIND,,*+2                                                 FT012120
        MZE     SYSTP1                                                      FT012130
END05   REM                                                                 FT012140
        STL     MONTOR                  INDICATE MONITORED RUN              FT012150
BEGINX  EQU     *               INITIAL ENTRY POINT FOR COMPILER            FT012160
        AXT     4,1             SORT THE GENERATOR KEY TABLES               FT012170
ONCE1   CAL     SORTKO,1        GET AN INFORMATION WORD                     FT012180
        STA     ONCE+1                                                      FT012190
        ARS     18                                                          FT012200
        STA     ONCE+2                                                      FT012210
        REM                                                                 FT012220
   ONCE CALL    SORT,,,,,,                                                  FT012230
        REM                                                                 FT012240
        CAL     GENDES+1,1      GET LENGTH OF ONE OF THE KEY TABLES         FT012250
        ARS     6               AND DETERMINE THE POWER-OF-TWO LENGTH       FT012260
        ORA     =0217000000000      OF THIS TABLE.  THIS INFORMATION         FT012270
```

```
         FAD     =0217000000000     IS USED BY THE GEN ROUTINE'S                FT012280
         SUB     =0200000000000     BINARY SEARCH SECTION.                      FT012290
         ARS     9+18               SHIFT COUNT INTO ADDRESS                    FT012300
         STA     GENBIN+1,1         SAVE IN ADDRESS OF BINSER CELL              FT012310
         PAX     ,4                 GET POINTER TO THE BINSER TABLE             FT012320
         CAL     BLOWER,4           DECREMENT OF THIS TABLE WORD HAS THE        FT012330
         STD     GENBIN+1,1         NEXT LOWER POWER OF TWO THAN THE TABLE      FT012340
         REM                        LENGTH.  SET INTO THE BINSER CELL.          FT012350
         REM                                                                    FT012360
         TIX     ONCE1,1,1                                                      FT012370
         REM                                                                    FT012380
IOS06    EQU     *                                                              FT012390
         BUFFER  BUFBIN,2,246,*SYSBIN                                           FT012400
         STL     IOEX                                                           FT012410
         BRA     BUFFER,,*+3                                                    FT012420
         PZE     BUFBIN,,2                                                      FT012430
         MON     246,,SYSBIN                                                    FT012440
END06    REM                                                                    FT012450
IOS07    EQU     *                                                              FT012460
         BUFFER  BUFUT1,2,179,SYSUT1                                            FT012470
         STL     IOEX                                                           FT012480
         BRA     BUFFER,,*+3                                                    FT012490
         PZE     BUFUT1,,2                                                      FT012500
         MZE     179,,SYSUT1                                                    FT012510
END07    REM                                                                    FT012520
IOS08    EQU     *                                                              FT012530
         BUFFER  BUFMIT,2,302,MINTAP                                            FT012540
         STL     IOEX                                                           FT012550
         BRA     BUFFER,,*+3                                                    FT012560
         PZE     BUFMIT,,2                                                      FT012570
         MZE     302,,MINTAP                                                    FT012580
END08    REM                                                                    FT012590
IOS09    EQU     *                                                              FT012600
         BUFFER  BUFMLT,2,302,*MLSTAP                                           FT012610
         STL     IOEX                                                           FT012620
         BRA     BUFFER,,*+3                                                    FT012630
         PZE     BUFMLT,,2                                                      FT012640
         MON     302,,MLSTAP                                                    FT012650
END09    REM                                                                    FT012660
IOS10    EQU     *                                                              FT012670
         XMODER  GETMER                                                         FT012680
         STL     IOEX                                                           FT012690
         BRA     XMODER,,*+2                                                    FT012700
         MZE     GETMER                                                         FT012710
END10    REM                                                                    FT012720
IOS11    EQU     *                                                              FT012730
         XEOF    GETEOF                                                         FT012740
         STL     IOEX                                                           FT012750
         BRA     XEOF,,*+2                                                      FT012760
         MZE     GETEOF                                                         FT012770
END11    REM                                                                    FT012780
         REM                                                                    FT012790
         LDQ     ERROR              INITIALIZE EXECUTION ERROR SUBROUTINE       FT012800
         STQ     2                                                              FT012810
         REM                                                                    FT012820
         CAL     (FPT).             INITIALIZE THE FLOATING TRAP CELL           FT012830
         SLW     8                                                              FT012840
         REM                                                                    FT012850
         STZ     0                  TO RESET POSSIBLE TAG IN CELL 0             FT012860
         SLF                        TURN OFF THE LIGHTS                         FT012870
```

```
         REM                                                          FT012880
         AXT     POOL-COLUMN-665,4    LENGTH OF THE POOL REGION       FT012890
         STZ     POOL+1,4       ZERO OUT THE POOL AREA THE FIRST TIME FT012900
         TIX     *-1,4,1                                              FT012910
         REM                                                          FT012920
         TRA     START                                                FT012930
         SPACE   2                                                    FT012940
         REM                  INFORMATION WORDS FOR TABLE SORTS       FT012950
         PZE     KEYTB0,,KEYTB0+LKYTB0-1                              FT012960
         PZE     KEYTB1,,KEYTB1+LKYTB1-1                              FT012970
         PZE     KEYTB2,,KEYTB2+LKYTB2-1                              FT012980
         PZE     KEYTB3,,KEYTB3+LKYTB3-1                              FT012990
SORTKO   EQU     *                                                    FT013000
```

```
        EJECT                                                                   FT013010
*       THE FOLLOWING INSTRUCTIONS MAKE UP THE NORMAL INITIALIZATION            FT013020
*       SECTION OF THE FASTRAN COMPILER.  THIS SECTION IS EXECUTED PRIOR        FT013030
*       TO THE COMPILATION OF A PROGRAM.                                        FT013040
        SPACE   3                                                               FT013050
        REM                                                                     FT013060
START   EQU     *                                                               FT013070
        SWT     6                                                               FT013080
        TRA     *+3             NO TIME IF SSW 6 UP                             FT013090
IOS12   EQU     *                                                               FT013100
        RCLOCK                  GET TIME IF SSW 6 DOWN                          FT013110
        STL     IOEX                                                            FT013120
        BRA     RCLOCK,,*+1                                                     FT013130
END12   REM                                                                     FT013140
        REM                                                                     FT013150
        SLN     4               TURN ON CONSOLE WATCHER'S LIGHT                 FT013160
        REM                                                                     FT013170
*       INITIALIZE THE SYMBOL TABLE                                             FT013180
        AXT     LSYMTB+1,4                                                      FT013190
        STZ     SYMTAB+1,4                                                      FT013200
        TIX     *-1,4,1                                                         FT013210
        REM                                                                     FT013220
*       INITIALIZE THE GETCH ROUTINE                                            FT013230
        STL     GETR1           SEQUENCE INVOLVED IN ONE-TIME TITLING SECT.     FT013240
        NOP     GET1            THIS CELL IS USED IN A 'TRA*'                   FT013250
        REM                                                                     FT013260
IOS13   EQU     *                                                               FT013270
        REWIND  SYSUT1          REWIND THE SCRATCH TAPE                         FT013280
        STL     IOEX                                                            FT013290
        BRA     REWIND,,*+2                                                     FT013300
        MZE     SYSUT1                                                          FT013310
END13   REM                                                                     FT013320
        REM                                                                     FT013330
        LXA     NXTLOC,4        INITIALIZE THE POOL AREA TO ZERO                FT013340
        TXL     *+3,4,0         SKIP OUT IF NO POOL WAS USED                    FT013350
        STZ     POOL+1,4                                                        FT013360
        TIX     *-1,4,1                                                         FT013370
        REM                                                                     FT013380
*       INITIALIZE ALL COMPILER FLAG CELLS TO ZERO.                             FT013390
        AXT     TRIGR2-TRIGR1,4     ZERO OUT BLOCK OF SWITCHES                  FT013400
        STZ     TRIGR2,4                                                        FT013410
        TIX     *-1,4,1                                                         FT013420
        REM                                                                     FT013430
        CAL     ALLSVN          INITIALIZE ASTACK+1 TO ALL ONES                 FT013440
        SLW     ASTACK+1                                                        FT013450
        REM                                                                     FT013460
        CAL     =0200500000000      INITIALIZE THE '7/9' PUNCH...               FT013470
        SLW     9LEFT           AFTER ZEROING OUT THE CELL.                     FT013480
        REM                                                                     FT013490
        ZSD     WORKS                                                           FT013500
        ZSD     GETWR1                                                          FT013510
        ZSD     FREWR1                                                          FT013520
        REM                                                                     FT013530
        CAL     CITBL9                                                          FT013540
        SLW     CITBLH                                                          FT013550
        AXT     CIT1-1,4        INITIALIZE CITBLD ROUTINE FOR CIT1              FT013560
        SXA     CTLOCA,4                                                        FT013570
        SXA     GETLCC,4                                                        FT013580
        AXT     CTFLG1-1,4                                                      FT013590
        SXA     CTLOCB,4                                                        FT013600
```

```
SXA     CTLOCC,4                                                FT013610
SXA     CTLOCD,4                                                FT013620
SXA     CTLOCE,4                                                FT013630
SXA     GETLCA,4                                                FT013640
SXA     GETLCB,4                                                FT013650
ZSA     STSH1                                                   FT013660
ZSD     TBUND-3                                                 FT013670
ZSD     TBUND-2                                                 FT013680
ZSD     TBUND-1                                                 FT013690
ZSD     TBUND                                                   FT013700
REM                                                             FT013710
AXT     1,1             X                                       FT013720
SXA     GETFG1,1        X                                       FT013730
SXA     GETFG2,1        X                                       FT013740
SXA     PROLOG+2,1                                              FT013750
SXA     CTBL1,1                                                 FT013760
SXA     CTBL2,1                                                 FT013770
SXA     NXTLOC,1                                                FT013780
REM                                                             FT013790
AXT     6,1             X                                       FT013800
SXA     GETFG2+1,1      X                                       FT013810
REM                                                             FT013820
AXT     20,1                                                    FT013830
SXA     STSH1-1,1                                               FT013840
REM                                                             FT013850
AXT     36,1                                                    FT013860
SXA     CTBL2+1,1                                               FT013870
REM                                                             FT013880
AXT     EQUTBL-INITFG,4 EQUIVALENCE MASTER DIRECTOR CELL        FT013890
PXA     ,4                                                      FT013900
SLW     INITFG                                                  FT013910
REM                                                             FT013920
REM             INITIALIZE FLAG CELLS PERTAINING TO             FT013930
REM             CLOSED AND RESERVED LIBRARY NAMES               FT013940
AXT     LSLNTB,4                                                FT013950
STZ     SLNTAB+LSLNTB,4                                         FT013960
TIX     *-1,4,2                                                 FT013970
REM                                                             FT013980
SXA     MAP22,4         SET MAP22 ADDRESS TO 2.                 FT013990
REM                                                             FT014000
TRA     STARTT                                                  FT014010
```

```
      TTL    P A S S  1  -  D E S C R I P T I O N                    FT014020
*                                                                    FT014030
      REM                                                            FT014040
      SPACE  9                                                       FT014050
*        *   *   *   *   *   *   *   *   *   *   *                   FT014060
      REM                                                            FT014070
*           F A S T R A N  C O M P I L E R                           FT014080
      REM                                                            FT014090
*     R E S E A R C H  C O M P U T I N G  C E N T E R                FT014100
      REM                                                            FT014110
*           I N D I A N A  U N I V E R S I T Y                       FT014120
      REM                                                            FT014130
*        B L O O M I N G T O N ,  I N D I A N A                      FT014140
      SPACE  3                                                       FT014150
*        *   *   *   *   *   *   *   *   *   *   *                   FT014160
      REM                                                            FT014170
*                       P A S S  1                                   FT014180
      REM                                                            FT014190
*        *   *   *   *   *   *   *   *   *   *   *                   FT014200
```

```
      EJECT                                                              FT014210
*     BRIEF DESCRIPTION OF PASS 1 STRUCTURE.                             FT014220
      SPACE  3                                                           FT014230
*     PASS 1 PERFORMS THE PRIMARY PASS OVER THE SOURCE DECK.             FT014240
*     THE SECTIONS OF PASS 1 ARE LISTED BELOW WITH A BRIEF STATEMENT     FT014250
*     OF THEIR FUNCTIONS.  MOST ROUTINES HAVE THEIR OWN DETAILED         FT014260
*     DESCRIPTIONS GIVEN IN THE BODY OF THE LISTING.                     FT014270
      REM                                                                FT014280
*     GETCH... READS AND LISTS SOURCE CARDS.  COMMENTS CARDS (C OR *)    FT014290
*     ARE DETECTED HERE AND INGORED AFTER LISTING.  OTHER CARDS ARE      FT014300
*     BURST WITH BLANKS DELETED FOR EASIER SCANNING.  THIS ROUTINE       FT014310
*     CONTROLS PAGE TITLING AND SUBTITLING FORMATS, AND DETECTS A        FT014320
*     POSSIBLE BINARY DECK CARD LABEL.                                   FT014330
      REM                                                                FT014340
*     GETLBL... SCANS THE LABEL FIELD OF  A CARD, DETECTING ANY          FT014350
*     COLUMN 1 CHARACTER AND ANY STATEMENT LABEL.                        FT014360
      REM                                                                FT014370
*     PROCXX... RESERVED WORD TEST AND DETERMINATION OF STATEMENT TYPE.  FT014380
*     (AN ARITHMETIC STATEMENT IS DETECTED BY DEFAULT, I.E. IT DOES      FT014390
*     NOT CONFORM TO A RESERVED TYPE.)                                   FT014400
      REM                                                                FT014410
*     PROCX1... TRANSACTS MISCELLANEOUS BUSINESS PRIOR TO ENTERING       FT014420
*     THE STATEMENT PROCESSOR DETERMINED IN PROCXX.  SEVERAL ERRORS      FT014430
*     ARE DETECTED HERE.  IF OBJECT CODE ON A PREVIOUS STATEMENT WAS     FT014440
*     DELAYED IN ORDER TO SEE THE CURRENT CARD LABEL, IT IS NOW PUT      FT014450
*     OUT.  THE STATEMENT LABEL (IF ANY) IS ENTERED INTO SYMTAB.         FT014460
      REM                                                                FT014470
*     SKEND... ENTERED FROM ALL STATEMENT PROCESSORS AT THE END OF       FT014480
*     THEIR PROCESSING.  AMONG OTHER MATTERS, ANY DO-STATEMENTS WHICH    FT014490
*     TERMINATE AT THE CURRENT STATEMENT WILL HAVE THEIR APPROPRIATE     FT014500
*     OBJECT CODE PRODUCED NOW.                                          FT014510
      REM                                                                FT014520
*     (STATEMENT PROCESSORS FOR ALL RESERVED WORD STATEMENTS).           FT014530
      REM                                                                FT014540
*     1ARITH... ARITHMETIC STATEMENT PROCESSING.  THIS PROCESSOR         FT014550
*     UTILIZES A LARGE NUMBER OF AUXILIARY SUBROUTINES.  SEE THEIR       FT014560
*     DESCRIPTIONS IN LATER PAGES.                                       FT014570
      REM                                                                FT014580
*     (PASS 1 UTLILTY PROGRAMS).                                         FT014590
      REM                                                                FT014600
```

```
*       GETCH ROUTINE...                                                    FT014620
        REM                                                                 FT014630
*       THIS ROUTINE READS SOURCE CARDS OFF THE MONITOR INPUT TAPE,         FT014640
*       LISTS THEM ON THE OUTPUT TAPE, AND FOR NON-COMMENTS, NON-*,         FT014650
*       OR NON-$ CARDS, BURSTS EACH STATEMENT ONE CHARACTER PER WORD.       FT014660
*       IF A STATEMENT CONSISTS OF MORE THAN ONE CARD, THE ENTIRE           FT014670
*       STATEMENT WILL BE BURST AT ONCE.  COLUMNS 73-80 OF THE FIRST        FT014680
*       CARD AND COLUMNS 1-6 AND 73-80 OF ALL SUCCEEDING CARDS WILL         FT014690
*       BE IGNORED.  NOTE THAT COLUMNS 1-6 (WITH COLUMN 6 BLANKED           FT014700
*       OUT) OF THE FIRST CARD OF A STATEMENT ARE CONSIDERED TO BE          FT014710
*       A PART OF THE STATEMENT.  TRAILING BLANKS WILL BE DELETED           FT014720
*       BEFORE THE CARD IS BURST.  THE BURST CARD IS STORED STARTING        FT014730
*       WITH 'COLUMN' AND GOING TOWARD HIGH CORE.  EACH CHARACTER           FT014740
*       OCCUPIES BITS P,1-5 OF A SEPARATE WORD IN COLUMN, WITH THE          FT014750
*       NEXT FIVE CHARACTERS OF THE STATEMENT FILLING OUT THE WORD.         FT014760
*       A MAXIMUM OF 666 CHARACGERS (10 CARDS + 6 LABEL CHARACTERS) IS      FT014770
*       ALLOWED.  AN OCTAL 77 IS ENTERED AS THE LAST CHARACTER ON           FT014780
*       THE CARD FOR USE BY THE SCANNER.                                    FT014790
        REM                                                                 FT014800
*       UPON ENTERING THE ROUTINE FOR THE FIRST TIME, THE TITLE FOR EACH    FT014810
*       PAGE OF THE OUTPUT LISTING WILL BE SET UP IN THE REGION             FT014820
*       TITLE (ALSO DATE AND PAGE COUNT).  IF THE TITLE CARD HAS            FT014830
*       $, *, OR C IN COLUMN 1, THE CARD IS NOT LISTED ON THE OUTPUT        FT014840
*       TAPE.  SUBTITLING IS PERMITTED BY MEANS OF A                        FT014850
*       CARD CONTAINING A $-SIGN IN COLUMN 1.  IN THIS CASE, COLUMNS        FT014860
*       2-72 OF $-SIGN CARDS WILL BE LISTED IN WORDS 4-15 OF THE            FT014870
*       SECOND LINE OF EACH PAGE.  $-SIGN CARDS CAUSE A PAGE EJECT          FT014880
*       EXCEPT AS THE SECOND CARD OF THE SOURCE DECK.  ALSO $-SIGN          FT014890
*       CARDS WILL NOT BE LISTED ON THE OUTPUT TAPE.  TO TURN OFF           FT014900
*       SUB-TITLING, USE A BLANK $-SIGN CARD.                               FT014910
        REM                                                                 FT014920
*       BINARY DECK LABELING IS EFFECTED BY THE USUAL FORTRAN RULES,        FT014930
*       I.E. IF THE FIRST NON-* CARD (OR NON-$ CARD IN FASTRAN) IS A        FT014940
*       COMMENTS CARD WITH AT LEAST ONE NON-BLANK CHARACTER IN COLUMNS      FT014950
*       2-7, THESE SIX CHARACTERS ARE USED AS THE BINARY DECK LABEL.  IF    FT014960
*       THIS CONDITION IS NOT MET, THE SUBROUTINE OR FUNCTION NAME          FT014970
*       IS USED, OR FOR A MAIN PROGRAM, THE CHARACTERS '(MAIN)'             FT014980
*       ARE USED (THIS LAST IS DIFFERENT FROM FORTRAN).                     FT014990
        SPACE   3                                                           FT015000
STARTT  NZT     ERHERE          DID ERROR OCCUR IN LAST STATEMENT           FT015010
        TRA     GETCH           NO                                          FT015020
        STZ     ERHERE          YES.  RESET FLAG CELL                       FT015030
        TSX     LIST,4          SKIP A LINE ON THE OUTPUT TAPE              FT015040
                BLANKS,,1                                                   FT015050
        REM                                                                 FT015060
GETCH   ZET     ENFILF          HAS AN END-OF-FILE BEEN ENCOUNTERED         FT015070
        ERROR   37,FINISH       NO END CARD FOUND.  ERROR                   FT015080
        STZ     GETR4           TURN OFF FLAG INDICATING NON-VOID STATEMENT FT015090
        REM                     UN CASE EOF IS ENCOUNTERED BEFORE NEXT      FT015100
        REM                     EXECUTABLE STATEMENT.                       FT015110
GETR    ZET     GETRFL                                                      FT015120
        TRA     GETR3           NO READ                                     FT015130
        REM                                                                 FT015140
IOS14   EQU     *                                                           FT015150
GETR5   READ    MINTAP,DEC,CARD,14                                          FT015160
GETR5   STL     IOEX                                                        FT015170
        BRA     READ,,*+3                                                   FT015180
        PZE     MINTAP,,DEC                                                 FT015190
        MZE     CARD,,14                                                    FT015200
```

```
END14   REM                                                          FT015210
        STL     GETRFL                                               FT015220
        TRA     GETR1                                                FT015230
GETR3   CAL     GETR2                                                FT015240
        STA     *+3                                                  FT015250
        AXT     0,4                                                  FT015260
        AXT     14,1                                                 FT015270
        CAL     **,4                                                 FT015280
        SLW     CARD+14,1                                            FT015290
        TXI     *+1,4,-1                                             FT015300
        TIX     *-3,1,1                                              FT015310
GETR1   TRA*    **          1 TIME TITLING SWITCH                    FT015320
        REM                 THIS SECTION SETS UP PAGE TITLING. 1ST   FT015330
        REM                 CARD OF THE SOURSE DECK WILL BE  USED AS A FT015340
        REM                 TITLE. IF IT ALSO HAPPENS TO BE A COMMENTS FT015350
        REM                 OR * CARD, IT WILL NOT GET LISTED.       FT015360
GET1    STL     LIS7            INITIALIZE FOR PAGE EJECT (LINE COUNT)FT015370
        STZ     LIS8            INITIALIZE PAGE COUNT TO ZERO        FT015380
        STZ     GET1B                                                FT015390
        CAL     CARD        BLANK OUT COLUMN 1 IF C, *, OR $         FT015400
        LGR     30          DO NOT LIST THESE CARDS IF TITLES.       FT015410
        LAS     =H00000C                                             FT015420
        TRA     *+2                                                  FT015430
        TSX     GET6+2,4    COMMENTS CARD. GO CHECK FOR LABELING     FT015440
        LAS     =H00000*                                             FT015450
        TRA     *+2                                                  FT015460
        TRA     GET1C           * CARD                               FT015470
        LAS     =H00000S    IS IT A SPACE STATEMENT.                 FT015480
        TRA     *+2         NO                                       FT015490
        TRA     GETR5       YES. IGNORE IT ON FIRST CARD.            FT015500
        LAS     =H00000E    IS IT EJECT.                             FT015510
        TRA     *+2         NO                                       FT015520
        TRA     GETR5       YES. IGNORE IT ON FIRST CARD             FT015530
        ERA     =H00000$    IS IT A SUB-TITLE CARD.                  FT015540
        TZE     GET1C       YES.  IGNORE SUB-TITLING AND USE AS TITLE FT015550
        STL     GET1B       NOT A COMMENTS, * OR $ CARD              FT015560
        STL     LABFLG      DECK LABELING NO LONGER POSSIBLE         FT015570
        REM                                                          FT015580
GET1A   AXT     12,4        MOVE FIRST CARD INTO TITLE BUFFER        FT015590
        CAL     CARD+12,4                                            FT015600
        SLW     TITLE+13,4                                           FT015610
        TIX     *-2,4,1                                              FT015620
        REM                                                          FT015630
        CAL     DATE8X      GET THE MONITOR DATE CELL                FT015640
        PAI                                                          FT015650
        LGR     24          SHIFT OUT ALL BUT THE MONTH              FT015660
        ANA     =077        KILL ALL BUT THE LAST CHARACTER           FT015670
        PAX     ,4                                                   FT015680
        LFT     10000       WAS THERE A TENS DIGIT.                  FT015690
        TXI     *+1,4,10    YES                                      FT015700
        CAL     MONTH,4     GET THE NAME OF THE MONTH                FT015710
        SLW     TITLE+15    PUT INTO THE TITLE LINE                  FT015720
        REM                                                          FT015730
        ZAC                                                          FT015740
        LGL     6           SHIFT IN THE TENS DIGIT OF THE DAY       FT015750
        TNZ     *+2         IS IT ZERO.                              FT015760
        CAL     BLANKS      YES                                      FT015770
        LGL     6           OTHER DIGIT                              FT015780
        ALS     6                                                    FT015790
        ORA     =H    00    INSERT TRAILING BLANK                    FT015800
```

```
         SLW     TITLE+14                                                FT015810
         REM                                                             FT015820
         XCL                     BRING IN THE YEAR                       FT015830
         ANA     =0777700000000                                          FT015840
         ORA     =H00                                                    FT015850
         SLW     TITLE+16        PUT YEAR INTO TITLE LINE                FT015860
         REM                                                             FT015870
GET1E    STL     GETR1             NO SUBTITLE. RESET 1 TIME TITLING SWITCH FT015880
         NOP     GET2                                                    FT015890
         ZET     GET1B                                                   FT015900
         TRA     GET2              LIST TITLING CARD                     FT015910
         TRA     GETR5                DO NOT LIST. RETURN FOR NEW CARD   FT015920
         REM     COMMENT, * OR $ CARDS                                   FT015930
GET1C    STZ     GET1B             TURN OFF LISTING OF THESE CARDS       FT015940
         CAL     CARD              BLANK OUT COLUMN 1                    FT015950
         ANA     =07777777777                                            FT015960
         ORA     =H 00000                                                FT015970
         SLW     CARD                                                    FT015980
         TRA     GET1A                                                   FT015990
         REM                                                             FT016000
GET2     CAL     CARD            CHECK FOR SUB-TITLING CARD              FT016010
         LGR     30                                                      FT016020
         LAS     =H000000S       IS IT SPACE.                            FT016030
         TRA     *+2             NO                                      FT016040
         TRA     GETSPC          YES. GO PROCESS THE SPACING             FT016050
         LAS     =H000000E       IS IT EJECT.                            FT016060
         TRA     *+2             NO                                      FT016070
         TRA     GETT2           YES.  GO SET FOR EJECT                  FT016080
         ERA     =H000000$                                               FT016090
         TZE     GETTTL          SUB-TITLE.  GO FIX UP THE TITLING       FT016100
         REM                                                             FT016110
         CAL     CARD+13         SAVE LOOK-AHEAD AND BLANK 'EM OUT       FT016120
         SLW     LASTLA          X                                       FT016130
         ANA     =0777700000000  X                                       FT016140
         ORA     =H00            X                                       FT016150
         SLW     CARD+13         X                                       FT016160
         TSX     LIST,4          LIST THIS SOURCE CARD                   FT016170
                 CARD,,14                                                FT016180
         CAL     CARD                                                    FT016190
         LGR     30                                                      FT016200
         LAS     =H000000C         IS IT COMMENTS                        FT016210
         TRA     *+2               NO                                    FT016220
         TSX     GET6,4          YES.  GO CHECK FOR DECK LABELING        FT016230
         LAS     =H000000*         IS IT *                               FT016240
         TRA     *+2               NO                                    FT016250
         TRA     GETR5           YES                                     FT016260
         STL     GETR4                                                   FT016270
         STL     LABFLG          DECK LABELING NO LONGER POSSIBLE        FT016280
         AXT     CARD+12,1                                               FT016290
         SXA     CARD1,1                                                 FT016300
         REM                     CHECK IS IT A CONTINUATION CARD         FT016310
         REM                                                             FT016320
IOS15    EQU     *                                                       FT016330
 GET2A   LREAD   MINTAP,DEC,ERASE,GETRFL   LOCATE NEXT SOURCE CARD       FT016340
GET2A    STL     IOEX                                                    FT016350
         BRA     LREAD,,*+3                                              FT016360
         PZE     MINTAP,,DEC                                             FT016370
         MZE     ERASE,,GETRFL                                           FT016380
END15    REM                                                             FT016390
         REM                                                             FT016400
```

```
         LXD     ERASE,4                                                  FT016410
         SXA     GETR2,4                                                  FT016420
         AXT     0,4                                                      FT016430
         CAL*    GETR2           IS COL. 6 NON-ZERO                       FT016440
         LGR     30                                                       FT016450
         LAS     =H00000C                                                 FT016460
         TRA     *+2                                                      FT016470
         TRA     GET3                                                     FT016480
         LAS     =H00000*                                                 FT016490
         TRA     *+2                                                      FT016500
         TRA     GET3                                                     FT016510
         LAS     =H00000$        IS THIS A SUBTITLE CARD.                 FT016520
         TRA     *+2             NO                                       FT016530
         TRA     GET3            YES.  STATEMENT IS NOW COMPLETE          FT016540
         LAS     =H00000S        IS IT SPACE.                             FT016550
         TRA     *+2             NO                                       FT016560
         TRA     GET3            YES.  TERMINATE THIS STATEMENT HERE.     FT016570
         LAS     =H00000E        IS IT EJECT.                             FT016580
         TRA     *+2                                                      FT016590
         TRA     GET3            YES.  TERMINATE THIS STATEMENT HERE      FT016600
         LGL     30                                                       FT016610
         ANA     =077                                                     FT016620
         TZE     GET3            NO. THEREFORE NOT A CONTINUATION CARD.   FT016630
         LAS     =H00000                                                  FT016640
         TRA     *+2                                                      FT016650
         TRA     GET4            COL 6 IS BLANK.  GO TEST FOR BLANK CARD  FT016660
         REM                     CONTINUATION CARD                        FT016670
         CAL     CARD1                                                    FT016680
         ADD     =11                                                      FT016690
         STA     CARD1                                                    FT016700
         AXT     11,1                                                     FT016710
         TXI     *+1,4,-1                                                 FT016720
GETR2    CAL     **,4                                                     FT016730
CARD1    SLW     **,1                                                     FT016740
         TIX     *-3,1,1                                                  FT016750
         AXC     13,4            SAVE LOOK-AHEAD AND BLANK 81-84          FT016760
         CAL*    GETR2           X                                        FT016770
         SLW     LASTLA          X                                        FT016780
         ANA     =0777700000000  X                                        FT016790
         ORA     =H00            X                                        FT016800
         SLW*    GETR2           X                                        FT016810
         CAL     GETR2                                                    FT016820
         STA     *+2                                                      FT016830
         TSX     LIST,4          LIST CARD JUST READ                      FT016840
                 **,,14                                                   FT016850
         TRA     GET2A           TRY AGAIN FOR A CONT. CARD               FT016860
         REM                                                              FT016870
GET4     CAL*    GETR2           TEST FOR COMPLETELY BLANK CARD           FT016880
         ERA     BLANKS                                                   FT016890
         TNZ     GET3            SKIP OUT IF NON-BLANK WORD.              FT016900
         TXI     *+1,4,-1                                                 FT016910
         TXH     *-4,4,-12       QUIT WHEN OUT OF WORDS                   FT016920
         TRA     CARD1+2         ALL BLANK.  GO LIST AND CONTINUE         FT016930
         SPACE   2                                                        FT016940
         REM                     CHECK IF WE CAN GET A DECK LABEL         FT016950
GET6     ZET     LABFLG          IS LABELING STILL POSSIBLE.              FT016960
         TRA*    3,4             NO                                       FT016970
         CAL     CARD                                                     FT016980
         LDQ     CARD+1                                                   FT016990
         LGL     7               COMPLETELY KILL THE 'C'                  FT017000
```

```
        LGR     1                                                       FT017010
        LAS     =H              ARE COLUMNS 2-7 BLANK.                  FT017020
        TRA     *+2             NO                                      FT017030
        CAL     =0              YES.  SET LABEL TO ZERO                 FT017040
        SLW     LABEL           ESTABLISH A LABEL                       FT017050
GET7    STL     LABFLG          NO MORE LABELING IS POSSIBLE FROM 'C' CARDSFT017060
        TRA*    3,4             RETURN                                  FT017070
        SPACE   2                                                       FT017080
        REM                     SET UP SUB-TITLING OPTION               FT017090
GETTTL  STL     SBTTL           SET THE SUBTITLING FLAG ON              FT017100
        AXT     12,4            WORDS ON THE CARD                       FT017110
        CAL     BLANKS          APPEND BLANK TO COLUMN 1                FT017120
        LGL     30              RESTORE THE WORD                        FT017130
        REM                                                             FT017140
GETT1   SLW     SUBTTL+15,4     MOVE CARD IMAGE INTO SUBTITLE BUFFER    FT017150
        ERA     BLANKS          AND NOTE ANY NON-BLANK WORD IN THE CARD FT017160
        TZE     *+2                                                     FT017170
        SLN     1               SET LIGHT IF NON-BLANK WORD             FT017180
        CAL     CARD+13,4       GET NEXT WORD                           FT017190
        TIX     GETT1,4,1                                               FT017200
        REM                                                             FT017210
        SLT     1               ANY NON-BLANK WORDS.                    FT017220
        STZ     SBTTL           NO.  SET THE SUBTITLING FLAG OFF        FT017230
GETT2   STL     LIS7            SET THE LINE COUNT FOR PAGE EJECT       FT017240
        TRA     GETR5           GO GET NEXT CARD.                       FT017250
        REM                                                             FT017260
GETSPC  CAL     CARD+1          PROCESS SPACING REQUIREMENTS            FT017270
        TSX     BCDFIX,4        GO GET THE COUNT                        FT017280
        ARS     18              INTO ADDRESS (IGNORE ERRORS IN THE COUNT) FT017290
        PAX     ,2                                                      FT017300
        ADD     LIS7            PINES SO FAR ON THIS PAGE               FT017310
        CAS     LPERPG          IS IT TOO MUCH.                         FT017320
        TRA     GETT2           YES.  EJECT                             FT017330
        TRA     GETT2                                                   FT017340
        TSX     LIST,4          SKIP LINES UNTIL THE COUNT IS EXHAUSTED FT017350
        PZE     BLANKS,,1                                               FT017360
        TIX     *-2,2,1                                                 FT017370
        TRA     GETR5           RETURN FOR ANOTHER CARD                 FT017380
        SPACE   2                                                       FT017390
        REM                                                             FT017400
        REM                     EXECUTED ONLY IF MODE ERROR OR END-OF-FILE FT017410
GETMER  STL     MODERR                                                  FT017420
        TRA     *+2                                                     FT017430
        REM                                                             FT017440
GETEOF  STL     ENFILF                                                  FT017450
        NZT     GETR4                                                   FT017460
        ERROR   37,FINISH       NO END CARD FOUND.  ERROR               FT017470
        REM                       CODE TO BURST STATEMENT               FT017480
GET3    LXA     CARD1,4           SCAN OFF TRAILING BLANKS              FT017490
        TIX     *+1,4,CARD                                              FT017500
        PXA     ,4                                                      FT017510
        PAC     ,4                                                      FT017520
        CAL     =H                                                      FT017530
GET3A   LAS     CARD-1,4                                                FT017540
        TRA     *+2                                                     FT017550
        TXI     *-2,4,1                                                 FT017560
        TXL     GETCH,4,0       BLANK CARD.  GO GET ANOTHER CARD        FT017570
        TXH     GET5,4,-2         LABEL ONLU.  IS AN ERROR              FT017580
        LDQ     =0767676767676                                          FT017590
        STQ     CARD,4                                                  FT017600
```

```
          PXA     ,4                                                    FT017610
          PAC     ,4                                                    FT017620
          TXI     **+1,4,1                                              FT017630
          PXA     ,4                                                    FT017640
          ACL     =1                                                    FT017650
          ACL     CARD2                                                 FT017660
          STA     GET3B                                                 FT017670
          AXT     0,2                                                   FT017680
          TOV     **+1                                                  FT017690
          CAL     =07                                                   FT017700
          LDQ     GET3B5                                                FT017710
          STQ     GET3B4                                                FT017720
          SXA     GET3B2,4                                              FT017730
GET3B     LDQ     **,4                                                  FT017740
          AXT     6,4                                                   FT017750
GET3B3    CRQ     SCANTT,1,1                                            FT017760
          TXH     GET3B2-1,1,62                                         FT017770
          ALS     6                                                     FT017780
          ORA     SCANTC,1                                              FT017790
GET3B4    ***     **,,**                                                FT017800
          TXI     **+1,2,-1                                             FT017810
          TIX     GET3B3,4,1                                            FT017820
GET3B2    AXT     **,4                                                  FT017830
          TIX     GET3B-1,4,1                                           FT017840
          TRA     GETCHX          RESERVED WORD TEST                    FT017850
          REM                                                           FT017860
GET3BO    TNO     GET3B4+2                                              FT017870
          LDI     GET3B6                                                FT017880
          STI     GET3B4                                                FT017890
          TRA     GET3B4                                                FT017900
          REM                                                           FT017910
GET3B5    TRA     GET3BO                                                FT017920
GET3B6    SLW     COLUMN,2                                              FT017930
          REM                                                           FT017940
SCANTT    DUP     1,48                                                  FT017950
                  *-SCANTT                                              FT017960
          PZE     63                                                    FT017970
          DUP     1,15                                                  FT017980
                  *-SCANTT                                              FT017990
          REM                                                           FT018000
          PZE     63                                                    FT018010
          PZE     63                                                    FT018020
          DUP     1,62                                                  FT018030
                  SCANTC-*                                              FT018040
SCANTC    EQU     *-1                                                   FT018050
          REM                                                           FT018060
GET5      ERROR   83,GETR                                               FT018070
CARD2     PZE     CARD                                                  FT018080
TITLE     BCI     1,1                                                   FT018090
          BSS     12                                                    FT018100
BLANK     BCI     1,                                                    FT018110
BLANKS    EQU     BLANK                                                 FT018120
          BSS     3                                                     FT018130
          BCI     1, PAGE                                               FT018140
          BSS     1                                                     FT018150
          REM                                                           FT018160
SUBTTL    BCI     3,                                                    FT018170
          BSS     12                                                    FT018180
          REM                                                           FT018190
GETCHX    EQU     *                                                     FT018200
```

```
*       GETLBL ROUTINE...                                                   FT018220
        REM                                                                 FT018230
*       ROUTINE TO SCAN OFF THE LABEL IN COLUMNS 1-5.  ROUTINE ALSO         FT018240
*       LOOKS AT COLUMN 1 FOR MODE DESIGNATIONS F, B, D, I.                 FT018250
*       ROUTINE SETS CELL MODFLG TO 1, 2, 3, 4 ACCORDING AS COLUMN 1        FT018260
*       IS B, D, I, F, RESPECTIVELY.  IF COLUMN 1 IS NOT ANY OF THE         FT018270
*       ABOVE, MODFLG IS ZEROED OUT.                                        FT018280
        REM                                                                 FT018290
*       THE PROCESSED LABEL WILL APPEAR IN THE CELL 'CARD' AT THE END       FT018300
*       SCAN.  BLANKS AND LEADING ZEROES WILL BE DROPPED.                   FT018310
*       THE LABEL IS NOT PUT INTO SYMTAB YET.                               FT018320
        SPACE   3                                                           FT018330
GETLBL  CAL     CARD                                                        FT018340
        ARS     30                                                          FT018350
        AXT     0,1                                                         FT018360
        LAS     =060                                                        FT018370
        TRA     *+2                                                         FT018380
        TRA     LABL1           BLANK IN COL 1                              FT018390
        LAS     =012                                                        FT018400
        TRA     *+3                                                         FT018410
        TRA     *+2                                                         FT018420
        TRA     LABL1           COL 1 IS NUMERIC                            FT018430
        LAS     =H00000F                                                    FT018440
        TRA     *+2                                                         FT018450
        TXI     LABL4,1,4       COL 1 IS F                                  FT018460
        LAS     =H00000B                                                    FT018470
        TRA     *+2                                                         FT018480
        TXI     LABL4,1,1       COL 1 IS B                                  FT018490
        LAS     =H00000D                                                    FT018500
        TRA     *+2                                                         FT018510
        TXI     LABL4,1,2       COL 1 IS D                                  FT018520
        LAS     =H00000I                                                    FT018530
        ERROR   80,SKEND        INVALID CHARACTER OR LETTER IN COL 1        FT018540
        TXI     *+2,1,3         COL 1 IS I                                  FT018550
        ERROR   80,SKEND                                                    FT018560
LABL4   CAL     =07777777777                                                FT018570
        ANA     CARD                                                        FT018580
        ORA     =H 00000                                                    FT018590
        SLW     CARD                                                        FT018600
        REM                                                                 FT018610
LABL1   CAL     CARD            GET LABEL IMAGE                             FT018620
        STZ     CARD            ZERO OUT LABEL CELL                         FT018630
        ARS     6               KILL COLUMN 6                               FT018640
        REM                                                                 FT018650
        LAS     =H0             IS LABEL BLANK.                             FT018660
        TRA     *+2             NO.                                         FT018670
        TRA     GETLBX-1        YES.                                        FT018680
        REM                                                                 FT018690
        LGR     30              NO. PUT IN MQ                               FT018700
        AXC     36,4            SET TERMINAL SHIFT                          FT018710
        AXT     5,2             NUMBER OF POSITIONS TO TEST                 FT018720
        REM                                                                 FT018730
LABL8   ZAC                                                                 FT018740
        LGL     6               SHIFT IN CHARACTER                          FT018750
        LAS     =060            IS IT BLANK.                                FT018760
        TRA     *+4             NO.  BAD CHARACTER                          FT018770
        TRA     LABL2           YES.  SKIP.                                 FT018780
        REM                                                                 FT018790
        LAS     =012            IS IT NIN-NUMERIC.                          FT018800
```

```
         TRA     *+1                YES.                                          FT018810
LABL3    ERROR   1,SKEND            YES                                           FT018820
         REM                                                                      FT018830
         NZT     CARD               NO.  NUMERIC.  IS LABEL CELL STILL ZERO.     FT018840
         TZE     LABL2              YES.  SKIP IF LEADING ZERO.                   FT018850
         REM                                                                      FT018860
         ORA     CARD               MERGE OLD LABEL WITH NEW DIGIT                FT018870
         ALS     6                  ADJUST FOR NEXT DIGIT                         FT018880
         SLW     CARD               RESET LABEL CELL                              FT018890
         TXI     *+1,4,6            REDUCE TERMINAL SHIFT COUNT                   FT018900
         REM                                                                      FT018910
LABL2    TIX     LABL8,2,1          REDUCE DIGIT COUNT                            FT018920
         REM                                                                      FT018930
         CAL     CARD               GET LABEL                                     FT018940
         TZE     LABL3              ZERO LABEL NOT ALLOWED                        FT018950
         ARS     6                  JUSTIFY                                       FT018960
         LDQ     =H                 TERMINAL BLANKS                               FT018970
         LGL     0,4                LEFT JUSTIFIED LABEL IN ACCUM.                FT018980
         REM                                                                      FT018990
         SLW     CARD               RESTORE LABEL                                 FT019000
         SXA     MODFLG,1           SAVE MODE INDICATION                          FT019010
GETLBX   EQU     *                                                                FT019020
```

```
*       RESERVED WORD TEST AND DETERMINATION OF STATEMENT TYPE.                 FT019040
        REM                                                                     FT019050
*       CONTROL PASSES TO THIS ROUTINE AFTER ANY LABEL AND/OR COLUMN            FT019060
*       1 MODE HAS BEEN SCANNED.                                                FT019070
        REM                                                                     FT019080
*       IF THERE IS AN F IN COLUMN 1, AN IMMEDIATE EXIT IS TAKEN TO THE         FT019090
*       F-CARD PROCESSOR.  IF NOT, THE FIRST CHARACTER IN THE FIRST             FT019100
*       WORD OF 'COLUMN' IS USED TO BRANCH ON A TABLE OF TRANSFERS.             FT019110
*       EACH INITIAL LETTER WHICH MAY BEGIN A RESERVED WORD HAS ITS             FT019120
*       OWN SUBSEQUENT TEST SECTION.  THERE, IT IS DETERMINED IF                FT019130
*       THIS STATEMENT BEGINS WITH A RESERVED WORD.  IF IT DOES NOT,            FT019140
*       THEN THE STATEMENT IS AN ARITHMETIC SUBSTITUTION STATEMENT.             FT019150
        REM                                                                     FT019160
*       IN ANY EVENT, THE CELL PROCSW IS FORMED, TELLING THE                    FT019170
*       ENTRY POINT TO THE CORRECT STATEMENT PROCESSOR, AND TELLING             FT019180
*       IF THIS IS AN EXECUTABLE, TRANSFER, DO, CALL, AND/OR END-TYPE           FT019190
*       STATEMENT.  CONTROL THEN PASSES TO THE PASS 1 DRIVER, WHERE             FT019200
*       SOME INITIAL BUSINESS IS HANDLED PRIOR TO PROCESSING THE                FT019210
*       STATEMENT.                                                              FT019220
        SPACE   3                                                               FT019230
PROCXX  LXA     MODFLG,4        GET THE MODE FLAG FOR THIS STATEMENT            FT019240
        TXH     PROCX2,4,3      IS IT AN F-CARD.                                FT019250
        REM                                                                     FT019260
        AXT     0,2                                                             FT019270
        CAL     COLUMN                                                          FT019280
        ARS     30                                                              FT019290
        PAC     ,4              KEY WORD TEST                                   FT019300
        TRA     *+1,4                                                           FT019310
        REM                     KEY WORD FIRST LETTER TABLE                     FT019320
        DUP     1,17                                                            FT019330
        STR     FORM11,,SKEND                                                   FT019340
        TRA     TA1.0           A   21                                          FT019350
        TRA     TB2.0           B   22                                          FT019360
        TRA     TC3.0           C   23                                          FT019370
        TRA     TD4.0           D   24                                          FT019380
        TRA     TE5.0           E   25                                          FT019390
        TRA     TF6.0           F   26                                          FT019400
        TRA     TG7.0           G   27                                          FT019410
        TRA     PARITH          H   30                                          FT019420
        TRA     TI8.0           I   31                                          FT019430
        DUP     1,7                                                             FT019440
        STR     FORM11,,SKEND                                                   FT019450
        TRA     PARITH          J   41                                          FT019460
        TRA     PARITH          K   42                                          FT019470
        TRA     TL13.0          L   43                                          FT019480
        TRA     PARITH          M   44                                          FT019490
        TRA     PARITH          N   45                                          FT019500
        TRA     TO15.0          O   46                                          FT019510
        TRA     TP9.0           P   47                                          FT019520
        TRA     PARITH          Q   50                                          FT019530
        TRA     TR10.0          R   51                                          FT019540
        DUP     1,8                                                             FT019550
        STR     FORM11,,SKEND                                                   FT019560
        TRA     TS11.0          S   62                                          FT019570
        TRA     PARITH          T   63                                          FT019580
        TRA     PARITH          U   64                                          FT019590
        TRA     PARITH          V   65                                          FT019600
        TRA     TW12.0          W   66                                          FT019610
        TRA     PARITH          X   67                                          FT019620
```

```
TRA     PARITH          Y   70                                  FT019630
TRA     PARITH          Z   71                                  FT019640
DUP     1,6                                                     FT019650
STR     FORM11,,SKEND                                           FT019660
```

```
         EJECT                                                          FT019670
*        ROUTINE TO CHECK IF A POSSIBLE IF, FORMAT, END, OR CALL        FT019680
*        STATEMENT IS REALLY AN ARITHMETIC STATEMENT (I.E. HAS A        FT019690
*        VALID EQUAL SIGN FOLLOWING A POSSIBLE SUBSCRIPT EXPRESSION).   FT019700
         REM                                                            FT019710
CHKSUB   SXA      EXCKSB,4                                              FT019720
         LDI      2                PREPARE TO LATER RESTORE THE ERROR TRAP FT019730
         AXT      CHKERR,4         ALTER THE ERROR TRAP CELL TO RETURN TO FT019740
         SXA      2,4              THIS ROUTINE INSTEAD OF THE ERROR ROUTINE. FT019750
         SXA      SCNB5,4          ALTER A CELL IN SCNBCD TO GIVE ERROR RETURNFT019760
         TSX      SCNBCD,4         NOW SCAN OFF ANY VALID CONSTANT OR     FT019770
         REM                       SYMBOL, TO AVOIDA POSSIBLE HOLLERITH   FT019780
         REM                       LITERAL, WHICH WILL BE CAUGHT AS AN ERROR. FT019790
         SKIP     RPCHAR           IS THE BREAK CHARACTER A RIGHT PAREN.  FT019800
         TXI      *-2,2,-1         NO.  RETURN FOR ANOTHER SCAN           FT019810
         REM                                                             FT019820
         AXC      PARITH-1,1       GET A POSSIBLE TRANSFER POINT          FT019830
         SKIP1    EQCHAR           IS THE NEXT CHARACTER AN EQUAL SIGN.   FT019840
         SXA      EXCKSB,1         SET THE PROPER TRANSFER POINT          FT019850
         REM                                                             FT019860
CHKERR   STI      2                RESTORE THE ERROR TRAP CELL            FT019870
         AXT      SCNB6,4          RESTORE THE ORIGINAL ADDRESS IN SCNBCD CELLFT019880
         SXA      SCNB5,4                                                FT019890
         REM                                                             FT019900
EXCKSB   AXT      **,4             NO.  RESTORE XR4 AND RETURN            FT019910
         TRA      1,4              PROCESSOR HAS BEEN FOUND               FT019920
```

```
        EJECT                                                          FT019930
PROCX2  REMARK  (F-CARD PROCESSOR)                                     FT019940
        SPRSW   1FCARD,NOXEQ,NOTRA,NODO                                FT019950
```

```
      EJECT                                                          FT019960
TA1.0 CAL     =HASSIGN                                               FT019970
      ERA     COLUMN,2                                               FT019980
      TNZ     PARITH                                                 FT019990
      TSX     BREAK,4       IS NEXT BREAK END-OF-STATEMENT           FT020000
      ERA     =077                                                   FT020010
      TNZ     PARITH        NO                                       FT020020
      REMARK  (ASSIGN STATEMENT PROCESSOR)                           FT020030
      SPRSW   1ASSNX,XEQ,NOTRA,NODO                                  FT020040
```

```
        EJECT                                                       FT020050
TB2.0   CAL     =HBACKSP        IS IT BACKSPACE                     FT020060
        ERA     COLUMN,2                                            FT020070
        TNZ     PARITH          NO                                  FT020080
        CAL     =HACE                                               FT020090
        ERA     COLUMN+6,2                                          FT020100
        ANA     =0777777000000                                      FT020110
        TNZ     PARITH          NO AGAIN                            FT020120
        REMARK  (BACKSPACE STATEMENT PROCESSOR)                     FT020130
        SPRSW   1BSRXX,XEQ,NOTRA,NODO                               FT020140
```

```
        EJECT                                                       FT020150
T03.0   CAL     =HCALL                                              FT020160
        ERA     COLUMN,2                                            FT020170
        ANA     =0777777770000                                      FT020180
        TNZ     T3.1                                                FT020190
        TXI     *+1,2,-4                                            FT020200
        TSX     BREAK,4                                             FT020210
        LAS     =077            IS THIS END OF STATEMENT.           FT020220
        TRA     *+2             NO                                  FT020230
        TRA     T3.3            YES                                 FT020240
        ERA     =H00000(        IS IT LEFT PAREN.                   FT020250
        TNZ     PARITH          NO.  ARITHMETIC STATEMENT           FT020260
        REM                                                         FT020270
        TXL     T3.3,2,-9       IF NAME BIGGER THAN 7 CHAR, THIS IS CALL FT020280
        LXD     FIRSTF,4        GET NUMBER OF EXECUTABLE STATEMENTS SO FAR FT020290
        TXL     *+2,4,0         SKIP IF NONE SO FAR                 FT020300
        TXL     T3.3,2,-8       NOW IT IS CALL IF NAME IS BIGGER THAN 6 CH.FT020310
        TSX     CHKSUB,4        OH, DEAR.  CHECK IF THIS IS ARITHMETIC FT020320
T3.3    REMARK  (CALL STATEMENT PROCESSOR)                          FT020330
        SPRSW   1CALLX,XEQ,NOTRA,NODO,BCALL                         FT020340
T3.1    CAL     =HCOMMON        IS IT COMMON                        FT020350
        ERA     COLUMN,2                                            FT020360
        TNZ     T3.2            NO                                  FT020370
        TXI     *+1,2,-6                                            FT020380
        TSX     BREAK,4                                             FT020390
        LAS     =077            TEST FOR EOS                        FT020400
        MACER                   MACHINE ERROR                       FT020410
        TRA     *+3                                                 FT020420
        ERA     =H00000,                                            FT020430
        TNZ     PARITH          NO                                  FT020440
        REMARK  (COMMON STATEMENT PROCESSOR)                        FT020450
        SPRSW   1COMNX,NOXEQ,NOTRA,NODO                             FT020460
T3.2    CAL     =HCONTIN        IS IT CONTINUE                      FT020470
        ERA     COLUMN,2                                            FT020480
        TNZ     PARITH          NO                                  FT020490
        CAL     =HUE                                                FT020500
        ERA     COLUMN+6,2                                          FT020510
        ANA     =0777700000000                                      FT020520
        TNZ     PARITH                                              FT020530
        REMARK  (CONTINUE STATEMENT PROCESSOR)                      FT020540
        SPRSW   1CONTX,XEQ,NOTRA,NODO                               FT020550
```

```
        EJECT                                                   FT020560
T04.0   CAL     =HDIMENS        IS IT DIMENSION                 FT020570
        ERA     COLUMN,2                                        FT020580
        TNZ     T4.1                                            FT020590
        CAL     =HION                                           FT020600
        ERA     COLUMN+6,2                                      FT020610
        ANA     =0777777000000                                  FT020620
        TNZ     PARITH                                          FT020630
        REMARK  (DIMENSION STATEMENT PROCESSOR)                 FT020640
        SPRSW   1DIMNX,NOXEQ,NOTRA,NODO                         FT020650
T4.1    CAL     =HDO                                            FT020660
        ERA     COLUMN,2                                        FT020670
        ANA     =0777700000000                                  FT020680
        TNZ     PARITH                                          FT020690
        TXI     *+1,2,-2                                        FT020700
        TSX     BREAK,4                                         FT020710
        ERA     =H00000=                                        FT020720
        TNZ     PARITH                                          FT020730
        TSX     BREAK,4                                         FT020740
        ERA     =H00000,                                        FT020750
        TNZ     PARITH          NOT A DO                        FT020760
        REMARK  (DO STATEMENT PROCESSING)                       FT020770
        SPRSW   PDO,XEQ,NOTRA,DO                                FT020780
```

```
        EJECT                                                           FT020790
T5.1    CAL     =HEND           IS IT END                               FT020800
        ERA     COLUMN,2                                                FT020810
        ANA     =0777777000000                                          FT020820
        TNZ     T5.2                                                    FT020830
        AXC     3,2                                                     FT020840
        SKIP    63              USUALLY THIS IS END OF STATEMENT        FT020850
        TRA     T5.4            AHA.  NOT SO THIS TIME...               FT020860
TE5.3   REMARK  (END STATEMENT PROCESSOR)                               FT020870
        SPRSW   PEND,NOXEQ,NOTRA,NODO,BEND                              FT020880
T5.4    SKIP    LPCHAR          IS THERE A LEFT PAREN.                  FT020890
        TRA     PARITH          NO.  ARITHMETIC STATEMENT               FT020900
        TSX     CHKSUB,4        YES.  GO CHECK IF THIS IS ARITHMETIC    FT020910
        TRA     TE5.3           OH, HO...  THIS IS ONE OF THOSE RIDICULOUS FT020920
        REM                     END STATEMENTS WITH THE MONITOR OPTIONS.FT020930
        REM                     THE OPTIONS ARE, OF COURSE, IGNORED.    FT020940
TE5.0   CAL     =HENDFIL                                                FT020950
        ERA     COLUMN,2        IS IT ENDFILE                           FT020960
        TNZ     T5.1                                                    FT020970
        CAL     =HE                                                     FT020980
        ERA     COLUMN+6,2                                              FT020990
        ANA     =0770000000000                                          FT021000
        TNZ     PARITH          NO                                      FT021010
        REMARK  (END FILE STATEMENT PROCESSOR)                          FT021020
        SPRSW   1EOFXX,XEQ,NOTRA,NODO                                   FT021030
T5.2    CAL     =HEQUIVA        IS IT EQUIVALENCE                       FT021040
        ERA     COLUMN,2                                                FT021050
        TNZ     PARITH                                                  FT021060
        CAL     =HLENCE                                                 FT021070
        ERA     COLUMN+6,2                                              FT021080
        ANA     =0777777777700                                          FT021090
        TNZ     PARITH                                                  FT021100
        REMARK  (EQUIVALENCE STATEMENT PROCESSOR)                       FT021110
        SPRSW   1EQUVX,NOXEQ,NOTRA,NODO                                 FT021120
```

```
        EJECT                                                                   FT021130
TF6.0   CAL     =HFORMAT                                                        FT021140
        ERA     COLUMN,2                                                        FT021150
        TNZ     T6.1                                                            FT021160
        AXC     6,2                                                             FT021170
        SKIP    LPCHAR          IS NEXT CHARACTER A LEFT PAREN.                 FT021180
        TRA     PARITH          NO                                              FT021190
        ZET     FORFLG          HAS ARRAY NAMED 'FORMAT' BEEN ENCOUNTERED.      FT021200
        TSX     CHKSUB,4        YES.  GO CHECK IF THIS IS ARITHMETIC STATE.      FT021210
        REMARK  (FORMAT STATEMENT PROCESSING)                                   FT021220
        SPRSW   PFMT,NOXEQ,NOTRA,NODO                                           FT021230
T6.1    CAL     =HFREQUE                                                        FT021240
        ERA     COLUMN,2                                                        FT021250
        TNZ     T6.2                                                            FT021260
        CAL     =HNCY                                                           FT021270
        ERA     COLUMN+6,2                                                      FT021280
        ANA     =0777777000000                                                  FT021290
        TNZ     PARITH                                                          FT021300
        REMARK  (FREQUENCY STATEMENT PROCESSING)                                FT021310
        SPRSW   1FREQX,NOXEQ,NOTRA,NODO                                         FT021320
T6.2    CAL     =HFUNCTI        IS IT FUNCTION                                  FT021330
        ERA     COLUMN,2                                                        FT021340
        TNZ     TF16.0                                                          FT021350
        CAL     =HON                                                            FT021360
        ERA     COLUMN+6,2                                                      FT021370
        ANA     =0777700000000                                                  FT021380
        TNZ     PARITH                                                          FT021390
        REMARK  (FUNCTION STATEMENT PROCESSING)                                 FT021400
        SPRSW   PFUNCT,NOXEQ,NOTRA,NODO                                         FT021410
```

```
        EJECT                                                                   FT021420
IG7.0   CAL     =HGOTO                                                          FT021430
        ERA     COLUMN,2        IS IT GOTO                                      FT021440
        ANA     =0777777770000                                                  FT021450
        TNZ     PARITH          NO                                              FT021460
        TXI     *+1,2,-4                                                        FT021470
        TSX     BREAK,4                                                         FT021480
        LAS     =077                                                            FT021490
        TRA     *+2                                                             FT021500
        TRA     PGOTO           TO GOTO N PROCESSOR                             FT021510
        LAS     =H00000,                                                        FT021520
        TRA     *+2                                                             FT021530
        TRA     PGOTOC            TO ASSIGNED GO TO                             FT021540
        ERA     =H00000(                                                        FT021550
        TNZ     PARITH                                                          FT021560
        TSX     BREAK,4                                                         FT021570
        LAS     =H00000,                                                        FT021580
        TRA     *+2                                                             FT021590
        TRA     *-3                                                             FT021600
        ERA     =H00000)                                                        FT021610
        TZE     *+2                                                             FT021620
        ERROR   50,SKEND                                                        FT021630
        TSX     BREAK,4                                                         FT021640
        ERA     =H00000,                                                        FT021650
        TNZ     PARITH                                                          FT021660
        REMARK  (GO TO (...),N STATEMENT PROCESSOR)                             FT021670
        SPRSW   1COMGO,XEQ,TRANS,NODO                                           FT021680
PGOTO   REMARK  (GO TO STATEMENT PROCESSOR)                                     FT021690
        SPRSW   1GOTOX,XEQ,TRANS,NODO                                           FT021700
PGOTOC  REMARK  (ASSIGNED GO TO STATEMENT PROCESSOR)                            FT021710
        SPRSW   1AGOTO,XEQ,TRANS,NODO                                           FT021720
```

```
        EJECT                                                                   FT021730
T18.0   CAL     =HIF(SEN                                                        FT021740
        ERA     COLUMN,2                                                        FT021750
        TNZ     T8.2                                                            FT021760
        CAL     =HSELIGH                                                        FT021770
        ERA     COLUMN+6,2                                                      FT021780
        TNZ     T8.1                                                            FT021790
        CAL     =HT                                                             FT021800
        ERA     COLUMN+12,2                                                     FT021810
        ANA     =0770000000000                                                  FT021820
        TZE     PIFSL           TO IS(SENSE LIGHT) PROCESSOR                    FT021830
        ERROR   15,SKEND        SYMBOL TOO LONG                                 FT021840
        REM                                                                     FT021850
T8.1    CAL     =HSESWIT                                                        FT021860
        ERA     COLUMN+6,2                                                      FT021870
        TNZ     PARITH                                                          FT021880
        CAL     =HCH                                                            FT021890
        ERA     COLUMN+12,2                                                     FT021900
        ANA     =0777700000000                                                  FT021910
        TZE     PIFSW           TO IF (SENSE SWITCH) PROCESSOR                  FT021920
        ERROR   15,SKEND        SYMBOL TOO LONG                                 FT021930
        REM                                                                     FT021940
T8.2    CAL     =HIF(                                                           FT021950
        ERA     COLUMN,2                                                        FT021960
        ANA     =0777777000000                                                  FT021970
        TNZ     T8.3                                                            FT021980
        AXC     3,2             SET THE COLUMN POINTER                          FT021990
        ZET     IFFLAG          HAS AN ARRAY NAMED 'IF' BEEN ENCOUJTERED.       FT022000
        TSX     CHKSUB,4        YES.  GO CHECK IF THIS IS AN ARITH. STATE.      FT022010
        REMARK  (IF STATEMENT PROCESSOR)                                        FT022020
        SPRSW   1IFXXX,XEQ,TRANS,NODO                                           FT022030
        REM                                                                     FT022040
T8.3    CAL     =HIFACCU                                                        FT022050
        ERA     COLUMN,2                                                        FT022060
        TNZ     T8.4                                                            FT022070
        CAL     =HMULATO                                                        FT022080
        ERA     COLUMN+6,2                                                      FT022090
        TNZ     PARITH                                                          FT022100
        CAL     =HROVERF                                                        FT022110
        ERA     COLUMN+12,2                                                     FT022120
        TNZ     ERRO15          SYMBOL TOO LONG                                 FT022130
        CAL     =HLOW                                                           FT022140
        ERA     COLUMN+18,2                                                     FT022150
        ANA     =0777777000000                                                  FT022160
        TZE     PIFAC           TO IF ACCUMULATOR OVERFLOW PROCESSOR            FT022170
        ERROR   15,SKEND        SYMBOL TOO LONG                                 FT022180
T8.4    CAL     =HIFQUOT                                                        FT022190
        ERA     COLUMN,2                                                        FT022200
        TNZ     T8.5                                                            FT022210
        CAL     =HIENTOV                                                        FT022220
        ERA     COLUMN+6,2                                                      FT022230
        TNZ     PARITH                                                          FT022240
        CAL     =HERFLOW                                                        FT022250
        ERA     COLUMN+12,2                                                     FT022260
        TZE     PIFMQ           TO IF QUOTIENT OVERFLOW PROCESSOR               FT022270
        ERROR   15,SKEND        SYMBOL TOO LONG                                 FT022280
T8.5    CAL     =HIFDIVI                                                        FT022290
        ERA     COLUMN,2                                                        FT022300
        TNZ     PARITH                                                          FT022310
        CAL     =HDECHEC                                                        FT022320
```

```
                ERA     COLUMN+6,2                                      FT022330
                TNZ     PARITH                                          FT022340
                CAL     =HK                                             FT022350
                ERA     COLUMN+12,2                                     FT022360
                ANA     =0770000000000                                  FT022370
                TZE     *+2                                             FT022380
T8.5A           ERROR   15,SKEND        SYMBOL TOO LONG                 FT022390
                REMARK  (IF DIVIDE CHECK STATEMENT PROCESSOR)           FT022400
                SPRSW   1IFDIV,XEQ,TRANS,NODO                           FT022410
PIFSL           REMARK  (IF (SENSE LIGHT) STATEMENT PROCESSOR)          FT022420
                SPRSW   1IFSLT,XEQ,TRANS,NODO                           FT022430
PIFSW           REMARK  (IF(SENSE SWITCH) STATEMENT PROCESSOR)          FT022440
                SPRSW   1IFSWT,XEQ,TRANS,NODO                           FT022450
PIFAC           REMARK  (IF ACCUMULATOR OVERFLOW STATEMENT PROCESSOR)   FT022460
                SPRSW   1IFACC,XEQ,TRANS,NODO                           FT022470
PIFMQ           REMARK  (IF QUOTIENT OVERFLOW STATEMENT PROCESSOR)      FT022480
                SPRSW   1IFQUO,XEQ,TRANS,NODO                           FT022490
```

```
        EJECT                                                   FT022500
TP9.0   CAL     =HPAUSE                                         FT022510
        ERA     COLUMN,2                                        FT022520
        ANA     =0777777777700                                  FT022530
        TNZ     T9.1                                            FT022540
        TXI     *+1,2,-5                                        FT022550
        TSX     BREAK,4                                         FT022560
        ERA     =077                                            FT022570
        TNZ     PARITH                                          FT022580
        REMARK  (PAUSE STATEMENT PROCESSING)                    FT022590
        SPRSW   1PAUSE,XEQ,NOTRA,NODO                           FT022600
T9.1    CAL     =HPRINT                                         FT022610
        ERA     COLUMN,2                                        FT022620
        ANA     =0777777777700                                  FT022630
        TNZ     T9.2                                            FT022640
        TXI     *+1,2,-5                                        FT022650
        TSX     BREAK,4                                         FT022660
        LAS     =077                                            FT022670
        MACER                   MACHINE ERROR                   FT022680
        TRA     *+3                                             FT022690
        ERA     =H00000,                                        FT022700
        TNZ     PARITH                                          FT022710
        REMARK  (PRINT STATEMENT PROCESSOR)                     FT022720
        SPRSW   1PRINT,XEQ,NOTRA,NODO                           FT022730
T9.2    CAL     =HPUNCH                                         FT022740
        ERA     COLUMN,2                                        FT022750
        ANA     =0777777777700                                  FT022760
        TNZ     PARITH                                          FT022770
        TXI     *+1,2,-5                                        FT022780
        TSX     BREAK,4                                         FT022790
        LAS     =077                                            FT022800
        MACER                   MACHINE ERROR                   FT022810
        TRA     *+3                                             FT022820
        ERA     =H00000,                                        FT022830
        TNZ     PARITH                                          FT022840
        REMARK  (PUNCH STATENENT PROCESSOR)                     FT022850
        SPRSW   1PUNCH,XEQ,NOTRA,NODO                           FT022860
```

```
        EJECT                                                                   FT022870
T10.0   CAL     =HREAD                                                          FT022880
        ERA     COLUMN,2                                                        FT022890
        ANA     =0777777770000                                                  FT022900
        TNZ     T10.4                                                           FT022910
        CAL     =HREADIN                                                        FT022920
        ERA     COLUMN,2                                                        FT022930
        TNZ     T10.2                                                           FT022940
        CAL     =HPUTTAP                                                        FT022950
        ERA     COLUMN+6,2                                                      FT022960
        TNZ     PARITH                                                          FT022970
        CAL     =HE                                                             FT022980
        ERA     COLUMN+12,2                                                     FT022990
        ANA     =0770000000000                                                  FT023000
        TNZ     T8.5A           SYMBOL TOO LONG                                 FT023010
        REMARK  (READ INPUT TAPE STATEMENT PROCESSOR)                           FT023020
        SPRSW   1RITXX,XEQ,NOTRA,NODO                                           FT023030
T10.2   CAL     =HREADTA                                                        FT023040
        ERA     COLUMN,2                                                        FT023050
        TNZ     T10.3                                                           FT023060
        CAL     =HPE                                                            FT023070
        ERA     COLUMN+6,2                                                      FT023080
        ANA     =0777700000000                                                  FT023090
        TNZ     PARITH                                                          FT023100
        REMARK  (READ TAPE STATEMENT PROCESSOR)                                 FT023110
        SPRSW   1RDTPX,XEQ,NOTRA,NODO                                           FT023120
T10.3   CAL     =HREADDR                                                        FT023130
        ERA     COLUMN,2                                                        FT023140
        TNZ     T10.1                                                           FT023150
        CAL     =HUM                                                            FT023160
        ERA     COLUMN+6,2                                                      FT023170
        ANA     =0777700000000                                                  FT023180
        TNZ     PARITH                                                          FT023190
        REMARK  (READ DRUM STATEMENT PROCESSOR)                                 FT023200
NODRM1  TRA     PARITH          PULL THIS CARD TO ACTIVATE DRUM STATEMENT       FT023210
        SPRSW   1RDRMX,XEQ,NOTRA,NODO                                           FT023220
T10.1   TXI     *+1,2,-4                                                        FT023230
        TSX     BREAK,4                                                         FT023240
        LAS     =077                                                            FT023250
        MACER                   MACHINE ERROR                                   FT023260
        TRA     *+3                                                             FT023270
        ERA     =H00000,                                                        FT023280
        TNZ     PARITH                                                          FT023290
        REMARK  (READ STATEMENT PROCESSOR)                                      FT023300
        SPRSW   1READX,XEQ,NOTRA,NODO                                           FT023310
T10.4   CAL     =HRETURN                                                        FT023320
        ERA     COLUMN,2                                                        FT023330
        TNZ     T10.5                                                           FT023340
        TXI     *+1,2,-6                                                        FT023350
        TSX     BREAK,4                                                         FT023360
        ERA     =077                                                            FT023370
        TNZ     PARITH                                                          FT023380
        REMARK  (RETURN STATEMENT PROCESSOR)                                    FT023390
        SPRSW   PRTRN,XEQ,TRANS,NODO                                            FT023400
T10.5   CAL     =HREWIND                                                        FT023410
        ERA     COLUMN,2                                                        FT023420
        TNZ     PARITH                                                          FT023430
        TXI     *+1,2,-6                                                        FT023440
        TSX     BREAK,4                                                         FT023450
        ERA     =077                                                            FT023460
```

```
TNZ     PARITH                                                  FT023470
REMARK  (REWIND STATEMENT PROCESSOR)                            FT023480
SPRSW   1REWXX,XEQ,NOTRA,NODO                                   FT023490
```

```
        EJECT
TS11.0  CAL     =HSENSEL                                        FT023500
        ERA     COLUMN,2                                        FT023510
        TNZ     T11.1                                           FT023520
        CAL     =HIGHT                                          FT023530
        ERA     COLUMN+6,2                                      FT023540
        ANA     =0777777770000                                  FT023550
        TNZ     PARITH                                          FT023560
        REMARK  (SENSE LIGHT STATEMENT PROCESSOR)               FT023570
        SPRSW   1SENSE,XEQ,NOTRA,NODO                           FT023580
T11.1   CAL     =HSUBROU                                        FT023590
        ERA     COLUMN,2                                        FT023600
        TNZ     T11.2                                           FT023610
        CAL     =HTINE                                          FT023620
        ERA     COLUMN+6,2                                      FT023630
        ANA     =0777777770000                                  FT023640
        TNZ     PARITH                                          FT023650
        REMARK  (SUBROUTINE STATEMENT PROCESSOR)                FT023660
        SPRSW   PSUBR,NOXEQ,NOTRA,NODO                          FT023670
T11.2   CAL     =HSTOP                                          FT023680
        ERA     COLUMN,2                                        FT023690
        ANA     =0777777770000                                  FT023700
        TNZ     TS11.3                                          FT023710
        TXI     *+1,2,-4                                        FT023720
        TSX     BREAK,4                                         FT023730
        ERA     =077                                            FT023740
        TNZ     PARITH                                          FT023750
        REMARK  (STOP STATEMENT PROCESSOR)                      FT023760
        SPRSW   1STOPX,XEQ,TRANS,NODO                           FT023770
                                                                FT023780
```

```
        EJECT
TW12.0  CAL     =HWRITET                                                        FT023790
        ERA     COLUMN,2                                                        FT023800
        TNZ     T12.1                                                           FT023810
        CAL     =HAPE                                                           FT023820
        ERA     COLUMN+6,2                                                      FT023830
        ANA     =0777777000000                                                  FT023840
        REMARK  (WRITE TAPE STATEMENT PROCESSOR)                                FT023850
        SPRSW   1WRTPX,XEQ,NOTRA,NODO                                           FT023860
T12.1   CAL     =HWRITED                                                        FT023870
        ERA     COLUMN,2                                                        FT023880
        TNZ     T12.2                                                           FT023890
        CAL     =HRUM                                                           FT023900
        ERA     COLUMN+6,2                                                      FT023910
        ANA     =0777777000000                                                  FT023920
        TNZ     PARITH                                                          FT023930
        REMARK  (WRITE DRUM STATEMENT PROCESSOR)                                FT023940
NODRM2  TRA     PARITH          PULL THIS CARD TO ACTIVATE DRUM STATEMENT       FT023950
        SPRSW   1WDRMX,XEQ,NOTRA,NODO                                           FT023960
T12.2   CAL     =HWRITEO                                                        FT023970
        ERA     COLUMN,2                                                        FT023980
        TNZ     PARITH                                                          FT023990
        CAL     =HUTPUTT                                                        FT024000
        ERA     COLUMN+6,2                                                      FT024010
        TNZ     PARITH                                                          FT024020
        CAL     =HAPE                                                           FT024030
        ERA     COLUMN+12,2                                                     FT024040
        ANA     =0777777000000                                                  FT024050
        TNZ     T8.5A                                                           FT024060
        REMARK  (WRITE OUTPUT TAPE STATEMENT PROCESSOR)                         FT024070
        SPRSW   1WOTXX,XEQ,NOTRA,NODO                                           FT024080
        REM                                                                     FT024090
        REM                                                                     FT024100
PARITH  REMARK  (ARITHMETIC STATEMENT PROCESSOR)                                FT024110
        SPRSW   1ARITH,XEQ,NOTRA,NODO                                           FT024120
                                                                                FT024130
```

```
        EJECT                                                          FT024140
*       PROCESSING FOR NON-FORTRAN STATEMENTS                          FT024150
        SPACE   2                                                      FT024160
TL13.0  CAL     =HLISTPA        IS IT PASS1                            FT024170
        ERA     COLUMN,2                                               FT024180
        TNZ     T13.2           NO                                     FT024190
        CAL     =HSS1OFF                                               FT024200
        ERA     COLUMN+6,2                                             FT024210
        TNZ     T13.4                                                  FT024220
        STZ     LPASS1                                                 FT024230
        REMARK  (TURN OFF LISTING OF PASS 1)                           FT024240
        TRA     T16.2                                                  FT024250
T13.4   ANA     =0777777000000                                         FT024260
        TNZ     PARITH                                                 FT024270
        STL     LPASS1          TURN ON LISTING OF PASS 1              FT024280
        REMARK  (TURN ON LISTING OF PASS1)                             FT024290
        TRA     T16.2                                                  FT024300
T13.2   CAL     =HLISTSY        IS IT LISTSYMBOLTABLE                  FT024310
        ERA     COLUMN,2                                               FT024320
        TNZ     PARITH                                                 FT024330
        CAL     =HMBOLTA                                               FT024340
        ERA     COLUMN+6,2                                             FT024350
        TNZ     PARITH                                                 FT024360
        CAL     =HBLE                                                  FT024370
        ERA     COLUMN+12,2                                            FT024380
        ANA     =0777777000000                                         FT024390
        TNZ     T8.5A                                                  FT024400
        REMARK  (TURN ON SYMBOL TABLE LIST FLAG)                       FT024410
        STL     LSTSYM                                                 FT024420
        TRA     T16.2                                                  FT024430
T15.0   CAL     =HONLINE        IS IT ONLINE                           FT024440
        ERA     COLUMN,2                                               FT024450
        TNZ     T15.1                                                  FT024460
        TXI     *+1,2,-6                                               FT024470
        SKIP    EOS                                                    FT024480
        TRA     PARITH                                                 FT024490
        REMARK  (PRINT OUTPUT ON LINE ALSO)                            FT024500
        STL     PRIQQ                                                  FT024510
        TRA     T16.2                                                  FT024520
T15.1   CAL     =HOFFLIN        IS IT OFF LINE                         FT024530
        ERA     COLUMN,2                                               FT024540
        TNZ     PARITH                                                 FT024550
        CAL     =HE                                                    FT024560
        ERA     COLUMN+6,2                                             FT024570
        ANA     =0770000000000                                         FT024580
        TNZ     PARITH                                                 FT024590
        REMARK  (TURN OFF ON-LINE LISTING)                             FT024600
        STZ     PRIQQ                                                  FT024610
        TRA     T16.2                                                  FT024620
TF16.0  CAL     =HFLOATI                                               FT024630
        ERA     COLUMN,2                                               FT024640
        TNZ     PARITH                                                 FT024650
        CAL     =HNGROUN                                               FT024660
        ERA     COLUMN+6,2                                             FT024670
        TNZ     PARITH                                                 FT024680
        CAL     =HDOFF                                                 FT024690
        ERA     COLUMN+12,2                                            FT024700
        ANA     =0777777770000                                         FT024710
        TNZ     T16.1                                                  FT024720
        STZ     ROUNDF                                                 FT024730
```

```
        REMARK  (FLOATING ROUNDING OFF)                                 FT024740
T16.2   SPRSW   SKEND,NOXEQ,NOTRA,NODO                                  FT024750
T16.1   ANA     =0770000000000                                          FT024760
        TNZ     PARITH                                                  FT024770
        STL     ROUNDF                                                  FT024780
        REMARK  (FLOATING ROUNDING ON)                                  FT024790
        TRA     T16.2                                                   FT024800
TS11.3  CAL     =HSYMBOL                                                FT024810
        ERA     COLUMN,2                                                FT024820
        TNZ     PARITH                                                  FT024830
        CAL     =HTABLE                                                 FT024840
        ERA     COLUMN+6,2                                              FT024850
        ARS     6                                                       FT024860
        TNZ     PARITH                                                  FT024870
        REMARK  (SYMBOL TABLE REQUESTED)                                FT024880
        CAL     =020                                                    FT024890
        ORS     18              SET MONITOR BIT ON                      FT024900
        SPRSW   SKEND,NOXEQ,NOTRA,NODO                                  FT024910
```

```
PROCX1  SLW     PROCSW          ENTER HERE VIA SPRSW MACRO                      FT024930
        PAI                                                                     FT024940
        LFT     BEND            IS IT THE END STATEMENT.                        FT024950
        TRA     PEND            YES                                             FT024960
        STA     LBL5              SET EXIT TO APPROPRIATE PROCESSOR             FT024970
        NZT     DOFLAG            WAS PREVIOUS STATEMENT A DO.                  FT024980
        TRA     *+4               NO                                            FT024990
        LNT     BXEQ              YES IT WAS. IS THIS STATEMENT EXECUTABLE      FT025000
        REM                       (XEQ BIT SHOULD BE ON).                       FT025010
        ERROR   65,LBL3         NON-EXECUTABLE STATEMENT NOT ALLOWED AS         FT025020
        REM                       FIRST STATEMENT FOLLOWING A DO.               FT025030
        STZ     DOFLAG            TURN OFF DO INDICATION                        FT025040
        LNT     BXEQ              IS THIS STATEMENT XECUTABLE.                  FT025050
        TRA     LBL3            NO. SKIP OVER ANY WAITING TRANSFER CODING       FT025060
        REM                      UNTIL NEXT EXECUTABLE STATEMENT IS             FT025070
        REM                     ENCOUNTERED.                                    FT025080
        REM                     THIS IS AN EXECUTABLE STATEMENT                 FT025090
        NZT     CARD            IS THERE A LABEL                                FT025100
        TRA     LBL4              NO                                            FT025110
        REM                       PROCESS THE LABEL.                            FT025120
LBL10   LDQ     CARD            YES.  DEFINE LABEL (CHECK FOR                   FT025130
        REM                       MULTIPLE DEFINITION).                         FT025140
        TSX     LOCATE,4          LOOK IT UP                                    FT025150
        PAX     ,4                                                              FT025160
        LFT     BLHSX+MSTRG       HAS LABEL PREVIOUSLY BEEN DEFINED EITHER      FT025170
        REM                       AS A LABEL OR A FORMAT NAME.                  FT025180
        ERROR   58,*+2            MULTIPLE LABEL DEFINITION                     FT025190
LBL8    SIL     BLHSX             NOT A STRING NAME OR MULTIPLE DEFINED.        FT025200
        REM                       FLAG AS DEFINED                               FT025210
        NZT     TRAFLG          IS PRECEEDING STATEMENT A TRANSFER COMMAND      FT025220
        SIL     BPATH           NO. THEREFORE PATH OF FLOW COMES IN FROM        FT025230
        REM                     ABOVE.                                          FT025240
        STI     EQUIV,4                                                         FT025250
        SLW     CARD            SAVE LABEL POINTER IN CARD                      FT025260
        REM                                                                     FT025270
        ZET     TRAFLG          WAS PRECEDING STATEMENT A TRANSFER.             FT025280
        TSX     LBL11,4         YES.  GO TO POST-PROCESSOR ROUTINE              FT025290
        REM                                                                     FT025300
        NZT     CARD            IS THERE A LABEL.                               FT025310
        TRA     *+4             NO                                              FT025320
        LXA     CARD,4          YES.  GET THE EQUIV POINTER                     FT025330
        CAL     PCOUNT          DEFINE THE LABEL                                FT025340
        STA     EQUIV,4                                                         FT025350
        LDI     PROCSW                                                          FT025360
        REM                                                                     FT025370
        NZT     DOPNTR          ARE THERE ACTIVE DO'S                           FT025380
        TRA     LBL4A           NO                                              FT025390
        REM                     IS THIS STATEMENT THE TERMINUS OF A DO          FT025400
        REM                                                                     FT025410
*                                                                               FT025420
        LXA     DOPNTR,4                                                        FT025430
        CAL     CARD                                                            FT025440
        ERA     DOSTAK,4                                                        FT025450
        TNZ     LBL4A           NO                                              FT025460
*                                                                               FT025470
        REM                     THIS IS TERMINAL STATEMENT OF A DO.             FT025480
        REM                     CHECK THAT IT IS NOT A DO, TRANSFER, OR         FT025490
        REM                     NON-EXECUTABLE COMMAND.                         FT025500
        LFT     BXEQ                                                            FT025510
```

```
        LFT     BDO+BTRANS                                              FT025520
        ERROR   66,SKEND                                                FT025530
        REM                     TURN ON DO TERMINATION IN SKEND         FT025540
        STL     DOTRMF                                                  FT025550
        TRA     LBL4A                                                   FT025560
        REM                                                             FT025570
LBL4    ZET     TRAFLG          IS PRECEDING STATEMENT A TRANSFER       FT025580
        ERROR   67,*+1                                                  FT025590
        REM                                                             FT025600
LBL4A   STZ     TRAFLG                                                  FT025610
LBL5    TRA     **              NO. GO TO PROCESSOR                     FT025620
        REM                                                             FT025630
LBL11   TRA*    TRAFLG                                                  FT025640
*                                                                       FT025650
*                                                                       FT025660
LBL3    LXA     PROCSW,4        STATEMENT IS NOT EXECUTABLE... IS IT    FT025670
        REM     A FORMAT STATEMENT                                      FT025680
        TXL     *+2,4,PFMT-1                                            FT025690
        TXL     LBL6,4,PFMT       YES, IT IS FORMAT                     FT025700
        STZ     CARD            CLEAR OUT ANY LABEL                     FT025710
        TRA*    LBL5                                                    FT025720
        REM                       NOW VALIDATE FORMAT LABEL             FT025730
LBL6    NZT     CARD              IS THERE A LABEL                      FT025740
        ERROR   68,SKEND                                                FT025750
        LDQ     CARD                                                    FT025760
        TSX     LOCATE,4          LACATE LABEL IN S.T.                  FT025770
        PAX     ,4                                                      FT025780
        TMI     LBL7              FORMAT NAME IS NEW                    FT025790
        LFT     BLHSX           LABEL NOT NEW.  CHECK HOW IT GOT IN     FT025800
        REM                       SYMBOL TABLE,                         FT025810
        ERROR   58,SKEND          LABEL ALREADY DEFINED EITHER AS A     FT025820
        REM                       STRAIGHT LABEL OR AS A STRING ,FORMAT)FT025830
LBL7    SIL BLHSX+MSTRG         LABEL IS NEW.  FLAG NAME AS STRING      FT025840
        REM                       AND TURN ON DEFINED INDICATION        FT025850
        STI     EQUIV,4                                                 FT025860
        TRA*    LBL5            GO TO PROCESSOR                         FT025870
```

```
*       SKEND ROUTINE.                                                  FT025880
                                                                        FT025890
        REM                                                             FT025900
*       SKEND IS THE TERMINATION ROUTINE FOR ALL THE STATEMENT          FT025910
*       PROCESSORS EXCEPT 'END'.                                        FT025920
                                                                        FT025930
        REM                                                             FT025940
*       A TEST IS MADE TO DETERMINE IF THE POOL AREA HAS OVERFLOWED.    FT025950
*       THE CELL 'FIRSTF' IS UPDATED.  THE ADDRESS HOLDS THE NUMBER     FT025960
*       OF STATEMENTS PROCESSED, AND THE DECREMENT HILDS THE NUMBER     FT025970
*       OF EXECUTABLE STATEMENTS.                                       FT025980
        REM                                                             FT025990
*       IF THIS STATEMENT TERMINATES ONE OR MORE DO LOOPS, THE TER-     FT026000
*       MINATION CODE IS PUT OUT.                                       FT026010
        REM                                                             FT026020
*       IF THIS IS A CALL OR TRANSFER-TYPE STATEMENT, THE INDREG        FT026030
*       ROUTINE IS ENTERED, WHICH RELEASES ALL NON-DO-INDEX-            FT026040
*       CONTAINING XRS.                                                 FT026050
        REM                                                             FT026060
*       WORKING CELLS ARE RELEASED, AND THE ROUTINE THEN RETURNS        FT026070
*       TO PROCESS THE NEXT CARD.                                       FT026080
        SPACE   3                                                       FT026090
SKEND   LXA     NXTLOC,4        TEST FOR POOL OVERFLOW                  FT026100
        TXL     *+2,4,POOL-COLUMN-667                                   FT026110
        STR     MSG100,,FINISH      *****TEMPORARY RETURN LOCATION      FT026120
        REM                                                             FT026130
        CAL     FIRSTF          GET STATEMENT COUNTER                   FT026140
        ADD     =1              BUMP                                    FT026150
        LDI     PROCSW                                                  FT026160
        LFT     BXEQ            COUNT EXECUTABLE STATEMENTS IN DECREMENT FT026170
        ADD     =01000000                                               FT026180
        STO     FIRSTF          RESTORE STATEMENT COUNTER               FT026190
        REM                                                             FT026200
        ZET     DOTRMF          IS THIS STATEMENT THE TERMINATION OF A DO FT026210
        TSX     DOTERM,4        YES. CLOSE OUT ALL DO LOOPS TERMINATED BY FT026220
        REM                     THIS STATEMENT                          FT026230
        STZ     DOTRMF          RESET THE DO TERMINATION FLAG           FT026240
        REM                                                             FT026250
        LDI     PROCSW          GET PROCESSOR SWITCH                    FT026260
        LFT     BTRANS+BCALL    WAS THIS A TRANSFER OR A CALL STATEMENT. FT026270
        TSX     INDREG,4        YES.  CLEAR OUT THE NON-DO INDEX REGISTERS FT026280
        REM                                                             FT026290
        LFT     BXEQ            ALTER THE FLAG CELL IF A EXECUTABLE STATE. FT026300
        STZ     RASCAL                                                  FT026310
        LFT     BCALL           NON-ZERO CELL IF A CALL STATEMENT       FT026320
        STL     RASCAL          THIS FLAG USED BY END PROCESSOR ONLY    FT026330
        REM                                                             FT026340
        STZ     WRKCEL          RESET WORKING STORAGE                   FT026350
        TRA     STARTT          GO GET NEXT CARD
```

```
        TTL     P A S S  1  P R O C E S S O R  -  S U B R O U T I N E   FT026360
*       SUBROUTINE NAME (A,B,C,...,Z)                                  FT026370
        REM                                                            FT026380
*       ENTRY...  PSUBR                                                FT026390
        SPACE   3                                                      FT026400
PSUBR   STZ     FNCTFL          FLAG AS A SUBROUTINE                   FT026410
        AXC     10,2                                                   FT026420
        TRA     PFUNCT+2        TO FUNCTION PROCESSOR                  FT026430
```

```
        TTL     P A S S  1  P R O C E S S O R  -  F U N C T I O N          FT026440
*       FUNCTION NAME(A,B,C,...,Z)                                        FT026450
        REM                                                               FT026460
*       ENTRY...  PFUNCT                                                  FT026470
*       THE FUNCTION STATEMENT MUST BE THE FIRST STATEMENT OF THE         FT026480
*       PROGRAM. THAT IS, CELL FIRSTF MUST BE ZERO. THE NAME OF THE       FT026490
*       ROUTINE IS OBTAINED, CHECKED FOR VALIDITY, AND STORED IN CELL     FT026500
*       SBNAME.                                                           FT026510
        REM                                                               FT026520
*       ALL FUNCTIONS MUST HAVE AT LEAST ONE ARGUMENT. EACH ARGUMENT IS   FT026530
*       ENTERED IN SYMTAB FLAGGED AS A PARAMETER WITH MODE DETERMINED     FT026540
*       BY THE FIRST LETTER.   A COUNT OF THE NUMBER OF PARAMETERS IS     FT026550
*       KEPT IN CELL NOARGS.                                              FT026560
        REM                                                               FT026570
*       TO DISTINGUISH WHETHER THIS IS FUNCTION OR SUBROUTINE CODING      FT026580
*       TEST CELL  FNCTFL WHICH WILL BE NON-ZERO FOR FUNCTIONS.           FT026590
        SPACE   3                                                         FT026600
PFUNCT  STL     FNCTFL          FLAG AS A FUNCTION                        FT026610
        AXC     8,2             GET FUNCTION NAME                         FT026620
        ZET     FIRSTF          IS THIS THE 1ST STATEMENT OF PROGRAM      FT026630
        ERROR   52,SKEND        NO.  SUBPROGRAM DECLARATION COMES TOO LATE FT026640
        TSX     SCNBCD,4                                                  FT026650
        SLW     SBNAME          SAVE THE NAME                             FT026660
        NZT     LABEL           HAS DECK LABEL ALREADY BEEN ASSIGNED.     FT026670
        SLW     LABEL           NO                                        FT026680
        REM                                                               FT026690
        NZT     FNCTFL                                                    FT026700
        TRA     PFUNC1          SUBROUTINE CODEING.  OMIT NAME TEST       FT026710
        CAL     SCNB1           VALIDATE THE FUNCTION NAME                FT026720
        STD     *+2                                                       FT026730
        LAC     SCNB1-1,4                                                 FT026740
        TXI     *+1,4,**        NO. CHARACTERS IN XR4                     FT026750
        TXL     PFUNC1,4,3      NAME LESS THAN 4 LETTERS                  FT026760
        REM                     IS LAST LETTER F.                         FT026770
        TXI     *+1,2,1                                                   FT026780
        SKIP    FCHAR           TEST FOR TERMINAL F                       FT026790
        TXI     *+2,2,-1                                                  FT026800
        ERROR   51,SKEND        LAST LETTER WAS F                         FT026810
        REM                                                               FT026820
*       NAME IS A VALID FUNCTION NAME.  DO NOT ENTER NAME IN SYMTAB NOW,  FT026830
*       BUT WAIT FOR ITS APPEARANCE ON THE LEFT HAND SIDE OF A SUBSTITU-  FT026840
*       TION STATEMENT.  THEN AS A PART OF THE END STATEMENT PROCESSING,  FT026850
*       TEST THAT FUNCTION NAME IS IN SYMTAB WITN BLHSX FLAG BIT ON.      FT026860
PFUNC1  STZ     NOARGS          GET ARGUMENTS                             FT026870
        SKIP1   LPCHAR                                                    FT026880
        TRA     PFUNC4                                                    FT026890
        NZT     FNCTFL                                                    FT026900
        TRA     PFUNC2          SUBROUTINE CODING.  MAY BE NO ARGUMENTS   FT026910
PFUNC3  ERROR   7,SKEND         FUNCTION MUST HAVE AT LEAST ONE ARGUMENT  FT026920
PFUNC4  TSX     SCNBCD,4                                                  FT026930
        XCL                                                               FT026940
        TSX     LOCATE,4                                                  FT026950
        TPL     PFUNC3          SYMBOL ALREADY IN SYMTAB. ERROR           FT026960
        PAX     ,4                                                        FT026970
        CAL     SYMTAB,4        DETERMINE MODE OD ARGUMENT                FT026980
        LAS     =0457777777777                                            FT026990
        SIL     MREAL                                                     FT027000
        TRA     PFUNC5                                                    FT027010
        LAS     =0307777777777                                            FT027020
        SIL     MINTG                                                     FT027030
```

```
        TRA     PFUNC5                                                  FT027040
        SIL     MREAL                                                   FT027050
PFUNC5  SIL     BARGT+BLHSX     APPEND PARAMETER FLAG AND DEFINITION FLAG FT027060
        STI     EQUIV,4                                                 FT027070
        CLA     NOARGS                                                  FT027080
        ADD     =1                                                      FT027090
        STO     NOARGS                                                  FT027100
        ALS     15                                                      FT027110
        ORS     EQUIV,4                                                 FT027120
        REM                     IS THIS THE LAST ARGUMENT               FT027130
        SKIP1   CMCHAR                                                  FT027140
        TRA     PFUNC4                                                  FT027150
        SKIP    RPCHAR                                                  FT027160
        TRA     PFUNC3                                                  FT027170
        REM                     NO MORE ARGUMENTS                       FT027180
PFUNC2  SKIP    63                                                      FT027190
        TRA     PFUNC3                                                  FT027200
PFUNC6  STZ     RTNFLG            TURN RETURN STATEMENT INDICATION OFF  FT027210
        TRA     SKEND                                                   FT027220
```

```
        TTL     P A S S  1  P R O C E S S O R  -  R E T U R N                FT027230
*       RETURN                                                              FT027240
        REM                                                                 FT027250
*       ENTRY...   PRTRN                                                    FT027260
*       VALID ONLY IN SUBROUTINE OR FUNCTION CODING.                        FT027270
*       RETURN STATEMENT MUST APPEAR AT LEAST ONCE IN A FUNCTION            FT027280
*       SUBPROGRAM.  RTNFLG CONTAINS A COUNT OF RETURN STATEMENTS.          FT027290
        REM                                                                 FT027300
*       THIS STATEMENT CAUSES A TRANSFER TO THE END OF THE TEXT,            FT027310
*       WHERE THE ACCUMULATOR WILL BE LOADED (FOR FUNCTIONS).  THEN         FT027320
*       A TRANSFER BACK UP TO THE PROLOGUE EXIT SEQUENCE IS GENERATED.      FT027330
        SPACE   3                                                           FT027340
PRTRN   NZT     SBNAME          IS THIS SUBROUTINE                          FT027350
        ERROR   53,SKEND        MAIN PROGRAM                                FT027360
        CAL     MODFLG          FUNCTION.  IS COL 1 OF THIS STATEMENT       FT027370
        NZT     RTNFLG          THE SAME AS ON ALL OTHER RETURN STATEMENTS  FT027380
        SLW     RTNMOD          ENCOUNTERED SO FAR.                         FT027390
        LAS     RTNMOD                                                      FT027400
        TRA     *+2             NO                                          FT027410
        TRA     *+2             YES                                         FT027420
        ERROR   54,*+1                                                      FT027430
        CLA     RTNFLG          STEP RETURN STATEMENT COUNT BY 1            FT027440
        ADD     =1                                                          FT027450
        STO     RTNFLG                                                      FT027460
        AXT     PRTN3,4         SET TRAFLG FOR POSSIBLE TRANSFER CODING     FT027470
        SXA     TRAFLG,4        TO JUMP TO END OF TEXT IN  CASE NEXT CARD   FT027480
        REM                     IS NOT AN END CARD.                         FT027490
        TRA     SKEND                                                       FT027500
        REM                                                                 FT027510
        REM                                                                 FT027520
*       ENTER HERE IF TRAFLG IS ON.  TEST IF THIS STATEMENT IS AN END       FT027530
*       STATEMENT.  IF SO OMIT THE TRA INSTR. NORMALLY GENERATED BY         FT027540
*       THE RETURN STATEMENT.                                               FT027550
PRTN3   SXA     PRTN3X,4                                                    FT027560
        LXA     PROCSW,4                                                    FT027570
        TXL     PRTN4,4,PEND-1                                              FT027580
        TXH     PRTN4,4,PEND                                                FT027590
PRTN3X  AXT     **,4            STATEMENT IS END.                           FT027600
        TRA     1,4                                                         FT027610
        REM                     STATEMENT NOT END.                          FT027620
PRTN4   CAL     TRAEND          NOT END.                                    FT027630
        TSX     CITBLD,4,R                                                  FT027640
        TRA     PRTN3X                                                      FT027650
        REM                                                                 FT027660
TRAEND  TRA     EQUIV-LOCEND                                                FT027670
```

```
*       CALL STATEMENT PROCESSOR                                        FT027690
*       ENTRY...  1CALLX                                                FT027700
        REM                                                             FT027710
*       A SUBROUTINE MAY NOT CALL ITSELF.  THE NAME IN THE CALL STATEMENTFT027720
*       MAY NOT BE USED AS A VARIABLE.  IT IS LEGAL FOR THE NAME TO BE  FT027730
*       USED ALSO AS A FUNCTION NAME, EITHER FORTRAN OR LIBRARY.        FT027740
        REM                                                             FT027750
*       THE ARGUMENT LIST IS PROCESSED BY THE COMPIL ROUTINE IN A MANNER FT027760
*       VERY SIMILAR TO PROCESSING A FORTRAN FUNCTION CALL.             FT027770
        REM                                                             FT027780
*       IF THERE IS NO ARGUMENT LIST, THE CODE  TSX NAME,4 IS GENERATED FT027790
*       HERE AND NOW.                                                   FT027800
        SPACE   3                                                       FT027810
1CALLX  AXC     4,2                                                     FT027820
        TSX     SCNBCD,4        GET SUBROUTINE NAME                     FT027830
        TMI     ERROR8          NOT SYMBOLIC                            FT027840
        REM                                                             FT027850
        LAS     SBNAME          COMPARE WITH THIS PROGRAM'S NAME        FT027860
        TRA     *+2                                                     FT027870
        ERROR   10,*+1          PROGRAM MAY NOT CALL ITSELF             FT027880
        REM                                                             FT027890
        SLW     FORM69          SAVE NAME OF SUBROUTINE                 FT027900
        XCL                                                             FT027910
        TSX     LOCATE,4        PUT IN SYMTAB                           FT027920
        SLW     ASTACK          SAVE EQUIV POINTER                      FT027930
        PAX     ,4                                                      FT027940
        TMI     1CALL2          NEW ITEM IN SYMTAB                      FT027950
        REM                                                             FT027960
        LFT     BEXTF+BARGT     OLD ITEM.  MUST BE EXTERNAL OR ARGUMENT FT027970
        LFT     BVARB+BARRY+BCOMN+BEQUV                                 FT027980
        ERROR   69,SKEND        MULTIPLE USAGE OF SUBROUTINE NAME       FT027990
        RIL     700000+BLIBF    RESET MODE BITS AND LIBRARY FLAG        FT028000
        REM                                                             FT028010
1CALL2  CAL     FORM69          GET BCD NAME                            FT028020
        LAS     =HI00000        DETERMINE A 'MODE'.  THIS IS NECESSARY TO FT028030
        TRA     *+1             FAKE OUT LATER SECTIONS OF COMPILER.    FT028040
        LAS     =HN(((((                                                FT028050
        SIL     BEXTF+MREAL     MARK AS EXTERNAL AND REAL               FT028060
        TRA     *+2                                                     FT028070
        SIL     BEXTF+MINTG     MARK AS FIXED AND EXTERNAL              FT028080
        REM                                                             FT028090
1CALL1  STI     EQUIV,4         SET EQUIV WORD                          FT028100
        CAL     COLUMN,2        GET NEXT CHARACTER                      FT028110
        ERA     =H(                                                     FT028120
        ARS     30                                                      FT028130
        TNZ     1CALL4          NOT A LEFT PAREN                        FT028140
        REM                                                             FT028150
        CAL     ASTACK          SET ASTACK WORD.  DO NOT MOVE           FT028160
        STI     ASTACK               ASTACK POINTER YET.                FT028170
        STA     ASTACK                                                  FT028180
        REM                                                             FT028190
        TSX     CMPLCL,4        PROCESS THE CALLING SEQUENCE            FT028200
        TRA     SKEND                                                   FT028210
        REM                                                             FT028220
        REM                                                             FT028230
1CALL4  SKIP    63              MUST BE END OF STATEMENT                FT028240
        ERROR   7,SKEND                                                 FT028250
        TSX     PUTCL2,4        RESTORE CELL 2                          FT028260
        LFT     BARGT           IS IT AN ARGUMENT.                      FT028270
```

```
        TRA     1CALL3          YES                                                     FT028280
        REM                                                                             FT028290
        CODECO  ASTACK,TSX4,T,XRFLAG    NO ARGUMENTS IN CALL STATEMENT                  FT028300
        TRA     SKEND           CODE IS...  TSX NAME,4                                  FT028310
        REM                                                                             FT028320
1CALL3  CAL     ASTACK                                                                  FT028330
        TSX     PROLOG,4        MAKE PROLOGUE ENTRIES                                   FT028340
        CODEC   TSX4,N,XRFLAG       CODE IS...  TSX **,4                                FT028350
        TRA     SKEND                                                                   FT028360
```

```
                TTL     P A S S  1  P R O C E S S O R  -  E N D                  FT028370
*               END STATEMENT PROCESSOR                                         FT028380
                REM                                                             FT028390
*               ENTRY...    PEND                                                FT028400
                REM                                                             FT028410
*               IF THIS IS A SUBPROGRAM, A RETURN SEQUENCE IS GENERATED HERE    FT028420
*               FOR FUNCTIONS, AND FOR SUBROUTINES WHICH CONTAIN A              FT028430
*               RETURN STATEMENT.  THE CELL 'LOCEND', WHICH                     FT028440
*               IS USED BY THE RETURN STATEMENTS, IS NOW SET TO THE CURRENT     FT028450
*               VALUE OF PCOUNT.                                                FT028460
                REM                                                             FT028470
*               IF THE LAST EXECUTABLE STATEMENT WAS NOT A TRANSFER-TYPE,       FT028480
*               THEN AN ERROR MESSAGE IS ISSUED.                                FT028490
                REM                                                             FT028500
PEND            NZT     TRAFLG          LAST EXECUTABLE STATEMENT MUST BE A     FT028510
                ZET     RASCAL          TRANSFER OR A CALL.                     FT028520
                TRA     *+2                                                     FT028530
                ERROR   33,*+3          PROGRAM RUNS OFF THE END                FT028540
                REM                                                             FT028550
                ZET     TRAFLG          WAS THE LAST EXEC. STATEMENT A TRANSFER.FT028560
                TSX     LBL11,4         YES.  GO TO THE POST-PROCESSOR ROUTINE  FT028570
                REM                     TO TURN OUT ANY WAITING TRANSFER CODING.FT028580
                REM                                                             FT028590
                REM                                                             FT028600
PEND4           NZT     SBNAME          IS IT A SUBPROGRAM.                     FT028610
                TRA     PEND99          NO                                      FT028620
                CLA     PCOUNT          SET THE VALUE OF THE END CARD ENTRY     FT028630
                STO     LOCEND          FOR RETURN STATEMENTS                   FT028640
                NZT     FNCTFL          IF A FUNCTION, THEN RTNFLG MUST BE ON.  FT028650
                TRA     PEND15          NOT A FUNCTION                          FT028660
                NZT     RTNFLG                                                  FT028670
                ERROR   55,*+1                                                  FT028680
                REM                                                             FT028690
                LDQ     SBNAME          GET THE NAME OF THIS SUNCTION           FT028700
                TSX     LOCATE,4        LOOK IT UP IN SYMBOL TABLE              FT028710
                PAX     ,4              PRESERVE THE POINTER                    FT028720
                TPL     PEND6           OLD ITEM IN THE TABLE                   FT028730
                STZ     SYMTAB,4        REMOVE THE SYMBOL FROM TABLE            FT028740
                ERROR   56,PEND15       AND PUT OUT AN ERROR MESSAGE            FT028750
                REM                                                             FT028760
PEND6           LNT     BLHSX           HAS NAME BEEN DEFINED.                  FT028770
                ERROR   56,PEND15       NO.                                     FT028780
                LFT     BARGT+BARRY+BEXTF+BINTF                                 FT028790
                ERROR   85,PEND15       ILLEGAL USE OF THE NAME                 FT028800
                REM                                                             FT028810
                SXA     PEND10,4        SAVE THE POINTER                        FT028820
                CAL     RTNMOD          GET MODE OF RETURNS                     FT028830
                LAS     =1                                                      FT028840
                TRA     PEND13          DOUBLE OR COMPLEX                       FT028850
                TRA     PEND14          BOOLEAN                                 FT028860
                REM                                                             FT028870
                CAL     PEND10          GET EQUIV POINTER FOR NAME              FT028880
                CODEO   CLA,V           CODE IS...  CLA NAME                    FT028890
                REM                                                             FT028900
PEND15          CAL     TRA             GET A TRANSFER OP CODE                  FT028910
                ZET     RTNFLG          IS THERE A RETURN STATEMENT IN THIS PROGRAMFT028920
                TSX     CITBLD,4,X      YES.  PUT OUT THE TRA TO RETURN SEQUENCE FT028930
                REM                                                             FT028940
PEND99          STZ     LPASS1          KILL LISTING OF THIS MESS               FT028950
                TSX     CITBLN,4,END    PASS ON THE END FLAG TO CITBLD          FT028960
```

```
        TRA     PPASS1          GO NOW TO POST PASS 1                   FT028970
        SPACE   3                                                       FT028980
PEND14  CAL     PEND10          GET POINTER TO NAME                     FT028990
        CODEO   CAL,V           CODE IS...  CAL NAME                    FT029000
        TRA     PEND15                                                  FT029010
        REM                                                             FT029020
PEND13  CAL     PEND10          GET POINTER TO NAME                     FT029030
        CODEO   CLA,V                                                   FT029040
        CAL     PEND10                                                  FT029050
        CODEO   LDQ,G                                                   FT029060
        CODEN   STO777                                                  FT029070
        CODEN   STQ776                                                  FT029080
        TRA     PEND15                                                  FT029090
        REM                                                             FT029100
```

```
        TTL     P A S S  1  P R O C E S S O R  -  F - C A R D  S C A N   FT029110
*       ENTRY...  1FCARD                                                FT029120
        REM                                                             FT029130
        REM                                                             FT029140
1FCARD  AXC     0,2             RESET POINTER TO COLUMN                 FT029150
        TSX     SCNBCD,4        GET NAME                                FT029160
        TMI     ERROR7          NOT A SYMBOLIC NAME                     FT029170
        REM                                                             FT029180
        SLW     FORM69          SAVE NAME IN ERROR MESSAGE CELL         FT029190
        XCL                                                             FT029200
        TSX     LOCATE,4        PUT NAME IN SYMBOL TABLE                FT029210
        TMI     1FCAR1          NEW SYMBOL                              FT029220
        REM                                                             FT029230
        LFT     BVARB+BCOMN+BARRY+BEQUV     ALL FLAGS MUST BE OFF       FT029240
        ERROR   69,1FCAR2                                               FT029250
        LFT     BARGT           IS IT AN UNCOMMITTED ARGUMENT.          FT029260
        SIL     BEXTF           YES.  TURN ON EXTERNAL FUNCTION FLAG    FT029270
        LNT     BEXTF           EXTERNAL FUNCTION FLAG MUST NOW BE ON   FT029280
        ERROR   69,1FCAR2                                               FT029290
        TRA     1FCAR2                                                  FT029300
        REM                                                             FT029310
1FCAR1  PAX     ,4              GET EQUIV POINTER                       FT029320
        CAL     FORM69          GET BCD NAME OF FUNCTION                FT029330
        LAS     =HI00000        DETERMINE THE MODE.  THIS IS NECESSARY  FT029340
        TRA     *+1             TO KEEP LATER SECTIONS OF COMPILER      FT029350
        LAS     =HN(((((        FROM LOUSING UP.                        FT029360
        SIL     BEXTF+MREAL     MARK AS EXTERNAL AND REAL               FT029370
        TRA     *+2                                                     FT029380
        SIL     BEXTF+MINTG     MARK AS EXTERNAL AND INTEGER            FT029390
        STI     EQUIV,4         RESET EQUIV WORD                        FT029400
        REM                                                             FT029410
1FCAR2  SKIP1   CMCHAR          TEST FOR COMMA                          FT029420
        TRA     1FCARD+1        COMMA FOUND.  RETURN FOR ANOTHER NAME   FT029430
        REM                                                             FT029440
        SKIP    63              TEST FOR END OF STATEMENT               FT029450
        ERROR   3,SKEND         NOT EOS.  ERROR                         FT029460
        TRA     SKEND                                                   FT029470
```

```
*       COMMON A,B,C,....                                                       FT029490
        REM                                                                     FT029500
*       ENTRY...  1COMNX                                                        FT029510
*       AN ENTRY IS MADE IN DIMTBL FOR EACH COMMON VARIABLE.                    FT029520
*       THE FIRST WORD OF THE DIMTBL ENTRY CONTAINS THE VALUE OF THE            FT029530
*       CURRENT COMMON COUNTER IN THE DECREMENT, AND EITHER THE EQUTBL          FT029540
*       OR EQUIV TABLE POINTER IN THE ADDRESS, DEPENDING IF BEQUV               FT029550
*       IS ON OR NOT.  IF THIS IS A NEW DIMTBL ENTRY, THREE WORDS ARE           FT029560
*       RESERVED TO ALLOW FOR POSSIBLE FUTURE DIMENSION OR ASSIGN INFO.         FT029570
*       STORAGE IS ALREADY AVAILABLE IN DIMTBL FOR THIS VARIABLE                FT029580
*       IF IT HAS PREVIOUSLY APPEARED IN A DIMENSION STATEMENT OR IN            FT029590
*       EITHER AN ASSIGN OR ASSIGNED GO TO STATEMENT.                           FT029600
*       THE EQUIV WORD ADDRESS GETS THE DIMTBL POINTER POSITION.                FT029610
*       THE FLAG BITS BCOMN AND BLHSX ARE TURNED ON.                            FT029620
*       A SYMBOL MAY NOT APPEAR MORE THAN ONCE IN A COMMON DECLARATION,         FT029630
*       AND ALL SYMBOLS MUST BE NON-SUBSCRIPTED NAMES.                          FT029640
        REM                                                                     FT029650
*       NOTE... DPOINT AND EPOINT ARE SYNONYMOUS WITH NXTLOC, AND               FT029660
*       DIMTBL AND EQUTBL ARE SYNONYMOUS WITH POOL.                             FT029670
        SPACE   3                                                               FT029680
1COMNX  AXC     6,2             RESET POINTER                                   FT029690
        TSX     SCNBCD,4        GET VARIABLE                                    FT029700
        TPL     *+2                                                             FT029710
        ERROR   2,SKEND         NOT A VARIABLE                                  FT029720
        REM                                                                     FT029730
        SLW     FORM69          SAVE NAME OF SYMBOL                             FT029740
        XCL                     MOVE FOR LOCATE ROUTINE                         FT029750
        TSX     LOCATE,4        PUT IN SYMTAB                                   FT029760
        PAX     ,1              GET EQUIV POINTER                               FT029770
        TPL     1COMN1          OLD ITEM                                        FT029780
        REM                                                                     FT029790
        CAL     FORM69          GET NAME OF SYMBOL                              FT029800
        LAS     =HI00000        NEW ITEM.  SET MODE BITS                        FT029810
        TRA     *+1                                                             FT029820
        LAS     =HN(((((                                                        FT029830
        SIL     MREAL           FLOATING POINT                                  FT029840
        TRA     *+2                                                             FT029850
        SIL     MINTG           FIXED POINT                                     FT029860
        PXA     ,1              RESTORE THE EQUIV POINTER                       FT029870
        REM                                                                     FT029880
1COMN4  LXA     DPOINT,4                                                        FT029890
        STA     DIMTBL,4        (DIMTBL) ADDRESS = EQUIV POINTER                FT029900
        REM                                                                     FT029910
        CLA     DPOINT                                                          FT029920
        ADD     =3              BUMP DPOINT BY THREE                            FT029930
        STO     DPOINT                                                          FT029940
        PXA     ,4                                                              FT029950
        REM                                                                     FT029960
1COMN5  SIL     BCOMN+BLHSX     TURN ON COMMON BIT AND DEFINITION BIT.          FT029970
        STI     EQUIV,1         SET UP EQUIV WORD                               FT029980
        STA     EQUIV,1         (EQUIV) ADDRESS = DPOINT                        FT029990
        REM                                                                     FT030000
        CAL     CPOINT          GET COMMON POINTER                              FT030010
        ADD     =1B17           BUMP COMMON POINTER                             FT030020
        STO     CPOINT                                                          FT030030
        REM                                                                     FT030040
        STD     DIMTBL,4        (DIMTBL) DECREMENT = CPOINT                     FT030050
        REM                                                                     FT030060
1COMN2  SKIP1   CMCHAR          CHECK FOR COMMA                                 FT030070
```

```
        TRA     1COMNX+1        COMMA FOUND.  RETURN FOR ANOTHER SYMBOL      FT030080
        REM                                                                   FT030090
        SKIP1   63              NOT A COMMA.  MUST BE END OF STATEMENT        FT030100
        TRA     SKEND           OKAY.                                         FT030110
        ERROR   5,SKEND                                                       FT030120
        REM                                                                   FT030130
        REM                                                                   FT030140
1COMN1  LFT     BCOMN           OLD SYMBOL.  IS COMMON BIT ON.                FT030150
        TRA     1COMN6          YES.  ERROR                                   FT030160
        REM                                                                   FT030170
        LFT     BEXTF+BINTF     MAY NOT BE MARKED AS SUBPROGRAM NAME          FT030180
        ERROR   69,1COMN2                                                     FT030190
        REM                                                                   FT030200
        LFT     BARRY+BASSN     NO.  IS STORAGE ALREADY AVAILABLE.            FT030210
        TRA     1COMN3          YES.                                          FT030220
        REM                                                                   FT030230
        LFT     BEQUV           NO.  IS EQUIVALENCE BIT ON.                   FT030240
        PIA                     YES.  (DIMTBL) ADDRESS = EQUTBL POINTER       FT030250
        TRA     1COMN4                                                        FT030260
        REM                                                                   FT030270
1COMN3  PIA                                                                   FT030280
        PAX     ,4                                                            FT030290
        TRA     1COMN5                                                        FT030300
        REM                                                                   FT030310
1COMN6  CAL     SYMTAB,1        GET BCD NAME OF SYMBOL IN ERROR               FT030320
        SLW     FORM27          PUT IN ERROR MESSAGE                          FT030330
        ERROR   27,1COMN2       PUT OUT MESSAGE AND RETURN                    FT030340
```

```
        TTL   P A S S  1  P R O C E S S O R  -  D I M E N S I O N          FT030350
*       DIMENSION A(I,J,K), B(I,J), C(I), ....                            FT030360
        REM                                                               FT030370
*       ENTRY...  1DIMNX                                                  FT030380
*       AN ENTRY IN THE DIMENSION STACK DIMTBL IS MADE FOR EACH           FT030390
*       VARIABLE ENCOUNTERED IN A DIMENSION STATEMENT.  THREE             FT030400
*       WORDS OF STORAGE IN DIMTBL ARE ALREADY AVAILABLE IF THE           FT030410
*       VARIABLE HAS PREVIOUSLY APPEARED IN A COMMON STATEMENT.           FT030420
*       IF NOT, A TWO OR THREE WORD ENTRY IS MADE, DEPENDING IF THE       FT030430
*       ARRAY IS ONE-DIMENSIONAL OR MORE-THAN-ONE-DIMENSIONAL.            FT030440
*       WORD TWO CONTAINS THE TOTAL LENGTH SPECIFIED BY THE               FT030450
*       SUBSCRIPTS (ADDRESS FIELD), THE FIRST DIMENSION (DECREMENT        FT030460
*       FIELD), AND THE NUMBER OF DIMENSIONS (PREFIX BITS).               FT030470
*       WORD THREE, IF NECESSARY, CONTAINS DIMENSIONS TWO (ADDRESS        FT030480
*       FIELD) AND THREE (DECREMENT FIELD).                               FT030490
        REM                                                               FT030500
*       IF BEQUV IS ON, THE EQUTBL POINTER (LOCATED IN THE ADDRESS        FT030510
*       OF THE EQUIV ENTRY) IS PLACED IN THE ADDRESS OF WORD ONE          FT030520
*       OF THE DIMTBL ENTRY.  IF BEQUV IS OFF, THE EQUIV TABLE            FT030530
*       POSITION IS PUT IN THE DIMTBL ADDRESS.  IN EITHER EVENT,          FT030540
*       THE ADDRESS OF THE EQUIV TABLE ENTRY IS SET TO THE                FT030550
*       DIMTBL POSITION.  BARRY IS TURNED ON.                             FT030560
        REM                                                               FT030570
*       IF COLUMN 1 CONTAINS A 'D' OR 'I', THE BIT BDOUB IS TURNED ON,    FT030580
*       INDICATING THAT THIS ARRAY WILL RECEIVE A DOUBLE-LENGTH STORAGE   FT030590
*       ASSIGNMENT.  A SYMBOL MAY NOT APPEAR MORE                         FT030600
*       THAN ONCE IN A DIMENSION STATEMENT, AND EACH SYMBOL MUST          FT030610
*       BE SUBSCRIPTED.  AN ASSIGN VARIABLE MAY NOT APPEAR IN A           FT030620
*       DIMENSION STATEMENT.                                              FT030630
*       AN ARRAY MUST APPEAR IN A DIMENSION STATEMENT PRIOR TO ITS        FT030640
*       FIRST USAGE IN AN EXECUTABLE STATEMENT.                           FT030650
        REM                                                               FT030660
*       NOTE... DPOINT AND EPOINT ARE SYNONYMOUS WITH NXTLOC, AND         FT030670
*       DIMTBL AND EQUTBL ARE SYNONYMOUS WITH POOL.                       FT030680
        SPACE   3                                                         FT030690
1DIMNX  AXC     9,2             RESET POINTER                             FT030700
        REM                                                               FT030710
1DIMN7  TSX     SCNBCD,4        GET ARRAY NAME                            FT030720
        TMI     ERROR8          INVALID ARRAY NAME                        FT030730
        SLW     .P              SAVE BCD NAME                             FT030740
        REM                                                               FT030750
        SKIP    LPCHAR          TEST FOR LEFT PAREN                       FT030760
        ERROR   9,SKEND         NO PAREN                                  FT030770
        REM                                                               FT030780
        LDQ     =1B17                                                     FT030790
        STQ     .P+6            SET MULTIPLIER                            FT030800
        AXT     1,1                                                       FT030810
        REM                                                               FT030820
1DIMN1  TSX     SCNBCD,4        GET DIMENSION                             FT030830
        TMI     *+4             IS SUBSCRIPT A NUMBER.                    FT030840
        REM                                                               FT030850
        ERROR   17,*+1          NO.  (ZERO SUBSCRIPT CAUGHT BY SCNBCD).   FT030860
        CAL     =1B17           FAKE A SUBSCRIPT                          FT030870
        TRA     1DIMN5                                                    FT030880
        REM                                                               FT030890
        TSX     BCDFIX,4        CONVERT TO FIXED POINT NUMBER             FT030900
        SLW     .P+5,1          SAVE DIMENSION                            FT030910
        XCA                                                               FT030920
        TOV     *+1             TURN OFF OVERFLOW LIGHT                   FT030930
        MPY     .P+6            COMPUTE TOTAL STORAGE REQUIRED            FT030940
```

```
        ALS     19              TRY TO TURN ON OVERFLOW                 FT030950
        TNO     *+2             CHECK FOR OVERFLOW                      FT030960
        ERROR   22,*+3          ARRAY TOO BIG FOR CORE                  FT030970
        REM                                                             FT030980
        ARS     2               RESTORE TO DECREMENT                    FT030990
1DIMN5  STO     .P+6            SIZE OF ARRAY                           FT031000
        REM                                                             FT031010
        SKIP    CMCHAR          TEST FOR COMMA (MORE SUBSCRIPTS)        FT031020
        TRA     *+3             NOT A COMMA                             FT031030
        TXH     ERR018,1,2      TOO MANY DIMENSIONS                     FT031040
        TXI     1DIMN1,1,1      BUMP COUNT AND LOOP BACK                FT031050
        REM                                                             FT031060
        STZ     .P+4,1          SET STOP CELL                           FT031070
        SKIP    RPCHAR          TEST FOR RIGHT PAREN                    FT031080
        ERROR   9,SKEND         NO PAREN                                FT031090
        REM                                                             FT031100
        PXD     ,1              GET NUMBER OF DIMENSIONS                FT031110
        LDQ     .P+6            GET TOTAL SIZE OF ARRAY                 FT031120
        RQL     3               MOVE INTO POSITION                      FT031130
        LGL     15              SET PREFIX AND ADDRESS                  FT031140
        ORA     .P+4            MOVE FIRST DIMENSION INTO DECREMENT     FT031150
        SLW     .P+6            SAVE FIRST WORD IN DIMENSION DECLARATION FT031160
        REM                                                             FT031170
        CAL     .P              TEST EACH ARRAY NAME TO SEE IF IT       FT031180
        LAS     =HFORMAT        IS 'FORMAT' OR 'IF'.  THIS IS TO SPEED  FT031190
        TRA     *+2             THE IF AND FORMAT PROCESSORS.           FT031200
        STL     FORFLG          FORMAT NAME FOUND.  SET THE FLAG        FT031210
        LAS     =HIF                                                    FT031220
        TRA     *+2                                                     FT031230
        STL     IFFLAG          IF ARRAY FOUND.  SET THE FLAG           FT031240
        XCL                                                             FT031250
        REM                                                             FT031260
        TSX     LOCATE,4        PUT IN SYMTAB                           FT031270
        REM                                                             FT031280
        STZ     .P+8            SET FLAG FOR OLD SYMBOL                 FT031290
        TPL     *+2             IS IT OLD SYMBOL.                       FT031300
        STL     .P+8            NEW SYMBOL.  SET FLAG                   FT031310
        REM                                                             FT031320
        PAX     ,4                                                      FT031330
        LFT     BARRY+BVARB+BEXTF+BINTF     ARE ANY OF THESE FLAGS ON.  FT031340
        TRA     1DIM13          YES.  ERROR.                            FT031350
        REM                                                             FT031360
        SIL     BARRY           NO.  TURN ON ARRAY FLAG                 FT031370
        CAL     .P              GET BCD NAME OF ARRAY                   FT031380
        LAS     =HI00000        NEW SYMBOL.  SET MODE BITS              FT031390
        TRA     *+1                                                     FT031400
        LAS     =HN(((((                                                FT031410
        SIL     MREAL           FLOATING POINT                          FT031420
        TRA     *+3                                                     FT031430
        SIL     MINTG           FIXED POINT                             FT031440
        TRA     1DIM17                                                  FT031450
        REM                                                             FT031460
        LDQ     MODFLG          GET COLUMN 1 FLAG CELL                  FT031470
        RQL     34              MOVE BIT 34 INTO MQ SIGN POSITION       FT031480
        TQP     1DIM17          IS IT 'D' OR 'I'.                       FT031490
        SIL     BDOUB           YES.  SET BDOUB BIT                     FT031500
        REM                                                             FT031510
1DIM17  ZET     .P+8            IS IT NEW SYMBOL.                       FT031520
        TRA     1DIM12+2        YES.                                    FT031530
        REM                                                             FT031540
```

```
         LFT     BCOMN           OLD SYMBOL.  IS IT IN COMMON.                  FT031550
         TRA     1DIM10          YES.                                           FT031560
         REM                                                                    FT031570
         LNT     BEQUV           NO.  IS IT IN EQUIVALENCE STACK.               FT031580
         TRA     1DIM12          NO.                                            FT031590
         REM                                                                    FT031600
         PIA                     YES.                                           FT031610
1DIMN8   LXA     DPOINT,1        GET DIMTBL POINTER                             FT031620
         STA     DIMTBL,1        PUT CORRECT POINTER IN ADDRESS                 FT031630
         PXA     ,1              RESTORE EQUIV ENTRY                            FT031640
1DIM11   STI     EQUIV,4                                                        FT031650
         STA     EQUIV,4                                                        FT031660
         TXI     *+1,1,1         BUMP DIMTBL POINTER                            FT031670
         REM                                                                    FT031680
1DIMN9   CAL     .P+6            INSERT CODE WORD IN DIMTBL                     FT031690
         SLW     DIMTBL,1                                                       FT031700
         TXI     *+1,1,1         BUMP POINTER                                   FT031710
         REM                                                                    FT031720
         NZT     .P+3            IS THERE MORE THAN ONE DIMENSION.              FT031730
         TRA     1DIMN6          NO.                                            FT031740
         REM                                                                    FT031750
         CAL     .P+2            GET THIRD DIMENSION OR ZERO                    FT031760
         ARS     18              SHIFT INTO ADDRESS                             FT031770
         ORA     .P+3            COMBINE WITH THE SECOND DIMENSION              FT031780
         SLW     DIMTBL,1        PUT IN DIMENSION STACK WITH ZERO PREFIX        FT031790
         TXI     *+1,1,1         BUMP COUNTER                                   FT031800
         REM                                                                    FT031810
1DIMN6   LNT     BCOMN           IS COMMON FLAG ON.                             FT031820
         SXA     DPOINT,1        NO.  UPDATE DIMTBL POINTER                     FT031830
         REM                                                                    FT031840
1DIM14   SKIP1   CMCHAR          CHECK FOR FURTHER ENTRIES                      FT031850
         TRA     1DIMN7          GET NEXT ENTRY                                 FT031860
         REM                                                                    FT031870
         SKIP    63              END OF STATEMENT REQUIRED                      FT031880
         ERROR   3,SKEND                                                        FT031890
         TRA     SKEND                                                          FT031900
         REM                                                                    FT031910
         REM                                                                    FT031920
1DIM12   LNT     BARGT           NOT IN EQUIVALENCE.  IS IT A PARAMETER.        FT031930
         TRA     1DIM15          NO. ERROR. SHOULD NOT BE AN OLD SYMBOL         FT031940
         PXA     ,4              YES.  TREAT AS NEW SYMBOL.                     FT031950
         TRA     1DIMN8                                                         FT031960
         REM                                                                    FT031970
1DIM15   CAL     SYMTAB,4        GET BCD NAME OF SYMBOL IN ERROR                FT031980
         SLW     FORM19+6        PUT IN THE ERROR MESSAGE                       FT031990
         ERROR   19,1DIM12+2     PUT OUT MESSAGE AND RETURN                     FT032000
         REM                                                                    FT032010
         REM                                                                    FT032020
1DIM10   PIA                     COMMON VARIABLE                                FT032030
         PAX.    ,1              GET DIMTBL POINTER                             FT032040
         TRA     1DIM11                                                         FT032050
         REM                                                                    FT032060
1DIM13   CAL     SYMTAB,4        GET BCD NAME OF ARRAY IN ERROR                 FT032070
         SLW     FORM21          PUT NAME INTO THE ERROR MESSAGES               FT032080
         SLW     FORM28+2                                                       FT032090
         SLW     FORM69                                                         FT032100
         REM                                                                    FT032110
         LFT     BARRY           IS NAME ALREADY IN DIMTBL AS ARRAY.            FT032120
         ERROR   21,*+1          YES                                            FT032130
         LFT     BEXTF+BINTF                                                    FT032140
```

```
ERROR   69,**1      NAME USED AS VARIABLE AND FUNCTION          FT032150
LFT     BASSN       IS VARIABLE IN ASSIGN-TYPE STATEMENTS.      FT032160
ERROR   28,**1      YES                                         FT032170
TRA     1DIM14      RETURN FOR NEXT ARRAY IN THE LIST           FT032180
```

```
*       EQUIVALENCE (A(I),B(J),...X(Z)), ....                          FT032200
        REM                                                            FT032210
*       ENTRY...  1EQUVX                                               FT032220
*       THE EQUIVALENCE TABLE EQUTBL IS COMPOSED OF 'CHAINS', WHICH    FT032230
*       IN TURN ARE FORMED FROM ONE OR MORE 'STACKS'.  A CHAIN CONTAINS FT032240
*       ALL SYMBOLS RELATED TO EACH OTHER THROUGH EQUIVALENCE          FT032250
*       DECLARATIONS.  A NEW STACK IS FORMED FROM THE NEW ITEMS IN A   FT032260
*       SINGLE EQUIVALENCE DECLARATION (THE PARENTHETICAL EXPRESSION   FT032270
*       IN THE EQUIVALENCE STATEMENT).  A SYMBOL APPEARS NO MORE THAN  FT032280
*       ONCE IN EQUTBL.                                                FT032290
        REM                                                            FT032300
*       A STACK IS FORMED AS ITEMS ARE PROCESSED FROM A DECLARATION LIST.FT032310
*       THE STACK ENTRIES ARE ONE WORD PER SYMBOL.  THE ADDRESS CONTAINS FT032320
*       THE SYMBOL TABLE POINTER, AND THE DECREMENT CONTAINS THE       FT032330
*       SUBSCRIPT (ONE, IF NO SUBSCRIPT IS GIVEN).  EACH STACK IS      FT032340
*       BEGUN AND ENDED WITH A CONTROL WORD CONTAINING A MINUS SIGN.   FT032350
*       THESE WORDS ARE USED TO DELIMIT A STACK AND, WHEN NECESSARY,   FT032360
*       TO LINK THIS STACK WITH OTHERS TO FORM A CHAIN.  THE LOCATION OF FT032370
*       EACH NEWLY BUILT STACK (UNLESS IT IS LINKED TO A FORMER STACK) ISFT032380
*       RECORDED IN THE ADDRESS OF A DIRECTOR WORD, WHICH IS STORED IN FT032390
*       POOL.  THE DECREMENT OF 'INITFG' POINTS TO THE FIRST OF THESE  FT032400
*       DIRECTOR WORDS, AND THE DECREMENT OF EACH DIRECTOR POINTS TO THE FT032410
*       NEXT ONE.  THE LAST ONE HAS A ZERO DECREMENT.                  FT032420
*       IF A SYMBOL ENCOUNTERED IN A LIST IS ALREADY IN AN EQUTBL      FT032430
*       CHAIN (NOT THE CURRENT CHAIN), THE DIFFERENCE IN OLD AND       FT032440
*       CURRENT SUBSCRIPTS (DEL) IS COMPUTED.  IF DEL IS NEGATIVE,     FT032450
*       THE MAGNITUDE OF DEL IS ADDED TO EACH MEMBER OF THE OLD CHAIN, FT032460
*       AND DEL1 IS SET TO ZERO.  IF DEL IS POSITIVE, DEL IS ADDED     FT032470
*       TO EACH MEMBER (IF ANY) OF THE CURRENT CHAIN, AND DEL1 IS SET  FT032480
*       EQUAL TO DEL.  DEL1 IS AN AMOUNT TO BE ADDED AUTOMATICALLY TO  FT032490
*       THE SUBSCRIPTS OF ALL FUTURE ITEMS IN THE CURRENT EQUIVALENCE  FT032500
*       LIST.  THE OLD CHAIN IS THEN LINKED TO THE CURRENT STACK       FT032510
*       (IF ANY), FORMING ONE LONGER CHAIN.  IF, IN THE CURRENT LIST,  FT032520
*       A SECOND LINKAGE TO THE PRESENT CHAIN (CONSISTING OF THE OLD   FT032530
*       CHAIN(S) AND THE CURRENT STACK LINKED TOGETHER) IS FOUND, THE  FT032540
*       OLD SUBSCRIPT IS COMPARED WITH THE EFFECTIVE CURRENT SUBSCRIPT  FT032550
*       (SUBSCRIPT PLUS DEL1).  IF THESE NUMBERS ARE NOT EQUAL, AN     FT032560
*       INCONSISTENCY ERROR IS RECORDED.                               FT032570
        REM                                                            FT032580
*       THE LINKING PROCESS MUST CONNECT STACKS IN THE SAME ORDER AS   FT032590
*       THEY WERE FORMED.  OTHERWISE, THE CORRECT ORDER OF VARIABLES IN FT032600
*       COMMON MIGHT BE ALTERED.                                       FT032610
*       THE DECREMENT FIELD OF THE TOP CONTROL WORD OF A STACK IS      FT032620
*       USED TO LINK THIS STACK WITH A STACK BUILT EARLIER IN EQUTBL.  FT032630
*       SIMILARLY, THE ADDRESS FIELD OF THE BOTTOM CONTROL WORD OF A   FT032640
*       STACK IS USED TO LINK TO A LATER STACK.                        FT032650
        REM                                                            FT032660
*       THE LOCATION OF A SYMBOL'S ENTRY IN EQUTBL IS RECORDED IN THE  FT032670
*       ADDRESS OF THE EQUIV TABLE WORD, UNLESS THE SYMBOL HAS ALREADY FT032680
*       APPEARED IN A COMMON, DIMENSION, ASSIGN, OR ASSIGNED GO TO     FT032690
*       STATEMENT.  IN THE LATTER CASES, THE EQUTBL POINTER IS PLACED IN FT032700
*       THE ADDRESS OF THE FIRST WORD IN THE COMMON/DIMENSION/ASSIGN   FT032710
*       ENTRY.  IN EITHER EVENT, THE FLAG BEQUV IS TURNED ON IN THE    FT032720
*       EQUIV TABLE ENTRY.                                             FT032730
        REM                                                            FT032740
*       A SUBPROGRAM PARAMETER MAY NOT APPEAR IN AN EQUIVALENCE        FT032750
*       DECLARATION.                                                   FT032760
        REM                                                            FT032770
*       NOTE... DPOINT AND EPOINT ARE SYNONYMOUS WITH NXTLOC, AND      FT032780
```

```
*       DIMTBL AND EQUTBL ARE SYNONYMOUS WITH POOL.                          FT032790
*       SPACE   3                                                            FT032800
*       ROUTINE FOR SCANNING EQUIVALENCE STATEMENTS.                         FT032810
*                                                                            FT032820
1EQUVX  AXC     11,2            RESET POINTER                                FT032830
        SKIP    LPCHAR          CHECK FOR LEFT PAREN                         FT032840
        ERROR   9,SKEND         NO PAREN                                     FT032850
        REM                                                                  FT032860
1EQUV3  STZ     PRIOR           RESET FLAG                                   FT032870
        STZ     LINKER                                                       FT032880
        REM                                                                  FT032890
1EQUV2  TSX     SCNBCD,4        GET NAME                                     FT032900
        TMI     ERROR8          NOT A VARIABLE                               FT032910
        SLW     .P              SAVE BCD NAME                                FT032920
        REM                                                                  FT032930
        LDQ     =1B17           SUBSCRIPT IS ONE                             FT032940
        STQ     .P+1                                                         FT032950
        SKIP    LPCHAR          CHECK FOR SUBSCRIPT                          FT032960
        TRA     STOEQU          NO PAREN, ERGO NO SUBSCRIPT GIVEN            FT032970
        REM                                                                  FT032980
        TSX     SCNBCD,4        GET SUBSCRIPT                                FT032990
        TPL     ERRO17          MUST BE NUMERICAL                            FT033000
        TSX     BCDFIX,4        CONVERT TO FIXED POINT NUMBER                FT033010
        SLW     .P+1            SAVE SUBSCRIPT                               FT033020
        REM                                                                  FT033030
        SKIP    RPCHAR          CHECK FOR RIGHT PAREN                        FT033040
        ERROR   9,SKEND         NO PAREN FOR SUBSCRIPT                       FT033050
        REM                                                                  FT033060
1EQUV1  TRA     STOEQU          PROCESS EQUIVALENCE OF SYMBOL                FT033070
        REM                                                                  FT033080
        SKIP1   CMCHAR          COMMA POSSIBLE IN THIS POSITION              FT033090
        TRA     1EQUV2          RETURN FOR ANOTHER VARIABLE                  FT033100
        REM                                                                  FT033110
        SKIP    RPCHAR          NO COMMA.  CHECK FOR RIGHT PAREN             FT033120
        ERROR   9,SKEND         NO ) EITHER.  ERROR.                         FT033130
        REM                                                                  FT033140
        ZET     LINKER          HAS A LINKAGE BEEN FOUND                     FT033150
        TRA     1EQU62          YES.  NO DIRECTOR WORD NEEDED                FT033160
        REM                                                                  FT033170
1EQU61  LXA     EPOINT,4        NO.  GET NEXT POOL LOCATION FOR DIRECTOR     FT033180
        CAL     TOP             GET TOP OF STACK JUST FINISHED               FT033190
        STA     EQUTBL,4        SET STACK LOCATION IN DIRECTOR WORD          FT033200
        LXA     INITFG,1        GET LAST DIRECTOR LOCATION                   FT033210
        SXA     INITFG,4        SET NEW DIRECTOR LOCATION                    FT033220
        PXD     ,4                                                           FT033230
        STD     EQUTBL,1        SET LINKER IN PREVIOUS DIRECTOR WORD         FT033240
        TXI     *+1,4,1         BUMP EQUTBL POINTER                          FT033250
        SXA     EPOINT,4                                                     FT033260
        REM                                                                  FT033270
1EQU62  SKIP1   63              MAY BE END OF STATEMENT                      FT033280
        TRA     SKEND           YES                                          FT033290
        REM                                                                  FT033300
        CAL     COLUMN,2        NO.  CHECK FOR ,( CHARACTERS                 FT033310
        ERA     =H,(                                                         FT033320
        ARS     24                                                           FT033330
        TNZ     ERROR7          BAD FORMAT                                   FT033340
        REM                                                                  FT033350
        TXI     1EQUV3,2,-2     RETURN FOR ANOTHER SET OF EQUIVALENCES       FT033360
```

```
        EJECT                                                           FT033370
*       ROUTINE TO HANDLE EQUIVALENCE TABLE ENTRIES.                    FT033380
*                                                                       FT033390
STOEQU  LDQ     .P              GET BCD SYMBOL                          FT033400
        TSX     LOCATE,4        PUT IN SYMTAB                           FT033410
        SLW     .P+2            SAVE EQUIV TABLE POINTER                FT033420
        TPL     1EQU20          OLD SYMBOL.  MODE ALREADY SET.          FT033430
        REM                                                             FT033440
*       NEW SYMBOL.  SET MODE BITS AND BEQUV FLAG.                      FT033450
        CAL     .P              NEW SYMBOL.  FIND MODE.                 FT033460
        LAS     =HI00000                                                FT033470
        TRA     *+1                                                     FT033480
        LAS     =HN(((((                                                FT033490
        SIL     MREAL+BEQUV     FLOATING POINT                          FT033500
        TRA     *+2                                                     FT033510
        SIL     MINTG+BEQUV     FIXED POINT                             FT033520
        TRA     1EQU26                                                  FT033530
        REM                                                             FT033540
        REM                                                             FT033550
*       OLD SYMBOL.                                                     FT033560
1EQU20  LFT     BARGT+BEXTF+BINTF                                       FT033570
        TRA     1EQU40          ILLEGAL FLAG                            FT033580
        REM                                                             FT033590
1EQU47  LFT     BEQUV           IS SYMBOL ALREADY IN EQUIVALENCE TABLE. FT033600
        TRA     1EQU21          YES.  GO TO SECTION 1.                  FT033610
        REM                                                             FT033620
        SIL     BEQUV           NO.  TURN ON BEQUV FLAG                 FT033630
        REM                                                             FT033640
*       NEW EQUTBL ENTRY.                                               FT033650
1EQU26  LXA     EPOINT,4        SET EQUTBL POINTER                      FT033660
        NZT     PRIOR           IS PRIOR FLAG ON                        FT033670
        TRA     1EQU23          NO.                                     FT033680
        REM                                                             FT033690
        TXI     *+1,4,-1        DROP POINTER                            FT033700
        CLA     .P+2            YES.  GET SYMTAB POINTER                FT033710
        ZET     LINKER          IS LINKER FLAG ON.                      FT033720
        ADD     DEL1            YES.  INCREASE SUBSCRIPT                FT033730
        REM                                                             FT033740
1EQU25  ADD     .P+1            ADD IN SUBSCRIPT                        FT033750
        SLW     EQUTBL,4        STORE EQUTBL ENTRY                      FT033760
        REM                                                             FT033770
        LXA     .P+2,1          GET SYMTAB POINTER                      FT033780
        STI     EQUIV,1         RESTORE EQUIV WORD                      FT033790
        LFT     BCOMN+BARRY+BASSN     IS SYMBOL IN POOL ALREADY.        FT033800
        TRA     1EQU51          YES.                                    FT033810
        REM                                                             FT033820
        PXA     ,4              NO.                                     FT033830
        STA     EQUIV,1         SET ADDRESS OF EQUIV WORD TO EQUTBL ENTRY FT033840
        TXI     1EQU52,4,1                                              FT033850
        REM                                                             FT033860
1EQU51  PIA                                                             FT033870
        PAX     ,1              GET DIMTBL POINTER                      FT033880
        PXA     ,4                                                      FT033890
        STA     DIMTBL,1        SET EQUTBL POINTER IN DIMTBL ENTRY      FT033900
        TXI     *+1,4,1                                                 FT033910
        REM                                                             FT033920
1EQU52  CLS     =0              SIGN BIT FOR BOTTOM LINK WORD           FT033930
        STO     EQUTBL,4        SET LINK WORD                           FT033940
        SXA     LO,4            SET LO = EPOINT                         FT033950
        TXI     *+1,4,1         BUMP POINTER                            FT033960
```

```
         SXA     TERMFG,4    SET CURRENT TERMINATION POINT OF EQUTBL      FT033970
         SXA     EPOINT,4    RESTORE POINTER                              FT033980
         TRA     1EQUV1+1    RETURN                                       FT033990
         REM                                                              FT034000
1EQU40   PAX     ,4          ILLEGAL FLAG.  GET SYMTAB POINTER            FT034010
         CAL     SYMTAB,4    GET BCD NAME OF VARAIBLE IN ERROR            FT034020
         SLW     FORM25+4    PUT NAME INTO ERROR MESSAGE                  FT034030
         SLW     FORM69                                                   FT034040
         LFT     BARGT                                                    FT034050
         ERROR   25,*+1      CANNOT BE ARGUMENT                           FT034060
         LFT     BEXTF+BINTF                                              FT034070
         ERROR   69,*+1      CANNOT BE SUBPROGRAM NAME                    FT034080
         TRA     1EQUV1+1                                                 FT034090
         REM                                                              FT034100
         REM                                                              FT034110
*        NEW EQUTBL ENTRY.  PRIOR FLAG OFF.                               FT034120
1EQU23   STL     PRIOR       TURN ON PRIOR FLAG                           FT034130
         LXA     TERMFG,1    GET CURRENT END OF EQUIVALENCE STACKS        FT034140
         SXD     *+2,1                                                    FT034150
         TNX     *+3,1,1     SKIP IF THIS IS THE BEGINNING OF ENTRIES     FT034160
         TXH     *+2,4,**    TEST EPOINT VERSUS TERMFG                    FT034170
         TXI     *+3,4,-1    EQUAL.  REDUCE EPOINT POINTER                FT034180
         REM                                                              FT034190
         CLS     =0          FLAG CELL                                    FT034200
         STO     EQUTBL,4    FLAG CELL FOR TOP OF THIS STACK              FT034210
         SXA     TOP,4       SET TOP OF THIS NEW STACK                    FT034220
         REM                                                              FT034230
         NZT     LINKER      IS LINKER FLAG ON.                           FT034240
         TXI     1EQU25-3,4,1 NO.                                         FT034250
         REM                                                              FT034260
1EQU24   LXA     LO,1        YES.  GET LO POINTER                         FT034270
         PXD     ,1          PUT IN DECREMENT                             FT034280
         STD     EQUTBL,4    SET (TOP) DECREMENT = LO                     FT034290
         REM                                                              FT034300
         CLA     TOP                                                      FT034310
         STA     EQUTBL,1    SET (LO) ADDRESS = TOP                       FT034320
         REM                                                              FT034330
         CLA     .P+2        GET EQUIV TABLE POINTER                      FT034340
         ORA     DEL1        INSERT SUBSCRIPT INCREMENT IN NEW EQUTBL WD  FT034350
         TXI     1EQU25,4,1  BUMP EPOINT AND TRANSFER                     FT034360
         REM                                                              FT034370
         REM                                                              FT034380
*        SECTION 1.  SYMBOL ALREADY IN EQUTBL.                            FT034390
         REM                                                              FT034400
1EQU21   PIA                                                              FT034410
         PAX     ,4          GET A POINTER FROM THE EQUIV WORD            FT034420
         LFT     BCOMN+BARRY+BASSN     IS SYMBOL IN POOL ALRAEDY.         FT034430
         CAL     DIMTBL,4    YES. GET WORD CONTAINING EQUTBL POINTER      FT034440
         REM                                                              FT034450
         PAX     ,4          EQUTBL POINTER                               FT034460
         NZT     LINKER      IS LINKER FLAG ON.                           FT034470
         TRA     1EQU29      NO.                                          FT034480
         REM                                                              FT034490
         SXA     .P+4,4      YES.  SAVE EQUTBL POINTER                    FT034500
         TXI     *+1,4,-1    SCAN UPWARD TO TOP OF STACK                  FT034510
1EQU53   CLA     EQUTBL,4    GET TABLE ENTRY                              FT034520
         TPL     *-2         NOT YET A LINK WORD                          FT034530
         REM                                                              FT034540
         ANA     =077777000000 SAVE DECREMENT ONLY                        FT034550
         TZE     1EQU54      TEST FOR ZERO DECREMENT                      FT034560
```

```
         REM                                                          FT034570
         PDX     ,4              RESET POINTER FOR NEXT SEGMENT OF STACK FT034580
         TXI     1EQU53,4,-1     CONTINUE THE SCAN                    FT034590
         REM                                                          FT034600
1EQU54   PXA     ,4              GET TOP OF THIS STACK                FT034610
         ERA     HI              COMPARE TOP OF STACK WITH HI         FT034620
         TZE     1EQU31          EQUAL                                FT034630
         REM                                                          FT034640
         REM                                                          FT034650
*        THIS ENTRY LINKS TWO DIFFERENT OLD STACKS IN EQUTBL.         FT034660
         REM                                                          FT034670
*        T1 RECORDS THE NEXT ENTRY INTO THE 'OTHER' STACK.            FT034680
*        WHEN ONE OF THE CHAINS HAS BEEN COMPLETELY PROCESSED, T1 WILL FT034690
*        BE SET TO ZERO.                                              FT034700
*        T3 IS USED TO TEMPORARILY HOLD A RESTARTING POINT IN         FT034710
*        ONE OF THE STACKS.                                           FT034720
*                                                                     FT034730
         LXA     .P+4,1          RESTORE EQUTBL POINTER               FT034740
         CLA     EQUTBL,1        GET TABLE ENTRY                      FT034750
         ANA     =077777000000       KILL ADDRESS FIELD               FT034760
         SUB     .P+1            LESS NEW SUBSCRIPT                   FT034770
         SUB     DEL1            LESS DEL1                            FT034780
         STO     DEL             NEW DEL                              FT034790
         REM                                                          FT034800
         TMI     *+3                                                  FT034810
         ADD     DEL1            SET DEL1 = DIFFERENCE IN SUBSCRIPTS  FT034820
         STO     DEL1                                                 FT034830
         REM                                                          FT034840
         LXA     HI,1            GET HI                               FT034850
         SXA     T3,1            SET T3 FOR ONE OF THE FOLLOWING CASES FT034860
         PXA     ,4              GET TOP OF STACK POINTER             FT034870
         LAS     HI              IS TOP OF STACK ABOVE HI POINT       FT034880
         STA     T1              NO.  SET ENTRY INTO 'OTHER STACK'.   FT034890
         TRA     1EQU38                                               FT034900
         REM                                                          FT034910
         SXA     T1,1            TOP OF STACK ABOVE HI.  SET T1 = HI  FT034920
         SXA     HI,4            HI = TOP OF STACK                    FT034930
         CLS     DEL             CHANGE SIGN OF DEL                   FT034940
         STO     DEL                                                  FT034950
         SXA     T3,4            T3 = HI                              FT034960
         REM                                                          FT034970
1EQU38   LXA     T3,4            RESET POINTER                        FT034980
         TXI     *+1,4,1         SCAN DOWN FROM T3                    FT034990
         CLA     EQUTBL,4        GET ENTRY                            FT035000
         TMI     1EQU35          LINK WORD FOUND                      FT035010
         REM                                                          FT035020
         CLA     DEL             GET DEL                              FT035030
         TMI     *+3             SKIP IF MINUS                        FT035040
         ADD     EQUTBL,4        BUMP SUBSCRIPT                       FT035050
         STO     EQUTBL,4                                             FT035060
         TXI     1EQU38+2,4,1                                         FT035070
         REM                                                          FT035080
1EQU35   PAX     ,1              LINK WORD.  TEST ADDRESS             FT035090
         PXA     ,1              SAVE ADDRESS ONLY                    FT035100
         XCA                     SAVE LINK WORD                       FT035110
         TXH     1EQU36,1,0      IS LINK ADDRESS ZERO.                FT035120
         REM                                                          FT035130
         NZT     T1              ADDRESS IS ZERO.  IS OTHER STACK DONE ALSO.FT035140
         TRA     1EQU32          YES.  SET 'LO' AND RETURN            FT035150
         REM                                                          FT035160
```

```
1EQU37 CLA     T1                                                    FT035170
       CLA     T1                                                    FT035180
       STA     EQUTBL,4        (LINK) ADDRESS = T1                   FT035190
       LXA     T1,1                                                  FT035200
       PXD     ,4                                                    FT035210
       STD     EQUTBL,1        (T1) DECREMENT = LINK                 FT035220
       SXA     T3,1            T3 = T1                               FT035230
       STQ     T1              T1 = (LINK) ADDRESS                   FT035240
       REM                                                           FT035250
       CLS     DEL             DEL = - DEL                           FT035260
       STO     DEL                                                   FT035270
       TRA     1EQU38                                                FT035280
       REM                                                           FT035290
1EQU36 NZT     T1              HAS ONE STACK BEEN COMPLETED YET.     FT035300
       TRA     1EQU39          YES.                                  FT035310
       REM                                                           FT035320
       PXA     ,1              GET NON-ZERO LINK ADDRESS             FT035330
       LAS     T1              COMPARE WITH T1                       FT035340
       TRA     1EQU37                                                FT035350
       PZE                     UNREACHABLE POINT                     FT035360
       REM                                                           FT035370
1EQU39 PAX     ,4              RESET POINTER TO NEXT STACK           FT035380
       TXI     1EQU38+2,4,1                                          FT035390
       REM                                                           FT035400
       REM                                                           FT035410
*      FIRST LINKAGE WITH EARLIER STACKS.                            FT035420
       REM                                                           FT035430
1EQU29 STL     LINKER          TURN ON LINKER FLAG                   FT035440
       CAL     EQUTBL,4        GET OLD ENTRY                         FT035450
       ANA     =077777000000       SAVE DECREMENT                    FT035460
       SUB     .P+1            SUBTRACT NEW SUBSCRIPT                FT035470
       CHS                                                           FT035480
       STO     DEL             SET NEW DEL                           FT035490
       SLW     DEL1            DEL1 = POSITIVE DEL                   FT035500
       TMI     *+2             SKIP IF DEL IS MINUS                  FT035510
       STZ     DEL1            SET DEL1 = ZERO                       FT035520
       SXA     .P+3,4          SAVE ENTRY POINTER                    FT035530
       REM                                                           FT035540
1EQU41 CLA     EQUTBL,4        SCAN UPWARD TO TOP OF THIS STACK      FT035550
       TMI     1EQU42          IS IT LINK WORD.                      FT035560
       REM                                                           FT035570
       CLA     DEL             NO.                                   FT035580
       TMI     *+3             NO ACTION IF DEL IS MINUS             FT035590
       ADD     EQUTBL,4        MODIFY SUBSCRIPT                      FT035600
       STO     EQUTBL,4                                              FT035610
       TXI     1EQU41,4,-1                                           FT035620
       REM                                                           FT035630
1EQU42 ANA     =077777000000   LINK WORD FOUND.  IS IT TOP OF STACK   FT035640
       TZE     *+3                                                   FT035650
       PDX     ,4              CONTINUE THE SCAN. TOP OF STACK NOT YET. FT035660
       TXI     1EQU41,4,-1                                           FT035670
       REM                                                           FT035680
       SXA     HI,4            HI = TOP OF STACK                     FT035690
       LXA     .P+3,4          SCAN DOWN FROM ENTRY.  RELOAD POINTER FT035700
       TXI     *+1,4,1                                               FT035710
       REM                                                           FT035720
1EQU43 CLA     EQUTBL,4                                              FT035730
       TMI     1EQU44          LINK WORD FOUND.                      FT035740
       REM                                                           FT035750
       CLA     DEL                                                   FT035760
```

```
         TMI     *+3                                                    FT035770
         ADD     EQUTBL,4                                               FT035780
         STO     EQUTBL,4                                               FT035790
         TXI     1EQU43,4,1                                             FT035800
         REM                                                            FT035810
1EQU44   ANA     =077777         LINK WORD FOUND.  SAVE ADDRESS FIELD   FT035820
         TZE     *+3             END OF STACK FOUND.                    FT035830
         PAX     ,4              CONTINUE THE SCAN DOWNWARD             FT035840
         TXI     1EQU43,4,1                                             FT035850
         REM                                                            FT035860
         ZET     PRIOR           END OF STACK FOUND.  IS PRIOR FLAG ON. FT035870
         TRA     1EQU45          YES.                                   FT035880
         REM                                                            FT035890
1EQU32   SXA     LO,4            NO.  LO = BOTTOM OF STACK              FT035900
         TRA     1EQUV1+1        RETURN                                 FT035910
         REM                                                            FT035920
1EQU45   CLA     TOP             PRIOR ON.  (BOT OF STACK) ADDRESS = TOPFT035930
         STA     EQUTBL,4                                               FT035940
         PXD     ,4                                                     FT035950
         LXA     TOP,4           (TOP) DECREMENT = BOTTOM OF STACK      FT035960
         STD     EQUTBL,4                                               FT035970
         REM                                                            FT035980
         CLA     DEL                                                    FT035990
1EQU46   TPL     1EQUV1+1        RETURN IF NO BUMPING OF NEW STACK IS NEEDEDFT036000
         REM                                                            FT036010
         TXI     *+1,4,1         STEP DOWN 'PRIOR' STACK                FT036020
1EQU48   CLA     EQUTBL,4                                               FT036030
         TMI     1EQUV1+1        BOTTOM OF STACK.  RETURN               FT036040
         ADD     DEL1            ADD DEL1 TO SUBSCRIPT                  FT036050
         STO     EQUTBL,4                                               FT036060
         TXI     1EQU48,4,1                                             FT036070
         REM                                                            FT036080
         REM                                                            FT036090
*        SECOND LINK WITH THIS STACK.  CHECK FOR CONSISTENCY.           FT036100
1EQU31   LXA     .P+4,1          RESTORE EQUTBL POINTER                 FT036110
         CLA     DEL1                                                   FT036120
         ADD     .P+1            ADD SUBSCRIPT                          FT036130
         ERA     EQUTBL,1                                               FT036140
         ANA     =077777000000       KILL THE ADDRESS FIELD             FT036150
         TZE     1EQUV1+1        RETURN.  FAIR WEATHER.                 FT036160
         REM                                                            FT036170
         CAL     .P              GET BCD NAME OF GUILTY SYMBOL          FT036180
         SLW     FORM26+10       PUT NAME INTO THE ERROR MESSAGE        FT036190
         ERROR   26,1EQUV1+1     PUT OUT MESSAGE AND RETURN             FT036200
```

```
        TTL     P A S S  1  P R O C E S S O R  -  F R E Q U E N C Y      FT036210
*       FREQUENCY ....                                                  FT036220
        REM                                                             FT036230
*       ENTRY...  1FREQX                                                FT036240
*       THIS STATEMENT IS IGNORED.                                      FT036250
        SPACE   3                                                       FT036260
1FREQX  TRA     SKEND           THIS STATEMENT IS JUST A NO-OP.         FT036270
```

```
         TTL     P A S S  1  P R O C E S S O R  -  C O N T I N U E      FT036280
*        CONTINUE                                                      FT036290
         REM                                                           FT036300
*        ENTRY...  ICONTX                                              FT036310
*        THIS STATEMENT IS TREATED AS A NO-OP.                         FT036320
         SPACE   3                                                     FT036330
ICONTX   AXC     8,2            RESET POINTER                         FT036340
         SKIP    63             END OF STATEMENT EXPECTED             FT036350
         ERROR   3,SKEND                                              FT036360
         TRA     SKEND                                                FT036370
```

```
        TTL     P A S S  1  P R O C E S S O R  -  G O  T O                FT036380
*       GO TO N                                                          FT036390
        REM                                                              FT036400
*       ENTRY...  1GOTOX                                                 FT036410
*       THE LABEL N IS PLACED IN THE SYMBOL TABLE AFTER IT IS CHECKED     FT036420
*       AGAINST THE CURRENT CARD LABEL.  THE LABEL N IS NOT              FT036430
*       PERMITTED TO EQUAL THE CURRENT CARD LABEL.  THE PATH-OF-FLOW     FT036440
*       FLAG 'BPATH' IS TURNED ON FOR THE LABEL N.  THE GENERATION       FT036450
*       OF THE CODE 'TRA N' IS DELAYED UNTIL THE LABEL ON THE            FT036460
*       NEXT EXECUTABLE STATEMENT IS FOUND.  IF THIS LABEL IS            FT036470
*       THE SAME AS N, NO CODE IS GENERATED.                             FT036480
        SPACE   3                                                        FT036490
1GOTOX  AXC     4,2             RESET POINTER                            FT036500
        AXT     1TRANS,4        SET ENTRY FOR LABEL ROUTINE              FT036510
        SXA     TRAFLG,4                                                 FT036520
        REM                                                              FT036530
        TSX     SCNBCD,4        GET LABEL N                              FT036540
        TPL     ERROR1          NOT A LABEL                              FT036550
        SLW     .P              SAVE BCD LABEL                           FT036560
        SKIP    63              END OF STATEMENT EXPECTED                FT036570
        ERROR   3,SKEND                                                  FT036580
        REM                                                              FT036590
        CLS     .P              GET BCD LABEL. SET 'REACHABLE LABEL' FLAG FT036600
        TSX     STOLBL,4        PUT N IS SYMTAB.  MINUS SIGN             FT036610
        SLW     1TRAN2          SAVE POINTER TO N                        FT036620
        LAS     CARD            COMPARE WITH CURRENT LABEL               FT036630
        TRA     *+2                                                      FT036640
        ERROR   6,*+1           INVALID TRANSFER                         FT036650
        TRA     SKEND                                                    FT036660
        REM                                                              FT036670
        REM                                                              FT036680
        REM                                                              FT036690
*       ROUTINE TO TEST IF LAST TRANSFER INSTRUCTION IS NEEDED.          FT036700
*       POINTER HAS BEEN SAVED IN CELL 1TRAN2.                           FT036710
*       ENTRY TO THIS ROUTINE WAS SET INTO TRAFLG BY PREVIOUS            FT036720
*       TRANSFER STATEMENT.  EXIT FROM THIS ROUTINE IS TO 1,4.           FT036730
*                                                                        FT036740
1TRANS  ZET     NOCODE          IS CODE STILL NECESSARY.                 FT036750
        TRA     1,4             NO.  RETURN                              FT036760
        CAL     1TRAN2          GET POINTER                              FT036770
        LAS     CARD            TEST AGAINST CURRENT LABEL               FT036780
        TRA     *+2                                                      FT036790
        TRA     1,4             NO TRANSFER INSTRUCTION NEEDED.  RETURN. FT036800
        REM                                                              FT036810
        SXA     1TRAN3,4                                                 FT036820
        CODEO   TRA,R           CODE IS...  TRA N                        FT036830
1TRAN3  AXT     **,4                                                     FT036840
        TRA     1,4                                                      FT036850
        REM                                                              FT036860
```

```
        TTL     P A S S  1  P R O C E S S O R - C O M P U T E D  G O  T OFT036870
*       COMPUTED GO TO (N1, N2, ... NM), I                                FT036880
        REM                                                               FT036890
*       ENTRY...  1COMGO                                                  FT036900
*       EACH LABEL N1, N2,...NM IS PUT IN THE SYMBOL TABLE AND THE        FT036910
*       PATH-OF-FLOW FLAG BPATH IS TURNED ON.  NONE OF THE LABELS MAY     FT036920
*       EQUAL THE CURRENT CARD LABEL.  THE VARIABLE I MUST BE             FT036930
*       FIXED POINT.  IF I IS NOT A PARAMETER, THE CODE 'LDC I,4'         FT036940
*       IS GENERATED.  IF I IS A SUBPROGRAM PARAMETER (I.E. BARGT         FT036950
*       IS ON), THE PROCESSOR TRANSFERS TO THE PROLOG ROUTINE.            FT036960
*       ON RETURN, THE CODE 'LDC **,4' IS GENERATED.                      FT036970
*       NEXT AN INSTRUCTION 'TRA *,4' IS PUT OUT.  THIS IS FOLLOWED       FT036980
*       BY A LIST OF TRANSFERS 'TRA NI' IN THE SAME ORDER AS THEY         FT036990
*       APPEAR IN THE STATEMENT.                                          FT037000
        SPACE   3                                                         FT037010
1COMGO  AXC     5,2             RESET POINTER                             FT037020
        AXT     1TRANS+1,4      SET EXIT FOR SUBSEQUENT LABEL ROUTINE     FT037030
        SXA     TRAFLG,4                                                  FT037040
        REM                                                               FT037050
        AXC     1,4             SET 'GO TO' COUNTER                       FT037060
1COMG1  SXA     .P,4            SAVE COUNTER                              FT037070
        TSX     SCNBCD,4        GET NI                                    FT037080
        TMI     *+2             IS IT A LABEL.                            FT037090
        ERROR   1,1COMG6        NO.  ERROR                                FT037100
        REM                                                               FT037110
        TSX     STOLBL,4        PUT N(I) IN SYMTAB.  MINUS SIGN.          FT037120
        LAS     CARD            COMPARE WITH CURRENT LABEL                FT037130
        TRA     *+2                                                       FT037140
        ERROR   6,*+1           IMPROPER TRANSFER                         FT037150
        REM                                                               FT037160
        LXA     .P,4            RESTORE COUNTER                           FT037170
        SLW     COLUMN,4        SAVE N POINTER                            FT037180
        REM                                                               FT037190
1COMG6  CAL     COLUMN,2        TEST FOR COMMA                            FT037200
        ERA     =H,                                                       FT037210
        ARS     30                                                        FT037220
        TNZ     *+3                                                       FT037230
        TXI     *+1,2,-1                                                  FT037240
        TXI     1COMG1,4,-1     COMMA FOUND                               FT037250
        REM                                                               FT037260
        SXD     1COMG2,4        SAVE NUMBER OF LABELS                     FT037270
        CAL     COLUMN,2        TEST FOR ), CHARACTERS                    FT037280
        ERA     =H),                                                      FT037290
        ARS     24                                                        FT037300
        TNZ     ERRO11          IMPROPER PUNCTUATION                      FT037310
        TXI     *+1,2,-2                                                  FT037320
        REM                                                               FT037330
        TSX     SCNBCD,4        GET I                                     FT037340
        TMI     ERROR8          NOT A VARIABLE                            FT037350
        SLW     FORM69          SAVE BCD NAME                             FT037360
        REM                                                               FT037370
        LAS     =HI00000        TEST FOR FIXED POINT                      FT037380
        TRA     *+1                                                       FT037390
        LAS     =HO00000                                                  FT037400
        TRA     *+1                                                       FT037410
        ERROR   12,SKEND                                                  FT037420
        REM                                                               FT037430
        TSX     SYMSTO,4        PUT I IN SYMTAB.                          FT037440
        SLW     .P+1            SAVE POINTER                              FT037450
        REM                                                               FT037460
```

```
        LFT     BEXTF+BINTF                                          FT037470
        ERROR   69,SKEND        MAY NOT BE SUBPROGRAM NAMES          FT037480
        REM                                                          FT037490
        SKIP    63              END OF STATEMENT EXPECTED            FT037500
        ERROR   3,SKEND                                              FT037510
        NOCODE                  TEST CODE FLAG                       FT037520
        REM                                                          FT037530
        LNT     BARGT           TEST FOR SUBROUTINE ARGUMENT         FT037540
        TRA     1COMG5          REGULAR VARIABLE                     FT037550
        REM                                                          FT037560
        CAL     .P+1            SUBROUTINE ARGUMENT.  GET I POINTER. FT037570
        TSX     PROLOG,4        FIX UP PROLOGUE                      FT037580
        CODEC   LDC4,N          CODE IS...  LDC **,4                 FT037590
        TRA     1COMG4                                               FT037600
        REM                                                          FT037610
1COMG5  CODECO  LDC4,.P+1,V      PROCEED.  CODE IS...  LDC I,4       FT037620
        REM                                                          FT037630
1COMG4  CODECO  TRA04,PCOUNT,P,XRFLAG     CODE IS...  TRA *,4        FT037640
        REM                                                          FT037650
        AXC     1,2             PUT OUT TRANSFER LIST                FT037660
1COMG3  CODECO  TRA(COLUMN,2)R      PICK UP LABEL POINTER            FT037670
1COMG2  TXL     SKEND,2,**      BRANCH WHEN FINISHED                 FT037680
        TXI     1COMG3,2,-1     BUMP POINTER AND RETURN              FT037690
```

```
*       ASSIGN LABEL TO NAME
        REM
*       ENTRY...  1ASSNX
*       THE ASSIGN VARIABLE MUST BE FIXED POINT, AND MAY NOT APPEAR IN
*       A DIMENSION STATEMENT.  IF THE SYMBOL 'NAME' HAS
*       NOT ALREADY APPEARED IN AN ASSIGN OR ASSIGNED GO TO STATEMENT
*       (I.E. BASSN AND BGOTO ARE BOTH OFF), A THREE WORD ENTRY IN
*       THE ASSIGN TABLE IS MADE.  THESE WORDS SHARE STORAGE IN THE
*       'POOL' AREA WITH A POSSIBLE THREE-WORD COMMON TABLE ENTRY.
*       WORD ONE CONTAINS THE EQUIV TABLE POINTER (OR THE EQUTBL POINTER
*       IF BEQUV IS ON).  THE DECREMENT OF THIS WORD IS AVAILABLE TO HOLD
*       POSSIBLE COMMON INFORMATION.  WORD TWO HAS THE
*       VARIABLE'S EQUIV TABLE POINTER IN THE ADDRESS FIELD.
*       WORD THREE HAS THE LABEL'S EQUIV TABLE POINTER IN THE ADDRESS.
*       THE ADDRESS OF THE EQUIV ENTRY FOR THE VARIABLE CONTAINS
*       THE ASSIGN TABLE ENTRY POINTER.
*       THE DECREMENT FIELD OF THE SECOND WORD OF THE PREVIOUS ASSIGN
*       TABLE (IF ANY) IS SET TO THE POINTER TO THE SECOND WORD OF
*       THIS CURRENT ASSIGN TABLE.  ALSO THE CELL ASNLCF, WHICH
*       HOLDS THE BEGINNING OF THE LAST-BUILT STACK, IS SET TO THIS
*       SAME VALUE.
        REM
*       IF BASSN OR BGOTO IS ON, THE ASSIGN TABLE IS SEARCHED TO
*       SEE IF THE LABEL ALREADY APPEARS IN THIS STACK.  IF IT DOES,
*       AND IF THE ENTRY HAS A MINUS SIGN, BPATH IS TURNED ON, SINCE
*       THE LABEL HAS APPEARED IN BOTH ASSIGN AND GO TO STATEMENTS.
*       IF THE LABEL DOES NOT APPEAR IN THE STACK, THEN THE LABEL
*       POINTER IS ADDED TO THE STACK WITH A PLUS SIGN, AND THE
*       LOCATION OF THE NEW ENTRY IS PLACED IN THE DECREMENT OF THE
*       PREVIOUS ENTRY.
*       IN ALL CASES, THE FLAG BASSN IS TURNED ON IN THE EQUIV TABLE
*       ENTRY FOR THE VARIABLE 'NAME'.
        REM
*       THE CODE 'AXT LABEL,4' IS PUT OUT.  IF NAME IS NOT A PARAMETER,
*       'SXA NAME,4' IS GENERATED.  IF NAME IS A PARAMETER, THE
*       PROLOG ROUTINE IS ENTERED.  ON RETURN, THE CODE 'SXA **,4'
*       IS PUT OUT.
        SPACE   3
1ASSNX  AXC     6,2             RESET POINTER
        STZ     .P+1            INITIALIZE CELL
        REM
        TSX     SCNBCD,4        GET LABEL
        TPL     ERROR1          NOT A LABEL
        REM
        SSP                     DON'T TURN ON BPATH.
        TSX     STOLBL,4        PUT IN SYMTAB
        SLW     .P              SAVE POINTER TO EQUIV
        REM
        CAL     COLUMN,2        CHECK FOR 'TO'
        ERA     =HTO
        ARS     24
        TNZ     ERROR7          INVALID FORM OF STATEMENT
        REM
        TXI     *+1,2,-2        BUMP POINTER
        TSX     SCNBCD,4        GET VARIABLE
        TMI     ERROR8          NOT A VARIABLE
        REM
        LAS     =HI00000        CHECK FOR FIXED POINT
        TRA     *+1
```

```
        LAS     =HN(((((                                                FT038300
        TRA     *+1                                                     FT038310
        ERROR   12,SKEND        NOT FIXED POINT                         FT038320
        REM                                                             FT038330
        TSX     SYMSTO,4        PUT NAME IN SYMBOL TABLE                FT038340
        SLW     .P+2            SAVE EQUIV POINTER                      FT038350
CALOVF  CAL     EQUIV-OVERFL      ADDRESS POINTS TO PSEUDO-EQUIV WORD   FT038360
STZOVF  STZ     EQUIV-OVERFL                                            FT038370
        REM                                                             FT038380
        REM                     PSEUDO-EQUIV WORD TO GET COMMON RELOCATION FT038390
OVERFL  VFD     018/MREAL+BCOMN,3/,015/00000                            FT038400
        PAX     ,4              GET EQUIV POINTER                       FT038410
        REM                                                             FT038420
        LFT     BARRY+BEXTF+BINTF                                       FT038430
        TRA     1ASS15          ILLEGAL FLAG                            FT038440
        REM                                                             FT038450
1ASS13  SKIP    63              END OF STATEMENT EXPECTED               FT038460
        ERROR   3,SKEND                                                 FT038470
        REM                                                             FT038480
        LXA     .P+2,4          RESTORE XR4                             FT038490
        PIA                                                             FT038500
        LFT     BASSN                                                   FT038510
        TRA     1ASSN1          NO.                                     FT038520
        REM                                                             FT038530
        SIL     BASSN+BLHSX     YES.  TURN ON ASSIGN AND DEFINITION FLAGS FT038540
        AXT     1ASS10+1,1      SET TRANSFER ADDRESS                    FT038550
        SXA     1ASS10,1                                                FT038560
        REM                                                             FT038570
*       PRODUCE INITIAL ENTRIES FOR A NEW ASSIGN STACK                  FT038580
1ASS11  STI     EQUIV,4         RESTORE EQUIV WORD                      FT038590
        LFT     BCOMN           IS SYMBOL IN COMMON.                    FT038600
        TRA     1ASSN9          YES.  TRANSFER                          FT038610
        REM                                                             FT038620
        LXA     NXTLOC,1        GET NEXT AVAILABLE LOCATION IN POOL     FT038630
        PXA     ,4              GET EQUIV TABLE POINTER                 FT038640
        LFT     BEQUV           IS SYMBOL IN EQUIVALENCE STACK.         FT038650
        PIA                     YES.  GET EQUTBL POINTER                FT038660
        STA     POOL,1          SET ASSIGN STACK ADDRESS                FT038670
        REM                                                             FT038680
        PXA     ,1              GET POOL POINTER                        FT038690
        STA     EQUIV,4         SET EQUIV WORD ADDRESS                  FT038700
        TXI     *+1,1,3         BUMP NXTLOC                             FT038710
        SXA     NXTLOC,1                                                FT038720
        REM                                                             FT038730
1ASSN9  ALS     18              MOVE ASSIGN TABLE POINTER TO DECREMENT  FT038740
        ADD     =1B17           BUMP                                    FT038750
        NZT     ASNFLG          IS THIS THE FIRST ASSIGN ENTRY.         FT038760
        TRA     1ASS12          YES.                                    FT038770
        REM                                                             FT038780
        LXD     ASNLCF,1        GET POINTER TO BEGINNING OF PREVIOUS STACK FT038790
        STD     POOL,1          SET LINKER TO CURRENT STACK IN FORMER STACKFT038800
        REM                                                             FT038810
1ASS14  STD     ASNLCF          BEGINNING OF LAST-BUILT STACK           FT038820
        PDX     ,1              GET START OF THIS STACK                 FT038830
        PXA     ,4              GET EQUIV TABLE POINTER                 FT038840
        SLW     POOL,1          SET IN ASSIGN TABLE                     FT038850
1ASS10  TRA     **              EXIT POINT FOR ASSIGNED GO TO ALSO      FT038860
        REM                                                             FT038870
        CAL     TASSNX          SET THE ASSIGN STATEMENT FLAG IN THE STACK FT038880
        ORS     POOL,1                                                  FT038890
```

```
         CAL     .P              GET LABEL POINTER                       FT038900
         SLW     POOL-1,1        PUT IN NEXT WORD IN ASSIGN ENTRY        FT038910
         TRA     1ASSN6                                                  FT038920
         REM                                                             FT038930
1ASS12   STD     ASNFLG          FIRST ASSIGN TABLE ENTRY.  SET FLAG CELLFT038940
         TRA     1ASS14                                                  FT038950
         REM                                                             FT038960
1ASSN1   SIL     BASSN+BLHSX     ASSIGN STACK ALREADY PRESENT.  TURN ON FLAGFT038970
         STI     EQUIV,4         RESTORE EQUIV WORD                      FT038980
         PAX     ,1                                                      FT038990
         TXI     *+1,1,2                                                 FT039000
         CAL     TASSNX          PUT FLAG IN THE STACK                   FT039010
         ORS     POOL+1,1                                                FT039020
         REM                                                             FT039030
1ASSN3   CAL     POOL,1          GET ELEMENT IN STACK                    FT039040
         STD     .P+1            SAVE LINKER                             FT039050
         ERA     .P              IS LABEL ALREADY IN STACK.              FT039060
         ANA     =077777         SAVE ADDRESS                            FT039070
         TZE     1ASSN4          YES.  LEAVE LOOP                        FT039080
         REM                                                             FT039090
         NZT     .P+1            NO.  IS THIS END OF STACK.              FT039100
         TRA     *+3             YES.                                    FT039110
         REM                                                             FT039120
         LXD     .P+1,1          NO.                                     FT039130
         TRA     1ASSN3          GET NEXT LABEL IN STACK                 FT039140
         REM                                                             FT039150
         CAL     NXTLOC          END OF STACK                            FT039160
         ALS     18              SET LINKER TO THIS LABEL                FT039170
         STD     POOL,1                                                  FT039180
         REM                                                             FT039190
1ASSN2   LXA     NXTLOC,1        SET POINTER                             FT039200
         CAL     .P              GET LABEL POINTER                       FT039210
         SLW     POOL,1          PUT IN ASSIGN STACK                     FT039220
         REM                                                             FT039230
         TXI     *+1,1,1         BUMP NXTLOC                             FT039240
         SXA     NXTLOC,1                                                FT039250
         REM                                                             FT039260
1ASSN6   STI     .P+3            SAVE INDICATORS                         FT039270
         CODECO  AXT4,.P,R,XRFLAG     CODE IS...  AXT LABEL,4            FT039280
         CODEC   PXA4,N           CODE IS... PXA 0,4                      FT039290
         CAL     .P+2            GET EQUIV POINTER                       FT039300
         REM                                                             FT039310
         LDI     .P+3            RESTORE INDICATORS                      FT039320
         LFT     BARGT           IS IT A PARAMETER.                      FT039330
         TRA     1ASSN5          YES.                                    FT039340
         REM                                                             FT039350
         CODEO   SLW,V           CODE IS...  SLW NAME,4                  FT039360
         TRA     SKEND                                                   FT039370
         REM                                                             FT039380
1ASSN5   TSX     PROLOG,4        SUBROUTINE ARGUMENT                     FT039390
         CODEC   SLW,N           CODE IS...  SLW **,4                    FT039400
         TRA     SKEND                                                   FT039410
         REM                                                             FT039420
1ASSN4   CLA     POOL,1                                                  FT039430
         TPL     1ASSN6          IS SIGN PLUS.                           FT039440
         REM                                                             FT039450
         PAX     ,4              GET EQUIV POINTER                       FT039460
         CAL     BPATHW                                                  FT039470
         ORS     EQUIV,4         TURN ON BPATH FLAG IN EQUIV WORD        FT039480
         TRA     1ASSN6                                                  FT039490
```

```
        REM
1ASS15  CAL     SYMTAB,4        ILLEGAL FLAG.  GET BCD NAME OF SYMBOL
        SLW     FORM28+2        PUT NAME INTO ERROR MESSAGE
        SLW     FORM69
        LFT     BARRY
        ERROR   28,*+1          MAY NOT BE ARGUMENT
        LFT     BEXTF+BINTF
        ERROR   69,*+1          MAY NOT BE SUBPROGRAM NAME
        TRA     1ASS13
        REM
BPATHW  VFD     018/BPATH
```

FT039500
FT039510
FT039520
FT039530
FT039540
FT039550
FT039560
FT039570
FT039580
FT039590
FT039600

```
        TTL     P A S S  1  P R O C E S S O R - A S S I G N E D  G O  T OFT039610
*       GO TO NAME, (N1, N2,..., NM)                                    FT039620
        REM                                                             FT039630
*       ENTRY...  1AGOTO                                                FT039640
*       THE ASSIGN STACK IS BUILT AS DESCRIBED IN THE ASSIGN STATEMENT. FT039650
*       IF A LABEL IN THE 'GO TO' LIST IS ALREADY IN THE STACK WITH A   FT039660
*       PLUS SIGN, BPATH IS TURNED ON, SINCE IT HAS NOW APPEARED IN     FT039670
*       BOTH ASSIGN AND GO TO STATEMENTS.  THE SIGN IS MADE MINUS.      FT039680
*       NEW LABELS IN THE STACK ARE GIVEN MINUS SIGNS.                  FT039690
*       NONE OF THE LABELS IN THE GO TO LIST ARE ALLOWED TO EQUAL       FT039700
*       THE CURRENT CARD LABEL.                                         FT039710
        REM                                                             FT039720
*       THE FLAG BGOTO IS TURNED ON IN THE EQUIV TABLE ENTRY            FT039730
*       FOR THE VARIABLE.                                               FT039740
        SPACE   3                                                       FT039750
1AGOTO  AXC     4,2             RESET POINTER                           FT039760
        STZ     .P+1            INITIALIZE TEMPORARY CELL               FT039770
        STZ     .P+4            INITIALIZE ERROR CELL                   FT039780
        REM                                                             FT039790
        AXT     1TRANS+1,4      SET NO-OP EXIT FROM NEXT CARD PROCESSOR FT039800
        SXA     TRAFLG,4                                                FT039810
        REM                                                             FT039820
        TSX     SCNBCD,4        GET VARIABLE NAME                       FT039830
        TMI     ERROR8          NOT A VARIABLE                          FT039840
        REM                                                             FT039850
        LAS     =HI00000        CHECK FOR FIXED POINT                   FT039860
        TRA     *+1                                                     FT039870
        LAS     =HN(((((                                                FT039880
        STL     .P+4            NOT FIXED POINT.  SET ERROR CELL ON.    FT039890
        ERROR   12,SKEND                                                FT039900
        REM                                                             FT039910
        TSX     SYMSTO,4        PUT NAME IN SYMBOL TABLE                FT039920
        PAX     ,4              GET EQUIV TABLE POINTER                 FT039930
        SLW     .P+2            SAVE EQUIV POINTER                      FT039940
        LFT     BARRY+BEXTF+BINTF                                       FT039950
        TRA     1AGO14          ILLEGAL FLAG                            FT039960
        REM                                                             FT039970
1AGO11  PIA                                                             FT039980
        LFT     BASSN           IS THE ASSIGN FLAG OFF.                 FT039990
        TRA     1AGOT3          NO.  ASSIGN STACK IS ALREADY PRESENT    FT040000
        REM                                                             FT040010
        SIL     BASSN           OFF.  BUILD NEW ASSIGN STACK            FT040020
        AXT     *+3,1           SET EXIT POINT FROM SECTION OF CODE     FT040030
        SXA     1ASS10,1             IN ASSIGN STATEMENT PROCESSOR.     FT040040
        TRA     1ASS11          GO TO THIS SECTION OF CODE              FT040050
        REM                                                             FT040060
        TXI     *+1,1,1         RETURN FROM OTHER CODE AT THIS POINT    FT040070
        CAL     =077777         SET LABEL ENTRY TO WIERD END-OF-STACK   FT040080
        SLW     POOL,1               CONDITION                          FT040090
        STZ     .P+3            ZERO ONLY FOR FIRST LABEL IN A STACK.   FT040100
        REM                                                             FT040110
1AGO12  CAL     TGOTOX          PUT FLAG INTO THE STACK                 FT040120
        ORS     POOL+1,1                                                FT040130
        REM                                                             FT040140
        CAL     COLUMN,2        CHECK FOR ,(                            FT040150
        ERA     =H,(                                                    FT040160
        ARS     24                                                      FT040170
        TNZ     ERROR7          INVALID PUNCTUATION                     FT040180
        SXA     1AGO15,1        SAVE START OF LABEL LIST                FT040190
        TXI     *+1,2,-2        BUMP POINTER                            FT040200
```

```
         REM                                                                FT040210
1AGOT4   TSX     SCNBCD,4    GET LABEL                                      FT040220
         TPL     ERROR1      NOT A LABEL                                    FT040230
         REM                                                                FT040240
         SSP                 DON'T TURN ON BPATH IN STOLBL.                 FT040250
         TSX     STOLBL,4    PUT IN SYMBOL TABLE                            FT040260
         SLW     .P          SAVE LABEL POINTER                             FT040270
         LAS     CARD        IS IT SAME AS CURRENT CARD.                    FT040280
         TRA     *+2         NO.                                            FT040290
         ERROR   6,*+1       YES.  ERROR                                    FT040300
         REM                                                                FT040310
1AGO15   AXT     **,1        RESET POINTER TO START OF LABEL LIST           FT040320
         LXA     .P,4        GET EQUIV POINTER                              FT040330
         ZET     .P+4        TEST ERROR CONDITION                           FT040340
         TRA     1AGOT6      ERROR FLAG ON.  SKIP TABLE BUILDING            FT040350
         REM                                                                FT040360
         REM                                                                FT040370
1AGOT8   CLA     POOL,1      GET LABEL ENTRY IN STACK                       FT040380
         STD     .P+1        SAVE LINKER                                    FT040390
         ERA     .P          COMPARE WITH CURRENT LABEL POINTER             FT040400
         ANA     =077777     SAVE ADDRESS POINTER                           FT040410
         TZE     1AGOT7      SAME.                                          FT040420
         REM                                                                FT040430
         NZT     .P+1        NOT THE SAME.  IS THIS END OF STACK.           FT040440
         TRA     *+3         YES.                                           FT040450
         REM                                                                FT040460
         LXD     .P+1,1      NO.  GET LINKER                                FT040470
         TRA     1AGOT8                                                     FT040480
         REM                                                                FT040490
         CLS     .P          GET LABEL POINTER.  MINUS SIGN                 FT040500
         NZT     .P+3        IS THIS THE FIRST LABEL ENTRY IN THE STACK.    FT040510
         TRA     1AGOT9      YES.                                           FT040520
         REM                                                                FT040530
         LXA     NXTLOC,4    GET POINTER TO NEXT OPEN POSITION IN STACK     FT040540
         STO     POOL,4      SET LABEL ENTRY INTO STACK                     FT040550
         REM                                                                FT040560
         PXD     ,4          GET POINTER                                    FT040570
         STD     POOL,1      SET LINKER INTO PRIOR LABEL ENTRY IN STACK     FT040580
         TXI     *+1,4,1     BUMP NXTLOC                                    FT040590
         SXA     NXTLOC,4    RESTORE NXTLOC                                 FT040600
         REM                                                                FT040610
1AGOT6   SKIP1   CMCHAR      CHECK FOR COMMA                                FT040620
         TRA     1AGOT4      COMMA FOUND.  RETURN FOR ANOTHER LABEL         FT040630
         REM                                                                FT040640
         CAL     COLUMN,2    NO COMMA.  CHECK FOR ) AND EOS                 FT040650
         ARS     24                                                         FT040660
         ERA     =03477                                                     FT040670
         TNZ     ERROR3      BAD PUNCTUATION                                FT040680
         REM                                                                FT040690
         ZET     .P+4        TEST ERROR FLAG                                FT040700
         TRA     SKEND       FLAG ON.  QUIT.                                FT040710
         REM                                                                FT040720
         CAL     .P+2        GET EQUIV POINTER                              FT040730
         LFT     BARGT       IS VARIABLE A PARAMETER.                       FT040740
         TRA     1AGO10      YES.                                           FT040750
         REM                                                                FT040760
         CODEO   TRAI,V      CODE IS...  TRA* NAME                          FT040770
         TRA     SKEND                                                      FT040780
         REM                                                                FT040790
1AGO10   TSX     PROLOG,4    PREPARE PROLOG TABLE ENTRY                     FT040800
```

```
        CODEC   TRAI,N          CODE IS...  TRA* **                     FT040810
1AGO13  TRA     SKEND           EXIT                                    FT040820
        REM                                                             FT040830
        REM                                                             FT040840
1AGOT7  CLS     POOL,1          LABEL FOUND IN STACK                    FT040850
        TPL     1AGOT6          SKIP IF LABEL HAD MINUS SIGN            FT040860
        REM                                                             FT040870
        SIL     BPATH           TURN ON BPATH IN EQUIV WORD             FT040880
        STI     EQUIV,4         RESTORE EQUIV WORD                      FT040890
1AGOT9  STO     POOL,1          SET SIGN OF LABEL ENTRY MINUS           FT040900
        STL     .P+3            THESE 2 INST. USED FOR 1ST LABEL IN STACK FT040910
        TRA     1AGOT6                                                  FT040920
        REM                                                             FT040930
1AGOT3  STL     .P+3            OLD ASSIGN STACK.  TURN ON STACK FLAG   FT040940
        PAX     ,1              PUT STACK POINTER IN INDEX              FT040950
        TXI     1AGO12,1,2      BUMP STACK POSITION TO FIRST LABEL ENTRY FT040960
        REM                                                             FT040970
1AGO14  CAL     SYMTAB,4        ILLEGAL FLAG.  GET BCD NAME OF SYMBOL   FT040980
        SLW     FORM28+2        PUT NAME INTO ERROR MESSAGE             FT040990
        SLW     FORM69                                                  FT041000
        LFT     BARRY                                                   FT041010
        ERROR   28,*+1          MAY NOT BE ARRAY                        FT041020
        LFT     BEXTF+BINTF                                             FT041030
        ERROR   69,*+1          MAY NOT BE SUBPROGRAM NAME              FT041040
        TRA     1AGO11                                                  FT041050
```

```
*       DO STATEMENT PROCESSOR                                          FT041070
        REM                                                             FT041080
*       ENTRY...    PDO                                                 FT041090
        REM                                                             FT041100
*       PRODUCES DO CODING AS FOLLOWS...                                FT041110
        REM                                                             FT041120
*       FOR  DO N  I=M1,M2,M3      WE GET                               FT041130
*              1.      LXD  M3,4     V FLAG                             FT041140
*              2.      SXD  .2,4     S FLAG                             FT041150
*              3.      LXD  M2,4     V FLAG                             FT041160
*              4.      SXD  *+4,4    P FLAG                             FT041170
*              5.      STZ  I        V FLAG                             FT041180
*              6.      LXD  M1,4     V FLAG                             FT041190
*              7.      TRA  *+2      + FLAG                             FT041200
*              8. .1   TXH  .3,4,** OR M2    S FLAG                     FT041210
*              9.      SXD  I,4      V FLAG                             FT041220
*                   ...                                                FT041230
*                   ...                                                FT041240
*             10.      LXD  I,4      V FLAG                             FT041250
*             11. .2   TXI  .1,4,** OR M3    P FLAG                     FT041260
*             13. .3   BSS  0                                           FT041270
        REM                                                             FT041280
*       IF M3=CONSTANT OR IS OMITTED, OMIT 1 AND 2 AND CHANGE 11 TO     FT041290
*       TXI  .1,4,M3                                                    FT041300
*       IF M2=CONSTANT OMIT 3 AND 4 AND COMPILE 8 AS TXH  .3,4,M2       FT041310
*       IF M1=CONSTANT CHANGE 6 TO AXT M1,4                             FT041320
        REM                                                             FT041330
*       FORMAT OF THE DOSTACK IS ...                                    FT041340
*                                                                       FT041350
*       WORD 3  POINTER TO TERMINAL LABEL                               FT041360
*       WORD 2  PZE  POINTER FOR INDEX,T,POINTER FOR .2 PS. CELL OR CONSTFT041370
*       WORD 1  PZE  PCOUNT FOR .1,,POINTER FOR .3 SP. CELL             FT041380
*                                                                       FT041390
*       WHERE T=0 DENOTES M3=CONSTANT                                   FT041400
*        AND T=1 DENOTES M3 NOT = CONSTANT                              FT041410
        SPACE   3                                                       FT041420
PDO     STL     DOFLAG          FLAG AS A DO SO THAT ON NEXT STATEMENT  FT041430
        REM                     CHECK CAN BE MADE THAT A NON-EXECUTABLE FT041440
        REM                     STATEMENT DOES NOT FOLLOE A DO.         FT041450
        LXA     DOPNTR,1                                                FT041460
        TXI     *+1,1,3                                                 FT041470
        TXH     PDO1,1,LDOSTK       DO STACK OVERFLOW                   FT041480
        SXA     DOPNTR,1                                                FT041490
        AXC     2,2                                                     FT041500
        TSX     SCNBCD,4                                                FT041510
        TPL     PDO3            ERROR SINCE LABEL NOT FOUND             FT041520
        STZ     DOSTAK+1,1      CLEAR OUT 2ND WORD OF DO STACK ENTRY    FT041530
        XCL                     IS LABEL ALREADY IN SYMTAB AS DEFINED   FT041540
        TSX     LOCATE,4                                                FT041550
        SLW     DOSTAK,1        SAVE LABEL IN DO-STACK                  FT041560
        TMI     PDO4            NO                                      FT041570
        LFT     BLHSX+MSTRG     IT'S NOT A NEW LABEL IN FACT .          FT041580
        ERROR   73,PDO2         IT IS ALREADY DEFINED.  POP UP DO STACK FT041590
        REM                     LABEL NOT ENCOUNTERED YET.  DO NOT ENTER ITFT041600
PDO4    TSX     SCNBCD,4        GET INDEX OF LOOP                       FT041610
        TSX     SYMSTA,4        ENTER INDEX IN SYMTAB ,IF NECESSARY. IF FT041620
        REM                     IT IS AN INTEGER VARIABLE. FLAG AS DEFINED FT041630
        LFT     BARGT           IS INDEX AN ARGUMENT                    FT041640
        ERROR   59,PDO2         YES.  NOT ALLOWED                       FT041650
```

```
        STA     DOSTAK+1,1      SAVE POINTER FOR INDEX                          FT041660
*                                                                               FT041670
*               IS INDEX OF THIS DO USED FOR INDEX IN ANY DO WITHIN WHICH       FT041680
*               THIS DO IS NESTED.  IF SO, NOT ALLLWED.  EXIT TO PDO2           FT041690
*               AND IGNORE THID DO.                                             FT041700
                                                                                FT041710
        SXA     *+1,1                                                           FT041720
        AXT     **,4                                                            FT041730
        TXI     *+1,4,-3                                                        FT041740
        TXL     *+5,4,0                                                         FT041750
        LAS     DOSTAK+1,4                                                      FT041760
        TXI     *-2,4,-3                                                        FT041770
        ERROR   86,PDO2                                                         FT041780
        TXI     *-4,4,-3                                                        FT041790
*                                                                               FT041800
        SKIP    EQCHAR          SKIP OVER = SIGN                                FT041810
        ERROR   60,PDO2         SOMETHING OTHER THAN = ENCOUNTERED              FT041820
        TSX     SCNBCD,4        GET M1                                          FT041830
        TMI     PDO9            M1=A CONSTANT                                   FT041840
        TSX     SYMSTB,4        CHECK FOR INTEGER VARIABLE AND ENTER IN         FT041850
        REM                     SYMTAB IF NOT ALREADY IN. DO NOT DEFINE         FT041860
        RIR     700000          KILL THE ARGUMENT BITS SO THAT THE              FT041870
        RIL     7               WORD M1VAR LOOKS LIKE AN ASTACK WORD            FT041880
        STI     M1VAR           TO THE CODE ROUTINE                             FT041890
        STA     M1VAR           SAVE POINTER TO M1                              FT041900
        STL     M1FLAG          FLAG M1 AS A VARIABLE                           FT041910
        SKIP    CMCHAR          SKIP OVER COMMA                                 FT041920
        ERROR   2,PDO2          NOT A COMMA                                     FT041930
        REM                     NOW GO GET M2                                   FT041940
PDO9A   TSX     SCNBCD,4                                                        FT041950
        TMI     PDO8            M2= A CONSTANT                                  FT041960
        TSX     SYMSTB,4        STORE M2 IN SYMTAB                              FT041970
        RIR     700000          KILL ARGUMENT BITS                              FT041980
        RIL     7                                                               FT041990
        STI     M2VAR           SAVE M2 EQUIVALENT                              FT042000
        STA     M2VAR                                                           FT042010
        STL     M2FLAG          FLAG M2 AS A VARIABLE                           FT042020
        SKIP    CMCHAR          SKIP OVER POSSIBLE COMMA                        FT042030
        TRA     PDO15           NOT A COMMA. MUST BE OCTAL 77.                  FT042040
        REM                                                                     FT042050
PDO8A   TSX     SCNBCD,4        GET M3                                          FT042060
        TMI     PDO7            M3= A CONSTANT                                  FT042070
        TSX     SYMSTB,4        M3 = VARIABEL                                   FT042080
        RIR     700000          RESET THE ARGUMENT BITS                         FT042090
        RIL     7                                                               FT042100
        STI     M3VAR           SAVE THE M3 EQUIVALENT                          FT042110
        STA     M3VAR                                                           FT042120
        CODE    LXD4,M3VAR                                                      FT042130
        TSX     NEXTLP,4        GET A CELL FMRO THE POOL                        FT042140
        ALS     18                                                              FT042150
        STD     DOSTAK+1,1      SAVE ITS LOCATION IN THE DOSTACK                FT042160
        ARS     18                                                              FT042170
        ORA     SXD4                                                            FT042180
        TSX     CITBLD,4,R                                                      FT042190
        TRA     PDO6                                                            FT042200
        REM                                                                     FT042210
PDO7    TSX     BCDFIX,4        CONVERT M3 TO OCTAL                             FT042220
PDO7A   ORA     =0100000                                                        FT042230
        STD     DOSTAK+1,1                                                      FT042240
        STT     DOSTAK+1,1                                                      FT042250
        TRA     PDO6
```

```
        REM                                                          FT042260
PD08    STZ     M2FLAG          FLAG M2 AS A CONSTANT                FT042270
        TSX     BCDFIX,4        CONVERT BCD CONSTANT TO OCTAL        FT042280
        SLW     M2VAR           SAVE                                 FT042290
        TRA     PD08A-2                                              FT042300
        REM                                                          FT042310
PD09    STZ     M1FLAG          M1 =CONSTANT                         FT042320
        TSX     BCDFIX,4                                             FT042330
        ARS     18                                                   FT042340
        SLW     M1VAR                                                FT042350
        TRA     PD09A-2                                              FT042360
        REM                                                          FT042370
PD06    SKIP    63              MUST BE END OF STATEMENT             FT042380
        ERROR   3,PD02                                               FT042390
        REM                                                          FT042400
        NZT     M2FLAG          IS M2 A VARIABLE.                    FT042410
        TRA     PD010           NO                                   FT042420
        CODE    LXD4,M2VAR      YES. TURN OUT INSTRUCTIONS 3 AND 4   FT042430
        CAL     SXD44                                                FT042440
        ACL     PCOUNT                                               FT042450
        TSX     CITBLD,4,P                                           FT042460
        REM                                                          FT042470
PD010   CAL     DOSTAK+1,1                                           FT042480
        ANA     =077777                                              FT042490
        ORA     STZ                                                  FT042500
        TSX     CITBLD,4,V                                           FT042510
        NZT     M1FLAG          IS M1 A CONSTANT                     FT042520
        TRA     PD011A          YES. PUT OUT AXT INSTEAD             FT042530
        CODE    LXD4,M1VAR                                           FT042540
PD011C  CAL     TRA2                                                 FT042550
        ACL     PCOUNT                                               FT042560
        TSX     CITBLD,4,P                                           FT042570
        TRA     PD011B                                               FT042580
PD011A  CAL     M1VAR           M1 A CONSTANT. TURN OUT THE AXT      FT042590
        ORA     AXT4                                                 FT042600
        TSX     CITBLD,4,N                                           FT042610
        TRA     PD011C                                               FT042620
        REM                     DEFINE LOCATION .1                   FT042630
PD011B  CAL     PCOUNT                                               FT042640
        STA     DOSTAK+2,1                                           FT042650
        TSX     NEXTLP,4                                             FT042660
        ALS     18                                                   FT042670
        STD     DOSTAK+2,1                                           FT042680
        ARS     18                                                   FT042690
        NZT     M2FLAG          IS M2 A VARIABLE                     FT042700
        ORA     M2VAR           NO. OR IN THE CONSTANT DECREMENT     FT042710
        ORA     TXH4                                                 FT042720
        ORS     XRFLAG                                               FT042730
        TSX     CITBLD,4,R                                           FT042740
        CAL     DOSTAK+1,1                                           FT042750
        ANA     =077777                                              FT042760
        ORA     SXD4                                                 FT042770
        TSX     CITBLD,4,V                                           FT042780
        TRA     SKEND                                                FT042790
*                                                                    FT042800
PD015   CAL     =1B17           M3 = 1                               FT042810
        TRA     PD07A                                                FT042820
        REM                     IN CASE OF ERRORS                    FT042830
PD03    ERROR   61,PD02                                              FT042840
PD01    ERROR   62,PD02                                              FT042850
```

```
        REM     BUMP DO STACK BACK DOWN                         FT042860
                                                                FT042870
PDO2    LXA     DOPNTR,4                                        FT042880
        TXI     *+1,4,-3                                        FT042890
        SXA     DOPNTR,4                                        FT042900
        TRA     SKEND           IGNORE THIS DO
```

```
        EJECT                                                           FT042910
*       DO TERMINATION ROUTINE                                          FT042920
        REM                                                             FT042930
*       DO TERMINATION ROUTINE.  TURNS OUT THE LXD 2,4 AND TXI .1,4,** ORFT042940
*       M3 FOR DO LOOP, THEN LOOKS IN DOSTACK TO SEE IF ANY OTHER DO'S  FT042950
*       TERMINATE ON THIS SAME LABEL STATEMENT.  IF SO THESE DO LOOPS   FT042960
*       ARE ALSO HANDLED IN THE SAME WAY.                               FT042970
        REM                                                             FT042980
*       SPECIAL CONSTANT CELLS .2 AND .3 ARE DEFINED HERE.              FT042990
        SPACE   3                                                       FT043000
DOTERM  SXA     DOTRMX,4                                                FT043010
        SXA     DOTRMX+1,1                                              FT043020
        LXA     DOPNTR,1                                                FT043030
DOTRM1  CAL     DOSTAK+1,1                                              FT043040
        PAX     ,4              GET THE EQUIV POINTER                   FT043050
        LDI     EQUIV,4         AND GET THE EQUIV WORD                  FT043060
        RIL     BDOIX           TURN OFF THE DO INDEX FLAG BIT          FT043070
        STI     EQUIV,4         AND RESET THE EQUIV WORD                FT043080
        REM                                                             FT043090
        ANA     =077777                                                 FT043100
        ORA     LXD4            TURN OUT THE LXD  I,4                   FT043110
        TSX     CITBLD,4,V                                              FT043120
        LDI     DOSTAK+1,1                                              FT043130
        RFT     100000                                                  FT043140
        TRA     DOTRM2          INCREMENT IS A CONSTANT                 FT043150
        REM                                                             FT043160
        REM     NOW DEFINE SPECIAL CONSTANT  .2                         FT043170
        REM                                                             FT043180
        CAL     DOSTAK+1,1                                              FT043190
        PDX     ,4                                                      FT043200
        CAL     PCOUNT                                                  FT043210
        SLW     EQUIV,4                                                 FT043220
        REM                                                             FT043230
        CAL     DOSTAK+2,1                                              FT043240
        ANA     =077777                                                 FT043250
DOTRM4  ORA     TXI4            TURN OUT THE TXI .1,4,**                FT043260
        TSX     CITBLD,4,P                                              FT043270
        CAL     DOSTAK+2,1                                              FT043280
        PDX     ,4                                                      FT043290
        CAL     PCOUNT                                                  FT043300
        SLW     EQUIV,4                                                 FT043310
        TXI     **+1,1,-3                                               FT043320
        SXA     DOPNTR,1                                                FT043330
        TXL     DOTRMX,1,0      IS DO STACK EMPTY                       FT043340
        CAL     DOSTAK-3,1      NO                                      FT043350
        ERA     DOSTAK,1                                                FT043360
        TZE     DOTRM1          NEXT DO ALSO TERMINATES HERE            FT043370
DOTRMX  AXT     **,4                                                    FT043380
        AXT     **,1                                                    FT043390
        TRA     1,4                                                     FT043400
        REM                                                             FT043410
        REM                     CODING TO TURN OUT TXI .1,4,M2          FT043420
        REM                                                             FT043430
DOTRM2  CAL     DOSTAK+1,1                                              FT043440
        ANA     =077777000000                                           FT043450
        SLW     DOTRM3                                                  FT043460
        CAL     DOSTAK+2,1                                              FT043470
        ANA     =077777                                                 FT043480
        ORA     DOTRM3                                                  FT043490
        TRA     DOTRM4                                                  FT043500
```

```
        TTL     P A S S  1  P R O C E S S O R  -  I F                    FT043510
*       IF (X)  N1,N2,N3                                                FT043520
        REM                                                             FT043530
*       ENTRY...  1IFXXX                                                FT043540
*       THE 'IF' STATEMENT PROCESSOR TRANSFERS INITIALLY TO THE         FT043550
*       COMPIL ROUTINE TO GENERATE THE CODE FOR EVALUATING THE          FT043560
*       PARENTHETICAL ALGEBRAIC OR LOGICAL EXPRESSION.  ON RETURN       FT043570
*       FROM COMPIL, EACH LABEL IS PLACED IN THE SYMBOL TABLE AFTER     FT043580
*       IT IS TESTED AGAINST THE CURRENT CARD LABEL.  NONE OF THE       FT043590
*       LABELS N1, N2, N3 MAY EQUAL THE CURRENT CARD LABEL.             FT043600
*       THE PATH-OF-FLOW FLAG 'BPATH' IS TURNED ON FOR EACH LABEL.      FT043610
*       THE GENERATION OF THE CODE IS DELAYED UNTIL THE LABEL ON THE    FT043620
*       NEXT EXECUTABLE STATEMENT IS OBTAINED.  THEN THE ROUTINE        FT043630
*       1TRIFF IS ENTERED, AND THE NECESSARY CODE PUT OUT.              FT043640
        SPACE   3                                                       FT043650
1IFXXX  AXC     2,2             RESET POINTER TO PAREN                  FT043660
        TSX     CMPLIF,4        PROCESS ARITHMETIC STATEMENT.  RETURN   FT043670
        REM                     WITH POINTER AT N1.                     FT043680
        AXT     1TRIFF,4        SET ENTRY FOR LABEL ROUTINE             FT043690
        SXA     TRAFLG,4                                                FT043700
        AXT     3,1             SET LOOPING INDEX                       FT043710
        REM                                                             FT043720
1IFXX2  TSX     SCNBCD,4        GET N(I)                                FT043730
        TPL     ERROR1          NOT A LABEL                             FT043740
        REM                                                             FT043750
        TSX     STOLBL,4        PUT N(I) IN SYMTAB.  MINUS SIGN.        FT043760
        SLW     .STT+4,1        SAVE POINTER IN SOME AVAILABLE CELLS    FT043770
        LAS     CARD            COMPARE WITH CURRENT LABEL              FT043780
        TRA     *+2                                                     FT043790
        ERROR   6,*+1           IMPROPER TRANSFER                       FT043800
        REM                                                             FT043810
        TNX     1IFXX1,1,1      ALL DONE                                FT043820
        REM                                                             FT043830
        SKIP    CMCHAR          CHECK FOR COMMA                         FT043840
        ERROR   2,SKEND         NO COMMA                                FT043850
        TRA     1IFXX2                                                  FT043860
        REM                                                             FT043870
1IFXX1  SKIP    63              END OF STATEMENT EXPECTED               FT043880
        ERROR   3,SKEND                                                 FT043890
        TRA     SKEND           QUIT FOR NOW                            FT043900
        SPACE   8                                                       FT043910
*       ROUTINE TO GENERATE CODE FOR THE IF STATEMENT.  THIS SECTION    FT043920
*       IS ENTERED FROM THE PASS 1 DRIVER AFTER THE LABEL ON THE NEXT   FT043930
*       EXECUTABLE STATEMENT HAS BEEN OBTAINED.                         FT043940
        REM                                                             FT043950
*       ONLY NECESSARY CODE IS PRODUCED.                                FT043960
        SPACE   3                                                       FT043970
1TRIFF  SXA     1TRIFX,4        SAVE XR4                                FT043980
        CAL     .STT+1          GET MINUS LABEL                         FT043990
        ERA     .STT+3          COMPARE WITH PLUS LABEL                 FT044000
        TNZ     1TRIF1          NOT EQUAL                               FT044010
        CAL     .STT+2          SAME.  GET ZERO LABEL                   FT044020
        ERA     CARD            IS IT SAME AS CURRENT STATEMENT LABEL.  FT044030
        TNZ     1TRIF2          NO                                      FT044040
        CAL     TNZ             YES                                     FT044050
        TRA     1TRIF5+1                                                FT044060
        REM                                                             FT044070
1TRIF2  CODECO  TZE,.STT+2,R    CODE IS...  TZE N2                      FT044080
        CAL     .STT+1          GET MINUS LABEL                         FT044090
        ERA     CARD            IS IT SAME AS CURRENT CARD LABEL.       FT044100
```

```
        TZE     1TRIFX          YES.  EXIT                              FT044110
        REM                                                             FT044120
1TRIF5  CODECO  TRA,.STT+1,R    CODE IS...  TRA N1                      FT044130
        REM                                                             FT044140
1TRIFX  AXT     **,4                                                    FT044150
        TRA     1,4                                                     FT044160
        REM                                                             FT044170
1TRIF1  CODECO  TZE,.STT+2,R    CODE IS...  TZE N2                      FT044180
        CAL     .STT+3          GET PLUS LABEL                          FT044190
        ERA     CARD                                                    FT044200
        TNZ     1TRIF3          NOT SAME AS CURRENT LABEL               FT044210
        CAL     TMI             SAME.  CODE IS...  TMI N1               FT044220
        TRA     1TRIF5+1                                                FT044230
        REM                                                             FT044240
1TRIF3  CAL     TPL             CODE IS...  TPL N3                      FT044250
        ORA     .STT+3                                                  FT044260
        TRA     1TRIF2+2                                                FT044270
```

```
*       IF DIVIDE CHECK N1, N2                                          FT044290
        REM                                                             FT044300
*       ENTRY...  1IFDIV                                                FT044310
*       NEITHER N1 NOR N2 MAY EQUAL THE CURRENT CARD LABEL.             FT044320
*       THE INITIAL CODE GENERATED IS 'DCT' AND 'TRA N1'.  THE CODE     FT044330
*       'TRA N2' IS DELAYED AND IS GENERATED ONLY IF THE NEXT EXECUTABLE FT044340
*       STATEMENT IS NOT LABELED N2.                                    FT044350
        SPACE   3                                                       FT044360
1IFDIV  AXC     13,2                                                    FT044370
        STZ     .P              ZERO FLAG                               FT044380
        TRA     1IFAC1                                                  FT044390
```

```
*       IF QUOTIENT OVERFLOW N1, N2                                     FT044410
        REM                                                             FT044420
*       ENTRY...  1IFQUO                                                FT044430
*       PROCESSING FOR THIS STATEMENT IS IDENTICAL TO THAT FOR THE      FT044440
*       'IF ACCUMULATOR OVERFLOW' STATEMENT.                            FT044450
        SPACE   3                                                       FT044460
1IFQUO  AXC     18,2            RESET POINTER                           FT044470
        TRA     *+2                                                     FT044480
```

```
*       IF ACCUMULATOR OVERFLOW N1, N2                                  FT044500
        REM                                                             FT044510
*       ENTRY...  1IFACC                                                FT044520
*       NEITHER N1 NOR N2 MAY EQUAL THE CURRENT CARD LABEL.             FT044530
*       IF N1 = N2, THE CODE 'STZ OVFLOW' IS GENERATED, WHERE 'OVFLOW'  FT044540
*       IS CELL 77462 OCTAL.  IF N1 IS NOT EQUAL TO N2, THE CODE        FT044550
*       'CAL OVFLOW', 'STZ OVFLOW', 'TNZ N1' IS GENERATED.  THE         FT044560
*       GENERATION OF THE FINAL INSTRUCTION 'TRA N2' IS DELAYED UNTIL   FT044570
*       IT IS ASCERTAINED THAT THE NEXT EXECUTABLE STATEMENT IS NOT     FT044580
*       LABELED N2.                                                     FT044590
        SPACE   3                                                       FT044600
1IFACC  AXC     21,2            RESET POINTER                           FT044610
        STL     .P              NON-ZERO FLAG                           FT044620
        REM                                                             FT044630
1IFAC1  AXT     1TRANS,4        SET ENTRY FOR LABEL ROUTINE             FT044640
        SXA     TRAFLG,4                                                FT044650
        REM                                                             FT044660
        TSX     SCNBCD,4        GET N1                                  FT044670
        TPL     ERROR1          NOT A LABEL                             FT044680
        REM                                                             FT044690
        TSX     STOLBL,4        PUT N1 IN SYMTAB.  MINUS SIGN.          FT044700
        LAS     CARD            COMPARE WITH CURRENT LABEL              FT044710
        TRA     *+2                                                     FT044720
        ERROR   6,*+1           IMPROPER TRANSFER                       FT044730
        REM                                                             FT044740
        SLW     .P+1            SAVE N1 POINTER                         FT044750
        SKIP    CMCHAR          COMMA EXPECTED                          FT044760
        ERROR   2,SKEND                                                 FT044770
        REM                                                             FT044780
        TSX     SCNBCD,4        GET N2                                  FT044790
        TPL     ERROR1          NOT A LABEL                             FT044800
        SLW     .P+3            SAVE BCD N2                             FT044810
        REM                                                             FT044820
        TSX     STOLBL,4        PUT N2 IN ST.  MINUS SIGN               FT044830
        LAS     CARD            COMPARE WITH CURRENT LABEL              FT044840
        TRA     *+2                                                     FT044850
        ERROR   6,*+1           IMPROPER TRANSFER                       FT044860
        REM                                                             FT044870
        SLW     1TRAN2          SAVE N2 POINTER                         FT044880
        REM                                                             FT044890
        SKIP    63              END OF STATEMENT EXPECTED               FT044900
        ERROR   3,SKEND                                                 FT044910
        NOCODE                  TEST CODE FLAG                          FT044920
        REM                                                             FT044930
        CAL     .P              TEST FLAG                               FT044940
        TNZ     1IFAC2                                                  FT044950
        CODEC   DCT,N           FOR DIVIDE.  CODE IS...  DCT            FT044960
        CODECO  TRA,.P+1,R                               TRA N1         FT044970
        TRA     SKEND                                                   FT044980
        REM                                                             FT044990
1IFAC2  CAL     .P+1            OVERFLOW.  GET N1                       FT045000
        ERA     1TRAN2          TEST N1 = N2.                           FT045010
        TNZ     1IFAC4          NO                                      FT045020
        CODEC   STZOVF,D        YES.  CODE IS...  STZ OVERFLOW.CELL     FT045030
        TRA     SKEND                                                   FT045040
        REM                                                             FT045050
1IFAC4  CODEC   CALOVF,D        CODE IS...  CAL OVERFLOW.CELL           FT045060
        CODEC   STZOVF,D                    STZ OVERFLOW.CELL           FT045070
        CODECO  TNZ,.P+1,R                  TNZ N1                      FT045080
```

```
         TTL P A S S  1  P R O C E S S O R  -  I F  S E N S E  S W I T C HFT045110
*        IF (SENSE SWITCH I) N1,N2                                       FT045120
         REM                                                             FT045130
*        ENTRY...  1IFSWT                                                FT045140
*        THE NUMBER I IS CHECKED TO ASSURE THAT IT IS BETWEEN 1 AND 6    FT045150
*        INCLUSIVE.  THE LABELS N1 AND N2 ARE TESTED TO DETERMINE        FT045160
*        THAT NEITHER IS EQUAL TO THE CURRENT CARD LABEL.  N1 AND N2     FT045170
*        ARE PUT IN THE SYMBOL TABLE AND THE PATH-OF-FLOW FLAG 'BPATH'   FT045180
*        IS TURNED ON.  THE CODE 'SWT I' IS GENERATED, FOLLOWED BY       FT045190
*        'TRA N2'.  THE GENERATION OF THE FINAL INSTRUCTION 'TRA N1'     FT045200
*        IS DELAYED UNTIL THE LABEL ON THE NEXT EXECUTABLE STATEMENT     FT045210
*        IS FOUND.  IF THIS LABEL IS THE SAME AS N1, NO CODE IS          FT045220
*        GENERATED.                                                      FT045230
         SPACE   3                                                       FT045240
1IFSWT   AXC     16,2          RESET POINTER AT N1                      FT045250
         STZ     .P            ZERO FLAG                                FT045260
         TRA     *+3                                                    FT045270
```

```
*       IF (SENSE LIGHT I) N1,N2                                        FT045290
        REM                                                             FT045300
*       ENTRY...  1IFSLT                                                FT045310
*       THE NUMBER I MUST BE FROM 1 TO 4 INCLUSIVE.  THE FIRST CODE     FT045320
*       GENERATED IS 'SLT I'.  THE OTHER DETAILS ARE THE SAME AS GIVEN  FT045330
*       IN THE SENSE SWITCH DESCRIPTION ABOVE.                          FT045340
        SPACE   3                                                       FT045350
1IFSLT  AXC     15,2            RESET POINTER                           FT045360
        STL     .P              NON-ZERO FLAG                           FT045370
        REM                                                             FT045380
        AXT     1TRANS,4        SET ENTRY FOR LABEL ROUTINE             FT045390
        SXA     TRAFLG,4                                                FT045400
        REM                                                             FT045410
        CAL     COLUMN-1,2      CHECK FOR CLOSING PARENTHESIS           FT045420
        ERA     =H)                                                     FT045430
        ARS     30                                                      FT045440
        TNZ     ERROR7          NO PARENTHESIS                          FT045450
        REM                                                             FT045460
        CAL     COLUMN-2,2      GET DIGIT                               FT045470
        ARS     30                                                      FT045480
        TNZ     *+2             ZERO NOT ALLOWED                        FT045490
1IFSL2  ERROR   5,1IFSL1                                                FT045500
        REM                                                             FT045510
        LDQ     =6              FOR SWITCH                              FT045520
        ZET     .P              TEST FLAG                               FT045530
        LDQ     =4              FOR LIGHT                               FT045540
        TLQ     1IFSL2          IMPROPER DIGIT                          FT045550
        SLW     .P+1            SAVE DIGIT                              FT045560
        REM                                                             FT045570
1IFSL1  TSX     SCNBCD,4        GET N1                                  FT045580
        TPL     ERROR1          NOT A LABEL                             FT045590
        SLW     .P+2            SAVE BCD N1                             FT045600
        REM                                                             FT045610
        TSX     STOLBL,4        PUT N1 IN SYMTAB.  MINUS SIGN           FT045620
        LAS     CARD            COMPARE WITH CURRENT LABEL              FT045630
        TRA     *+2                                                     FT045640
        ERROR   6,*+1           IMPROPER TRANSFER                       FT045650
        REM                                                             FT045660
        SLW     1TRAN2          SAVE POINTER FOR N1                     FT045670
        SKIP    CMCHAR          TEST FOR COMMA                          FT045680
        ERROR   2,SKEND                                                 FT045690
        REM                                                             FT045700
        TSX     SCNBCD,4        GET N2                                  FT045710
        TPL     ERROR1          NOT A LABEL                             FT045720
        REM                                                             FT045730
        TSX     STOLBL,4        PUT N2 IN ST.  MINUS SIGN               FT045740
        LAS     CARD            COMPARE N2 WITH CURRENT LABEL           FT045750
        TRA     *+2                                                     FT045760
        ERROR   6,*+1           IMPROPER TRANSFER                       FT045770
        REM                                                             FT045780
        SLW     .P+5            SAVE N2 POINTER                         FT045790
        REM                                                             FT045800
        SKIP    63              END OF STATEMENT EXPECTED               FT045810
        ERROR   3,SKEND                                                 FT045820
        NOCODE                  TEST CODE FLAG                          FT045830
        REM                                                             FT045840
        CAL     SLT             FOR LIGHT.  CODE IS...  SLT I           FT045850
        NZT     .P              TEST FLAG                               FT045860
        CAL     SWT             FOR SWITCH.  CODE IS...  SWT I          FT045870
```

```
CODEO    .P+1,N                                                    FT045880
REM                                                                FT045890
CODECO   TRA,.P+5,R     CODE IS...  TRA N2                         FT045900
TRA      SKEND                                                     FT045910
```

```
        TTL     P A S S  1  P R O C E S S O R  -  S E N S E  L I G H T   FT045920
*       SENSE LIGHT I                                                   FT045930
        REM                                                             FT045940
*       ENTRY...  1SENSE                                                FT045950
*       THE NUMBER I MUST BE FROM 0 TO 4 INCLUSIVE.  THE CODE           FT045960
*       IS 'SLF I'.                                                     FT045970
        SPACE   3                                                       FT045980
1SENSE  AXC     11,2            SET POINTER AT END OF STATEMENT         FT045990
        SKIP    63              ONLY ONE DIGIT ALLOWED                  FT046000
        ERROR   5,SKEND                                                 FT046010
        REM                                                             FT046020
        CAL     COLUMN-2,2      CHECK SIZE OF DIGIT                     FT046030
        ARS     30                                                      FT046040
        LDQ     =4                                                      FT046050
        TLQ     ERROR5          GREATER THAN 4, OR NOT A DIGIT          FT046060
        REM                                                             FT046070
        CODEO   SLF,N           CODE IS...  SLN I                       FT046080
        TRA     SKEND                                                   FT046090
```

```
        TTL     P A S S  1  P R O C E S S O R  -  S T O P                FT046100
                                                                        FT046110
*       STOP I                                                          FT046120
        REM                                                             FT046130
*       ENTRY...  1STOPX                                                FT046140
*       THE NUMBER I MUST BE ABSENT (I.E. ZERO) OR BE AN OCTAL          FT046150
*       NUMBER FROM 0 TO 77777.  THE NUMBER IS PUT IN THE CONSTANT      FT046160
*       TABLE AS AN OCTAL ADDRESS INTEGER.  THE CODE 'CAL CONSTANT'     FT046170
*       AND 'LDQ CONSTANT' IS GENERATED, PLACING THE OCTAL NUMBER IN    FT046180
*       THE ADDRESS FIELDS OF BOTH THE AC AND THE MQ.  THE CODE         FT046190
*       'HTR *' IS PUT OUT.                                             FT046200
        SPACE   3                                                       FT046210
1STOPX  AXC     4,2             RESET POINTER FOR 'STOP'                FT046220
        STL     .P              NON-ZERO FLAG                           FT046230
        REM                                                             FT046240
        AXT     1TRANS+1,4      SET EXIT FOR SUBSEQUENT LABEL ROUTINE   FT046250
        SXA     TRAFLG,4                                                FT046260
        TRA     *+3
```

```
        TTL     P A S S  1  P R O C E S S O R  -  P A U S E                FT046270
*       PAUSE I                                                           FT046280
        REM                                                               FT046290
*       ENTRY...  1PAUSE                                                  FT046300
*       THE TREATMENT IS THE SAME AS FOR THE STOP STATEMENT, EXCEPT       FT046310
*       THAT THE FINAL CODE IS 'HTR *+1'.  THE HPR OPERATION CANNOT BE    FT046320
*       USED BECAUSE OF POSSIBLE COMPLICATIONS WITH THE                   FT046330
*       DATA CHANNEL TRAP.                                                FT046340
        SPACE   3                                                         FT046350
1PAUSE  AXC     5,2             RESET POINTER FOR 'PAUSE'                 FT046360
        STZ     .P              ZERO FLAG                                 FT046370
        REM                                                               FT046380
        SKIP    63              TEST IF OCTAL NUMBER PRESENT              FT046390
        TRA     1PAUS3          NUMBER PRESENT                            FT046400
        CODEC   LGR74,N         NUMBER NOT PRESENT.  CODE IS...  LGR 74   FT046410
        TRA     1PAUS4                                                    FT046420
        REM                                                               FT046430
1PAUS3  ZAC                                                               FT046440
        LDQ     COLUMN,2        GET NUMBER. TEST FOR OCTAL                FT046450
        AXT     6,2             SET FOR SIX BITES                         FT046460
        LGL     3                                                         FT046470
        TRA     *+3                                                       FT046480
        REM                                                               FT046490
1PAUS2  ARS     3               DROP 'BCD' PART OF NUMBER                 FT046500
        LGL     6               SHIFT IN NEXT BCD DIGIT                   FT046510
        PAI                                                               FT046520
        RFT     7               TEST HIGH-ORDER THREE BITS                FT046530
        TRA     *+3             ALL BITS NOT OFF                          FT046540
        TIX     1PAUS2,2,1      RETURN                                    FT046550
        ERROR   13,SKEND        NUMBER TOO LONG                           FT046560
        REM                                                               FT046570
        LGL     3                                                         FT046580
        PAI                                                               FT046590
        RNT     77              TEST FOR END OF STATEMENT                 FT046600
        ERROR   4,SKEND         NOT AN OCTAL NUMBER                       FT046610
        ARS     6               JUSTIFY OCTAL NUMBER                      FT046620
        NOCODE                                                            FT046630
        REM                                                               FT046640
1PAUS1  TSX     ASCON1,4        PUT OCTAL NUMBER IN CONSTANT TABLE        FT046650
        SLW     .P+1                                                      FT046660
        CODEO   CAL,C           CODE IS...  CAL OCTAL                     FT046670
        CODECO  LDQ,.P+1,C                  LDQ OCTAL                     FT046680
        REM                                                               FT046690
1PAUS4  CAL     PCOUNT          GET PCOUNT                                FT046700
        NZT     .P              IS IT 'STOP'.                             FT046710
        ADD     =1              NO.  CODE IS...  HTR *+1                  FT046720
        TSX     CITBLD,4,P      YES. CODE IS...  HTR *                    FT046730
        TRA     SKEND                                                     FT046740
```

```
        TTL     P A S S  1  P R O C E S S O R  -  F O R M A T                FT046750
*       FORMAT (...)                                                        FT046760
        REM                                                                 FT046770
*               TEMPORARY FORMAT STATEMENT PROCESSOR.  DOES NOT CHECK       FT046780
*               FORMAT STRING FOR VALIDITY                                  FT046790
        SPACE   3                                                           FT046800
PFMT    CAL     NXTLOC          SET POOL POINTER INTO ADDRESS OF EQUICALENTFT046810
        STA     EQUIV,4                                                     FT046820
        AXT     -1,1                                                        FT046830
        PAX     ,4                                                          FT046840
        TXI     *+1,4,1                                                     FT046850
*                                                                           FT046860
*       DROP ANY LEADING WORDS OF BLANKS AND WORD FORMAT                    FT046870
*                                                                           FT046880
        AXT     6,2                                                         FT046890
        LDQ     CARD,1                                                      FT046900
        ZAC                                                                 FT046910
        LGL     6                                                           FT046920
        ERA     =H00000(                                                    FT046930
        TZE     *+3                                                         FT046940
        TIX     *-4,2,1                                                     FT046950
        TXI     *-7,1,-1                                                    FT046960
        ORA     =H      (                                                   FT046970
        TNX     *+3,2,1                                                     FT046980
        LGL     6                                                           FT046990
        TRA     *-2                                                         FT047000
PFMT1   SLW     POOL,4          FIRST NON-BLANK WORD FOUND.  STASH WORD IN  FT047010
        REM                     POOL                                        FT047020
        TXI     *+1,1,-1                                                    FT047030
        CAL     CARD,1                                                      FT047040
        LAS     =0767676767676                                              FT047050
        TXI     PFMT1,4,1                                                   FT047060
        TRA     *+2                                                         FT047070
        TXI     PFMT1,4,1                                                   FT047080
        PXA     ,4              END OF FORMAT STRING.  SET NO. WORD IN STRIFT047090
        REM                     INTO POOL                                   FT047100
        SUB     NXTLOC                                                      FT047110
        LXA     NXTLOC,1                                                    FT047120
        STA     POOL,1                                                      FT047130
        TXI     *+1,4,1                                                     FT047140
        SXA     NXTLOC,4        RESET 0 POOL POINTER TO NEXT AVAIL. LOC.    FT047150
        TRA     SKEND           IN POOL                                     FT047160
```

```
*       BACKSPACE I                                                     FT047180
        REM                                                             FT047190
*       ENTRY...  1BSRXX                                                FT047200
*       SEE THE WRITE OUTPUT TAPE STATEMENT DESCRIPTION FOR THE         FT047210
*       TREATMENT OF THE LOGICAL TAPE NUMBER I AND THE TRANSFER VECTOR  FT047220
*       ENTRY.  THE INITIALIZATION ROUTINE NAME IS (BST).               FT047230
*       THE APPROPRIATE 'CAL ---' CODE IS GENERATED, FOLLOWED BY        FT047240
*       A 'TSX (BST),4' INSTRUCTION.                                    FT047250
        SPACE   3                                                       FT047260
1BSRXX  AXT     (BST),4         GET SUBROUTINE ENTRY                    FT047270
        AXC     9,2             RESET POINTER                           FT047280
        REM                                                             FT047290
1BSRX1  SXA     1RIT99,4        SAVE ENTRY                              FT047300
        STL     .P+3            FLAG3 NON-ZERO FOR NON-DATA-TRANSMIT    FT047310
        TRA     1RITX2          PROCESSING CONTINUED IN RIT SECTION     FT047320
```

```
*       REWIND I                                                        FT047340
        REM                                                             FT047350
*       ENTRY...  1REWXX                                                FT047360
*       THE TREATMENT IS THE SAME AS FOR BACKSPACE, EXCEPT THAT THE     FT047370
*       EXTERNAL ROUTINE CALLED BY THE TSX OPERATION IS (RWT).          FT047380
        SPACE   3                                                       FT047390
1REWXX  AXT     (RWT),4         ENTRY FOR REWIND                        FT047400
        AXC     6,2                                                     FT047410
        TRA     1BSRX1                                                  FT047420
```

```
        TTL     P A S S  1  P R O C E S S O R  -  E N D  F I L E     FT047430
*       END FILE I                                                   FT047440
        REM                                                         FT047450
*       ENTRY...  1EOFXX                                            FT047460
*       THE TREATMENT IS THE SAME AS FOR BACKSPACE, EXCEPT THAT THE FT047470
*       EXTERNAL ROUTINE CALLED BY THE TSX OPERATION IS (EFT).      FT047480
        SPACE   3                                                   FT047490
1EOFXX  AXT     (EFT),4         ENTRY FOR END OF FILE               FT047500
        AXC     7,2                                                 FT047510
        TRA     1BSRX1                                              FT047520
```

```
        TTL     P A S S  1  P R O C E S S O R  -  R E A D                FT047530
                                                                        FT047540
*       READ FMT, LIST
                                                                        FT047550
        REM                                                             FT047560
*       ENTRY...  1READX
*       THE TREATMENT OF THE TRANSFER VECTOR ENTRY, THE FORMAT NAME,    FT047570
*       AND THE INPUT LIST IS THE SAME AS FOR THE READ INPUT TAPE       FT047580
*       STATEMENT.  THE INITIALIZATION CODE IS  'TSX (CSH),4'.          FT047590
*       THE TERMINATION CODE IS 'TSX (RTN),4'.                          FT047600
                                                                        FT047610
        SPACE   3                                                       FT047620
1READX  AXT     (CSH),4         INITIALIZATION ENTRY FOR READ           FT047630
        AXT     (RTN),1         TERMINATION ENTRY FOR INPUT             FT047640
        STL     .P+6            FLAG6 NON-ZERO FOR INPUT                FT047650
        AXC     4,2                                                     FT047660
        REM                                                             FT047670
1READ1  SXA     1RIT11+1,1      SET TERMINATION ROUTINE NAME            FT047680
        STZ     .P+3            FLAG3 IS ZERO FOR DATA-TRANSMIT         FT047690
        STZ     .P+4            FLAG4 IS ZERO FOR BCD                   FT047700
        REM                                                             FT047710
        SXA     *+2,4           SET TRANSFER VECTOR NAME                FT047720
        TSX     OUTPTV,4                                                FT047730
        PZE     **              CODE IS...  TSX (XXX),4                 FT047740
        TRA     1RITX7          PROCESSING CONTINUED IN RIT SECTION
```

```
        TTL     P A S S  1  P R O C E S S O R  -  P R I N T                FT047750
*       PRINT FMT, LIST                                                   FT047760
        REM                                                               FT047770
*       ENTRY...  1PRINT                                                  FT047780
*       THE TREATMENT IS THE SAME AS FOR THE READ STATEMENT, EXCEPT       FT047790
*       THAT THE INITIALIZATION ROUTINE NAME IS (SPH), AND THE LIST IS    FT047800
*       AN OUTPUT LIST.                                                   FT047810
        SPACE   3                                                         FT047820
1PRINT  AXT     (SPH),4         INITIALIZATION ENTRY FOR PRINT            FT047830
        AXT     (FIL),1         TERMINATION ENTRY FOR OUTPUT              FT047840
        STZ     .P+6            FLAG6 IS ZERO FOR OUTPUT                  FT047850
        AXC     5,2                                                       FT047860
        TRA     1READ1                                                    FT047870
```

```
        TTL     P A S S  1  P R O C E S S O R  -  P U N C H                FT047880
*       PUNCH FMT, LIST                                                   FT047890
        REM                                                               FT047900
*       ENTRY...  1PUNCH                                                  FT047910
*       THE TREATMENT IS THE SAME AS FOR THE READ STATEMENT, EXCEPT       FT047920
*       THAT THE INITIALIZATION ROUTINE NAME IS (SCH), AND THE LIST IS    FT047930
*       AN OUTPUT LIST.                                                   FT047940
        SPACE   3                                                         FT047950
1PUNCH  AXT     (SCH),4         INITIALIZATION FOR PUNCH                  FT047960
        TRA     1PRINT+1                                                  FT047970
```

```
        TTL     P A S S  1  P R O C E S S O R  -  R E A D  T A P E      FT047980
*       READ TAPE I, LIST                                              FT047990
        REM                                                            FT048000
*       ENTRY...  1RDTPX                                               FT048010
*       THE TREATMENT IS THE SAME AS FOR THE READ INPUT TAPE STATEMENT, FT048020
*       EXCEPT THAT THERE IS NO FORMAT STATEMENT NAME, AND THE         FT048030
*       EXTERNAL SUBROUTINE NAMES ARE (TSB) FOR INITIALIZATION AND     FT048040
*       (RLR) FOR TERMINATION.                                         FT048050
        REM                                                            FT048060
        SPACE   3                                                      FT048070
1RDTPX  AXT     (TSB),4         INITIALIZATION FOR READ TAPE           FT048080
        AXT     (RLR),1         TERMINATION POINT FOR READ TAPE        FT048090
        STL     .P+6            FLAG6 NON-ZERO FOR INPUT               FT048100
        AXC     8,2                                                    FT048110
        REM                                                            FT048120
1RDTP1  SXA     1RIT99,4        SAVE TRANSFER VECTOR ENTRIES           FT048130
        SXA     1RIT11+1,1                                             FT048140
        STZ     .P+3            FLAG3 IS ZERO FOR DATA-TRANSMIT        FT048150
        STL     .P+4            FLAG4 IS NON-ZERO FOR BINARY           FT048160
        TRA     1RITX2          PROCESSING CONTINUED IN RIT SECTION    FT048170
```

```
        TTL     P A S S  1  P R O C E S S O R  -  W R I T E  T A P E     FT048180
*       WRITE TAPE I, LIST                                              FT048190
        REM                                                             FT048200
*       ENTRY...  1WRTPX                                                FT048210
*       THE TREATMENT IS THE SAME AS FOR THE WRITE OUTPUT TAPE STATEMENT,FT048220
*       EXCEPT THAT THERE IS NO FORMAT STATEMENT NAME, AND THE          FT048230
*       EXTERNAL SUBROUTINE NAMES ARE (STB) FOR INITIALIZATION AND      FT048240
*       (WLR) FOR TERMINATION.                                          FT048250
        SPACE   3                                                       FT048260
1WRTPX  AXT     (STB),4         INITIALIZATION FOR WRITE TAPE           FT048270
        AXT     (WLR),1         TERMINATION FOR WRITE TAPE              FT048280
        STZ     .P+6            FLAG6 IS ZERO FOR OUTPUT                FT048290
        AXC     9,2                                                     FT048300
        TRA     1RDTP1                                                  FT048310
```

```
*       WRITE OUTPUT TAPE I, FMT, LIST                                  FT048330
        REM                                                             FT048340
*       ENTRY...  1WOTXX                                                FT048350
*       IF I IS A NUMBER, IT IS CONVERTED TO A BINARY DECREMENT INTEGER  FT048360
*       AND IS PUT IN THE CONSTANT TABLE.  THE CODE 'CAL INTEGER'       FT048370
*       IS GENERATED.                                                   FT048380
*       IF I IS A FIXED POINT VARIABLE NAME, IT IS PUT IN THE           FT048390
*       SYMBOL TABLE.  IF THE SYMBOL IS NOT A PARAMETER (SUBPROGRAM     FT048400
*       ARGUMENT), THE CODE 'CAL VARIABLE' IS GENERATED.  IF I IS A     FT048410
*       PARAMETER (I.E BARGT IS ON), THE PROLOG ROUTINE IS ENTERED.     FT048420
*       ON RETURN, THE CODE 'CAL **' IS PUT OUT.                        FT048430
        REM                                                             FT048440
*       THE NAMES OF THE EXTERNAL SUBROUTINES FOR INITIALIZATION, (STH), FT048450
*       AND TERMINATION, (FIL), ARE PUT IN THE TRANSFER VECTOR LIST.    FT048460
*       THE INITIALIZATION CODE 'TSX (STH),4' IS GENERATED.             FT048470
        REM                                                             FT048480
*       IF FMT IS A STATEMENT LABEL, IT IS PUT IN THE SYMBOL TABLE.     FT048490
*       IT MUST NOT ALREADY BE IN THE TABLE UNLESS IT IS FLAGGED AS     FT048500
*       A STRING (I.E. THE MSTRG BITS ARE ON).  FOR A NEW SYMBOL        FT048510
*       TABLE ENTRY, THE MSTRG BITS ARE TURNED ON.  THE CODE 'PZE STRING'FT048520
*       IS TURNED OUT.                                                  FT048530
*       IF FMT IS A VARIABLE NAME, IT IS PUT IN THE SYMBOL TABLE.       FT048540
*       THE NAME MUST ALREADY BE IN THE TABLE FLAGGED AS AN ARRAY       FT048550
*       (BARRY IS ON).  IF FMT IS NOT A SUBROUTINE PARAMETER, THE       FT048560
*       CODE 'PZE VARIABLE' IS PUT OUT.  IF FMT IS A PARAMETER, THE     FT048570
*       PROLOG ROUTINE IS ENTERED.  ON RETURN, THE CODE 'PZE **'        FT048580
*       IS PUT OUT.                                                     FT048590
        REM                                                             FT048600
*       THE PROCESSOR NEXT TRANSFERS TO THE OUTPUT LIST PROCESSOR       FT048610
*       TO GENERATE THE OUTPUT CODE.  ON RETURN, THE TERMINATION        FT048620
*       CODE 'TSX (FIL),4' IS PUT OUT.                                  FT048630
        SPACE   3                                                       FT048640
1WOTXX  AXT     (STH),4         INITIALIZATION ENTRY FOR BCD WRITE      FT048650
        AXT     (FIL),1         TERMINATION FOR BCD WRITE               FT048660
        STZ     .P+6            FLAG6 IS ZERO FOR OUTPUT                FT048670
        AXC     15,2                                                    FT048680
        TRA     1RITX1                                                  FT048690
```

```
*       READ INPUT TAPE I, FMT, LIST                                    FT048710
        REM                                                             FT048720
*       ENTRY...  1RITXX                                                FT048730
*       THE TREATMENT IS THE SAME AS FOR THE WRITE OUTPUT TAPE          FT048740
*       STATEMENT, EXCEPT THAT THE LIST IS AN INPUT LIST AND THE        FT048750
*       EXTERNAL TERMINATION ROUTINE NAME IS (RTN).                     FT048760
        SPACE   3                                                       FT048770
1RITXX  AXT     (TSH),4         INITIALIZATION ENTRY FOR BCD READ TAPE  FT048780
        AXT     (RTN),1         TERMINATION POINT FOR BCD TAPE READ     FT048790
        STL     .P+6            FLAG6 NON-ZERO FOR INPUT                FT048800
        AXC     13,2                                                    FT048810
        REM                                                             FT048820
1RITX1  SXA     1RIT99,4        SAVE ENTRIES                            FT048830
        SXA     1RIT11+1,1                                              FT048840
        STZ     .P+3            FLAG3 IS ZERO FOR DATA-TRANSMIT         FT048850
        STZ     .P+4            FLAG4 IS ZERO FOR BCD                   FT048860
        REM                                                             FT048870
*       ENTER HERE FROM BINARY AND NON-DATA-TRANSMITTING STATEMENTS.    FT048880
1RITX2  TSX     SCNBCD,4        GET TAPE NUMBER                         FT048890
        TPL     1RITX3          VARIABLE TAPE NAME                      FT048900
        REM                                                             FT048910
        TSX     BCDFIX,4        CONVERT TAPE NUMBER TO FIXED POINT BINARY FT048920
        TSX     ASCON1,4        PUT IN CONSTANT TABLE                   FT048930
        CODEO   CAL,C           CODE IS...  CAL TAPE.NUMBER             FT048940
        REM                                                             FT048950
1RIT12  TSX     OUTPTV,4        PUT NAME IN TRANSFER VECTOR             FT048960
1RIT99  PZE     **              CODE IS...  TSX (XXX),4                 FT048970
        REM                                                             FT048980
        NZT     .P+3            TEST FLAG3                              FT048990
        TRA     1RITX4          CONTINUE                                FT049000
        SKIP    63              TEST EOS FOR END FILE, BACKSP, AND REWIND FT049010
        ERROR   3,SKEND                                                 FT049020
        TRA     SKEND                                                   FT049030
        REM                                                             FT049040
1RITX4  ZET     .P+4            TEST FLAG4                              FT049050
        TRA     1RITX5          BINARY...  SKIP FORMAT FETCH            FT049060
        SKIP    CMCHAR          BCD.  TEST FOR COMMA                    FT049070
        ERROR   2,SKEND                                                 FT049080
        REM                                                             FT049090
*       ENTER HERE FROM READ, PRINT, AND PUNCH STATEMENTS.              FT049100
1RITX7  TSX     SCNBCD,4        GET FORMAT STATEMENT LABEL              FT049110
        TPL     1RIT10          VARIABLE FORMAT                         FT049120
        REM                                                             FT049130
        XCL                     ARGUMENT TO MQ                          FT049140
        TSX     LOCATE,4        FIND POSITION IN SYMTAB                 FT049150
        SLW     .P+7            SAVE POINTER                            FT049160
        TPL     1RITX6          FOR OLD SYMTAB ENTRY                    FT049170
        REM                                                             FT049180
        PAX     ,4                                                      FT049190
        SIL     MSTRG           TURN ON STRING MODE BITS                FT049200
        STI     EQUIV,4         RESTORE EQUIVALENT                      FT049210
        TRA     *+4                                                     FT049220
        REM                                                             FT049230
1RITX6  IIL     MSTRG           TEST FOR STRING MODE BITS ON            FT049240
        LFT     700000                                                  FT049250
        ERROR   24,1RITX5       STRING MODE BITS NOT ON.                FT049260
        REM                                                             FT049270
        CODEC   .P+7,F          CODE IS...  PZE FORMAT                  FT049280
        REM                                                             FT049290
```

```
        REM                                                                 FT049300
1RITX5  SKIP    CMCHAR          TEST FOR COMMA                              FT049310
        TRA     1RIT13          NO COMMA                                    FT049320
        REM                                                                 FT049330
        AXT     POOL-ISTKFG,4   RESET ISTACK ROUTING ADDRESS                FT049340
        SXA     ISTKFG,4        FOR USE BY THE LIST PROCESSOR               FT049350
        STZ     REGIST          RESET REGISTER INDICATION                   FT049360
        TSX     IOLIST,4        GO TO LIST PROCESSOR                        FT049370
        TRA     1RIT17          ON RETURN, SKIP END OF STATEMENT TEST       FT049380
        REM                                                                 FT049390
1RIT13  SKIP    63              MUST BE END OF STATEMENT                    FT049400
        ERROR   3,SKEND                                                     FT049410
        REM                                                                 FT049420
1RIT17  NZT     *+3             IS THERE A TERMINATION ENTRY.               FT049430
        TRA     SKEND           NO.                                         FT049440
        REM                                                                 FT049450
1RIT11  TSX     OUTPTV,4        PUT NAME IN TRANSFER VECTOR                 FT049460
        PZE     **              CODE IS...  TSX (XXX),4                     FT049470
        TRA     SKEND                                                       FT049480
        REM                                                                 FT049490
*                                                                           FT049500
*       PROCESSING FOR VARIABLE TAPE NUMBER                                 FT049510
1RITX3  LAS     =H100000        TEST FOR FIXED POINT                        FT049520
        TRA     *+1                                                         FT049530
        LAS     =H000000                                                    FT049540
        TRA     *+1                                                         FT049550
        ERROR   12,1RITX4       NOT FIXED POINT                             FT049560
        REM                                                                 FT049570
        SLW     FORM69                                                      FT049580
        TSX     SYMSTO,4        PUT IN SYMTAB                               FT049590
        REM                                                                 FT049600
        LFT     BEXTF+BINTF                                                 FT049610
        ERROR   69,*+1          MAY NOT BE SUBPORGRAM NAME                  FT049620
        REM                                                                 FT049630
        LNT     BARGT           IS IT SUBROUTINE ARGUMENT.                  FT049640
        TRA     1RITX9          NO.  REGULAR VARIABLE                       FT049650
        REM                                                                 FT049660
        TSX     PROLOG,4        YES.  FIX UP PROLOGUE                       FT049670
        CODEC   CAL,N           CODE IS...  CAL **                          FT049680
        TRA     1RIT12                                                      FT049690
        REM                                                                 FT049700
1RITX9  CODEO   CAL,V           CODE IS...  CAL TAPE.NUMBER                 FT049710
        TRA     1RIT12                                                      FT049720
        REM                                                                 FT049730
*                                                                           FT049740
*       PROCESSING FOR VARIABLE FORMAT STATEMENT SPECIFICATION              FT049750
1RIT10  XCL                                                                 FT049760
        TSX     LOCATE,4        FIND SPOT FOR VARIABLE FORMAT IN SYMTAB     FT049770
        LNT     BARRY           IS IT DIMENSIONED.                          FT049780
        ERROR   14,1RIT16       NO.  PUT OUT MESSAGE AND RETURN             FT049790
        REM                                                                 FT049800
        PAX     ,4                                                          FT049810
        SIL     BVARB           MARK AS VARIABLE IN EXECUTABLE STATEMENT    FT049820
        STI     EQUIV,4         RESTORE EQUIV WORD                          FT049830
        REM                                                                 FT049840
1RIT16  LNT     BARGT           IS IT SUBROUTINE ARGUMENT.                  FT049850
        TRA     1RIT15                                                      FT049860
        REM                                                                 FT049870
        TSX     PROLOG,4        YES.  FIX UP PROLOGUE                       FT049880
        CODEC   PZE,N           CODE IS...  PZE **                          FT049890
```

```
        TRA     1RITX5                                                  FT049900
        REM                                                             FT049910
1RIT15  TSX     CITBLD,4,V      CODE IS...  PZE FORMAT                  FT049920
        TRA     1RITX5                                                  FT049930
```

```
        TTL     P A S S  1  P R O C E S S O R  -  W R I T E  D R U M   FT049940
*****   NOTE  *****  READ AND WRITE DRUM ARE DEACTIVATED IN THIS       FT049950
*       VERSION OF FASTRAN                                            FT049960
        REM                                                           FT049970
REORG1  EQU     *               POINT OR REORIGIN TO DEACTIVATE DRUM  FT049980
        SPACE   3                                                     FT049990
*       WRITE DRUM I, J, LIST                                         FT050000
        REM                                                           FT050010
*       ENTRY...  1WDRMX                                              FT050020
*       THE DRUM NUMBER I AND THE DRUM LOCATION J MUST EACH BE A      FT050030
*       FIXED POINT CONSTANT OR A FIXED POINT VARIABLE.               FT050040
*       THE CODE 'CAL I', 'TSX (SDR),4', 'CAL J', AND 'LDA' IS        FT050050
*       TURNED OUT.  THE TREATMENT OF I AND J AND THE TRANSFER        FT050060
*       VECTOR ENTRY IS SIMILAR TO THAT DISCUSSED IN THE WRITE        FT050070
*       OUTPUT TAPE STATEMENT DESCRIPTION.                            FT050080
*       THE DRUM LIST PROCESSOR IS ENTERED TO GENERATE THE OUTPUT     FT050090
*       CODE SEQUENCES.                                               FT050100
        SPACE   3                                                     FT050110
1WDRMX  AXC     9,2                                                   FT050120
        AXT     (SDR),4         GET ENTRY TO WRITE DRUM ROUTINE       FT050130
        TRA     1RDRM1                                                FT050140
```

```
         TTL    P A S S  1  P R O C E S S O R  -  R E A D  D R U M          FT050150
*        READ DRUM I, J, LIST                                               FT050160
         REM                                                                FT050170
*        ENTRY...  1RDRMX                                                   FT050180
*        THE PROCEDURE IS SIMILAR TO WRITE DRUM, EXCEPT THAT THE            FT050190
*        EXTERNAL SUBROUTINE NAME IS (DRS), AND THE LIST IS TREATED         FT050200
*        AS AN INPUT LIST.                                                  FT050210
         SPACE  3                                                           FT050220
1RDRMX   AXC    8,2                                                        FT050230
         AXT    (DRS),4                                                    FT050240
         REM                                                                FT050250
1RDRM1   SXA    1RDRM4,4       SAVE TRANSFER VECTOR ENTRY                  FT050260
         TSX    SCNBCD,4       GET DRUM NUMBER I                           FT050270
         TPL    1RDRM2         VARIABLE                                    FT050280
         REM                                                                FT050290
         TSX    BCDFIX,4       CONSTANT                                    FT050300
         TSX    ASCON1,4       PUT IN CONSTANT TABLE                       FT050310
         CODEO  CAL,C          CODE IS...  CAL I                           FT050320
         REM                                                                FT050330
1RDRM5   SKIP   CMCHAR         CHECK FOR COMMA                             FT050340
         ERROR  2,SKEND                                                     FT050350
         REM                                                                FT050360
         TSX    OUTPTV,4       PUT ENTRY IN TRANSFER VECTOR                FT050370
1RDRM4   PZE    **             CODE IS...  TSX (XXX),4                     FT050380
         REM                                                                FT050390
         TSX    SCNBCD,4       GET DRUM LOCATION J                         FT050400
         TPL    1RDRM2         VARIABLE                                    FT050410
         REM                                                                FT050420
         TSX    BCDFIX,4       CONSTANT                                    FT050430
         TSX    ASCON1,4       PUT IN CONSTANT TABLE                       FT050440
         CODEO  CAL,C          CODE IS...  CAL J                           FT050450
         CODEC  LDA,N          CODE IS...  LDA                             FT050460
         REM                                                                FT050470
         SKIP1  CMCHAR         CHECK FOR POSSIBLE COMMA                    FT050480
         TRA    IODRM          TO DRUM LIST PROCESSOR                      FT050490
         REM                                                                FT050500
1RDRM6   SKIP   63             END OF STATEMENT REQUIRED                   FT050510
         ERROR  3,SKEND                                                     FT050520
         TRA    SKEND                                                       FT050530
         REM                                                                FT050540
         REM                                                                FT050550
*        PROCESSING OF VARIABLE DRUM NUMBER AND VARIABLE DRUM ADDRESS       FT050560
*                                                                           FT050570
1RDRM2   LAS    =HI00000       TEST FOR FIXED POINT                        FT050580
         TRA    *+1                                                         FT050590
         LAS    =H000000                                                    FT050600
         TRA    *+1                                                         FT050610
         ERROR  12,SKEND       NOT FIXED POINT                             FT050620
         REM                                                                FT050630
         TSX    SYMSTO,4       OKAY.  PUT IN SYMTAB                        FT050640
         LNT    BARGT          TEST FOR SUBROUTINE ARGUMENT                FT050650
         TRA    1RDRM3         NOT AN ARGUMENT                             FT050660
         REM                                                                FT050670
         TSX    PROLOG,4       SUBROUTINE ARGUMENT.  FIX UP PROLOG.        FT050680
         CODEC  CAL,N          CODE IS...  CAL **                          FT050690
         TRA    1RDRM5                                                      FT050700
         REM                                                                FT050710
1RDRM3   CODEO  CAL,V          CODE IS...  CAL X                           FT050720
         TRA    1RDRM5                                                      FT050730
         SPACE  3                                                           FT050740
```

```
IODRM   TRA     SKEND           ***** TEMPORARY DRUM LIST PROCESSOR *****  FT050750
        SPACE   3                                                          FT050760
        ORG     REORG1          TO DELETE DRUM CODE FROM THIS VERSION      FT050770
```

```
      TTL    P A S S 1  -  I / O  L I S T  P R O C E S S O R                FT050780
*     COMMENTARY ON I/O LIST PROCESSOR-                                     FT050790
      SPACE  2                                                              FT050800
*     ON ENTRY TO I O L I S T THE COLUMN POINTER IS POSITIONED AT THAT       FT050810
*     CHARACTER IN THE I/O STATEMENT FOLLOWING THE COMMA AFTER THE           FT050820
*     FORMAT STATEMENT NUMBER(BCD) OR AFTER THE TAPE NUMBER (BINARY).       FT050830
      REM                                                                   FT050840
*     THE I/O LIST PROCESSOR IS BROKEN INTO TWO PARTS.  THE FIRST PART       FT050850
*     (BEGINNING AT I O L S T 1) BUILDS A S T A C K AND P S T A C K          FT050860
*     AND THE SECOND PART (I O L S T L) PROCESSES THESE TWO LISTS AND        FT050870
*     GENERATES THE OBJECT CODE.                                            FT050880
      SPACE  2                                                              FT050890
*     I/O LIST PROCESSOR - PART 1-                                          FT050900
*        I/O LIST ITEMS, LEFT AND RIGHT PARENTHESES CAUSE A S T A C K       FT050910
*        ENTRIES TO BE GENERATED.  ASTACK ENTRIES FOR LIST ITEMS ARE        FT050920
*        HANDLED LIKE ITEMS IN AN ARITHMETIC STATEMENT.  A LEFT PAREN.      FT050930
*        IN AN IOLIST WILL CAUSE THE FOLLOWING ASTACK ENTRY TO BE           FT050940
*        GENERATED,                                                         FT050950
*                          MTH   (PAREN. LEVEL),,0                          FT050960
*        IF INDEXING IS INVOLVED IN AN I/O LIST THEN THE ASTACK ENTRY       FT050970
*        FOR THE LEFT PAREN. WHICH BEGINS THE RANGE OF THE INDEX IS,        FT050980
*                       MTH   (PAREN. LEVEL),,(PSTACK PTR. FOR INDEX)       FT050990
*        A RIGHT PARENTHESIS THAT DOES NOT FOLLOW INDEXING                  FT051000
*        SPECIFICATIONS IN AN I/O LIST DOES NOT CAUSE GENERATION OF         FT051010
*        AN ASTACK ENTRY.  RIGHT PARENS. FOLLOWING INDEXING SPECS.          FT051020
*        CAUSE THE FOLLOWING ASTACK ENTRY TO BE GENERATED,                  FT051030
*                          MTW   (PSTACK INDEX PTR.),,0                     FT051040
*        ENTRIES IN P S T A C K ARE MADE ONLY WHEN INDEXING IS INVOLVED     FT051050
*        (I.E. I = I1,I2,I3).  WHENEVER INDEXING IS ENCOUNTERED FIVE        FT051060
*        P S T A C K WORDS ARE GENERATED AS FOLLOWS,                        FT051070
*                 MTH   (ASTACK PTR. FOR THIS PAREN. LEVEL),,0              FT051080
*                 INDEXING VARIABLE (I)                                     FT051090
*                 INITIAL INDEX VALUE (I1)                                  FT051100
*                 FINAL INDEX VALUE (I2)                                    FT051110
*                 INDEX INCREMENT (I3)                                      FT051120
*        DURING EXECUTION OF PART 1, LOCATIONS (CURPTR-1), (CURPTR-2),      FT051130
*        AND (CURPTR-3) CONTAIN THE A S T A C K POINTERS FOR PAREN.         FT051140
*        LEVELS 1, 2, AND 3.  PARENTHESIS NESTING CANNOT BE GREATER         FT051150
*        THAN 3.  LOCATION I O L V L CONTAINS THE CURRENT PAREN. LEVEL.     FT051160
      SPACE  2                                                              FT051170
*     I/O LIST PROCESSOR - PART 2-                                          FT051180
*        THE SECOND PART OF THE I/O LIST PROCESSOR SIMPLY USES THE          FT051190
*        A S T A C K AND P S T A C K ENTRIES GENERATED BY PART 1 AND        FT051200
*        OUTPUT THE OBJECT CODE ON C I T USING THE CODE AND CODEN           FT051210
*        ROUTINES.  ITEMS IN INPUT LISTS ARE FLAGGED AS BEING DEFINED       FT051220
*        BY TURNING ON B L H S X IN E Q U I V.  THE (SLO) OR (SLI)          FT051230
*        SEQUENCES ARE GENERATED FOR NON-SUBSCRITED ARRAY ITEMS.            FT051240
      SPACE  2                                                              FT051250
*     CONSIDER THE FOLLOWING EXAMPLE,                                       FT051260
      REM                                                                   FT051270
*                        .                                                  FT051280
*                        .                                                  FT051290
*                        .                                                  FT051300
*        DIMENSION A(100),B(10,100)                                         FT051310
*                        .                                                  FT051320
*                        .                                                  FT051330
*                        .                                                  FT051340
*        READ INPUT TAPE 5,100,C,(D),((B(I,J),I=1,10),A(J),J=J1,100,3)      FT051350
      SPACE  2                                                              FT051360
*           THE FOLLOWING A S T A C K ENTRIES WOULD BE GENERATED BY         FT051370
```

```
*          PART 1,
*                    ENTRY                    REMARKS                    FT051380
*                    -----                    -------                    FT051390
*                                                                        FT051400
      REM                                                                FT051410
*                    C                        NORMAL ASTACK ENTRY        FT051420
*                    MTH   1,,0               ENTRY FOR LEFT PAREN.      FT051430
*                    D                        NORMAL ASTACK ENTRY        FT051440
*                    MTH   1,,6               ADDR. = LEVEL 1            FT051450
*                                             DECR. = PSTACK PTR. FOR    FT051460
*                                             THIS LEVEL (SEE BELOW)     FT051470
*                    MTH   2,,1               LEVEL,,PSTACK POINTER      FT051480
*                    B(I,J)                   NORMAL ASTACK ENTRY        FT051490
*                    MTW   1,,0               ADDR. = PSTACK POINTER     FT051500
*                                             GENERATED BY RIGHT         FT051510
*                                             PAREN. FOLLOWING INDEX     FT051520
*                                             ON I.                      FT051530
*                    A(J)                     NORMAL ASTACK ENTRY        FT051540
*                    MTH   6,,0               ADDR. = PSTACK POINTER     FT051550
*                                             GENERATED BY RIGHT         FT051560
*                                             PAREN. FOLLOWING INDEX     FT051570
*                                             ON J.                      FT051580
      SPACE  2                                                           FT051590
*          THE FOLLOWING P S T A C K ENTRIES WOULD BE GENERATED BY        FT051600
*          THE TWO INDEXING PARAMETERS I AND J,                           FT051610
*                    ENTRY                    REMARKS                    FT051620
*                    -----                    -------                    FT051630
      REM                                                                FT051640
*                    MTH   2,,5               ADDR. = PAREN. LEVEL       FT051650
*                                             DECR. = ASTACK POINTER     FT051660
*                    I                        FIRST INDEX VAR.           FT051670
*                    1                        INITIAL VALUE              FT051680
*                    10                       FINAL VALUE                FT051690
*                    1                        INCREMENT                  FT051700
*                    MTH   1,,4               ADDR. = PAREN. LEVEL       FT051710
*                                             DECR. = ASTACK POINTER     FT051720
*                    J                        SECOND INDEX VAR.          FT051730
*                    J1                       INITIAL VALUE.  THIS       FT051740
*                                             AN ASTACK-TYPE ENTRY.      FT051750
*                    100                      FINAL VALUE                FT051760
*                    3                        INCREMENT                  FT051770
      SPACE  2                                                           FT051780
*        THE OBJECT CODE GENERATED BY THIS READ STATEMENT WOULD BE,       FT051790
      REM                                                                FT051800
*                                                                        FT051810
*                    .                                                   FT051820
*                    .                                                   FT051830
*                    .                                                   FT051840
*              CAL   =05000000          TAPE NUMBER                      FT051850
*              TSX   $(TSH),4           TAPE TO STORAGE HOLLERITH        FT051860
*              PZE   (LOC. FORMAT 100)                                   FT051870
*              STR                                                       FT051880
*              STQ   C                  INPUT C                          FT051890
*              STR                                                       FT051900
*              STQ   D                  INPUT D                          FT051910
*              LXD   J1,4               INITIALIZE INDEX ON J            FT051920
*            X SXD   J,4                                                 FT051930
*              AXT   1,4                INITIALIZE INDEX ON I            FT051940
*            Y SXD   I,4                                                 FT051950
*              LDQ   =10                INDEX CALCS. FOR B(I,J)          FT051960
*              MPY   J                                                   FT051970
*              LDQ   =1
```

```
*       STO     WORK1                                           FT051980
*       MPY     I                                               FT051990
*       ADD     WORK1                                           FT052000
*       ALS     17                                              FT052010
*       STO     WORK1                                           FT052020
*       STR                                                     FT052030
*       LXD     WORK1,4                                         FT052040
*       STQ     B+11,4          INPUT B(I,J)                    FT052050
*       LXD     I,4             TERMINATION TEST FOR INDEX I    FT052060
*       TXI     *+1,4,1                                         FT052070
*       TXL     Y,4,10                                          FT052080
*       STR                                                     FT052090
*       LXD     J,4             GET INDEX FOR A                 FT052100
*       STQ     A+1,4           INPUT A(J)                      FT052110
*       LXD     J,4             TERMINATION TEST FOR INDEX J    FT052120
*       TXI     *+1,4,3                                         FT052130
*       TXL     X,4,100                                         FT052140
*       TSX     $(RTN),4        RESTORE TO NORMAL               FT052150
*               .                                               FT052160
*               .                                               FT052170
*               .                                               FT052180
```

```
        EJECT                                                          FT052190
IOLIST  SXA     EXIOLS,4        SAVE XR4 FOR RETURN                    FT052200
        STZ     DBLMOD          NEEDED FOR 'SCAN' TO WORK OKAY         FT052210
        STZ     PSTACK          INITIALIZE FIRST PSTACK ENTRY          FT052220
        STZ     PPTR             PSTACK POINTER                        FT052230
        STZ     ASTACK           FIRST ASTACK ENTRY                    FT052240
        STZ     IOLVL            PAREN LEVEL INDICATOR                 FT052250
        STZ     NODOAR          INITIALIZE NO. DO PARAMS TO ZERO       FT052260
        AXT     0,1             XR1 IS RUNNING ASTACK POINTER          FT052270
        TSX     SKIP,4,EOS      IS NEXT END-OF-STATEMENT CHARACTER     FT052280
        TRA     *+2             NO, SKIP NEXT INSTR.                   FT052290
        TRA     EXIOLS          YES, EXIT --- NO LIST                  FT052300
IOLST1  TSX     SKIP,4,LPCHAR   IS NEXT CHARACTER (                    FT052310
        TRA     IOLST2          NO, GO TO IOLST2                       FT052320
        LXA     IOLVL,4         GET CURRENT PAREN LEVEL                FT052330
        TXH     IOLSTA,4,2      IF LEVEL THREE, ERROR                  FT052340
        TXI     *+1,4,1         INCREASE LEVEL BY 1                    FT052350
        SXA     IOLVL,4         SAVE IN IOLVL                          FT052360
        PXA     ,4              BUILD ASTACK ENTRY                     FT052370
        ORA     =0700000000000   MTH   LEVEL NO.,,0                    FT052380
        SLW     ASTACK,1        SET ASTACK ENTRY                       FT052390
        PXA     ,1              INDICATE LAST ASTACK POS. THIS LEVEL   FT052400
        STA     CURPTR,4        SAVE THIS IN CURPTR VECTOR             FT052410
        TXI     IOLST1,1,1      BUMP ASTACK PTR. AND LOOK FOR (        FT052420
IOLST2  TSX     SCAN,4          SCAN OFF NEXT LIST ELEMENT             FT052430
        ERROR   48,SKEND        E-O-S IMPOSSIBLE                       FT052440
        TRA     *+2             CONSTANT --- GO PUT IT OUT             FT052450
        ERROR   48,SKEND        MISPLACED OPERATOR                     FT052460
        SLW     ASTACK,1        NO, SAVE THIS IN ASTACK                FT052470
        TXI     *+1,1,1         BUMP ASTACK POINTER                    FT052480
        STZ     ASTACK,1        KEEP NEXT ASTACK ENTRY CLEAR           FT052490
        CAL     COLUMN,2        OBSERVE NEXT CHAR. IN STATEMENT        FT052500
        ANA     =0770000000000   X                                     FT052510
        LAS     =H=00000        IS IT =                                FT052520
        TRA     *+2             NO                                     FT052530
        TXI     IOLST6,2,-1     YES, GO TO IOLST6                      FT052540
        LAS     =H(00000        NO, IS IT (                            FT052550
        TRA     *+2             NO                                     FT052560
        TXI     IOLST4,2,-1     YES, GO TO IOLST4                      FT052570
IOLST3  LAS     =H,00000        NO, IS IT ,                            FT052580
        TRA     *+2             NO                                     FT052590
        TXI     IOLST1,2,-1     YES, GO TO IOLST1                      FT052600
        LAS     =0770000000000  IS IT END-OF-STATEMENT                 FT052610
        TRA     *+2             NO                                     FT052620
        TRA     IOLSTL          GO TO PART 2 OF IOLIST PROCESSOR       FT052630
        LAS     =H)00000        NO, IS IT )                            FT052640
        ERROR   47,SKEND        INVALID LIST                           FT052650
        TXI     *+2,2,-1        YES, GO TO IOLST9                      FT052660
        TRA     *-2             NO, AN ERROR                           FT052670
IOLST9  LXA     IOLVL,4         PROCESS RIGHT PARENTHESIS              FT052680
        TXL     IOLSTB,4,0      ERROR IF LEVEL ALREADY ZERO            FT052690
        TXI     *+1,4,-1        DECREASE LEVEL BY 1                    FT052700
        SXA     IOLVL,4         SAVE IN IOLVL                          FT052710
        CAL     COLUMN,2        GET NEXT CHARACTER OF STATEMENT        FT052720
        ANA     =0770000000000   X                                     FT052730
        TRA     IOLST3          GO LOOK FOR , E-OS- )                  FT052740
IOLST4  LDI     ASTACK+1,1      PROCESS LEFT PARENTHESIS               FT052750
        LNT     BARRY           IS THIS AN ARRAY                       FT052760
        ERROR   47,SKEND        SUBSCRIPTED VAR. NOT AN ARRAY          FT052770
        TXI     *+1,1,-1      JUSTIFY THE INDEX                        FT052780
```

```
        TSX     1INDEX,4                                                FT052790
        TXI     *+1,1,1         RESTORE THE INDEX                       FT052800
        TRA     IOLST4-3            GO LOOK FOR , E-O-S )               FT052810
IOLST6  LDI     ASTACK+1,1          PROCESS =                           FT052820
        STZ     ASTACK+1,1          PICK-UP INDEX VARIABLE              FT052830
        TXI     *+1,1,-1            CLEAR INDEX VAR. FROM ASTACK        FT052840
        SXA     APTR,1              SAVE ASTACK POINTER                 FT052850
        IIL     MINTG               IS THIS A FIXED POINT INTEGER       FT052860
        LFT     700000               X                                  FT052870
        ERROR   46,SKEND            INDEX VAR. NOT FIXED POINT          FT052880
        NZT     IOLVL               CHECK THAT LEVEL IS NON-ZERO        FT052890
        ERROR   46,SKEND            INDEXING AT LEVEL ZERO              FT052900
        LXA     IOLVL,4             GET ASTACK POINTER FOR THIS LEVEL   FT052910
        CAL     CURPTR,4             X                                  FT052920
        ORA     =0700000000000       X                                  FT052930
        LXA     PPTR,1              GENERATE PSTACK ENTRY               FT052940
        SLW     PSTACK,1             MTH   ASTACK PTR. THIS LEVEL,,0    FT052950
        PAX     ,4                  SET DECREMENT OF ASTACK ENTRY TO    FT052960
        TXI     *+1,1,1              PSTACK POINTER                     FT052970
        PXD     ,1                   X                                  FT052980
        STD     ASTACK,4            PUT THIS IS ASTACK ENTRY            FT052990
        STI     PSTACK,1             X                                  FT053000
IOLST7  TSX     DOARG,4             GO SCAN OFF INDEXING PARAMETERS     FT053010
        TXI     *+1,1,1             BUMP PSTACK POINTER                 FT053020
        SLW     PSTACK,1            STORE PARAMETERS IN PSTACK          FT053030
        CLA     NODOAR              BUMP NO. INDEX PARAMS BY 1          FT053040
        ADD     =1                   X                                  FT053050
        STO     NODOAR               X                                  FT053060
        TSX     SKIP,4,RPCHAR       IS NEXT CHARACTER )                 FT053070
        TRA     IOLST8              NO, MAKE SURE IT IS ,               FT053080
        CAL     =2                  DETERMINE NO. DO PARAMETERS         FT053090
        LAS     NODOAR              ARE THERE TWO PARAMETERS            FT053100
        ERROR   46,SKEND            NO, NOT ENOUGH PARAMETERS           FT053110
        TRA     IOLST5              YES, GO SET THIRD TO 1              FT053120
        CAL     =3                   X                                  FT053130
        LAS     NODOAR              ARE THERE THREE PARAMETERS          FT053140
        NOP                         YES                                 FT053150
        TRA     IOLST5+3            YES, GO TO IOLST5+3                 FT053160
        ERROR   46,SKEND            TOO MANY INDEXING PARAMETERS        FT053170
IOLST5  CAL     =1                  SET THIRD INDEXING PARAMETER TO 1   FT053180
        TXI     *+1,1,1             BUMP PSTACK POINTER                 FT053190
        SLW     PSTACK,1            SET THIS PARAMETER IN PSTACK        FT053200
        TXI     *+1,1,1                                                 FT053210
        SXA     PPTR,1              SAVE PSTACK PTR. FOR NEXT TIME AROUND FT053220
        TXI     *+1,1,-5                                                FT053230
        PXA     ,1                                                      FT053240
        LXA     APTR,1              RESTORE XR1 TO ASTACK POINTER       FT053250
        ORA     =0600000000000                                          FT053260
        SLW     ASTACK,1                                                FT053270
        TXI     *+1,1,1                                                 FT053280
        STZ     NODOAR              RESET NO. DO PARAMS TO ZERO         FT053290
        TRA     IOLST9              GO CONTINUE LIST PROCESSING         FT053300
IOLST8  TSX     SKIP,4,CMCHAR       IS NEXT CHARACTER ,                 FT053310
        ERROR   46,SKEND            COMMA EXPECTED IN INDEXING PORTION  FT053320
        TRA     IOLST7              YES, SKIP THE , AND GO FOR NEXT PARAM FT053330
DOARG   SXA     EXDOAR,4            SAVE XR4                            FT053340
        TSX     SCNBCD,4            SCAN OFF NEXT SYMBOL OR CONSTANT    FT053350
        TMI     DOARG1              TRANSFER IF IT WAS A CONSTANT       FT053360
        TSX     SYMSTB,4            ENTER IN SYMTAB AND VERIFY FIXED PT. FT053370
        RIL     7                   TURN OFF SUBR. ARG. BITS            FT053380
```

```
        RIR     700000                                          FT053390
        STI     DOARG2          GENERATE PSTACK ENTRY           FT053400
        STA     DOARG2           X                              FT053410
        CAL     DOARG2           X                              FT053420
EXDOAR  AXT     **,4            RESTORE XR4                     FT053430
        TRA     1,4             EXIT                            FT053440
DOARG1  TSX     BCDFIX,4        CONVERT TO ADDRESS INTEGER      FT053450
        ARS     18               X                              FT053460
        TRA     EXDOAR           X                              FT053470
IOLSTA  ERROR   48,SKEND        NESTING LEVEL .GRT. 3           FT053480
IOLSTB  ERROR   48,SKEND        TOO MANY RIGHT PARENTHESES      FT053490
IOLSTL  ZET     IOLVL           E-O-S ENCOUNTERED. IS PAREN LEVEL ZEROFT053500
        ERROR   48,SKEND        NO, AND ERROR                   FT053510
        SXD     *+5,1           YES, SAVE LENGTH OF ASTACK      FT053520
        AXT     0,1             INITIALIZE POINTER FOR PART 2   FT053530
        CAL     ASTACK,1        PICK UP NEXT ASTACK ENTRY       FT053540
        TNZ     IOLSTZ          PROCESS NON-ZERO ENTRIES ONLY   FT053550
        TXI     *+1,1,1         BUMP PART 2 ASTACK POINTER      FT053560
        TXL     *-3,1,**        GO FOR NEXT ENTRY IF NOT DONE   FT053570
EXIOLS  AXT     **,4            FINAL IOLIST EXIT               FT053580
        TRA     1,4             X                               FT053590
IOLSTZ  PAI                     PUT ASTACK ENTRY IN INDICATORS  FT053600
        LNT     700000          IS THIS A LEFT PAREN ENTRY      FT053610
        TRA     IOLSTQ          NO, GO SEE IF RIGHT PAREN       FT053620
        LFT     077777          YES, DOES IT HAVE INDEXING      FT053630
        TRA     *+2             YES                             FT053640
        TXI     EXIOLS-1,1,1    NOTHING TO DO WITH THIS ENTRY   FT053650
        PDX     ,2              SET XR2 TO PSTACK POINTER       FT053660
        STZ     IOLT11          I3                              FT053670
        STZ     IOLT12          I2                              FT053680
        STZ     IOLT13          POOL (I3)                       FT053690
        STZ     IOLT14          POOL (I2)                       FT053700
        LDI     PSTACK-3,2      PICK UP I3 FROM PSTACK          FT053710
        LFT     777777          IS I3 A VARIABLE                FT053720
        TRA     *+3             YES                             FT053730
        STI     IOLT11          NO, A CONSTANT, PUT IT IN IOLT11 FT053740
        TRA     IOLSTM          GO LOOK AT I2                   FT053750
        STI     SAVE            I3 A VARIABLE, SO GENERATE LXD  I3,4 FT053760
        TSX     NEXTLP,4        GET NEXT POOL LOCATION          FT053770
        SLW     IOLT13          SAVE THIS POOL POINTER IN IOLT13 FT053780
        CODE    LXD4,SAVE       GENERATE LXD  (I3),4            FT053790
        CAL     IOLT13          GENERATE SXD WITH SPECIAL 'R' FLAG FT053800
        ORA     SXD4            X                               FT053810
        TSX     CITBLD,4,R      GENERATE SXD  **,4              FT053820
IOLSTM  LDI     PSTACK-2,2      PICK UP I2 FROM ASTACK          FT053830
        LFT     777777          IS I2 A VARIABLE                FT053840
        TRA     *+3             YES                             FT053850
        STI     IOLT12          NO, A CONSTANT, PUT IT IN IOLT12 FT053860
        TRA     IOLSTN          GO LOOK AT I1                   FT053870
        STI     SAVE            I2 IS VARIABLE, SO GENERATE LXD  I2,4 FT053880
        TSX     NEXTLP,4        GET NEXT POOL LOCATION          FT053890
        SLW     IOLT14          SAVE THIS POOL POINTER IN IOLT14 FT053900
        CODE    LXD4,SAVE       GENERATE LXD  (I2),4            FT053910
        CAL     IOLT14          GENERATE SXD WITH SPECIAL 'R' FLAG FT053920
        ORA     SXD4            X                               FT053930
        TSX     CITBLD,4,R      GENERATE SXD  **,4              FT053940
IOLSTN  LDI     PSTACK-1,2      PICK UP PSTACK ENTRY FOR I1     FT053950
        LFT     777777          IS I1 A VARIABLE                FT053960
        TRA     IOLSTO          YES, GO TO IOLSTO               FT053970
        PIA                     GENERATE AXT  (I1),4            FT053980
```

```
        ORA     AXT4                X                                   FT053990
        TSX     CITBLD,4,N          X                                   FT054000
        TRA     IOLSTP              X                                   FT054010
IOLSTO  STI     SAVE                I1 IS A VARIABLE                    FT054020
        CODE    LXD4,SAVE           GENERATE LXD  (I1),4                FT054030
IOLSTP  CAL     IOLT13              SET UP PSTACK AS FOLLOWS (THIS INDEX) FT054040
        ALS     18                   MTH (ASTACK PTR.),,0               FT054050
        ORA     PCOUNT                   (POOL PTR. I2),,0              FT054060
        SLW     PSTACK-3,2           PZE (I2),,(I3)                     FT054070
        LDQ     IOLT14                                                  FT054080
        CAL     PSTACK,2                    (I)                         FT054090
        STQ     PSTACK,2                                                FT054100
        SLW     PSTACK-2,2           PZE (PCOUNT),,(POOL PTR. (I3))     FT054110
        SLW     SAVE                GENERATE SXD  (I),4                 FT054120
        PAX     ,4                  INDICATE INDEXING VARIABLE IN IOLIST FT054130
        LDI     EQUIV,4              AS DEFINED                         FT054140
        SIL     BLHSX                X                                  FT054150
        STI     EQUIV,4              X                                  FT054160
        TSX     INDXT,4             VERIFY I NOT DO INDEX               FT054170
        CODE    SXD4,SAVE           X                                   FT054180
        CAL     IOLT11              X                                   FT054190
        ALS     18                  X                                   FT054200
        ORA     IOLT12              X                                   FT054210
        SLW     PSTACK-1,2          X                                   FT054220
        TXI     EXIOLS-1,1,1        NOW GO GET NEXT ASTACK ENTRY        FT054230
IOLSTQ  LNT     600000              IS THIS A RIGHT PAREN ENTRY         FT054240
        TRA     IOLSTR              NO, GO PROCESS THIS LIST ITEM       FT054250
        PAX     ,2                  YES, GENERATE LXD  (I),4            FT054260
        CAL     PSTACK-3,2          X                                   FT054270
        SLW     SAVE          X                                         FT054280
        CODE    LXD4,SAVE           X                                   FT054290
        CAL     PSTACK-4,2          IF I3 IS A CONSTANT THEN GENERATE   FT054300
        ALS     3                                                       FT054310
        PDX     ,4                                                      FT054320
        TXL     *+3,4,0                                                 FT054330
        CAL     PCOUNT              X                                   FT054340
        SLW     EQUIV,4                                                 FT054350
        CAL     PSTACK-2,2          X                                   FT054360
        ANA     =077777000000       X                                   FT054370
        ORA     TXI14               X                                   FT054380
        ACL     PCOUNT              X                                   FT054390
        TSX     CITBLD,4,P          OUT GOES THE TXI                    FT054400
        CAL     PSTACK-1,2                                              FT054410
        TZE     *+4                                                     FT054420
        PAX     ,4                                                      FT054430
        CAL     PCOUNT                                                  FT054440
        SLW     EQUIV,4                                                 FT054450
        CAL     PSTACK-2,2          NOW PUT OUT THE TXL                 FT054460
        ANA     =077777                                                 FT054470
        ALS     18                  X                                   FT054480
        SLW     SAVE                X                                   FT054490
        CAL     PSTACK-4,2                                              FT054500
        ANA     =077777                                                 FT054510
        ORA     SAVE                X                                   FT054520
        ORA     TXL4                X                                   FT054530
        TSX     CITBLD,4,P                                              FT054540
        TXI     EXIOLS-1,1,1        GO CONTINUE ASTACK PROCESSING       FT054550
IOLSTR  SLW     SAVE                TO OUTPUT LIST ITEMS                FT054560
        PAX     ,4                  GET EQUIV POINTER                   FT054570
        NZT     IOFLAG              IS THIS INPUT OR OUTPUT             FT054580
```

```
        TRA     IOLSTY          OUTPUT                                  FT054590
        RFT     700000          INPUT ... SO FLAG LIST ITEM AS BEING    FT054600
        TRA     *+2              DEFINED                                FT054610
        TRA     *+3             X                                       FT054620
        CAL     POOL-1,4        X                                       FT054630
        PAX     ,4              X                                       FT054640
        LDI     EQUIV,4         X                                       FT054650
        SIL     BLHSX           X                                       FT054660
        STI     EQUIV,4         X                                       FT054670
        IIL     MINTG                                                   FT054680
        LFT     700000                                                  FT054690
        TRA     *+2             NO                                      FT054700
        TSX     INDXT,4         YES, GO SEE IF 'DO' INDEX               FT054710
IOLSTY  LDI     ASTACK,1        GET THIS ASTACK ENTRY AGAIN             FT054720
        STI     A                                                       FT054730
        SXA     A+1,1                                                   FT054740
        RFT     700000          IS THIS ITEM SUBSCRIPTED                FT054750
        TRA     IOLSTK          YES                                     FT054760
IOLSTS  LFT     BARRY           IS IT AN ARRAY                          FT054770
        TRA     IOLSTU          YES                                     FT054780
        LFT     BEXTF+BINTF     IS IT A FUNCTION OF SOME SORT           FT054790
        ERROR   47,SKEND        YES, INVALID LIST ITEM                  FT054800
        ZET     IOFLAG          IS THIS INPUT OR OUTPUT                 FT054810
        TRA     IOLSTT          INPUT                                   FT054820
        CODE    LDQ,A           GENERATE LDQ  (ITEM)                    FT054830
        CODEN   STR                      STR                            FT054840
        TRA     IOLSTX          X                                       FT054850
IOLSTT  CODEN   STR             GENERATE STR                            FT054860
        CODE    STQ,A                    STQ  (ITEM)                    FT054870
        TRA     IOLSTX          X                                       FT054880
IOLSTU  ZET     IOFLAG          INPUT OR OUTPUT                         FT054890
        TRA     IOLSTV          INPUT                                   FT054900
        TSX     OUTPTV,4        OUTPUT (SLO) IN TRANSFER VECTOR         FT054910
        PZE     (SLO)           X                                       FT054920
        TRA     IOLSTW          X                                       FT054930
IOLSTV  TSX     OUTPTV,4        OUTPUT (SLI) IN TRANSFER VECTOR         FT054940
        PZE     (SLI)                                                   FT054950
IOLSTW  CAL     SAVE            GENERATE PZE  (ARRAY + 1)               FT054960
        ANA     =077777                                                 FT054970
        TSX     CITBLD,4,D      X                                       FT054980
        LXA     SAVE,4          GENERATE PZE  (LENGTH)                  FT054990
        CAL     EQUIV,4                                                 FT055000
        PAX     ,4              X                                       FT055010
        CAL     DIMTBL-1,4      X                                       FT055020
        ANA     =077777         X                                       FT055030
        TSX     CITBLD,4,N                                              FT055040
        TXI     EXIOLS-1,1,1                                            FT055050
IOLSTK  PIA                     SUBSCRIPTED ITEM.  GO PUT OUT CODE      FT055060
        TSX     ICODEX,4        FOR THE SUBSCRIPT                       FT055070
        TRA     IOLSTS+2                                                FT055080
        REM                                                             FT055090
IOLSTX  EQU     EXIOLS-2                                                FT055100
```

```
         TTL     P A S S  1  -  A R I T H M E T I C  S T A T E M E N T S  FT055110
*        SUBSTITUTION STATEMENT PROCESSOR                                FT055120
*        ENTRY...   1ARITH                                               FT055130
         REM                                                             FT055140
*        THE PROCESSOR IS ENTERED BY DEFAULT VIA THE RESERVED WORD TESTS. FT055150
*        IN OTHER WORDS, STATEMENTS FAILING TO MEET REQUIREMENTS FOR     FT055160
*        RESERVED WORDS ARE CONSIDERED TO BE ARITHMETIC STATEMENTS.      FT055170
*        THE COMPIL ROUTINE IS ENTERED IMMEDIATELY, AND THE STATEMENT    FT055180
*        PROCESSED.  THE STATEMENT MUST, OF COURSE, CONTAIN AN EQUAL     FT055190
*        SIGN.                                                           FT055200
         REM                                                             FT055210
*        IF CELL 2 HAS BEEN SAVED (FOR DP AND CA STATEMENTS), IT IS      FT055220
*        RESTORED PRIOR TO EXITING FROM THIS PROCESSOR.                  FT055230
         SPACE   3                                                       FT055240
1ARITH   AXC     0,2            SET POINTER TO FIRST LETTER OF STATEMENT FT055250
         TSX     COMPIL,4                                                FT055260
         NZT     EQULFG         DID AN EQUAL SIGN OCCUR                  FT055270
         ERROR   60,SKEND       NO = SIGN.  THEREFORE NOT A SUBSTITUTION FT055280
         REM                    STATEMENT                                FT055290
         TSX     PUTCL2,4       RESTORE CELL 2                           FT055300
         TRA     SKEND                                                   FT055310
```

```
*       COMPILE SUBROUTINE FOR ARITHMETIC SEQUENCES                     FT055330
*       ENTRIES...  COMPIL  (ARITHMETIC STATEMENTS)                     FT055340
*                   CMPLIF  (IF STATEMENTS)                             FT055350
*                   CMPLCL  (CALL STATEMENTS)                           FT055360
        REM                                                             FT055370
*       THIS ROUTINE IS THE HEART OF ARITHMETIC STATEMENT PROCESSING.   FT055380
*       IT IS ENTERED FROM THE SUBSTITUTION STATEMENT PROCESSOR, THE    FT055390
*       IF-STATEMENT PROCESSOR, AND THE CALL STATEMENT PROCESSOR.       FT055400
        REM                                                             FT055410
*       ELEMENTS OF THE STATEMENT (OPERATORS AND OPERANDS) ARE SCANNED  FT055420
*       OFF BY THE SCAN ROUTINE AND PLACED INTO A STACK (THE ARITHMETIC FT055430
*       STACK, ASTACK) ACCORDING TO CERTAIN RULES OF OPERATOR           FT055440
*       HEIRARCHY.  SEE THE DESCRIPTION OF THE OPERATOR RANK TABLE      FT055450
*       FOR THE ORDER AND HEIRARCHY OF OPERATORS.                       FT055460
        REM                                                             FT055470
*       THE GENERATION OF OBJECT PROGRAM CODE IS DELAYED AS LONG        FT055480
*       AS POSSIBLE, I.E. THE STACK IS BUILT UNTIL AN OPERATOR IS       FT055490
*       ENCOUNTERED WHICH CANNOT BE PLACED IN THE STACK BECAUSE OF THE  FT055500
*       OPERATOR HEIRARCHY RULES.  CODE IS THEN PRODUCED, LAST-ENTERED  FT055510
*       ITEMS FIRST, AND THE STACK SHORTENED UNTIL A POINT IS REACHED   FT055520
*       THAT ALLOWS THE WAITING OPERATOR TO BE ENTERED INTO THE STACK.  FT055530
*       STACK BUILDING THE COMMENCES AS BEFORE.  WHEN THE               FT055540
*       END-OF-STATEMENT IS REACHED, CODE IS PRODUCED UNTIL THE         FT055550
*       STACK IS CLEARED.                                               FT055560
        REM                                                             FT055570
*       THE FORM OF AN ASTACK ENTRY IS AS FOLLOWS...  FOR OPERATORS,    FT055580
*       THE OPERATOR KEY (FROM OPERATOR RANK TABLE) IS USED.  FOR       FT055590
*       SYMBOLIC OPERANDS, THE EQUIV WORD IS USED, WITH THE SUBROUTINE  FT055600
*       ARGUMENT BITS (BITS 15-20) CLEARED OUT, AND WITH THE EQUIV      FT055610
*       POINTER IN THE ADDRESS FIELD.  CONSTANT OPERANDS, AND TEMPORARY FT055620
*       OPERANDS PRODUCED AT INTERMEDIATE STAGES IN THE CODE GENERATION, FT055630
*       HAVE ONLY THE CORRECT MODE BITS (PREFIX) SET ON.                FT055640
        REM                                                             FT055650
*       OPERATORS ARE DISTINGUISHED FROM OPERANDS BY BIT 16...          FT055660
*               BIT 16 = 0    OPERAND                                   FT055670
*               BIT 16 = 1    OPERATOR                                  FT055680
        REM                                                             FT055690
*       CONSTANT OPERANDS ARE DISTINGUISHED FROM OTHERS BY BIT 17...    FT055700
*               BIT 17 = 0    VARIABLE OR INTERMEDIATE RESULT           FT055710
*               BIT 17 = 1    CONSTANT                                  FT055720
        REM                                                             FT055730
*       AN INTERMEDIATE RESULT HELD IN WORKING STORAGE WILL HAVE THE    FT055740
*       HIGH-ORDER BIT OF THE ADDRESS (BIT 21) ON, AND THE REMAINDER    FT055750
*       OF THE ADDRESS WILL BE THE WORKING CELL POINTER.                FT055760
        REM                                                             FT055770
*       A SUBSCRIPTED VARIABLE WILL HAVE A NON-ZERO TAG FIELD, AND      FT055780
*       THE ADDRESS WILL POINT TO THE SUBSCRIPT STACK (ISTACK) ENTRY.   FT055790
        REM                                                             FT055800
*       BIT 15 IS USED IN PROCESSING OF ARITHMETIC STATEMENT FUNCTIONS  FT055810
*       TO INDICATE A DUMMY ARGUMENT.                                   FT055820
        REM                                                             FT055830
        SPACE   3                                                       FT055840
CMPLCL  CAL     =1              ENTRY FOR CALL STATEMENT                FT055850
        SLW     CALLFG          SET SUBPROGRAM LEVEL TO 1               FT055860
        TRA     *+2                                                     FT055870
        REM                                                             FT055880
CMPLIF  STZ     CALLFG          ENTRY FOR IF STATEMENT.  ZERO CALL FLAG FT055890
        STL     EXITFG          ON FOR CALL AND IF STATEMENTS           FT055900
        TRA     *+3                                                     FT055910
```

```
        REM                                                     FT055920
COMPIL  STZ     EXITFG      ENTRY FOR ARITHMETIC STATEMENTS     FT055930
        STZ     CALLFG                                          FT055940
        SAVE    CMPLX                                           FT055950
        REM                                                     FT055960
        LXA     MODFLG,4    GET MODE OF THIS STATEMENT          FT055970
        CAL     GENDES,4    GET DESCRIPTOR FOR BINARY SEARCH IN GEN FT055980
        STA     GEN98       SET ADDRESS OF 'EQUAL COMPARE' INSTRUCTION FT055990
        STA     BAND        ORIGIN OF TABLE FOR THIS STATEMENT  FT056000
        STD     BA          LENGTH OF TABLE                     FT056010
        CAL     GENBIN,4    GET POWER-OF-TWO LENGTH OF TABLE    FT056020
        STA     BPOWER+1    AND THE RAISE/LOWER TABLE POINTER   FT056030
        ARS     18          AND PUT INTO THE BINSER INSTRUCTIONS. FT056040
        STA     BPOWER                                          FT056050
        REM                                                     FT056060
        CAL     MODFLG      GET MODE                            FT056070
        ARS     1           GET A POSSIBLE DP OR CA BIT INDICATOR FT056080
        STA     DBLMOD      SET THE DBLMOD FLAG                 FT056090
        REM                                                     FT056100
        STZ     EQULFG                                          FT056110
        STZ     REGIST                                          FT056120
        STZ     REGIST+1                                        FT056130
        STZ     REGFLG                                          FT056140
        STZ     PCNT                                            FT056150
        STZ     ARITFN                                          FT056160
        STZ     ARFLAG                                          FT056170
        STZ     SUBFLG                                          FT056180
        STZ     CXFLAG                                          FT056190
        STZ     SAVCL2                                          FT056200
        STZ     SETUP2                                          FT056210
        STZ     HFLAG                                           FT056220
        AXT     POOL-ISTKFG,4    INITIALIZE THE ISTACK DIRECTOR ADDRESSFT056230
        SXA     ISTKFG,4    PUT INTO ADDRESS OF FLAG WORD       FT056240
        AXT     -1,1                                            FT056250
        REM                                                     FT056260
GETNXT  TSX     SCAN,4      TO SCANNER TO GET NEXT ITEM         FT056270
        TRA     EOEPRO      END OF STATEMENT ENCOUNTERED.       FT056280
        TRA     STACK       CONSTANT ENCOUNTERED.  EQUIVALENT   FT056290
        REM                 WILL BE IN LAC.                     FT056300
        TRA     CMP1        OPERATOR ENCOUNTERED.  EQUIVALENT WILL BE FT056310
        REM                 IN LAC.                             FT056320
STACK   TXI     *+1,1,1     OPERAND (SYMBOLIC) ENCOUNTERED.  EQUIV. FT056330
        REM                 IN LAC.                             FT056340
        SLW     ASTACK,1    STACK ITEM IN ARITHMETIC STACK      FT056350
        TXL     GETNXT,1,LASTAK   GO FOR MORE, UNLESS STACK OVERFLOWS FT056360
        REM                                                     FT056370
        STR     MSG100,,SKEND      ARITHMETIC STACK OVERFLOW    FT056380
        SPACE   6                                               FT056390
*       OPERATOR  PROCESSOR                                     FT056400
CMP1    STA     *+2         SAVE OP EQUIV TEMPORARILY AND  GO TO FT056410
        SLW     SYMTS       OPERATOR PROCESSOR.                 FT056420
        TRA     **                                              FT056430
```

```
        EJECT
*               PLUS PROCESSOR                                          FT056440
*                                                                       FT056450
*                                                                       FT056460
PLUSP   LDI     ASTACK,1        IS THE TOS AN OPERATOR.                 FT056470
        LFT     BOPRX                                                   FT056480
        TRA     GETNXT          PLUS IS UNARY.  WE IGNORE IT.           FT056490
        REM                                                             FT056500
ADTEST  CAL     SYMTS           GET OPERATOR LEVEL OF OPERATOR TO BE ADDED FT056510
        LDQ     ADDXXX          GET INSTRUCTION TO BE INSERTED          FT056520
        ANA     =03760000000                                            FT056530
        SLW     CMPS1                                                   FT056540
        STQ     ADDXXX+1        SET 'EQUAL CONDITION' INSTRUCTION       FT056550
ADTST2  CAL     ASTACK+1,1      GET PREVIOUS OPERATOR KEY               FT056560
        ANA     =03760000000                                            FT056570
        LAS     CMPS1                                                   FT056580
ADDXXX  TRA     (PROC3          LEVELS OF OPS PERMIT STACKING           FT056590
        REM                     OF NEW OPERAND                          FT056600
        ***     **              INSTRUCTION IS INSERTED HERE.           FT056610
        TSX     GEN,4           CANNOT ADD.  GENERATE CODE AND RECYCLE FOR FT056620
        TRA     ADTST2          ANOTHER TEST.                           FT056630
*                                                                       FT056640
*       MUST GENERATE CODE IF OPERATOR LEVEL OF OP. TO BE ADDED IS GREAT-FT056650
*       ER THAN THAT OF TOP OPERATOR IN STACK, OR CONVERSELY, IF TOP    FT056660
*       OPERATOR OF STACK HAS A LOWER LEVEL THAN OPERATOR TO BE ADDED.  FT056670
*       THE ACTION TAKEN FOR EQUAL OPERATOR LEVELS VARIES FROM OP TO OP. FT056680
*       THE PERTINENT TRANSFER INSTRUCTION IS SET FOR EACH TYPE OF OP.  FT056690
        SPACE   6                                                       FT056700
*               MINUS PROCESSOR                                         FT056710
*                                                                       FT056720
MINUSP  LDI     ASTACK,1        MUST DECIDED IF MINUS IS BINARY OR UNARY. FT056730
        LNT     BOPRX           IS TOS AN OPERATOR                      FT056740
        TXL     *+2,1,-1        BINARY MINUS UNLESS STACK IS EMPTY      FT056750
        TXI     STACK+1,1,1     UNARY MINUS.  STACK IT.                 FT056760
        REM                                                             FT056770
        REM                     CHANGE MINUS KEY                        FT056780
        REM                     SCANNER ALWAYS RETURNS THE UNARY MINUS  FT056790
        REM                     EQUIVALENT. MUST BE REPLACED BY BINARY  FT056800
        REM                     MINUS EQUIBALENT (NEGKEY) IF MINUS IS   FT056810
        REM                     BINARY.                                 FT056820
        CAL     NEGKEY                                                  FT056830
        SLW     SYMTS                                                   FT056840
MINUS1  LDQ     MINXXX          GET INSTRUCTION TO BE INSERTED          FT056850
        TRA     ADTEST+2                                                FT056860
        REM                                                             FT056870
MINXXX  TSX     STKPRT,4                                                FT056880
        SPACE   6                                                       FT056890
*               = SIGN PROCESSOR                                        FT056900
*                                                                       FT056910
EQUALP  NZT     EQULFG          HAS EQUAL SIGN APPEARED ALREADY.        FT056920
        ZET     EXITFG          NO.  IS THIS AN IF OR CALL STATEMENT.   FT056930
        ERROR   45,SKEND        YES                                     FT056940
        TXH     OTHROP+1,1,-2       ERROR IF STACK IS EMPTY             FT056950
        REM                                                             FT056960
        STL     EQULFG          TURN ON EQUAL SIGN FLAG                 FT056970
        REM                                                             FT056980
        LDI     ASTACK,1        GET SYMBOL                              FT056990
        LFT     BOPRX+BCNST     CANNOT BE OPERATOR OR CONSTANT          FT057000
        ERROR   84,SKEND                                                FT057010
        REM                                                             FT057020
        LXA     SCAN5B,4        GET EQUIV ADDRESS FOR PREVIOUS SYMBOL   FT057030
```

```
         LDI     EQUIV,4         GET EQUIV WORD                                  FT057040
         SIL     BLHSX           TURN ON THE DEFINITION FLAG FOR SYMBOL          FT057050
         STI     EQUIV,4         RESTORE                                         FT057060
         TRA     ADTEST+1                                                        FT057070
         SPACE   6                                                               FT057080
*                DIVIDE PROCESSOR                                                FT057090
*                                                                                FT057100
*                TREATMENT IS SAME AS OTHER ARITHMETIC OPERATORS,                FT057110
*                EXCEPT THAT TWO CONSECUTIVE DIVIDE OPERATORS CANNOT             FT057120
*                BE ALLOWED INTO THE STACK (THIS RESULTS IN INCORRECT            FT057130
*                COMPILATION).                                                   FT057140
         REM                                                                     FT057150
DIVIDP   TXH     OTHROP+1,1,-2       EXPRESSION MAY NOT START WITH DIVIDE        FT057160
         TRA     MINUS1                                                          FT057170
         SPACE   6                                                               FT057180
*                MULTIPLY PROCESSOR                                              FT057190
*                                                                                FT057200
OTHROP   TXL     ADTEST+1,1,-2 STACK NOT EMPTY                                   FT057210
         ERROR   11,SKEND        EXPRESSION MAY NOT START WITH ONE OF THESE      FT057220
         REM                     OPERATORS                                       FT057230
         SPACE   6                                                               FT057240
*                EXPONENTIAL PROCESSOR                                           FT057250
*                                                                                FT057260
EXPONP   TXH     OTHROP+1,1,-2        ** MAY NOT BEGIN AN EXPRESSION             FT057270
         LDQ     EXPXXX          GET INSTRUCTION TO BE INSERTED                  FT057280
         TRA     ADTEST+2                                                        FT057290
         REM                                                                     FT057300
EXPXXX   ERROR   74,SKEND        CONSECUTIVE ** OPS ARE ILLEGAL                  FT057310
```

```
        EJECT                                                                   FT057320
*               PAREN PROCESSING....CURRENT PAREN COUNT IS KEPT IN CELL         FT057330
*               PCNT.  THE MEANING OF A LEFT PAREN IS DETERMINED BY THE         FT057340
*               ITEM IMMEDIATELY PRECEEDING IT IN THE ARITHMETIC STACK.         FT057350
*               THERE ARE ESSENTIALLY 8 TYPES OF PARENS...                      FT057360
*                    TYPE 0 - NESTING PAREN ONLY                                FT057370
*                    TYPE 1 - EXTERNAL LIBRARY FUNCTION PAREN                   FT057380
*                    TYPE 2 - OPEN (INTERNAL) FUNCTION PAREN                    FT057390
*                    TYPE 3 - ARITHMETIC STATEMENT FUNCTION PAREN               FT057400
*                    TYPE 4 - EXTERNAL FORTRAN FUNCTION PAREN                   FT057410
*                    TYPE 5 - CALL STATEMENT PAREN                              FT057420
*                    TYPE 6 - IF STATEMENT PAREN                                FT057430
*                    TYPE 7 - SUBSCRIPT PAREN (DOES NOT GET INTO ASTACK)        FT057440
*                                                                               FT057450
*               THE TYPE THAT A LEFT PAREN IS WILL BE FOUND IN THW PAREN        FT057460
*               STACK (PSTACK), AS FOLLOWS...                                   FT057470
*                PZE  ASTACK POINTER OF ITEM BEFORE PAREN,,TYPE                 FT057480
*               PCNT SERVES ALSO AS THE POINTER FOR PSTACK.                     FT057490
*               PNFLAG HOLDS THE SAME AS THE CURRENT PSTACK ENTRY               FT057500
*                                                                               FT057510
*               IN ADDITION THE PAREN TYPE IS ALSO ENTERED IN THE ADDRESSFT057520
*               FIELD OF THE ASTACK EQUIVALENT OF THE PAREN.                    FT057530
*                                                                               FT057540
*               LEFT PAREN PROCESSING                                           FT057550
*                                                                               FT057560
LPARNP  LDI     ASTACK,1        GET CURRENT STACK ENTRY                         FT057570
        LFT     BOPRX           BIT ON DUE TO OPERATOR, OR DUE TO THE           FT057580
        TRA     *+3             ALL-BIT ASTACK+1 WORD (ENCOUNTERED FOR          FT057590
        REM                     IF AND CALL STATEMENTS)                         FT057600
        LFT     BARRY           IS IT AN ARRAY                                  FT057610
        TRA     (PROC4          YES.  PROCESS SUBSCRIPT.                        FT057620
        REM                                                                     FT057630
        LXA     PCNT,4          REMEMBER LOCATION IN ASTACK OF                  FT057640
        TXI     *+1,4,1         ITEM BEFORE LEFT PAREN FOR LATER CLASSIFIC-FT057650
        SXA     PCNT,4          ATION.                                          FT057660
        STZ     PSTACK,4                                                        FT057670
        TXH     (PROC5,1,-2     IF STACK IS EMPTY, IT MUST BE IF OR CALL        FT057680
        PXA     ,1                                                              FT057690
        STA     PSTACK,4        SET ASTACK POINTER IN PSTACK ADDRESS FOR        FT057700
        REM                     ALL BUT 'IF' (UNNECESSARY) AND 'CALL'           FT057710
        REM                     (ALREADY PROPERLY SET TO ZERO).                 FT057720
*                                                                               FT057730
*               NOW CLASSIFY TYPE OF PAREN ACCORDING TO NATURE OF TOP OF        FT057740
*               STACK                                                           FT057750
*                                                                               FT057760
        AXT     0,4             SET THE PAREN TYPE FLAG INDICATOR               FT057770
        LFT     BOPRX                                                           FT057780
        TRA     (PROC1          AN OPERATOR, SO PAREN IS MERELY NEXTING         FT057790
        SPACE   6                                                               FT057800
        REM                     ITEM PRECEDING PAREN IS AN OPERAND OF SOME      FT057810
        REM                     KIND. MAY BE EXTERNAL FUNCTION (LIBRARY OR      FT057820
        REM                     OTHERWISE), AN ARITHMETIC STATEMENT (INTER-FT057830
        REM                     NAL) FUNCTION, AN ARRAY, OR AN OPEN INTERNAFT057840
        REM                     L FUNCTION.                                     FT057850
        REM                                                                     FT057860
        LFT     BINTF           IS IT AN INTERNAL FUNCTION (OPEN OR             FT057870
        REM                     ARITHMETIC STATEMENT.)                          FT057880
        TRA     (PROC2          YES                                             FT057890
        REM                                                                     FT057900
        SPACE   6                                                               FT057910
```

```
         REM                        NOT AN ARRAY OR INTERNAL FUNCTION.                FT057920
         REM                        IS IT AN EXTERNAL LIBRARY OR AN EXTERNAL          FT057930
         REM                        FORTRAN FUNCTION                                  FT057940
         REM                                                                          FT057950
         LFT     BVARB+BCOMN+BEQUV                                                    FT057960
         ERROR   69,SKEND           VARIABLE-TYPE FLAGS MUST BE OFF                   FT057970
         REM                                                                          FT057980
         LNT     BEXTF                                                                FT057990
         TRA     *+4                NOT FLAGGED AS EXTERNAL AND IS ALSO NOT           FT058000
         REM                        AN ARRAY.  THEREFORE DEFINE IT TO BE A            FT058010
         REM                        FORTRAN (EXTERNAL) FUNCTION.                      FT058020
         LFT     BLIBF                                                                FT058030
         TXI     (PROC1-2,4,1       FLAG AS EXTERNAL LIBRARY FUNCTION                 FT058040
         TRA     (PROC6             FLAG AS EXTERNAL FORTRAN FUNCTION                 FT058050
         REM                                                                          FT058060
         PIA                                                                          FT058070
         PAX     ,4                                                                   FT058080
         REM                                                                          FT058090
         CAL     SYMTAB,4           GET BCD NAME                                      FT058100
         ERA     SBNAME             COMPARE WITH THIS PROGRAM'S NAME                  FT058110
         TNZ     *+2                                                                  FT058120
         ERROR   10,*+1             PROGRAM MAY NOT CALL ITSELF                       FT058130
         REM                                                                          FT058140
         LDI     EQUIV,4            GET EQUIVALENT OF OPERAND                         FT058150
         SIL     BEXTF              FLAG IT AS AN EXTERNAL FUNCTION                   FT058160
         STI     EQUIV,4                                                              FT058170
         STI     ASTACK,1                                                             FT058180
         PXA     ,4                 EQUIV POINTER                                     FT058190
         STA     ASTACK,1           FIX UP ASTACK WORD                                FT058200
         REM                                                                          FT058210
(PROC6   LXA     CALLFG,4           BUMP SUBPROGRAM COUNT BY 1                        FT058220
         TXI     *+1,4,1                                                              FT058230
         SXA     CALLFG,4                                                             FT058240
         AXT     4,4                                                                  FT058250
         TXH     *+2,1,0            EXTERNAL FUNCTION CANNOT BEGIN ASTACK             FT058260
         ERROR   84,SKEND                                                             FT058270
(PROC1   PXD     ,4                 ENTER PAREN TYPE NO. IN DECREMENT OF              FT058280
         REM                        PSTACK ENTRY FOR THIS LEFT PAREN.                 FT058290
         LXA     PCNT,4                                                               FT058300
         STD     PSTACK,4                                                             FT058310
         CAL     PSTACK,4           RESET PNFLAG                                      FT058320
         SLW     PNFLAG                                                               FT058330
(PROC3   CAL     SYMTS                                                                FT058340
         TXI     STACK+1,1,1                                                          FT058350
         REM                                                                          FT058360
         REM                        DECIDE IF AN ARITHMETIC ATATEMENT FUNCTION        FT058370
         REM                        NAME OR AN OPEN FUNCTION NAME.                    FT058380
         REM                                                                          FT058390
(PROC2   LNT     BOPNF                                                                FT058400
         TXI     (PROC1,4,3         ARITHMETIC STATEMENT FUNCTION.  MARK IT 3.        FT058410
         TXI     (PROC1,4,2         OPEN ROUTINE.  MARK IT 2.                         FT058420
         REM                                                                          FT058430
         REM                        PROCESS ARRAY SUBSCRIPT                           FT058440
(PROC4   TSX     1INDEX,4                                                             FT058450
         CAL     ASTACK,1           GET ISTACK POINTER IN ADDRESS                     FT058460
         TSX     ICODEX,4           PUT OUT ANY CODE NEEDED TO FORM SUBSCRIPT         FT058470
         STL     SUBFLG             SUBSCRIPT HAS OCCURRED IN STATEMENT               FT058480
         TRA     GETNXT                                                               FT058490
         REM                                                                          FT058500
         REM                                                                          FT058510
(PROC5   NZT     EXITFG             ON FOR IF AND CALL STATEMENTS
```

```
        ERROR   7,SKEND                                                 FT058520
        REM                                                             FT058530
        AXT     5,4             PAREN TYPE FIVE FOR CALL STATEMENT      FT058540
        NZT     CALLFG          ON FOR CALL STATEMENT                   FT058550
        TXI     (PROC1,4,1      PAREN TYPE 6 FOR IF STAETMENT           FT058560
        REM                                                             FT058570
        TXI     (PROC1,1,1      SUBROUTINE NAME WAS SNUCK INTO ASTACK   FT058580
        REM                     BY 1CALLX ROUTINE.  NOW ADJUST THE      FT058590
        REM                     ASTACK POINTER.                         FT058600
        SPACE   6                                                       FT058610
*               COMMA PROCESSOR                                         FT058620
*                                                                       FT058630
*               THIS SECTION DECIDES IF COMMA IS ...                    FT058640
*                 A) AT PAREN LEVEL ZERO IN WHICH CASE MUST BE END OF   FT058650
*               EXPRESSION.  VALID FOR I/O LISTS ONLY.                  FT058660
*                 B) A POSSIBLE COMPLEX CONSTANT IF PAREN BEFOR THIS    FT058670
*               COMMA IS MEARELY NEXTING.                               FT058680
*                 C) END OF A SUBPROGRAM ARGUMENT IF PAREN BEFORE       FT058690
*               THIS COMMA IS OF TYPE 1,2,3,4, OR 5.                    FT058700
*                                                                       FT058710
COMMAP  NZT     PCNT                                                    FT058720
        TRA     ADTEST+1        I/O LIST.  TREAT AS END OF EXPRESSION.  FT058730
        REM                                                             FT058740
        LXD     PNFLAG,4        GET TYPE OF PREVIOUS PAREN              FT058750
        TXH     ADTEST+1,4,0    STACK IF PAREN NOT NESTING              FT058760
        REM                                                             FT058770
        STL     CXFLAG          POSSIBLE COMPLEX CONSTANT.              FT058780
        TRA     (PROC3          SET THE FLAG AND STACK THE COMMA        FT058790
        SPACE   6                                                       FT058800
*               CLOSE PAREN PROCESSING                                  FT058810
*                                                                       FT058820
*               CLOSE PARENS OF OF THE FOLLOWING TYPES...               FT058830
*               1.  AT PAREN LEVEL ZERO.  ERROR                         FT058840
*               2. AS MATCHING PAREN FOR NESTING LEFT PAREN. IF CXFLAG ISFT058850
*               ON, EXPRESSION MUST BE COMPLEX CONSTANT. OTHERWISE, CLOSEFT058860
*               PAREN SIMPLY FORCES GENERATION OF CODE.                 FT058870
*               3.  INDICATE END OF CALLING SEQUENCE FOR INTERNAL AND   FT058880
*                    EXTERNAL FUNCTIONS, AND SUBROUTINE CALLS.          FT058890
*               4.  INDICATES END OF ARITHMETIC EXPRESSION IN           FT058900
*               IF STATEMENT.                                           FT058910
*                                                                       FT058920
RPARNP  ZET     CXFLAG          HAS A POSSIBLE COMPLEX CONSTANT SITUATIONFT058930
        TRA     CXCHK           YES.  CHECK IF TOP OF STACK IS A COMPLEX FT058940
        REM                     CONSTANT OR IF IT IS AN ERROR           FT058950
        NZT     PCNT                                                    FT058960
        ERROR   91,SKEND        TOO MANY RIGHT PARENS                   FT058970
        REM                                                             FT058980
        REM                     NOW GENERATE CODE UNTIL FORM (A OR ,A IS FT058990
        REM                     REACHED.                                FT059000
        TSX     STKPRT,4        DUMP ASTACK                             FT059010
RPN1    CAL     ASTACK+1,1                                              FT059020
        LAS     LPNKEY                                                  FT059030
        TRA     *+2                                                     FT059040
        TRA     RPN2                                                    FT059050
        LAS     CMAKEY                                                  FT059060
        TRA     *+2                                                     FT059070
        TRA     RPN2                                                    FT059080
        TSX     GEN,4           GENERATE                                FT059090
        TRA     RPN1                                                    FT059100
        REM                                                             FT059110
```

```
RPN2    LDC     PNFLAG,4                                                FT059120
        TRA*    *+1,4                                                   FT059130
        TRA     )PROC0          0 - MATCHING PAREN MERELY NESTING       FT059140
        TRA     )PROC1          1 - EXTERNAL LIBRARY FUNCTION           FT059150
        TRA     )PROC2          2 - OPEN (INTERNAL) FUNCTION            FT059160
        TRA     )PROC3          3 - ARITHMETIC STATEMENT FUNCTION       FT059170
        TRA     )PROC4          4 - EXTERNAL FORTRAN FUNCTION           FT059180
        TRA     )PROC5          5 - CALL STATEMENT                      FT059190
        TRA     )PROC6          6 - IF STATEMENT                        FT059200
        SPACE   6                                                       FT059210
        REM                     NESTING PAREN                           FT059220
        REM                                                             FT059230
)PROC0  CAL     ASTACK,1        REMOVE THE NESTING PAREN                FT059240
        SLW     ASTACK+1,1      MOVE THE OPERAND UP IN THE STACK        FT059250
        PXA     ,1              DETERMINE IF THIS OPERAND IS IN A REGISTER FT059260
        TXI     *+1,1,-1        ADJUST THE ASTACK POINTER               FT059270
        ERA     REGIST+1                                                FT059280
        TNZ     )2.1            NOT IN REGISTER                         FT059290
        SXA     REGIST+1,1      REVISE THE REGISTER CELL POINTER        FT059300
        TRA     )2.1                                                    FT059310
        SPACE   6                                                       FT059320
*                               LIBRARY FUNCTION PROCESSING             FT059330
)PROC1  TXI     *+1,1,-1        ADJUST THE END OF STACK POINTER         FT059340
        SXD     )1.1,1          SET END OF STACK-1 IN TESTING CELLS     FT059350
        SXD     )1.4,1                                                  FT059360
        SXD     )1.6,1                                                  FT059370
        SXD     )1.11,1                                                 FT059380
        STZ     ARFLAG                                                  FT059390
        LXA     PNFLAG,1        SET POINTER TO START OF FUNCTION STACK  FT059400
        TXI     *+1,1,2                                                 FT059410
        REM                                                             FT059420
        ZET     DBLMOD                                                  FT059430
        TRA     )1.19           SKIP IF DOUBLE OR COMPLEX               FT059440
        CAL     REGIST                                                  FT059450
        LBT                     IS A RESULT IN THE ACCUM.               FT059460
        TRA     )1.19           NO                                      FT059470
        REM                                                             FT059480
        CAL     ASTACK,1        GET ASTACK WORD FOR OPERAND             FT059490
        ERA     REGIST+1        IS IT SAME AS THE ACCUMULATOR.          FT059500
        TZE     )1.1            YES.  SKIP THE CLA INSTRUCTION          FT059510
)1.19   TSX     STORE,4         NO.  STORE OFF THE PRESENT REGISTER     FT059520
        TSX     )1.22,4         GO LOAD UP THE CODE ROUTINE CELLS       FT059530
        REM                                                             FT059540
        CAL     MODFLG                                                  FT059550
        LAS     =1              DETERMINE COLUMN 1 MODE OF STATEMENT    FT059560
        TRA     )1.18           DOUBLE OR COMPLEX STATEMENT             FT059570
        TRA     )1.8            LOGICAL                                 FT059580
        CODE    CLA,A           PUT OUT THE CODE TO LOAD THE AC         FT059590
        REM                                                             FT059600
)1.1    TXH     )1.2,1,**       QUIT IF NO MORE ARGUMENTS               FT059610
        TXI     *+1,1,2                                                 FT059620
)1.4    TXH     )1.3,1,**       GO LOAD MQ IF ONLY ONE MORE ARGUMENT    FT059630
        AXT     TOPCOR-1,4      INITIALIZE THE STORAGE LOCATION         FT059640
        SXA     STOCEL,4        FOR THE FUNCTION ARGUMENTS.             FT059650
        REM                                                             FT059660
)1.5    TXI     *+1,1,2                                                 FT059670
        TSX     )1.22,4         GO LOAD UP THE CODE ROUTINE CELLS A, A+1 FT059680
        CODE    LDQ,A           PREPARE TO STORE OFF THIS ARGUMENT      FT059690
        REM                                                             FT059700
        CAL     STOCEL          GET LOCATION FOR THIS ARGUMENT          FT059710
```

```
        SUB     =1
        SLW     STOCEL                                                  FT059720
        CODEO   STQ,N           STORE OFF THIS ARGUMENT                 FT059730
)1.6    TXL     )1.5,1,**       RETURN IF MORE ARGUMENTS                FT059740
        REM                                                             FT059750
)1.3    LXA     PNFLAG,1        GET ASTACK POINTER TO TOP OF FUNCTION STACKFT059760
        TXI     **+1,1,4        BUMP TO THE MQ ARGUMENT                 FT059770
        TSX     )1.22,4         GO LOAD UP THE CODE ROUTINE CELLS A,A+1 FT059780
        CODE    LDQ,A                                                   FT059790
        REM                                                             FT059800
)1.2    ZET     ARITFN          ARE WE IN AN ARITH. STATEMENT FUNCTION. FT059810
        TRA     )1.20           YES                                     FT059820
        REM                                                             FT059830
)1.21   LXA     PNFLAG,1                                                FT059840
        CAL     ASTACK,1                                                FT059850
        LDI     ASTACK,1                                                FT059860
        ANA     =077777         SAVE ONLY EQUIV POINTER                 FT059870
        LFT     BARGT           IS IT A SUBPROGRAM PARAMETER.           FT059880
        TRA     )1.7            YES.                                    FT059890
        REM                                                             FT059900
        CODEO   TSX4,T,XRFLAG       CODE IS...  TSX FUNCTION,4          FT059910
        TRA     )5.6+2          PUT OUT RESULT                          FT059920
        REM                                                             FT059930
)1.7    TSX     PROLOG,4        MAKE PROLOG ENTRY                       FT059940
        CODEC   TSX4,N,XRFLAG       CODE IS...  TSX **,4                FT059950
        TRA     )5.6+2                                                  FT059960
        REM                                                             FT059970
)1.8    CODE    CAL,A           LOGICAL STATEMENT.  CODE IS...  CAL A  FT059980
        TRA     )1.1                                                    FT059990
        SPACE   2                                                       FT060000
)1.18   TSX     PUTCL2,4        RESTORE CELL 2                          FT060010
        CAL     =0100000                                                FT060020
        SLW     STOCEL          INITIALIZE HIGH-CORE POINTER            FT060030
        TRA     )1.10                                                   FT060040
        REM                                                             FT060050
)1.16   TSX     )1.22,4         GO LOAD UP THE CODE ROUTINE CELLS A, A+1FT060060
        REM                                                             FT060070
)1.10   LFT     /MINTG*700000                                           FT060080
        TRA     )1.12           REAL (DP OR CA) VARIABLE                FT060090
        CODE    CLA,A           INTEGER                                 FT060100
)1.13   CAL     STOCEL          GET POINTER                             FT060110
        SUB     =1              LOWER BY 1                              FT060120
        CODEO   STO,N                                                   FT060130
        CAL     STOCEL                                                  FT060140
        SUB     =2              LOWER BY 2                              FT060150
        SLW     STOCEL          RESTORE THE POINTER                     FT060160
        LFT     /MINTG*700000                                           FT060170
        TRA     )1.14           NOT AN INTEGER                          FT060180
        CODEO   STZ,N           ZERO THE OTHER HIGH-CORE LOCATION       FT060190
)1.11   TXH     )1.2,1,**       QUIT WHEN ALL ARGUMENTS DONE            FT060200
        TXI     )1.16,1,2       BUMP ASTACK POINTER AND RETURN FOR MORE FT060210
        REM                                                             FT060220
)1.12   CODLD   A               GET STORAGE INTO AC AND MQ              FT060230
        TRA     )1.13                                                   FT060240
)1.14   CODEO   STQ,N                                                   FT060250
        TRA     )1.11                                                   FT060260
        REM                                                             FT060270
)1.20   CAL     PCOUNT          ARITFN IS ON.  SAVE XR4 IN THE CODE     FT060280
        ADD     =2              BUMP PCOUNT TO AFTER THE 'TSX'          FT060290
        ORA     SXA4            GET OP CODE                             FT060300
                                                                        FT060310
```

```
        TSX     CITBLD,4,P      CODE IS...  SXA *+2,4                       FT060320
        TRA     )1.21                                                       FT060330
        REM                                                                 FT060340
)1.22   LDI     ASTACK,1        GET THE FIRST ARGUMENT                      FT060350
        LFT     BEXTF+BINTF     ARGUMENT CANNOT BE SUBPROGRAM NAME          FT060360
        ERROR   76,SKEND                                                    FT060370
        STI     A               SET WORDS FOR CODE ROUTINE                  FT060380
        SXA     A+1,1                                                       FT060390
        TRA     1,4                                                         FT060400
        SPACE   6                                                           FT060410
)PROC2  TSX     OPNGEN,4        PUT OUT THE CODE FOR BUILT-IN FUNCTION      FT060420
)2.1    LXA     PCNT,4          REMOVE PAREN FROM STACK                     FT060430
        TXI     *+1,4,-1        LOWER PCNT                                  FT060440
        SXA     PCNT,4                                                      FT060450
        CAL     PSTACK,4                                                    FT060460
        SLW     PNFLAG          RESET PNFLAG TO CURRENT PAREN TYPE          FT060470
        TRA     GETNXT          EXIT AND CONTINUE THE SCAN                  FT060480
        SPACE   6                                                           FT060490
*       ARITHMETIC STATEMENT FUNCTION PROCESSING                            FT060500
        REM                                                                 FT060510
*       THE CALLING SEQUENCE FOR ARITHMETIC STATEMENT FUNCTIONS IS          FT060520
*       THE SAME AS FOR FORTRAN FUNCTIONS, EXCEPT THAT ANY DOUBLE-LENGTH    FT060530
*       ARGUMENTS WILL HAVE TWO ENTRIES IN THE CALLING SEQUENCE.            FT060540
*       THE SECOND ENTRY HAS THE ADDRESS OF THE SECOND (LOW-ORDER)          FT060550
*       PART OF THE ARGUMENT.                                               FT060560
        REM                                                                 FT060570
)PROC3  STL     ARFLAG          FLAG THAT THIS IS INTERNAL FUNCTION CALL    FT060580
        LXA     PNFLAG,4        GET START OF THIS FUNCTION STRING           FT060590
        CAL     ASTACK,4        AND GO GET THE FUNCTION KEY FROM STACK      FT060600
        PAX     ,4              EQUIV POINTER                               FT060610
        CAL     EQUIV,4         EQUIV WORD                                  FT060620
        PAX     ,4              ARITH. STATEMENT FUNCTION STACK POINTER     FT060630
        TXI     *+1,4,1         BUMP TO SECOND WORD                         FT060640
        SXD     )5.35,4         SET DECREMENTS FOR LATER TESTS              FT060650
        SXD     )5.37,4                                                     FT060660
        CAL     POOL,4          MODE IS IN ADDRESS OF SECOND STACK WORD     FT060670
        ERA     MODFLG          COMPARE WITH THIS STATEMENT'S MODE          FT060680
        ANA     =077777                                                     FT060690
        TZE     )5.30           SAME MODES.  OKAY                           FT060700
        ERROR   35,SKEND        INCONSISTENT MODES                          FT060710
        SPACE   6                                                           FT060720
*                               CALL AND FORTRAN FUNCTION PAREN PROCESSING  FT060730
)PROC4  EQU     *               ENTRY FOR FORTRAN FUNCTION PAREN            FT060740
)PROC5  EQU     *               ENTRY FOR CALL STATEMENT PAREN              FT060750
        REM                                                                 FT060760
        LXA     CALLFG,4        REDUCE THE SUBPROGRAM COUNT BY 1            FT060770
        TXI     *+1,4,-1                                                    FT060780
        SXA     CALLFG,4                                                    FT060790
        STZ     ARFLAG          NOT AN ARITH. STATEMENT FUNCTION            FT060800
        REM                                                                 FT060810
)5.30   TSX     STORE,4         STORE OFF ANY REGISTER RESULT               FT060820
        NZT     ARFLAG          SKIP IF THIS IS A CALL TO AN INTERNAL FN... FT060830
        TSX     PUTCL2,4        RESTORE CELL 2                              FT060840
        REM                                                                 FT060850
)5.10   SXD     )5.1,1          SET END OF STACK IN THE TESTING CELLS       FT060860
        SXD     )5.4,1                                                      FT060870
        STZ     ARGCNT          ZERO THE NUMBER OF ARGUMENTS                FT060880
        STZ     SPCELL          ZERO THE SPECIAL PCOUNT INDICATOR           FT060890
        REM                                                                 FT060900
        LXA     PNFLAG,1        BEGINNING OF SUBPROGRAM STRING IN ASTACK    FT060910
```

```
        TXI     *+1,1,2         BUMP TO FIRST ARGUMENT                          FT060920
)5.1    TXH     )5.2,1,**       QUIT IF OUT OF ARGUMENTS                        FT060930
        REM                                                                     FT060940
        LDI     ASTACK,1        GET ARGUMENT                                    FT060950
        TSX     ARGBMP,4        BUMP ARGUMENT COUNTER BY 1                      FT060960
        REM                                                                     FT060970
        LNT     BCNST           SKIP IF THE ARGUMENT IS A CONSTANT              FT060980
        RFT     40000           OR IN WORKING STORAGE                           FT060990
        TXI     )5.13,1,2                                                       FT061000
        REM                                                                     FT061010
        PIA                     GET EQUIV POINTER OR ISTACK POINTER             FT061020
        PAX     ,4                                                              FT061030
        RFT     700000          IS VARIABLE SUBSCRIPTED.                        FT061040
        CAL     POOL-1,4        YES.  GET EQUIV POINTER                         FT061050
        PAX     ,4              SET XR FOR GOOD THIS TIME                       FT061060
        CAL     LHSXWD          GET A DECREMENT BLHSX BIT                       FT061070
        NZT     ARFLAG          SKIP IF THIS IS A CALL TO AN INTERNAL FN.       FT061080
        ORS     EQUIV,4         PUT INTO THE EQUIV WORD                         FT061090
        REM                                                                     FT061100
        RFT     700000          IS ITEM SUBSCRIPTED.                            FT061110
        TRA     )5.17           YES                                             FT061120
        LNT     BARIT           IS THIS ARGT FROM AN INTERNAL FUNCTION...       FT061130
        TXI     )5.13,1,2       NO                                              FT061140
        REM                                                                     FT061150
        PIA                     ARGUMENT IS ALSO AN ARGUMENT IN THIS            FT061160
        ANA     =077777         ARITHMETIC STATEMENT FUNCTION.                  FT061170
        CODEO   CAL4,N          CODE IS...  CAL ARG.NO,4                        FT061180
        ZET     SPCELL          HAS A SPECIAL CELL ALREADY BEEN OBTAINED.       FT061190
        TRA     *+3                                                             FT061200
        TSX     NEXTLP,4                                                        FT061210
        SLW     SPCELL                                                          FT061220
        REM                                                                     FT061230
        CAL     ARGCNT          GET ARGUMENT NUMBER IN THIS FUNCTION            FT061240
        TSX     CITBLN,4,Q      PUT OUT AS A CONSTANT ADDEND                    FT061250
        CODECO  STA,SPCELL,R        CODE IS...  STA ARGCNT+SPCELL               FT061260
        REM                                                                     FT061270
        ZET     DBLMOD          CHECK IF THIS IS A DOUBLE-LENGTH ARGUMENT       FT061280
        NZT     ARFLAG          AND IF THIS IS ALSO AN ARITH. STATE. FN.        FT061290
        TXI     )5.1,1,2        NO                                              FT061300
        REM                                                                     FT061310
        LNT     MREAL           YES, IT IS.  IS THIS A REAL ARGUMENT.           FT061320
        TXI     )5.1,1,2        NO                                              FT061330
        REM                                                                     FT061340
        PIA                     PUT OUT THE CODE                                FT061350
        ACL     =1              BUMP ARG NO BY 1                                FT061360
        ANA     =077777         SAVE ONLY EQUIV POINTER                         FT061370
        CODEO   CAL4,N          CODE IS...  CAL ARG+1,4                         FT061380
        REM                                                                     FT061390
        TSX     ARGBMP,4        BUMP ARGUMENT COUNTER BY 1                      FT061400
        TSX     CITBLN,4,Q      PUT OUT AS CONSTANT ADDEND                      FT061410
        CODECO  STA,SPCELL,R    CODE IS...  STA ARGCNT+1+SPCELL                 FT061420
        TXI     )5.1,1,2        RETURN FOR ANOTHER ARGUMENT                     FT061430
        REM                                                                     FT061440
)5.17   PIA                     GET THE ISTACK POINTER                          FT061450
        PAX     ,4                                                              FT061460
        CAL     POOL,4          IF EITHER POOL WORD HAS A MINUS SIGN,           FT061470
        ORA     POOL-2,4        THEN THE SUBSCRIPT INVOLVES A VARIABLE.         FT061480
        PBT                                                                     FT061490
        TXI     )5.13,1,2       NO MINUS SIGNS.  IGNORE                         FT061500
        REM                                                                     FT061510
```

```
        REM                                                          FT061520
        REM             SUBSCRIPTED ARGUMENT INVOLVING VARIABLE      FT061530
        REM             SUBSCRIPT                                    FT061540
        ZET     SPCELL  HAS A SPECIAL CELL ALREADY BEEN OBTAINED.    FT061550
        TRA     )5.11   YES.  DON'T GET ANOTHER                      FT061560
        TSX     NEXTLP,4 GET A POOL POINTER (RELATIVE TO EQUIV)      FT061570
        SLW     SPCELL  SAVE THE POINTER.  THIS CELL IS USED TO      FT061580
        REM             RETAIN THE DEFINITION OF OF THE              FT061590
        REM             BEGINNING OF THE CALLING SEQUENCE.  IT       FT061600
        REM             REFERS TO EQUIV AS ITS BASE, AND NEEDS       FT061610
        REM             THE CIT FLAG 'R'.                            FT061620
)5.11   STI     A                                                    FT061630
        SXA     A+1,1                                                FT061640
        CODE    PXA4,A  CODE IS...  LXD VARB,4                       FT061650
        REM                         PXA ARRAY+CONADD,4               FT061660
        CAL     PCOUNT                                               FT061670
        SUB     =1                                                   FT061680
        CODEO   SUB,P   CODE IS...  SUB *-1                          FT061690
        REM                                                          FT061700
        CAL     ARGCNT                                               FT061710
        TSX     CITBLN,4,Q PUT OUT THE ADDITIVE CONSTANT             FT061720
        CODECO  STA,SPCELL,R CODE IS...  STA SPECIAL.CELL+ARGCNT     FT061730
        REM                                                          FT061740
        ZET     DBLMOD  IS THIS A DOUBLE-LENGTH STATEMENT.           FT061750
        NZT     ARFLAG  AND A CALL TO AN ARITH. STATEMENT FN...      FT061760
        TXI     )5.1,1,2 NO                                          FT061770
        REM                                                          FT061780
        LNT     MREAL   YES.  IS THIS ARGUMENT REAL.                 FT061790
        TXI     )5.1,1,2 NO                                          FT061800
        LXA     A,4     GET ISTACK POINTER                           FT061810
        CAL     POOL-1,4 2ND ISTACK WORD                             FT061820
        PAX     ,4      GET EQUIV POINTER                            FT061830
        CAL     EQUIV,4 EQUIV WORD                                   FT061840
        PAX     ,4      GET DIMTBL POINTER                           FT061850
        CAL     POOL-1,4 GET LENGTH OF THIS ARRAY                    FT061860
        ANA     =077777 SAVE EQUIV POINTER                           FT061870
        CODEO   NOP,N   CODE IS...  NOP LENGTH.OF.ARRAY              FT061880
        REM                                                          FT061890
        CAL     PCOUNT                                               FT061900
        SUB     =1                                                   FT061910
        CODEO   ADD,P   CODE IS...  ADD *-1                          FT061920
        REM                                                          FT061930
        TSX     ARGBMP,4 BUMP ARGUMENT COUNTER BY 1                  FT061940
        TSX     CITBLN,4,Q PUT OUT AS AN ADDEND                      FT061950
        CODECO  STA,SPCELL,R      CODE IS...  STA SPCELL+ARGCNT+1    FT061960
        TXI     )5.1,1,2 RETURN FOR ANOTHER ARGUMENT                 FT061970
        REM                                                          FT061980
)5.13   ZET     DBLMOD  IF THIS ARGUMENT IS A DOUBLE-LENGTH          FT061990
        NZT     ARFLAG  ARGUMENT FOR AN ARITHMETIC STATEMENT         FT062000
        TRA     )5.1    FUNCTION, THEN BUMP ARGCNT BY 1.             FT062010
        REM                                                          FT062020
        LNT     MREAL                                                FT062030
        TRA     )5.1    NOT A REAL ARGUMENT                          FT062040
        TSX     ARGBMP,4 BUMP ARGUMENT COUNTER BY 1                  FT062050
        TRA     )5.1                                                 FT062060
        REM                                                          FT062070
        REM                                                          FT062080
        REM             SECOND PASS.  PREPARE THE CALLING SEQUENCE   FT062090
)5.2    NZT     ARITFN  IS THIS AN ARITH. STATE. FN. DEFINITION..    FT062100
        TRA     )5.16   NO                                           FT062110
```

```
        REM                                                          FT062120
        CAL     ARGCNT          YES.  SAVE XR4                       FT062130
        ADD     =1                                                   FT062140
        TSX     CITBLN,4,Q      PUT OUT ADDEND                       FT062150
        CODECO  SXA4,SPCELL,R       CODE IS...  SXA END.OF.LIST+1,4  FT062160
        REM                                                          FT062170
)5.16   LXA     SPCELL,4                                             FT062180
        TXL     )5.9,4,0        TRA IF NO PRELIMINARY CODE WAS PRODUCED FT062190
        REM                                                          FT062200
        CAL     PCOUNT          DEFINE THE SPECIAL CELL AS THE BEGINNING FT062210
        STA     EQUIV,4         OF THE CALLING SEQUENCE.             FT062220
        REM                                                          FT062230
)5.9    LXA     PNFLAG,1        GET ASTACK POINTER                   FT062240
        CAL     ASTACK,1                                             FT062250
        STZ     ARGCNT          RESET THE ARGUMENT COUNTER           FT062260
        ZET     ARFLAG          IS THIS A CALL TO AN INTERNAL FUNCTION. FT062270
        TRA     )5.14           YES                                  FT062280
        REM                                                          FT062290
        LDI     ASTACK,1        GET FLAGS                            FT062300
        LFT     BARGT           IS IT A SUBROUTINE PARAMETER.        FT062310
        TRA     )5.8            YES                                  FT062320
        REM                                                          FT062330
        ANA     =077777                                              FT062340
        CODEO   TSX4,T,XRFLAG CODE IS...  TSX SUBPROGRAM,4           FT062350
        REM                                                          FT062360
        TXI     *+1,1,2         BUMP ASTACK POINTER                  FT062370
)5.4    TXH     )5.6,1,**       QUIT WHEN ALL ARGUMENTS DONE         FT062380
        LDI     ASTACK,1                                             FT062390
        STI     A                                                    FT062400
        REM                                                          FT062410
        TSX     ARGBMP,4        BUMP ARGUMENT COUNTER BY 1           FT062420
        REM                                                          FT062430
        NZT     ARFLAG          CONTINUE TO PROCESS THE ARGUMENTS IF THIS FT062440
        TRA     )5.38           IS NOT AN ARITH ST. FN. CALL         FT062450
        REM                                                          FT062460
        LXA     ARGCNT,4        GET NUMBER OF THIS ARGUMENT          FT062470
)5.35   TXI     *+1,4,**        BUMP BY ORIGIN OF THE STACK          FT062480
        PIA                                                          FT062490
        ERA     POOL,4          COMPARE THIS MODE WITH DEFINED MODE  FT062500
        ARS     33                                                   FT062510
        TZE     *+2             OKAY                                 FT062520
        ERROR   36,SKEND        DIFFERENT MODES.  ERROR              FT062530
        REM                                                          FT062540
)5.38   RFT     700000          IS ARGUMENT SUBSCRIPTED.             FT062550
        TRA     )5.5            YES.  CODE MAY HAVE ALREADY BEEN INSERTED FT062560
        REM                                                          FT062570
        LFT     BARIT           IS THIS ARGT OF AN INTERNAL FUNCTION. FT062580
        TRA     )5.20           YES                                  FT062590
        REM                                                          FT062600
)5.12   SXA     A+1,1                                                FT062610
        ZET     DBLMOD          IS THIS A DOUBLE-LENGTH STATEMENT,   FT062620
        NZT     ARFLAG          AND ARE WE CALLING AN ARITH. STATE. FN... FT062630
        TRA     )5.19           NO                                   FT062640
        REM                                                          FT062650
        LNT     MREAL           YES.  IS THIS A REAL ARGUMENT.       FT062660
        TRA     )5.19           NO                                   FT062670
        REM                                                          FT062680
        CAL     TSX             YES.  PREPARE TO PUT OUT BOTH TSX INSTRUCTS FT062690
        SLW     COD31           2ND INSTRUCTION FOR THE CODE ROUTINE FT062700
        STL     SPCDFG          SIGNAL CODE ROUTINE FOR 2 INSTRUCTIONS FT062710
```

```
        TSX     CODE+1,4        GO PUT OUT THE CODE                              FT062720
        PZE     TSX,,A                                                           FT062730
        TSX     ARGBMP,4        BUMP ARGUMENT COUNTER BY 1                       FT062740
        TXI     )5.4,1,2                                                         FT062750
        REM                                                                      FT062760
)5.19   CODE    TSX,A           CODE IS...  TSX ARGUMENT                         FT062770
        TXI     )5.4,1,2                                                         FT062780
        REM                                                                      FT062790
)5.5    LXA     A,4             GET ISTACK POINTER                               FT062800
        CAL     POOL,4          IF EITHER ISTACK WORD HAS A MINUS SIGN,          FT062810
        ORA     POOL-2,4        THE CODE HAS ALREADY BEEN PRODUCED.              FT062820
        PBT                                                                      FT062830
        TRA     )5.12           CONSTANT SUBSCRIPT.  PUT OUT CODE                FT062840
        REM                                                                      FT062850
)5.20   CODEN   TSX             NO ARGUMENT ADDRESS.  CODE IS...  TSX **       FT062860
        REM                                                                      FT062870
        ZET     DBLMOD                                                           FT062880
        NZT     ARFLAG                                                           FT062890
        TXI     )5.4,1,2                                                         FT062900
        REM                                                                      FT062910
        LNT     MREAL           IS IT REAL                                       FT062920
        TXI     )5.4,1,2        NO.  GO DO NEXT ARGUMENT                         FT062930
        REM                                                                      FT062940
        CODEN   TSX             PUT OUT SECOND TSX                               FT062950
        TSX     ARGBMP,4        BUMP ARGUMENT COUNTER BY 1                       FT062960
        TXI     )5.4,1,2                                                         FT062970
        REM                                                                      FT062980
)5.6    LXD     PNFLAG,1        GET PAREN TYPE                                   FT062990
        TXH     )5.7,1,4        TRA IF CALL STATEMENT                            FT063000
        REM                                                                      FT063010
        CAL     AXT4            PREPARE TO POSSIBLY RESTORE XR4 IN THE CODE      FT063020
        ZET     ARITFN          IS THIS IN AN ARITH. STATEMENT FUNCTION...       FT063030
        TSX     CITBLD,4,N      YES.  CODE IS...  AXT **,4                       FT063040
        REM                                                                      FT063050
        NZT     ARFLAG          IS THIS A CALL TO AN ARITH ST FN...              FT063060
        TRA     )5.36           NO                                               FT063070
        LXA     ARGCNT,4                                                         FT063080
)5.37   TXI     *+1,4,**        BUMP BY ORIGIN OF STACK +1                       FT063090
        PXD     ,4                                                               FT063100
        LXD     *-2,4           GET POINTER TO 2ND STACK WORD                    FT063110
        ERA     POOL,4          COMPARE ARGCNT+ORIGIN+1 WITH ORIGIN+LNGTH+1      FT063120
        ARS     18                                                               FT063130
        TZE     *+2             OKAY                                             FT063140
        ERROR   36,*+1          DIFFERENT NUMBER OF ARGUMENTS                    FT063150
        REM                                                                      FT063160
)5.36   AXC     )2.1-1,4        SET EXIT POINT FOR RESULT ROUTINE                FT063170
        SXA     GENXIT,4                                                         FT063180
        LXA     PNFLAG,1        FUNCTION PAREN.  RECORD RESULT IN                FT063190
        CAL     ASTACK,1        ASTACK, AND CONTINUE TO PROCESS STATEMENT.       FT063200
        ALS     2               MOVE PERTINENT MODE BIT INTO P-BIT               FT063210
        PBT                     BIT IS ON IF REAL RESULT                         FT063220
        RESULT  INTEGR,AC                                                        FT063230
        RESULT  REAL,AC                                                          FT063240
        REM                                                                      FT063250
)5.7    SKIP    63              END OF STATEMENT EXPECTED FOR CALL               FT063260
        ERROR   3,SKEND                                                          FT063270
        TRA     CMPLX                                                            FT063280
        REM                                                                      FT063290
)5.8    ANA     =077777         SAVE EQUIV POINTER                               FT063300
        TSX     PROLOG,4        MAKE PROLOGUE ENTRIES                            FT063310
```

```
        CODEC   TSX4,N,XRFLAG      CODE IS...  TSX **,4                          FT063320
        TXI     )5.4,1,2                                                        FT063330
        REM                                                                     FT063340
)5.14   PAX     ,4              PUT OUT TSX TO INTERNAL FUNCTION                FT063350
        CAL     EQUIV,4         GET THE EQUIV WORD                              FT063360
        PAX     ,4              AND GET THE ARITH. STATE. FN STACK POINTER      FT063370
        CAL     POOL,4          DEFINITION OF FUNCTION                          FT063380
        ANA     =077777                                                         FT063390
        CODEO   TSX4,P,XRFLAG       CODE IS...  TSX INTERNAL.FUNCTION,4         FT063400
        TXI     )5.4,1,2                                                        FT063410
        REM                                                                     FT063420
ARGBMP  CAL     ARGCNT          SUBROUTINE TO BUMP ARGUMENT COUNTER BY 1        FT063430
        ADD     =1                                                              FT063440
        SLW     ARGCNT                                                          FT063450
        TRA     1,4                                                             FT063460
        REM                                                                     FT063470
LHSXWD  VFD     018/BLHSX                                                       FT063480
        SPACE   6                                                               FT063490
*       TERMINATION OF IF STATEMENT COMPILE                                     FT063500
        REM                                                                     FT063510
)PROC6  TSX     PLACE,4         PUT RESULT INTO THE AC                          FT063520
        TSX     PUTCL2,4        RESTORE CELL 2                                  FT063530
        LXA     CMPLX,4                                                         FT063540
        TRA     1,4             RELOAD XR4 AND RETURN                           FT063550
```

```
        EJECT                                                                   FT063560
*       FINAL PROCESSING FOR ARITHMETIC STATEMENT FUNCTION DEFINITIONS.         FT063570
*       ENTERED VIA THE CODE GENERATORS AT END-OF-STATEMENT PROCESSING.         FT063580
        REM                                                                     FT063590
ASFEND  ZET     SUBFLG          WAS THERE ANY SUBSCRIPTING.                     FT063600
        ERROR   90,*+1          YES.  NOT ALLOWED                               FT063610
        LXA     ARITFN,4        START OF THE STACK                              FT063620
        CAL     POOL-1,4        SECOND STACK WORD (HAS END OF STACK POINT)      FT063630
        STD     ASFEN1          SET TRANSFER INSTRUCTION DECREMENT              FT063640
        ARS     18              MOVE INTO ADDRESS                               FT063650
        SUB     ARITFN          GET LENGTH OF STACK+1                           FT063660
        ORA     TRA04           OR IN THE TRANSFER INSTRUCTION                  FT063670
        TSX     CITBLD,4,N      PUT OUT THE RETURN CODE                         FT063680
        CAL     PCOUNT          SET LARSTS TO THE CURRENT LENGTH OF             FT063690
        SLW     LARSTS          ARITHMETIC STATEMENT FUNCTIONS.                 FT063700
        REM                                                                     FT063710
        LXA     ARITFN,4        GET START OF STACK                              FT063720
        TXI     *+1,4,2         BUMP TO START OF ARGUMENTS                      FT063730
ASFEN2  CAL     POOL,4          CHANGE ALL ARGUMENTS FROM THEIR BCD             FT063740
        TZE     ASFEN1-1        NAMES TO THEIR 'EQUIV-LIKE' MODE                FT063750
        LDI     =0              REPRESENTATIONS.                                FT063760
        LAS     =HI00000                                                        FT063770
        TRA     *+1                                                             FT063780
        LAS     =HN(((((                                                        FT063790
        SIL     MREAL                                                           FT063800
        TRA     *+2                                                             FT063810
        SIL     MINTG                                                           FT063820
        STI     POOL,4                                                          FT063830
        TXI     *+1,4,1         BUMP TO NEXT ARGUMENT                           FT063840
ASFEN1  TXL     ASFEN2,4,**     CONTINUE IF MORE ARGUMENTS                      FT063850
        REM                                                                     FT063860
        TSX     CITBLF,4        TELL CITBLD THAT THIS FUNCTION IS OVER          FT063870
        LXA     LWORKS,4        UPDATE THE GETWRK ROUTINE TO RESERVE AND        FT063880
        SXD     GETWR1,4        REMEMBER ANY WORKING CELLS USED IN              FT063890
        SXD     FREWR1,4        THIS ARITHMETIC STATEMENT FUNCTION CODE.        FT063900
        TRA     SKEND           QUIT                                            FT063910
```

```
        EJECT                                                           FT063920
*       ROUTINE TO CHECK VALIDITY OF A COMPLEX CONSTANT                 FT063930
        REM                                                             FT063940
CXCHK   CAL     MODFLG          GET COLUMN 1 MODE                       FT063950
        ERA     =3              IS IT A COMPLEX STATEMENT.              FT063960
        TNZ     ERROR7          NO.  BAD FORM SOMEWHERE                 FT063970
        REM                                                             FT063980
        LDI     ASTACK,1        GET CURRENT ASTACK WORD                 FT063990
        LNT     MREAL+BCNST     BITS MUST BE ON                         FT064000
CXCHK2  ERROR   43,SKEND                                                FT064010
        REM                                                             FT064020
        CAL     ASTACK+1,1      GET EITHER A MINUS OR A COMMA           FT064030
        LAS     UNEGKY          IS IT MINUS.                            FT064040
        TRA     *+2             NO                                      FT064050
        TXI     CXCHK1,1,-1     YES.  BUMP POINTER AND GO               FT064060
CXCHK3  ERA     CMAKEY          MUST BE A COMMA                         FT064070
        TNZ     CXCHK2                                                  FT064080
        REM                                                             FT064090
        LDI     ASTACK+2,1      GET FIRST PART KEY WORD                 FT064100
        LNT     MREAL+BCNST     BITS MUST BE ON                         FT064110
        ERROR   43,SKEND                                                FT064120
        REM                                                             FT064130
        CAL     ASTACK+3,1      GET EITHER A MINUS OR THE LEFT PAREN    FT064140
        LAS     UNEGKY          IS IT MINUS.                            FT064150
        TRA     *+2             NO                                      FT064160
        TXI     CXCHK4,1,-1     YES.  BUMP POINTER AND GO               FT064170
CXCHK5  ERA     LPNKEY          MUST BE THE LEFT PAREN                  FT064180
        TNZ     CXCHK2          ERROR IF NOT                            FT064190
        REM                                                             FT064200
        CAL     CX1             GET THE  TWO PARTS OF THE CONSTANT      FT064210
        LDQ     CX2                                                     FT064220
        TSX     ASCON2,4        GO PUT IN CONSTANT TABLE                FT064230
        ORA     REALCT          OR MODE AND BCNST TO THE CONSTANT POINTER FT064240
        LXA     PNFLAG,1        RESET ASTACK POINTER                    FT064250
        TXI     *+1,1,1         ADJUST ASTACK POINTER TO POINT AT PAREN  FT064260
        SLW     ASTACK,1        PUT CONSTANT WORD INTO ASTACK           FT064270
        STZ     CXFLAG          ZERO OUT THE COMPLEX CONSTANT INDICATOR FT064280
        TRA     )2.1            ADJUST PSTACK AND QUIT                  FT064290
        REM                                                             FT064300
CXCHK1  CLS     CX2             SET SIGN OF SECOND PART MINUS           FT064310
        STO     CX2                                                     FT064320
        CAL     ASTACK+1,1      GET THE NEXT ASTACK WORD                FT064330
        TRA     CXCHK3                                                  FT064340
        REM                                                             FT064350
CXCHK4  CLS     CX1             SET SIGN OF FIRST PART MINUS            FT064360
        STO     CX1                                                     FT064370
        CAL     ASTACK+3,1      GET THE NEXT ASTACK WORD                FT064380
        TRA     CXCHK5                                                  FT064390
        REM                                                             FT064400
REALCT  VFD     018/MREAL+BCNST                                         FT064410
```

```
        EJECT                                                           FT064420
*       END OF STATEMENT.  PROCESS THE STACK                            FT064430
        REM                                                             FT064440
EOEPRO  TXL     CMPLX,1,0       IS STACK EMPTY                          FT064450
        TSX     STKPRT,4        ********** DUMP ASTACK                 FT064460
        REM                                                             FT064470
        TSX     GEN,4           NO                                      FT064480
        TXH     *-1,1,0         GENERATE UNTIL ONLY ONE ITEM REMAINS    FT064490
        REM                                                             FT064500
CMPLX   RESTOR                                                          FT064510
        TRA     1,4                                                     FT064520
```

```
        EJECT                                                           FT064530
*               ROUTINE TO DUMP THE CONTENTS OF ASTACK ON THE LIST TAPE FT064540
        REM                                                             FT064550
STKPRT  TRA     1,4             ALTER THIS INSTRUCTION TO GET ASTACK    FT064560
        REM                     PRINT-OUT.                              FT064570
        SAVE    EXSTKP                                                  FT064580
        TSX     LIST,4                                                  FT064590
        PON     XYZYX,,4                                                FT064600
        TXI     *+1,1,1                                                 FT064610
        LDQ     ASTACK+1,1                                              FT064620
        TSX     CINSTR,4                                                FT064630
        SLW     EOEP1+3                                                 FT064640
        STQ     EOEP1+4                                                 FT064650
        TSX     LIST,4                                                  FT064660
        EOEP1,,5                                                        FT064670
        TIX     *-6,1,1                                                 FT064680
EXSTKP  RESTOR                                                          FT064690
        TRA     1,4             RETURN                                  FT064700
        REM                                                             FT064710
EOEP1   BCI     3,                                                      FT064720
        PZE                                                             FT064730
        PZE                                                             FT064740
        REM                                                             FT064750
XYZYX   BCI     3,                                                      FT064760
        BCI     1,ASTACK                                                FT064770
```

```
*       SCAN ROUTINE                                                    FT064790
        REM                                                             FT064800
*       THE SCAN ROUTINE IS DIVIDED INTO SEVERAL SECTIONS... THE        FT064810
*       SYMBOLIC NAME SCANNER, THE NUMBER SCANNER, THE HOLLERITH        FT064820
*       LITERAL SCANNER, AND THE OPERATOR SCANNERS.                     FT064830
        REM                                                             FT064840
*       ENTRY...                TSX SCAN,4                              FT064850
        REM                                                             FT064860
*       EXITS...                TRA ...  RETURN HERE IF END OF STATEMENT FT064870
*                               TRA ...  RETURN HERE IF A CONSTANT      FT064880
*                               TRA ...  RETURN HERE IF OPEARTOR        FT064890
*                               ... ...  RETURN HERE IF A SYMBOLIC OPERAND FT064900
        SPACE   2                                                       FT064910
*       ALL CONSTANTS ENCOUNTERED WILL BE ENTERED INTO THE CONSTANT     FT064920
*       TABLE CONTAB, AND THE CONSTANT EQUIVALENT WILL BE RETURNED TO   FT064930
*       THE LOGICAL ACCUMULATOR ON EXITING FROM SCAN.  FORMAT OF THE    FT064940
*       CONSTANT EQUIVALENT IS...                                       FT064950
        REM                                                             FT064960
*               BITS  S,1-2     MODE                                    FT064970
*               BITS   3-16     NOT USED                                FT064980
*               BIT      17     =1. INDICATES A CONSTANT.  THIS BIT IS  FT064990
*                               USED TO FLAG ENTRIES IN THE ASTACK ONLY. FT065000
*                               THIS BIT WILL BE OFF FOR VARIABLES IN   FT065010
*                               ASTACK.                                 FT065020
*               BITS  18-20     NOT USED                                FT065030
*               BITS  21-35     POINTER ADDRESS TO CONTAB.              FT065040
        SPACE   2                                                       FT065050
*       FOR SYMBOLIC OPERANDS AN ENTRY WILL BE MADE IN SYMTAB IF THE    FT065060
*       SYMBOL IS NOT ALREADY THERE.  THE MODE WILL BE SET.  IF THE     FT065070
*       SYMBOL IS ALREADY IN SYMTAB AND THE NAME DOES NOT END IN -F, IT FT065080
*       WILL BE ASSUMED TO BE CORRECT, AND THE EQUIVALENT WILL BE       FT065090
*       RETURNED TO THE LOGICAL ACCUMULATOR ON EXITING FROM THE SCAN.   FT065100
*       THE EQUIVALENT FOR SYMBOLIC OPERANDS WILL HAVE THE FORM...      FT065110
        REM                                                             FT065120
*               BITS S,1-14     SAME AS IN SYMTAB.                      FT065130
*               BITS  15-20     ALL ZERO.  IF THESE BITS ARE NEEDED,    FT065140
*                               THEY MUST BE OBTAINED FROM EQUIV.       FT065150
*               BITS  21-35     EQUIV POINTER.  ALLOWS ACTUAL SYMBOL    FT065160
*                               EQUIVALENT TO BE OBTAINED.  THIS ADDRESS FT065170
*                               MAY BE CHANGED TO POINT TO ISTACK IF THE FT065180
*                               VARIABLE IS SUBSCRIPIED.                FT065190
        REM                                                             FT065200
*       NOTE THAT THE ARGUMENT NUMBER FOR PARAMETERS IS NOT AVAILABLE   FT065210
*       IN THE ASTACK EQUIVALENT.                                       FT065220
        SPACE   2                                                       FT065230
*       OPERATORS RETURN AN EQUIVALENT ALSO TO THE LOGICAL ACCUMULATOR  FT065240
*       BEFORE EXITING FROM SCAN.  THIS EQUIVALENT IS TAKEN FROM THE    FT065250
*       OPERATOR RANK TABLE (Q.V., AND ALSO THE DEFINITION OF THE       FT065260
*       RANK MACRO).                                                    FT065270
        REM                                                             FT065280
*       THE ORDER OF THE SECTIONS OF THE SCANNER IS IMPORTANT.  THEY    FT065290
*       MUST BE...  SYMBOLIC, NUMBER, OPERATORS.  THIS THEN AFFORDS     FT065300
*       A CONVENIENT RAPID TEST, USED IN MANY PARTS OF FASTRAN, TO      FT065310
*       DETERMINE IF A CHARACTER IS NUMERIC, ALPHABETIC, OR SPECIAL.    FT065320
*       THE TEST IS MADE BY CONVERTING ON THE SCAN TABLE, AND DETERMININGFT065330
*       IF THE RESULTING LOCATION IS BELOW (LETTER), EQUAL, OR ABOVE    FT065340
*       (SPECIAL) THE ENTRY FOR NUMERAL (SCAN1).                        FT065350
        REM                                                             FT065360
        SPACE   3                                                       FT065370
```

```
SCAN    SXA     SCANX,1                                                 FT065380
        SXA     SCANX+1,4                                               FT065390
SCANA   LDQ     COLUMN,2        PICK UP CHARACTER AND CLASSIFY          FT065400
        CRQ     SCANT,1,1                                               FT065410
        SXA     *+1,1                                                   FT065420
        TRA     **                                                      FT065430
        REM          SCAN CONVERSION TABLE                              FT065440
SCANT   DUP     1,10                                                    FT065450
        SCAN1           00-11  NUMBERS                                  FT065460
        SCAN2           12 ERROR                                        FT065470
        SCAN13          13 =                                            FT065480
        SCAN2           14 '  (ERROR)                                   FT065490
        DUP     1,3                                                     FT065500
        SCAN2           15-17  ERROR                                    FT065510
        SCAN20          20 +                                            FT065520
        DUP     1,9                                                     FT065530
        SCAN5           21-31  LETTERS                                  FT065540
        SCAN2           32 ERROR                                        FT065550
        SCAN33          33 .                                            FT065560
        SCAN34          34 )                                            FT065570
        DUP     1,3                                                     FT065580
        SCAN2           35-37  ERROR                                    FT065590
        SCAN40          40 -                                            FT065600
        DUP     1,9                                                     FT065610
        SCAN5           41-51  LETTERS                                  FT065620
        SCAN2           52 ERROR                                        FT065630
        SCAN2         53 $                                              FT065640
        SCAN54          54 *                                            FT065650
        DUP     1,3                                                     FT065660
        SCAN2           55 - 57  ERROR                                  FT065670
        SCAN60          60 BLANK                                        FT065680
        SCAN61          61 /                                            FT065690
        DUP     1,8     62 - 71  LETTERS                                FT065700
        SCAN5                                                           FT065710
        SCAN2           72 ERROR                                        FT065720
        SCAN73          73 ,                                            FT065730
        SCAN74          74 (                                            FT065740
        SCAN2           75 ERROR                                        FT065750
        SCAN2           76 ERROR                                        FT065760
        SCAN77          77 END OF STATEMENT                             FT065770
        REM                                                             FT065780
```

```
         EJECT                                                          FT065790
         REM     SYMBOLIC NAME OR KEY WORD PROCESSOR                    FT065800
         REM                                                            FT065810
SCAN5    SXD     SCAN5A,2          SAVE POINTER POSITION OF 1ST CHARACTER FT065820
         CAL     COLUMN,2                                               FT065830
         TXI     *+1,2,-1          GET NEXT CHARACTER                   FT065840
         LDQ     COLUMN,2                                               FT065850
         CRQ     SCANT,1,1                                              FT065860
         TXL     *-3,1,SCAN1       IF IT'S ALPHAMERIC GO BACK FOR NEXT CHAR FT065870
         SXA     *+1,2           CALC. NUMBER OF CHARACTERS IN SYMBOL   FT065880
         AXC     **,4                                                   FT065890
SCAN5A   TXI     *+1,4,**                                               FT065900
         ZET     ARITFN          IS THIS STATEMENT AN ARITH. STATE. FN... FT065910
         TRA     SCANAE          YES.  GO CHECK FOR FUNCTION ARGUMENT   FT065920
         REM                                                            FT065930
SCAN5L   LDI     COLUMN-1,2      DOES NAME END IN AN F.                 FT065940
         IIL     260000                                                 FT065950
         LFT     770000                                                 FT065960
         TRA     *+2             NO                                     FT065970
         TXH     SCAN5H,4,3      YES.  IS IT 4 OR MORE LETTERS LONG     FT065980
         TXH     ERRO15,4,6      NO.  IS IT MORE THAN 6 CHARACTERS LONG. FT065990
         REM                     SYMBOLIC NAME FOUND.                   FT066000
         ANA     SCANT1,4          NO.  APPEND TRAILING BLAMKS.         FT066010
         ORA     SCANT2,4                                               FT066020
         SLW     FORM16+6                                               FT066030
         SLW     FORM69                                                 FT066040
         XCL                                                            FT066050
         TSX     LOCATE,4        PUT SYMBOLIC NAME IN SYMBOL TABLE      FT066060
         STA     SCAN5N          EQUIV POINTER                          FT066070
         TMI     SCAN5J          SKIP IF NEW SYMBOL                     FT066080
         REM                                                            FT066090
         LFT     BINTF                                                  FT066100
         ERROR   16,SKEND        INTERNAL FUNCTION MUST HAVE TERMINAL F FT066110
SCAN5C   RIL     700000          RESET MODE BITS                        FT066120
         LFT     BEXTF           IF BEXTF IS ON, RESET POSSIBLE BLIBF BIT FT066130
         RIL     BLIBF                                                  FT066140
         REM                                                            FT066150
SCAN5J   CAL     FORM69          BCD NAME OF SYMBOL                     FT066160
         SLW     FORM23+3                                               FT066170
         LAS     =HI00000        MODE TEST                              FT066180
         TRA     *+1                                                    FT066190
         LAS     =HN{{{{{                                               FT066200
         SIL     MREAL           FLOATING POINT                         FT066210
         TRA     SCAN5S                                                 FT066220
         SIL     MINTG           FIXED POINT                            FT066230
         REM                                                            FT066240
SCAN5T   LFT     BARRY           IS IT AN ARRAY.                        FT066250
         TRA     SCAN5N-1        YES.                                   FT066260
         REM                                                            FT066270
         SKIP1   LPCHAR          IS THERE A LEFT PAREN NEXT.            FT066280
         TXI     SCAN5K,2,1      YES.  RESTORE POINTER AND SKIP         FT066290
         LFT     BARGT+BEXTF                                            FT066300
         TRA     *+2             CANNOT TAKE ANY ACTION                 FT066310
         SIL     BVARB           SET VARIABLE BIT ON                    FT066320
SCAN5N   AXT     **,4                                                   FT066330
         STI     EQUIV,4         RESTORE THE EQUIV WORD                 FT066340
SCAN5E   STI     SCAN5B          FIX UP ASTACK EQUIV ENTRY              FT066350
         SXA     SCAN5B,4        INSERT EQUIV POINTER                   FT066360
         CAL     SCAN5B                                                 FT066370
         ANA     =0777770077777      CCLEAR OUT POSSIBLE ARG NO. AND    FT066380
```

```
SCANX   AXT     **,1                                                    FT066390
        AXT     **,4                                                    FT066400
        TRA     4,4                                                     FT066410
        REM                                                             FT066420
SCAN5K  LFT     BVARB+BCOMN+BEQUV       ALL FLAGS MUST BE OFF           FT066430
        ERROR   69,SKEND                                                FT066440
        SIL     BEXTF           MARK IT AS AN EXTERNAL FUNCTION         FT066450
        TRA     SCAN5N                                                  FT066460
        REM                                                             FT066470
SCAN5S  NZT     DBLMOD          IS IT DOUBLE OR COMPLEX.                FT066480
        TRA     SCAN5T          NO                                      FT066490
        LFT     BARRY           IF BARRY IS ON, BDOUB MUST ALREADY BE ON FT066500
        LFT     BDOUB                                                   FT066510
        TRA     *+2                                                     FT066520
        ERROR   49,*+2          SINGLE ARRAY USED IN DOUBLE CONTEXT     FT066530
        SIL     BDOUB           TURN ON THE DOUBLE LENGTH FLAG          FT066540
        TRA     SCAN5T                                                  FT066550
```

```
         EJECT                                                          FT066560
         REM      TEST SYMBOL FOR ARITHMETIC STATEMENT FUNCTION ARGUMENT FT066570
SCANAE   ANA      SCANT1,4       REMOVE NON-ESSENTIAL CHARACTERS         FT066580
         ORA      SCANT2,4       AND ADD BLANKS                          FT066590
         LXA      ARITFN,1       GET START OF STACK                      FT066600
         TXI      **+1,1,1                                               FT066610
         SXD      SCANAI,1       SAVE  IN FUTURE INSTRUCTION             FT066620
         TXI      **+1,1,1                                               FT066630
         REM                                                            FT066640
SCANAH   LAS      POOL,1         TEST IF THIS SYMBOL IS AN ARGUMENT      FT066650
         TRA      **+2                                                   FT066660
         TRA      SCANAG         YES.  MATCHES A MEMBER OF THE STACK     FT066670
         TXI      **+1,1,1                                               FT066680
SCANAF   TXL      SCANAH,1,**    RETURN FOR MORE IF NOT THROUGH THE STACK FT066690
         TRA      SCAN5L         RETURN TO MAIN SCANNER.  NOT AN ARGUMENT FT066700
         REM                                                            FT066710
SCANAG   LDI      =0                                                     FT066720
         LAS      =HI00000       DETERMINE THE MODE OF THE ARGUMENT      FT066730
         TRA      **+1                                                   FT066740
         LAS      =HN(((((                                               FT066750
         SIL      MREAL+BARIT    REAL MODE + ARGUMENT BIT                FT066760
         TRA      **+2                                                   FT066770
         SIL      MINTG+BARIT    FIXED MODE + ARGUMENT BIT               FT066780
         SIL      BARIT          TURN ON THE ARGUMENT BIT                FT066790
         STI      SCAN5B         RESET THE SAVE CELL                     FT066800
SCANAI   TIX      **+1,1,**      LOWER BY START OF STACK, THUS GETTING   FT066810
         SXA      SCAN5B,1       THE ARGUMENT NUMBER                     FT066820
         CAL      SCAN5B                                                 FT066830
         TRA      SCANX          EXIT THE SCANNER                        FT066840
```

```
        EJECT                                                                   FT066850
        REM     FUNCTION NAME ENCOUNTERED                                       FT066860
        REM                                                                     FT066870
SCAN5H  TXH     ERR015,4,7      IS NAME TOO LONG (GREATER THAN 7 CHAR.).        FT066880
        SXA     SCAN5Q,4        SAVE LENGTH OF SYMBOL                           FT066890
        SLW     FORM69          SAVE BCD NAME OF SYMBOL                         FT066900
        TXL     SCAN5M,4,6      IS IT LESS THAN SEVEN CHARACTERS.               FT066910
        REM                                                                     FT066920
        SKIP    LPCHAR          NO.  LEFT PAREN MUST BE NEXT                    FT066930
        ERROR   15,SKEND                                                        FT066940
        CAL     FORM69                                                          FT066950
        TXI     SCAN5U,2,1      RESTORE THE POINTER AND CONTINUE                FT066960
        REM                                                                     FT066970
SCAN5M  ANA     SCANT1,4        KILL THE NON-ESSENTIAL BITS OF SYMBOL           FT066980
        ORA     SCANT2,4        REPLACE BY BLANKS                               FT066990
        SLW     FORM69          SAVE BCD NAME OF SYMBOL                         FT067000
        REM                                                                     FT067010
        XCL                                                                     FT067020
        TSX     LOCATE,4        FIND IN SYMTAB                                  FT067030
        STO     SCAN5B          SAVE EQUIV POINTER AND ACCUM SIGN               FT067040
        STA     SCAN5N          SET EQUIV ADDRESS                               FT067050
        LFT     BARRY           IS IT AN ARRAY.                                 FT067060
        ERROR   20,*+1          YES.  NOT ALLOWED                               FT067070
        REM                                                                     FT067080
        SKIP1   LPCHAR          IS THERE A LEFT PAREN NEXT.                     FT067090
        TXI     SCAN5R,2,1      YES.  RESTORE THE POINTER AND GO                FT067100
        LFT     BINTF           NO PAREN.  IS FUNCTION BIT ON.                  FT067110
        ERROR   57,SKEND        YES.  NOT ALLOWED WITHOUT PAREN                 FT067120
        TRA     SCAN5C          TREAT AS REGULAR VARIABLE                       FT067130
        REM                                                                     FT067140
SCAN5R  CLA     SCAN5B          GET THE 'LOCATE' SIGN INDICATION                FT067150
        TPL     *+4             SKIP IF A OLD SYMBOL                            FT067160
        PAX     ,4              GET EQUIV POINTER                               FT067170
        STZ     SYMTAB,4        ZERO OUT INDICATIONS OF -F SYMBOL               FT067180
        REM                                                                     FT067190
        CAL     FORM69          GET BCD NAME OF SYMBOL                          FT067200
SCAN5Q  AXT     **,4            REGAIN THE LENGTH OF SYMBOL                     FT067210
        TXI     *+1,4,-1        PREPARE TO DELETE THE TERMINAL F                FT067220
        ANA     SCANT1,4                                                        FT067230
        ORA     SCANT2,4                                                        FT067240
SCAN5U  SLW     FORM23+3                                                        FT067250
        SLW     FORM69                                                          FT067260
        NZT     DBLMOD          IS IT DOUBLE PRECISION OR COMPLEX.              FT067270
        TRA     SCAN5P          NO.  SKIP                                       FT067280
        REM                                                                     FT067290
        LXD     SCAN5A,1        DOUBLE-LENGTH STATEMENT.  MUST ADD A            FT067300
        TXL     SCAN5P,1,0      'D' OR 'I' TO NAME... EXCEPT FOR                FT067310
        XCL                     ARITHMETIC STATEMENT FUNCTION NAMES.            FT067320
        SXA     SCAN5I,4        SAVE XR4                                        FT067330
        TSX     LOCATE,4        LOOK UP THE NAME IN SYMTAB                      FT067340
        LFT     BINTF           IF AN INTERNAL FUNCTION, THEN GO ON             FT067350
        TRA     SCAN5O          WITH PROCESSING                                 FT067360
        REM                                                                     FT067370
        TPL     SCAN5I          IF NOT A NEW SYMBOL, SKIP                       FT067380
        PAX     ,4              NEW SYMBOL.  REMOVE THE SYMBOL FROM             FT067390
        STZ     SYMTAB,4        THE TABLE                                       FT067400
SCAN5I  AXT     **,4            RESTORE XR4 AND APPEND 'D' OR 'I'.              FT067410
        TXH     ERR015,4,5      IS THE NAME TOO LONG.                           FT067420
        LDI     MODFLG          GET COLUMN 1 MODE                               FT067430
        LDQ     FORM69          A '1' IS INSERTED FOR THE 'I', AND A '4'        FT067440
```

```
         CAL     =1              FOR THE 'D'.  POST-PASS1 CHANGES THESE.  FT067450
         RNT     1               IS THIS A COMPLEX STATEMENT.             FT067460
         CAL     =4              NO.  SET UP THE DP LEADING BIT           FT067470
         LGL     30              RESTORE THE NEW NAME TO THE ACCUM        FT067480
         REM                                                              FT067490
         REM                     SEARCH BUILT-IN FUNCTION TABLE           FT067500
SCAN5P   AXT     LOPENT,4        LENGTH OF OPEN FUNCTION TABLE            FT067510
         LAS     LIBTAB+LOPENT+1,4       NAME FROM TABLE                  FT067520
         TRA     *+2             NOT FOUND YET                            FT067530
         TRA     ROPEN           OPEN FUNCTION LOCATED                    FT067540
         TIX     *-3,4,2                                                  FT067550
         REM                                                              FT067560
         XCL                                                              FT067570
         TSX     LOCATE,4                                                 FT067580
SCAN5O   STO     SCAN5B          SAVE EQUIV POINTER AND ACCUM SIGN        FT067590
         LFT     BARRY+BVARB+BCOMN+BEQUV     ALL FLAGS MUST BE OFF        FT067600
         ERROR   69,SKEND                                                 FT067610
         REM                                                              FT067620
         PAX     ,4              GET EQUIV POINTER                        FT067630
         RIL     700000          RESET THE MODE BITS                      FT067640
         CAL     FORM69          GET BCD NAME OF SYMBOL                   FT067650
         ERA     =HX                                                      FT067660
         ARS     30                                                       FT067670
         TZE     SCAN5F          FIXED MODE                               FT067680
         SIL     MREAL           SET REAL MODE BITS                       FT067690
         TRA     SCAN5G                                                   FT067700
         REM                                                              FT067710
SCAN5F   SIL     MINTG           INTEGR MODE BITS                         FT067720
         ZET     DBLMOD          IS IT DOUBLE-LENGTH STATEMENT.           FT067730
         ERROR   34,*+1                                                   FT067740
         REM                                                              FT067750
SCAN5G   LXD     SCAN5A,1        IS THIS THE START OF A STATEMENT.        FT067760
         TXL     SCANAA,1,0      IF SO, GO TO ARITH STATEMENT FN. PROCESSOR FT067770
         REM                                                              FT067780
         LNT     BINTF           IF OFF, THEN TURN ON BEXTF+BLIBF         FT067790
         SIL     BEXTF+BLIBF     NO.  DEFINE AS EXTERNAL LIBRARY FUNCTION FT067800
         TRA     SCAN5E-1                                                 FT067810
         REM                                                              FT067820
ROPEN    CAL     LIBTAB+LOPENT,4     GET KEY WORD FOR THIS FUNCTION       FT067830
         LXD     SCAN5A,4                                                 FT067840
         TXH     SCANX,4,0       OPERATOR (BUILT-IN) ROUTINE NAME         FT067850
         ERROR   88,SKEND        OPEN FUNCTION CANNOT BE DEFINED BY PROGRAM FT067860
```

```
         EJECT                                                          FT067870
*        ARITHMETIC STATEMENT FUNCTION DEFINITION.                      FT067880
         REM                                                            FT067890
*        THE EQUIV ENTRY FOR AN ARITHMETIC STATEMENT FUNCTION HAS THE   FT067900
*        ARITH. STATE. FN. STACK LOCATION IN THE ADDRESS, AND BINTF IS  FT067910
*        TURNED ON.  THE MODE OF THE FUNCTION IS SET IN THE PREFIX.     FT067920
         REM                                                            FT067930
*        THE ARGUMENTS AND OTHER INFORMATION FOR EACH FUNCTION ARE KEPT FT067940
*        IN A STACK IN POOL.  THE STACK IS INITIALLY FORMED AS FOLLOWS... FT067950
*        THE FIRST WORD CONTAINS THE EQUIV ENTRY WITH THE ADDRESS ALTERED FT067960
*        TO CONTAIN THE DEFINITION OF THE FUNCTION (CURRENT VALUE OF     FT067970
*        PCOUNT).  THE SECOND WORD HAS THE COLUMN 1 MODE IN ITS ADDRESS, FT067980
*        AND THE POOL POINTER TO THE END OF THIS STACK IN THE DECREMENT. FT067990
*        SUBSEQUENT WORDS CONTAIN THE BCD NAMES OF THE DUMMY ARGUMENTS.  FT068000
*        IF AN ARGUMENT IS REAL AND THE COLUMN 1 MODE IS 'D' OR 'I', THEN FT068010
*        ONE EXTRA CELL IS SKIPPED TO AID IN LATER FORMING CALLING       FT068020
*        SEQUENCES TO THIS FUNCTION.                                     FT068030
         REM                                                            FT068040
*        THE FLAG CELL ARITFN CONTAINS THE LOCATION OF THE STACK        FT068050
*        IN ITS ADDRESS.                                                FT068060
         REM                                                            FT068070
*        AS THE STATEMENT IS SCANNED, THE LIST OF DUMMY ARGUMENTS IS    FT068080
*        SEARCHED WHENEVER A VARIABLE NAME IS ENCOUNTERED.              FT068090
*        WHEN A DUMMY NAME IS LOCATED, THE ASTACK ENTRY IS SET TO       FT068100
*        CONTAIN A BARIT BIT (BIT 15).                                  FT068110
*        THIS WILL CAUSE THE CODE ROUTINE TO PICK UP SUCH ARGUMENTS     FT068120
*        FROM THE CALLING SEQUENCE INDIRECTLY (WHENEVER POSSIBLE).      FT068130
         REM                                                            FT068140
*        AFTER THE STATEMENT IS PROCESSED, THE DUMMY ARGUMENT NAMES IN  FT068150
*        THE STACK ARE REPLACED WITH EQUIV-LIKE MODE DESIGNATIONS.      FT068160
         REM                                                            FT068170
SCANAA   CLA     SCAN5B      GET THE RETURNED WORD FROM LOCATE ROUTINE  FT068180
         TMI     *+2         ERROR IF ALREADY IN SYMTAB                 FT068190
         ERROR   78,SKEND                                               FT068200
         SIL     BINTF       SET ARITHMETIC FUNCTION BIT ON             FT068210
         STI     SCAN5B      RESTORE THE SAVE CELL                      FT068220
         CAL     NXTLOC      GET POOL POINTER                           FT068230
         STA     SCAN5B      SET SAVE CELL ADDRESS TO START OF STACK    FT068240
         CAL     SCAN5B                                                 FT068250
         SLW     EQUIV,4     RESTORE THE EQUIV WORD                     FT068260
         REM                                                            FT068270
         LDI     PROCSW      GET THE PROCESSOR SWITCH                   FT068280
         RIL     BXEQ        TURN OFF THE EXECUTABLE STATEMENT BIT      FT068290
         STI     PROCSW      RESTORE SWITCH                             FT068300
         REM                                                            FT068310
         LXD     FIRSTF,1    DETERMINE THAT NO EXECUTABLE STATEMENT     FT068320
         TXL     *+2,1,0     HAS ALREADY OCCURRED.                      FT068330
         ERROR   89,*+1                                                 FT068340
         REM                                                            FT068350
         LXA     NXTLOC,1    GET POINTER TO START OF STACK              FT068360
         SXA     ARITFN,1    SAVE POINTER AND MARK THIS AS AN ARITH FN. FT068370
         SLW     POOL,1      SET DESCRIPTOR WORD IN THE STACK           FT068380
         CAL     PCOUNT      DEFINE THE FUNCTION, AND SAVE THE          FT068390
         STA     POOL,1      DEFINITION IN THE FIRST STACK ADDRESS      FT068400
         CAL     MODFLG      SAVE THE MODE IN THE SECOND STACK ADDRESS  FT068410
         STA     POOL-1,1                                               FT068420
         TXI     *+1,1,2     BUMP POINTER                               FT068430
         TXI     *+1,2,-1    SKIP THE LEFT PAREN (ALREADY CHECKED FOR)  FT068440
         REM                                                            FT068450
SCANAC   TSX     SCNBCD,4    GET AN ARGUMENT                            FT068460
```

```
        TMI     ERROR8          MUST BE SYMBOLIC                        FT068470
        SLW     POOL,1          SAVE IN THE STACK                       FT068480
        REM                                                             FT068490
        NZT     DBLMOD          IS THIS A DOUBLE-LENGTH STATEMENT.      FT068500
        TRA     SCANAJ          NO                                      FT068510
        LAS     =HI00000        YES.  IF THIS IS A REAL VARIABLE,       FT068520
        TRA     *+1             THEN SKIP A CELL IN THE STACK           FT068530
        LAS     =HN(((((        SO THAT THE CALLING SEQUENCE WILL       FT068540
        TXI     *+2,1,1         COME OUT RIGHT.                         FT068550
        NOP                                                             FT068560
        REM                                                             FT068570
SCANAJ  SKIP1   CMCHAR          IS IT A COMMA.                          FT068580
        TXI     SCANAC,1,1      YES.  GO GET ANOTHER ARGUMENT           FT068590
        CAL     COLUMN,2        GET NEXT TWO CHARACTERS, AND            FT068600
        ERA     =H)=            DETERMINE THAT THEY ARE  )=             FT068610
        ARS     24                                                      FT068620
        TNZ     ERROR7                                                  FT068630
        REM                                                             FT068640
        LXA     SCAN5B,4        GET THE START OF STACK                  FT068650
        SXD     SCANAF,1        SET TEST INSTRUCTION DECREMENT          FT068660
        PXD     ,1              TERMINATION OF STACK                    FT068670
        STD     POOL-1,4        SAVE END-OF-STACK IN 2ND WORD DECREMENT FT068680
        TXI     *+1,1,1         BUMP POOL POINTER                       FT068690
        SXA     NXTLOC,1        ALL DONE.  RESET THE POOL POINTER       FT068700
        REM                                                             FT068710
        LDI     POOL,4          GET THE DEFINITION WORD FROM STACK      FT068720
        RIL     BINTF           RESET THE INTERNAL FUNCTION BIT         FT068730
        STI     ASTACK          PUT INTO THT FIRST ASTACK WORD          FT068740
        CAL     ASFKEY          GET OPERATOR KEY FOR ARITH. ST. FUNCTIONS FT068750
        SLW     ASTACK-1        PUT INTO SECOND ASTACK WORD             FT068760
        AXT     1,1             SET ASTACK POINTER TO PROPER VALUE      FT068770
        REM                                                             FT068780
        TSX     CITBLE,4        TELL CITBLD OF START OF ARITH. STATE. FN. FT068790
        TXI     GETNXT,2,-2     GO TO COMPIL ROUTINE, AND SKIP THE )=   FT068800
```

```
                EJECT                                                                FT068810
*               NUMBER PROCESSOR...                                                  FT068820
*               THIS ROUTINE IS ENTERED WHEN THE INITIAL CHARACTER SCANNED           FT068830
*               IS A DIGIT.  IF THE COLUMN 1 MODE IS 'B', THE OCTAL CONSTANT SCANFT068840
*               SECTION IS ENTERED.  OTHERWISE, ANY INTEGER PART OF THE NUMBER       FT068850
*               IS SCANNED OFF AND COMPUTED.  WHEN A NON-DIGIT CHARACTER IS          FT068860
*               ENCOUNTERED, THIS INTEGER SECTION IS EXITED.  IF THE CHARACTER       FT068870
*               IS NOT A DECIMAL POINT OR AN H, THE INTEGER IS STASHED IN CONTAB     FT068880
*               AS A FIXED POINT DECREMENT INTEGER.                                  FT068890
                REM                                                                  FT068900
*               IF THE CHARACTER IS AN H, THE HOLLERITH LITERAL SCANNER IS           FT068910
*               ENTERED, WITH THE COMPUTED INTEGER BEING THE CHARACTER COUNT.        FT068920
                REM                                                                  FT068930
*               IF THE CHARACTER AFTER THE INTEGER IS A DECIMAL POINT, THE           FT068940
*               FLOATING POINT SCANNER IS ENTERED.  THIS SECTION IS ALSO             FT068950
*               ENTERED WHEN THE ORIGINAL SCAN IDENTIFIES A DECIMAL POINT.           FT068960
*               THE FRACTIONAL PART (IF ANY) OF THE NUMBER IS SCANNED OFF.           FT068970
*               THE BINARY EQUIVALENT OF EACH DIGIT OF THE FRACTION IS COMPUTED      FT068980
*               BY MULTIPLYING IT BY THE APPROPRIATE FULL WORD FIXED POINT           FT068990
*               POWER OF 10 (IN DOUBLE PRECISION), AND ADDING THE RESULT TO          FT069000
*               THE PREVIOUS FRACTION COMPUTATION.  IF THE INTEGER PART IS           FT069010
*               ZERO, ANY LEADING FRACTIONAL ZEROES ARE COUNTED, AND THE             FT069020
*               FRACTION COMPUTATION BEGINS WITH THE FIRST NON-ZERO DIGIT,           FT069030
*               AS IF IT WERE THE DIGIT REPRESENTING 10**-1.                         FT069040
*               IF AN E IS ENCOUNTERED, THE EXPONENT IS SCANNED OFF.  THIS           FT069050
*               EXPONENT IS COMBINED WITH THAT ONE GENERATED BECAUSE OF ANY          FT069060
*               LEADING FRACTIONAL ZEROES.  WHEN THE END OF THE CONSTANT IS          FT069070
*               ENCOUNTERED, THE INTEGER AND FRACTION ARE COMBINED AND               FT069080
*               NORMALIZED (FIRST NON-ZERO BIT IN POSITION 9).  THE RESULTING        FT069090
*               DOUBLE PRECISION CONSTANT IS MULTIPLIED (DOUBLE PRECISION) BY        FT069100
*               THE APPROPRIATE POWER OF 10 AS SPECIFIED BY THE PREVIOUSLY           FT069110
*               COMPUTED EXPONENT.  (A FLOATING POINT TRAP MAY OCCUR HERE...         FT069120
*               ANY REGISTER INVOLVED IS SET TO ZERO.)                               FT069130
*               IF THIS IS A SINGLE PRECISION OR COMPLEX STATEMENT, THE HIGH-        FT069140
*               ORDER PART OF THE CONSTANT IS ROUNDED AND STASHED IN CONTAB.         FT069150
*               IF DOUBLE PRECISION, BOTH PARTS OF THE CONSTANT ARE STASHED.         FT069160
                SPACE   2                                                            FT069170
SCAN1           CAL     MODFLG          GET THE COLUMN 1 FLAG                        FT069180
                ERA     =1              IS IT A BOOLEAN STATEMENT.                   FT069190
                TZE     SCNOCT          YES.  NUMBER MUST BE OCTAL                   FT069200
                REM                                                                  FT069210
SCAN1T          CLA     COLUMN,2                                                     FT069220
                ARS     30                                                           FT069230
                STO     INTG                                                         FT069240
                REM                                                                  FT069250
SCAN1X          STZ     EXPONA                                                       FT069260
                STZ     DECPT           RESET DOUBLE PRECISION FLAG                  FT069270
                STZ     FRAC              OUT                                        FT069280
                STZ     FRAC+1                                                       FT069290
                STZ     EXPON                                                        FT069300
                REM                                                                  FT069310
SCAN1A          TXI     *+1,2,-1          GET NEXT CHARACTER                         FT069320
                CAL     COLUMN,2                                                     FT069330
                ARS     30                                                           FT069340
                SLW     CHAR                                                         FT069350
                LAS     =10                                                          FT069360
                TRA     SCAN1B           *GEEATER THAN 10                            FT069370
                MACER                   MACHINE ERROR.  OCTAL 12 FROM DECIMAL TAPE   FT069380
                REM                                                                  FT069390
                CLA     INTG            ANOTHER DIGIT                                FT069400
```

```
         ALS     2           MULTIPLY OLD DIGIT(S) BY 10 AND              FT069410
         ADD     INTG        ADD ON THE NEW DIGIT                         FT069420
         ALS     1                                                        FT069430
         ADD     CHAR                                                     FT069440
         STO     INTG                                                     FT069450
         TRA     SCAN1A       *GO BACK FOR MORE                           FT069460
         REM                                                              FT069470
SCAN1B   LAS     =H00000.      WHAT IS NEW CHARACTER                      FT069480
         TRA     *+2                                                      FT069490
         TRA     SCAN1C       *DECIMAL POINT                              FT069500
         LAS     =H00000H    IS THIS A POSSIBLE HOLLERITH FIELD.          FT069510
         TRA     *+2                                                      FT069520
         TXI     SCANH,2,-1  YES.  ADJUST COLUMN POINTER AND GO SCAN LIT  FT069530
         REM                                                              FT069540
         REM                                                              FT069550
SCN1A    CAL     INTG        ENTER INTEGER INTO CONSTANT TABLE            FT069560
         ALS     18                                                       FT069570
SCAN1Z   TSX     ASCON1,4,INTEGR    PUT CONSTANT INTO CONTAB              FT069580
         STA     SCAN14      BUILD KEY WORD.  SAVE CONTAB POINTER         FT069590
         CAL     ,4          GET MODE BITS                                FT069600
         ANA     =07000000   SAVE ONLY THE MODE BITS                      FT069610
         ALS     15          TO LEFT END OF THE WORD                      FT069620
         ORA     =1B17       ADD THE CONSTANT FLAG BIT                    FT069630
         ORA     SCAN14      ADD CONTAB POINTER                           FT069640
         TRA     SCANX       RETURN AND EXIT FROM SCANNER                 FT069650
         SPACE   6                                                        FT069660
         REM                   FLOATING NUMBER                            FT069670
SCAN1C   AXT     21,1          SET K = .1 DOUBLE PRECISION                FT069680
         STL     DECPT       DECIMAL POINT ENCOUNTERED                    FT069690
         TXI     *+1,2,-1                                                 FT069700
         CAL     COLUMN,2                                                 FT069710
         ARS     30                                                       FT069720
         TNZ     SCAN1M                                                   FT069730
         TXL     SCAN1M,1,20                                              FT069740
         ZET     INTG                                                     FT069750
         TRA     SCAN1M                                                   FT069760
         CLA     EXPONA                                                   FT069770
         SUB     =1                                                       FT069780
         STO     EXPONA                                                   FT069790
         TXI     SCAN1C+3,2,-1                                            FT069800
SCAN1M   CAS     =10                                                      FT069810
         TRA     SCAN1E          *NON-NUMERIC                             FT069820
         MACER                MACHINE ERROR.  OCTAL 12 FROM DECIMAL TAPE  FT069830
         REM                                                              FT069840
SCAN1L   LDQ     =0                                                       FT069850
         LRS     4                                                        FT069860
         STQ     SYMBOL+3                                                 FT069870
         MPY     F10M,1                                                   FT069880
         LLS     4                                                        FT069890
         ADD     FRAC                                                     FT069900
         STO     FRAC                                                     FT069910
         TOV     *               TURN OFF OVERFLOW INDICATOR              FT069920
         XCA                                                              FT069930
         ADD     FRAC+1                                                   FT069940
         STO     FRAC+1                                                   FT069950
         TNO     *+2                                                      FT069960
         TSX     AF,4                                                     FT069970
         REM                                                              FT069980
         LDQ     SYMBOL+3                                                 FT069990
         MPY     F10L,1                                                   FT070000
```

```
         LRS     31                                                    FT070010
         ADD     FRAC                                                  FT070020
         STO     FRAC                                                  FT070030
         XCA                                                           FT070040
         TOV     *                                                     FT070050
         ADD     FRAC+1                                                FT070060
         STO     FRAC+1                                                FT070070
         TNO     *+2                                                   FT070080
         TSX     AF,4                                                  FT070090
         TIX     *+1,1,1          STEP POWER                           FT070100
         TRA     SCAN1C+2       GO BACK FOR MORE                       FT070110
         REM                                                           FT070120
AF       CLA     FRAC                                                  FT070130
         ADD     =1                                                    FT070140
         STO     FRAC                                                  FT070150
         TNO     1,4              MUST BE OFF                          FT070160
         MACER                  OVERFLOW LOGIC ERROR                   FT070170
         REM                                                           FT070180
SCAN1E   ERA     =H000000E      IS IT E                                FT070190
         TZE     SCAN1D          *YES                                  FT070200
SCAN1J   CAL     INTG                                                  FT070210
         ORA     FRAC             IS FLOATING POINT WORD ZERO          FT070220
         ORA     FRAC+1                                                FT070230
         LDQ     =0             ERASE THE GARBAGE IN THE MQ            FT070240
         TZE     SCAN1Y         YES                                    FT070250
         REM                                                           FT070260
         CLA     =0233000000000  SET INITIAL EXPONENT                  FT070270
SCAN1R   STO     SYMBOL         NORMALIZE THE FLOATING POINT NUMBER    FT070280
         LDI     INTG                                                  FT070290
         LFT     777000                                                FT070300
         TRA     SCAN1P          *GO TO SHIFT RIGHT                    FT070310
         LFT     400                                                   FT070320
         TRA     SCAN1Q          *IT IS NORMALIZED                     FT070330
         REM          SHIFT LEFT                                       FT070340
         CLA     INTG                                                  FT070350
         LDQ     FRAC                                                  FT070360
         LLS     1                                                     FT070370
         STO     INTG                                                  FT070380
         LLS     34                                                    FT070390
         LDQ     FRAC+1                                                FT070400
         LLS     1                                                     FT070410
         STO     FRAC                                                  FT070420
         STQ     FRAC+1                                                FT070430
         CLA     SYMBOL                                                FT070440
         SUB     =1B8                                                  FT070450
         TRA     SCAN1R                                                FT070460
         REM          SHIFT RIGHT                                      FT070470
SCAN1P   CLA     FRAC                                                  FT070480
         ADD     =1                                                    FT070490
         LRS     35                                                    FT070500
         ADD     INTG                                                  FT070510
         LRS     1                                                     FT070520
         STO     INTG                                                  FT070530
         STQ     FRAC                                                  FT070540
         CLA     SYMBOL                                                FT070550
         ADD     =1B8                                                  FT070560
         TRA     SCAN1R                                                FT070570
         REM          NUMBER IS NORMALIZED                             FT070580
SCAN1Q   CAL     FRAC              APPEND MACHINE EXPONENT             FT070590
         ARS     8                                                     FT070600
```

```
        ORA     SYMBOL                                                  FT070610
        SUB     =27B8                                                   FT070620
        XCA                                                             FT070630
        CLA     INTG                                                    FT070640
        ORA     SYMBOL                                                  FT070650
        STO     FRAC                                                    FT070660
        STQ     FRAC+1                                                  FT070670
        CLA     EXPON                                                   FT070680
        ADD     EXPONA                                                  FT070690
        LAS     =38                                                     FT070700
        TRA     SCAN1F          *EXCESSIVE EXPONENT SIZE                FT070710
        NOP                                                             FT070720
        ADD     =2B2                                                    FT070730
        PAX     ,4                                                      FT070740
        LDQ     FRAC                                                    FT070750
        FMP     P10M,4                                                  FT070760
        STO     SYMBOL+1                                                FT070770
        STQ     SYMBOL+2                                                FT070780
        LDQ     P10M,4                                                  FT070790
        FMP     FRAC+1                                                  FT070800
        FRN                                                             FT070810
        UFA     SYMBOL+2                                                FT070820
        FRN                                                             FT070830
        STO     SYMBOL+2                                                FT070840
        LDQ     FRAC                                                    FT070850
        FMP     P10L,4                                                  FT070860
        FRN                                                             FT070870
        UFA     SYMBOL+2                                                FT070880
        FRN                                                             FT070890
        FAD     SYMBOL+1                                                FT070900
        REM                                                             FT070910
SCAN1Y  LDI     MODFLG                                                  FT070920
        RNT     2               IS THIS A DOUBLE-LENGTH STATEMENT.      FT070930
        TRA     SCAN1N          NO.  SINGLE PRECISION                   FT070940
        RFT     1               DP OR CA.  IS THIS COMPLEX STATEMENT.   FT070950
        TRA     SCAN1S          YES.                                    FT070960
        TSX     ASCON2,4,REAL   PUT DOUB. PREC. CONSTANT INTO TABLE     FT070970
        TRA     SCAN1Z+1        PREPARE THE KEY WORD                    FT070980
        REM                                                             FT070990
SCAN1N  FRN                     ROUND THE SINGLE-PRECISION CONSTANT     FT071000
        TSX     ASCON1,4,REAL       REAL CONSTANT                       FT071010
        TRA     SCAN1Z+1        PREPARE THE KEY WORD                    FT071020
        REM                                                             FT071030
SCAN1S  FRN                     ROUND THE PART OF COMPLEX CONSTANT      FT071040
        SLW     CX2             SAVE VALUE OF CONSTANT                  FT071050
        NZT     CXFLAG          HAS ONE PART OF CONSTANT ALREADY BEEN GOT. FT071060
        SLW     CX1             NO.  SAVE IN FIRST-PART CELL            FT071070
        TSX     SCAN1Z+1,4,REAL     BUILD ASTACK WORD.  NO CONSTANT STASH FT071080
        REM                                                             FT071090
SCAN1D  TXI     *+1,2,-1          E ENCOUNTERED. GET NEXT CHARACTER     FT071100
        AXT     ERR071,4        GET PROPER TRANSFER ADDRESS             FT071110
        CAL     COLUMN,2                                                FT071120
        ARS     30                WHAT IS TI                            FT071130
        LAS     =H00000+                                                FT071140
        TRA     *+2                                                     FT071150
        TRA     SCAN1G            *+                                    FT071160
        LAS     =H00000-                                                FT071170
        TRA     *+2                                                     FT071180
        TRA     SCAN1H            -                                     FT071190
        REM                                                             FT071200
```

```
SCAN1K  SLW     CHAR                                                    FT071210
        SXA     *+2,4           SET TRANSFER ADDRESS                    FT071220
        LAS     =10                                                     FT071230
        TRA     **              ADDRESS SET PREVIOUSLY                  FT071240
        MACER                   MACHINE ERROR.  OCTAL CHARACTER 12      FT071250
        CLA     EXPON                                                   FT071260
        ALS     2                                                       FT071270
        ADD     EXPON                                                   FT071280
        ALS     1                                                       FT071290
        ACL     CHAR                                                    FT071300
        STO     EXPON                                                   FT071310
        AXT     SCAN1J,4        GET PROPER TRANSFER ADDRESS             FT071320
SCAN1G  TXI     *+1,2,-1                                                FT071330
        CAL     COLUMN,2                                                FT071340
        ARS     30                                                      FT071350
        TRA     SCAN1K                                                  FT071360
        REM                                                             FT071370
SCAN1H  CLS     =0                - AFTER E                             FT071380
        STO     EXPON                                                   FT071390
        TRA     SCAN1G                                                  FT071400
        REM                                                             FT071410
SCAN1F  ERROR   72,*+1                                                  FT071420
        LGR     74              ZERO EVERYTHING                         FT071430
        TRA     SCAN1Y          PUT OUT A ZERO CONSTANT                 FT071440
        REM                                                             FT071450
        REM                     FLOATING POINT TRAP ROUTINE             FT071460
(FPT).  TRA     *+1             CELL 8 TRA INTO FPT ROUTINE             FT071470
        LDI     0               GET THE TRAP WORD                       FT071480
        LFT     4               IS IT OVERFLOW.                         FT071490
        ERROR   72,EXFPT-2      YES.  PRINT MESSAGE AND CONTINUE        FT071500
        LFT     2               IS IT AC AND MQ.                        FT071510
        ZAC                     YES.  ZERO THE ACCUM                    FT071520
        LDQ     =0              ZERO THE MQ                             FT071530
EXFPT   TRA*    0               RETURN                                  FT071540
        SPACE   5                                                       FT071550
*               OCTAL CONSTANT SCANNER                                  FT071560
        REM                                                             FT071570
SCNOCT  STZ     INTG            ZERO THE CONSTANT HOLDER                FT071580
        SXA     SCNOC2,2        SAVE THE COLUMN POINTER                 FT071590
        AXT     13,4                                                    FT071600
        CAL     COLUMN,2                                                FT071610
        TXI     *+1,2,-1                                                FT071620
        LGR     33                                                      FT071630
        TNZ     SCNOC1          NOT AN OCTAL CHARACTER                  FT071640
        REM                                                             FT071650
        CAL     INTG                                                    FT071660
        LGL     3               MOVE DIGIT INTO NUMBER                  FT071670
        SLW     INTG            RESTORE THE PARTIAL INTEGER             FT071680
        TIX     SCNOCT+2,4,1                                            FT071690
        ERROR   72,SKEND        OCTAL NUMBER TOO BIG FOR COMPUTER       FT071700
        REM                                                             FT071710
SCNOC1  LGL     3               RESTORE THE WHOLE CHARACTER             FT071720
        LAS     =H000000H       IS IT AN H.                             FT071730
        TRA     *+2             NO                                      FT071740
        TRA     SCNOC2          YES.  HOLLERITH LITERAL                 FT071750
        LAS     =060            IS IT A BLANK.                          FT071760
        TRA     *+3             NO.  PREPARE TO EXIT                    FT071770
        TRA     SCNOCT+2        YES.  SKIP THE BLANK                    FT071780
        LAS     =012            IS THIS AN ILLEGAL OCTAL DIGIT.         FT071790
        CAL     INTG            NO.  PREPARE TO EXIT                    FT071800
```

```
        TXI     SCAN1N+1,2,1    PUT IN CONTAB AS A 'REAL' NUMBER        FT071810
        ERROR   4,SKEND         ILLEGAL OCTAL DIGIT.  ERROR             FT071820
        REM                                                             FT071830
SCNOC2  AXT     **,2            HOLLERITH LITERAL.  RESTORE THE COLUMN  FT071840
        TRA     SCAN1T          POINTER AND RETURN FOR LITERAL SCAN     FT071850
        REM                                                             FT071860
```

```
        EJECT                                                                   FT071870
*       HOLLERITH LITERAL SCANNER.                                              FT071880
*       THIS SCANNER IS ENTERED WHEN A FIXED POINT INTEGER IS FOLLOWED          FT071890
*       BY THE CHARACTER H.  SINCE THE FORM OF THE STATEMENT AS IT              FT071900
*       APPEARS IN 'COLUMN' CONTAINS NO BLANKS, AND SINCE THE LITERAL           FT071910
*       MAY HAVE BLANK CHARACTERS IN IT, IT IS NECESSARY TO SCAN THE            FT071920
*       ORIGINAL STATEMENT (IN 'CARD') TO OBTAIN THE LITERAL.                   FT071930
*       THE COUNT OF NON-BLANK CHARACTERS THUS FAR ENCOUNTERED                  FT071940
*       (DOWN THROUGH THE H) IS THE COLUMN POINTER (XR 2).  THIS SAME           FT071950
*       NUMBER OF NON-BLANK CHARACTERS IS SCANNED OFF THE CARD ARRAY.           FT071960
*       THE CORRECT NUMBER OF HOLLERITH CHARACTERS IS THEN SCANNED              FT071970
*       FROM CARD, AND PACKED TO THE LEFT TO FORM CONSTANT WORD(S).             FT071980
*       IF THIS IS NOT IN A SUBPROGRAM CALLING SEQUENCE, ONLY ONE               FT071990
*       WORD IS STASHED IN THE CONSTANT TABLE.  IF THIS IS IN A CALLING         FT072000
*       SEQUENCE, ALL THE SCANNED CHARACTERS ARE STASHED (6 TO A WORD) INFT072010
*       CONTAB.  IN THIS LATTER CASE, A WORD OF ALL SEVENS (OCTAL)              FT072020
*       IS ADDED TO THE END OF THE LITERAL.                                     FT072030
        REM                                                                     FT072040
*       INFORMATION IS SAVED TO ENABLE A RESTART OF THE SCAN FROM               FT072050
*       THIS LITERAL, IN CASE ANOTHER LITERAL IS ENCOUNTERED IN THIS            FT072060
*       SAME STATEMENT.                                                         FT072070
        REM                                                                     FT072080
*       NOTE THAT LITERALS IN DOUBLE PRECISION AND COMPLEX STATEMENTS           FT072090
*       ARE USUALLY MEANINGLESS, BUT ALLOWED NEVERTHELESS.  ALSO,               FT072100
*       WHILE LITERALS IN BOOLEAN STATEMENTS WILL NOT WORK CORRECTLY            FT072110
*       IN IBM'S FORTRAN COMPILER, THEY WILL FUNCTION CORRECTLY IN              FT072120
*       FASTRAN.                                                                FT072130
        SPACE   2                                                               FT072140
SCANH   SXD     SCANH4,2                                                        FT072150
        ZET     HFLAG           IS THIS THE FIRST LITERAL OF STATEMENT.         FT072160
        TRA     SCANH2          NO.  PICK UP THE SCAN WHERE WE LEFT OFF         FT072170
        REM                                                                     FT072180
        STL     HFLAG           YES.  FIRST TIME IN STATEMENT                   FT072190
        AXT     0,1             SET NON-BLANK CHARACTER COUNT TO ZERO           FT072200
        AXC     1,2             SET FOR FIRST WORD OF STATEMENT                 FT072210
        AXT     6,4             SET BYTE COUNT TO 6                             FT072220
        LDQ     CARD,2          GET STATEMENT WORD                              FT072230
        REM                                                                     FT072240
SCANH0  ZAC                                                                     FT072250
        LGL     6                                                               FT072260
        ERA     =H00000         IS IT A NON-BLANK CHARACTER.                    FT072270
        TZE     SCANH3          NO                                              FT072280
        REM                                                                     FT072290
        TXI     *+1,1,-1        YES.  UP THE NON-BLANK CHARACTER COUNT          FT072300
SCANH4  TXL     SCANH1,1,**     HAVE CORRECT NUMBER OF NON-BLANK                FT072310
        REM                     CHARACTERS BEEN FOUND YET.                      FT072320
SCANH3  TIX     SCANH0,4,1      NO.  IS THIS WORD USED UP.                      FT072330
        TXI     SCANH0-2,2,-1         YES.  GET NEXT WORD ON CARD               FT072340
        REM                                                                     FT072350
SCANH1  SAVE    SCANH2          SAVE MQ AND XRS FOR POSSIBLE RESTART            FT072360
        STQ     SCANHT                                                          FT072370
        REM                                                                     FT072380
        REM                     NOW SCAN OFF THE LITERAL                        FT072390
        AXT     COLUMN,1        PUT 'COLUMN' INTO INSTRUCTION ADDRESS           FT072400
        SXA     SCANHA,1                                                        FT072410
        LXA     INTG,1          GET NUMBER OF HOLLERITH CHARACTERS NEEDED       FT072420
        REM                                                                     FT072430
SCANHC  CAL     CARD,2          GET CURRENT WORD FROM CARD                      FT072440
        LDQ     CARD+1,2        ALSO GET NEXT WORD                              FT072450
        XEC     SCANHB,4        LEFT-JUSTIFY THE LITERAL                        FT072460
```

```
         TXH     *+3,1,5         SKIP IF LITERAL FILLS WORD (OR MORE)          FT072470
         REM                                                                   FT072480
         ANA     SCANT1,1        SAVE ONLY PERTINENT CHARACTERS                FT072490
         ORA     SCANT2,1        ADD TERMINAL BLANKS                           FT072500
         REM                                                                   FT072510
SCANHA   SLW     **              SAVE WORDS IN THE COLUMN REGION               FT072520
         CAL     *-1             BUMP ADDRESS OF THE STORE WORD                FT072530
         ACL     =1                                                            FT072540
         STA     *-3             RESTORE THE ADDRESS                           FT072550
         REM                                                                   FT072560
         TXI     *+1,2,-1        BUMP 'CARD' WORD COUNTER BY 1                 FT072570
         TIX     SCANHC,1,6      RETURN IF MORE LITERAL CHARACTERS NEEDED      FT072580
         REM                                                                   FT072590
         LAC     SCANHA,4        GET COUNTER FOR NUMBER OF LITERAL WORDS       FT072600
         TXI     *+1,4,COLUMN        GET RID OF ALL BUT THE WORD COUNT+1       FT072610
         LXD     SCANH4,2        RELOAD THE COLUMN POINTER                     FT072620
         AXT     6,1             INITIALIZE TO SIX BYTES PER WORD              FT072630
         LDQ     COLUMN-1,4      GET A WORD FROM THE STACK                     FT072640
         REM                                                                   FT072650
SCANHF   ZAC                     COUNT NUMBER OF NON-BLANK CHARACTERS IN LITFT072660
         LGL     6               SHIFT IN A CHARACTER                          FT072670
         ERA     =H00000                                                       FT072680
         TZE     *+2             SKIP IF A BLANK                               FT072690
         TXI     *+1,2,-1        ADJUST THE COLUMN POINTER                     FT072700
         TIX     SCANHF,1,1      RETURN IF NOT DONE WITH THIS WORD             FT072710
         TXI     *+1,4,1         BUMP THE WORD COUNT                           FT072720
         TXH     SCANHF-2,4,0    RETURN FOR ANOTHER WORD IF NECESSARY          FT072730
         REM                                                                   FT072740
         CAL     COLUMN          NON-BLANK COUNT DONE.  GET WORD OF LITERAL    FT072750
         NZT     CALLFG          CAN THIS BE IN A SUBPROGRAM ARGUMENT LIST.    FT072760
         TRA     SCAN1N+1        NO.  GO STASH THE LITERAL                     FT072770
         REM                                                                   FT072780
         LXA     INTG,1          YES, POSSIBLY.  GET NUMBER OF CHARACTERS      FT072790
         TXH     SCANHD,1,6      IF MORE THAN ONE WORD, GO USE THE             FT072800
         REM                     BRUTE-FORCE STASHING ROUTINE.                 FT072810
         LDQ     ALLSVN          TERMINAL WORD FOR SUBROUTINE ARGUMENTS        FT072820
         XCL                     EXCHANGE TO EFFECT CORRECT STASHING           FT072830
         TSX     ASCON2,4,REAL       GO STASH CONSTANT                         FT072840
         SUB     =1              POINTER POINTS TO WRONG WORD OF THE PAIR      FT072850
         TRA     SCAN1Z+1        PREPARE ASTACK WORD                           FT072860
         REM                                                                   FT072870
         REM                     LONG LITERAL.  BRUTE-FORCE STASHING USED      FT072880
SCANHD   LXA     NOCONT,4        GET PRESENT NUMBER OF CONSTANTS IN TABLE      FT072890
         SXA     SCANHG,2        SAVE THE COLUMN POINTER                       FT072900
         AXT     0,2                                                           FT072910
SCANHV   CAL     COLUMN,2        GET A WORD OF THE LITERAL                     FT072920
         SLW     CONTAB,4        PUT INTO THE CONSTANT TABLE                   FT072930
         TXI     *+1,4,1         BUMP TABLE POINTER                            FT072940
         TXH     SCANHH,4,NOCNST     RESET TABLE IF FULL, AND PUT OUT MSG      FT072950
         TXI     *+1,2,-1        BUMP THE LITERAL WORD COUNTER                 FT072960
         TIX     SCANHV,1,6      STASH ANOTHER WORD IF MORE TO GO              FT072970
         REM                                                                   FT072980
         CAL     ALLSVN          TERMINAL WORD FOR SUBROUTINE CALLS            FT072990
         SLW     CONTAB,4        STASH IT                                      FT073000
         TXI     *+1,4,1                                                       FT073010
         TXH     SCANHH,4,NOCNST     RESET TABLE IF FULL, AND PUT OUT MSG      FT073020
         CAL     NOCONT          ADDRESS OF ACCUM GETS THE START OF STRING     FT073030
         SXA     NOCONT,4        RESET NOCONT TO ITS NEW VALUE                 FT073040
SCANHG   AXT     **,2            RESTORE THE COLUMN POINTER                    FT073050
         TSX     SCAN1Z+1,4,REAL     GO PREPARE THE ASTACK WORD                FT073060
```

```
         REM                                                        FT073070
SCANHH   AXT     0,4             RESET TABLE IF FULL               FT073080
         STR     MSG100,,SCANHG-1        ERROR MESSAGE             FT073090
         SPACE   2                                                  FT073100
         REM                     SHIFT TABLE                        FT073110
         LGL     6                                                  FT073120
         LGL     12                                                 FT073130
         LGL     18                                                 FT073140
         LGL     24                                                 FT073150
         LGL     30                                                 FT073160
         LGL     36                                                 FT073170
SCANHB   EQU     *                                                  FT073180
         REM                                                        FT073190
SCANH2   RESTOR          ENTER HERE FOR ALL BUT FIRST TIME IN STATE.FT073200
         LDQ     SCANHT                                             FT073210
         TRA     SCANH3                                             FT073220
```

```
        EJECT                                                                FT073230
*       OPERATOR SCANNERS                                                    FT073240
        SPACE   3                                                            FT073250
        REM                     = SIGN PROCESSING                            FT073260
SCAN13  CAL     EQLKEY                                                       FT073270
SCAN3A  TXI     *+1,2,-1                                                     FT073280
        LXA     SCANX,1                                                      FT073290
        TRA     3,4                                                          FT073300
        SPACE   3                                                            FT073310
        REM                     + PROCESSING                                 FT073320
SCAN20  CAL     PLUSKY                                                       FT073330
        TRA     SCAN3A                                                       FT073340
        SPACE   3                                                            FT073350
        REM                     . PROCESSING                                 FT073360
SCAN33  LDQ     COLUMN+1,2      GET NEXT CHARACTER                           FT073370
        CRQ     SCANT,1,1                                                    FT073380
        TXL     ERRO79,1,SCAN1-1        NON-NUMERIC ERROR                    FT073390
        TXH     ERRO79,1,SCAN1                                               FT073400
        STZ     INTG                                                         FT073410
        CAL     MODFLG                                                       FT073420
        ERA     =1                                                           FT073430
        TZE     ERRO79                                                       FT073440
        TXI     SCAN1X,2,1      ADJUST COLUMN POINTER SO THAT THE            FT073450
        REM                     NUMBER SCANNER WILL CATCH THE DECIMAL POINT  FT073460
        SPACE   3                                                            FT073470
        REM                       ) PROCESSING                               FT073480
SCAN34  CAL     RPNKEY                                                       FT073490
        TRA     SCAN3A                                                       FT073500
        SPACE   3                                                            FT073510
        REM                     - PROCESSING                                 FT073520
SCAN40  CAL     UNEGKY                                                       FT073530
        TRA     SCAN3A                                                       FT073540
        SPACE   3                                                            FT073550
        REM                     * PROCESSING                                 FT073560
SCAN54  CAL     COLUMN+1,2      * SIGN                                       FT073570
        ERA     =H*                                                          FT073580
        ARS     30                                                           FT073590
        TZE     *+3                                                          FT073600
        CAL     MPYKEY          * ONLY                                       FT073610
        TRA     SCAN3A                                                       FT073620
        REM                                                                  FT073630
        CAL     EXPKEY          **                                           FT073640
        TXI     SCAN3A,2,-1                                                  FT073650
        SPACE   3                                                            FT073660
        REM                     BLANK PROCESSING                             FT073670
SCAN60  TXI     SCANA,2,-1                                                   FT073680
        SPACE   3                                                            FT073690
        REM                     / PROCESSING                                 FT073700
SCAN61  CAL     DIVKEY                                                       FT073710
        TRA     SCAN3A                                                       FT073720
        SPACE   3                                                            FT073730
        REM                     , PROCESSING                                 FT073740
SCAN73  CAL     CMAKEY                                                       FT073750
        TRA     SCAN3A                                                       FT073760
        SPACE   5                                                            FT073770
        REM                     ( PROCESSING                                 FT073780
SCAN74  CAL     LPNKEY                                                       FT073790
        TRA     SCAN3A                                                       FT073800
        SPACE   5                                                            FT073810
        REM                     END-OF-STATEMENT PROCESSING                  FT073820
```

```
SCAN77 LXA     SCANX,1                                          FT073830
       TRA     1,4                                              FT073840
       SPACE   6                                                FT073850
*      ILLEGAL FIRST CHARACTER                                  FT073860
       REM                                                      FT073870
SCAN2  ERROR   77,SKEND                                         FT073880
```

```
        TTL     P A S S  1  -  S U B S C R I P T  S C A N N E R          FT073890
*       THIS ROUTINE SCANS A SUBSCRIPT EXPRESSION AND PREPARES A        FT073900
*       SUMMARY OF THE DATA NEEDED TO COMPUTE THE SUBSCRIPT.            FT073910
        REM                                                             FT073920
*       CALLING SEQUENCE...  TSX 1INDEX,4                               FT073930
*       WHERE   XR2 POINTS TO THE BEGINNING OF THE SUBSCRIPT PROPER.    FT073940
*               XR1 HAS THE ASTACK POINTER                              FT073950
        REM                                                             FT073960
        REM                                                             FT073970
1INDEX  SXA     1INDX1,4                                                FT073980
        SXA     1INDX1+1,1                                              FT073990
        AXT     .ENT-.STT,4     ZERO THE WORKING CELLS                  FT074000
        STZ     .ENT,4                                                  FT074010
        TIX     *-1,4,1                                                 FT074020
        REM                                                             FT074030
        CAL     ASTACK,1        GET SYMBOL REFERENCE                    FT074040
        STA     1IND16          SAVE POINTER FOR POSSIBLE USE LATER     FT074050
        PAX     ,4                                                      FT074060
        CAL     EQUIV,4         EQUIV POINTER                           FT074070
        PAX     ,4                                                      FT074080
        REM                                                             FT074090
        CAL     POOL-1,4        GET DIMTBL INFORMATION                  FT074100
        ARS     18                                                      FT074110
        STA     DIM1            FIRST DIMENSION                         FT074120
        ARS     15                                                      FT074130
        STA     DIMN            NUMBER OF DIMENSIONS                    FT074140
        REM                                                             FT074150
        CAL     POOL-2,4        GET NEXT DIMENSION, IF ANY              FT074160
        ARS     18                                                      FT074170
        STA     DIM2            SECOND DIMENSION                        FT074180
        REM                                                             FT074190
        AXT     0,1             XR1 BUMPED FOR EACH NEW SEGMENT         FT074200
1INDX4  TSX     SCNBCD,4        GET VARIABLE OR CONSTANT                FT074210
        TMI     1INDX2          CONSTANT                                FT074220
        REM                                                             FT074230
1IND13  LAS     =HI00000        CHECK FOR FIXED POINT                   FT074240
        TRA     *+1                                                     FT074250
        LAS     =HN(((((                                                FT074260
        ERROR   12,*+2          NOT FIXED POINT                         FT074270
        MACER                   MACHINE ERROR                           FT074280
        REM                                                             FT074290
        TSX     SYMSTO,4        FIX UP SYMBOL                           FT074300
        SLW     VARB,1          SAVE EQUIV POINTER                      FT074310
        STI     VIND,1          SAVE EQUIV WORD                         FT074320
        REM                                                             FT074330
        REM                                                             FT074340
        CAL     COLUMN,2        GET BREAK CHARACTER                     FT074350
        TXI     *+1,2,-1                                                FT074360
        ARS     30                                                      FT074370
        STZ     ASIGN                                                   FT074380
        LAS     =H00000+        IS IT PLUS SIGN.                        FT074390
        TRA     *+2                                                     FT074400
        TRA     1INDX5          YES                                     FT074410
        REM                                                             FT074420
        STL     ASIGN           NO.  INDICATE A POSSIBLE MINUS SIGN     FT074430
        LAS     =H00000-        IS IT A MINUS SIGN.                     FT074440
        TRA     *+2                                                     FT074450
        TRA     1INDX5          YES                                     FT074460
        REM                                                             FT074470
1INDX6  LAS     =H00000)        IS IT CLOSING PAREN.                    FT074480
```

```
         TRA     *+2                                                FT074490
         TXI     1INDX3,1,1      YES.  SCAN COMPLETED               FT074500
         REM                                                        FT074510
         ERA     =H00000,        NO.  IS IT A COMMA.                FT074520
         TNZ     1IND16          ILLEGAL FORM OF SUBSCRIPT          FT074530
         TXI     1INDX4,1,1      RETURN FOR ANOTHER SUBSCRIPT       FT074540
         REM                                                        FT074550
1INDX5   TSX     SCNBCD,4        GET CONSTANT                       FT074560
         TPL     1IND16          ERROR.  NOT A CONSTANT             FT074570
         REM                                                        FT074580
         TSX     BCDFIX,4        CONVERT CONSTANT                   FT074590
         ZET     ASIGN           SET SIGN OF CONSTANT               FT074600
         SSM                                                        FT074610
         ARS     18                                                 FT074620
1INDX7   STO     CADD,1          ADDITIVE CONSTANT                  FT074630
         REM                                                        FT074640
         CAL     COLUMN,2        GET BREAK CHARACTER                FT074650
         ARS     30                                                 FT074660
         TXI     1INDX6,2,-1     GO TEST BREAK CHARACTER            FT074670
         REM                                                        FT074680
1INDX2   TSX     BCDFIX,4        CONSTANT MULTIPLIER                FT074690
         ARS     18                                                 FT074700
         XCL                     SAVE THIS CONSTANT                 FT074710
         REM                                                        FT074720
         CAL     COLUMN,2        GET BREAK CHARACTER                FT074730
         ARS     30                                                 FT074740
         LAS     =H00000*        IS IT AN ASTERISK.                 FT074750
         TRA     *+2                                                FT074760
         TXI     *+3,2,-1        YES                                FT074770
         XCL                     GET THE CONSTANT AGAIN             FT074780
         TRA     1INDX7                                             FT074790
         REM                                                        FT074800
         STQ     CMLT,1          CONSTANT IS A MULTIPLIER           FT074810
         TSX     SCNBCD,4        ASTERISK.  GET VARIABLE            FT074820
         TPL     1IND13          PROCESS REST OF EXPRESSION         FT074830
         TRA     1IND16          NOT A VARIABLE.  ILLEGAL FORM      FT074840
         REM                                                        FT074850
1INDX3   TXL     1IND14-1,1,1    ANY SINGLY-DIMENSIONED SUBSCRIPT ALLOWED FT074860
         PXA     ,1                                                 FT074870
         SUB     DIMN            CHECK FOR CORRECT DIMENSIONING OF SUBSCRIPTFT074880
         TZE     1IND14          CORRECT NUMBER OF DIMENSIONS       FT074890
         REM                                                        FT074900
1IND16   AXT     **,4            GET SYMTAB POINTER                 FT074910
         CAL     SYMTAB,4        GET BCD NAME OF OFFENDING ARRAY    FT074920
         SLW     FORM42+6        PUT INTO ERROR MESSAGE             FT074930
         ERROR   42,SKEND                                           FT074940
         REM                                                        FT074950
         SXA     DIMN,1          FOR PSEUDO-SINGLY DIMENSIONED ARRAYS FT074960
1IND14   LDQ     =0              COMPUTE THE CONSTANT ADDEND        FT074970
         CAL     =2                                                 FT074980
         LAS     DIMN            HOW MANY DIMENSIONS.               FT074990
         TRA     1INDX8          ONE DIMENSION                      FT075000
         TRA     1INDX9          TWO DIMENSIONS                     FT075010
         REM                                                        FT075020
         CLS     =1              THREE DIMENSIONS                   FT075030
         ADD     CADD-2          THIRD ADDITIVE CONSTANT            FT075040
         XCA                                                        FT075050
         MPY     DIM2            TIMES SECOND DIMENSION             FT075060
         REM                                                        FT075070
1INDX9   XCA                                                        FT075080
```

```
        SUB     =1                                                      FT075090
        ADD     CADD-1          SECOND ADDITIVE CONSTANT                FT075100
        XCA                                                             FT075110
        MPY     DIM1            TIMES FIRST DIMENSION                   FT075120
        REM                                                             FT075130
1INDX8  XCA                                                             FT075140
        ADD     CADD            FIRST ADDITIVE CONSTANT                 FT075150
        PAC     ,4              TAKE COMPLEMENT FOR POSITIVE CONSTANT ADD. FT075160
        TPL     *+2                                                     FT075170
        PAX     ,4              FOR MINUS CONSTANT ADDEND               FT075180
        TXI     *+1,4,1         ADD ONE FOR ARRAYS                      FT075190
        SXA     CONADD,4        CONSTANT ADDEND                         FT075200
        REM                                                             FT075210
        CAL     =2              GET CONSTANT MULTIPLIERS FOR ISTACK     FT075220
        LAS     DIMN                                                    FT075230
        TRA     1IND10          ONE DIMENSION                           FT075240
        TRA     1IND11          TWO DIMENSIONS                          FT075250
        REM                                                             FT075260
        NZT     VIND-2          THREE DIMENSIONS.  IS THERE A 3RD VARIABLE.FT075270
        TRA     1IND11          NO.                                     FT075280
        REM                                                             FT075290
        LDQ     DIM2            YES                                     FT075300
        MPY     DIM1                                                    FT075310
        ZET     CMLT-2          IS THERE A THIRD MULTIPLIER.            FT075320
        MPY     CMLT-2          YES                                     FT075330
        STQ     SAV3            SAVE THIRD CONSTANT                     FT075340
        REM                                                             FT075350
1IND11  NZT     VIND-1          IS THERE A SECOND VARIABLE.             FT075360
        TRA     1IND10          NO.                                     FT075370
        REM                                                             FT075380
        LDQ     DIM1            YES.                                    FT075390
        ZET     CMLT-1          IS THERE A SECOND MULTIPLIER.           FT075400
        MPY     CMLT-1          YES                                     FT075410
        STQ     SAV2            SAVE SECOND CONSTANT                    FT075420
        REM                                                             FT075430
1IND10  NZT     VIND            IS THERE A FIRST VARIABLE.              FT075440
        TRA     IBUILD          NO.  PROCEED TO BUILD ISTACK            FT075450
        CAL     CMLT            IS THERE A FIRST MULTIPLIER.            FT075460
        TNZ     *+2                                                     FT075470
        CAL     =1              NO MULTIPLIER EXPLICITLY GIVEN          FT075480
        SLW     SAV1                                                    FT075490
        REM                     NOW GO BUILD THE ISTACK ENTRY           FT075500
```

```
        EJECT                                                                   FT075510
*       THIS SECTION BUILDS THE ISTACK ENTRY.                                   FT075520
*       THE FORM OF THE ENTRY IS...                                             FT075530
*               FIRST WORD ADDRESS HAS THE ASTACK POINTER.  THE                 FT075540
*               DECREMENT MAY CONTAIN THE ADDRESS OF THE NEXT UNUSED            FT075550
*               SIX-WORD ISTACK ENTRY.                                          FT075560
*               SECOND WORD IS THE ORIGINAL ASTACK WORD.                        FT075570
*               THIRD WORD ADDRESS IS THE CONSTANT ADDEND.                      FT075580
*               FOR A SUBSCRIPT WHICH REQUIRES NO EXTRA CODE, THE               FT075590
*               THIRD WORD DECREMENT HAS THE EQUIV POINTER (IF ANY)             FT075600
*               FOR THE SINGLE VARIABLE INVOLVED (THIS ISTACK WORD HAS A        FT075610
*               MINUS SIGN).                                                    FT075620
*               IF CODE MUST BE GENERATED TO COMPUTE THE SUBSCRIPT,             FT075630
*               THE THIRD WORD DECREMENT HAS ZERO UNTIL THE CODE IS             FT075640
*               GENERATED, AT WHICH TIME IT WILL RECEIVE THE NUMBER             FT075650
*               OF THE WORKING CELL USED TO COMPUTE AND HOLD THE                FT075660
*               SUBSCRIPT.  THE ISTACK WORD HAS A PLUS SIGN FOR THIS CASE       FT075670
*               IF THE SUBSCRIPT HAS ONLY A CONSTANT ADDEND, THE                FT075680
*               DECREMENT OF THE THIRD WORD IS ZERO (PLUS SIGN).                FT075690
*               ALSO IF CODE MUST BE GENERATED TO COMPUTE THE SUBSCRIPT,        FT075700
*               THE SIGN OF THE FIRST ISTACK WORD IS MADE MINUS.                FT075710
*                                                                               FT075720
*               ONE OR MORE OF THE NEXT THREE WORDS MAY CONTAIN                 FT075730
*               THE LINEAR COEFFICIENTS (IN DECREMENTS) FOR THE                 FT075740
*               SUBSCRIPT VARIABLES (EQUIV POINTERS IN ADDRESSES).              FT075750
*               ANY WORDS NOT USED ARE ZEROED OUT.  THEY ARE STILL A            FT075760
*               PART OF THE ISTACK ENTRY.                                       FT075770
*                                                                               FT075780
*       THE ASTACK ENTRY GETS THE ADDRESS OF THE ISTACK ENTRY IN ITS            FT075790
*       ADDRESS FIELD, AND A NON-ZERO TAG FIELD.                                FT075800
*                                                                               FT075810
*       THE ISTACK ENTRIES ARE LINKED TOGETHER SO THAT THEY MAY BE              FT075820
*       RE-USED FOR EACH STATEMENT.  THE LINKING DIRECTOR IS THE CELL           FT075830
*       ISTKFG, WHICH WILL CONTAIN THE START OF ISTACKS IN ITS ADDRESS          FT075840
*       AND THE ADDRESS OF THE LAST-USED (IN THIS STATEMENT) ISTACK ENTRY       FT075850
*       IN ITS DECREMENT.  THIS CELL IS INITIALIZED TO ZERO AT THE              FT075860
*       BEGINNING OF EACH COMPILATION.  ON ENTRY TO THE COMPIL ROUTINE,         FT075870
*       THE DECREMENT IS SET EQUAL TO THE ADDRESS.                              FT075880
        REM                                                                     FT075890
        REM                                                                     FT075900
IBUILD  CAL     CMLT            IF THESE THREE WORDS ARE ZERO, THE              FT075910
        ACL     VIND-2          SUBSCRIPT IS JUST A VARIABLE IN THE             FT075920
        ACL     VIND-1          FIRST DIMENSION POSSIBLY PLUS A CONSTANT.       FT075930
        SLW     ASIGN                                                           FT075940
        REM                                                                     FT075950
        LXA     ISTKFG,1        GET ADDRESS OF LAST-USED ISTACK ENTRY           FT075960
        CAL     POOL,1          GET FIRST WORD OF ENTRY                         FT075970
        PDX     ,4              GET LINKER TO NEXT ISTACK ENTRY                 FT075980
        TXH     IBLD2,4,0       SKIP IF NON-ZERO                                FT075990
        REM                                                                     FT076000
        LXA     NXTLOC,4        NO MORE AVAILABLE SPACE.  GET NEW SPACE         FT076010
        PXD     ,4                                                              FT076020
        STD     POOL,1          SET LINKER INTO PREVIOUS ISTACK ENTRY           FT076030
        TXI     *+1,4,6         BUMP NXTLOC                                     FT076040
        SXA     NXTLOC,4        RESET NXTLOC                                    FT076050
        TXI     *+1,4,-6        AND RESTORE THE POINTER                         FT076060
        REM                                                                     FT076070
IBLD2   SXA     ISTKFG,4        SET THE FLAG TO CURRENT ISTACK ENTRY USED       FT076080
        LXA     1INDX1+1,1      RESTORE XR1 (ASTACK POINTER)                    FT076090
        LDQ     ASTACK,1        GET ASTACK WORD                                 FT076100
```

```
        STQ     POOL-1,4        PUT INTO SECOND ISTACK WORD                     FT076110
        PXA     ,4              PUT POINTER INTO ACCUM                          FT076120
        STA     ASTACK,1        SET ASTACK POINTER TO ISTACK                    FT076130
        COM                                                                     FT076140
        STT     ASTACK,1        NON-ZERO TAG INTO ASTACK                        FT076150
        REM                                                                     FT076160
        PXA     ,1              GET ASTACK POINTER                              FT076170
        STA     POOL,4          PUT IN ADDRESS OF FIRST ISTACK WORD             FT076180
        STP     POOL,4          ERASE ANY PREVIOUS SIGN BIT                     FT076190
        REM                                                                     FT076200
        ZAC                                                                     FT076210
        ZET     ASIGN           DETERMINE IF ANY CODE IS NEEDED                 FT076220
        TRA     IBUIL1          YES. CODE IS NEEDED.                            FT076230
        CAL     VIND            GET A POSSIBLE FIRST VARIABLE                   FT076240
        TZE     *+3             IF ZERO, SUBSCRIPT IS ONLY A CONSTANT           FT076250
        REM                                                                     FT076260
        CLS     VARB            GET VARIABLE EQUIV POINTER                      FT076270
        ALS     18              TO DECREMENT                                    FT076280
IBUIL1  ORA     CONADD          INSERT THE CONSTANT ADDEND                      FT076290
        STO     POOL-2,4        PUT C.A. AND POINTER INTO ISTACK 3RD WORD       FT076300
        REM                                                                     FT076310
        ZET     ASIGN           IS CODE NEEDED.                                 FT076320
        TRA     IBUIL2          YES.  CONTINUE BUILDING ISTACK                  FT076330
        REM                                                                     FT076340
TINDX1  AXT     **,4                                                            FT076350
        AXT     **,1                                                            FT076360
        TRA     1,4                                                             FT076370
        REM                                                                     FT076380
        REM                                                                     FT076390
IBUIL2  PXA     ,4                                                              FT076400
        PAX     ,1                                                              FT076410
        REM                                                                     FT076420
        CLS     POOL,1          SET SIGN MINUS IN FIRST POOL WORD.              FT076430
        STO     POOL,1          THIS INDICATES THAT CODE MUST BE GENERATED      FT076440
        REM                     TO COMPUTE THE SUBSCRIPT                        FT076450
        REM                                                                     FT076460
        STZ     POOL-3,1                                                        FT076470
        STZ     POOL-4,1                                                        FT076480
        STZ     POOL-5,1                                                        FT076490
        REM                                                                     FT076500
        TXI     *+1,1,3                                                         FT076510
        CAL     SAV1            GET FIRST MULTIPLIER                            FT076520
        TZE     IBUIL3                                                          FT076530
        ALS     18              SHIFT INTO DECREMENT                            FT076540
        TSX     ASCON1,4        PUT CONSTANT INTO CONSTANT TABLE                FT076550
        ALS     18              MOVE POINTER INTO DECREMENT                     FT076560
        ORA     VARB            ADD IN EQUIV POINTER FOR FIRST VARIABLE         FT076570
        SLW     POOL,1          MAKE ISTACK (POOL) ENTRY                        FT076580
        TXI     *+1,1,1         BUMP POOL POINTER                               FT076590
        REM                                                                     FT076600
IBUIL3  CAL     SAV2                                                            FT076610
        TZE     IBUIL4                                                          FT076620
        ALS     18                                                              FT076630
        TSX     ASCON1,4                                                        FT076640
        ALS     18                                                              FT076650
        ORA     VARB-1          OR IN NEXT EQUIV POINTER                        FT076660
        SLW     POOL,1          PUT INTO ISTACK                                 FT076670
        TXI     *+1,1,1                                                         FT076680
        REM                                                                     FT076690
IBUIL4  CAL     SAV3                                                            FT076700
```

```
TZE     1INDX1                                          FT076710
ALS     18                                              FT076720
TSX     ASCON1,4                                        FT076730
ALS     18                                              FT076740
ORA     VARB-2          ADD IN THIRD EQUIV POINTER      FT076750
SLW     POOL,1          PUT INTO ISTACK                 FT076760
TRA     1INDX1                                          FT076770
```

```
        EJECT                                                                   FT076780
*       ROUTINE TO PRODUCE CODE NECESSARY TO EVALUATE A SUBSCRIPT.              FT076790
*       ENTRY IS...                                                             FT076800
*               CAL   ISTACK.POINTER                                            FT076810
*               TSX   ICODEX,4                                                  FT076820
*       ON RETURN, THE DECREMENT OF THE THIRD ISTACK WORD HAS THE               FT076830
*       WORKING CELL POINTER WHERE THE RESULT OF THE COMPUTATION                FT076840
*       IS STORED, IF ANY CODE WAS PRODUCED.                                    FT076850
        REM                                                                     FT076860
        REM                                                                     FT076870
ICODEX  SAVE    EXICOD                                                          FT076880
        STZ     ASIGN           INITIALIZE FLAG CELL                            FT076890
        PAX     ,2              GET ISTACK POINTER                              FT076900
        CLA     POOL,2          GET FIRST ISTACK WORD                           FT076910
        TPL     EXICOD          EXIT IF NO CODE IS NEEDED                       FT076920
        REM                                                                     FT076930
        TXI     *+1,2,3         BUMP POINTER AND SET DECREMENT OF TEST INST     FT076940
        SXD     1IND24,2                                                        FT076950
        TXI     *+1,2,2         SET POINTER TO BOTTOM OF STACK                  FT076960
        TSX     STORE,4         MUST STORE OFF ANY CURRENT REGISTER             FT076970
        REM                                                                     FT076980
        TSX     GETWRK,4        GET A WORKING CELL                              FT076990
        SLW     TEMPX           SAVE                                            FT077000
        REM                                                                     FT077010
1IND15  CAL     POOL,2          GET MULTIPLIER                                  FT077020
        TZE     1IND19+1        NO CODE HERE                                    FT077030
        PAX     ,1              GET EQUIV POINTER                               FT077040
        ARS     18              MOVE CONTAB POINTER INTO ADDRESS                FT077050
        CODEO   LDQ,C           CODE IS...  LDQ CONSTANT                        FT077060
        REM                                                                     FT077070
        NZT     ASIGN           HAS CODE ALREADY BEEN PRODUCED.                 FT077080
        TRA     1IND27          NO                                              FT077090
        REM                                                                     FT077100
        CODECO  STO,TEMPX,W     CODE IS...  STO TEMPORARY                       FT077110
        REM                                                                     FT077120
1IND27  LDI     EQUIV,1         GET EQUIV WORD FOR VARIABLE                     FT077130
        PXA     ,1              EQUIV POINTER                                   FT077140
        LFT     BARGT           IS IT A PARAMETER                               FT077150
        TRA     1IND18          YES                                             FT077160
        REM                                                                     FT077170
        CODEO   MPY,V           CODE IS...  MPY VARIABLE                        FT077180
        REM                                                                     FT077190
1IND25  NZT     ASIGN           HAS CODE ALREADY BEEN PRODUCED.                 FT077200
        TRA     1IND19          NO                                              FT077210
        REM                                                                     FT077220
        CODECO  ADD,TEMPX,W     CODE IS...  ADD TEMPORARY                       FT077230
1IND19  STL     ASIGN           SIGNAL THAT CODE HAS BEEN PRODUCED              FT077240
        REM                                                                     FT077250
1IND24  TXL     *+2,2,**        DECREMENT WAS SET EARLIER                       FT077260
        TXI     1IND15,2,-1     LOWER POINTER AND RETURN                        FT077270
        REM                                                                     FT077280
1IND22  CODEC   ALS17,N         CODE IS...  ALS 17                              FT077290
        CODECO  STO,TEMPX,W     CODE IS...  STO TEMPORARY                       FT077300
        REM                                                                     FT077310
        CAL     TEMPX           WORKING CELL POINTER                            FT077320
        ALS     18              SHIFT INTO DECREMENT                            FT077330
        STD     POOL+1,2        SET 3RD ISTACK WORD DECREMENT TO WRK PTR         FT077340
EXICOD  RESTOR                                                                  FT077350
        TRA     1,4                                                             FT077360
        REM                                                                     FT077370
```

```
        REM
1IND18  TSX     PROLOG,4        PREPARE PROLOGUE ENTRIES                FT077380
                                                                        FT077390
        CODEC   MPY,N           CODE IS...  MPY **                      FT077400
        TRA     1IND25                                                  FT077410
```

```
         TTL A R I T H M E T I C  C O D E  K E Y  G E N E R A T O R          FT077420
*        GEN ROUTINE.  THIS ROUTINE BUILDS THE CODE GENERATOR KEY WORD       FT077430
*        FOR THE SEQUENCE 'OPERAND/OPERATOR/OPERAND' OR 'OPERATOR/OPERAND'FT077440
*        THIS ROUTINE PROCESSES ALL OPERATORS EXCEPT PARENS AND COMMA.       FT077450
*        NOTE THAT THE 'OPERATOR/OPERAND' CASE MAY ONLY HAVE A UNARY         FT077460
*        MINUS AS THE OPERATOR.                                              FT077470
         REM                                                                 FT077480
*        THE FORM OF THE GENERATED KEY WORD IS...                            FT077490
*                 BITS P,1-6          OPERATOR KEY CODE                      FT077500
*                 BITS 7-9            MODE OF LEFT-HAND OPERAND, IF ANY      FT077510
*                 BITS 10-11          REGISTER LOCATION OF THIS OPERAND      FT077520
*                 BITS 12-14          MODE OF RIGHT-HAND OPERAND             FT077530
*                 BITS 15-16          REGISTER LOCATION OF THIS OPERAND      FT077540
*                 BITS 17-35          NOT USED                               FT077550
         REM                                                                 FT077560
*        AFTER THE KEY WORD IS FORMED BY THE GEN ROUTINE, THE                FT077570
*        APPROPRIATE GENERATOR KEY TABLE IS                                  FT077580
*        SEARCHED (BINARY SEARCH).  IF THE PROPER KEY IS NOT FOUND,          FT077590
*        AN ERROR MESSAGE IS ISSUED.                                         FT077600
*        IF THE KEY IS FOUND, THE ROUTINE TRANSFERS TO THE ADDRESS           FT077610
*        SPECIFIED IN THE ADDRESS FIELD OF THE LOCATED KEY.  THE PROPER      FT077620
*        CODE IS THEN PUT OUT.                                               FT077630
         REM                                                                 FT077640
*        THE APPLICABLE GENERATOR KEY TABLE IS ESTABLISHED AT THE            FT077650
*        BEGINNING OF THE COMPILE ROUTINE ACCORDING TO THE COLUMN            FT077660
*        1 MODE OF THE STATEMENT.                                            FT077670
*        THE ASTACK ENTRIES FOR THE LEFT-HAND AND RIGHT-HAND OPERANDS        FT077680
*        ARE PLACED IN THE CELLS 'A' AND 'B' FOR USE BY THE CODE ROUTINE.    FT077690
*        ALSO, THE ASTACK POINTERS ARE PLACED IN THE CELLS 'A+1' AND 'B+1'FT077700
         SPACE  3                                                            FT077710
GEN      STZ    AUXFG1          INDICATES ENTRY NOT FROM PLACE OR STORE      FT077720
         SXA    GENXIT,4                                                    FT077730
         LDQ    =0              ZERO THE MQ                                  FT077740
         REM                                                                 FT077750
         CAL    ASTACK,1        B OPERAND ENTRY                              FT077760
         PAI                                                                 FT077770
         LFT    BEXTF+BINTF                                                  FT077780
         TRA    GEN8                                                         FT077790
         SLW    B               SET CELLS FOR THE CODE ROUTINE               FT077800
         SXA    B+1,1                                                        FT077810
         REM                                                                 FT077820
         RQL    34              SKIP TWO BITS FOR B-OPERAND REGISTER         FT077830
         ARS    33              MODE OF B OPERAND                            FT077840
         LGR    3               BUILD KEY WORD                               FT077850
         NZT    AUXFG1          SKIP IF NOT FROM PLACE OR STORE ROUTINES     FT077860
         TXI    GEN10,1,-2      BUMP ASTACK POINTER TO THE 'A' OPERAND       FT077870
         CAL    AUXILX          GET OP KEY FOR PLACE AND STORE SEQUENCES     FT077880
         LGR    12              FORM THE KEY WORD                            FT077890
         TRA    GEN11                                                        FT077900
         REM                                                                 FT077910
AUXILX   VFD    24/,7/AUXIL                                                  FT077920
         REM                                                                 FT077930
GEN10    CAL    ASTACK,1        GET THE 'A' OPERAND                          FT077940
         PAI                                                                 FT077950
         SLW    A               SET CELLS FOR CODE ROUTINE                   FT077960
         SXA    A+1,1                                                        FT077970
         REM                                                                 FT077980
         ARS    19              GET OPERATOR FLAG INTO LOW-ORDER BIT         FT077990
         LBT                    IS IT AN OPERATOR.                           FT078000
         TRA    GEN1            NO                                           FT078010
```

```
         REM                                                          FT078020
         TXI   *+1,1,1     YES. (MUST BE UNARY MINUS CODE SEQUENCE)   FT078030
         REM               RESTORE THE ASTACK POINTER                 FT078040
         STZ   A+1         DELETE A-OPERAND INDICATIONS               FT078050
         RQL   31          ZERO OUT THE A OPERAND BITS                FT078060
         CAL   ASTACK,1    GET THE OPERATOR                           FT078070
         REM                                                          FT078080
GEN2     ARS   29          OPERATOR KEY                               FT078090
         LGR   7           BUILD KEY WORD                             FT078100
         REM                                                          FT078110
GEN11    STQ   GEN6        TEMPORARY SAVE                             FT078120
         NZT   REGIST                                                 FT078130
         TRA   GEN5        SKIP IF NOTHING IN A REGISTER CURRENTLY    FT078140
         REM                                                          FT078150
         CAL   REGIST+1    GET STACK POINTER FOR OPERAND IN REGISTER  FT078160
         LAS   B+1         IS IT SAME AS B-OPERAND.                   FT078170
         TRA   *+2         NO                                         FT078180
         TRA   GEN3        YES                                        FT078190
         ERA   A+1         IS IT SAME AS A-OPERAND.                   FT078200
         TZE   GEN4        YES                                        FT078210
         REM                                                          FT078220
         STQ   STOSAV      SAVE THE GEN6 WORD TEMPORARILLY            FT078230
         TSX   STORE,4     MUST PUT REGISTER RESULT IN WORKING STORE  FT078240
         CAL   STOSAV      RESET THE GEN6 WORD FOR THE ORIGINAL CODE  FT078250
         SLW   GEN6                                                   FT078260
         TRA   GEN5                                                   FT078270
         REM                                                          FT078280
GEN3     CAL   REGIST      GET CODE FOR REGISTER                      FT078290
         ALS   19          MOVE INTO POSITION FOR B-OPERAND KEY       FT078300
         TRA   *+3                                                    FT078310
         REM                                                          FT078320
GEN4     CAL   REGIST      GET CODE FOR REGISTER                      FT078330
         ALS   24          MOVE INTO POSITION FOR A-OPERAND KEY       FT078340
         ORS   GEN6        FORM COMPLETED KEY WORD                    FT078350
         REM                                                          FT078360
GEN5     SXA   BEXIT,1     SAVE XR1                                   FT078370
BPOWER   AXT   **,4        GET POWER OF 2 LENGTH OF TABLE             FT078380
         AXT   **,1        INITIAL RAISE/LOWER TABLE POINTER          FT078390
         REM                                                          FT078400
BA       TXH   BR,4,**     SKIP IF OUTSIDE THE TABLE                  FT078410
         CAL   =0777777700000     MASK                              FT078420
BAND     ANA   **,4        AND WITH A WORD FROM THE TABLE             FT078430
         REM                                                          FT078440
         LAS   GEN6        COMPARE WITH THE KEY WORD                  FT078450
         TIX   BLOWER,1,1  ELEMENT ABOVE KEY.  LOWER POINTER          FT078460
         TRA   *+2         ELEMENT FOUND, OR END OF SEARCH            FT078470
BR       TIX   BRAISE,1,1  ELEMENT BELOW KEY.  RAISE POINTER          FT078480
         REM                                                          FT078490
BEXIT    AXT   **,1        RESTORE XR1                                FT078500
         ERA   GEN6        TEST TO SEE IF EQUALITY FOUND              FT078510
GEN98    TZE*  **,4        IF EQUALITY FOUND, GO PUT OUT THE CODE.    FT078520
         REM               DEPENDS ON GENERATOR KEYS HAVING ZERO TAGS FT078530
         ERROR 7,SKEND     NOT EQUAL.  ERROR                          FT078540
         REM                                                          FT078550
GENXIT   AXT   **,4                                                   FT078560
         TRA   1,4         RETURN                                     FT078570
         REM                                                          FT078580
GEN1     ARS   14          BUILD THE KEY WORD                         FT078590
         RQL   34          ZERO OUT TWO BITS                          FT078600
         LGR   3           MODE BITS FOR A-OPERAND                    FT078610
```

```
         CAL     ASTACK-1,1      GET OPERATOR                                    FT078620
GEN9     LFT     BEXTF+BINTF     FLAGS MUST BE OFF                               FT078630
         TRA     *+2                                                             FT078640
         TRA     GEN2                                                            FT078650
         REM                                                                     FT078660
GEN8     PIA                     REGAIN THE ASTACK WORD                          FT078670
         PAX     ,4              POINTER                                         FT078680
         RFT     700000          IS IT SUBSCRIPTED.                              FT078690
         CAL     POOL-1,4        YES.  GET WORD WITH EQUIV POINTER IN IT         FT078700
         PAX     ,4              GET EQUIV POINTER FOR SURE THIS TIME            FT078710
         CAL     SYMTAB,4        GET BCD NAME OF OFFENDING SYMBOL                FT078720
         SLW     FORM69          PUT INTO ERROR MESSAGE                          FT078730
         ERROR   69,SKEND                                                        FT078740
         REM                                                                     FT078750
         REM                                                                     FT078760
         PZE     KEYTB3+LKYTB3,,LKYTB3       COMPLEX ARITHMETIC KEYS             FT078770
         PZE     KEYTB2+LKYTB2,,LKYTB2       DOUBLE PRECISION KEYS               FT078780
         PZE     KEYTB1+LKYTB1,,LKYTB1       BOOLEAN KEYS                        FT078790
GENDES   PZE     KEYTB0+LKYTB0,,LKYTB0       REAL AND INTEGER KEYS               FT078800
         REM                                                                     FT078810
         BSS     4               HOLDS POWER-OF-TWO LENGTHS OF THE KEY           FT078820
         REM                     TABLES IN DECREMENT, AND STARTING VALUE         FT078830
         REM                     OF RAISE-LOWER TABLE POINTER IN ADDRESS.        FT078840
GENBIN   EQU     *-1             SET DURING PRE-INITIALIZATION.                  FT078850
         SPACE   2                                                               FT078860
*        LISTS USED IN THE BINARY SEARCH OVER THE KEY TABLES                     FT078870
         REM                                                                     FT078880
         TXI     BA,4,16384      TABLE FOR LOWERING THE SEARCH POINT.            FT078890
         TXI     BA,4,8192                                                       FT078900
         TXI     BA,4,4096                                                       FT078910
         TXI     BA,4,2048                                                       FT078920
         TXI     BA,4,1024                                                       FT078930
         TXI     BA,4,512                                                        FT078940
         TXI     BA,4,256                                                        FT078950
         TXI     BA,4,128                                                        FT078960
         TXI     BA,4,64                                                         FT078970
         TXI     BA,4,32                                                         FT078980
         TXI     BA,4,16                                                         FT078990
         TXI     BA,4,8                                                          FT079000
         TXI     BA,4,4                                                          FT079010
         TXI     BA,4,2                                                          FT079020
         TXI     BA,4,1                                                          FT079030
BLOWER   XEC     *,1                                                             FT079040
         REM                                                                     FT079050
         TXI     BA,4,-16384     TABLE FOR RAISING THE SEARCH POINT              FT079060
         TXI     BA,4,-8192                                                      FT079070
         TXI     BA,4,-4096                                                      FT079080
         TXI     BA,4,-2048                                                      FT079090
         TXI     BA,4,-1024                                                      FT079100
         TXI     BA,4,-512                                                       FT079110
         TXI     BA,4,-256                                                       FT079120
         TXI     BA,4,-128                                                       FT079130
         TXI     BA,4,-64                                                        FT079140
         TXI     BA,4,-32                                                        FT079150
         TXI     BA,4,-16                                                        FT079160
         TXI     BA,4,-8                                                         FT079170
         TXI     BA,4,-4                                                         FT079180
         TXI     BA,4,-2                                                         FT079190
         TXI     BA,4,-1                                                         FT079200
BRAISE   XEC     *,1                                                             FT079210
```

```
        TTL     O P E N   F U N C T I O N   K E Y   G E N E R A T O R   FT079220
*       ROUTINE TO GENERATE THE CODE KEY WORD FOR OPEN (BUILT-IN)       FT079230
*       FUNCTIONS.                                                      FT079240
        REM                                                             FT079250
OPNGEN  SXA     OPN1,4          SAVE RETURN POINT                       FT079260
        LDQ     =0              ZERO THE MQ                             FT079270
        CAL     ASTACK,1        GET OPERAND                             FT079280
        PAI                     SAVE THE ASTACK WORD                    FT079290
        LFT     BEXTF+BINTF     FLAGS MUST BE OFF                       FT079300
        TRA     GEN8                                                    FT079310
        SLW     B                                                       FT079320
        SXA     B+1,1                                                   FT079330
        REM                                                             FT079340
        ARS     33                                                      FT079350
        LGR     3                                                       FT079360
        TXI     *+1,1,-2        BUMP ASTACK POINTER TO NEXT ITEM        FT079370
        CAL     ASTACK,1                                                FT079380
        PBT                     IS IT THE FUNCTION CODE YET.            FT079390
        TRA     *+4             NO                                      FT079400
        RQL     31                                                      FT079410
        STZ     A+1             DELETE ANY A-OPERAND INDICATIONS        FT079420
        TRA     OPN1                                                    FT079430
        REM                                                             FT079440
        PAI                     SAVE THE ASTACK WORD                    FT079450
        SLW     A                                                       FT079460
        SXA     A+1,1                                                   FT079470
        ARS     33                                                      FT079480
        RQL     34                                                      FT079490
        LGR     3                                                       FT079500
        REM                                                             FT079510
        LXA     PNFLAG,4                                                FT079520
        CLA     ASTACK,4        GET KEY WORD AND LOSE HIGH-ORDER BIT    FT079530
        LAS     =15B6           IS IT MAX OR MIN FUNCTION.              FT079540
        TXI     NOTMAX,1,-2     NO.  RESTORE POINTER                    FT079550
        REM                                                             FT079560
        TRA     *+1             MAX OR MIN FUNCTION                     FT079570
        ARS     30              MOVE KEY NUMBER AND DIVIDE BY TWO       FT079580
        STA     MAXPTR          SAVE IN DIRECTOR WORD                   FT079590
        CAL     ASTACK+2,1      GET 'NEXT' ARGUMENT FROM ASTACK         FT079600
        PBT                                                             FT079610
        TRA     *+2             NOT THE OPERATION CODE                  FT079620
        TXI     OPN1,1,-2       NO MORE ARGUMENTS.  GO SET EXIT POINT   FT079630
        REM                                                             FT079640
        CAL     ASTACK,4        GET OPERATOR KEY WORD                   FT079650
        AXC     OPNGEN,4        GET RETURN POINT FOR MAX AND MIN CODE   FT079660
        TRA     OPN1+1          GO PUT OUT CODE                         FT079670
        REM                                                             FT079680
NOTMAX  CAL     ASTACK,1        NOT MAX OR MIN.  MUST BE ALL THE ARGUMENTS FT079690
        PBT                     IF BIT IS OFF, THEN ERROR (MORE ARGUMENTS) FT079700
        ERROR   7,SKEND                                                 FT079710
        REM                                                             FT079720
OPN1    AXT     **,4            GET THE CURRENT EXIT POINT FOR GEN      FT079730
        SXA     GENXIT,4        SET SO 'RESULT' ROUTINE EXITS PROPERLY  FT079740
        TRA     GEN9            GO PUT OUT CODE                         FT079750
```

```
        TTL     A R I T H M E T I C  C O D E  G E N E R A T O R S
*       SUBROUTINES CALLED BY THE GENERATORS
        SPACE   3
        REM
        REM                         ROUTINE TO OUTPUT THE SEQUENCE
        REM                              UFA  =0233000000000
        REM                              FAD  =0233000000000
        REM
        REM
INTXXX  SXA     INTXX,4
        NZT     FLAG)1          HAS OCTAL 233000000000 BEEN ASSIGNED YET.
        TSX     SET.1,4         NO.  GO ASSIGN THE CONSTANT
        CAL     UFA)1
        TSX     CITBLD,4,C
        CAL     FAD)1
        TSX     CITBLD,4,C
INTXX   AXT     **,4
        TRA     1,4
        SPACE   5
        REM                         ROUTINE TO OUTPUT THE SEQUENCE
        REM                              UFA  =0233000000000
        REM                              LRS
        REM                              ANA  =0377777
        REM                              LLS
        REM                              ALS  18
        REM
XINTXX  SXA     XINTX,4
        NZT     FLAG)1
        TSX     SET.1,4
        NZT     FLAG)2
        TSX     SET.2,4
        CAL     UFA)1
        TSX     CITBLD,4,C
        CODEN   LRS
        CAL     ANA)2
        TSX     CITBLD,4,C
        CODEN   LLS
        CODEN   ALS18
XINTX   AXT     **,4
        TRA     1,4
        SPACE   5
        REM                         ROUTINE TO OUTPUT YHE SEQUENCE
        REM                              LRS  18
        REM                              ORA  =0233000000000
        REM                              FAD  =0233000000000
        REM
FLOXXX  SXA     FLOXX,4
        NZT     FLAG)1
        TSX     SET.1,4
        CODEN   LRS18
        CAL     ORA)1
        TSX     CITBLD,4,C
        CAL     FAD)1
        TSX     CITBLD,4,C
FLOXX   AXT     **,4
        TRA     1,4
        SPACE   6
        REM                     ROUTINE TO ASSIGN EXPONENT TO STORAGE
SET)1   SXA     EXSET1,4
        CAL     =0233000000000
        TSX     ASCON1,4        ASSIGN CONSTANT TO STORAGE
```

```
        STA     FAD)1           SET THE ADDRESSES OF THE PERTINENT      FT080360
        STA     ORA)1           INSTRUCTIONS                            FT080370
        STA     UFA)1                                                   FT080380
        STL     FLAG)1          SET NONZERO FLAG                        FT080390
EXSET1  AXT     **,4                                                    FT080400
        TRA     1,4                                                     FT080410
        REM                                                             FT080420
SET.1   EQU     SET)1                                                   FT080430
        SPACE   6                                                       FT080440
        REM             ROUTINE TO ASSIGN 0377777 TO STORAGE            FT080450
SET)2   SXA     EXSET1,4                                                FT080460
        CAL     =0377777        GET INTEGER MASK BITS                   FT080470
        TSX     ASCON1,4        PUT INTO STORAGE                        FT080480
        STA     ANA)2           PUT POINTER INTO INSTRUCTION            FT080490
        STL     FLAG)2          SET NONZERO FLAG                        FT080500
        TRA     EXSET1                                                  FT080510
        REM                                                             FT080520
SET.2   EQU     SET)2                                                   FT080530
        SPACE   6                                                       FT080540
        REM               ROUTINE TO OUTPUT A CALL ENTRY TO AN          FT080550
        REM             EXTERNAL SUBROUTINE                             FT080560
        REM               ENTRY...                                      FT080570
        REM                    TSX  OUTPTV,4                            FT080580
        REM                         NAME.OF.SUBROUTINE                  FT080590
        REM                                                             FT080600
OUTPTV  SXA     OUTPTX,4                                                FT080610
        CLA     1,4                                                     FT080620
        STL*    1,4                                                     FT080630
        ANA     =077777                                                 FT080640
        ORA     TSX4                                                    FT080650
        ORS     XRFLAG                                                  FT080660
        CITBLD  S                                                       FT080670
OUTPTX  AXT     **,4                                                    FT080680
        TRA     2,4                                                     FT080690
        SPACE   6                                                       FT080700
*       ROUTINE TO DETECT CASES LIKE  I = 0  AND TURN OUT A STZ.        FT080710
        REM                                                             FT080720
STZTSD  STL     DVSTFG          DOUBLE PRECISION ENTRY                  FT080730
        TRA     *+2                                                     FT080740
        REM                                                             FT080750
STZTST  STZ     DVSTFG          SINGLE PRECISION AND BOOLEAN ENTRY      FT080760
        LDI     B               GET ASTACK WORD                         FT080770
        LNT     BCNST           IS THIS A CONSTANT                      FT080780
        TRA     2,2             NO.  RETURN                             FT080790
        REM                                                             FT080800
        LXA     B,4             YES.  GET THE CONTAB POINTER            FT080810
        ZET     CONTAB,4        IS CONSTANT ZERO.                       FT080820
        TRA     2,2             NO.  RETURN                             FT080830
        REM                                                             FT080840
        ZET     DVSTFG          IS THIS DOUBLE-LENGTH.                  FT080850
        TRA     STZTS1          YES                                     FT080860
        REM                                                             FT080870
        CODE    STZ,A           NO.  CODE IS...  STZ  A                 FT080880
        TRA*    1,2             RETURN, SKIPPING OTHER CODE             FT080890
        REM                                                             FT080900
STZTS1  CLA     1,2             GET THE CONTROL WORD                    FT080910
        TMI     *+3             SKIP IF THIS IS THE  D = I CASE         FT080920
        ZET     CONTAB+1,4      IS SECOND PART ZERO.                    FT080930
        TRA     2,2             NO.  RETURN                             FT080940
        REM                                                             FT080950
```

```
         CODSTZ  A               PUT OUT THE CODE                          FT080960
         TRA*    1,2                                                       FT080970
         SPACE   6                                                         FT080980
*        ROUTINE TO TURN OUT IN-LINE CODE                                  FT080990
*        FOR EXPONENTIATION SEQUENCES...  R**2 THROUGH R**7.               FT081000
         REM                                                               FT081010
REXPC1   LDI     B               GET THE ASTACK WORD FOR THE EXPONENT      FT081020
         LNT     BCNST           IS IT A CONSTANT                          FT081030
         TRA     1,4             NO                                        FT081040
         REM                                                               FT081050
         SXA     EXREX,4         SAVE XR4                                  FT081060
         LXA     B,4             GET THE CONTAB POINTER                    FT081070
         CLA     CONTAB,4        AND GET THE CONSTANT                      FT081080
         TMI     EXREX           QUIT IF NEGATIVE (SHOULDN'T HAPPEN)       FT081090
         PDC     ,4              SIZE OF EXPONENT COMPLEMENTED             FT081100
         TXL     EXREX,4,-8      POWER GREATER THAN 7, OR ZERO             FT081110
         TXH     GENXIT,4,-2     R**1      NO CODE NECESSARY               FT081120
         REM                                                               FT081130
         SXA     REXPC4,4        SAVE EXPONENT (COMPLEMENTED)              FT081140
         CAL     REGIST          GET REGISTER LOCATION OF ARGUMENT         FT081150
         ANA     =02             SAVE ONLY THE MQ BIT, IF ANY              FT081160
         SLW     T1              SAVE                                      FT081170
         TSX     STORE,4         STORE OFF THE ARGUMENT                    FT081180
         CAL     ASTACK,1        AND RESET THE 'A' WORD AGAIN              FT081190
         SLW     A               (ASTACK MAY HAVE BEEN CHANGED BY STORE)   FT081200
         REM                                                               FT081210
         ZET     T1              WAS ARGUMENT IN THE MQ.                   FT081220
         TRA     *+3             YES.  DON'T LOAD THE MQ AGAIN             FT081230
         CODE    LDQ,A                                                     FT081240
         CODE    FMP,A           SQUARE THE NUMBER                         FT081250
         REM                                                               FT081260
REXPC4   AXT     **,4            REGAIN THE EXPONENT (COMPLEMENTED)        FT081270
         TXH     REXPC9,4,-3     IS IT POWER 2.                            FT081280
         CODEN   XCA             NO     CONTINUE                           FT081290
         REM                                                               FT081300
         LXA     REXPC4,4        REGAIN EXPONENT                           FT081310
         TRA     *-2,4           BRANCH                                    FT081320
         TRA     REXP3                                                     FT081330
         TRA     REXP4                                                     FT081340
         TRA     REXP5                                                     FT081350
         TRA     REXP6                                                     FT081360
         REM                                                               FT081370
REXP7    CODE    FMP,A           R**7                                      FT081380
         CODEN   XCA                                                       FT081390
         REM                                                               FT081400
REXP5    CODEN   STQ777          R**5                                      FT081410
         STL     T1              SET AN AVAILABLE FLAG CELL                FT081420
         TRA     REXP6+1         SKIP                                      FT081430
         REM                                                               FT081440
REXP6    STZ     T1              R**6      SET FLAG CELL                   FT081450
         CODE    FMP,A                                                     FT081460
         CODEN   XCA                                                       FT081470
         ZET     T1              IS THIS POWER 6.                          FT081480
         TRA     *+3             NO                                        FT081490
         REM                                                               FT081500
REXP4    CODEN   STQ777          R**4                                      FT081510
         CODEN   FMP777                                                    FT081520
REXPC9   RESULT  REAL,AC,ROUND                                             FT081530
         REM                                                               FT081540
REXP3    CODE    FMP,A           R**3                                      FT081550
```

```
         RESULT  REAL,AC,ROUND                                          FT081560
         REM                                                           FT081570
EXREX    AXT     **,4          RESTORE XR4 AND RETURN                  FT081580
         TRA     2,4                                                   FT081590
```

```
        EJECT                                                       FT081600
        REM                    - CODE GENERATORS -                  FT081610
        REM                                                         FT081620
        REM                          ABSF(B)                        FT081630
ABS01   CODE    CLA,B                                               FT081640
ABS02   CODEN   SSP                                                 FT081650
        RESULT  REAL,AC                                             FT081660
ABS03   CODEN   XCA                                                 FT081670
        TRA     ABS02                                               FT081680
        REM                                                         FT081690
        REM                          XABSF(I)                       FT081700
XABS01  CODE    CLA,B                                               FT081710
XABS02  CODEN   SSP                                                 FT081720
        RESULT  INTEGR,AC                                           FT081730
XABS03  CODEN   XCA                                                 FT081740
        TRA     XABS02                                              FT081750
        REM                                                         FT081760
        REM                          INTF(B)                        FT081770
INT01   CODE    CLA,B                                               FT081780
INT02   TSX     INTXXX,4                                            FT081790
        RESULT  REAL,AC                                             FT081800
INT03   CODEN   XCA                                                 FT081810
        TRA     INT02                                               FT081820
        REM                                                         FT081830
        REM                          XINTF(B)                       FT081840
XINT01  CODE    CLA,B                                               FT081850
XINT02  TSX     XINTXX,4                                            FT081860
        RESULT  INTEGR,AC                                           FT081870
XINT03  CODEN   XCA                                                 FT081880
        TRA     XINT02                                              FT081890
        REM                                                         FT081900
        REM                          FLOATF(I)                      FT081910
FLOAT1  CODE    CLA,B                                               FT081920
FLOAT2  TSX     FLOXXX,4                                            FT081930
        RESULT  REAL,AC                                             FT081940
FLOAT3  CODEN   XCA                                                 FT081950
        TRA     FLOAT2                                              FT081960
        REM                                                         FT081970
        REM                          XFIXF(A)                       FT081980
XFIX01  EQU     XINT01                                              FT081990
XFIX02  EQU     XINT02                                              FT082000
XFIX03  EQU     XINT03                                              FT082010
        SPACE   6                                                   FT082020
XLOC01  LDI     B               XLOCF(I)   OR   XLOCF(R)            FT082030
        LFT     BVARB+BARGT     MUST BE ONE OR THE OTHER            FT082040
        TRA     *+2                                                 FT082050
        ERROR   7,SKEND                                             FT082060
        REM                                                         FT082070
        RFT     700000          IS IT SUBSCRIPTED                   FT082080
        TRA     XLOC0A          YES.                                FT082090
XLOC0B  CODE    AXT4,B                                              FT082100
        TRA     XLOC0C                                              FT082110
        REM                                                         FT082120
XLOC0A  LXA     B,4             SEE IF SUBSCRIPT IS ONLY A CONSTANT ADDEND FT082130
        CAL     POOL,4                                              FT082140
        ORA     POOL-2,4                                            FT082150
        PBT                                                         FT082160
        TRA     XLOC0B                                              FT082170
        REM                                                         FT082180
        CODE    PXA4,B                                              FT082190
```

```
        CAL     PCOUNT                                                  FT082200
        SUB     =1                                                      FT082210
        CODEO   SUB,P                                                   FT082220
        CODEN   PAX4                                                    FT082230
XLOCOC  CODEN   PXD4                                                    FT082240
        RESULT  INTEGR,AC                                               FT082250
        SPACE   6                                                       FT082260
MXMIN1  CODE    CLA,A       CODE GENERATORS FOR ALL MAX AND MIN'S       FT082270
MXMIN4  CODE    LDQ,B                                                   FT082280
MXMINA  CODEP   TLQ2                                                    FT082290
        CODEN   XCA                                                     FT082300
        REM                                                             FT082310
        CLA     ASTACK,1    ROUTING SEQUENCE                            FT082320
MAXPTR  AXC     **,4        GET PREVIOUSLY SET DIRECTOR                 FT082330
        TPL     MAXINT,4    IF NOT KEY WORD, GO TO INTERMEDIATE CODE    FT082340
        CAL     XCA         PREPARE FOR POSSIBLE CODE SEQUENCE          FT082350
        TRA     MAXTRM,4    GO TO TERMINAL SEQUENCE                     FT082360
        REM                                                             FT082370
MXMIN2  CODE    LDQ,A                                                   FT082380
        TRA     MXMINA                                                  FT082390
MXMIN3  CODE    CLA,A                                                   FT082400
        TRA     MXMINA                                                  FT082410
MXMIN5  CODE    CLA,B                                                   FT082420
        TRA     MXMINA                                                  FT082430
        REM                                                             FT082440
MAXINT  EQU     *                                                       FT082450
        RESULT  REAL,MQ     MIN1                                        FT082460
        RESULT  INTEGR,MQ   MIN0                                        FT082470
        RESULT  INTEGR,AC   MAX0                                        FT082480
        RESULT  REAL,AC     MAX1                                        FT082490
        RESULT  INTEGR,MQ   XMIN0                                       FT082500
        RESULT  REAL,MQ     XMIN1                                       FT082510
        RESULT  REAL,AC     XMAX1                                       FT082520
        RESULT  INTEGR,AC   XMAX0                                       FT082530
        REM                                                             FT082540
MAXTRM  EQU     *                                                       FT082550
        TRA     SIGNOC      MIN1                                        FT082560
        TSX     CITBLD,4,N  MIN0.  PUT OUT XCA                          FT082570
        TSX     FLOXXX,4    MAX0.  FLOAT THE INTEGER                    FT082580
        RESULT  REAL,AC     MAX1                                        FT082590
        RESULT  INTEGR,MQ   XMIN0                                       FT082600
        TSX     CITBLD,4,N  XMIN1.  PUT OUT XCA                         FT082610
        TSX     XINTXX,4    XMAX1.  FIX THE REAL RESULT                 FT082620
        RESULT  INTEGR,AC   XMAX0                                       FT082630
        SPACE   6                                                       FT082640
XSIGN1  CODE    CLA,A       XSIGNF(I,J)                                 FT082650
XSIGN4  CODE    LDQ,B                                                   FT082660
XSIGNA  CODEN   LLS                                                     FT082670
        RESULT  INTEGR,AC                                               FT082680
XSIGN2  CODE    LDQ,A                                                   FT082690
XSIGNB  CODEN   LRS                                                     FT082700
        RESULT  INTEGR,MQ                                               FT082710
XSIGN3  CODE    CLA,A                                                   FT082720
        TRA     XSIGNA                                                  FT082730
XSIGN5  CODE    CLA,B                                                   FT082740
        TRA     XSIGNB                                                  FT082750
        SPACE   2                                                       FT082760
SIGNO1  CODE    CLA,A       SIGNF(A,B)                                  FT082770
SIGNO4  CODE    LDQ,B                                                   FT082780
SIGNOA  CODEN   LLS                                                     FT082790
```

```
        RESULT  REAL,AC                                                  FT082800
SIGN02  CODE    LDQ,A                                                   FT082810
SIGN0B  CODEN   LRS                                                     FT082820
SIGN0C  NZT     MODFLG          IF BOOLEAN, MUST GET RESULT OUT OF MQ.  FT082830
        RESULT  REAL,MQ                                                 FT082840
        CODEN   XCA                                                     FT082850
        RESULT  REAL,AC                                                 FT082860
SIGN03  CODE    CLA,A                                                   FT082870
        TRA     SIGN0A                                                  FT082880
SIGN05  CODE    CLA,B                                                   FT082890
        TRA     SIGN0B                                                  FT082900
        SPACE   2                                                       FT082910
DIM01   CODE    CLA,A           DIMF(A,B)                               FT082920
DIM04   CODE    FSB,B                                                   FT082930
DIM0A   CODEP   TPL2                                                    FT082940
        CODEN   ZAC                                                     FT082950
        RESULT  REAL,AC                                                 FT082960
DIM03   CODEN   XCA                                                     FT082970
DIM02   CODE    FSB,A                                                   FT082980
        CODEN   CHS                                                     FT082990
        TRA     DIM0A                                                   FT083000
DIM05   CODEN   XCA                                                     FT083010
        TRA     DIM04                                                   FT083020
        SPACE   2                                                       FT083030
XDIM01  CODE    CLA,A           XDIMF(I,J)                              FT083040
XDIM04  CODE    SUB,B                                                   FT083050
XDIM0A  CODEP   TPL2                                                    FT083060
        CODEN   ZAC                                                     FT083070
        RESULT  INTEGR,AC                                               FT083080
XDIM03  CODEN   XCA                                                     FT083090
XDIM02  CODE    SUB,A                                                   FT083100
        CODEN   CHS                                                     FT083110
        TRA     XDIM0A                                                  FT083120
XDIM05  CODEN   XCA                                                     FT083130
        TRA     XDIM04                                                  FT083140
        SPACE   2                                                       FT083150
XMOD01  CODE    CLA,A           XMODF(I,J)                              FT083160
XMOD04  CODEN   LRS35                                                   FT083170
        CODE    DVP,B                                                   FT083180
        RESULT  INTEGR,AC                                               FT083190
XMOD05  CODEN   XCA                                                     FT083200
        TRA     XMOD04                                                  FT083210
XMOD02  CODEN   STO777                                                  FT083220
XMOD0B  CODE    CLA,A                                                   FT083230
        CODEN   LRS35                                                   FT083240
        CODEN   DVP777                                                  FT083250
        RESULT  INTEGR,AC                                               FT083260
XMOD03  CODEN   STQ777                                                  FT083270
        TRA     XMOD0B                                                  FT083280
        SPACE   2                                                       FT083290
        REM             MODF(A,B)                                       FT083300
MOD02   TSX     STORE,4                                                 FT083310
        CAL     ASTACK-2,1                                              FT083320
        SLW     A                                                       FT083330
        CAL     ASTACK-4,1                                              FT083340
        SLW     B                                                       FT083350
MOD01   CODE    CLS,A                                                   FT083360
MOD0A2  CODE    FDP,B                                                   FT083370
        CODEN   XCA                                                     FT083380
        TSX     INTXXX,4                                                FT083390
```

```
         CODEN   XCA                                                    FT083400
         CODE    FMP,B                                                  FT083410
MODA1    CODE    FAD,A                                                  FT083420
MODA3    CODEP   TNZ2                                                   FT083430
         CODEN   SSP                                                    FT083440
         RESULT  REAL,AC                                                FT083450
MOD03    EQU     MOD02                                                  FT083460
MOD04    EQU     MOD02                                                  FT083470
MOD05    EQU     MOD02                                                  FT083480
         SPACE   2                                                      FT083490
         REM                      I=I                                   FT083500
IEQI01   TSX     STZTST,2       I = I                                   FT083510
         PZE     IEQIA                                                  FT083520
         CAL     A              SKIP OUT IF A = B                       FT083530
         ERA     B                                                      FT083540
         TZE     IEQIB                                                  FT083550
         CODE    CLA,B                                                  FT083560
IEQI02   CODE    STO,A                                                  FT083570
IEQIA    LDI     A                                                      FT083580
         TSX     INDXT,4        TEST FOR VALID CHANGE OF INDEX OR SUBSCRIPTFT083590
IEQIB    TXL     CMPLX,1,0                                              FT083600
         ERROR   84,SKEND                                               FT083610
         REM                                                            FT083620
IEQI03   CODE    STQ,A                                                  FT083630
         TRA     IEQIA                                                  FT083640
         REM                                                            FT083650
         REM                      I+I                                   FT083660
IADD01   CODE    CLA,A                                                  FT083670
IADD04   CODE    ADD,B                                                  FT083680
         RESULT  INTEGR,AC                                              FT083690
IADD02   CODE    ADD,A                                                  FT083700
         RESULT  INTEGR,AC                                              FT083710
IADD03   CODEN   XCA                                                    FT083720
         TRA     IADD02                                                 FT083730
IADD05   CODEN   XCA                                                    FT083740
         TRA     IADD04                                                 FT083750
         REM                                                            FT083760
         REM                      I-I                                   FT083770
ISUB01   CODE    CLA,A                                                  FT083780
ISUB04   CODE    SUB,B                                                  FT083790
         RESULT  INTEGR,AC                                              FT083800
ISUB02   CODEN   CHS                                                    FT083810
         CODE    ADD,A                                                  FT083820
         RESULT  INTEGR,AC                                              FT083830
ISUB03   CODEN   XCA                                                    FT083840
         TRA     ISUB02                                                 FT083850
ISUB05   CODEN   XCA                                                    FT083860
         TRA     ISUB04                                                 FT083870
         REM                                                            FT083880
         REM                      I*I                                   FT083890
IMPY01   CODE    LDQ,A                                                  FT083900
IMPY05   CODE    MPY,B                                                  FT083910
IMPYA    CODEN   ALS17                                                  FT083920
         RESULT  INTEGR,AC                                              FT083930
IMPY03   CODE    MPY,A                                                  FT083940
         TRA     IMPYA                                                  FT083950
IMPY02   CODEN   XCA                                                    FT083960
         TRA     IMPY03                                                 FT083970
IMPY04   CODEN   XCA                                                    FT083980
         TRA     IMPY05                                                 FT083990
```

```
        REM                                                          FT084000
        REM                    I/I                                   FT084010
IDIV01  CODE    CLA,A                                                FT084020
IDIV04  CODEN   LRS35                                                FT084030
IDIVC   CODE    DVP,B                                                FT084040
IDIVA   CODEN   LLS18                                                FT084050
        RESULT  INTEGR,MQ                                            FT084060
IDIV02  CODEN   STO777                                               FT084070
IDIVB   CODE    CLA,A                                                FT084080
        CODEN   LRS35                                                FT084090
        CODEN   DVP777                                               FT084100
        TRA     IDIVA                                                FT084110
IDIV03  CODEN   STQ777                                               FT084120
        TRA     IDIVB                                                FT084130
IDIV05  CODEN   ZAC                                                  FT084140
        CODEN   LLS                                                  FT084150
        TRA     IDIVC                                                FT084160
        REM                                                          FT084170
        REM                    -I                                    FT084180
INEG01  CODE    CLS,B                                                FT084190
        RESULT  INTEGR,AC                                            FT084200
INEG03  CODEN   XCA                                                  FT084210
INEG02  CODEN   CHS                                                  FT084220
        RESULT  INTEGR,AC                                            FT084230
        REM                                                          FT084240
        REM                    I**I                                  FT084250
IEXP01  CODE    CLA,A                                                FT084260
IEXP04  CODE    LDQ,B                                                FT084270
IEXPA   TSX     OUTPTV,4                                             FT084280
                EXP(1                                                FT084290
        RESULT  INTEGR,AC                                            FT084300
IEXP02  CODEN   XCA                                                  FT084310
IEXP03  CODE    CLA,A                                                FT084320
        TRA     IEXPA                                                FT084330
IEXP05  CODEN   XCA                                                  FT084340
        TRA     IEXP04                                               FT084350
        REM                                                          FT084360
        REM                    R=I                                   FT084370
REQI01  TSX     STZTST,2                                             FT084380
        PZE     IEQIB                                                FT084390
        CODE    CLA,B                                                FT084400
REQI02  TSX     FLOXXX,4                                             FT084410
        TRA     IEQI02                                               FT084420
REQI03  CODEN   XCA                                                  FT084430
        TRA     REQI02                                               FT084440
        REM                                                          FT084450
        REM                    I=R                                   FT084460
IEQR01  TSX     STZTST,2                                             FT084470
        PZE     IEQIA                                                FT084480
        CODE    CLA,B                                                FT084490
IEQR02  TSX     XINTXX,4                                             FT084500
        TRA     IEQI02                                               FT084510
IEQR03  CODEN   XCA                                                  FT084520
        TRA     IEQR02                                               FT084530
        REM                                                          FT084540
        REM                    R=R                                   FT084550
REQR01  EQU     IEQI01                                               FT084560
REQR02  EQU     IEQI02                                               FT084570
REQR03  EQU     IEQI03                                               FT084580
        REM                                                          FT084590
```

```
        REM                         R+R                                 FT084600
RADD01  CODE    CLA,A                                                   FT084610
RADD04  CODE    FAD,B                                                   FT084620
        RESULT  REAL,AC,ROUND                                           FT084630
RADD02  CODE    FAD,A                                                   FT084640
        RESULT  REAL,AC,ROUND                                           FT084650
RADD03  CODEN   XCA                                                     FT084660
        TRA     RADD02                                                  FT084670
RADD05  CODEN   XCA                                                     FT084680
        TRA     RADD04                                                  FT084690
        REM                                                             FT084700
        REM                         R-R                                 FT084710
RSUB01  CODE    CLA,A                                                   FT084720
RSUB04  CODE    FSB,B                                                   FT084730
        RESULT  REAL,AC,ROUND                                           FT084740
RSUB02  CODEN   CHS                                                     FT084750
        TRA     RADD02                                                  FT084760
RSUB03  CODEN   XCA                                                     FT084770
        TRA     RSUB02                                                  FT084780
RSUB05  CODEN   XCA                                                     FT084790
        TRA     RSUB04                                                  FT084800
        REM                                                             FT084810
        REM                         -R                                  FT084820
RNEG01  CODE    CLS,B                                                   FT084830
        RESULT  REAL,AC                                                 FT084840
RNEG03  CODEN   XCA                                                     FT084850
RNEG02  CODEN   CHS                                                     FT084860
        RESULT  REAL,AC                                                 FT084870
        REM                                                             FT084880
        REM                         R*R                                 FT084890
RMPY01  CODE    LDQ,A                                                   FT084900
RMPY05  CODE    FMP,B                                                   FT084910
        RESULT  REAL,AC,ROUND                                           FT084920
RMPY02  CODEN   XCA                                                     FT084930
RMPY03  CODE    FMP,A                                                   FT084940
        RESULT  REAL,AC,ROUND                                           FT084950
RMPY04  CODEN   XCA                                                     FT084960
        TRA     RMPY05                                                  FT084970
        REM                                                             FT084980
        REM                         R/R                                 FT084990
RDIV01  CODE    CLA,A                                                   FT085000
RDIV04  CODE    FDP,B                                                   FT085010
        RESULT  REAL,MQ                                                 FT085020
RDIV02  CODEN   STO777                                                  FT085030
RDIVA   CODE    CLA,A                                                   FT085040
        CODEN   FDP777                                                  FT085050
        RESULT  REAL,MQ                                                 FT085060
RDIV03  CODEN   STQ777                                                  FT085070
        TRA     RDIVA                                                   FT085080
RDIV05  CODEN   XCA                                                     FT085090
        TRA     RDIV04                                                  FT085100
        REM                                                             FT085110
        REM                         R**R                                FT085120
REXP01  CODE    CLA,A                                                   FT085130
REXP04  CODE    LDQ,B                                                   FT085140
REXPA   TSX     OUTPTV,4                                                FT085150
                EXP(3                                                   FT085160
        RESULT  REAL,AC                                                 FT085170
REXP02  CODE    LDQ,A                                                   FT085180
        CODEN   XCA                                                     FT085190
```

```
         TRA     REXPA                                                  FT085200
REXP03   CODE    CLA,A                                                  FT085210
         TRA     REXPA                                                  FT085220
REXP05   CODEN   XCA                                                    FT085230
         TRA     REXP04                                                 FT085240
         REM                                                            FT085250
         REM                      R**I                                  FT085260
REXPI1   TSX     REXPC1,4                                               FT085270
         CODE    CLA,A                                                  FT085280
         TRA     REXPAB                                                 FT085290
REXPI4   TSX     REXPC1,4                                               FT085300
REXPAB   CODE    LDQ,B                                                  FT085310
REXPIA   TSX     OUTPTV,4                                               FT085320
         EXPI2                                                          FT085330
         RESULT  REAL,AC                                                FT085340
REXPI2   CODEN   XCA                                                    FT085350
REXPI3   CODE    CLA,A                                                  FT085360
         TRA     REXPIA                                                 FT085370
REXPI5   TSX     REXPC1,4                                               FT085380
         CODEN   XCA                                                    FT085390
         TRA     REXPAB                                                 FT085400
         SPACE   2                                                      FT085410
ASFRR1   CODE    CLA,B          ARITH. STATE. FN...  R = R             FT085420
ASFRR2   TRA     ASFEND                                                 FT085430
ASFRR3   CODEN   XCA                                                    FT085440
         TRA     ASFEND                                                 FT085450
         SPACE   2                                                      FT085460
ASFII1   EQU     ASFRR1         ARITH. STATE. FN...  I = I             FT085470
ASFII2   EQU     ASFRR2                                                 FT085480
ASFII3   EQU     ASFRR3                                                 FT085490
         SPACE   2                                                      FT085500
ASFRI1   CODE    CLA,B          ARITH. STATE. FN...  R = I             FT085510
ASFRI2   TSX     FLOXXX,4                                               FT085520
         TRA     ASFEND                                                 FT085530
ASFRI3   CODEN   XCA                                                    FT085540
         TRA     ASFRI2                                                 FT085550
         SPACE   2                                                      FT085560
ASFIR1   CODE    CLA,B          ARITH. STATE. FN...  I = R             FT085570
ASFIR2   TSX     XINTXX,4                                               FT085580
         TRA     ASFEND                                                 FT085590
ASFIR3   CODEN   XCA                                                    FT085600
         TRA     ASFIR2                                                 FT085610
         SPACE   6                                                      FT085620
                                 B = B                                  FT085630
BEQB01   TSX     STZTST,2                                               FT085640
         PZE     IEQIB                                                  FT085650
         CODE    CAL,B                                                  FT085660
BEQB02   CODE    SLW,A                                                  FT085670
         TRA     IEQIB                                                  FT085680
         REM                                                            FT085690
                                 - B                                    FT085700
BCOM01   CODE    CAL,B                                                  FT085710
BCOM02   CODEN   COM                                                    FT085720
         RESULT  REAL,AC                                                FT085730
         REM                                                            FT085740
                                 B + B                                  FT085750
BOR01    CODE    CAL,B                                                  FT085760
BOR02    CODE    ORA,A                                                  FT085770
         RESULT  REAL,AC                                                FT085780
BOR04    CODE    ORA,B                                                  FT085790
```

```
         RESULT  REAL,AC                                                    FT085800
         REM                                                                FT085810
                                 B * B                                      FT085820
BAND01   CODE    CAL,B                                                      FT085830
BAND02   CODE    ANA,A                                                      FT085840
         RESULT  REAL,AC                                                    FT085850
BAND04   CODE    ANA,B                                                      FT085860
         RESULT  REAL,AC                                                    FT085870
         SPACE   2                                                          FT085880
ASFRR4   CODE    CAL,B          ARITH. STATE. FN...  B = B                  FT085890
ASFRR5   TRA     ASFEND                                                     FT085900
         REM                                                                FT085910
         SPACE   8                                                          FT085920
DABS01   CODLD   B              DABSF(A)                                   FT085930
DABS02   CODEN   SSP                                                        FT085940
         CODEN   LRS                                                        FT085950
DABS0A   CODEN   STO777                                                     FT085960
         CODEN   STQ776                                                     FT085970
         RESULT  REAL,PAC                                                   FT085980
DABS03   CODEN   CLA775                                                     FT085990
         CODEN   LDQ774                                                     FT086000
         TRA     DABS02                                                     FT086010
         SPACE   2                                                          FT086020
DFLOT1   CODE    CLA,B          DFLOATF(I)                                  FT086030
DFLOT2   TSX     FLOXXX,4                                                   FT086040
         TRA     DABS0A                                                     FT086050
DFLOT3   CODEN   XCA                                                        FT086060
         TRA     DFLOT2                                                     FT086070
         SPACE   2                                                          FT086080
DFIX01   STR     MSGFIX,,SKEND     ***PUT OUT COMPLAINT AGAINST IBM         FT086090
DFIX02   EQU     DFIX01                                                     FT086100
DFIX03   EQU     DFIX01                                                     FT086110
                 6                                                          FT086120
MSGFIX   BCI     6,IBM'S 'FIXF' GENERATES NONSENSE CODE                     FT086130
```

```
        EJECT                                                   FT086140
DADD01  CODLAC  A               D + D                           FT086150
DADD04  SETUP   (DFAD)                                          FT086160
DADD0A  CODSTR  B                                               FT086170
        RESULT  REAL,PAC                                        FT086180
DADD05  TSX     MQ.AC,4                                         FT086190
        TRA     DADD04                                          FT086200
DADD03  TSX     MQ.AC,4                                         FT086210
DADD02  SETUP   (DFAD)                                          FT086220
DADD0B  CODSTR  A                                               FT086230
        RESULT  REAL,PAC                                        FT086240
        SPACE   2                                               FT086250
DSUB01  CODLAC  A               D - D                           FT086260
DSUB04  SETUP   (DFSB)                                          FT086270
        TRA     DADD0A                                          FT086280
DSUB05  TSX     MQ.AC,4                                         FT086290
        TRA     DSUB04                                          FT086300
DSUB03  CODEN   CLS775                                          FT086310
        CODEN   LDQ774                                          FT086320
DSUB0A  CODEN   LRS                                             FT086330
        CODEN   STO777                                          FT086340
        CODEN   STQ776                                          FT086350
        TRA     DADD02                                          FT086360
DSUB02  CODEN   CLS777                                          FT086370
        CODEN   LDQ776                                          FT086380
        TRA     DSUB0A                                          FT086390
        SPACE   2                                               FT086400
DNEG01  CODSUB  B               - D                             FT086410
        RESULT  REAL,PAC                                        FT086420
DNEG02  CODEN   CLS777                                          FT086430
        CODEN   LDQ776                                          FT086440
DNEG0A  CODEN   LRS                                             FT086450
        CODEN   STO777                                          FT086460
        CODEN   STQ776                                          FT086470
        RESULT  REAL,PAC                                        FT086480
DNEG03  CODEN   CLS775                                          FT086490
        CODEN   LDQ774                                          FT086500
        TRA     DNEG0A                                          FT086510
        SPACE   2                                               FT086520
DMPY01  CODLMQ  A               D * D                           FT086530
DMPY05  SETUP   (DFMP)                                          FT086540
        TRA     DADD0A                                          FT086550
DMPY04  TSX     AC.MQ,4                                         FT086560
        TRA     DMPY05                                          FT086570
DMPY02  TSX     AC.MQ,4                                         FT086580
DMPY03  SETUP   (DFMP)                                          FT086590
        TRA     DADD0B                                          FT086600
        SPACE   2                                               FT086610
DDIV01  CODLAC  A               D / D                           FT086620
DDIV04  SETUP   (DFDP)                                          FT086630
        CODSTR  B                                               FT086640
        RESULT  REAL,PMQ                                        FT086650
DDIV05  TSX     MQ.AC,4                                         FT086660
        TRA     DDIV04                                          FT086670
DDIV03  TSX     STORE,4                                         FT086680
        LXA     B+1,4                                           FT086690
        CAL     ASTACK,4                                        FT086700
        SLW     B                                               FT086710
        TRA     DDIV01                                          FT086720
DDIV02  EQU     DDIV03                                          FT086730
```

```
        SPACE   2                                                       FT086740
DEXPR1  CODLAC  A               D ** D                                  FT086750
DEXPR4  CODLMQ  B                                                       FT086760
DEXPRA  TSX     OUTPTV,4                                                FT086770
        DEXP(3                                                          FT086780
        RESULT  REAL,AC                                                 FT086790
DEXPR5  TSX     MQ.AC,4                                                 FT086800
        TRA     DEXPR4                                                  FT086810
DEXPR2  TSX     AC.MQ,4                                                 FT086820
DEXPR3  CODLAC  A                                                       FT086830
        TRA     DEXPRA                                                  FT086840
        SPACE   2                                                       FT086850
DEXPI1  CODLAC  A               D ** I                                  FT086860
DEXPI4  CODE    LDQ,B                                                   FT086870
DEXPIA  TSX     OUTPTV,4                                                FT086880
        DEXP(2                                                          FT086890
        RESULT  REAL,AC                                                 FT086900
DEXPI5  TSX     MQ.AC,4                                                 FT086910
        TRA     DEXPI4                                                  FT086920
DEXPI2  CODEN   STO775                                                  FT086930
DEXPIB  CODLAC  A                                                       FT086940
        CODEN   LDQ775                                                  FT086950
        TRA     DEXPIA                                                  FT086960
DEXPI3  CODEN   STQ775                                                  FT086970
        TRA     DEXPIB                                                  FT086980
        SPACE   2                                                       FT086990
DEQD01  TSX     STZTSD,2        D = D                                   FT087000
        PZE     IEQ1B                                                   FT087010
        CODLD   B                                                       FT087020
DEQDOA  CODSTO  A                                                       FT087030
        TRA     CMPLX                                                   FT087040
DEQD02  ZET     REGFLG                                                  FT087050
        TRA     DEQDOA                                                  FT087060
        CODEN   CLA777                                                  FT087070
        CODEN   LDQ776                                                  FT087080
        TRA     DEQDOA                                                  FT087090
DEQD03  ZET     REGFLG                                                  FT087100
        TRA     DEQDOA                                                  FT087110
        CODEN   CLA775                                                  FT087120
        CODEN   LDQ774                                                  FT087130
        TRA     DEQDOA                                                  FT087140
        SPACE   2                                                       FT087150
DEQI01  TSX     STZTSD,2        D = I                                   FT087160
        MZE     IEQ1B                                                   FT087170
        CODE    CLA,B                                                   FT087180
DEQI02  TSX     FLOXXX,4                                                FT087190
        TRA     DEQDOA                                                  FT087200
DEQI03  CODEN   XCA                                                     FT087210
        TRA     DEQI02                                                  FT087220
        SPACE   2                                                       FT087230
IEQD01  EQU     IEQR01          I = D                                   FT087240
IEQDOA  EQU     IEQR02                                                  FT087250
IEQD02  ZET     REGFLG                                                  FT087260
        TRA     IEQDOA                                                  FT087270
        CODEN   CLA777                                                  FT087280
        TRA     IEQDOA                                                  FT087290
IEQD03  ZET     REGFLG                                                  FT087300
        TRA     IEQDOA                                                  FT087310
        CODEN   CLA775                                                  FT087320
        TRA     IEQDOA                                                  FT087330
```

```
        SPACE   2                                                       FT087340
ASFDD1  CODLAC  B               ARITH. STATE. FN...  D = D              FT087350
ASFDD2  TRA     ASFEND                                                  FT087360
ASFDD3  TSX     MQ.AC,4                                                 FT087370
        TRA     ASFEND                                                  FT087380
        SPACE   2                                                       FT087390
ASFDI1  CODE    CLA,B           ARITH. STATE. FN...  D = I              FT087400
ASFDI2  TSX     FLOXXX,4                                                FT087410
        CODEN   STO777                                                  FT087420
        CODEN   STQ776                                                  FT087430
        TRA     ASFEND                                                  FT087440
ASFDI3  CODEN   XCA                                                     FT087450
        TRA     ASFDI2                                                  FT087460
        SPACE   6                                                       FT087470
CMPY01  CODLMQ  A               C * C                                   FT087480
CMPY05  SETUP   (IFMP)                                                  FT087490
CMPY0A  CODSTR  B                                                       FT087500
        RESULT  REAL,AC                                                 FT087510
CMPY04  TSX     AC.MQ,4                                                 FT087520
        TRA     CMPY05                                                  FT087530
CMPY02  TSX     AC.MQ,4                                                 FT087540
CMPY03  SETUP   (IFMP)                                                  FT087550
        CODSTR  A                                                       FT087560
        RESULT  REAL,AC                                                 FT087570
        SPACE   2                                                       FT087580
CDIV01  CODLAC  A               C / C                                   FT087590
CDIV04  SETUP   (IFDP)                                                  FT087600
        CODSTR  B                                                       FT087610
        RESULT  REAL,MQ                                                 FT087620
CDIV05  TSX     MQ.AC,4                                                 FT087630
        TRA     CDIV04                                                  FT087640
CDIV03  TSX     STORE,4                                                 FT087650
        LXA     B+1,4                                                   FT087660
        CAL     ASTACK,4                                                FT087670
        SLW     B                                                       FT087680
        TRA     CDIV01                                                  FT087690
CDIV02  EQU     CDIV03                                                  FT087700
        SPACE   2                                                       FT087710
CEXPI1  CODLAC  A               C ** I                                  FT087720
CEXPI4  CODE    LDQ,B                                                   FT087730
CEXPIA  TSX     OUTPTV,4                                                FT087740
                IEXPI2                                                  FT087750
        RESULT  REAL,AC                                                 FT087760
CEXPI5  TSX     MQ.AC,4                                                 FT087770
        TRA     CEXPI4                                                  FT087780
CEXPI2  CODEN   STO775                                                  FT087790
CEXPIB  CODLAC  A                                                       FT087800
        CODEN   LDQ775                                                  FT087810
        TRA     CEXPIA                                                  FT087820
CEXPI3  CODEN   STQ775                                                  FT087830
        TRA     CEXPIB                                                  FT087840
```

```
        EJECT                                                                   FT087850
*       THE FOLLOWING ARE SOME ABYSMAL TEMPORARY CODE SEQUENCES FOR             FT087860
*       C + C       C - C       DSIGNF      ISIGNF                              FT087870
        REM                                                                     FT087880
CADD01  CODLAC  A               C + C                                           FT087890
CADD04  CODLD   B                                                               FT087900
CADD0A  CODEN   STQ774                                                          FT087910
        CODEN   FAD777                                                          FT087920
CADD0B  CODEN   STO777                                                          FT087930
        CODEN   CLA776                                                          FT087940
CADD0D  CODEN   FAD774                                                          FT087950
CADD0E  CODEN   STO776                                                          FT087960
        RESULT  REAL,AC                                                         FT087970
CADD02  CODLD   A                                                               FT087980
        TRA     CADD0A                                                          FT087990
CADD03  CODLD   A                                                               FT088000
CADD0C  CODEN   STQ776                                                          FT088010
        CODEN   FAD775                                                          FT088020
        TRA     CADD0B                                                          FT088030
CADD05  CODLD   B                                                               FT088040
        TRA     CADD0C                                                          FT088050
        SPACE   6                                                               FT088060
CSUB01  CODSUB  B               C - C                                           FT088070
        TRA     CADD02                                                          FT088080
CSUB02  CODLD   A                                                               FT088090
        CODEN   STQ774                                                          FT088100
        CODEN   FSB777                                                          FT088110
CSUB0B  CODEN   STO777                                                          FT088120
        CODEN   CLS776                                                          FT088130
        TRA     CADD0D                                                          FT088140
CSUB03  CODLD   A                                                               FT088150
        CODEN   STQ776                                                          FT088160
        CODEN   FSB775                                                          FT088170
CSUB0A  CODEN   STO777                                                          FT088180
        CODEN   CLS774                                                          FT088190
        CODEN   FAD776                                                          FT088200
        TRA     CADD0E                                                          FT088210
CSUB04  CODLD   B                                                               FT088220
        CODEN   STQ774                                                          FT088230
        CODEN   CHS                                                             FT088240
        CODEN   FAD777                                                          FT088250
        TRA     CSUB0A                                                          FT088260
CSUB05  CODLD   B                                                               FT088270
        CODEN   STQ776                                                          FT088280
        CODEN   CHS                                                             FT088290
        CODEN   FAD775                                                          FT088300
        TRA     CSUB0B                                                          FT088310
        SPACE   6                                                               FT088320
DSIGN1  CODLAC  A                                                               FT088330
DSIGN4  CODLD   B                                                               FT088340
        CODEN   STO775                                                          FT088350
        CODEN   CLA776                                                          FT088360
DSIGNB  CODEN   LLS                                                             FT088370
        CODEN   STO776                                                          FT088380
        CODEN   CLA777                                                          FT088390
        CODEN   LDQ775                                                          FT088400
        CODEN   LLS                                                             FT088410
        CODEN   STO777                                                          FT088420
        RESULT  REAL,AC                                                         FT088430
DSIGN2  CODLD   A               DSIGNF(A,B)  AND  ISIGNF(A,B)                   FT088440
```

```
          CODEN   STQ774                                  FT088450
          CODEN   LDQ777                                  FT088460
          CODEN   LLS                                     FT088470
          CODEN   STO777                                  FT088480
          CODEN   CLA774                                  FT088490
          CODEN   LDQ776                                  FT088500
OSIGNA    CODEN   LLS                                     FT088510
          TRA     CADDOE                                  FT088520
DSIGN3    CODLD   A                                       FT088530
          CODEN   STQ776                                  FT088540
          CODEN   LDQ775                                  FT088550
          CODEN   LLS                                     FT088560
          CODEN   STO777                                  FT088570
          CODEN   CLA776                                  FT088580
          CODEN   LDQ774                                  FT088590
          TRA     DSIGNA                                  FT088600
DSIGN5    CODLD   B                                       FT088610
          CODEN   STO777                                  FT088620
          CODEN   CLA774                                  FT088630
          TRA     DSIGNB                                  FT088640
```

```
*       CODE ROUTINE.                                                   FT088660
        REM                                                             FT088670
*       THIS ROUTINE IS ENTERED DURING PROCESSING OF ARITHMETIC         FT088680
*       STATEMENTS VIA COMPIL, GEN, AND ULTIMATELY THE CODE             FT088690
*       GENERATORS.  ONE OR MORE OBJECT INSTRUCTIONS ARE GENERATED.     FT088700
        REM                                                             FT088710
*       THE ENTRY IS...                                                 FT088720
*               TSX  CODE,4                                             FT088730
*               PZE  OP,,STACK                                          FT088740
*       WHERE 'OP' IS THE ADDRESS OF THE OPERATION CODE, AND STACK IS THEFT088750
*       LOCATION OF A WORD CONTAINING A COPY OF THE APPROPRIATE ASTACK  FT088760
*       ENTRY.  THE CORRESPONDING ASTACK POINTER MUST BE IN THE ADDRESS FT088770
*       FIELD OF STACK+1.                                               FT088780
        REM                                                             FT088790
*       THE ENTRY IN STACK MAY DESCRIBE A CONSTANT, VARIABLE,           FT088800
*       ARITHMETIC STATEMENT FUNCTION ARGUMENT, OR AN                   FT088810
*       INTERMEDIATE RESULT HELD IN WORKING STORAGE.                    FT088820
*       IF A VARIABLE IS A PARAMETER, PROLOG IS ENTERED.                FT088830
*       IF THE TAG FIELD OF THE STACK WORD IS NON-ZERO, THE VARIABLE    FT088840
*       IS SUBSCRIPTED.  THE ISTACK POINTER IS THEN THE ADDRESS FIELD OF FT088850
*       STACK.  CODE IS GENERATED TO LOAD XR4 WITH A PREVIOUSLY COMPUTED FT088860
*       QUANTITY AND/OR SUPPLY A CONSTANT ADDEND TO CITBLD OR PROLOG.   FT088870
        REM                                                             FT088880
*       THIS ROUTINE IS EQUIPPED TO PROCESS TWO INSTRUCTIONS SUPPLIED   FT088890
*       BY THE DP AND CA CODE DRIVERS.  THE FLAG SPCDFG IS THE          FT088900
*       ROUTING KEY FOR THIS MODE OF OPERATION.                         FT088910
        REM                                                             FT088920
        REM                                                             FT088930
        REM                                                             FT088940
CODE    STZ     SPCDFG          SET ROUTING FLAG                        FT088950
        SXA     EXCODE,4                                                FT088960
        SXA     EXCODE+1,1                                              FT088970
COD32   CAL     1,4             GET OP CODE AND STACK POINTER           FT088980
        STA     COD1            SET OP CODE INSTRUCTION                 FT088990
        PDC     ,4              STACK POINTER                           FT089000
        CAL     ,4              GET STACK WORD                          FT089010
        LDI     ,4                                                      FT089020
        STZ     COD6                                                    FT089030
        STZ     COD4                                                    FT089040
        REM                                                             FT089050
        RFT     700000          IS THERE AN ISTACK ENTRY FOR THIS ONE   FT089060
        TRA     COD5            YES.  INDEXING MUST TAKE PLACE          FT089070
        REM                                                             FT089080
        ANA     =077777         NO.  SAVE EQUIV POINTER                 FT089090
COD1    ORA     **              OR IN THE OPERATION CODE                FT089100
        ORS     XRFLAG                                                  FT089110
        REM                                                             FT089120
        LFT     BCNST           IS IT A CONSTANT.                       FT089130
        TRA     COD3            YES                                     FT089140
        REM                                                             FT089150
COD13   RFT     40000           IS ARGUMENT IN WORKING STORAGE.         FT089160
        TRA     COD2            YES.                                    FT089170
        REM                                                             FT089180
        LFT     BEXTF           IS IT EXTERNAL                          FT089190
        TRA     COD12           YES.                                    FT089200
        REM                                                             FT089210
        LFT     BARGT           IS IT A SUBPROGRAM PARAMETER.           FT089220
        TRA     COD7            YES                                     FT089230
        REM                                                             FT089240
```

```
         LFT     BARIT           IS IT AN ARITH. STATE. FUNCTION ARGUMENT.  FT089250
         TRA     COD14           YES                                        FT089260
         REM                                                                FT089270
         STA     COD31           SET EQUIV POINTER INTO SECOND INSTRUCTION  FT089280
         ZSD     COD37           NO CONSTANT ADDEND HERE                    FT089290
         TSX     CITBLD,4,V                                                 FT089300
         TRA     COD47+1                                                    FT089310
         REM                                                                FT089320
COD2     ANA     =0777777737777    WORKING STORAGE. REMOVE HIGH-ORDER BIT   FT089330
         STA     COD31           SAVE WORKING POINTER IN SECOND INSTRUCTION  FT089340
         TSX     FREWRK,4        FREE UP THIS WORKING CELL                  FT089350
         TSX     CITBLD,4,W           PUT OUT THE CODE                      FT089360
         NZT     SPCDFG          TEST THE ROUTING FLAG                      FT089370
         TRA     EXCODE          NO SECOND INSTRUCTION NEEDED               FT089380
         CAL     COD31           GET THE SECOND INSTRUCTION                 FT089390
         ACL     =1              BUMP THE WORKING CELL POINTER              FT089400
         STZ     SPCDFG          ALTER THE ROUTING FLAG                     FT089410
         TRA     COD2+2          GO PUT OUT THIS CODE                       FT089420
         REM                                                                FT089430
COD3     STA     COD31           CONSTANT.  SAVE POINTER IN 2ND INSTRUCTION FT089440
         TSX     CITBLD,4,C      PUT OUT THE CODE                           FT089450
         NZT     SPCDFG          TEST THE ROUTING FLAG                      FT089460
         TRA     EXCODE          NO MORE TO DO                              FT089470
         CAL     COD31           GET THE SECOND INSTRUCTION                 FT089480
         SUB     =1              BUMP THE CONTAB POINTER DOWN BY ONE        FT089490
         TSX     CITBLD,4,C      PUT OUT THE CODE                           FT089500
         REM                                                                FT089510
EXCODE   AXT     **,4                                                       FT089520
         AXT     **,1                                                       FT089530
         TRA     2,4             RETURN                                     FT089540
         REM                                                                FT089550
         REM                                                                FT089560
COD7     STA     COD4            PROLOGUE REQUIRED.  SAVE POINTER           FT089570
         ANA     =077777         SAVE ONLY ADDRESS                          FT089580
         ZSD     COD40           NO CONSTANT ADDEND HERE                    FT089590
         TRA     COD42                                                      FT089600
         REM                                                                FT089610
         REM                                                                FT089620
         REM                                                                FT089630
         REM             PROCESS SUBSCRIPTING FOR THIS VARIABLE             FT089640
COD5     PAX     ,1              GET ISTACK POINTER                         FT089650
         CLA     POOL-2,1        ISTACK ENTRY                               FT089660
         ARS     18              GET POINTER                                FT089670
         TMI     COD8            POINTER IS TO SYMTAB                       FT089680
         LDQ     POOL,1          GET THE FIRST ISTACK WORD                  FT089690
         TQP     COD10           SKIP IF PLUS SIGN (ONLY CONSTANT ADDEND)   FT089700
         REM                                                                FT089710
         ORA     LXD4            OR IN THE OP CODE                          FT089720
         ORS     XRFLAG                                                     FT089730
         TSX     FREWRK,4        FREE UP THE WORKING CELL                   FT089740
         TSX     CITBLD,4,W      CODE IS...  LXD WORKING,4                  FT089750
         REM                                                                FT089760
         STL     COD6            SET FLAG TO 'LOAD XR' STATUS               FT089770
COD10    CLA     POOL-2,1        GET THE CONSTANT ADDEND FROM ISTACK ENTRY  FT089780
         PAX     ,4                                                         FT089790
         SXD     COD37,4         SET THE 'BUMPING' INSTRUCTIONS             FT089800
         SXD     COD40,4                                                    FT089810
         REM                                                                FT089820
         LDI     POOL-1,1        GET ASTACK WORD                            FT089830
         LFT     BARGT                                                      FT089840
```

```
         TRA     COD11    SUBPROGRAM PARAMETER                          FT089850
         REM                                                            FT089860
         PXA     ,4                                                     FT089870
         TSX     CATEST,4 GO PROCESS THE CONSTANT ADDEND                FT089880
         REM                                                            FT089890
         CAL     POOL-1,1 SECOND ISTACK ENTRY                           FT089900
         PAI                                                            FT089910
         ANA     =077777  SAVE POINTER                                  FT089920
         STA     COD31    PUT INTO SECOND INSTRUCTION                   FT089930
         ORA*    COD1     OR IN THE OP CODE                             FT089940
COD38    ZET     COD6     SKIP IF NO XR WAS LOADED                      FT089950
         ORA     =4B20    TAG OF 4                                      FT089960
         ORS     XRFLAG                                                 FT089970
COD47    ***     **       INSTRUCTION SET BY CATEST ROUTINE             FT089980
         NZT     SPCDFG   TEST THE ROUTING FLAG                         FT089990
         TRA     EXCODE   NO MORE CODE NEEDED HERE                      FT090000
         REM                                                            FT090010
         CAL     =077777  ADDEND OF MINUS ONE                           FT090020
         LNT     BARRY    SKIP IF NOT AN ARRAY                          FT090030
         TRA     COD37+2                                                FT090040
         REM                                                            FT090050
         LXA     COD31,4  GET THE EQUIV POINTER                         FT090060
         CAL     EQUIV,4  AND GET THE EQUIV WORD                        FT090070
         PAX     ,4       AND GET THE DIMTBL POINTER                    FT090080
         CAL     POOL-1,4 AND GET THE ARRAY LENGTH                      FT090090
         PAC     ,4       COMPLEMENTED                                  FT090100
COD37    TXI     **+1,4,** BUMP BY A POSSIBLE CONSTANT ADDEND           FT090110
         PXA     ,4       PUT TOTAL ADDEND INTO ACCUM                   FT090120
         TSX     CATEST,4 GO PROCESS THE CONSTANT ADDEND                FT090130
         REM                                                            FT090140
         CAL     COD31    GET THE SECOND INSTRUCTION                    FT090150
         STZ     SPCDFG   RESET THE ROUTING FLAG                        FT090160
         TRA     COD38    GO PUT OUT THE CODE AGAIN                     FT090170
         REM                                                            FT090180
COD8     PAX     ,4       GET EQUIV WORD                                FT090190
         ORA     LXD4     OR IN THE OPERATION CODE                      FT090200
         ORS     XRFLAG                                                 FT090210
         LDI     EQUIV,4                                                FT090220
         LFT     BARGT    IS IT AN ARGUMENT.                            FT090230
         TRA     COD9     YES                                           FT090240
         REM                                                            FT090250
         TSX     CITBLD,4,V CODE IS...  LXD VARIABLE,4                  FT090260
         TRA     COD10-1                                                FT090270
         REM                                                            FT090280
COD9     TSX     PROLOG,4 PUT OUT PROLOGUE                              FT090290
         CAL     LXD4     GET THE OPERATION CODE                        FT090300
         TSX     CITBLD,4,N CODE IS...  LXD **,4                        FT090310
         TRA     COD10-1                                                FT090320
         REM                                                            FT090330
COD12    TSX     CITBLD,4,T CODE IS... (OP) TRANSFER.VECTOR             FT090340
         TRA     EXCODE                                                 FT090350
         REM                                                            FT090360
COD11    CAL     POOL-1,1 GET ORIGINAL ASTACK WORD                      FT090370
         STA     COD4                                                   FT090380
         PXD     ,4       CONSTANT ADDEND                               FT090390
         ORA     COD4     EQUIV POINTER                                 FT090400
COD42    TSX     PROLGS,4 PUT OUT PROLOGUE AND ADDEND                   FT090410
         REM                                                            FT090420
         CAL*    COD1     GET OPERATION CODE                            FT090430
COD41    ZET     COD6     SKIP IF NO XR WAS LOADED                      FT090440
```

```
         ORA     =4B20           TAG OF 4                                        FT090450
         ORS     XRFLAG                                                          FT090460
         TSX     CITBLD,4,N      CODE IS...  (OP) **                            FT090470
         NZT     SPCDFG          TEST THE ROUTING FLAG                           FT090480
         TRA     EXCODE          NO MORE WORK TO DO HERE                         FT090490
         REM                                                                     FT090500
         CAL     =077777000000   DECREMENT ADDEND OF -1                         FT090510
         LNT     BARRY           SKIP IF NOT AN ARRAY                            FT090520
         TRA     COD40+2                                                         FT090530
         LXA     COD4,4          GET THE EQUIV POINTER                           FT090540
         CAL     EQUIV,4         AND GET THE EQUIV WORD                          FT090550
         PAX     ,4              AND GET THE DIMTBL POINTER                      FT090560
         CAL     POOL-1,4        AND GET THE ARRAY LENGTH                        FT090570
         PAC     ,4              COMPLEMENTED                                    FT090580
         REM                                                                     FT090590
COD40    TXI     **+1,4,**       BUMP BY A POSSIBLE CONSTANT ADDEND              FT090600
         PXD     ,4                                                              FT090610
         ORA     COD4            GET THE EQUIV POINTER                           FT090620
         TSX     PROLGS,4        GO PUT OUT THE PROLOGUE                         FT090630
         REM                                                                     FT090640
         CAL     COD31           GET THE SECOND INSTRUCTION                      FT090650
         STZ     SPCDFG          RESET THE ROUTING FLAG                          FT090660
         TRA     COD41                                                           FT090670
         REM                                                                     FT090680
COD14    STA     COD31           SAVE ARGUMENT POINTER IN SECOND INSTRUCTION     FT090690
         ORA     =060400000      ADD IN XR4 AND INDIRECT ADDRESS BITS            FT090700
         TSX     CITBLD,4,N      PUT OUT THE CODE                                FT090710
         REM                                                                     FT090720
         NZT     SPCDFG          IS THERE A SECOND INSTRUCTION.                  FT090730
         TRA     EXCODE          NO                                              FT090740
         CAL     COD31           YES.  GET THE INSTRUCTION                       FT090750
         ACL     =060400001      ADD IN XR4, INDIRECT BITS, AND BUMP ADDRESS     FT090760
         TSX     CITBLD,4,N      PUT OUT THE SECOND INSTRUCTION                  FT090770
         TRA     EXCODE                                                          FT090780
         SPACE   3                                                               FT090790
         REM     ROUTINE TO PUT OUT A CONSTANT ADDEND FOR A VARIABLE.            FT090800
         REM     USES SPECIAL CITBLD FLAGS 'D' FOR CA OF +1 AND 'G' FOR          FT090810
         REM     CA OF -1.  ANY OTHER NON-ZERO CA IS GIVEN TO CITBLN             FT090820
         REM     WITH FLAG 'Q'.                                                  FT090830
         REM                                                                     FT090840
CATEST   SXA     EXCATS,4                                                        FT090850
         PAX     ,4              GET THE CA                                      FT090860
         TXL     EXCATS-2,4,0    FOR NO CONSTANT ADDEND                          FT090870
         LDQ     CITXD                                                           FT090880
         TXL     EXCATS-1,4,1    FOR CA OF +1.                                   FT090890
         LDQ     CITXG                                                           FT090900
         TXH     EXCATS-1,4,-2   FOR CA OF -1.                                   FT090910
         TSX     CITBLN,4,Q      FOR OTHER CA'S.  GIVE THE CA TO CITBLN          FT090920
         LDQ     CITXV           INSTRUCTION FOR THIS CASE                       FT090930
         REM                                                                     FT090940
         STQ     COD47           SET THE INSTRUCTION FOR THE CODE ROUTINE        FT090950
EXCATS   AXT     **,4                                                            FT090960
         TRA     1,4                                                             FT090970
         REM                                                                     FT090980
CITXV    TSX     CITBLD,4,V      STRAIGHT VARIABLE RELOCATION                    FT090990
CITXD    TSX     CITBLD,4,D      VARIABLE RELOCATION AND CA OF +1.               FT091000
CITXG    TSX     CITBLD,4,G      VARIABLE RELOCATION AND CA OF -1.               FT091010
```

```
*       ROUTINE TO RESTORE CELL 2                                       FT091030
        REM                                                             FT091040
PUTCL2  CLA     SAVCL2          GET THE CELL-2 FLAG                     FT091050
        TPL     1,4             CELL 2 IS NOT NOW SAVED                 FT091060
        SXA     EXPUT2,4                                                FT091070
        TSX     FREWRK,4        FREE UP THE WORKING CELL                FT091080
        CODEO   LDQ,W                                                   FT091090
        CODEN   STQ2            RESTORE CELL 2                          FT091100
        STZ     SAVCL2          RESET THE POINTER                       FT091110
        STZ     SETUP2          RESET THE ROUTINE NAME CELL             FT091120
        REM                                                             FT091130
EXPUT2  AXT     **,4                                                    FT091140
        TRA     1,4                                                     FT091150
        SPACE   6                                                       FT091160
*       ROUTINE TO ESTABLISH OBJECT-TIME LINKAGES TO DP AND CA ROUTINES FT091170
        REM                                                             FT091180
SETUP   SXA     EXSETP,4                                                FT091190
        CAL     SAVCL2          SKIP IF CELL 2 IS STILL SAVED           FT091200
        TNZ     SETUP1                                                  FT091210
        REM                                                             FT091220
        CODEN   LDQ2            GET CELL 2                              FT091230
        TSX     GETWRK,4        GET A WORKING CELL                      FT091240
        SSM                     SET SIGN MINUS AND SAVE WRKCEL POINTER  FT091250
        STO     SAVCL2                                                  FT091260
        CODEO   STQ,W                                                   FT091270
        REM                                                             FT091280
SETUP1  LXA     EXSETP,4                                                FT091290
        CAL     1,4             GET THE ROUTINE TO BE LOADED            FT091300
        LAS     SETUP2          IS IT SAME AS ONE ALREADY LOADED.       FT091310
        TRA     *+2             NO                                      FT091320
        TRA     EXSETP+1        YES.  ALL DONE                          FT091330
        REM                                                             FT091340
        SLW     SETUP2          SAVE NAME OF ROUTINE IN CELL            FT091350
        STL*    1,4             MARK USAGE IN SPECIAL NAME TABLE        FT091360
        ORA     LDQ                                                     FT091370
        TSX     CITBLD,4,S      GET THE TRANSFER VECTOR ENTRY           FT091380
        CODEN   STQ2            PUT INTO CELL 2                         FT091390
        REM                                                             FT091400
EXSETP  AXT     **,4                                                    FT091410
        TRA     2,4                                                     FT091420
        SPACE   6                                                       FT091430
*       ROUTINE TO PUT PSEUDO-AC INTO PSEUDO-MQ                         FT091440
        REM                                                             FT091450
AC.MQ   SXA     EXACMQ,4                                                FT091460
        ZET     REGFLG          SKIP IF AC IS ALREADY IN THE AC+MQ      FT091470
        TRA     AC.MQ1                                                  FT091480
        REM                                                             FT091490
        CODEN   CLA777                                                  FT091500
        CODEN   LDQ776                                                  FT091510
AC.MQ1  CODEN   STO775                                                  FT091520
        CODEN   STQ774                                                  FT091530
        REM                                                             FT091540
EXACMQ  AXT     **,4                                                    FT091550
        TRA     1,4                                                     FT091560
        SPACE   6                                                       FT091570
*       ROUTINE TO PUT PSEUDO-MQ INTO PSEUDO-AC                         FT091580
        REM                                                             FT091590
MQ.AC   SXA     EXACMQ,4                                                FT091600
        ZET     REGFLG          SKIP IF PSEUDO-MQ IS ALREADY IN AC+MQ   FT091610
```

```
          TRA     MQ.AC1                                                    FT091620
          REM                                                               FT091630
          CODEN   CLA775                                                    FT091640
          CODEN   LDQ774                                                    FT091650
MQ.AC1    CODEN   STO777                                                    FT091660
          CODEN   STQ776                                                    FT091670
          TRA     EXACMQ                                                    FT091680
          SPACE   6                                                         FT091690
*         ROUTINE TO LOAD THE PSEUDO-AC FROM STORAGE                        FT091700
          REM                                                               FT091710
CODLAC    SXA     EXCODE+1,1      SAVE XR1 IN CODE ROUTINE                  FT091720
          AXC     CDLAC1-2,1      GET RETURN POINT FROM CODE ROUTINE        FT091730
          REM                                                               FT091740
CDLAC2    CAL     LDQ             GET OP CODE FOR SECOND INSTRUCTION        FT091750
          SLW     COD31           SAVE IN CODE CELL                         FT091760
          SXA     EXCODE,1        SET RETURN POINT IN CODE ROUTINE EXIT SEQ. FT091770
          SXA     EXSPCD,4        SAVE XR4                                  FT091780
          STL     SPCDFG          SET THE SPECIAL CODE ROUTING FLAG         FT091790
          TRA     COD32           GO TO CODE ROUTINE                        FT091800
          REM                                                               FT091810
          SPACE   6                                                         FT091820
*         ROUTINE TO PUT OUT TWO STZ INSTRUCTIONS                           FT091830
          REM                                                               FT091840
CODSTZ    CAL     STZ                                                       FT091850
          TRA     CODSTR+1                                                  FT091860
          SPACE   6                                                         FT091870
*         ROUTINE TO LOAD THE PSEUDO-MQ FROM STORAGE                        FT091880
          REM                                                               FT091890
CODLMQ    SXA     EXCODE+1,1                                                FT091900
          AXC     CDLMQ1-2,1                                                FT091910
          TRA     CDLAC2                                                    FT091920
          SPACE   6                                                         FT091930
*         ROUTINE TO PUT NEGATIVE OF STORAGE INTO PSEUDO-AC                 FT091940
          REM                                                               FT091950
CODSUB    SXA     EXCODE+1,1                                                FT091960
          AXC     CDSUB1-2,1                                                FT091970
          TRA     CDLAC2                                                    FT091980
          SPACE   6                                                         FT091990
*         ROUTINE TO OUTPUT THE STR AND PZE INSTRUCTIONS                    FT092000
          REM                                                               FT092010
CODSTR    CAL     PZE             GET SECOND INSTRUCTION                    FT092020
          SLW     COD31           SAVE SECOND INSTRUCTION FOR CODE ROUTINE  FT092030
          STL     SPCDFG                                                    FT092040
          TRA     CODE+1                                                    FT092050
          SPACE   6                                                         FT092060
*         ROUTINE TO LOAD THE AC+MQ FROM STORAGE                            FT092070
          REM                                                               FT092080
CODLD     CAL     LDQ                                                       FT092090
          TRA     CODSTR+1                                                  FT092100
          SPACE   6                                                         FT092110
*         ROUTINE TO STORE THE AC+MQ IN VARIABLE STORAGE                    FT092120
          REM                                                               FT092130
CODSTO    CAL     STQ                                                       FT092140
          TRA     CODSTR+1                                                  FT092150
          SPACE   6                                                         FT092160
*         RE-ENTRY POINTS FROM THE CODE ROUTINE                             FT092170
CDSUB1    CODEN   LRS             RE-ENTRY FROM CODE ROUTINE                FT092180
          REM                                                               FT092190
CDLAC1    CODEN   STO777          RE-ENTRY FROM CODE ROUTINE                FT092200
          CODEN   STQ776                                                    FT092210
```

```
        REM                                                          FT092220
EXSPCD  AXT     **,4                                                 FT092230
        TRA     2,4                                                  FT092240
        REM                                                          FT092250
CDLMQ1  CODEN   STO775      RE-ENTRY FROM CODE ROUTINE               FT092260
        CODEN   STQ774                                               FT092270
        TRA     EXSPCD                                               FT092280
```

```
        TTL     A R I T H M E T I C   U T I L I T Y  -  R E S U L T     FT092290
*       RESULT ROUTINE.                                                 FT092300
*       ROUTINE TO RECORD REGISTER LOCATION AND MODE OF A CURRENT       FT092310
*       RESULT.  THIS ROUTINE IS CALLED BY THE CODE GENERATORS.         FT092320
        REM                                                             FT092330
*       TYPICAL CALL IS VIA THE 'RESULT' MACRO...                       FT092340
*               RESULT  REAL,AC                                         FT092350
        REM                                                             FT092360
*       THE MODE OF THE RESULT IS PUT INTO ASTACK AT THE CURRENT        FT092370
*       ASTACK LOCATION (XR 1).  REGIST IS SET TO THE REGISTER LOCATION FT092380
*       (1 FOR AC, AND 2 FOR MQ... 0 INDICATES EITHER NO RESULT,        FT092390
*       OR A RESULT IN STORAGE).  REGIST+1 GETS THE ASTACK POINTER      FT092400
*       OF THIS RESULT.                                                 FT092410
        REM                                                             FT092420
*       IF THE STATEMENT IS DOUBLE PRECISION OR COMPLEX, 'AC' AND 'MQ'  FT092430
*       REFER TO THE PSEUDO-REGISTERS IN HIGH CORE.  IF THE ACTUAL      FT092440
*       AC+MQ ALSO CONTAIN VALID PSEUDO-REGISTER RESULTS, THE FLAGS     FT092450
*       'PAC' OR 'PMQ' ARE USED (IN THE RESULT MACRO).  IN THIS CASE,   FT092460
*       REGFLG IS SET NON-ZERO.                                         FT092470
        SPACE   3                                                       FT092480
*       ENTRY FOR FLOATING ROUND OPTION.                                FT092490
*       ROUNDX IS CALLED VIA THE RESULT MACRO WHEN THE MACRO HAS        FT092500
*       THE THIRD ARGUMENT = ROUND.  IF THE FLOATING ROUND OPTION       FT092510
*       IS IN EFFECT (ROUNDF IS NON-ZERO), THE CODE 'FRN' IS PUT OUT    FT092520
*       BEFORE ENTERING THE NORMAL RESULT SECTION.                      FT092530
        REM                                                             FT092540
ROUNDX  NZT     ROUNDF          IS FLOATING ROUND OPTION IN EFFECT.     FT092550
        TRA     RESULT          NO.  SKIP THIS CODE                     FT092560
        SXA     *+3,4           SAVE XR4                                FT092570
        CAL     FRN                                                     FT092580
        TSX     CITBLD,4,N      CODE IS...  FRN                         FT092590
        AXT     **,4                                                    FT092600
        REM                                                             FT092610
        REM                                                             FT092620
*       NORMAL RESULT ENTRY.                                            FT092630
        REM                                                             FT092640
RESULT  CAL     ,4              GET CALLING INSTRUCTION                 FT092650
        ARS     18              DECREMENT TO ADDRESS                    FT092660
        ANA     =03             SAVE LOCATION CODE FOR RESULT           FT092670
        STO     REGIST                                                  FT092680
        SXA     REGIST+1,1                                              FT092690
        REM                                                             FT092700
        LDI     ,4              GET THE CALLING INSTRUCTION             FT092710
        STZ     REGFLG          INITIALIZE REGFLG                       FT092720
        LFT     4               IF AC AND MQ CONTAIN VALID RESULTS,     FT092730
        STL     REGFLG          IN ADDITION TO A PSEUDO-REG, SET THE FLAGFT092740
        REM                                                             FT092750
        CAL     ,4                                                      FT092760
        ANA     =070000000      SAVE THE MODE INDICATION OF THE RESULT  FT092770
        ALS     12              POSITION THE MODE BITS INTO PREFIX      FT092780
        SLW     ASTACK,1        MODIFY ASTACK                           FT092790
        REM                                                             FT092800
        TRA     GENXIT          RETURN TO GEN ROUTINE                   FT092810
```

```
         TTL     A R I T H M E T I C  U T I L I T Y  -  S T O R E          FT092820
*        ROUTINE TO STORE THE ACCUMULATOR OR MQ IN TEMPORARY STORAGE.      FT092830
*        PUTS NUMBER OF WORKING CELL INTO ADDRESS OF CURRENT ASTACK WORD,  FT092840
*        WITH A 1-BIT IN THE HIGH ORDER POSITION OF THE ADDRESS.           FT092850
         REM                                                               FT092860
STORE    NZT     REGIST          IS IT ALREADY IN STORAGE.                 FT092870
         TRA     1,4             YES                                       FT092880
         STZ     AUXFLG          SET FLAG TO 'STORE' MODE                  FT092890
         SXA     EXSTOR,4                                                  FT092900
         LDQ     =0              ZERO THE MQ                               FT092910
         CAL     REGIST                                                    FT092920
         LGR     2               REGISTER OF THIS STORABLE WORD            FT092930
         LXA     REGIST+1,4      GET ASTACK POINTER                        FT092940
         CAL     ASTACK,4        GET ASTACK WORD                           FT092950
         ARS     33              SHIFT MODE INTO POSITION                  FT092960
         LGR     3                                                         FT092970
         CAL     AUXILX          GET 'OP' CODE FOR STORE OPERATION         FT092980
         LGR     12              ZERO A-OPERAND BITS AND MOVE IN THE OP CODEFT092990
         STQ     GEN6            SET THE KEY WORD IN ITS CELL              FT093000
         TRA     GEN5            GO NOW TO THE GEN ROUTINE                 FT093010
```

```
         TTL     A R I T H M E T I C  U T I L I T Y  -  P L A C E        FT093020
*        ROUTINE TO PLACE CURRENT ASTACK WORD INTO THE ACCUMULATOR        FT093030
*        USED BY IF STATEMENT PROCESSOR ONLY.                             FT093040
         REM                                                              FT093050
PLACE    STL     AUXFLG          SET FLAG TO 'PLACE' MODE                 FT093060
         STL     AUXFG1          FLAG TO THE GEN ROUTINE                  FT093070
         TRA     GEN+1           GO FIND PROPER CODE SEQUENCE             FT093080
         REM                                                              FT093090
```

```
*       CODE GENERATORS FOR THE STORE AND PLACE ROUTINES                FT093110
        REM                                                             FT093120
AUXIL1  CODE    CLA,B                                                   FT093130
        TRA     GENXIT                                                  FT093140
        REM                                                             FT093150
AUXIL2  NZT     AUXFLG                                                  FT093160
        TRA     AUXIL9          FOR STORE MODE                          FT093170
        CODEN   XCA                                                     FT093180
        TRA     GENXIT                                                  FT093190
        REM                                                             FT093200
AUXIL3  CODE    CAL,B           BOOLEAN                                 FT093210
        TRA     GENXIT                                                  FT093220
        REM                                                             FT093230
AUXIL5  STZ     AUXFG1          SET FLAG TO 'MQ' MODE                   FT093240
        TRA     AUXIL4+1                                                FT093250
        REM                                                             FT093260
AUXIL4  STL     AUXFG1          SET FLAG TO 'AC' MODE                   FT093270
        NZT     AUXFLG                                                  FT093280
        TRA     AUXIL8          FOR STORE MODE                          FT093290
        ZET     REGFLG                                                  FT093300
        TRA     GENXIT          RESULT ALREADY IN AC                    FT093310
        REM                                                             FT093320
        CAL     CLA777          GET CODE                                FT093330
        NZT     AUXFG1                                                  FT093340
        CAL     CLA775          GET CODE FOR MQ LOAD                    FT093350
        TSX     CITBLD,4,N      CODE TO LOAD THE ACCUM                  FT093360
        TRA     GENXIT                                                  FT093370
        REM                                                             FT093380
AUXIL6  ZET     AUXFLG                                                  FT093390
        TRA     GENXIT          RESULT ALREADY IN ACCUM                 FT093400
        TSX     PREPWK,4        GO PREPARE THE WORKING CELL             FT093410
        CODEO   STO,W           STORE THE ACCUMULATOR                   FT093420
EXSTOR  AXT     **,4                                                    FT093430
        TRA     1,4                                                     FT093440
        REM                                                             FT093450
AUXIL7  ZET     AUXFLG                                                  FT093460
        TRA     GENXIT          RESULT ALREADY IN ACCUM                 FT093470
        TSX     PREPWK,4        GO PREPARE THE WORKING CELL             FT093480
        CODEO   SLW,W           STORE OFF THE BOOLEAN ACCUMULATOR       FT093490
        TRA     EXSTOR                                                  FT093500
        REM                                                             FT093510
AUXIL9  TSX     PREPWK,4        GO PREPARE THE WORKING CELL             FT093520
        CODEO   STQ,W           STORE OFF THE MQ                        FT093530
        TRA     EXSTOR                                                  FT093540
        REM                                                             FT093550
AUXIL8  ZET     REGFLG                                                  FT093560
        TRA     AUXI10          SKIP IF ACCUM AND MQ ALREADY HAVE DATA  FT093570
        REM                                                             FT093580
        CAL     CLA775                                                  FT093590
        ZET     AUXFG1                                                  FT093600
        ACL     =2              MODIFY INSTRUCTION FOR THE PSEUDO-AC    FT093610
        TSX     CITBLD,4,N      PUT OUT CODE TO LOAD THE ACCUM          FT093620
        CAL     LDQ774                                                  FT093630
        ZET     AUXFG1                                                  FT093640
        ACL     =2                                                      FT093650
        TSX     CITBLD,4,N      PUT OUT CODE TO LOAD THE MQ             FT093660
        REM                                                             FT093670
AUXI10  TSX     PREPWK,4,2      PREPARE TWO WORKING CELLS               FT093680
        STA     AUXI11          SAVE THE WORKING CELL POINTER           FT093690
```

```
         CODEO   STO,W       STORE OFF THE ACCUMULATOR                    FT093700
         CAL     AUXI11      REGAIN THE WORKING CELL POINTER              FT093710
         ADD     =1          BUMP THE POINTER                             FT093720
         TSX     CITBLD,4,W  STORE OFF THE MQ                             FT093730
         TRA     EXSTOR                                                   FT093740
         REM                                                              FT093750
AUXI11   STQ     **          CODE WORD FOR STORE ROUTINE                  FT093760
         SPACE   4                                                        FT093770
PREPWK   SXA     EXPREP,4    ROUTINE TO PREPARE A WORKING CELL            FT093780
         CAL     ,4          GET CALLING INSTRUCTION                      FT093790
         STD     *+1         PUT NUMBER OF WORDS WANTED INTO INSTRUCTIONFT093800
         TSX     GETWRK,4,** GO GET THE WORKING CELLS                     FT093810
         ORA     =1B21       ADD IN A HIGH-ORDER BIT FOR ASTACK           FT093820
         LXA     REGIST+1,4  GET ASTACK POINTER                           FT093830
         STA     ASTACK,4    MODIFY ASTACK                                FT093840
         ANA     =037777     KILL THE HIGH-ORDER BIT                      FT093850
         STZ     REGIST      RE-INITIALIZE THE REGISTER CELLS             FT093860
         STZ     REGIST+1                                                 FT093870
         STZ     REGFLG                                                   FT093880
EXPREP   AXT     **,4                                                     FT093890
         TRA     1,4                                                      FT093900
```

```
*       ROUTINE TO OBTAIN ONE OR TWO WORKING CELLS FROM THE WORKING      FT093920
*       POOL.  CALLING SEQUENCE IS...                                    FT093930
*                              TSX GETWRK,4,XX                           FT093940
*       WHERE  XX  IS 0 IF ONE CELL IS WANTED, AND 2 IF TWO              FT093950
*       CONSECUTIVE CELLS ARE WANTED.                                    FT093960
*       ADDRESS OF LOGICAL ACCUMULATOR WILL CONTAIN THE POINTER          FT093970
*       TO THE FIRST OF THESE CELLS.                                     FT093980
*                                                                        FT093990
*       AT PRESENT, ONLY 36 CELLS ARE AVAILABLE.                         FT094000
        SPACE   2                                                        FT094010
GETWRK  SXA     EXWRK,4                                                  FT094020
        CAL     ,4              GET THE CALLING INSTRUCTION              FT094030
        ANA     =07000000       SAVE ONLY THE LOWER DECREMENT            FT094040
        SLW     GETWR7                                                   FT094050
        AXT     GETWR3,4        PREPARE TO SET UP TRANSFER ADDRESS       FT094060
        TZE     *+2                                                      FT094070
        AXT     GETWR4,4        TWO CELLS WANTED                         FT094080
        SXA     GETWR5,4                                                 FT094090
        REM                                                              FT094100
        CAL     WRKCEL          GET WORKING CELL KEY WORD                FT094110
        SSM                                                              FT094120
        AXT     1,4             INITIALIZE XR4                           FT094130
        LGR     1               MOVE NEXT BIT INTO MQ SIGN               FT094140
GETWR5  TQP     **              ADDRESS SET FROM ABOVE                   FT094150
        TXI     *+1,4,1         BUMP COUNTER                             FT094160
GETWR2  TXL     *-3,4,36        CONTINUE THE SEARCH IF NOT OUT OF BITS   FT094170
        STR     MSG100,,SKEND       ERROR MESSAGE                        FT094180
        REM                                                              FT094190
GETWR4  LBT                     IS AN ADJACENT SECOND BIT AVAILABLE.     FT094200
        TRA     *+2             YES.  GOODIE                             FT094210
        TXI     GETWR2,4,1      NO.  RETURN TO LOOK AGAIN                FT094220
        ORA     =1              TURN ON THE BIT                          FT094230
        REM                                                              FT094240
GETWR3  LRS     0               TURN ON THE BIT TO MARK IT IN USE        FT094250
        SXA     *+1,4           PUT SHIFT COUNT INTO INSTRUCTION         FT094260
        LGL     **              RE-ADJUST THE KEY WORD                   FT094270
        SLW     WRKCEL          AND RESTORE IT                           FT094280
        REM                                                              FT094290
GETWR1  TXI     *+1,4,**        BUMP BY LENGTH OF CELLS RESERVED FOR     FT094300
        REM                     ARITH. STATE. FUNCTIONS.                 FT094310
        ZET     GETWR7                                                   FT094320
        TXI     *+1,4,1         BUMP LENGTH IF TWO CELLS OBTAINED        FT094330
WORKS   TXL     *+3,4,**        SHOULD LWORKS BE BUMPED.                 FT094340
        SXD     *-1,4           YES.  UPDATE THE TEST INSTRUCTION        FT094350
        SXA     LWORKS,4        AND UPDATE LWORKS                        FT094360
        REM                                                              FT094370
        ZET     GETWR7                                                   FT094380
        TXI     *+2,4,-2        ADJUST POINTER FOR TWO-CELL CASE         FT094390
        TXI     *+1,4,-1        ADJUST POINTER FOR THE ONE-CELL CASE     FT094400
        PXA     ,4              PUT POINTER INTO ADDRESS OF ACCUM        FT094410
        REM                                                              FT094420
EXWRK   AXT     **,4                                                     FT094430
        TRA     1,4                                                      FT094440
```

```
        TTL A R I T H M E T I C  U T I L I T Y  P R O G.  -  F R E W R K FT094450
*       ROUTINE TO RELEASE ONE WORKING CELL FROM SERVICE.                 FT094460
*       CALLING SEQUENCE IS...                                            FT094470
*                                CAL  (NUMBER OF WORKING CELL)            FT094480
*                                TSX  FREWRK,4                            FT094490
        SPACE   2                                                         FT094500
FREWRK  SXA     EXWRK,4                                                   FT094510
        SLW     GETWR7          SAVE POINTER                              FT094520
        PAC     ,4                                                        FT094530
FREWR1  TXI     *+1,4,**        BUMP BY NUMBER OF CELLS RESERVED BY       FT094540
        REM                     ARITH. STATE. FUNCTIONS.                  FT094550
        CAL     WRKCEL          GET KEY WORD                              FT094560
        LGR     1,4             SHIFT KEY WORD                            FT094570
        LRS     0               KILL THE APPROPRIATE BIT                  FT094580
        LGL     1,4             SHIFT BACK INTO PROPER POSITION           FT094590
        SLW     WRKCEL          RESTORE THE KEY WORD                      FT094600
        CAL     GETWR7          RESTORE THE ACCUMULATOR                   FT094610
        TRA     EXWRK                                                     FT094620
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  A S C O N 1 FT094630
*       CONSTANT STASHING ROUTINES...  ASCON1 AND ASCON2                 FT094640
        REM                                                             FT094650
*       ROUTINES TO ENTER ONE- AND TWO-WORD CONSTANTS INTO THE CONSTANT  FT094660
*       TABLE.                                                          FT094670
        REM                                                             FT094680
*       ENTRY FOR ONE-WORD CONSTANT...                                  FT094690
*               CAL   CONSTANT                                          FT094700
*               TSX   ASCON1,4                                          FT094710
        REM                                                             FT094720
*       ENTRY FOR TWO-WORD CONSTANT                                     FT094730
*               CAL   FIRST.PART                                        FT094740
*               LDQ   SECOND.PART                                       FT094750
*               TSX   ASCON2,4                                          FT094760
        REM                                                             FT094770
*       ON RETURN FROM THIS ROUTINE, THE ADDRESS OF THE ACCUMULATOR     FT094780
*       WILL POINT TO THE POSITION OF THE CONSTANT IN THE CONSTANT TABLE.FT094790
*       IN THE CASE OF TWO-WORD CONSTANTS, THE POINTER POINTS TO THE    FT094800
*       MOST SIGNIFICANT PART (I.E. THE PART WHICH IS HIGHEST IN CORE   FT094810
*       IN THE OBJECT PROGRAM).                                         FT094820
        REM                                                             FT094830
*       CONTAB GOES BACKWARD IN MEMORY, WITH INDEXING STARTING AT ZERO. FT094840
*       NOCONT = NUMBER OF CONSTANTS CURRENTLY IN THE TABLE.            FT094850
*       NOCNST = MAXIMUM ALLOWABLE LENGTH OF CONTAB.                    FT094860
        REM                                                             FT094870
*       IF CONSTANT TABLE OVERFLOW OCCURS, THE CURRENT LENGTH (NOCONT)  FT094880
*       IS RESET TO ZERO, AND AN ERROR MESSAGE ISSUED.                  FT094890
        SPACE   4                                                       FT094900
ASCON1  SXA     ASCX,4                                                  FT094910
        LXA     NOCONT,4                                                FT094920
        TXL     ASC2,4,0        TABLE EMPTY                             FT094930
        LAS     CONTAB+1,4                                              FT094940
        TRA     *+2                                                     FT094950
        TXI     ASCX-1,4,-1     CONSTANT FOUND.  ADJUST POINTER         FT094960
        TIX     *-3,4,1                                                 FT094970
        LXA     NOCONT,4        NOT IN TABLE. ADD IT.                   FT094980
ASC2    TXI     *+1,4,1                                                 FT094990
        TXH     ASC1,4,NOCNST                                           FT095000
ASC3    SLW     CONTAB+1,4      TABLE NOT FULL                          FT095010
        SXA     NOCONT,4                                                FT095020
        TXI     *+1,4,-1                                                FT095030
        PXA     ,4                                                      FT095040
ASCX    AXT     **,4                                                    FT095050
        TRA     1,4                                                     FT095060
ASC1    AXT     1,4             CONSTANT TABLE FULL.  RESET IT.         FT095070
        STR     MSG100,,ASC2                                            FT095080
        REM                     ENTRY FOR 2 WORD CONSTANTS              FT095090
ASCON2  SXA     ASCX,4                                                  FT095100
        XCL                                                             FT095110
        LXA     NOCONT,4                                                FT095120
        TRA     ASC5                                                    FT095130
        LAS     CONTAB+1,4                                              FT095140
        TRA     *+2                                                     FT095150
        TRA     ASC4            FIRST PART FOUND                        FT095160
ASC5    TIX     *-3,4,1                                                 FT095170
        LXA     NOCONT,4        NOT IN                                  FT095180
ASC6    TXI     *+1,4,2                                                 FT095190
        TXH     ASC1,4,NOCNST   TABLE FULL                              FT095200
        SXA     NOCONT,4                                                FT095210
        SLW     CONTAB+2,4                                              FT095220
```

```
        STQ     CONTAB+1,4                                      FT095230
        TXI     ASCX-1,4,-1     ADJUST POINTER AND RETURN       FT095240
ASC4    XCL                                                     FT095250
        LAS     CONTAB,4                                        FT095260
        TRA     *+2                                             FT095270
        TRA     ASCX-1          RETURN                          FT095280
        XCL                                                     FT095290
        TRA     ASC5                                            FT095300
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  N E X T L P FT095310
*       ROUTINE TO GET NEXT LOCATION IN POOL                             FT095320
        SPACE   3                                                        FT095330
NEXTLP  CAL     NXTLOC                                                   FT095340
        ADD     =1                                                       FT095350
        SLW     NXTLOC                                                   FT095360
        ACL     *+2                                                      FT095370
        TRA     1,4                                                      FT095380
        REM                                                              FT095390
        PZE     LSYMTB                                                   FT095400
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  P R O L O G FT095410
*       THIS SUBROUTINE ENTERS A WORD INTO THE PROLOGUE TABLE            FT095420
*       (P R L T B L).  ENTRIES INTO THIS TABLE ARE MADE DURING PASS 1.  FT095430
        SPACE   2                                                        FT095440
*         CALLING SEQUENCE,                                              FT095450
        REM                                                              FT095460
*                       .                                                FT095470
*                       .                                                FT095480
*                       .                                                FT095490
*             CAL     (ARGUMENT)                                         FT095500
*             TSX     PROLGS,4                                           FT095510
*                       .                                                FT095520
*                       .                                                FT095530
*                       .                                                FT095540
        REM                                                              FT095550
        REM                                                              FT095560
*           WHERE,                                                       FT095570
*              ARGUMENT  (3-17) = INCREMENT TO BUMP SUBR. ARG. LOCATION  FT095580
*              ARGUMENT (21-35) = POINTER IN S Y M T A B FOR THIS        FT095590
*                                 ARGUMENT'S ENTRY                       FT095600
        SPACE   2                                                        FT095610
*       ENTRY AT P R O L O G FORCES INCREMENT TO BE ZERO                  FT095620
        SPACE   3                                                        FT095630
PROLOG  ANA     =077777         SAVE ONLY ADDRESS                        FT095640
PROLGS  SXA     EXPROL,4        SAVE XR4                                 FT095650
        AXC     **,4            GET -(PRLTBL) POINTER                    FT095660
        TXL     PROL1,4,-PRLNGH-1  ERROR IF PRLTBL FULL ALREADY          FT095670
        PAX     ,4                 GET EQUIV POINTER                     FT095680
        XCL                        SAVE ARGUMENT IN MQ                   FT095690
        CAL     EQUIV,4            GET THIS ENTRY'S EQUIV WORD           FT095700
        ARS     15                 SET ACC(P,1-35) TO,                   FT095710
        RQL     3                    ARGUMENT NO. (P,1-5)                FT095720
        LGL     15                   DELTA (6-20)                        FT095730
        ALS     15                   PCOUNT (21-35)                      FT095740
        ACL     PCOUNT             STORE THIS NEW ENTRY IN PROLOGUE      FT095750
        XEC     PROLOG+2       RESET XR4 BEFORE STORING                  FT095760
        SLW     PRLTBL-1,4           TABLE (PRLTBL)                      FT095770
        LXA     PROLOG+2,4     BUMP PRLTBL POINTER                       FT095780
        TXI     *+1,4,1              X                                   FT095790
        SXA     PROLOG+2,4                                               FT095800
EXPROL  AXT     **,4               RESTORE XR4                           FT095810
        TRA     1,4                 AND EXIT                             FT095820
        REM                                                              FT095830
        REM     ... THE FOLLOWING MACRO IS EXECUTED IFF A CALL IS MADE   FT095840
        REM      TO ENTER A WORD INTO PRLTBL AND THE TABLE IS FULL.       FT095850
        REM                                                              FT095860
PROL1   STR     MSG100,,EXPROL                                           FT095870
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  C I T B L D FT095880
*       THIS SUBROUTINE IS RESPONSIBLE FOR INSERTING ENTRIES INTO THE    FT095890
*       COMPILED INSTRUCTION TABLE (C I T) AND CIT FLAG (C I T F L G)    FT095900
*       TABLE. EACH C I T ENTRY HAS ASSOCIATED WITH IT ONE AND ONLY ONE  FT095910
*       FLAG BYTE IN C I T F L G.  THUS C I T IS 6 TIMES LONGER THAN     FT095920
*       C I T F L G.                                                     FT095930
        REM                                                              FT095940
*       NOW C I T IS BROKEN INTO TWO HALVES, C I T 1 AND C I T 2.  IF    FT095950
*       C I T 1 IS FILLED, ENTRIES ARE MADE IN C I T 2.  IF SUFFICIENT   FT095960
*       ENTRIES ARE MADE TO FILL C I T 2 THEN IT (C I T 2) IS DUMPED ON  FT095970
*       S Y S U T 1 ALONG WITH ITS FLAG TABLE (C I T F L G 2).           FT095980
*       SUCCESSIVE ENTRIES ARE PUT INTO C I T 2.  THUS, ONCE FILLED,     FT095990
*       C I T 1 AND C T F L G 1 ARE NOT DISTURBED AND ONLY C I T 2 AND   FT096000
*       C T F L G 2 BLOCKS ARE DUMPED ONTO S Y S U T 1.  THUS, IN PASS 2 FT096010
*       C I T 1 IS PROCESSED WHILE S Y S U T 1 IS REWINDING.             FT096020
        REM                                                              FT096030
*       WHEN A C I T 2 BLOCK IS FULL EXCEPT FOR ITS LAST ENTRY, A FLAG   FT096040
*       OF 76 (OCTAL) IS INSERTED INTO C T F L G 2 AND THESE TWO BLOCKS  FT096050
*       ARE DUMPED ONTO S Y S U T 1 AND C I T B L D IS REINITIALIZED TO  FT096060
*       START FILLING C I T 2 AGAIN WITH SUBSEQUENT ENTRIES.             FT096070
        REM                                                              FT096080
*       A FLAG OF 77 (OCTAL) INDICATES THAT C I T B L D IS TO TRUNCATE   FT096090
*       THE CURRENT C I T AND C I T F L G BLOCKS.  IN THIS CASE THE 76   FT096100
*       FLAG IS OMITTED.                                                 FT096110
        SPACE   3                                                        FT096120
*          CALLING SEQUENCE,                                             FT096130
        REM                                                              FT096140
        REM                                                              FT096150
        REM                  .                                           FT096160
        REM                  .                                           FT096170
        REM                  .                                           FT096180
        REM                                                              FT096190
        REM     ACC (P,1-35) = C I T ENTRY                               FT096200
        REM     TSX   CITBLD,4,F                                         FT096210
        REM                                                              FT096220
        REM                  .                                           FT096230
        REM                  .                                           FT096240
        REM                  .                                           FT096250
        REM                                                              FT096260
        REM     WHERE F IS THE 6 BIT FLAG TO BE ENTERED INTO C I T F L G FT096270
        SPACE   3                                                        FT096280
*       ENTRY AT C I T B L D WILL BUMP P C O U N T  BY ONE AND ENTRY AT  FT096290
*       C I T B L N WILL NOT BUMP P C O U N T.                           FT096300
        REM                                                              FT096310
*       AN ENTRY TO C I T B L E IS MADE AT THE BEGINNING OF EACH         FT096320
*       ARITHEMETIC STATEMENT FUNCTION AND AN ENTRY AT C I T B L F IS    FT096330
*       MADE AT THE END OF EACH ARITHEMETIC STATEMENT.                   FT096340
        SPACE   3                                                        FT096350
CITBLN  STL     CITBL5          ENTRY WITH NO STEP OF PCOUNT             FT096360
        TRA     *+2                                                      FT096370
CITBLD  STZ     CITBL5          ENTRY WITH STEP OF PCOUNT BY 1           FT096380
CITBLH  ZET     CITBL6               IS THIS END OF ALL A. S. F.         FT096390
        TRA     CITBLG               YES                                 FT096400
        NZT     CITBL7               DID ANY A. S. F. EXIST              FT096410
        TRA     CITBLG               NO                                  FT096420
        NZT     LPASS1          IS PASS 1 OUTPUT TO BE LISTED.           FT096430
        TRA     CITBL4+1        NO                                       FT096440
        SXA     CITBLX,4        LIST INSTRUCTION, PCOUNT, AND CITBLD FLAG FT096450
        SLW     CITBL4                                                   FT096460
        XCL                                                              FT096470
        TSX     CINSTR,4
```

```
          SLW     CITBL2+2                                                  FT096480
          STQ     CITBL2+3                                                  FT096490
          CAL     PCOUNT                                                    FT096500
          TSX     OCTBCD,4                                                  FT096510
          SLW     CITBL2+1                                                  FT096520
CITBL3    LXA     CITBLX,4                                                  FT096530
          CAL     0,4                                                       FT096540
          ARS     18                                                        FT096550
          ANA     =077                                                      FT096560
          PAC     ,4                                                        FT096570
          CAL     CITBL1,4                                                  FT096580
          SLW     CITBL2+4                                                  FT096590
          TSX     LIST,4                                                    FT096600
                  CITBL2,,5                                                 FT096610
CITBLX    AXT     **,4                                                      FT096620
          CAL     CITBL4                                                    FT096630
          TRA     CITBL4+1                                                  FT096640
CITBL2    BCI     1,                                                        FT096650
          PZE                                                               FT096660
          PZE                                                               FT096670
          PZE                                                               FT096680
          PZE                                                               FT096690
CITBL4    BSS     1                                                         FT096700
          REM                                                               FT096710
          SXA     EXCIT,1                                                   FT096720
          SXA     EXCIT+1,2          INDEX                                  FT096730
          SXA     EXCIT+2,4          SAVE XR4                               FT096740
          STI     SAVIND                                                    FT096750
          LDQ     0,4                SET MQ = TSX  CITBLD,4,F               FT096760
 CTBL1    AXC     1,1                CIT POINTER                            FT096770
 CTBL2    AXC     1,2                FLAG WORD POINTER                      FT096780
          AXT     36,4               FLAG BYTE POINTER                      FT096790
 CTLOCA   SLW     **,1               STASH CIT WORD IN CIT1 OR CIT2         FT096800
          RQL     12                 SET MQ TO FXXXXX                       FT096810
 CTLOCE   CAL     **,2               PICK UP CURRENT FLAG WORD              FT096820
          LGL     6                  SHIFT F INTO LOW-ORDER ACCUMULATOR     FT096830
          PAI                        PUT FLAG IN LOW-ORDER INDICATORS       FT096840
          RNT     77                 IS THIS FLAG = 77                      FT096850
          TRA     CTBL4              NO, GO TEST FOR END OF CIT             FT096860
          STL     CITSW2             INDICATE 77 FLAG ENCOUNTERED           FT096870
          TXL     *+4,4,6            YES, GO STORE FLAG WORD                FT096880
          IXI     *+1,4,-6           LEFT-JUSTIFY CURRENT FLAG WORD         FT096890
          SXA     *+1,4              X                                      FT096900
          LGL     ..                                                        FT096910
 CTLOCB   SLW     **,2               SET THIS (THE LAST) FLAG WORD          FT096920
          NZT     CITSW1             EXIT IF THIS IS 77 FLAG AND NO RECS.   FT096930
          TRA     EXCIT               WRITTEN ON SYSUT1 YET                 FT096940
 IOS16    EQU     *                                                         FT096950
 CTBL3    WRITE   SYSUT1,BINARY,CTFLG2,*CTBL2       WRITE THIS FLAG BLOCK   FT096960
 CTBL3    STL     IOEX                                                      FT096970
          BRA     WRITE,,*+3                                                FT096980
          PZE     SYSUT1,,BINARY                                            FT096990
          MON     CTFLG2,,CTBL2                                             FT097000
 END16    REM                                                               FT097010
 IOS17    EQU     *                                                         FT097020
          WRITE   SYSUT1,BINARY,CIT2,*CTBL1         WRITE THIS CIT BLOCK    FT097030
          STL     IOEX                                                      FT097040
          BRA     WRITE,,*+3                                                FT097050
          PZE     SYSUT1,,BINARY                                            FT097060
          MON     CIT2,,CTBL1                                               FT097070
```

```
END17  REM                                                          FT097080
       STL     CITSW1          INDICATE RECORDS ARE ON SYSUT1       FT097090
       ZET     CITSW2          WAS THIS 77 FLAG                     FT097100
       TRA     CTBL7           YES, GO WEOF, REWIND SYSUT1          FT097110
       AXT     1,1             RESTORE FLAG WORD AND CIT POINTERS   FT097120
       SXA     CTBL2,1         X                                    FT097130
       SXA     CTBL1,1         X                                    FT097140
       AXT     36,4            RESET FLAG BYTE POINTER              FT097150
       SXA     CTBL2+1,4                                            FT097160
EXCIT  AXT     **,1            RESTORE                              FT097170
       AXT     **,2             INDEX                               FT097180
       AXT     **,4              REGISTERS                          FT097190
       LDI     SAVIND                                               FT097200
       ZET     CITBL5      SHOULD PCOUNT BE INCREMENTRD             FT097210
       TRA     1,4                                                  FT097220
       CLA     PCOUNT      YES.  BUMP IT BY 1                       FT097230
       ADD     =1                                                   FT097240
       STO     PCOUNT                                               FT097250
       TRA     1,4               ... AND EXIT                       FT097260
       REM                                                          FT097270
       REM     ... TO HANDLE TEST FOR END OF CIT BLOCK              FT097280
       REM                                                          FT097290
CTBL4  LXA     CTBL1,1         GET CIT POINTER INTO XR1             FT097300
CTLOCC SLW     **,2            STORE THIS FLAG WORD                 FT097310
       TXH     CTBL5,1,CITLNG-2 IF CIT FULL, GO CHECK WHICH BLOCK   FT097320
       TIX     CTBL6,4,6       IF FLAG WORD NOT FULL, GO TO CTBL6   FT097330
       LXA     CTBL2,2         IF NOT END OF CIT, BUMP              FT097340
       TXI     *+1,2,1          FLAG WORD COUNT AND CIT COUNT       FT097350
       SXA     CTBL2,2          X                                   FT097360
       TXI     EXCIT-3,1,1     GO EXIT                              FT097370
       REM                                                          FT097380
       REM     ... TO HANDLE FULL CIT BLOCK                         FT097390
       REM                                                          FT097400
CTBL5  LDQ     OCT76           MOVE 76 FLAG INTO PLACE              FT097410
       LGL     6               X                                    FT097420
CTLOCD SLW     **,2            STORE CURRENT FLAG                   FT097430
       ZET     CTSWTH                                               FT097440
       TRA     CTBL3                                                FT097450
       STL     CTSWTH                                               FT097460
       AXT     CIT2-1,1    RE-INITIALIZE FOR CIT2                   FT097470
       SXA     CTLOCA,1                                             FT097480
       AXT     CTFLG2-1,1                                           FT097490
       SXA     CTLOCB,1                                             FT097500
       SXA     CTLOCC,1                                             FT097510
       SXA     CTLOCD,1                                             FT097520
       SXA     CTLOCE,1                                             FT097530
       TRA     EXCIT-5         RETURN                               FT097540
       REM                                                          FT097550
       REM     ... TO BUMP CIT COUNTER IF FLAG WORD NOT FULL        FT097560
       REM                                                          FT097570
CTBL6  TXI     *+1,1,1         BUMP CIT WORD COUNTER                FT097580
       SXA     CTBL1,1         PLACE IN CTBL1                       FT097590
       TRA     EXCIT-1         GO SAVE BYTE POINTER AND EXIT        FT097600
       REM                                                          FT097610
       REM     ... TO WRITE FINAL E-O-F AND REWIND SYSUT1           FT097620
       REM                                                          FT097630
IOS18  EQU     *                                                    FT097640
CTBL7  WEOF    SYSUT1          WRITE END-OF-FILE                    FT097650
CTBL7  STL     IOEX                                                 FT097660
       BRA     WEOF,,*+2                                            FT097670
```

```
        MZE     SYSUT1                                                  FT097680
END18   REM                                                             FT097690
IOS19   EQU     *                                                       FT097700
        REWIND  SYSUT1          REWIND SYSUT1                           FT097710
        STL     IOEX                                                    FT097720
        BRA     REWIND,,*+2                                             FT097730
        MZE     SYSUT1                                                  FT097740
END19   REM                                                             FT097750
        TRA     EXCIT           RETURN                                  FT097760
        REM                                                             FT097770
        REM     ... TO HANDLE END OF ARITH. STMNT. FCNS.                FT097780
        REM                                                             FT097790
CITBLE  STZ     CITBL6          ENTRY FOR START OF A. S. F.             FT097800
        STL     CITBL7                                                  FT097810
        TRA     1,4                                                     FT097820
        REM                                                             FT097830
CITBLF  STL     CITBL6          ENTRY FOR END OF A. S. F.               FT097840
        TRA     1,4                                                     FT097850
        REM                                                             FT097860
CITBLG  STZ     CITBL6          CONTROL HERE IF END OF A. S. F.         FT097870
        LDQ     CITBL8                                                  FT097880
        STQ     CITBLH          SET TO SPEED UP NORMAL CITBLD           FT097890
        ZET     SNAME           IS THIS A SUBPROGRAM                    FT097900
        TRA     0,4             GO RE-ENTER CITBLD OR CITBLN            FT097910
        SLW     CITB15          SAVE THE ACCUMULATOR                    FT097920
        CAL     PCOUNT          BUMP PCOUNT BY 3                        FT097930
        ACL     =3                                                      FT097940
        SLW     PCOUNT                                                  FT097950
        STL     (FPT)           INDICATE (FPT) USED                     FT097960
        SXA     *+2,4                                                   FT097970
        TSX     CITBLN,4,H      GO OUTPUT 'H' FLAG FOR CITPRO           FT097980
        AXT     **,4                                                    FT097990
        CAL     CITB15          RESTORE THE ACCUM                       FT098000
        TRA     0,4             GO RE-ENTER CITBLD OR CITBLN            FT098010
        SPACE   3                                                       FT098020
CITBL8  TRA     CITBLH+4                                                FT098030
CITBL9  ZET     CITBL6                                                  FT098040
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  S C N B C D FT098050
*       ROUTINE TO SCAN  A STATEMENT, AND RETURN ONLY VARIABLE NAMES     FT098060
*       AND STATEMENT LABELS IN BCD.  RESULT WILL BE IN THE LOGICAL      FT098070
*       ACCUMULATOR.  THE ACCUM SIGN WILL BE SET MINUS IF THE ITEM       FT098080
*       FOUND IS A LABEL, AND WILL BE SET PLUS IF IT IS A VARIABLE NAME. FT098090
        REM                                                              FT098100
*       ENTRY...                TSX SCNBCD,4                             FT098110
*               XR 2 MUST BE POINTING TO THE STARTING POSITION IN COLUMN.FT098120
        REM                                                              FT098130
*       THERE IS AN ERROR RETURN IN THE ROUTINE IN CASE THE FIRST        FT098140
*       CHARACTER TO BE SCANNED IS A BREAK CHARACTER, OR IF A LABEL IS   FT098150
*       GREATER THAN 99999 OR ZERO, OR IF A SYMBOL IS TOO LONG.          FT098160
        SPACE   3                                                        FT098170
SCNBCD  SXA     SCNBCX,4                                                 FT098180
        SXA     SCNBCX+1,1                                               FT098190
        LDQ     COLUMN,2                                                 FT098200
        CRQ     SCANT,1,1                                                FT098210
        TXL     *+3,1,SCAN1-1   LETTER                                   FT098220
        TXL     SCNB2,1,SCAN1   ERROR IF NOT NUMERAL                     FT098230
        ERROR   81,SKEND                                                 FT098240
        REM                     VARIABLE - EXIT ON BREAK CHARACTER       FT098250
        REM                                                              FT098260
        SXD     SCNB1,2                                                  FT098270
        CAL     COLUMN,2                                                 FT098280
        TXI     *+1,2,-1                                                 FT098290
        LDQ     COLUMN,2                                                 FT098300
        CRQ     SCANT,1,1                                                FT098310
        TXL     *-3,1,SCAN1                                              FT098320
        SXA     *+1,2                                                    FT098330
        AXC     **,4                                                     FT098340
SCNB1   TXI     *+1,4,**                                                 FT098350
        TXH     SCNB3,4,6                                                FT098360
        ANA     SCANT1,4                                                 FT098370
        ORA     SCANT2,4        APPEND TRAILING BLANKS                   FT098380
SCNBCX  AXT     **,4                                                     FT098390
        AXT     **,1                                                     FT098400
        TRA     1,4                                                      FT098410
SCNB3   ERROR   15,SKEND                                                 FT098420
        REM                                                              FT098430
SCNB2   SXD     SCNB4,2         ALL NUMERIC LABEL SCANNER                FT098440
        CAL     COLUMN,2                                                 FT098450
        TXI     *+1,2,-1                                                 FT098460
        LDQ     COLUMN,2                                                 FT098470
        CRQ     SCANT,1,1                                                FT098480
SCNB5   TXL     *+2,1,SCAN1-1                                            FT098490
        TXL     *-4,1,SCAN1                                              FT098500
SCNB6   SXA     *+1,2           BREAK CHARACTER OR LETTER FOUND          FT098510
        AXC     **,4                                                     FT098520
SCNB4   TXI     *+1,4,**                                                 FT098530
        TXH     SCNB7,4,5                                                FT098540
        ANA     SCANT1,4                                                 FT098550
        TNZ     *+2                                                      FT098560
        ERROR   82,SKEND                                                 FT098570
        REM                                                              FT098580
        ORA     SCANT2,4                                                 FT098590
        LDQ     =H                                                       FT098600
        LGR     6                                                        FT098610
        TNZ     *-1                                                      FT098620
        XCL                                                              FT098630
        SSM                                                              FT098640
```

```
         TRA     SCNBCX                                          FT098650
         REM                                                     FT098660
SCNB7    ERROR   70,SKEND                                       FT098670
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  L O C A T E FT098680
*       ROUTINE TO LOCATE A SYMBOL IN THE SYMBOL TABLE, OR IF IT IS      FT098690
*       NOT IN THE TABLE, TO FIND A SPOT FOR IT.                         FT098700
        REM                                                              FT098710
*       ENTRY...                                                         FT098720
*             LDQ  SYMBOL                                                FT098730
*             TSX  LOCATE,4                                              FT098740
*       RETURN...                                                        FT098750
        REM                                                              FT098760
*       POSITION OF THE SYMBOL IN THE TABLE IS RETURNED IN THE ADDRESS   FT098770
*       OF THE ACCUMULATOR.  THE ACCUM SIGN IS SET MINUS IF THE SYMBOL   FT098780
*       WAS NOT ALREADY IN THE TABLE.                                    FT098790
        REM                                                              FT098800
*       THE SYMBOL TABLE OPERATES USING A SCRAMBLED SORT TECHNIQUE.      FT098810
*       THE BCD SYMBOLS THEMSELVES ARE STORED IN A REGION CALLED         FT098820
*       SYMTAB, WHILE THEIR EQUIVALENTS ARE STORED IN A REGION           FT098830
*       CALLED EQUIV.  BOTH THESE REGIONS GO BACKWARD IN MEMORY,         FT098840
*       WITH INDEXING STARTING AT ZERO.  (NOTE THAT THE FIRST CELL       FT098850
*       IN SYMTAB IS NOT AVAILABLE).                                     FT098860
        SPACE   3                                                        FT098870
LOCATE  STQ     LOCAT1                                                   FT098880
        MPY     =0400040004000                                           FT098890
        ANA     =07777                                                   FT098900
        SXA     LOCATX,4                                                 FT098910
        PAX     ,4                                                       FT098920
        CAL     LOCAT1                                                   FT098930
        TXH     *+2,4,0                                                  FT098940
        TXI     *+1,4,1                                                  FT098950
LOCAT5  NZT     SYMTAB,4        IS THIS SPOT MEPTY                       FT098960
        TRA     LOCAT3          YES.  GO ENTER SYMBOL                    FT098970
        LAS     SYMTAB,4        NO.  SCAN FORWARD FOR SYMBOL.            FT098980
        TXI     *-3,4,1                                                  FT098990
        TRA     LOCAT2          SYMBOL FOUND.                            FT099000
        TXI     *-5,4,1                                                  FT099010
LOCAT3  SLW     SYMTAB,4        ENTER THE SYMBOL                         FT099020
        STZ     EQUIV,4                                                  FT099030
        TXH     LOCAT4,4,LSYMTB-1     IS THIS THE BOTTOM OF THE TABLE.   FT099040
        PXA     ,4              NO                                       FT099050
        SSM                     MARK SYMBOL AS NOT IN                    FT099060
        LDI     EQUIV,4                                                  FT099070
LOCATX  AXT     **,4                                                     FT099080
        TRA     1,4                                                      FT099090
LOCAT2  PXA     ,4                                                       FT099100
        LDI     EQUIV,4                                                  FT099110
        LXA     LOCATX,4                                                 FT099120
        TRA     1,4                                                      FT099130
LOCAT4  AXT     1,4                                                      FT099140
        STZ     SYMTAB-LSYMTB                                            FT099150
        TRA     LOCAT5                                                   FT099160
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  S Y M S T O FT099170
*       ROUTINE TO ENTER AN INTEGER VARIABLE INTO SYMTAB.                FT099180
        SPACE   3                                                        FT099190
SYMSTO  SXA     SYMSTX,4                                                 FT099200
        XCL                                                              FT099210
        TSX     LOCATE,4                                                 FT099220
SYMSTY  SIL     MINTG+BVARB                                              FT099230
        PAX     ,4                                                       FT099240
        STI     EQUIV,4                                                  FT099250
        LFT     BEXTF+BINTF                                              FT099260
        TRA     SYMER                                                    FT099270
SYMSTX  AXT     **,4                                                     FT099280
        TRA     1,4                                                      FT099290
        REM                     ROUTINE TO ENTER INTEGER VARIABLE IS     FT099300
        REM                     SYMTAB. CHECK IF INTEGER MODE WITH ERROR EXFT099310
        REM                     EXIT IF NOT.                             FT099320
SYMSTB  STZ     SYMST1          SUPPRESS FLAGGINH AS DEFINED             FT099330
        TRA     *+2                                                      FT099340
SYMSTA  STL     SYMST1          ENTRY FOR THE DO INDEX VARIABLES         FT099350
        SXA     SYMSTX,4                                                 FT099360
        LAS     =HI00000                                                 FT099370
        TRA     *+1                                                      FT099380
        LAS     =H000000                                                 FT099390
        NOP                                                              FT099400
        ERROR   12,PD02                                                  FT099410
        XCL                                                              FT099420
        TSX     LOCATE,4                                                 FT099430
        ZET     SYMST1          SHOULD WE DEFINE IT                      FT099440
        SIL     BLHSX+BDOIX     YES.  AND TURN ON CURRENT DO INDEX BIT   FT099450
        TRA     SYMSTY          NO                                       FT099460
        REM                                                              FT099470
SYMER   CAL     SYMTAB,4        GET BCD NAME OF OFFENDING SYMBOL         FT099480
        SLW     FORM69                                                   FT099490
        ERROR   69,SKEND                                                 FT099500
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  S T O L B L FT099510
        REM                     ROUTINE TO ENTER LABEL IN THE SYMBOL     FT099520
        REM                     TABLE.   ENTRY...                        FT099530
        REM                          CAL  SYMBOL                         FT099540
        REM                          SSM                                 FT099550
        REM                          TSX  STOLBL,4                       FT099560
        REM                     OR                                       FT099570
        REM                          CAL  SYMBOL                         FT099580
        REM                          TSX  STOLBL,4                       FT099590
        REM                     AC SIGN SHOULD BE SET MINUS IF LABEL IS  FT099600
        REM                     TO BE FLAGGED AS HAVING APPEARED IN S    FT099610
        REM                     TRANSFER COMMAND AND HENCE MUST APPEAR   FT099620
        REM                     IN THE LABEL FIELD.                      FT099630
        REM                     IF LABEL IS ALREADY IN TABLE, IT WILL    FT099640
        REM                     BE FLAGGED AGAIN .                       FT099650
        REM                     LOCATION IN TABLE IS RETURNED TO THE     FT099660
        REM                     ADDRESS OF THE AC.                       FT099670
STOLBL  SXA     STOLBX,4                                                 FT099680
        STO     STOLB1          SAVE AC SIGN                             FT099690
        XCL                                                              FT099700
        TSX     LOCATE,4        FIND SYMBOL                              FT099710
        PAX     ,4                                                       FT099720
        LDQ     STOLB1                                                   FT099730
        TQP     *+2             IS LABEL REACHABLE                       FT099740
        SIL     BPATH           YES                                      FT099750
        STI     EQUIV,4         RESTORE EQUIVALENT IN S.T.               FT099760
        LFT     700000          IS IT ANYTHING BUT A LABEL.              FT099770
        ERROR   63,*+1        YES                                        FT099780
STOLBX  AXT     **,4                                                     FT099790
        TRA     1,4                                                      FT099800
```

```
        TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  S K I P     FT099810
*       ROUTINES TO TEST THE NEXT CHARACTER OF STATEMENT AGAINST A GIVEN FT099820
*       CHARACTER SUPPLIED IN THE DECREMENT.                             FT099830
*       CALLING SEQUENCES TO THE TWO ROUTINES ARE...                     FT099840
        REM                                                              FT099850
*               TSX SKIP,4,OCTAL VALUE OF CHARACTER                      FT099860
*               TRA ...         RETURN IF CHARACTERS DO NOT MATCH        FT099870
*               ... ...         RETURN IF CHARACTERS MATCH               FT099880
        REM                                                              FT099890
        REM                                                              FT099900
*               TSX SKIP1,4,OCTAL VALUE OF CHARACTER                     FT099910
*               TRA ...         RETURN IF CHARACTERS MATCH               FT099920
*               ... ...         RETURN IF CHARACTERS DO NOT MATCH        FT099930
        REM                                                              FT099940
SKIP    CAL     0,4                                                      FT099950
        ALS     12                                                       FT099960
        ERA     COLUMN,2                                                 FT099970
        ARS     30                                                       FT099980
        TNZ     1,4                                                      FT099990
        TXI     *+1,2,-1                                                 FT100000
        TRA     2,4                                                      FT100010
        SPACE   4                                                        FT100020
SKIP1   CAL     0,4                                                      FT100030
        ALS     12                                                       FT100040
        ERA     COLUMN,2                                                 FT100050
        ARS     30                                                       FT100060
        TNZ     2,4                                                      FT100070
        TXI     *+1,2,-1                                                 FT100080
        TRA     1,4                                                      FT100090
```

```
                TTL     P A S S  1  U T I L I T Y  P R O G R A M  -  B R E A K   FT100100
                REM     THIS ROUTINE GETS THE NEXT BREAK CHARACTER.  INITIALLY   FT100110
                REM     XR2 SHOULD BE SET AT 1ST CHARACTER TO BE EXAMINED, ROUTINFT100120
                REM     E WILL EXIT WITH XR2 POINTING TO CHARACTER FOLLOWING     FT100130
                REM     THAT WHICH CAUSED THE EXIT.  ROUTINE EXITS WITH BCD      FT100140
                REM     VALUE OF BREAK CHARACTER IN LAC RIGHT JUSTIFIED WITH     FT100150
                REM     LEADING ZEROS.                                           FT100160
                REM                                                              FT100170
BREAK           SXA     BREAKX,1                                                 FT100180
                LDQ     COLUMN,2                                                 FT100190
                CRQ     SCANT,1,1                                                FT100200
                TXH     *+2,1,SCAN1+1                                            FT100210
                TXI     *-3,2,-1                                                 FT100220
                CAL     COLUMN,2                                                 FT100230
                ARS     30                                                       FT100240
                TXI     *+1,2,-1                                                 FT100250
BREAKX          AXT     **,1                                                     FT100260
                TRA     1,4                                                      FT100270
```

```
        TTL     P A S S  1  -  I N D E X  C H E C K I N G                FT100280
*       THE FOLLOWING ROUTINES HANDLE INDEXING IN FASTRAN.               FT100290
        REM                                                              FT100300
*       THIS ROUTINE IS ENTERED WHENEVER AN INTEGER VARIABLE IS CHANGED   FT100310
*       IN A SUBSTITUTION STATEMENT, OR IS THE IMPLIED DO PARAMETER      FT100320
*       IN AN INPUT/OUTPUT LIST, OR IS READ IN VIA AN INPUT LIST.        FT100330
        REM                                                              FT100340
*       A TEST IS MADE TO DETERMINE IF THIS VARIABLE IS CURRENTLY A      FT100350
*       DO INDEX.  IF SO, THIS IS AN ERROR.                              FT100360
        REM                                                              FT100370
*       ENTRY...    LDI  SOME.FORM.OF.EQUIV.WORD.HAVING.FLAG.BITS        FT100380
*                   TSX  INDXT,4                                         FT100390
*                   (RETURN)                                             FT100400
        REM                                                              FT100410
*       NOTE THAT THIS ROUTINE WILL BE ENLARGED WHEN BETTER INDEXING     FT100420
*       IS INSTALLED IN FASTRAN.                                         FT100430
        SPACE   3                                                        FT100440
INDXT   LFT     BDOIX           IS THIS A DO INDEX                       FT100450
        ERROR   87,*+1          YES.  ERROR                              FT100460
        TRA     1,4             RETURN                                   FT100470
```

```
      TTL   POST PASS 1 - DESCRIPTION                          FT100480
*                                                              FT100490
      REM                                                      FT100500
      SPACE 9                                                  FT100510
*       *   *   *   *   *   *   *   *   *   *   *              FT100520
      REM                                                      FT100530
*           FASTRAN COMPILER                                   FT100540
      REM                                                      FT100550
*     RESEARCH COMPUTING CENTER                                FT100560
      REM                                                      FT100570
*         INDIANA UNIVERSITY                                   FT100580
      REM                                                      FT100590
*       BLOOMINGTON, INDIANA                                   FT100600
      SPACE 3                                                  FT100610
*       *   *   *   *   *   *   *   *   *   *   *              FT100620
      REM                                                      FT100630
*               POST PASS 1                                    FT100640
      REM                                                      FT100650
*       *   *   *   *   *   *   *   *   *   *   *              FT100660
```

```
      EJECT                                                              FT100670
*     ON ENTRY TO POST PASS 1, THE COMPLETE SOURCE PROGRAM HAS BEEN      FT100680
*     SCANNED AND THE COMPILED INSTRUCTION TABLE (C I T) AND THE         FT100690
*     ASSOCIATED FLAG TABLE (C I T F L G) HAVE BEEN GENERATED AND, IF    FT100700
*     OF SUFFICIENT LENGTH, RESIDE ON SYSUT1.                            FT100710
      REM                                                                FT100720
*     THE FOLLOWING IS AN OVERALL PICTURE OF POST PASS 1.  (DETAILED     FT100730
*     COMMENTS PRECEDE EACH SECTION).                                    FT100740
      SPACE  3                                                           FT100750
*     POST PASS 1 PERFORMS THE FOLLOWING FUNCTIONS IN THE INDICATED      FT100760
*     ORDER,                                                             FT100770
      SPACE  2                                                           FT100780
*        PPASS1-  LIST SYMBOL TABLE AND POOL AREA IFF 'LSTSYM' NON-ZERO  FT100790
*                 AND GO TO 'FINISH' IFF 'NOCODE' NON-ZERO.             FT100800
      REM                                                                FT100810
*        ASNASN-  VERIFY ASSIGNED GO TO USAGE.                          FT100820
      REM                                                                FT100830
*        ASEQUX-  DEFINE VARIABLES APPEARING IN EQUIVALENCE STATEMENTS. FT100840
      REM                                                                FT100850
*        TVECTL-  COUNT NUMBER OF LIBRARY ROUTINES USED IN OBJECT       FT100860
*                 PROGRAM.                                              FT100870
      REM                                                                FT100880
*        PROLEN-  CALCULATE LENGTH OF OBJECT TIME PROLOGUE.             FT100890
      REM                                                                FT100900
*        SYPASS-  CHECK VALIDITY OF SYMBOL TABLE ENTRIES.               FT100910
      REM                                                                FT100920
*        ASCMN2-  ASSIGN COMMON STORAGE FOR OBJECT PROGRAM              FT100930
      REM                                                                FT100940
*        OBJLEN-  CALCULATE LENGTHS OF VARIOUS SECTIONS OF THE OBJECT   FT100950
*                 PROGRAM.                                              FT100960
      REM                                                                FT100970
*        OUTUND-  OUTPUT UNDEFINED LABELS, STRINGS (FORMATS),           FT100980
*                 VARIABLES AND STATEMENTS WITH NO PATH OF FLOW.        FT100990
      REM                                                                FT101000
*        TSTPP2-  GO TO 'FINISH' IFF 'NOCODE' NON-ZERO.                 FT101010
      REM                                                                FT101020
*        TSTPST-  PUNCH SYMBOL TABLE IF NECESSARY                       FT101030
      REM                                                                FT101040
*        CDLABL-  SET-UP FOR BINARY DECK LABELLING.                     FT101050
      REM                                                                FT101060
*        OUTPGC-  WRITE PROGRAM CARD ON SYSBIN.                         FT101070
      REM                                                                FT101080
*        STVECT-  STASH TRANSFER VECTOR INTO BINARY CARD IMAGES.        FT101090
      REM                                                                FT101100
*        GENPRL-  GENERATE OBJECT TIME PROLOGUE.                        FT101110
      SPACE  5                                                           FT101120
*     THIS IS THE END OF POST PASS 1.  AT THIS POINT PASS 2 IS ENTERED.  FT101130
```

```
         TTL     P O S T  P A S S  1  -  P P A S S 1                    FT101140
*        POST PASS 1 IS ENTERED AT THIS POINT.  A COUPLE OF TESTS ARE   FT101150
*        MADE BEFORE POST PASS 1 PROPER IS ENTERED.                     FT101160
         SPACE   3                                                      FT101170
PPASS1   ZET     LSTSYM              ARE SYMTAB AND POOL TO BE OUTPUT   FT101180
         TSX     STDUMP,4            YES, OUTPUT SYMTAB AND POOL        FT101190
         ZET     NOCODE              NO, WAS FATAL ERROR DETECTED IN PASS 1FT101200
         TRA     FINISH              YES, EXIT THIS COMPILATION         FT101210
         REM                         NO, CONTINUE POST PASS 1 PROCESSING FT101220
         SLF                   TURN OFF THE CONSOLE WATCHER'S LIGHT     FT101230
```

```
        TTL     P O S T  P A S S  1  -  A S N A S N                    FT101240
*       POST PASS 1 PROCESSOR FOR ASSIGN VARIABLES AND LABELS          FT101250
*                                                                      FT101260
*       ENTRY...  ASNASN                                               FT101270
*       THIS PROCESSOR IS ENTERED AT THE BEGINNING OF POST PASS 1      FT101280
*       PROCESSING.  THE ASSIGN TABLES ARE SCANNED.  FOR THE           FT101290
*       VARIABLES, BOTH TASSN AND TGOTO MUST BE ON.                    FT101300
*       ALL LABEL ENTRIES MUST HAVE MINUS SIGNS, INDICATING THAT THEY  FT101310
*       APPEAR IN AN ASSIGNED GO TO LIST.                              FT101320
        REM                                                            FT101330
        REM                                                            FT101340
ASNASN  NZT     ASNFLG          ARE THE TABLES FINISHED YET            FT101350
        TRA     ASEQUX          YES.  CONTINUE TO NEXT SECTION         FT101360
        REM                                                            FT101370
        STZ     .P              INITIALIZE FLAG CELL                   FT101380
        AXT     15,4            SET XR4 SO THE MESSAGE PRINT TEST WORKS FT101390
        REM                                                            FT101400
        LXD     ASNFLG,1        GET START OF AN ASSIGN TABLE           FT101410
        CAL     POOL,1          GET VARIABLE ENTRY                     FT101420
        STD     ASNFLG          SET BEGINNING OF NEXT ASSIGN STACK     FT101430
        REM                                                            FT101440
        PAX     ,2              GET EQUIV TABLE POINTER                FT101450
        PAI                                                            FT101460
        RNT     TASSN+TGOTO     ARE BOTH FLAG BITS ON.                 FT101470
        TRA     ASNAS2          NO.  THERE IS THEN ONE OF TWO ERRORS.  FT101480
        REM                                                            FT101490
ASNAS1  TXI     *+1,1,1         BUMP POINTER FOR NEXT LABEL ENTRY      FT101500
        CLA     POOL,1          GET LABEL ENTRY                        FT101510
        SLW     .P+1            SAVE SYMTAB POINTER                    FT101520
        PDX     ,1              GET NEXT POINTER                       FT101530
        TMI     ASNAS4          MINUS SIGN IS OKAY                     FT101540
        REM                                                            FT101550
        ZET     .P              TEST FLAG                              FT101560
        TRA     ASNAS5          NO INITIALIZATION NEEDED               FT101570
        REM                                                            FT101580
        CAL     SYMTAB,2        GET BCD NAME OF GUILTY VARIABLE        FT101590
        SLW     FORM29+13       PUT NAME INTO ERROR MESSAGE            FT101600
        TSX     LIST,4          SKIP A LINE                            FT101610
        PZE     BLANK,,1                                               FT101620
        ERROR   29,*+1                                                 FT101630
        REM                                                            FT101640
ASNAS6  AXT     16,4            PUT BLANKS INTO BUFFER                 FT101650
        CAL     =H                                                     FT101660
        SLW     COLUMN+16,4                                            FT101670
        TIX     *-1,4,1                                                FT101680
        REM                                                            FT101690
        AXT     14,4            LOAD XR FOR STORAGE OF LABEL NAMES     FT101700
        ZET     .P              ARE WE DONE FOR THIS TIME.             FT101710
        TRA     ASNAS4          YES.                                   FT101720
        REM                                                            FT101730
        STL     .P              SET FLAG NON-ZERO                      FT101740
ASNAS5  LXA     .P+1,2          GET DIRECTOR TO SYMTAB                 FT101750
        CAL     SYMTAB,2        GET BCD NAME OF LABEL                  FT101760
        SLW     COLUMN+16,4     PUT INTO BUFFER                        FT101770
        TIX     ASNAS4,4,1                                             FT101780
        REM                                                            FT101790
ASNAS7  TSX     LIST,4          WRITE OUT THIS BUFFER                  FT101800
                COLUMN,,16                                             FT101810
        TRA     ASNAS6                                                 FT101820
        REM                                                            FT101830
```

```
ASNAS4  TXH     ASNAS1+1,1,0    IS THIS ALL THE LABELS IN THIS TABLE.          FT101840
        TXL     ASNAS7,4,13     WRITE LINE IF BUFFER HAS INFORMATION IN IT     FT101850
        TRA     ASNASN          YES, RETURN FOR NEXT TABLE                     FT101860
        REM                                                                    FT101870
        REM                                                                    FT101880
ASNAS2  CAL     SYMTAB,2        GET BCD NAME OF GUILTY VARIABLE                FT101890
        SLW     FORM30+3        PUT NAME INTO ERROR MESSAGES                   FT101900
        SLW     FORM31+4                                                       FT101910
        REM                                                                    FT101920
        TSX     LIST,4          PUT OUT BLANK LINE                             FT101930
        PZE     BLANK,,1                                                       FT101940
        REM                                                                    FT101950
        RFT     TASSN           IS IT TASSN THAT IS OFF.                       FT101960
        ERROR   30,ASNASN       NO. THUS TGOTO IS OFF.                         FT101970
        ERROR   31,ASNASN       YES, TASSN IS OFF.  PRINT MESSAGE              FT101980
        SPACE   2                                                              FT101990
TASSNX  VFD     18/,018/TASSN   WHOLE-WORD REPRESENTATIONS OF FLAGS            FT102000
TGOTOX  VFD     18/,018/TGOTO                                                  FT102010
```

```
        TTL     P O S T  P A S S  1  -  A S E Q U X                    FT102020
*       POST-PASS 1 PROCESSOR FOR EQUIVALENCE STACKS.                  FT102030
*                                                                      FT102040
*       ENTRY...  ASEQUX                                               FT102050
*       THIS ROUTINE ASSIGNS THE SYMBOLS IN THE EQUIVALENCE            FT102060
*       CHAINS TO COMMON OR VARIABLE STORAGE.  EACH LINKED CHAIN       FT102070
*       IS TREATED IN ITS TURN AS FOLLOWS...                           FT102080
*       THE STACK IS SCANNED TO FIND THE LARGEST DEL (DELMAX)          FT102090
*       AND TO DETERMINE IF ANY ELEMENT IS IN COMMON.  THE STACK       FT102100
*       IS THEN SCANNED AGAIN.  IF ANY ELEMENT OF THE CHAIN IS         FT102110
*       IN COMMON, THE FLAG BITS BCOMN AND BLHSX ARE                   FT102120
*       TURNED ON FOR EACH ELEMENT IN THIS CHAIN.                      FT102130
*       VARIABLES ARE ASSIGNED TO STORAGE (COMMON OR LOW-CORE          FT102140
*       VARIABLE) BY THE EXPRESSION...                                 FT102150
*               LOCATION = CURRENT LCOMMN (OR LVARBS) + DELMAX - DEL   FT102160
*       THE OVERHANG (HANG) IS COMPUTED FROM THE EXPRESSION...         FT102170
*               HANG = LENGTH + DELMAX - DEL.                          FT102180
*       THE MAXIMUM OVERHANG (HANGMX) TO DATE IS FIGURED FROM...       FT102190
*               HANGMX = MAXIMUM(HANG, HANGMX).                        FT102200
*       AT THE END OF THIS STACK, THE APPROPRIATE COUNTER (LCOMMN      FT102210
*       OR LVARBS) IS INCREASED BY HANGMX.  THE PROCESSOR THEN         FT102220
*       RETURNS TO TREAT THE NEXT EQUIVALENCE CHAIN.                   FT102230
*                                                                      FT102240
*       THE LOCATION ASSIGNED TO EACH VARIABLE IS PLACED IN THE ADDRESS FT102250
*       FIELD OF THE EQUIV TABLE WORD.  NOTE THAT THE DIMTBL POINTER   FT102260
*       FOR ARRAYS IS LOST AT THIS POINT.                              FT102270
        REM                                                            FT102280
        REM                                                            FT102290
ASEQUX  ZET     NOCODE          IS PROGRAM IN ERROR.                   FT102300
        TRA     EXASEQ          YES. DON'T PROCESS E Q U T B L STACKS  FT102310
        REM                                                            FT102320
ASEQU6  STZ     HANGMX                                                 FT102330
        STZ     DELMAX                                                 FT102340
        STZ     COMFLG                                                 FT102350
        STZ     CTCELL                                                 FT102360
        REM                                                            FT102370
ASEQU7  LXD     INITFG,4        GET POINTER TO STACK DIRECTOR          FT102380
        TXH     *+2,4,0         IS THIS END OF EQUTBL STACKS.          FT102390
        REM                                                            FT102400
        TRA     EXASEQ                                                 FT102410
        REM                                                            FT102420
        CAL     POOL,4          NO.  GET DIRECTOR TO NEXT STACK        FT102430
        SLW     INITFG                                                 FT102440
        LXA     INITFG,4        GET EQUTBL POSITION OF TOP OF STACK    FT102450
        REM                                                            FT102460
        LDI     EQUTBL,4        GET TOP OF STACK LINKER WORD           FT102470
        LNT     300000          HAS THIS STACK ALREADY BEEN PROCESSED. FT102480
        TXI     *+2,4,1         NO.  BUMP POINTER                      FT102490
        TRA     ASEQU7          YES.  SKIP TO NEXT STACK               FT102500
        REM                                                            FT102510
ASEQU2  CLA     EQUTBL,4        GET ENTRY FROM EQUTBL                  FT102520
        TMI     ASEQU1          LINK WORD FOUND                        FT102530
        PAX     ,1              GET EQUIV TABLE POINTER                FT102540
        ARS     18              SHIFT DEL TO ADDRESS                   FT102550
        LDQ     DELMAX                                                 FT102560
        TLQ     *+2                                                    FT102570
        XCA                     PUT LARGER IN AC                       FT102580
        STA     DELMAX          DELMAX = LARGEST DEL                   FT102590
        REM                                                            FT102600
        LDI     EQUIV,1         GET EQUIV ENTRY                        FT102610
```

```
        LFT     BCOMN           IS COMMON FLAG ON.                              FT102620
        STL     COMFLG          YES.  SET COMFLG ON                             FT102630
        LFT     BLHSX           IS THIS VARIABLE DEFINED.                       FT102640
        STL     DEFINE          YES.  TURN ON THE DEFINE FLAG                   FT102650
        TXI     ASEQU2,4,1      RETURN FOR ANOTHER ENTRY                        FT102660
        REM                                                                     FT102670
ASEQU1  PAX     ,4              LINK WORD FOUND.  GET ADDRESS                   FT102680
        TXH     ASEQU2-2,4,0    IS THIS BOTTOM OF STACK.  NO.                   FT102690
        REM                                                                     FT102700
        AXT     LCOMMN,4        BOTTOM OF STACK.                                FT102710
        NZT     COMFLG          SET PROPER COUNTER IN TRANSFER CELL             FT102720
        AXT     LVARBS,4                                                        FT102730
        SXA     CTCELL,4                                                        FT102740
        REM                                                                     FT102750
        LXA     INITFG,4        PASS THROUGH THE CHAIN AGAIN                    FT102760
        TXI     *+1,4,1         BUMP POINTER                                    FT102770
        REM                                                                     FT102780
ASEQU9  CLA     EQUTBL,4        GET ENTRY                                       FT102790
        TMI     ASEQU3          LINK WORD FOUND.                                FT102800
        REM                                                                     FT102810
        PAX     ,1              SET ENTRY TO EQUIV TABLE                        FT102820
        ARS     18              SHIFT DEL TO ADDRESS                            FT102830
        SUB     DELMAX          DEL - DELMAX                                    FT102840
        SLW     ORIGIN          ORIGIN = DELMAX - DEL                           FT102850
        REM                                                                     FT102860
        LDI     EQUIV,1         GET EQUIV TABLE ENTRY                           FT102870
        LFT     BARRY           IS IT AN ARRAY.                                 FT102880
        TRA     ASEQU5          YES.                                            FT102890
        REM                                                                     FT102900
        CAL     =1              NO.                                             FT102910
ASEQU4  LFT     BDOUB           RE-ENTRY FROM ARRAY LOOP.                       FT102920
        ALS     1               DOUBLE LENGTH IF BDOUB IS ON                    FT102930
        ADD     ORIGIN          LENGTH + ORIGIN = HANG                          FT102940
        LDQ     DELMAX                                                          FT102950
        TLQ     *+2             TEST AC AGAINST MQ                              FT102960
        XCA                     PUT LARGER IN AC                                FT102970
        REM                                                                     FT102980
        LDQ     HANGMX          HANG VERSUS HANGMX                              FT102990
        TLQ     *+2                                                             FT103000
        XCA                     PUT LARGER IN AC                                FT103010
        STO     HANGMX          HANGMX = LARGEST HANG                           FT103020
        REM                                                                     FT103030
        CLA*    CTCELL          GET PROPER COUNTER                              FT103040
        ADD     ORIGIN          OVERALL RELOCATABLE ORIGIN OF SYMBOL            FT103050
        REM                                                                     FT103060
        ZET     COMFLG          SKIP IF NOT IN COMMON STACK                     FT103070
        SIL     BCOMN+BLHSX     TURN ON COMMON BIT AND DEFINITION BIT.          FT103080
        ZET     DEFINE          IS THIS STACK DEFINED.                          FT103090
        SIL     BLHSX           YES.  TURN ON THE EQUIV DEFINED BIT             FT103100
        STI     EQUIV,1         RESTORE ENTRY                                   FT103110
        STA     EQUIV,1         NO.  SET DEFINITION OF ENTRY IN EQUIV TABLE     FT103120
        REM                                                                     FT103130
        TXI     ASEQU9,4,1      RETURN FOR ANOTHER ENTRY                        FT103140
        REM                                                                     FT103150
ASEQU5  PIA                     ARRAY.  GET LENGTH FROM DIMTBL                  FT103160
        PAX     ,2                                                              FT103170
        CAL     DIMTBL-1,2                                                      FT103180
        ANA     =077777         SAVE ONLY ADDRESS                               FT103190
        TRA     ASEQU4                                                          FT103200
        REM                                                                     FT103210
```

```
ASEQU3  PAX     ,4              LINK WORD FOUND                                 FT103220
        TXL     ASEQU8,4,0      TRANSFER IF THIS IS THE END OF THE CHAIN         FT103230
        REM                                                                     FT103240
        CAL     =-3B2           NOT YET END OF STACK                            FT103250
        STP     EQUTBL,4        SET 'USED STACK' PREFIX BITS IN LINK WORD       FT103260
        TXI     ASEQU9,4,1      CONTINUE PROCESSING THIS STACK                  FT103270
        REM                                                                     FT103280
ASEQU8  CLA     HANGMX          UPDATE THE COMMON OR VARIABLE COUNTER           FT103290
        ADD*    CTCELL                                                          FT103300
        STO*    CTCELL                                                          FT103310
        TRA     ASEQU6          RETURN FOR NEXT CHAIN                           FT103320
        REM                                                                     FT103330
EXASEQ  EQU     *                                                               FT103340
```

```
        TTL     P O S T  P A S S  1  -  T V E C T L                     FT103350
*       THE FOLLOWING ROUTINE SCANS THE LIBRARY NAME TABLE (S L N T A B) FT103360
*       TO DETERMINE WHICH LIBRARY ROUTINES ARE USED BY THE OBJECT      FT103370
*       PROGRAM.  A LINEAR SEARCH IS MADE THROUGH S L N T A B AND EACH  FT103380
*       ROUTINE USED IS DEFINED AS A TRANSFER VECTOR ENTRY, THE FIRST   FT103390
*       HAVING THE DEFINITION 1, THE SECOND 2, ETC.  THIS DEFINITION    FT103400
*       GOVERNS THE ORDER IN WHICH THE ELEMENTS OF THE TRANSFER VECTOR  FT103410
*       WILL APPEAR IN THE OBJECT PROGRAM.  NOTICE THAT THIS ROUTINE    FT103420
*       COUNTS LIBRARY ROUTINES ONLY ... EXTERNAL (NON-LIBRARY)         FT103430
*       REFERENCES ARE COUNTED LATER ON IN POST PASS 1 (S Y P A S S).   FT103440
        SPACE   3                                                       FT103450
TVECTL  AXT     0,1             XR1 HAS NUMBER OF ROUTINES USED         FT103460
        AXT     -2,4            XR4 IS TABLE POINTER                    FT103470
 TVEL1  ZET     SLNTAB-2,4      IS THIS ROUTINE USED IN TEXT            FT103480
        TXI     TVEL2,1,1       YES, BUMP TOTAL BY ONE                  FT103490
        TXL     *+2,4,-LSLNTB   EXIT IF DONE                            FT103500
        TXI     TVEL1,4,-2      OTHERWISE BUMP POINTER AND GO BACK      FT103510
        SXA     LTVECT,1        SAVE THIS (PARTIAL) COUNT IN LTVECT     FT103520
        REM                                                             FT103530
```

```
        TTL     P O S T  P A S S  1  -  P R O L E N                    FT103540
*       THE FOLLOWING ROUTINE CALCULATES THE LENGTH OF THE OBJECT TIME  FT103550
*       PROLOGUE TO BE GENERATED FOR THIS PROGRAM.                      FT103560
        SPACE   2                                                       FT103570
*       THIS VALUE IS DEPENDENT ON,                                     FT103580
        REM                                                             FT103590
*          1. INDEX REGISTER SAVING REQUIREMENTS                        FT103600
*          2. THE CONTENTS OF THE PROLOGUE TABLE (P R L T B L)          FT103610
*          3. THE PRESENCE OR ABSENCE OF ARITHMETIC STATEMENT FUNCTIONS FT103620
        SPACE   2                                                       FT103630
*       THE PROLOGUE TABLE IS FIRST SORTED IN ORDER THAT ALL PROLOGUE   FT103640
*       TABLE ENTRIES FOR EACH ARGUMENT ARE TOGETHER (TO MINIMIZE LENGTH FT103650
*       OF PROLOGUE).                                                   FT103660
        SPACE   2                                                       FT103670
*       FORMAT OF P R L T B L ENTRIES,                                  FT103680
        REM                                                             FT103690
*          BITS (S,1-5) = SUBROUTINE ARGUMENT NUMBER                    FT103700
*                (6-20) = ARGUMENT INCREMENT (DELTA)                    FT103710
*               (21-35) = LOCATION (RELATIVE 0) OF TEXT TO BE INITIALIZEDFT103720
        SPACE   2                                                       FT103730
*       SEE G E N P R L FOR A DISCUSSION OF THE ACTUAL LAYOUT OF THE    FT103740
*       OBJECT TIME PROLOGUE.                                           FT103750
        SPACE   3                                                       FT103760
PROLEN  AXT     0,4             XR4 IS USED TO CONTAIN LPROLG           FT103770
        NZT     SUBR            IS THIS A SUBROUTINE COMPILATION        FT103780
        TRA     PROLA           NO, EXIT THIS SECTION                   FT103790
        STZ     ERASE4          ERASE4 = 0 IFF NO XR*S YET AT ALL       FT103800
        LXA     PROLOG+2,1      SET XR1 TO LENGTH OF PRLTBL             FT103810
        LXA     PROLOG+2,2      SET XR2 TO LENGTH OF PRLTBL             FT103820
        TXL     PROL9,1,1       GO GET XR USAGE IF PRLTBL EMPTY         FT103830
        SXD     *+2,1           SAVE LENGTH OF PRLTBL                   FT103840
        AXT     PRLTBL-2,1      CALCULATED ADDR. OF LAST ENTRY IN       FT103850
        TXI     *+1,1,..        PRLTBL                                  FT103860
        SXA     *+3,1           SAVE THIS ADDR. IN SORT CALL SEQ.       FT103870
        CALL    SORT,PRLTBL,..  SORT PRLTBL                             FT103880
        TXI     *+1,1,1         XR1 IS LAST PRLTBL LOC. + 1             FT103890
        SXA     PROL2,1         SVAE THIS IN PROL2                      FT103900
        STZ     ERASE           ERASE = ARG. NO. AND DELTA              FT103910
        STZ     ERASE1          ERASE1 = ARG. NO.                       FT103920
        STZ     ERASE2          ERASE2 = DELTA                          FT103930
        STZ     ERASE3          ERASE3 = 0 IFF NO XR*S YET THIS ARG.    FT103940
        AXT     0,4             RESET XR4 AT BEGINNING                  FT103950
        TXI     *+1,2,-1        SET XR2 TO LENGTH OF PRLTBL             FT103960
PROL2   CAL     ..,2            PICK-UP PRLTBL ENTRY                    FT103970
        ANA     ARGDEL          MASK OUT PCOUNT                         FT103980
        LAS     ERASE           IS THIS SAME ARG. NO. AND DELTA         FT103990
        TRA     PROL4           NO, GO TO PROL4                         FT104000
        TRA     PROL3           YES, GO TO PROL3                        FT104010
        MACER                   PRLTBL OUT OF SORT                      FT104020
        REM                                                             FT104030
        REM     ... TO HANDLE SAME ARG. NO. AND DELTA                   FT104040
        REM                                                             FT104050
PROL3   TXI     *+1,4,1         BUMP LENGTH OF PROLOGUE BY 1            FT104060
        TIX     PROL2,2,1       STEP THROUGH PRLTBL                     FT104070
        TRA     PROL8           AT END OF PRLTBL, GO GET XR NEEDS       FT104080
        REM                                                             FT104090
        REM     ... TO HANDLE CHANGE IN ARG. NO. AND/OR DELTA           FT104100
        REM                                                             FT104110
PROL4   SLW     ERASE           SAVE NEW ARG. NO. AND DELTA IN ERASE    FT104120
        ANA     ARGBIT          MASK OUT DELTA                          FT104130
```

```
        LAS     ERASE1          COMPARE WITH LAST ARG. NO.              FT104140
        TRA     PROL7           GO TO PROL7 IF NEW ARG. NO.             FT104150
        TRA     PROL5           GO TO PROL5 IFF SAME ARG. NO.           FT104160
        MACER                   PRLTBL OUT OF SORT                      FT104170
        REM                                                             FT104180
        REM     ... TO HANDLE CHANGE IN DELTA ONLY                      FT104190
        REM                                                             FT104200
PROL5   NZT     ERASE3          HAVE XR*S BEEN USED FOR THIS ARG. NO.   FT104210
        TXI     *+1,4,1         NO, BUMP LPROLG BY 1                    FT104220
        CAL*    PROL2           SET EARSE2 TO THIS NEW DELTA            FT104230
        ANA     DELBIT           X                                      FT104240
        SLW     ERASE2           X                                      FT104250
        STL     ERASE3          INDICATE XR USED FOR THIS ARGUMENT      FT104260
PROL6   TXI     PROL3+1,4,2     BUMP LPROLG BY 2 AND STEP THRU PRLTBL   FT104270
        REM                                                             FT104280
        REM     ... TO HANDLE CHANGE IN ARG. NO.                        FT104290
        REM                                                             FT104300
PROL7   ZET     ERASE3          WAS XR USED FOR LAST ARGUMENT           FT104310
        STL     ERASE4          YES, TURN EARSE4 ON                     FT104320
        SLW     ERASE1          SAVE NEW ARGUMENT NUMBER                FT104330
        XEC     PROL2           GET THIS PRLTBL ENTRY AGAIN             FT104340
        ANA     DELBIT          SAVE THIS DELTA IN ERASE2               FT104350
        SLW     ERASE2           X                                      FT104360
        STZ     ERASE3          INDICATE NO XR*S THIS ARGUMENT YET      FT104370
        ZET     ERASE2          IS THIS DELTA ZERO                      FT104380
        TXI     PROL6-1,4,2     NO, THEN BUMP LPROLG BY 2               FT104390
        TRA     PROL6           YES, GO TO PROL6 AND BUMP LPROLG        FT104400
        REM                                                             FT104410
        REM     ... TO HANDLE COUNTING OF XR SAVING REQUIREMENTS        FT104420
        REM                                                             FT104430
PROL8   ZET     ERASE3          TO HANDLE SPECIAL CASE WHERE LAST ARG   FT104440
        STL     ERASE4           USED XR BUT NO OTHER ARGS. DID         FT104450
PROL9   SXA     LPBODY,4        SET LBODY TO LENGTH OF PROLOGUE BODY    FT104460
        NZT     RTNFLG          DID A 'RETURN' STATEMENT OCCUR          FT104470
        TRA     PROLA-2         NO, DO NOT SAVE XR'S IN PROLOGUE        FT104480
        CAL     XRFLAG          GET TEXT XR NEEDS                       FT104490
        ZET     ERASE4          OR IN PROLOGUE XR NEEDS                 FT104500
        ORA     TG1BIT           X                                      FT104510
        SLW     XRFLAG                                                  FT104520
        PAI                     COUNT NUMBER OF TOTAL XR*S REQUIRED     FT104530
        AXT     0,2              BUILD THIS COUNT IN XR2                FT104540
        RFT     400000           X                                      FT104550
        TXI     *+1,2,1          X                                      FT104560
        RFT     200000           X                                      FT104570
        TXI     *+1,2,1          X                                      FT104580
        RFT     100000           X                                      FT104590
        TXI     *+1,2,1          X                                      FT104600
        SXA     NOXRS,2         SAVE THIS COUNT IN NOXRS                FT104610
        TXL     PROLA-3,2,0     GO TO PROLA-3 IF NO XR'S NEED SAVING    FT104620
        PXD     ,2              OTHERWISE BUMP LPROLG BY 2 FOR EACH     FT104630
        ALS     1                XR THAT NEEDS SAVING.                  FT104640
        STD     *+1              ONE FOR THE SXA AND ONE FOR THE AXT    FT104650
        TXI     *+1,4,..         X                                      FT104660
        TXI     *+1,4,1         BUMP LPROLG BY 1 FOR THE TRA (N),4      FT104670
        ZET     LARSTS          BUMP LPROLG BY 1 IF THIS IS A SUBR.     FT104680
        TXI     *+1,4,1          AND THERE WERE ARITH. STMNT. FCNS.     FT104690
PROLA   SXA     LPROLG,4        SAVE THIS FINAL COUNT IN LPROLG         FT104700
```

```
        TTL     P O S T  P A S S  1  -  S Y P A S S                    FT104710
*       THE FOLLOWING ROUTINE CONSTITUTES THE POST PASS 1 SYMBOL TABLE  FT104720
*       (S Y M T A B / E Q U I V) PASS.  THIS ROUTINE HAS TWO MAIN      FT104730
*       FUNCTIONS,                                                      FT104740
        REM                                                             FT104750
*          1. DEFINE LOW CORE VARIABLE STORAGE, FORMATS AND EXTERNAL    FT104760
*             TRANSFER VECTOR ENTRIES AND,                              FT104770
*          2. COLLECT ALL UNDEFINED LABELS, UNDEFINED STRINGS, UNDEFINED FT104780
*             VARIABLES AND STATEMENTS WITH NO PATH OF FLOW FOR LATER   FT104790
*             OUTPUT.                                                   FT104800
        SPACE   3                                                       FT104810
*       EACH E Q U I V ENTRY IS EXAMINED FOR WHICH THE ASSOCIATED       FT104820
*       S Y M T A B ENTRY IS NON-ZERO.  THE FOLLOWING ACTIONS ARE TAKEN FT104830
*       FOR THE VARIOUS E Q U I V ENTRIES POSSIBLE,                     FT104840
        SPACE   2                                                       FT104850
*          EXTERNAL SYMBOL- A. EXTERNAL SYMBOLS ARE CHAINED TOGETHER    FT104860
*                             USING THE DECREMENT OF THE E Q U I V WORD.FT104870
        REM                                                             FT104880
*                          B. THE LENGTH OF THE TRANSFER VECTOR         FT104890
*                             (L T V E C T) IS BUMPED BY ONE AND THE    FT104900
*                             SYMBOL IS GIVEN THIS DEFINITION (IN       FT104910
*                             ADDRESS OF E Q U I V WORD).               FT104920
        REM                                                             FT104930
*                          C. IF THE FIRST CHARACTER OF THE SYMBOL IS    FT104940
*                             '4' THIS CHARACTER IS CHANGED TO 'D'.  IF FT104950
*                             THE FIRST CHARACTER IS '1' IT IS CHANGED  FT104960
*                             TO 'I'.                                   FT104970
        REM                                                             FT104980
*          STATEMENT LABEL- A. IF THE STATEMENT LABEL IS UNDEFINED IT IS FT104990
*                             STACKED WITH OTHER UNDEFINED LABELS.      FT105000
        REM                                                             FT105010
*                          B. IF THE STATEMENT LABEL HAS NO PATH OF     FT105020
*                             FLOW TO IT THIS LABEL IS STACKED WITH     FT105030
*                             OTHER LABELS OF THIS TYPE.                FT105040
        REM                                                             FT105050
*             FORMAT (STRING)- A. IF THE FORMAT IS UNDEFINED IT IS STACKED FT105060
*                             WITH OTHER UNDEFINED FORMATS.             FT105070
        REM                                                             FT105080
*                          B. IF NOT UNDEFINED, THIS STRING IS CHAINED   FT105090
*                             TO PREVIOUS STRINGS USING THE DECREMENT   FT105100
*                             OF FIRST P O O L ENTRY FOR THIS STRING.   FT105110
        REM                                                             FI105120
*                          C. THE STRING IS DEFINED (IN ADDRESS OF      FT105130
*                             E Q U I V) TO BE (LENGTH OF STRINGS SO    FT105140
*                             FAR (L S T R I N)) + (LENGTH OF THIS      FT105150
*                             STRING) - 1.                              FT105160
        REM                                                             FT105170
*                          D. L S T R I N IS BUMPED BY THE LENGTH OF    FT105180
*                             THIS STRING.                              FT105190
        REM                                                             FT105200
*          SUBROUTINE ARGUMENT- A. NO ACTION TAKEN BY S Y P A S S.      FT105210
        REM                                                             FT105220
*             INTEGER VARIABLE- A. IF THE VARIABLE IS UNDEFINED AND     FT105230
*                                B V A R B IS ON (INDICATING THAT IT WAS FT105240
*                                USED IN A VARIABLE CONTEXT) THIS VARIABLEFT105250
*                                IS STACKED WITH OTHER UNDEFINED VARIABLESFT105260
*                             B. IF DEFINED OR B V A R B OFF, A TEST FOR FT105270
*                                FOR THE APPEARANCE OF THIS VARIABLE IN ANFT105280
        REM                                                             FT105290
*                                EQUIVALENCE STATEMENT IS MADE.          FT105300
```

```
*                          C. SUCCESSIVE TESTS ARE MADE FOR THE          FT105310
*                             VARIABLE'S APPEARANCE IN A COMMON          FT105320
*                             STATEMENT, WHETHER OR NOT IT IS AN ARRAY   FT105330
*                             AND WHETHER OR NOT IT IS A '2-CELL'        FT105340
*                             VARIABLE (I.E. DOUBLE PRECISION OR         FT105350
*                             COMPLEX).  IN ANY CASE, THE VARIABLE IS    FT105360
*                             DEFINED APPROPRIATELY IN THE ADDRESS OF    FT105370
*                             ITS E Q U I V ENTRY.                       FT105380
        REM                                                              FT105390
*          REAL VARIABLE- A. SAME ACTION AS THAT FOR INTEGER VARIABLE.   FT105400
        REM                                                              FT105410
*          NONE OF THESE- A. A CATASTROPHIC ERROR MESSAGE IS ISSUED.     FT105420
        SPACE   3                                                        FT105430
SYPASS  AXT     102,1           ZERO OUT THE ERROR CELLS                 FT105440
        STZ     CARD+102,1                                               FT105450
        TIX     *-1,1,1                                                  FT105460
        REM                                                              FT105470
        CAL     LVARBS          SET IVARBS TO VALUE-1 OF LVARBS ON       FT105480
        SUB     =1              ENTRY TO SYMBOL TABLE PASS               FT105490
        SLW     IVARBS                                                   FT105500
        AXT     LSYMTB,1        SET XR1 TO LENGTH OF SYMBOL TABLE        FT105510
        ZET     SYMTAB,1        IS THIS SYMBOL TABLE ENTRY ZERO          FT105520
        TRA     *+3             NO, GO EXAMINE EQUIV ENTRY               FT105530
SPASS1  TIX     *-2,1,1         YES, TIX THROUGH SYMBOL TABLE            FT105540
        TRA     ASCMN2          GO GET LCOMMN                            FT105550
        STZ     ERASE1          INITIALIZE DOUBLE/COMPLEX SWITCH         FT105560
        LDI     EQUIV,1         PUT EQUIV ENTRY IN INDICATORS            FT105570
        IIL     700000          INVERT LEFT-MOST 3 BITS                  FT105580
        LFT     BEXTF           IS THIS AN EXTERNAL SYMBOL               FT105590
        TRA     SPASS4          YES, GO PROCESS IT                       FT105600
        LNT     /MLABL*700000   NO, IS THIS A LABEL                      FT105610
        TRA     SPASSA          NO, GO SEE IF IT IS A STRING             FT105620
        LNT     BLHSX           YES, HAS THIS LABEL BEEN DEFINED         FT105630
        ERRORY  0               UNDEFINED LABEL                          FT105640
        LNT     BPATH           YES, IS THIS LABEL IN PATH OF FLOW       FT105650
        ERRORY  3               NO PATH OF FLOW                          FT105660
        TRA     SPASS1          YES, GO CONTINUE EQUIV PROCESSING        FT105670
        REM                                                              FT105680
SPASSA  LNT     /MSTRG*700000   IS THIS A STRING                         FT105690
        TRA     SPASSB          NO, GO SEE IF IT IS A SUBR. ARGUMENT     FT105700
        LNT     BLHSX           YES, HAS THE STRING BEEN DEFINED         FT105710
        ERRORY  1               UNDEFINED STRING                         FT105720
        PIA                     ACC = EQUIV ENTRY                        FT105730
        PAX     ,4              XR4 = POOL POINTER FOR THIS STRING       FT105740
        CAL     POOL,4          SET ERASE = LENGTH OF THIS STRING        FT105750
        SLW     ERASE           X                                        FT105760
        LXA     STSW1,2         GET POINTER TO LAST STRING ENCOUNTERED   FT105770
        TXL     *+3,2,0         TRANSFER IFF THIS IS FIRST STRING        FT105780
        PXD     ,4              OTHERWISE CHAIN THIS STRING TO LAST      FT105790
        STD     POOL,2          X                                        FT105800
        CAL     LSTRIN          GET LENGTH OF STRINGS SO FAR             FT105810
        ACL     ERASE           DEFINE THIS STRING                       FT105820
        SUB     =1              DEF. IS LSTRIN+LEN-1                     FT105830
        STA     EQUIV,1         THIS IS DEFINITION OF THIS STRING        FT105840
        STD     POOL,4          SET LAST DECR. IN CHAIN TO ZERO          FT105850
        SXA     STSW1,4         SAVE POOL POINTER FOR NEXT LINK          FT105860
        TXH     *+2,2,0         TRANSFER IF THIS IS OTHER THAN FIRST     FT105870
        SXA     STSW2,4         SAVE POOL POINTER OF FIRST STRING        FT105880
        ADD     =1                                                       FT105890
        SLW     LSTRIN          X                                        FT105900
```

```
        TRA     SPASS1          GO CONTINUE PROCESSING                  FT105910
SPASSB  LFT     BARGT           IS THIS A SUBROUTINE ARGUMENT           FT105920
        TRA     SPASSE          YES.  GO CHECK IF ALSO IN COMMON        FT105930
        LNT     /MINTG*700000   NO, IS THIS AN INTEGER VARIABLE         FT105940
        TRA     SPASSD          NO, GO SEE IF REAL VARIABLE             FT105950
SPASSC  LFT     BLHSX           YES, HAS IT BEEN DEFINED                FT105960
        TRA     *+3                                                     FT105970
        LFT     BVARB                                                   FT105980
        ERRORY  2               UNDEFINED VARIABLE                      FT105990
        LFT     BEQUV           YES, DOES IT APPEAR IN AN EQUIVALENCE   FT106000
        TRA     SPASSG          YES, GO SEE IF COMMON VARIABLE          FT106010
        LNT     BCOMN           NO, DOES IT APPEAR IN COMMON STMNT.     FT106020
        TRA     SPASS2          NO, GO SEE IF IT IS AN ARRAY            FT106030
SPASSF  STI     ERASE           YES.  SAVE THE INDICATORS               FT106040
        LDI     EQUIV,1         RESET INDICATORS TO EQUIV ENTRY         FT106050
```

```
        EJECT                                                           FT106060
*       POST PASS 1 PROCESSOR FOR NON-EQUIVALENCED COMMON VARIABLES     FT106070
*                                                                       FT106080
*       ENTRY...  ASCMNX  AND  ASCMN2                                   FT106090
*       THIS PROCESSOR IS USED AFTER THE VARIABLES IN THE               FT106100
*       EQUIVALENCE STACK EQUTBL HAVE BEEN ASSIGNED TO STORAGE,         FT106110
*       AND DURING THE POST PASS 1 SYMBOL TABLE SCAN.                   FT106120
*       THIS ROUTINE IS ENTERED AT LOCATION ASCMNX FOR EACH NON-        FT106130
*       EQUIVALENCED COMMON VARIABLE.  THE CORRESPONDING ENTRY IN DIMTBL FT106140
*       (FIRST WORD) IS STACKED IN THE COMMON STACK COMTBL (LOCATED     FT106150
*       OVER 'COLUMN' BEYOND THE LITERALS).                             FT106160
        REM                                                             FT106170
*       AT THE END OF THE SYMBOL TABLE SCAN, THE ROUTINE IS AGAIN       FT106180
*       ENTERED AT LOCATION ASCMN2, AND COMTBL IS SORTED ON THE VALUE   FT106190
*       OF THE COMMON POINTER CPOINT (LOCATED IN THE                    FT106200
*       DECREMENTS).  EACH ENTRY IN COMTBL, IN ORDER, IS ASSIGNED       FT106210
*       TO COMMON STORAGE.  ON EXIT, THE COMMON COUNTER LCOMMN          FT106220
*       HAS THE ADDRESS (RELATIVE TO TOP OF COMMON) OF THE NEXT         FT106230
*       AVAILABLE STORAGE CELL.                                         FT106240
        REM                                                             FT106250
*       THE STORAGE ASSIGNMENT (DEFINITION) OF THE VARIABLE IS          FT106260
*       PLACED IN THE EQUIV WORD ADDRESS.  NOTE THAT A POSSIBLE         FT106270
*       POINTER TO THE DIMTBL WILL BE LOST AT THIS POINT.               FT106280
        REM                                                             FT106290
        REM                                                             FT106300
ASCMNX  PIA                     GET E Q U I V WORD                      FT106310
        PAX     ,2              SET POINTER                             FT106320
        CAL     DIMTBL,2        GET COMMON INFORMATION FROM DIMTBL      FT106330
        REM                                                             FT106340
        LXA     COMCNT,2        GET COMMON TABLE POSITION POINTER       FT106350
        TXI     *+1,2,1         BUMP                                    FT106360
        SXA     COMCNT,2        RESTORE POINTER                         FT106370
        REM                                                             FT106380
        TXL     *+2,2,LCOLMN    TEST FOR TABLE OVERFLOW                 FT106390
        REM                                                             FT106400
        STR     MSG100,,FINISH                                          FT106410
        REM                                                             FT106420
        SLW     COLUMN+666,2    PUT COMMON ENTRY INTO THE COMTBL        FT106430
        REM                                                             FT106440
        TRA     SPASS1                                                  FT106450
```

```
        EJECT                                                          FT106460
SPASS2  LNT     BARRY           IS THIS AN ARRAY                       FT106470
        TRA     SPASS3          NO, GO SET EQUIV ENTRY AND BUMP LVARBSFT106480
        PIA                     YES, SET XR2 TO DIMTBL ENTRY LOCATION  FT106490
        PAX     ,2              X                                      FT106500
        CAL     LVARBS          GET CURRENT LVARBS                     FT106510
        ACL     DIMTBL-1,2      LVARBS + ARRAY SIZE                    FT106520
        LFT     BDOUB           IS IT A DOUBLE-LENGTH VARIABLE.        FT106530
        ACL     DIMTBL-1,2      YES.  DOUBLE THE STORAGE ALLOCATION    FT106540
SPASSK  STA     LVARBS          RECORD THE NEW LVARBS                  FT106550
        SUB     =1              REDUCE THE COUNT                       FT106560
        STA     EQUIV,1         DEFINE THE ARRAY IN EQUIV              FT106570
        TRA     SPASS1          GO CONTINUE EQUIV PROCESSING           FT106580
SPASS3  CAL     LVARBS          GET NUMBER LOW-CORE VARIABLES SO FAR   FT106590
        ACL     ONE             BUMP LVARBS BY 1                       FT106600
        LFT     BDOUB           IS THIS A 2-CELL VARIABLE              FT106610
        ACL     ONE             YES, BUMP LVARBS BY 1                  FT106620
        TRA     SPASSK                                                 FT106630
SPASS4  LFT     BARGT           IS THIS EXTERNAL SYMBOL AN ARGUMENT.   FT106640
        TRA     SPASS1          YES.  SKIP OUT                         FT106650
        LXA     TVSW1,4       CHAIN EXTERNAL REFERENCES THRU EQUIV     FT106660
        TXL     *+3,4,0         TRANSFER IF THIS IS FIRST EXTERNAL     FT106670
        PXD     ,1              PUT CURRENT EQUIV POINTER INTO         FT106680
        STD     EQUIV,4          DECREMENT OF LAST EQUIV ENTRY FOR     FT106690
        REM                      EXTERNAL                              FT106700
        CAL     LTVECT          SET ADDRESS OF THIS EQUIV ENTRY        FT106710
        ACL     ONE             BUMP LTVECT BY ONE                     FT106720
        SLW     EQUIV,1          TO T. V. ELEMENT NUMBER               FT106730
        SXA     TVSW1,1         SAVE CURRENT EQUIV POINTER FOR NEXT    FT106740
        STA     LTVECT          X                                      FT106750
        TXH     *+2,4,0         SAVE POOL PTR. OF FIRST T. V. ELEMENT  FT106760
        SXA     TVSW2,1         X                                      FT106770
        LDI     SYMTAB,1        IF FIRST CHARACTER OF EXTERNAL NAME    FT106780
        LFT     700000           IS 4 OR 1 CHANGE THIS CHARACTER TO    FT106790
        TRA     SPASS1           'D' OR 'I' RESPECTIVELY               FT106800
        SIL     200000           X                                     FT106810
        LFT     10000            X                                     FT106820
        SIL     100000           X                                     FT106830
        STI     SYMTAB,1         X                                     FT106840
        TRA     SPASS1          GO CONTINUE EQUIV PROCESSING           FT106850
SPASSD  LNT     /MREAL*700000   IS THIS A REAL VARIABLE                FT106860
        ERRORY  4               INVALID EQUIV ENTRY                    FT106870
        TRA     SPASSC          YES, GO PROCESS THIS REAL VARIABLE     FT106880
        REM                                                            FT106890
SPASSG  LFT     BCOMN                                                  FT106900
        TRA     SPASS1                                                 FT106910
        PIA                                                            FT106920
        ANA     =077777                                                FT106930
        SSM                                                            FT106940
        ADD     IVARBS                                                 FT106950
        STA     EQUIV,1                                                FT106960
        TRA     SPASS1                                                 FT106970
SPASSE  LFT     BCOMN           IS THE ARGUMENT IN COMMON.             FT106980
        TRA     SPASSF          YES.  GO ENTER INTO THE COMMON STACK   FT106990
        TRA     SPASS1          NO.  NO FURTHER PROCESSING NECESSARY   FT107000
```

```
         TTL     P O S T  P A S S  1  -  A S C M N 2                    FT107010
*        PASS OVER EQUIV TABLE COMPLETED.  SORT THE COMMON STACK.       FT107020
*        ASSIGN STORAGE TO COMMON VARIABLES.  STORAGE DEFINITION IS     FT107030
*        PLACED IN ADDRESS OF EQUIV WORD.  NOTE THAT THE DIMTBL POINTER FT107040
*        FOR ARRAYS IS LOST AT THIS POINT.                              FT107050
         REM                                                            FT107060
ASCMN2   NZT     COMCNT          WERE ANY COMMON VARIABLES PUT IN TABLE. FT107070
         TRA     OBJLEN               NO, GO TO OBJLEN                  FT107080
         REM                                                            FT107090
         REM                                                            FT107100
         LAC     COMCNT,1        GET NUMBER OF ITEMS IN COMMON TABLE    FT107110
         TXI     *+1,1,COLUMN+666    GET ABSOLUTE ADDRESS OF END OF TABLE FT107120
         SXA     *+2,1                                                  FT107130
         REM                                                            FT107140
         CALL    SORT,**,COLUMN+666-1     SORT THE COMMON TABLE         FT107150
         REM                                                            FT107160
         REM                                                            FT107170
         LXA     COMCNT,1        GET NUMBER OF COMMON VARIABLES IN STACK FT107180
         REM                                                            FT107190
ASCMN4   CAL     COLUMN+666,1    GET COMMON STACK ENTRY                 FT107200
         PAX     ,4              GET EQUIV POINTER                      FT107210
         REM                                                            FT107220
         CAL     EQUIV,4         GET EQUIV ENTRY FOR THIS COMMON VARIABLE FT107230
         PAI                                                            FT107240
         PAX     ,2              GET DIMTBL POINTER                     FT107250
         CAL     LCOMMN                                                 FT107260
         STA     EQUIV,4         ASSIGN STORAGE POSITION TO THIS VARIABLE FT107270
         LFT     BARRY           IS IT AN ARRAY.                        FT107280
         TRA     ASCMN3          YES                                    FT107290
         REM                                                            FT107300
         CAL     =1              LENGTH IS 1                            FT107310
         REM                                                            FT107320
ASCMN6   LFT     BDOUB           IS IT DOUB. PREC. OR COMPLEX.          FT107330
         ALS     1               YES.  DOUBLE LENGTH OF STORAGE         FT107340
         REM                                                            FT107350
         ADD     LCOMMN          BUMP COMMON COUNTER                    FT107360
         STO     LCOMMN                                                 FT107370
         TIX     ASCMN4,1,1      RETURN FOR ANOTHER COMMON TABLE ENTRY  FT107380
         REM                                                            FT107390
ASCMN5   TRA     OBJLEN               AT END, GO TO OBJLEN              FT107400
         REM                                                            FT107410
ASCMN3   CAL     DIMTBL-1,2      GET LENGTH OF ARRAY                    FT107420
         ANA     =077777         SAVE ADDRESS FIELD (LENGTH) ONLY       FT107430
         TRA     ASCMN6                                                 FT107440
```

```
        TTL     P O S T  P A S S  1  -  O B J L E N                    FT107450
*       THE FOLLOWING ROUTINE CALCULATES THE QUANTITIES,               FT107460
        SPACE   2                                                      FT107470
*         C M N B R K = T O P C O M - L C O M M N                      FT107480
*                       (COMMON BREAK)                                 FT107490
        REM                                                            FT107500
*         L T V P R L = L T V E C T + L P R O L G                      FT107510
*                       (TRANSFER VECTOR + PROLOGUE)                   FT107520
        REM                                                            FT107530
*         L T V P R T = L T V P R L + P C O U N T                      FT107540
*                       (LENGTH THRU TEXT)                             FT107550
        REM                                                            FT107560
*         L T P T S C = L T V P R T + L C O N S T                      FT107570
*                       (LENGTH THRU CONSTANTS)                        FT107580
        REM                                                            FT107590
*         L T H R U S = L T P T S C + L S T R I N                      FT107600
*                       (LENGTH THRU STRINGS)                          FT107610
        REM                                                            FT107620
*         L T H R U W = L T H R U S + L W O R K S                      FT107630
        REM                                                            FT107640
*         P G M B R K = L T H R U W + L V A R B S                      FT107650
        REM                                                            FT107660
*                       (PROGRAM BREAK)                                FT107670
        SPACE   2                                                      FT107680
*       IN ADDITION TO THIS, THE OBJECT PROGRAM ENTRY POINT (E N T R Y) FT107690
*       IS CALCULATED.  ALSO, IF P G M B R K EXCEEDS C M N B R K, A    FT107700
*       CATASTROPHIC ERROR MESSAGE IS ISSUED.                          FT107710
        SPACE   2                                                      FT107720
*       FINALLY, THE VARIOUS WORDS FOR THE PROGRAM CARD ARE FORMATTED  FT107730
*       AND READIED FOR PUNCHING (THIS CARD IS NOT ACTUALLY PUNCHED    FT107740
*       UNTIL JUST BEFORE THE TRANSFER VECTOR IS 'STASHED').           FT107750
        SPACE   3                                                      FT107760
OBJLEN  CAL     TOPCOM                                                 FT107770
        SUB     LCOMMN                                                 FT107780
        SLW     CMNBRK                                                 FT107790
        LXA     LWORKS,4        SET UP FOR COMPLEMENTATION OF          FT107800
        TXI     **+1,4,-1        WORKING STORAGE RELOCATION IN         FT107810
        SXD     FLAGW+2,4        C I T P R O                           FT107820
        CAL     LTVECT                                                 FT107830
        ACL     LPROLG                                                 FT107840
        SLW     LTVPRL                                                 FT107850
        ACL     PCOUNT                                                 FT107860
        SLW     LTVPRT                                                 FT107870
        ACL     LCONST                                                 FT107880
        SLW     LTPTSC                                                 FT107890
        ACL     LSTRIN                                                 FT107900
        SLW     LTHRUS                                                 FT107910
        ACL     LWORKS                                                 FT107920
        SLW     LTHRUW                                                 FT107930
        ACL     LVARBS                                                 FT107940
        SLW     PGMBRK                                                 FT107950
        LAS     CMNBRK                                                 FT107960
        ERROR   32,OUTUND       OBJECT PROGRAM TOO LONG                FT107970
        NOP                     X                                      FT107980
        SUB     =1              PUT END-OF-PROGRAM INTO DECREMENT      FT107990
        ALS     18              OF THE FLAGN INSTRUCTION.              FT108000
        STD     FLGN2                                                  FT108010
        CAL     PCOUNT          SET LENGTH OF TEXT                     FT108020
        SUB     LARSTS                                                 FT108030
        SLW     LTEXTS          X                                      FT108040
```

```
        CAL     LTVECT          CALCULATE ENTRY POINT RELATIVE TO ZEROFT108050
        NZT     SUBR              IF MAIN PROGRAM,                    FT108060
        TRA     OBJL1               ENTRY = LTVECT + LARSTS           FT108070
        ACL     NOXRS             IF SUBPROGRAM,                      FT108080
        ZET     RTNFLG          NO XR'S SAVED IF NO RETURN            FT108090
        ACL     ONE                 ENTRY = LTVECT + NOXRS + 1        FT108100
        NZT     SNAME            X                                    FT108110
OBJL1   ACL     LARSTS          X                                     FT108120
        STA     ENTRY            X                                    FT108130
        REM                                                           FT108140
        REM     THE FOLLOWING INSTRUCTIONS SET-UP THE PROGRAM CARD.   FT108150
        REM      THE CARD IS ACTUALLY PUNCHED IN THE POST PASS 1 DRIVER.FT108160
        REM                                                           FT108170
        CAL     LTVECT                                                FT108180
        ALS     18                                                    FT108190
        ACL     PGMBRK                                                FT108200
        SLW     PCWD3                                                 FT108210
        ACL     PCWD1                                                 FT108220
        ACL     CMNBRK                                                FT108230
        ACL     SNAME                                                 FT108240
        ACL     ENTRY                                                 FT108250
        SLW     PCWD2                                                 FT108260
```

```
        TTL     P O S T  P A S S  1  -  O U T U N D                     FT108270
*       THE FOLLOWING ROUTINE FORMATS AND OUTPUTS ERRORS DETECTED IN    FT108280
*       S Y P A S S OF THE FOLLOWING TYPES,                             FT108290
        SPACE   2                                                       FT108300
*          1. STATEMENTS WITH NO PATH OF FLOW                           FT108310
        REM                                                             FT108320
*          2. UNDEFINED VARIABLES                                       FT108330
        REM                                                             FT108340
*          3. UNDEFINED STRINGS (FORMATS)                               FT108350
        REM                                                             FT108360
*          4. UNDEFINED STATEMENT LABELS                                FT108370
        SPACE   3                                                       FT108380
OUTUND  AXT     4,1             SET TO PICK UP FOUR TYPES OF ERRORS     FT108390
  UND1  CLA     TBUND+1,1                                               FT108400
        PDX     ,2                                                      FT108410
        TXL     UND5+1,2,0                                              FT108420
        TXI     *+1,2,-1                                                FT108430
        SXD     UND3,2                                                  FT108440
        SXA     UND5,1                                                  FT108450
        STA     UND2                                                    FT108460
        CLA     MSUND+1,1                                               FT108470
        STO     UND2-4                                                  FT108480
        AXT     *+3,4           SET UP THE RETURN FROM THE              FT108490
        SXA     EREX,4          PRINT SECTION OF THE ERROR ROUTINE.     FT108500
        TRA     ERR1            GO TO ERROR ROUTINE TO SET NOCODE, ETC  FT108510
        REM                     RETURN HERE FROM ERROR ROUTINE          FT108520
        TSX     LIST,4                                                  FT108530
        PON     ..,,..                                                  FT108540
        AXT     0,1                                                     FT108550
        AXT     0,2                                                     FT108560
        LDQ     BLANKS                                                  FT108570
  UND2  CLA     ..,1                                                    FT108580
        STO     LIN10+1,2                                               FT108590
        STQ     LIN10,2                                                 FT108600
        TXI     *+1,2,-2                                                FT108610
        TXI     *+1,1,1                                                 FT108620
  UND3  TXH     UND4,1,..                                               FT108630
        TXH     UND2,2,-16                                              FT108640
        TSX     LIST,4                                                  FT108650
        PZE     LIN10,,16                                               FT108660
        TRA     UND2-2                                                  FT108670
        REM                                                             FT108680
  UND4  PXA     ,2                                                      FT108690
        PAC     ,2                                                      FT108700
        SXD     *+2,2                                                   FT108710
        TSX     LIST,4                                                  FT108720
        PZE     LIN10,,..                                               FT108730
  UND5  AXT     ..,1                                                    FT108740
        TIX     UND1,1,1                                                FT108750
        AXT     16,1                                                    FT108760
        CAL     BLANKS                                                  FT108770
        SLW     LIN10+17,1                                              FT108780
        TIX     *-1,1,1                                                 FT108790
        TRA     TSTPP2          GO TEST 'NOCODE'                        FT108800
        SPACE   3                                                       FT108810
        PON     UNMS4,,7                                                FT108820
        PON     UNMS3,,5                                                FT108830
        PON     UNMS2,,6                                                FT108840
MSUND   PON     UNMS1,,6                                                FT108850
        REM                                                             FT108860
```

UNMS1	BCI	6, ****	UNDEFINED STATEMENT LABELS-	FT108870
UNMS2	BCI	6, ****	UNDEFINED FORMAT NUMBERS-	FT108880
UNMS3	BCI	5, ****	UNDEFINED VARIABLES-	FT108890
UNMS4	BCI	7, ****	STATEMENTS WITH NO PATH OF FLOW-	FT108900

```
TSTPP2 ZET     NOCODE          EXIT IF FATAL ERROR FOUND          FT108920
       TRA     FINISH          OTHERWISE GO ON TO TSTPST          FT108930
```

```
        TTL     P O S T  P A S S  1  -  T S T P S T                    FT108940
*       THE FOLLOWING ROUTINE PUNCHES THE OBJECT PROGRAM SYMBOL TABLE   FT108950
*       INTO COLUMN BINARY CARDS AS REQUIRED BY THE FMS-II OBJECT TIME  FT108960
*       DEBUGGING PACKAGE.  THE FORMAT OF THESE CARDS CAN BE FOUND IN,  FT108970
        REM                                                             FT108980
*          IBM 7090/7094 PROGRAMMING SYSTEMS - FORTRAN II OPERATIONS    FT108990
*                        (PUBLICATION NO. C28-6066-4)                   FT109000
        REM                                                             FT109010
*       THE SYMBOLS REQUIRED IN THIS TABLE ARE ASSEMBLED INTO THE       FT109020
*       LOW-CORE PORTION OF THE POOL REGION (I.E. BEGINNING AT LOCATION FT109030
*       C O L U M N AND PROCEEDING UPWARD).  THESE ARE THEN SORTED,     FT109040
*       FORMATTED AND OUTPUT ON SYSBIN.                                 FT109050
        SPACE   3                                                       FT109060
TSTPST  LDI     PRGBOX          WAS 'SYMBOL TABLE' CC ENCOUTERED        FT109070
        RNT     STCC             BY SYSTEM MONITOR                      FT109080
        TRA     OUTPGC-1        NO, GO PUNCH PORGRAM CARD               FT109090
        CLS     NXTLOC          YES, PREPARE TO PUNCH SYMBOL TABLE      FT109100
        ALS     18              GET LENGTH OF AVAILABLE POOL REGION     FT109110
        ADD     MAXNLC          X                                       FT109120
        STD     PST03           SAVE THIS IN PST03                      FT109130
        AXT     LSYMTB,1        PREPARE TO SCAN COMPLETE SYMTAB         FT109140
        AXT     0,2             POOL POINTER FOR STORING SYMBOLS        FT109150
        AXT     0,4             2 X (NO. SYMBOLS STORED)                FT109160
PST01   ZET     SYMTAB,1        IS THERE A SYMTAB ENTRY HERE            FT109170
        TRA     *+3             YES, SKIP NEXT COUPLE OF INSTRUCTIONS   FT109180
PST02   TIX     PST01,1,1       NO, SCAN ALL OF SYMTAB                  FT109190
        TRA     PST05           AT END OF SYMTAB GO TO PST05            FT109200
        SPACE   2                                                       FT109210
        LDI     EQUIV,1         GET EQUIV ENTRY FOR THIS SYMBOL         FT109220
        LFT     700000          IS THIS AN EXTERNAL FORMULA NUMBER      FT109230
        TRA     PST04           NO, GO SEE IF IT IS A VARIABLE          FT109240
        SIL     300000          YES, SET PREFIX = 3                     FT109250
        CAL     SYMTAB,1        DO NOT ENTER IF A T. V. ELEMENT         FT109260
        ANA     =0600000000000  X                                       FT109270
        TNZ     PST02           X                                       FT109280
        TXI     *+1,4,2         IS THERE ROOM FOR THIS ENTRY IN POOL     FT109290
PST03   TXH     PST15,4,..      EXIT IF POOL REGION OVERFLOW            FT109300
        RIL     77777           YES, CLEAR OUT EQUIV GARBAGE            FT109310
        RIR     700000          X                                       FT109320
        STI     COLUMN+1,2      STORE THIS EQUIV WORD IN POOL REGION    FT109330
        CAL     SYMTAB,1        NOW GET THE ASSOCIATED SYMBOL           FT109340
        SLW     COLUMN,2         AND PUT IT INTO THE POOL REGION ALSO   FT109350
        TXI     PST02,2,-2      BUMP POOL PTR AND GO FOE NEXT SYMBOL    FT109360
PST04   LNT     200000          IS THIS SYMBOL A VARIABLE               FT109370
        TRA     PST02           NO, (STRING) IGNORE THIS SYMBOL         FT109380
        LFT     BARGT           DO NOT ENTER IF SUBPROGRAM ARGUMENT     FT109390
        TRA     PST02           X                                       FT109400
        RIL     700000          YES, TURN OFF PREFIX BITS               FT109410
        LFT     BCOMN           IF COMMON SET PREFIX = 7                FT109420
        SIL     200000          IF NON-COMMON SET PREFIX = 5            FT109430
        SIL     500000          X                                       FT109440
        TRA     PST03-1         GO STORE THIS SYMBOL                    FT109450
        SPACE   3                                                       FT109460
PST05   SXD     PST08,2         CALCULATE UPPER LIMIT OF POOL USED      FT109470
        TXL     OUTPGC-1,2,0    EXIT IF SYMBOL TABLE NULL               FT109480
        AXT     COLUMN,2        X                                       FT109490
        SXD     *+1,4           X                                       FT109500
PST16   TXI     *+1,2,..        X                                       FT109510
        TXI     *+1,2,-1        X                                       FT109520
        SXA     *+3,2           SAVE THIS IN SORT CALL                  FT109530
```

```
PST06 CALL    SORT1,COLUMN+1,..                                        FT109540
      CAL     LABEL             SAVE OBJECT DECK LABEL                 FT109550
      SLW     ERASE             X                                      FT109560
      CAL     =HST              SET-UP FOR 'ST' LABEL ON SYMTAB CARDS  FT109570
      SLW     LABEL             X                                      FT109580
      TSX     CDLABL,4          SET-UP COLS. 73-80                     FT109590
      CAL     SNAME             GET SUBPROGRAM NAME (OR ZEROES)        FT109600
      SLW     9LEFT+2           THIS IS WORD 3 OF FIRT SYMTAB CARD     FT109610
      LXD     PST05+3,1         GET NUMBER OF SYMBOLS IN SYMBOL TABLE  FT109620
      PXD     ,1                X                                      FT109630
      ARS     1                 X                                      FT109640
      ANA     =077777000000     X                                      FT109650
      SLW     9LEFT+3           THIS IS WORD 4 OF FIRST CARD           FT109660
      ACL     9LEFT+2           GET (PARTIAL) CHECKSUM                 FT109670
      SLW     SUM               X                                      FT109680
      AXT     20,4              SET TO STASH SYMBOLS AND EQUIV'S       FT109690
      AXT     0,2                INTO BINARY CARD IMAGES               FT109700
PST07 CAL     COLUMN,2          GET NEXT SYMBOL                        FT109710
      SLW     9LEFT+24,4        STORE THIS IN CARD IMAGE               FT109720
      ACL     SUM               ACCUMULATE SUM                         FT109730
      SLW     SUM               X                                      FT109740
      LDI     COLUMN+1,2        SET IND AND LAC TO EQUIV ENTRY         FT109750
      CAL     COLUMN+1,2        X                                      FT109760
      ANA     =077777           CLEAR OUT PREFIX                       FT109770
      LFT     400000            IS THIS AN EXTERNAL FORMULA NUMBER     FT109780
      TRA     PST12             NO, GO TO PST12                        FT109790
      ACL     LTVPRL            BUMP BY LENGTH OF T.V. AND PROLOGUE    FT109800
      ORA     =0200004000000    PUT IN IBM'S REQUIRED BITS             FT109810
      SLW     9LEFT+25,4        PUT THIS INTO CARD IMAGE               FT109820
      ACL     SUM               ACCUMULATE CHECKSUM                    FT109830
      SLW     SUM               X                                      FT109840
      TXI     *+1,2,-2          ARE THERE MORE SYMBOLS TO GO           FT109850
PST08 TXL     PST09,2,..        NO, GO TO PST09                        FT109860
      TIX     PST07,4,2         IS THIS CARD FULL -  NO, GO TO PST07   FT109870
      CAL     STW1              SET WORD 1 FOR IBM                     FT109880
      SLW     9LEFT             X                                      FT109890
IOS20 EQU     *                                                        FT109900
      WRITE   *SYSBIN,BINARY,9LEFT,24,COL73,3   WRITE THIS CARD        FT109910
      STL     IOEX                                                     FT109920
      BRA     WRITE,,*+4                                               FT109930
      PTW     SYSBIN,,BINARY                                           FT109940
      PZE     9LEFT,,24                                                FT109950
      MZE     COL73,,3                                                 FT109960
END20 REM                                                              FT109970
      TSX     CDLABL,4          BUMP CARD SEQUENCE NUMBER              FT109980
      AXT     22,4              SET FOR 11 SYMBOLS/CARD                FT109990
      STZ     SUM               RESET CHECKSUM                         FT110000
      TRA     PST07             GO START ON NEXT CARD                  FT110010
      SPACE   2                                                        FT110020
PST09 AXT     0,1               AT END FILL LAST CARD WITH BLANKS      FT110030
      TXI     *+1,4,-2          X                                      FT110040
      TXL     PST10,4,0         X                                      FT110050
      STZ     9LEFT+24,4        X                                      FT110060
      STZ     9LEFT+25,4        X                                      FT110070
      TXI     *+1,1,-2          X                                      FT110080
      TIX     *-3,4,2           X                                      FT110090
PST10 TXI     *+1,1,22          GET WORD COUNT                         FT110100
      PXD     ,1                X                                      FT110110
      ORA     =0600500000000    COMPLETE WORD 1                        FT110120
      SLW     9LEFT             STORE THIS IN CARD IMAGE               FT110130
```

```
IOS21   EQU     *                                                       FT110140
        WRITE   *SYSBIN,BINARY,9LEFT,24,COL73,3   WRITE THIS LAST CARD  FT110150
        STL     IOEX                                                    FT110160
        BRA     WRITE,,*+4                                              FT110170
        PTW     SYSBIN,,BINARY                                          FT110180
        PZE     9LEFT,,24                                               FT110190
        MZE     COL73,,3                                                FT110200
END21   REM                                                             FT110210
        CAL     ERASE           RESTORE OBJECT DECK LABEL               FT110220
        SLW     LABEL           X                                       FT110230
        STZ     TLSWTH          INDICATE TO RE-SEQUENCE                 FT110240
        CAL     =0200500000000  RESET 9LEFT FOR 'STASH'                 FT110250
        SLW     9LEFT           X                                       FT110260
        AXT     23,1            CLEAR COLUMN BINARY IMAGE               FT110270
        STZ     9LEFT+24,1      X                                       FT110280
        TIX     *-1,1,1                                                 FT110290
        STL     STOUT           INDICATE ST OUT FOR M A P               FT110300
        SPACE   3                                                       FT110310
        LXD     PST16,1         CLEAR POOL AREA USED ABOVE COLUMN       FT110320
        TXL     PST11,1,666     X                                       FT110330
        TXI     *+1,1,-666      X                                       FT110340
        AXT     0,2             X                                       FT110350
        STZ     COLUMN+666,2    X                                       FT110360
        TXI     *+1,2,-1                                                FT110370
        TIX     *-2,1,1         X                                       FT110380
PST11   TRA     OUTPGC-1        EXIT P S T                              FT110390
        REM                                                             FT110400
PST12   LNT     100000          IS THIS A SOURCE PROG. VAR. OF SOME     FT110410
        TRA     PST14            TYPE.  NO, MUST BE ERASABLE            FT110420
        LNT     200000          YES, IS IT A COMMON VARIABLE            FT110430
        TRA     PST13           NO                                      FT110440
        SSM                     YES, DO COMMON RELOCATION               FT110450
        ADD     TOPCOM          X                                       FT110460
        ORA     =0200000000000  PUT IN IBM'S REQUIRED BITS              FT110470
        TRA     PST08-4         GO PUT THIS INTO CARD IMAGE             FT110480
        REM                                                             FT110490
PST13   ACL     LTHRUW          DO LOW-CORE VARIABLE RELOCATION         FT110500
        ORA     =0200002000000                                          FT110510
        TRA     PST08-4         GO STASH THIS ENTRY                     FT110520
        REM                                                             FT110530
PST14   ACL     LTHRUS          DO ERASABLE RELOCATION                  FT110540
        ACL     LVARBS          X                                       FT110550
        TRA     PST08-4         GO STASH THIS ENTRY                     FT110560
        REM                                                             FT110570
PST15   ERROR   38,*+1          POOL AREA OVERFLOW -- NO SYMBOL TABLE   FT110580
        STZ     NOCODE          LAST ERROR NOT FATAL                    FT110590
        TRA     OUTPGC-1        EXIT SYMBOL TABLE SECTION               FT110600
        REM                                                             FT110610
```

```
        TTL     P O S T  P A S S  1  -  C D L A B L                     FT110620
        REM     1)  SUBROUTINE CONVERTS BCD NAME TO COLUMN BINARY.      FT110630
        REM     2)  EACH CALL INCREMENTS SEQUENCE NUMBER.               FT110640
        REM     3)  IF TLSWTH IS SET TO ZERO, ROUTINE RESETS SEQUENCE   FT110650
        REM         NUMBERING TO ZERO AND PICKS UP NEW LABEL FROM 'LABEL'FT110660
        REM                                                             FT110670
        REM     CALLING SEQUENCE                                        FT110680
        REM             SNAME  = BCD LABEL TO BE USED.                  FT110690
        REM             TLSWTH = 0, IF SEQUENCE NUMBERING IS TO BE      FT110700
        REM                          RESET TO ZERO, AND IF NEW NAME     FT110710
        REM                          IS TO BE USED.                     FT110720
        REM                                                             FT110730
        REM             TSX   CDLABL,4                                  FT110740
        REM             (RETURN)                                        FT110750
        REM                                                             FT110760
        REM                                                             FT110770
CDLABL  SXA     LBLOUT,1        SAVE XR*S                               FT110780
        SXA     LBLOUT+1,2                                              FT110790
        SXA     LBLOUT+2,4                                              FT110800
        ZET     TLSWTH          HAS TITLE BEEN CONVERTED TO COL. IMAGE  FT110810
        TRA     SQNO            YES, PUT SEQUENCE NO. IN CARD           FT110820
        STZ     COL73           NO,  ZERO OUT COLUMNS 73 THRU 80.       FT110830
        STZ     COL73+1                                                 FT110840
        STZ     COL73+2                                                 FT110850
        CAL     LABEL           PICK UP THE NEW NAME                    FT110860
        TNZ     *+3             SKIP IF LABEL HAS BEEN FOUND            FT110870
        NZT     SNAME           NO LABEL.  IS IT A SUBROUTINE.          FT110880
        CAL     =H(MAIN)        NO.   SET LABELING TO 'MAIN PROGRAM'    FT110890
        SLW     TMPTL           SAVE NAME                               FT110900
        XCL                                                             FT110910
        AXT     0,1             COUNT TRAILING BLANKS                   FT110920
        AXT     6,2                                                     FT110930
CNTBLK  ZAC                                                             FT110940
        LGL     6                                                       FT110950
        ERA     =060                                                    FT110960
        TXI     *+1,1,-6        ASSUME BLANK FOUND, COUNT WITH MINUS ARITH.FT110970
        TZE     *+2                                                     FT110980
        AXT     0,1             NON-BLANK FOUND, RESET COUNTER          FT110990
        TIX     CNTBLK,2,1      REPEAT LOOP FOR ALL SIX BCD CHARACTERS  FT111000
        TXL     *+3,1,0         SKIP IF NO BLANK CHARACTERS             FT111010
        TXH     *+2,1,-24       IF 4 OR MORE TRAILING BLANKS, RESET COUNTERFT111020
        AXT     -24,1                                                   FT111030
        TXI     *+1,1,-12       CALCULATE NO. OF DIGITS IN SQNCE NO.    FT111040
        SXA     NUMCNT,1        SAVE COUNT OF DIGITS                    FT111050
        TXI     *+1,1,36        CALCULATE SHIFT NEEDED ON SEQUENCE NO.  FT111060
        SXA     STNUM,1         SAVE NO. OF LEADING CHARACTERS (MINUS 2)FT111070
        TXI     *+1,1,12        INITIALIZE CONVERSION ROUT. TO ACTUAL COUNTFT111080
        AXT     3,4             INITIALIZE CARD WORD POINTER FOR TITLE. FT111090
        AXT     24,2            INITIALIZE SHIFT POINTER FOR TITLE.     FT111100
CNVT1   LDQ     TMPTL           BEGIN CONVERSION OF BCD TO COL. IMAGE.  FT111110
CNVT    SXA     COLSFT,2        INITIALIZE SHIFT INSTRUCTION            FT111120
        ZAC                     LOOK UP COL. IMAGE IN TABLE             FT111130
        LGL     6               GET ADDRESS OF ENTRY IN TABLE           FT111140
        STQ     TMPTL           SAVE REST OF CHARACTERS TO BE CONVERTED.FT111150
        XCA                     CALCULATE LOCATION IN CONVERSION TABLE. FT111160
        ZAC                     X                                       FT111170
        DVP     =3                  X                                   FT111180
        XCA                         X                                   FT111190
        ADD     TBLHD               X                                   FT111200
        STA     PCKENT              X                                   FT111210
```

```
         MPY     =12             X                                       FT111220
         XCA                     X                                       FT111230
         STA     BTPTN           X                                       FT111240
PCKENT   CAL     **              PICK UP TABLE ENTRY                     FT111250
BTPTN    ARS     **        MOVE 12 BIT IMAGE TO RIGHT IN AC.             FT111260
         ANA     =07777    REMOVE ALL OTHER BITS                         FT111270
COLSFT   ALS     **        SHIFT TO POSITION.                            FT111280
         ORS     COL73+3,4 STORE IN CARD IMAGE                           FT111290
         TXI     **+1,2,-12 MOVE SHIFT POINTER                           FT111300
         TNX     ENDLP,2,-36 RESET SHIFT POINTER AT END (-12)            FT111310
         TXI     **+1,4,-1 RESET COLUMN POINTER.                         FT111320
ENDLP    TIX     CNVT1,1,6 ARE ALL CHARACTERS CONVERTED AND STORED.      FT111330
         ZET     TLSWTH    YES.  IF SEQUENCE NUMBER JUST INSERTED,       FT111340
         TRA     LBLOUT          RETURN                                  FT111350
         SXA     PNT2,2    TITLE JUST ADDED.  SAVE SHIFT POINTER AND     FT111360
         SXA     PNT4,4  COLUMN POINTER.                                 FT111370
         STL     TLSWTH    TURN OFF TITLE SWITCH                         FT111380
         CLA     =-1       INITIALIZE SEQUENCE NUMBERING                 FT111390
         STO     SQSUM     X                                             FT111400
SQNO     CLA     SQSUM     INCREMENT SEQUENCE NO. IN BCD ARITHMETIC.     FT111410
         ADD     =1                                                      FT111420
         CVR     BCDCNV,,6 CONVERT VIA TABLE LOOKUP                      FT111430
         SLW     SQSUM                                                   FT111440
PNT2     AXT     **,2      ZERO OUT PREVIOUS NUMBER IN CARD IMAGE.       FT111450
         CAL     =0777777770000                                          FT111460
         SXA     **+1,2    SET SHIFT COUNT                               FT111470
         ALS     **        X                                             FT111480
PNT4     AXT     **,4      PICK UP COLUMN POINTER.                       FT111490
         ANS     COL73+3,4 ZERO OUT PARTIAL WORD                         FT111500
         TNX     STNUM1,4,1 ZERO OUT REMAINDER OF CARD                   FT111510
         STZ     COL73+3,4 X                                             FT111520
         TRA     *-2       X                                             FT111530
STNUM1   LDQ     SQSUM     INITIALIZE CONVERSION ROUT. FOR SEQUENCE NO   FT111540
STNUM    RQL     **        SHIFT OFF UNUSABLE LEADING BCD DIGITS.        FT111550
         LXA     PNT4,4    SET COLUMN POINTER                            FT111560
NUMCNT   AXC     **,1      SET COUNTER FOR NUMBER OF DIGITS              FT111570
         TRA     CNVT      PUT NO. IN CARD IMAGE                         FT111580
TBLHD    PZE     CONVTB                                                  FT111590
LBLOUT   AXT     **,1      RESTORE XR*S                                  FT111600
         AXT     **,2                                                    FT111610
         AXT     **,4                                                    FT111620
         TRA     1,4                                                     FT111630
```

```
         TTL     P O S T  P A S S  1  -  O U T P G C                    FT111640
*        THE FOLLOWING MACRO WRITES THE PROGRAM CARD ON SYSBIN.  THE    FT111650
*        VARIOUS WORDS IN THE PROGRAM CARD WERE FORMATTED IN ROUTINE    FT111660
*        O B J L E N.                                                   FT111670
         SPACE   3                                                      FT111680
         TSX     CDLABL,4            SET-UP FOR OBJECT DECK LABELLING   FT111690
IOS22    EQU     *                                                      FT111700
OUTPGC   WRITE   *SYSBIN,1,PCWD1,3,CMNBRK,1,SNAME,1,ENTRY,1,SUM,18,COL73,3 FT111710
OUTPGC   STL     IOEX                                                   FT111720
         BRA     WRITE,,*+8                                             FT111730
         PTW     SYSBIN,,1                                              FT111740
         PZE     PCWD1,,3                                               FT111750
         PZE     CMNBRK,,1                                              FT111760
         PZE     SNAME,,1                                               FT111770
         PZE     ENTRY,,1                                               FT111780
         PZE     SUM,,18                                                FT111790
         MZE     COL73,,3                                               FT111800
END22    REM                                                            FT111810
```

```
        TTL     P O S T  P A S S  1  -  S T V E C T                    FT111820
*       THE FOLLOWING ROUTINE STASHES THE TRANSFER VECTOR INTO BINARY   FT111830
*       CARD IMAGES.  TRANSFER VECTOR ELEMENTS REPRESENTING LIBRARY     FT111840
*       ROUTINES ARE STASHED FIRST AND THEN NON-LIBRARY EXTERNAL        FT111850
*       REFERENCES ARE STASHED.  THE NON-LIBRARY REFERENCES WERE CHAINED FT111860
*       TOGETHER BY THE POST PASS 1 ROUTINE S Y P A S S.                FT111870
        SPACE   2                                                       FT111880
*       THIS ROUTINE HAS BEEN SET UP AS A SUBROUTINE BECAUSE A MODIFIED FT111890
*       VERSION OF THIS ROUTINE IS USED BY ROUTINE M A P TO LIST THE    FT111900
*       TRANSFER VECTOR IN THE STORAGE MAP IN POST PASS 2.              FT111910
        SPACE   3                                                       FT111920
        TSX     STVECT,4        STASH TRANSFER VECTOR                   FT111930
        TRA     GENPRL          GO TO NEXT POST PASS 1 ROUTINE          FT111940
        REM                                                             FT111950
        REM                                                             FT111960
STVECT  NZT     LTVECT          IMMEDIATE EXIT IFF VECTOR IS EMPTY      FT111970
        TRA     1,4             X                                       FT111980
        SXA     EXSTV,4         SAVE XR4 FOR RETURN                     FT111990
        AXT     -2,2            XR2 IS USED AS SPECIAL NAME TABLE PTR.  FT112000
  STV1  CAL     SLNTAB-1,2      GET NEXT TABLE ENTRY                    FT112010
        ZET     SLNTAB-2,2      IS THIS NAME USED IN OBJECT PROGRAM     FT112020
        TSX     STASH,4         YES, GO STASH IT                        FT112030
        TXL     *+2,2,-LSLNTB   NO, TRANSFER IFF AT END OF TABLE        FT112040
        TXI     STV1,2,-2       BUMP POINTER AND GET NEXT NAME          FT112050
        NZT     TVSW2           ARE THERE ANY OTHER VECTOR ELEMENTS     FT112060
        TRA     EXSTV           NO, GO EXIT THE SCENE, MAN              FT112070
        LXA     TVSW2,2         GET PTR. TO FIRST TV EL. IN SYMTAB      FT112080
  STV2  CAL     SYMTAB,2        SET ACC TO THIS ELEMENT OF TV           FT112090
        TSX     STASH,4         STASH THIS ELEMENT                      FT112100
        CAL     EQUIV,2         GET POINTER TO NEXT ELEMENT             FT112110
        PDX     ,2              PUT THIS INTO XR2                       FT112120
        TXH     STV2,2,0        GO GET NEXT IFF THERE IS A NEXT         FT112130
EXSTV   AXT     **,4            AT END, MAKE LIKE A B-I-G BIRD          FT112140
        TRA     1,4              AND FLY AWAY                           FT112150
```

```
      REM   THE FOLLOWING SUBROUTINE GENERATES THE OBJECT-TIME          FT112170
      REM    PROLOGUE CODE.  ON ENTRY TO THIS SECTION, THE PROGRAM      FT112180
      REM    CARD IS OUT ON SYSBIN AND THE TRANSFER VECTOR HAS BEEN     FT112190
      REM    STASHED.                                                   FT112200
      REM                                                               FT112210
      REM   THE PROLOGUE CODE IS BROKEN INTO THREE SECTIONS,            FT112220
      REM         SECT. 1 = XR SAVING                                   FT112230
      REM         SECT. 2 = BODY OF PROLOGUE                            FT112240
      REM         SECT. 3 = APPENDIX                                    FT112250
      REM                                                               FT112260
      REM   SECTION 3 PRECEDES SECTIONS 1 AND 2 IN THE OBJECT-TIME      FT112270
      REM    CODE.  THE PROLOGUE THUS APPEARS AS,                       FT112280
      REM       APPEND AXT   **,1      APPENDIX-- CONTAINS AXT*S AS     FT112290
      REM              AXT   **,2       NECESSARY AND FINAL SUBR.      FT112300
      REM              AXT   **,4       SUBROUTINE EXIT.               FT112310
      REM              TRA   (N+1),4   N = NO. SUBR. ARGUMENTS         FT112320
      REM                                                               FT112330
      REM       ENTRY  SXA   APPEND,1  XR SAVING AS REQUIRED.          FT112340
      REM              SXA   APPEND+1,2 *ENTRY* IS SUBPROGRAM           FT112350
      REM              SXA   APPEND+2,4 ENTRY POINT                     FT112360
      REM                                                               FT112370
      REM              CAL   1,4       BODY OF PROLOGUE-- SETS          FT112380
      REM              STA   --         ADDRESSES IN TEXT AS            FT112390
      REM              STA   --         REQUIRED.                       FT112400
      REM              PAX   ,1                                         FT112410
      REM              SXA   --,1                                       FT112420
      REM              TXI   *+1,1,--                                   FT112430
      REM              SXA   --,1                                       FT112440
      REM              SXA   --,1                                       FT112450
      REM                    .                                         FT112460
      REM                    .                                         FT112470
      REM                    .                                         FT112480
      SPACE  2                                                          FT112490
GENPRL NZT   SUBR      IS THIS A SUBROUTINE COMPILATION                 FT112500
      TRA    PASS2     YES, EXIT FROM THIS SECTION                      FT112510
      NZT    RTNFLG    IS THERE A RETURN STATEMENT.                     FT112520
      TRA    GENPA     NO.                                              FT112530
      LDI    XRFLAG    ARE ANY XR*S TO BE SAVED IN APPENDIX             FT112540
      RFT    700000    YES, GO GENERATE SECTIONS 3 AND 1                FT112550
      TRA    GENP9      OF THE PROLOGUE                                 FT112560
      CAL    NOARGS    SET ACC TO NO. OF SUBR. ARGUMENTS                FT112570
      ACL    TRA4      SET ACC TO TRA (NOARGS),4                        FT112580
      TSX    STASH,4   STASH THIS INSTRUCTION                           FT112590
GENPA NZT    LPBODY    IS SECTION 2 EMPTY                               FT112600
      TRA    GENPD     YES, GET READY TO EXIT THIS SECTION              FT112610
      STZ    ERASE     ERASE = ARG. AND DELTA                           FT112620
      STZ    ERASE1    ERASE1 = ARG.                                    FT112630
      STZ    ERASE2    ERASE2 = DELTA                                   FT112640
      STZ    ERASE3    ERASE3 = XR YET THIS ARG.                        FT112650
      REM              ERASE4 = PCOUNT                                  FT112660
      LXA    PROL2,1   XR1 = PRLTBL + LPROLG                            FT112670
      SXA    *+3,1     SAVE THIS IN FOLLOWING CAL                       FT112680
      LXA    PROLOG+2,2 XR2 = LENGTH OF PROLOG TABLE                    FT112690
      TXI    *+1,2,-1  SET XR2 TO LENGTH OF PRLTBL                      FT112700
GENP1 CAL    **,2      PICK-UP NEXT PRLTBL ENTRY                        FT112710
      STA    ERASE4    SAVE PCOUNT FOR THIS ENTRY IN ERASE4             FT112720
      ANA    ARGDEL    SAVE ARG. AND DELTA                              FT112730
      LAS    ERASE     IS IT SAME AS LAST ARG. AND DELTA                FT112740
      TRA    GENP3     NO, GO TO GENP3                                  FT112750
```

```
        TRA     GENP2           YES, GO TO GENP2                        FT112760
        MACER                   PRLTBL OUT OF SORT                      FT112770
        REM                                                             FT112780
        REM     ... TO HANDLE SAME ARGUMENT AND DELTA                   FT112790
        REM                                                             FT112800
GENP2   ZET     ERASE2          IS THIS DELTA NON-ZERO                  FT112810
        TRA     GENP7           YES, GO TO GENP7                        FT112820
        CAL     STA             NO, GET STAREADY FOR STASHING           FT112830
        ACL     LTVPRL          STA   LTVECT+LPROLG                     FT112840
        ACL     ERASE4          STA   LTVECT + LPROLG + PCOUNT          FT112850
        SSM                     INDICATE ADDRESS RELOCATION             FT112860
        TSX     STASH,4         GO STASH THIS INSTRUCTION               FT112870
        TIX     GENP1,2,1       GO GET NEXT PRLTBL ENTRY                FT112880
        TRA     GENPD           GET READY TO EXIT THIS SECTION          FT112890
        REM                                                             FT112900
        REM     ... TO HANDLE NEW ARG. AND/OR DELTA                     FT112910
        REM                                                             FT112920
GENP3   SLW     ERASE           SAVE NEW ARG. AND DELTA                 FT112930
        ANA     ARGBIT          ISOLATE THIS ARG.                       FT112940
        LAS     ERASE1          IS THIS A NEW ARGUMENT                  FT112950
        TRA     GENP8           YES, GO TO GENP8                        FT112960
        TRA     GENP4           NO, GO TO GENP4                         FT112970
        MACER                   PRLTBL OUT OF SORT                      FT112980
        REM                                                             FT112990
        REM     ... SAME ARG. BUT NEW DELTA                             FT113000
        REM                                                             FT113010
GENP4   ZET     ERASE3          ARE XR*S  IN USE FOR THIS ARG.          FT113020
        TRA     GENP5           YES, GO TO GENP5                        FT113030
        STL     ERASE3          NO, INDICATE THAT THEY ARE NOW          FT113040
        CAL     PAX1            STASH     PAX  ,1                       FT113050
        TSX     STASH,4         X                                       FT113060
        XEC     GENP1           GET THIS PRLTBL ENTRY AGAIN             FT113070
        ANA     DELBIT          ISOLATE THIS NEW DELTA                  FT113080
        ALS     3               PUT NEW DELTA IN ACC (3-17)             FT113090
        SLW     ERASE2          SAVE IN ERASE2                          FT113100
GENP6   ACL     TXI11           GENERATE    TXI PGMCTR+1,1,DELTA        FT113110
        ACL     PGMCTR          X                                       FT113120
        SSM                     INDICATE ADDRESS RELOCATION             FT113130
        TSX     STASH,4         STAST THIS TXI                          FT113140
GENP7   CAL     SXA1            NOW GENERATE    SXA  (TEXT),1          FT113150
        ACL     LTVPRL          X                                       FT113160
        ACL     ERASE4          X                                       FT113170
        SSM                     INDICATE ADDRESS RELOCATION             FT113180
        TSX     STASH,4         STASH THIS    SXA  (TEXT),1             FT113190
        TRA     GENP3-2                                                 FT113200
        REM                                                             FT113210
        REM     ... SAME ARG. BUT NEW DELTA WITH XR*S ALREADY IN USE    FT113220
        REM                                                             FT113230
GENP5   LDC     ERASE2,1        XR1 = -(OLD DELTA)                      FT113240
        XEC     GENP1           GET PRLTBL ENTRY AGAIN                  FT113250
        ANA     DELBIT          ISOLATE NEW DELTA                       FT113260
        ALS     3               SHIFT NEW DELTA INTO ACC (3-17)         FT113270
        SLW     ERASE2          SAVE THIS IN ERASE2                     FT113280
        STD     *+1             X                                       FT113290
        TXI     *+1,1,..        GET (NEW DELTA - OLD DELTA)             FT113300
        PXD     ,1              PUT THIS INTO ACC                       FT113310
        TRA     GENP6           NOW GO STASH THIS TXI AND SXA           FT113320
        REM     ... TO HANDLE NEW ARGUMENT NUMBER                       FT113330
        REM                                                             FT113340
        REM                                                             FT113350
```

```
GENP8   SLW     ERASE1          SAVE NEW ARG. IN ERASE1                 FT113360
        XEC     GENP1           GET THIS PRLTBL ENTRY AGAIN             FT113370
        ARS     30              SHIFT NEW DELTA INTO ACC(30-35)         FT113380
        ACL     CAL4            ACC = CAL  (ARG NO.),4                  FT113390
        TSX     STASH,4         STASH THIS CAL                          FT113400
        XEC     GENP1           GET PRLTBL ENTRY AGAIN                  FT113410
        ANA     ARGDEL          SAVE NEW ARG. AND DELTA IN ERASE        FT113420
        SLW     ERASE           X                                       FT113430
        ANA     DELBIT          SAVE NEW DELTA IN ERASE2                FT113440
        TNZ     GENP4+2         IF NON-ZERO DELTA, GO TO GENP4+2        FT113450
        SLW     ERASE2          X                                       FT113460
        STZ     ERASE3          INDICATE NO XR*S YET THIS ARG.          FT113470
        TRA     GENP2+2         OTHERWISE GO GENERATE STA               FT113480
        REM                                                             FT113490
        REM     ... TO HANDLE GENERATION OF SECTS. 1 AND 3 OF PROLOGUE  FT113500
        REM                                                             FT113510
GENP9   CAL     AXT1          PUGENERATE AXT*S IN APPENDIX AS REQD.     FT113520
        RFT     100000          X                                       FT113530
        TSX     STASH,4         X                                       FT113540
        CAL     AXT2            X                                       FT113550
        RFT     200000          X                                       FT113560
        TSX     STASH,4         X                                       FT113570
        CAL     AXT4            X                                       FT113580
        RFT     400000          X                                       FT113590
        TSX     STASH,4         X                                       FT113600
        CAL     NOARGS          GENERATE FINAL SUBR. EXIT (TRA  (N),4)  FT113610
        ACL     TRA4            X                                       FT113620
        TSX     STASH,4         X                                       FT113630
        CAL     ENTRY           GET ADDRESS OF FIRST AXT                FT113640
        SUB     NOXRS           X                                       FT113650
        SUB     ONE             X                                       FT113660
        SLW     ERASE                                                   FT113670
        RNT     100000          GENERATE SXA IFF XR1 TO BE SAVED        FT113680
        TRA     GENPB           OTHERWISE GO TO GENPB                   FT113690
        ACL     SXA1            X                                       FT113700
        SSM                     INDICATE ADDRESS RELOCATION             FT113710
        TSX     STASH,4         GO STASH THIS INSTRUCTION               FT113720
        CAL     ERASE           SET UP FOR NEXT XR                      FT113730
        ACL     ONE             X                                       FT113740
        STA     ERASE           X                                       FT113750
GENPB   RNT     200000          GENERATE SXA  --,2 IF NECESSARY          FT113760
        TRA     GENPC           OTHERWISE GO TO GENPC                   FT113770
        ACL     SXA2            X                                       FT113780
        SSM                     X                                       FT113790
        TSX     STASH,4         X                                       FT113800
        CAL     ERASE           SET UP FOR NEXT XR                      FT113810
        ACL     ONE             X                                       FT113820
        STA     ERASE           X                                       FT113830
GENPC   RNT     400000          GENERATE SXA  --,4 IF NECESSARY          FT113840
        TRA     GENPA           EXIT THIS SECT. GENPRL IF NOT NEC.      FT113850
        ACL     SXA4            X                                       FT113860
        SSM                     X                                       FT113870
        TSX     STASH,4         X                                       FT113880
        TRA     GENPA           EXIT THIS SECTION                       FT113890
        REM                                                             FT113900
        REM     ... TO HANDLE GENERATION OF TRA AROUND A. S. F.         FT113910
        REM                                                             FT113920
GENPD   CAL     LARSTS            IF A. S. F. ARE INVOLVED IN THIS      FT113930
        TZE     PASS2              SUBPROGRAM THEN AT END OF PROLOGUE   FT113940
        ACL     LTVECT             GENERATE TRANSFER AROUND THEM TO     FT113950
```

```
ACL     LPROLG                ENTRY POINT OF TEXT PROPER      FT113960
ACL     TRA                   X                               FT113970
SSM                           X                               FT113980
TSX     STASH,4               X                               FT113990
TRA     PASS2                                                 FT114000
```

```
         TTL   P O S T  P A S S  1  S U B R O U T I N E S  -  S T D U M PFT114010
*        THIS SUBROUTINE IS ENTERED ON ENTRY TO POST PASS 1 IF L S T S Y MFT114020
*        IS NON-ZERO.  THIS CELL WILL BE NON-ZERO IF THE STATEMENT       FT114030
         REM                                                             FT114040
*                  LIST SYMBOL TABLE                                     FT114050
         REM                                                             FT114060
*        APPEARED ANYWHERE IN THE SOURCE PROGRAM.                        FT114070
         SPACE  3                                                        FT114080
*        THIS ROUTINE OUTPUTS ON M L S T A P  EACH SYMBOL, ITS RELATIVE  FT114090
*        LOCATION IN S Y M T A B AND ITS CORRESPONDING E Q U I V ENTRY.  FT114100
         REM                                                             FT114110
*        THIS ROUTINE ALSO DUMPS THE USED PORTION OF P O O L (4 ENTRIES/ FT114120
*        LINE) IN OCTAL ONTO M L S T A P.                                FT114130
         SPACE  3                                                        FT114140
STDUMP   SAVE   STDUMX                                                   FT114150
IOS23    EQU    *                                                       FT114160
         WRITE  *MLSTAP,DEC,STDMP1,3                                     FT114170
         STL    IOEX                                                     FT114180
         BRA    WRITE,,*+3                                               FT114190
         PTW    MLSTAP,,DEC                                              FT114200
         MZE    STDMP1,,3                                                FT114210
END23    REM                                                             FT114220
IOS24    EQU    *                                                       FT114230
         WRITE  *MLSTAP,DEC,STDM10,10                                    FT114240
         STL    IOEX                                                     FT114250
         BRA    WRITE,,*+3                                               FT114260
         PTW    MLSTAP,,DEC                                              FT114270
         MZE    STDM10,,10                                               FT114280
END24    REM                                                             FT114290
IOS25    EQU    *                                                       FT114300
         WRITE  *MLSTAP,DEC,STDM11,10                                    FT114310
         STL    IOEX                                                     FT114320
         BRA    WRITE,,*+3                                               FT114330
         PTW    MLSTAP,,DEC                                              FT114340
         MZE    STDM11,,10                                               FT114350
END25    REM                                                             FT114360
IOS26    EQU    *                                                       FT114370
         WRITE  *MLSTAP,DEC,STDM12,10                                    FT114380
         STL    IOEX                                                     FT114390
         BRA    WRITE,,*+3                                               FT114400
         PTW    MLSTAP,,DEC                                              FT114410
         MZE    STDM12,,10                                               FT114420
END26    REM                                                             FT114430
IOS27    EQU    *                                                       FT114440
         WRITE  *MLSTAP,DEC,STDM13,10                                    FT114450
         STL    IOEX                                                     FT114460
         BRA    WRITE,,*+3                                               FT114470
         PTW    MLSTAP,,DEC                                              FT114480
         MZE    STDM13,,10                                               FT114490
END27    REM                                                             FT114500
IOS28    EQU    *                                                       FT114510
         WRITE  *MLSTAP,DEC,STDMP4,1                                     FT114520
         STL    IOEX                                                     FT114530
         BRA    WRITE,,*+3                                               FT114540
         PTW    MLSTAP,,DEC                                              FT114550
         MZE    STDMP4,,1                                                FT114560
END28    REM                                                             FT114570
         AXT    4096,1                                                   FT114580
         AXT    0,2                                                      FT114590
         ZET    SYMTAB,2                                                 FT114600
```

```
         TRA     STDMP3                                                 FT114610
STDMPA   TXI     *+1,2,1                                                FT114620
         TIX     *-3,1,1                                                FT114630
         TRA     PLDMPX                                                 FT114640
STDUMX   RESTOR                                                         FT114650
         TRA     1,4                                                    FT114660
STDMP1   BCI     3,1SYMBOL TABLE DUMP                                   FT114670
STDMP2   BCI     1,0                                                    FT114680
STDMP3   PXA     ,2                                                     FT114690
         TSX     OCTBCD,4                                               FT114700
         SLW     STDMP4+1                                               FT114710
         CAL     SYMTAB,2                                               FT114720
         SLW     STDMP4+2                                               FT114730
         CAL     EQUIV,2                                                FT114740
         LGR     33                                                     FT114750
         ALS     12                                                     FT114760
         ORA     =H    0                                                FT114770
         SLW     STDMP5                                                 FT114780
         AXT     4,4                                                    FT114790
STDMP7   ZAC                                                            FT114800
         LGL     1                                                      FT114810
         ALS     11                                                     FT114820
         LGL     1                                                      FT114830
         ALS     11                                                     FT114840
         LGL     1                                                      FT114850
         ORA     =H 0 0 0                                               FT114860
         SLW     STDMP6+4,4                                             FT114870
         TIX     STDMP7,4,1                                             FT114880
         ZAC                                                            FT114890
         LGL     6                                                      FT114900
         STQ     STDMP9                                                 FT114910
         TSX     OCTBCD,4                                               FT114920
         ORA     =H    000                                              FT114930
         SLW     STDMP8                                                 FT114940
         CAL     STDMP9                                                 FT114950
         ARS     21                                                     FT114960
         TSX     OCTBCD,4                                               FT114970
         SLW     STDMP8+1                                               FT114980
IOS29    EQU     *                                                      FT114990
         WRITE   *MLSTAP,DEC,STDMP4,10                                  FT115000
         STL     IOEX                                                   FT115010
         BRA     WRITE,,*+3                                             FT115020
         PTW     MLSTAP,,DEC                                            FT115030
         MZE     STDMP4,,10                                             FT115040
END29    REM                                                            FT115050
         TRA     STDMPA                                                 FT115060
STDMP4   BCI     1,                                                     FT115070
         PZE                                                            FT115080
         PZE                                                            FT115090
STDMP5   PZE                                                            FT115100
STDMP6   BSS     4                                                      FT115110
STDMP8   BSS     2                                                      FT115120
STDMP9   BSS     1                                                      FT115130
STDM10   BCI     ,                    M   D V L P A E C A G A E I       FT115140
STDM11   BCI     ,                    O   O A H A S Q O R O R X N       FT115150
STDM12   BCI     ,                    D   U R S T S U M G T R T T  ARG  FT115160
STDM13   BCI     ,        LOCN SYMBOL E   B B X H N V N T O Y F F   NO  ADDR FT115170
         REM                                                            FT115180
         REM                                                            FT115190
*        ROUTINE TO DUMP POOL REGION                                    FT115200
```

```
*                                                                  FT115210
IOS30   EQU     *                                                  FT115220
PLDMPX  WRITE   *MLSTAP,DEC,STDMP2,1                               FT115230
PLDMPX  STL     IOEX                                               FT115240
        BRA     WRITE,,*+3                                         FT115250
        PTW     MLSTAP,,DEC                                        FT115260
        MZE     STDMP2,,1                                          FT115270
END30   REM                                                        FT115280
IOS31   EQU     *                                                  FT115290
        WRITE   *MLSTAP,DEC,PLDMP1,3                               FT115300
        STL     IOEX                                               FT115310
        BRA     WRITE,,*+3                                         FT115320
        PTW     MLSTAP,,DEC                                        FT115330
        MZE     PLDMP1,,3                                          FT115340
END31   REM                                                        FT115350
IOS32   EQU     *                                                  FT115360
        WRITE   *MLSTAP,DEC,STDMP2,1                               FT115370
        STL     IOEX                                               FT115380
        BRA     WRITE,,*+3                                         FT115390
        PTW     MLSTAP,,DEC                                        FT115400
        MZE     STDMP2,,1                                          FT115410
END32   REM                                                        FT115420
        REM                                                        FT115430
        LXA     NXTLOC,1                                           FT115440
        SXD     PLDMP2+1,1                                         FT115450
        AXT     0,1                                                FT115460
        REM                                                        FT115470
PLDMP6  AXT     12,4                                               FT115480
        CAL     =H                                                 FT115490
        SLW     PLDMP3,4                                           FT115500
        TIX     *-1,4,1                                            FT115510
        REM                                                        FT115520
        PXA     ,1                                                 FT115530
        TSX     OCTBCD,4                                           FT115540
        SLW     PLDMP4+1                                           FT115550
        AXT     12,2                                               FT115560
        REM                                                        FT115570
PLDMP2  TXI     *+1,1,1                                            FT115580
        TXH     PLDMP5,1,**                                        FT115590
        LDQ     POOL+1,1                                           FT115600
        TSX     CINSTR,4                                           FT115610
        SLW     PLDMP3+1,2                                         FT115620
        STQ     PLDMP3+2,2                                         FT115630
        TIX     PLDMP2,2,3                                         FT115640
        REM                                                        FT115650
IOS33   EQU     *                                                  FT115660
        WRITE   *MLSTAP,DEC,PLDMP4,14                              FT115670
        STL     IOEX                                               FT115680
        BRA     WRITE,,*+3                                         FT115690
        PTW     MLSTAP,,DEC                                        FT115700
        MZE     PLDMP4,,14                                         FT115710
END33   REM                                                        FT115720
        TRA     PLDMP6                                             FT115730
        REM                                                        FT115740
IOS34   EQU     *                                                  FT115750
PLDMP5  WRITE   *MLSTAP,DEC,PLDMP4,14                              FT115760
PLDMP5  STL     IOEX                                               FT115770
        BRA     WRITE,,*+3                                         FT115780
        PTW     MLSTAP,,DEC                                        FT115790
        MZE     PLDMP4,,14                                         FT115800
```

```
END34   REM                                                    FT115810
        TRA     STDUMX                                         FT115820
        REM                                                    FT115830
PLDMP1  BCI     3,ODUMP OF POOL AREA                           FT115840
PLDMP4  BCI     1,                                             FT115850
PLDMP3  BES     13                                             FT115860
```

```
      TTL F M S I I  E D I T O R  C O N T R O L  C A R D  N O.  2          FT115870
*                                                                         FT115880
      SPACE   6                                                           FT115890
*     THE FOLLOWING 'INSTRUCTIONS' CAUSE PUNCHING OF THE EDITOR           FT115900
      LBL     2FSTR                                                       FT115910
*     PROGRAM CARD FOR THE SECOND PART OF FASTRAN (FASTRAN RESIDES ON     FT115920
*     THE SYSTEM TAPE AS TWO RECORDS --- NOS. 30 AND 31).                 FT115930
      SPACE   3                                                           FT115940
BEGRC2 EQU    *                                                           FT115950
      ORG     0                                                           FT115960
      9LP     5                                                           FT115970
      IORT    BEGRC2,,COLUMN-BEGRC2+1                                     FT115980
      TXI     BEGINZ,,31          ENTRY PT.,,PORGRAM NO.                  FT115990
      ABS                                                                 FT116000
      ORG     BEGRC2                                                      FT116010
```

```
      TTL   P A S S  2  -  D E S C R I P T I O N                    FT116030
*                                                                   FT116040
      REM                                                           FT116050
      SPACE  9                                                      FT116060
*     *  *  *  *  *  *  *  *  *  *  *  *                            FT116070
      REM                                                           FT116080
*           F A S T R A N  C O M P I L E R                          FT116090
      REM                                                           FT116100
*     R E S E A R C H  C O M P U T I N G  C E N T E R               FT116110
      REM                                                           FT116120
*           I N D I A N A  U N I V E R S I T Y                      FT116130
      REM                                                           FT116140
*              B L O O M I N G T O N ,  I N D I A N A              FT116150
      SPACE  3                                                      FT116160
*     *  *  *  *  *  *  *  *  *  *  *  *                            FT116170
      REM                                                           FT116180
*                        P A S S  2                                 FT116190
      REM                                                           FT116200
*     *  *  *  *  *  *  *  *  *  *  *  *                            FT116210
```

```
      EJECT                                                                 FT116220
*     ON ENTRY TO PASS 2 COMPILATION IS TO THE POINT WHERE THE PROGRAM      FT116230
*     CARD IS OUT AND THE TRANSFER VECTOR AND PROLOGUE HAVE BEEN            FT116240
*     'STASHED' INTO CARD IMAGES.                                           FT116250
      REM                                                                   FT116260
*     THE FOLLOWING IS AN OVERALL PICTURE OF PASS 2. (DETAILED COMMENTS     FT116270
*     PRECEDE EACH SECTION).                                                FT116280
      REM                                                                   FT116290
*     PASS 2 PERFORMS THE FOLLOWING FUNCTIONS IN THE INDICATED ORDER,       FT116300
      SPACE   2                                                             FT116310
*        CITPRO-  GENERATE AND STASH OBJECT TIME TEXT CODE.  THIS IS        FT116320
*                 DONE BY PROCESSING C I T AND C I T F L G.                 FT116330
      REM                                                                   FT116340
*        SCONST-  STASH OBJECT TIME CONSTANTS.                              FT116350
      REM                                                                   FT116360
*        SSTRIN-  STASH OBJECT TIME STRINGS (FORMATS).                      FT116370
      REM                                                                   FT116380
*        LSTASH-  CLOSE PUNCH FILE BY FORCING OUT LAST CARD.                FT116390
      SPACE   5                                                             FT116400
*     THIS IS THE END OF PASS 2.  AT THIS POINT POST PASS 2 IS ENTERED.      FT116410
```

```
      TTL    P A S S  2  -  C I T P R O                                FT116420
*     THE FOLLOWING ROUTINE IS RESPONSIBLE FOR PROCESSING C I T AND    FT116430
*     C I T F L G.  THAT IS, THE OBJECT TIME TEXT CODE IS GENERATED    FT116440
*     USING THE C I T AND C I T F L G FILES GENERATED WITH THE PASS 1  FT116450
*     SUBROUTINE C I T B L D.                                          FT116460
                                                                       FT116470
      SPACE  2                                                         FT116480
*     NOW C I T P R O IS BROKEN INTO THREE SECTIONS,                   FT116490
      REM                                                              FT116500
*        GETFLG-  THIS SECTION OBTAINS THE NEXT C I T WORD AND ITS     FT116510
*                 ASSOCIATED FLAG.  IF NECESSARY, G E T F L G READS    FT116520
*                 SYSUT1 TO PULL IN THE NEXT C I T AND C I T F L G     FT116530
*                 BLOCKS.  WHEN G E T F L G RECOGNIZES THE 'END-OF-    FT116540
*                 C I T' FLAG (= 77 OCTAL), CONTROL IS SENT, BY        FT116550
*                 G E T F L G, TO THE NEXT ROUTINE IN PASS 2 (THAT IS, FT116560
*                 S C O N S T).                                        FT116570
      REM                                                              FT116580
*        FLAG PROCESSORS-  THE SEVERAL FLAG PROCESSORS SIMPLY PERFORM  FT116590
*                          THE REQUIRED RELOCATION ON A C I T ENTRY    FT116600
*                          ACCORDING TO THE VALUE OF ITS ASSOCIATED    FT116610
*                          FLAG.                                       FT116620
      REM                                                              FT116630
*        2STASH-  THIS ROUTINE IS BUILT INTO THE PROCESSOR FOR FLAG = NFT116640
*                 AND IS RESPONSIBLE FOR STASHING THE FINALIZED OBJECT FT116650
*                 TIME INSTRUCTION INTO A BINARY CARD IMAGE AND        FT116660
*                 INSERTING THE PROPER RELOCATION SPECIFICATION INTO   FT116670
*                 THIS CARD IMAGE.  AFTER STASHING, 2 S T A S H SENDS  FT116680
*                 CONTROL TO G E T F L G TO PROCESS THE NEXT C I T     FT116690
*                 ENTRY.
```

```
        EJECT                                                                   FT116700
*       THIS IS THE FIRST OF THE THREE C I T P R O SECTIONS.  THE NEXT          FT116710
*       C I T WORD AND ITS ASSOCIATED FLAG ARE OBTAINED AND PASSED ON TO        FT116720
*       THE NEXT SECTION OF C I T P R O.                                        FT116730
        SPACE   3                                                               FT116740
        NZT     CITSW1          WAS CIT2 PUT ON TAPE                            FT116750
        TRA     GETFG1          NO, BYPASS READ OF SYSUT1                       FT116760
IOS35   EQU     *                                                               FT116770
GETFG5  READ    SYSUT1,BINARY,CTFLG2,-1   READ CIT2 FLAG BLOCK                  FT116780
GETFG5  STL     IOEX                                                            FT116790
        BRA     READ,,*+3                                                       FT116800
        PZE     SYSUT1,,BINARY                                                  FT116810
        MZE     CTFLG2,,-1                                                      FT116820
END35   REM                                                                     FT116830
IOS36   EQU     *                                                               FT116840
        READ    SYSUT1,BINARY,CIT2,-1     READ CIT2 BLOCK                       FT116850
        STL     IOEX                                                            FT116860
        BRA     READ,,*+3                                                       FT116870
        PZE     SYSUT1,,BINARY                                                  FT116880
        MZE     CIT2,,-1                                                        FT116890
END36   REM                                                                     FT116900
GETFG1  AXC     1,1             CIT WORD POINTER                                FT116910
        REM                                                                     FT116920
 PASS2  EQU     GETFG1          DEFINE SYMBOL 'PASS2'                           FT116930
GETFLG  EQU     GETFG1          DEFINE SYMBOL 'GETFLG'                          FT116940
        REM                                                                     FT116950
GETFG2  AXC     1,2             FLAG WORD POINTER                               FT116960
        AXT     6,4             FLAG WORD BYTE POINTER                          FT116970
GETLCA  LDQ     ..,2            SET MQ = CURRENT FLAG WORD                      FT116980
        ZAC                     PREPARE ACC TO ACCEPT NEW FLAG                  FT116990
        LGL     6               SHIFT NEW FLAG INTO ACC                         FT117000
GETLCB  STQ     ..,2            SAVE REMAINDER OF FLAG WORD                     FT117010
        STA     GETFGA          SAVE THIS FLAG TEMPORARILY                      FT117020
        LAS     =076            IS THIS FLAG = 76 OR 77 (OCTAL)                 FT117030
        TRA     SCONST          FLAG = 77 ... GO TO SCONST                      FT117040
        TRA     GETFG4          FLAG = 76 ... GO READ NEXT CIT BLOCK            FT117050
GETLCC  CAL     ..,1            SET ACC TO THIS CIT ENTRY                       FT117060
        LXA     GETFG1,1        SET XR1 TO CIT WORD POINTER                     FT117070
        TIX     GETFG3,4,1      IF END OF FLAG WORD SET-UP FOR NEXT             FT117080
        AXT     6,4             RESET FLAG WORD BYTE POINTER                    FT117090
        LXA     GETFG2,2        BUMP FLAG WORD POINTER                          FT117100
        TXI     *+1,2,1         X                                               FT117110
        SXA     GETFG2,2        X                                               FT117120
GETFG3  TXI     *+1,1,1         BUMP CIT WORD POINTER                           FT117130
        SXA     GETFG1,1        X                                               FT117140
        SXA     GETFG2+1,4      SAVE FLAG WORD BYTE POINTER                     FT117150
        LXA     GETFGA,1        SET XR1 TO FLAG                                 FT117160
```

```
         EJECT                                                          FT117170
*        THIS IS THE SECOND OF THE THREE C I T P R O SECTIONS.  THE     FT117180
*        VARIOUS PROCESSORS FOR THE C I T FLAGS RESIDE HERE.  ON ENTRY  FT117190
*        TO THIS SECTION THE CURRENT C I T FLAG RESIDES IN XR1.         FT117200
         SPACE  3                                                       FT117210
         TXH    CITER,1,MAXFLG      ERROR IF FLAG TOO HIGH             FT117220
         TRA*   FLGTBL,1            TRANSFER TO APPROPRIATE FLAG PROCESSORFT117230
FLGTBL   EQU    *                                                      FT117240
         FLAGH                                                          FT117250
         FLAGG                                                          FT117260
         FLAGD                                                          FT117270
         FLAGQ                                                          FT117280
         FLAGE                                                          FT117290
         FLAGW                                                          FT117300
         FLAGX                                                          FT117310
         FLAGS                                                          FT117320
         FLAGC                                                          FT117330
         FLAGF                                                          FT117340
         FLAGT                                                          FT117350
         FLAGV                                                          FT117360
         FLAGR                                                          FT117370
         FLAGP                                                          FT117380
FLGTBL   FLAGN                                                          FT117390
MAXFLG   EQU    *-FLGTBL-1          FOR USE IN TEST ABOVE              FT117400
         SPACE  2                                                       FT117410
CITER    STR    MCR,,SCONST         ILLEGAL CIT FLAG ENCOUNTERED       FT117420
         REM                                                           FT117430
         REM                                                           FT117440
         REM                                                           FT117450
         REM    THE FOLLOWING ARE THE FLAG PROCESSORS.  CONTROL IS SENT FT117460
         REM     TO THE PROPER PROCESSOR VIA ROUTINE CITPRO.           FT117470
         SPACE  2                                                      FT117480
         REM                                                           FT117490
         REM    FLAG = P                                               FT117500
         REM                                                           FT117510
FLAGP    ACL    LTVPRL              TEXT RELOCATION                    FT117520
         SSM                        SPECIFY ADDRESS RELOCATION         FT117530
         TRA    FLAGN               GO STASH THIS INSTRUCTION          FT117540
         REM                                                           FT117550
         REM    FLAG = R                                               FT117560
         REM                                                           FT117570
FLAGR    PAX    ,1                  TEXT RELOCATION ON EQUIV ENTRY     FT117580
         SLW    ERASE               SAVE CIT WORD                      FT117590
         CAL    EQUIV,1             GET EQUIV ENTRY                    FT117600
         STA    ERASE               BUILD WORD FOR TEXT RELOCATION     FT117610
         CAL    ERASE               SET ACC FOR TEXT RELOCATION        FT117620
         TRA    FLAGP               GO DO TEXT RELOCATION              FT117630
         REM                                                           FT117640
         REM    FLAG = V                                               FT117650
         REM                                                           FT117660
FLAGV    PAX    ,1                  COMMON OR VARIABLE RELOCATION      FT117670
         SLW    ERASE               SAVE CIT WORD                      FT117680
         LDI    EQUIV,1             GET EQUIV ENTRY                    FT117690
         PIA                        SET EQUIV ENTRY INTO ACC           FT117700
         STL    OVERFG              FLAG FOR POSSIBLE ALTERNANT RELOCATIONFT117710
         LNT    BCOMN               IS COMMON FLAG ON                  FT117720
         TRA    FLGV1                                                  FT117730
         ANA    =077777                                                FT117740
         SSM                        YES, DO COMMON RELOCATION          FT117750
         ADD    TOPCOM              X                                  FT117760
```

```
        TRA     FLAGT+4                                          *FT117770
FLGV1   ACL     LTHRUW          DO VARIABLE RELOCATION            FT117780
        TRA     FLAGT+4                                          FT117790
        REM                                                      FT117800
        REM     FLAG = T                                         FT117810
        REM                                                      FT117820
FLAGT   PAX     ,1              EXTERNAL SYMBOL RELOCATION       FT117830
        SLW     ERASE           SAVE WORD TO BE STASHED          FT117840
        CAL     EQUIV,1         GET EQUIV ENTRY                  FT117850
        SUB     ONE                                              FT117860
        STA     ERASE           BUILD WORD TO BE RELOCATED       FT117870
        CAL     ERASE           SET ACC                          FT117880
        TRA     FLAGP+1         GO STASH THIS WORD               FT117890
        REM                                                      FT117900
        REM     FLAG = F                                         FT117910
        REM                                                      FT117920
FLAGF   PAX     ,1              STRING RELOCATION                FT117930
        SLW     ERASE           SAVE ARGUMENT                    FT117940
        CAL     EQUIV,1         GET EQUIV ENTRY                  FT117950
        ACL     LTPTSC          BUMP BY LENGTH TO STRINGS        FT117960
        TRA     FLAGT+4       GO STASH THIS WORD                 FT117970
        REM                                                      FT117980
        REM     FLAG = C                                         FT117990
        REM                                                      FT118000
FLAGC   ACL     LTVPTS          CONSTANT RELOCATION              FT118010
        TRA     FLAGP+1         GO STASH                         FT118020
        REM                                                      FT118030
        REM     FLAG = S                                         FT118040
        REM                                                      FT118050
FLAGS   PAC     ,1              LIBRARY SUBROUTINE ENTRY RELOCATION FT118060
        SLW     ERASE           X                                FT118070
        CAL     0,1             X                                FT118080
        SUB     ONE             X                                FT118090
        STA     ERASE           X                                FT118100
        CAL     ERASE           X                                FT118110
        TRA     FLAGP+1         X                                FT118120
        REM                                                      FT118130
        REM     FLAG = X                                         FT118140
        REM                                                      FT118150
FLAGX   CAL     TRA             SUBPROGRAM EXIT                  FT118160
        ACL     LTVECT          GENERATE TRANSFER TO RESTORE SECTION FT118170
        TRA     FLAGP+1          OF PROLOGUE AND GO STASH        FT118180
        REM                                                      FT118190
        REM     FLAG = W                                         FT118200
        REM                                                      FT118210
FLAGW   SLW     ERASE           WORKING STORAGE RELOCATION       FT118220
        PAC     ,4                                               FT118230
        TXI     *+1,4,..        THIS DECREMENT SET BY SYPASS     FT118240
        SXA     ERASE,4                                          FT118250
        CAL     ERASE                                            FT118260
        ACL     LTHRUS                                           FT118270
        TRA     FLAGP+1                                          FT118280
        REM                                                      FT118290
        REM     FLAG = E                                         FT118300
        REM                                                      FT118310
FLAGE   PAX     ,1              NON-LIBRARY EXTERNAL REFERENCE   FT118320
        SLW     ERASE           X                                FT118330
        CAL     EQUIV,1         X                                FT118340
        TRA     FLAGS+3         X                                FT118350
        REM                                                      FT118360
```

```
        REM     FLAG = Q                                                FT118370
        REM                                                             FT118380
FLAGQ   PAX     ,4              SAVE CONSTANT ADD-IN                    FT118390
        TZE     GETFLG                                                  FT118400
        SXD     FLGN1,4         X                                       FT118410
        STL     ADDIN           FLAG THAT ADD-IN IS PRESENT             FT118420
        TRA     GETFLG                                                  FT118430
        REM                                                             FT118440
        REM     FLAG = D                                                FT118450
        REM                                                             FT118460
FLAGD   AXT     1,1             ADD-IN = 1 AND VARIABLE RELOCATION      FT118470
        SXD     FLGN1,1         X                                       FT118480
        STL     ADDIN           X                                       FT118490
        TRA     FLAGV           X                                       FT118500
        REM                                                             FT118510
        REM     FLAG = G                                                FT118520
        REM                                                             FT118530
FLAGG   AXT     -1,1          ADD-IN = -1 AND VARIABLE RELOCATION       FT118540
        TRA     FLAGD+1                                                 FT118550
        REM                                                             FT118560
        REM     FLAG = H                                                FT118570
        REM                                                             FT118580
FLAGH   CAL     CAL             OUTPUT THE SEQUENCE,                    FT118590
        SSM                                                             FT118600
        TSX     STASH,4             CAL   (FPT)                         FT118610
        CAL     SLW8                SLW   8                             FT118620
        TSX     STASH,4             STZ   77462                         FT118630
        CAL     STZOV                                                   FT118640
        SSM                                                             FT118650
        TSX     STASH,4                                                 FT118660
        TRA     GETFLG                                                  FT118670
        REM                                                             FT118680
        REM     FLAG = N                                                FT118690
        REM                                                             FT118700
FLAGN   ZET     ADDIN           IS AN ADDIN IN EFFECT                   FT118710
        TRA     FLGN1-2         YES, GO GET IT                          FT118720
        STZ     OVERFG          ZERO OUT THE SPECIAL FLAG              *FT118730
```

```
        EJECT                                                                   FT118740
*       THIS IS THE THIRD OF THE THREE C I T P R O SECTIONS.  IT IS HERE        FT118750
*       THAT THE FINALIZED BINARY WORDS ARE STASHED INTO CARD IMAGES.           FT118760
        SPACE   2                                                               FT118770
*       ON ENTRY TO THIS SECTION THE BINARY WORD TO BE STASHED RESIDES          FT118780
*       IN THE ACC (P, 1-35) AND THE SIGN BIT OF THE ACC IS ON IFF              FT118790
*       ADDRESS RELOCATION IS TO BE PERFORMED ON THIS WORD AT BSS TIME.         FT118800
        SPACE   3                                                               FT118810
2STASH  AXT     20,2            XR2 IS IMAGE WORD POINTER                       FT118820
 STSH1  AXT     0,4             XR4 IS RELOCATION BIT COUNTER                   FT118830
        SLW     9LEFT+24,2      STORE THIS WORD IN CARD IMAGE                   FT118840
        ACL     SUM             ACCUMULATE CHECK-SUM                            FT118850
        SLW     SUM             X                                               FT118860
        TXI     *+1,4,2         BUMP RELOCATION BIT COUNTER BY 1                FT118870
        TPL     STSH2           GO TO STSH2 IFF NO RELOCATION                   FT118880
        TXI     *+1,4,1         OTHERWISE BUMP RELOCATION BIT COUNTER           FT118890
        LDQ     ADREL           SET LOW MQ TO 010                               FT118900
        ZET     OVERFG          IF OTHER RELOCATION, GET THE BITS              *FT118910
        LDQ     =03                                                            *FT118920
        STZ     OVERFG          ZERO OUT THE FLAG                              *FT118930
        ZAC                     SET ACC TO ZERO                                 FT118940
        LGL     72,4            MOVE RELOCATION BITS INTO 8-RIGHT               FT118950
        ORS     8LEFT            AND/OR 8-LEFT                                  FT118960
        LGL     36              X                                               FT118970
        ORS     8RIGHT          X                                               FT118980
 STSH2  SXA     STSH1,4         SAVE RELOCATION BIT COUNTER                     FT118990
        CAL     PGMCTR          BUMP PGMCTR BY 1                                FT119000
        ACL     ONE             X                                               FT119010
        SLW     PGMCTR          X                                               FT119020
        TIX     EXSTSH-1,2,1    GO EXIT IF IMAGE NOT FULL                       FT119030
        TXI     STSH3,2,339     RE-COMPLEMENT WD CNT AND PUT IN 7-9             FT119040
 STSH3  SXD     9LEFT,2         SAVE INSTRUCTION WORD COUNT IN 9-LEFT           FT119050
        CAL     SUM             ACL 9-LEFT INTO CHECK-SUM                       FT119060
        ACL     9LEFT           X                                               FT119070
        ACL     9LEFT+2         ADD RELOCATION BITS TO CHECK-SUM                FT119080
        ACL     9LEFT+3         X                                               FT119090
        SLW     SUM             X                                               FT119100
        TSX     CDLABL,4        PREPARE 73-80 FOR LABELLING                     FT119110
IOS37   EQU     *                                                               FT119120
        WRITE   *SYSBIN,BINARY,9LEFT,24,COL73,3   WRITE BIN. CARD IMAGE         FT119130
        STL     IOEX                                                            FT119140
        BRA     WRITE,,*+4                                                      FT119150
        PTW     SYSBIN,,BINARY                                                  FT119160
        PZE     9LEFT,,24                                                       FT119170
        MZE     COL73,,3                                                        FT119180
END37   REM                                                                     FT119190
        CAL     9LEFT           GET 9-LEFT ADDR. FOR NEXT CARD IMAGE            FT119200
        ACL     =20             GET ORIGIN OF NEXT CARD                         FT119210
        STA     9LEFT           X                                               FT119220
        AXT     20,4            RESET WORD POINTER                              FT119230
        SXA     STSH1-1,4       X                                               FT119240
        AXT     0,4             RESET RELOCATION BIT COUNTER                    FT119250
        SXA     STSH1,4         X                                               FT119260
        STZ     SUM             RESET CHECK-SUM                                 FT119270
        STZ     8LEFT           RESET RELOCATION WORDS IN IMAGE                 FT119280
        STZ     8RIGHT          X                                               FT119290
 STSH4  TRA     *+2             GO BACK TO SECTION 1 OF C I T P R O             FT119300
        SXA     STSH1-1,2       SAVE WORD POINTER                               FT119310
EXSTSH  TRA     GETFLG          GO BACK TO SECTION 1 OF C I T P R O             FT119320
        SPACE   3                                                               FT119330
```

```
        EJECT                                                           FT119340
*       THIS IS THE TAIL-END OF THE PASS 2 PROCESSOR FOR FLAG = N       FT119350
        SPACE   3                                                       FT119360
        SLW     ERASE           SAVE WORD TO BE STASHED                 FT119370
        PAX     ,4              XR4 = UNBUMPED ADDRESS                  FT119380
FLGN1   TXI     **+1,4,**       DO THE ADD-IN                           FT119390
        SXA     ERASE,4         GENERATE MODIFIED WORD                  FT119400
        LRS     0               SAVE SIGN OF ACCUMULATOR                FT119410
        CAL     ERASE           RESTORE ACCUMULATOR                     FT119420
        LLS     0                AND ITS SIGN                           FT119430
        STZ     ADDIN                                                   FT119440
        NZT     OVERFG          SKIP IF NOT A VARIABLE ADDEND           FT119450
        TRA     FLAGN+2                                                 FT119460
FLGN2   TXH     **+4,4,**       IS VARIABLE REFERENCE ABOVE PROG. BRK.  FT119470
        LNT     BCOMN           NO.  IS THIS A COMMON VARIABLE.         FT119480
        TRA     FLAGN+2         NO.  RELOCATION BITS ARE 010            FT119490
        TRA     FLAGN+3         YES.  RELOCATION BITS ARE 011           FT119500
        LFT     BCOMN           ABOVE PRGBRK.  IS IT IN COMMON.         FT119510
        TRA     FLAGN+2         YES.  RELOCATION BITS ARE 010           FT119520
        TRA     FLAGN+3         NO.  RELOCATION BITS ARE 011            FT119530
```

```
        TTL     P A S S  2  -  S C O N S T                              FT119540
*       THE FOLLOWING ROUTINE STASHES THE CONSTANTS REQUIRED BY THE     FT119550
*       OBJECT PROGRAM INTO BINARY CARD IMAGES.  THE CONSTANTS IN       FT119560
*       C O N T A B ARE PUT OUT IN THE ORDER THAT THEY APPEAR IN THAT   FT119570
*       TABLE.  AFTER ALL OF THESE CONSTANTS HAVE BEEN STASHED CONTROL  FT119580
*       IS RELINQUISHED TO THE NEXT ROUTINE IN PASS 2 (S S T R I N).    FT119590
        SPACE   3                                                       FT119600
SCONST  NZT     NOCONT          ARE ANY NORMAL CONSTANTS IN USE         FT119610
        TRA     SSTRIN          NO, GO TO S S T R I N IMMEDIATELY       FT119620
        LXA     NOCONT,2                                                FT119630
        SXD     EXSCON-1,2                                              FT119640
        AXT     1,2                                                     FT119650
        CAL     CONTAB+1,2                                              FT119660
        TSX     STASH,4                                                 FT119670
        TXI     *+1,2,1                                                 FT119680
        TXL     *-3,2,**                                                FT119690
EXSCON  BSS     0               GO TO S S T R I N AT END OF S C O N S TFT119700
```

```
        TTL     P A S S  2  -  S S T R I N                              FT119710
*       THE FOLLOWING ROUTINE STASHES STRINGS (FORMATS) INTO BINARY CARD FT119720
*       IMAGES.  THESE STRINGS ACTUALLY RESIDE IN P O O L BUT WERE      FT119730
*       CHAINED TOGETHER IN S Y P A S S (IN POST PASS 1).  AFTER ALL THE FT119740
*       STRINGS ARE STASHED CONTROL IS GIVEN TO THE NEXT ROUTINE IN PASS FT119750
*       2 (L S T A S H).                                               FT119760
        SPACE   3                                                      FT119770
SSTRIN  NZT     LSTRIN          ARE THERE ANY STRINGS TO STASH         FT119780
        TRA     LSTASH          NO, GO TO L S T A S H IMMEDIATELY      FT119790
        LXA     STSW2,1         GET POOL POINTER OF FIRST STRING       FT119800
 SSTR1  CAL     POOL,1          GET POOL ENTRY FOR THIS STRING         FT119810
        SLW     ERASE           SAVE THIS ENTRY IN ERASE               FT119820
        PAX     ,2              XR2 = LENGTH OF THIS STRING            FT119830
        SXD     *+1,2                                                  FT119840
        TXI     *+1,1,..                                               FT119850
        CAL     POOL,1                                                 FT119860
        TSX     STASH,4          AND GO STASH IT                       FT119870
        TXI     *+1,1,-1        BUMP XR1 TO GET NEXT STRING WORD       FT119880
        TIX     *-3,2,1         GO BACK FOR NEXT WORD IF NOT AT END    FT119890
        LXD     ERASE,1         GET SAVED POOL WORD FOR LAST STRING    FT119900
        TXH     SSTR1,1,0        ARE THERE MORE STRINGS. YES, GO.      FT119910
 EXSTR  BSS     0               GNO, GO TO LSTASH                      FT119920
```

```
         TTL     P A S S  2  -  L S T A S H                              FT119930
*        THE FOLLOWING ROUTINE IS USED TO FORCE OUT THE LAST (PERHAPS    FT119940
*        PARTIALLY FILLED) BINARY CARD IMAGE ONTO SYSBIN.  THIS IS DONE  FT119950
*        USING THE STASH ROUTINE THAT RESIDES IN SECTION 3 OF C I T P R O.FT119960
*        UPON COMPLETION OF L S T A S H, CONTROL IS PASSED ON TO POST    FT119970
*        PASS 2.                                                         FT119980
         SPACE   3                                                       FT119990
LSTASH   LXA     STSH1-1,2          PICK UP WORD POINTER                 FT120000
         TXH     PPASS2,2,19        IF IMAGE FULL GO TO POST PASS 2      FT120010
         AXT     EXLSTH,4           SET RETURN IN STASH                  FT120020
         SXA     EXSTSH,4           X                                    FT120030
         STZ     9LEFT+24,2         ZERO UNUSED COLUMNS OF LAST CARD     FT120040
         TIX     *-1,2,1            X                                    FT120050
         LAC     STSH1-1,2          GET WORD COUNT FOR LAST CARD         FT120060
         TXI     STSH3,2,340        SET-UP DECR. OF 9 LEFT               FT120070
         REM                                                             FT120080
EXLSTH   AXT     GETFLG,4           RESET STSH4 IN STASH ROUTINE         FT120090
         SXA     EXSTSH,4           X                                    FT120100
         TRA     PPASS2             GO TO POST PASS 2                    FT120110
```

```
        TTL     P A S S  2  -  M I S C E L L A N E O U S                 FT120120
*       THE FOLLOWING INSTRUCTIONS SIMPLY MAKE-UP THE TAIL END OF       FT120130
*       G E T F L G (SECTION 1 OF C I T P R O).  ENTRY IS MADE HERE WHEN FT120140
*       AN 'END-OF-C I T' FLAG (=76 OCTAL) IS ENCOUNTERED.  ROUTINE     FT120150
*       G E T F L G IS RE-INITIATED TO START WITH A NEW C I T AND       FT120160
*       C I T F L G BLOCK.                                              FT120170
        SPACE   3                                                       FT120180
GETFG4  AXT     1,1             RESET C I T WORD POINTER                FT120190
        SXA     GETFG1,1        X                                       FT120200
        SXA     GETFG2,1        ALSO RESET FLAG WORD POINTER            FT120210
        AXT     6,1             RESET FLAG WORD BYTE POINTER            FT120220
        SXA     GETFG2+1,1      X                                       FT120230
        ZET     GTSWCH          END OF CIT1 OR CIT2                     FT120240
        TRA     GETFG5-2        END OF CIT2                             FT120250
        AXT     CTFLG2-1,1      END OF CIT1, RE-INITIALIZE FOR CIT2     FT120260
        SXA     GETLCA,1                                                FT120270
        SXA     GETLCB,1                                                FT120280
        AXT     CIT2-1,1                                                FT120290
        SXA     GETLCC,1                                                FT120300
        STL     GTSWCH          TURN ON CIT2 SWITCH                     FT120310
        TRA     GETFG5-2        GO SEE IF CIT2 IS ON TAPE               FT120320
        SPACE   3                                                       FT120330
*       THIS IS THE TAIL-END OF THE T V E C T L ROUTINE                 FT120340
        SPACE   3                                                       FT120350
 TVEL2  PXA     ,1              DEFINE THIS TRANSFER VECTOR ELEMENT     FT120360
        STA     SLNTAB-2,4      X                                       FT120370
        TRA     TVEL1+2         GO LOOK FOR MORE LIBRARY REFERENCES     FT120380
```

```
      TTL   P O S T  P A S S  2  -  D E S C R I P T I O N          FT120390
*                                                                  FT120400
      REM                                                          FT120410
      SPACE  9                                                     FT120420
*        *   *   *   *   *   *   *   *   *   *   *                 FT120430
      REM                                                          FT120440
*           F A S T R A N  C O M P I L E R                         FT120450
      REM                                                          FT120460
*     R E S E A R C H  C O M P U T I N G  C E N T E R              FT120470
      REM                                                          FT120480
*           I N D I A N A  U N I V E R S I T Y                     FT120490
      REM                                                          FT120500
*              B L O O M I N G T O N ,  I N D I A N A             FT120510
      SPACE  3                                                     FT120520
*        *   *   *   *   *   *   *   *   *   *   *                 FT120530
      REM                                                          FT120540
*                    P O S T  P A S S  2                          FT120550
      REM                                                          FT120560
*        *   *   *   *   *   *   *   *   *   *   *                 FT120570
```

```
                                                                       FT120580
      EJECT                                                            FT120590
*     ON ENTRY TO POST PASS 2, THE OBJECT PROGRAM HAS BEEN COMPLETELY  FT120600
*     OUTPUT ONTO RELOCATABLE BINARY CARDS, THE RESULT OF AN ERROR-FREEFT120610
*     COMPILATION.  POST PASS 2 IS CONCERNED WITH THE FOLLOWING TWO    FT120620
*     ITEMS,                                                           FT120630
      SPACE  2                                                         FT120640
*        1. MAPPING OF THE OBJECT PROGRAM TO INDICATE THE EXACT CORE   FT120650
*           STORAGE LAYOUT OF THE OBJECT PROGRAM (RELATIVE TO ZERO).   FT120660
      REM                                                              FT120670
*        2. DETERMINING IF THE NEXT PROGRAM ON THE INPUT TAPE IS       FT120680
*           ANOTHER FASTRAN COMPILATION.  IF THE NEXT PROGRAM DOES     FT120690
*           INVOLVE FASTRAN THEN POST PASS 2 SENDS CONTROL TO THE      FT120700
*           INITIALIZATION SECTION OF THE COMPILER TO BEGIN THIS NEXT   FT120710
*           COMPILATION.  IF THE NEXT PROGRAM DOES NOT INVOLVE FASTRAN
*           THEN CONTROL IS RELINQUISHED TO THE SYSTEM MONITOR.        FT120720
```

```
        TTL     P O S T  P A S S  2  -  M A P                          FT120730
*       THE FOLLOWING ROUTINE IS USED TO OUTPUT ON THE SYSTEM OUTPUT TAPEFT120740
*       THE STORAGE MAP ASSOCIATED WITH THIS COMPILATION.  THIS MAP     FT120750
*       CONTAINS THE FOLLOWING ITEMS IN THE INDICATED ORDER (ITEMS WHICH FT120760
*       DO NOT APPLY TO A PARTICULAR PROGRAM ARE NOT MAPPED),          FT120770
        SPACE   2                                                       FT120780
*          STORAGE NOT USED BY PROGRAM                                  FT120790
*          INDEX REGISTERS USED BY THE OBJECT PROGRAM                   FT120800
*          ORIGIN AND LENGTH OF FOLLOWING,                              FT120810
*             TRANSFER VECTOR                                           FT120820
*             PROLOGUE                                                  FT120830
*             ARITHMETIC STATEMENTS                                     FT120840
*             TEXT                                                      FT120850
*             CONSTANTS                                                 FT120860
*             STRINGS (FORMATS)                                         FT120870
*             WORKING STORAGE                                           FT120880
*             LOW-CORE VARIABLES                                        FT120890
*             COMMON STORAGE                                            FT120900
*          TRANSFER VECTOR                                              FT120910
*          OCTAL LOCATIONS OF NUMBERED EXECUTABLE STATEMENTS            FT120920
*          OCTAL LOCATIONS OF STRINGS (FORMATS)                         FT120930
*          OCTAL LOCATIONS OF NON-DIMENSIONED NON-COMMON VARIABLES      FT120940
*          OCTAL LOCATIONS OF DIMENSIONED NON-COMMON VARIABLES          FT120950
*          OCTAL LOCATIONS OF NON-DIMENSIONED COMMON VARIABLES          FT120960
*          OCTAL LOCATIONS OF DIMENSIONED COMMON VARIABLES              FT120970
        SPACE   3                                                       FT120980
PPASS2  CAL     LABEL           GET LABEL OF PROGRAM BEING COMPILED     FT120990
        TZE     MAP1            TRANSFER IF MAIN PROGRAM                FT121000
        SLW     LIN1+9          STORE SUBPROGRAM NAME IN HEADING        FT121010
        CAL     BLANKS          X                                       FT121020
        SLW     LIN1+10         X                                       FT121030
MAP1    TSX     LIST,4          GO LIST MAP HEADING                     FT121040
        PTW     LIN1,,11        PAGE EJECT                              FT121050
        CAL     GARB1           RESTORE LIN1 TO 'MAIN PROGRAM'          FT121060
        SLW     LIN1+9          X                                       FT121070
        CAL     GARB1+1         X                                       FT121080
        SLW     LIN1+10         X                                       FT121090
        CAL     PGMBRK          GET PROGRAM BREAK AND COMMON BREAK      FT121100
        TSX     OCTBCD,4         FOR 'STORAGE NOT USED ...' MESSAGE     FT121110
        ARS     6               RIGHT-ADJUST PROGRAM BREAK              FT121120
        ORA     GARB2           LEADING BLANK                           FT121130
        SLW     LIN2+5          INSERT IN MESSAGE                       FT121140
        CAL     CMNBRK          GET COMMON BREAK                        FT121150
        TSX     OCTBCD,4        CONVERT TO OCTAL                        FT121160
        SLW     LIN2+7          INSERT IN MESSAGE                       FT121170
        TSX     LIST,4          PUT OUT OCTAL PROGRAM LIMITS            FT121180
        PON     LIN2,,10        DOUBLE SPACED                           FT121190
        CAL     PGMBRK          GET PROGRAM BREAK FOR DECIMAL CONVER.   FT121200
        TSX     BINBCD,4        DO DECIMAL CONVERSION                   FT121210
        SLW     LIN3+5          INSERT IN DECIMAL MESSAGE               FT121220
        CAL     CMNBRK          DO SAME FOR COMMON BREAK                FT121230
        TSX     BINBCD,4        X                                       FT121240
        ALS     6               LEFT-ADJUST                             FT121250
        ORA     GARB3           TRAILING BLANK                          FT121260
        SLW     LIN3+7          INSERT IN DECIMAL MESSAGE               FT121270
        TSX     LIST,4          PUT OUT DECIMAL MESSAGE                 FT121280
        PZE     LIN3,,10        SINGLE SPACED                           FT121290
```

```
EJECT                                                          FT121300
REM     ... TO PUT OUT INDEX REGISTER USAGE MESSAGE            FT121310
REM                                                            FT121320
CAL     XRFLAG          XRFLAG CONTAINS 'OR'-ED XR'S USED      FT121330
ANA     TG7BIT          SAVE TAG ONLY                          FT121340
ALS     3               MOVE TO DECREMENT                      FT121350
PDX     ,1              XR1 = XR'S USED                        FT121360
CAL     GARB4,1         GET APPROPRIATE WORD FOR MESSAGE       FT121370
SLW     LIN4+7          INSERT IN MESSAGE                      FT121380
TSX     LIST,4          PUT OUT THIS MESSAGE                   FT121390
PON     LIN4,,8         DOUBLE SPACED                          FT121400
```

```
      EJECT                                                          FT121410
      REM     ... TO PUT OUT TABLE OF LIMITS OF VARIOUS SECTIONS OF  FT121420
      REM           THE OBJECT PROGRAM                               FT121430
      REM                                                            FT121440
      TSX     LIST,4              PUT OUT 'ORIGIN       LENGTH' HEADING FT121450
      PON     LIN5,,13            DOUBLE SPACED                      FT121460
      TSX     LIST,4              PUT OUT 'OCTAL  --  DECIMAL' MESSAGE FT121470
      PZE     LIN6,,14            SINGLE SPACED                      FT121480
      AXT     8,1                 PREPARE FIRST 8 TABLE ENTRIES      FT121490
      STZ     MAPORG              X                                  FT121500
MAP2  CAL     LTVECT+8,1          GET LENGTH OF NEXT ENTRY           FT121510
      TZE     MAP3                IF ZERO GO GET NEXT ENTRY          FT121520
      TSX     BINBCD,4            CONVERT LENGTH TO DECIMAL          FT121530
      ALS     6                   RIGHT-ADJUST                       FT121540
      ORA     GARB3               TRAILING BLANK                     FT121550
      SLW     LIN7+13             STORE THIS IN MESSAGE              FT121560
      CAL     LTVECT+8,1          NOW CONVERT LENGTH TO OCTAL        FT121570
      TSX     OCTBCD,4            X                                  FT121580
      SLW     LIN7+11             INSERT IN MESSAGE                  FT121590
      CAL     MAPORG              CONVERT ORIGIN OF THIS SECTION TO  FT121600
      TSX     BINBCD,4             DECIMAL                           FT121610
      ALS     6                   LEFT-ADJUST                        FT121620
      ORA     GARB3               TRAILING BLANK                     FT121630
      SLW     LIN7+8              INSERT IN MESSAGE                  FT121640
      CAL     MAPORG              NOW CONVERT ORIGIN TO OCTAL        FT121650
      TSX     OCTBCD,4            X                                  FT121660
      SLW     LIN7+6              INSERT IN MESSAGE                  FT121670
      AXT     4,2                 MOVE SECTION NAME INTO PLACE       FT121680
      CAL*    GARB5,1             USED INDIRECT TABLE ADDRESSING     FT121690
      SLW     LIN7+4,2            MOVE INTO OUTPUT AREA              FT121700
      TIX     *-2,2,1             MOVE TOTAL OF FOUR WORDS           FT121710
      TSX     LIST,4              LIST THIS TABLE ENTRY              FT121720
      PON     LIN7,,14            DOUBLE SPACED                      FT121730
      CAL     MAPORG              UPDATE MAPORG                      FT121740
      ACL     LTVECT+8,1          X                                  FT121750
      SLW     MAPORG              X                                  FT121760
MAP3  TIX     MAP2,1,1            GO BACK FOR NEXT TABLE ENTRY       FT121770
      CAL     CMNBRK              LAST TABLE ENTRY IS 'COMMON STORAGE' FT121780
      LAS     TOPCOM              SKIP IFF NO COMMON                 FT121790
      TRA     *+2                 SOMETHING'S MESSED UP              FT121800
      TRA     MAP28               SKIP MAP OF COMMON                 FT121810
      ADD     =1                  X                                  FT121820
      SLW     ERASE               X                                  FT121830
      TSX     BINBCD,4            CONVERT ORIGIN TO DECIMAL          FT121840
      ALS     6                    LEFT-ADJUST                       FT121850
      ORA     GARB3               TRAILING BLANK                     FT121860
      SLW     LIN8+8              INSERT IN MESSAGE                  FT121870
      CAL     ERASE               NOW CONVERT ORG COMMON TO OCTAL    FT121880
      TSX     OCTBCD,4            X                                  FT121890
      SLW     LIN8+6              INSERT IN MESSAGE                  FT121900
      CAL     TOPCOM              CALCULATE LENGTH OF COMMON         FT121910
      SUB     CMNBRK              LENGTH = TOPCOM - CMNBRK           FT121920
      SLW     TEMP                SAVE THIS LENGTH FOR CONVER. TO OCTAL FT121930
      TSX     BINBCD,4            CONVERT LENGTH TO DECIMAL          FT121940
      ALS     6                   LEFT-ADJUST                        FT121950
      ORA     GARB3               TRAILING BLANK                     FT121960
      SLW     LIN8+13             INSERT IN MESSAGE                  FT121970
      CAL     TEMP                CONVERT COMMON LENGTH TO OCTAL     FT121980
      TSX     OCTBCD,4            X                                  FT121990
      SLW     LIN8+11             INSERT IN MESSAGE                  FT122000
```

```
TSX     LIST,4          LIST THIS (THE LAST) TABLE ENTRY        FT122010        PAGE 318
PON     LIN8,,14        DOUBLE SPACED                           FT122020
```

```
      EJECT                                                          FT122030
      REM     ... TO PUT OUT THE TRANSFER VECTOR IN THE MAP          FT122040
      REM                                                            FT122050
MAP28 NZT     LTVECT            IS THE TRANSFER VECTOR EMPTY         FT122060
      TRA     MAP7              YES, BYPASS THIS SECTION             FT122070
      AXT     16,3                                                   FT122080
      LDQ     BLANKS                                                 FT122090
      STQ     LIN10+17,2                                             FT122100
      TIX     *-1,2,1                                                FT122110
      TSX     LIST,4            PUT OUT BLANK LINE                   FT122120
      PON     BLANKS,,1         DOUBLE SPACED                        FT122130
      TSX     LIST,4            PUT OUT MAP HEADING FOR THIS SECTION FT122140
      PON     LIN9,,3           DOUBLE SPACED                        FT122150
      AXT     MAP4,4            SUBROUTINE 'STVECT' IS GOING TO BE   FT122160
      SXA     STV1+2,4           USED TO PICK-UP TRANSFER VECTOR     FT122170
      SXA     STV2+1,4           ELEMENTS FOR THE MAP                FT122180
      TSX     STVECT+2,4        AFTER SETTING UP 'STVECT' GO TO IT   FT122190
      TRA     MAP6              GO TO MAP6 AT END                    FT122200
 MAP4 SLW     LIN10+17,1        THIS SECTION GETS CONTROL WHEN STVECT FT122210
      TIX     MAP5+1,1,2         FINDS A T.V. ELEMENT                FT122220
      SXA     MAP5,4            WHEN LINE IS FULL, WRITE IT          FT122230
      TSX     LIST,4            X                                    FT122240
      PON     LIN10,,16         DOUBLE-SPACED (FIRST TIME ONLY)      FT122250
      CLA     *                 FORCE SUBSEQUENT LINES SINGLE SPACED FT122260
      STP     *-2               X                                    FT122270
      STP     MAP6+6            X                                    FT122280
      AXT     16,1              RESET FOR NEW LINE                   FT122290
 MAP5 AXT     **,4              X                                    FT122300
      TRA     1,4               RETURN TO STVECT                     FT122310
 MAP6 TXH     MAP30,1,15        EXIT IF LAST LINE EMPTY              FT122320
      TXI     *+1,1,-16         PREPARE TO OUTPUT LAST T.V. LINE     FT122330
      PXA     ,1                X                                    FT122340
      PAC     ,1                X                                    FT122350
      SXD     *+2,1             X                                    FT122360
      TSX     LIST,4            OUTPUT LAST LINE                     FT122370
      PON     LIN10,,..         DOUBLE IFF THIS IS ONLY LINE TO GO OUTFT122380
MAP30 AXT     STASH,1           RESTORE STVECT TO NORMAL             FT122390
      SXA     STV1+2,1          X                                    FT122400
      SXA     STV2+1,1          X                                    FT122410
      CAL     MAP3-4            RESTORE PON'S                        FT122420
      STP     MAP4+4            X                                    FT122430
      STP     MAP6+6            X                                    FT122440
      NZT     TVSW2             CLEAR SYMTAB OF EXTERNAL REFS.       FT122450
      TRA     *+6                                                    FT122460
      LXA     TVSW2,1                                                FT122470
      STZ     SYMTAB,1                                               FT122480
      CAL     EQUIV,1                                                FT122490
      PDX     ,1                                                     FT122500
      TXH     *-3,1,0                                                FT122510
      AXT     16,1              INITIALIZE OUTPUT AREA TO BLANKS     FT122520
      CAL     BLANKS            X                                    FT122530
      SLW     LIN10+17,1        X                                    FT122540
      TIX     *-1,1,1           X                                    FT122550
```

```
      EJECT                                                                FT122560
      REM     ... TO SET-UP 'POOL' AREA WITH NUMBERED STATEMENTS,          FT122570
      REM           FORMATS, AND ALL COMBINATIONS OF DIMENSIONED AND       FT122580
      REM           AND NON-DIMENSIONED AND COMMON AND NON-COMMON          FT122590
      REM           VARIABLES                                              FT122600
      REM                                                                  FT122610
MAP7  AXT     LSYMTB,1          INITIALIZE FOR SYMTAB SEARCH               FT122620
      AXT     1,2               X                                          FT122630
      ZET     SYMTAB,1          IS NEXT SYMTAB ENTRY ZERO                  FT122640
      TRA     MAP8              NO, GO SEE WHAT KIND OF SYMBOL IT IS       FT122650
      TIX     *-2,1,1           YES, GO LOOK AT NEXT SYMBOL                FT122660
      TRA     MAP29             GO PUT OUT LAST PORTION OF MAP             FT122670
MAP8  LDI     EQUIV,1           GET EQUIV ENTRY FOR THIS SYMBOL            FT122680
      LFT     400000            WAS THIS ENTRY IN ERROR                    FT122690
      TRA     MAP7+4            YES, GO GET NEXT ENTRY                     FT122700
      IIL     700000            NO, INVERT SYMTAB MODE BITS                FT122710
      LNT     /MLABL*700000     IS THIS A STATEMENT LABEL                  FT122720
      TRA     MAP10             NO, GO SEE IF STRING                       FT122730
      CAL     EQUIV,1           YES, CHAIN TO LAST LABEL                   FT122740
      STA     POOL,2            X                                          FT122750
      PXD     ,1                X                                          FT122760
      STD     POOL,2            POOL ENTRY = DEFN.,,SYMTAB POINTER         FT122770
      STZ     POOL-1,2          SET NEXT POOL ENTRY TO ZERO                FT122780
      NZT     GARB15            IS THIS FIRST LABEL SO FAR                 FT122790
      TRA     MAP9              YES, GO INITIALIZE CHAINING                FT122800
      LXD     GARB15,4          NO, CHAIN THIS LABEL TO LAST ONE           FT122810
      PXA     ,2                X                                          FT122820
      STA     POOL-1,4          X                                          FT122830
      SXD     GARB15,2          X                                          FT122840
      TXI     MAP7+4,2,2        BUMP POOL POINTER FOR NEXT ENTRY           FT122850
MAP9  SXA     GARB15,2          FIRST LABEL ENCOUNTERED                    FT122860
      SXD     GARB15,2          SET-UP FIRST CHAIN LINK                    FT122870
      TXI     MAP7+4,2,2        BUMP POOL POINTER FOR NEXT ENTRY           FT122880
MAP10 LNT     /MSTRG*700000     IS THIS A STRING                           FT122890
      TRA     MAP12             NO, GO SEE IF VARIABLE                     FT122900
      CAL     EQUIV,1           YES, GET EQUIV ENTRY                       FT122910
      STA     POOL,2            X                                          FT122920
      PXD     ,1                X                                          FT122930
      STD     POOL,2            POOL ENTRY = DEFN.,,SYMTAB POINTER         FT122940
      STZ     POOL-1,2          SET NEXT POOL ENTRY TO ZERO                FT122950
      NZT     GARB16            IS THIS FIRST STRING SO FAR                FT122960
      TRA     MAP11             YES, GO INITIALIZE CHAINING                FT122970
      LXD     GARB16,4          NO, GO CHAIN THIS STRING TO LAST ONE       FT122980
      PXA     ,2                X                                          FT122990
      STA     POOL-1,4          X                                          FT123000
      SXD     GARB16,2          X                                          FT123010
      TXI     MAP7+4,2,2        BUMP POOL POINTER FOR NEXT ENTRY           FT123020
MAP11 SXA     GARB16,2          FIRST STRING ENCOUNTERED                   FT123030
      SXD     GARB16,2          SET-UP FIRST CHAIN LINK                    FT123040
      TXI     MAP7+4,2,2        BUMP POOL POINTER FOR NEXT ENTRY           FT123050
MAP12 LFT     BARGT             IS THIS A SUBROUTINE PARAMETER             FT123060
      TRA     MAP7+4            YES, DO NOT PROCESS THIS ENTRY             FT123070
      LNT     /MINTG*700000     NO, IS THIS AN INTEGER VARIABLE            FT123080
      TRA     MAP14             NO, GO SEE IF REAL VARIABLE                FT123090
      LNT     BCOMN             YES, IS THIS VARIABLE IN COMMON            FT123100
      TRA     MAP15             NO, GO SEE IF IT'S AN ARRAY                FT123110
      LNT     BARRY             YES, IS IT AN ARRAY                        FT123120
      TRA     MAP19             NO, MUST BE NON-DIMENSIONED, COMMON        FT123130
      CAL     EQUIV,1           CHAIN THIS TO LAST DIMEN., COMMON VAR.     FT123140
      STA     POOL,2            X                                          FT123150
```

```
        PXD     ,1              X                                       FT123160
        STD     POOL,2          POOL ENTRY = DEFN.,,SYMTAB POINTER      FT123170
        STZ     POOL-1,2        SET NEXT POOL ENTRY TO ZERO             FT123180
        NZT     GARB20          IS THIS FIRST DIMEN., COMMON SO FAR     FT123190
        TRA     MAP13           YES, GO INITIALIZE CHAINING             FT123200
        LXD     GARB20,4        NO, CHAIN THIS VARIABLE TO LAST ONE     FT123210
        PXA     ,2              X                                       FT123220
        STA     POOL-1,4        X                                       FT123230
        SXD     GARB20,2        X                                       FT123240
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123250
MAP13   SXA     GARB20,2        FIRST DIMEN., COMMON ENCOUNTERED        FT123260
        SXD     GARB20,2        SET-UP FIRST CHAIN LINK                 FT123270
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123280
MAP14   LNT     /MREAL*700000   IS THIS A REAL VARIABLE                  FT123290
        TRA     MAP7+4          NO, DO NOT PROCESS THIS SYMTAB ENTRY    FT123300
        TRA     MAP12+4         YES, GO TO VARIABLE PROCESSOR           FT123310
MAP15   LNT     BARRY           IS THIS AN ARRAY                        FT123320
        TRA     MAP17           NO, MUST BE NON-DIMEN., NON-COMMON      FT123330
        CAL     EQUIV,1         CHAIN TO LAST DIMEN., NON-COMMON VAR.   FT123340
        STA     POOL,2          X                                       FT123350
        PXD     ,1              X                                       FT123360
        STD     POOL,2          POOL ENTRY = DEFN.,,SYMTAB POINTER      FT123370
        STZ     POOL-1,2        SET NEXT POOL ENTRY TO ZERO             FT123380
        NZT     GARB18          IS THIS FIRST DIMEN., NON-COMMON VAR.   FT123390
        TRA     MAP16           YES, GO INITIALIZE CHAINING             FT123400
        LXD     GARB18,4        NO, CHAIN THIS VARIABLE TO LAST ONE     FT123410
        PXA     ,2              X                                       FT123420
        STA     POOL-1,4        X                                       FT123430
        SXD     GARB18,2        X                                       FT123440
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123450
MAP16   SXA     GARB18,2        FIRST DIMEN., NON-COMMON ENCOUNTERED    FT123460
        SXD     GARB18,2        SET-UP FIRST CHAIN LINK                 FT123470
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123480
MAP17   CAL     EQUIV,1         CHAIN THIS TO LASTNON-DIMEN., NON-COMMFT123490
        STA     POOL,2          X                                       FT123500
        PXD     ,1              X                                       FT123510
        STD     POOL,2          POOL ENTRY = DEFN.,,SYMTAB POINTER      FT123520
        STZ     POOL-1,2        SET NEXT POOL ENTRY TO ZERO             FT123530
        NZT     GARB17          IS THIS FIRST NON-DIMEN., NON-COMMON    FT123540
        TRA     MAP18           YES, GO INITIALIZE CHAINING             FT123550
        LXD     GARB17,4        NO, CHAIN THIS VARIABLE TO LAST ONE     FT123560
        PXA     ,2              X                                       FT123570
        STA     POOL-1,4        X                                       FT123580
        SXD     GARB17,2        X                                       FT123590
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123600
MAP18   SXA     GARB17,2        FIRST NON-DIMEN., NON-COMMON HIT        FT123610
        SXD     GARB17,2        SET-UP FIRST CHAIN LINK                 FT123620
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123630
MAP19   CAL     EQUIV,1         CHAIN THIS TO LAST NON-DIMEN., COMMON   FT123640
        STA     POOL,2          X                                       FT123650
        PXD     ,1              X                                       FT123660
        STD     POOL,2          POOL ENTRY = DEFN.,,SYMTAB POINTER      FT123670
        STZ     POOL-1,2        SET NEXT POOL ENTRY TO ZERO             FT123680
        NZT     GARB19          IS THIS FIRST NON-DIMEN., COMMON        FT123690
        TRA     MAP20           YES, GO INITIALIZE CHAINING             FT123700
        LXD     GARB19,4        NO, CHAIN THIS VARIABLE TO LAST ONE     FT123710
        PXA     ,2              X                                       FT123720
        STA     POOL-1,4        X                                       FT123730
        SXD     GARB19,2        X                                       FT123740
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123750
```

```
MAP20   SXA     GARB19,2        FIRST NON-DIMEN., COMMON ENCOUNTERED    FT123760
        SXD     GARB19,2        SET-UP FIRST CHAIN LINK                 FT123770
        TXI     MAP7+4,2,2      BUMP POOL POINTER FOR NEXT ENTRY        FT123780
        REM                                                             FT123790
MAP29   PXA     ,2              SET NXTLOC = MAX(NXTLOC, NO. POOL       FT123800
        LAS     NXTLOC           CELLS USED BY MAP)                     FT123810
        SLW     NXTLOC          X                                       FT123820
        TRA     *+2             X                                       FT123830
        NOP                     X                                       FT123840
```

```
                EJECT                                                                FT123850
                REM     ... TO OUTPUT LOCATIONS OF NUMBERED STATEMENTS, FORMATS,     FT123860
                REM           AND VARIABLES                                          FT123870
                REM                                                                  FT123880
MAP21           STZ     ERASE       'NO LINE THIS SECTION YET' SWITCH                FT123890
                STZ     ERASE1      'EXIT THIS SECTION' SWITCH                       FT123900
                AXT     0,1         INITIALIZE SECTION COUNTER                       FT123910
                SXA     MAP27+2,1   SAVE SECTION COUNTER                             FT123920
                NZT     GARB15,1    IS THIS SECTION EMPTY                            FT123930
                TRA     MAP27+2     YES, GO LOOK AT NEXT SECTION                     FT123940
                CAL     GARB15,1    GET FIRST POOL ENTRY FOR THIS SECTION            FT123950
                PAX     ,2          X                                                FT123960
                CAL     POOL,2      DEFN.,,SYMTAB POINTER                            FT123970
                PDX     ,4          XR4 = SYMTAB POINTER                             FT123980
                CAL     SYMTAB,4    PICK-UP THIS SYMBOL                              FT123990
                TSX     RTJUST,4    RIGHT JUSTIFY SYMBOL                             FT124000
MAP22           AXC     **,4        CURRENT LINE POSITION POINTER                    FT124010
                SLW     LIN10-1,4   PUT THIS SYMBOL IN MESSAGE                       FT124020
                CAL     POOL,2      GET POOL ENTRY AGAIN                             FT124030
                LXA     MAP27+2,4   GET CURRENT SECTION POINTER                      FT124040
                TXL     *+3,4,3     SUBTRACT FROM TOPCOM IF COMMON SECT.             FT124050
                ANA     =077777      X                                               FT124060
                SSM                  X                                               FT124070
                ADD*    GARB23,4    ADD PROPER CONSTANT                              FT124080
                TSX     OCTBCD,4    CONVERT LOCATION TO OCTAL                        FT124090
                ARS     6           RIGHT-ADJUST                                     FT124100
                ORA     GARB2       LEADING BLANK                                    FT124110
                XEC     MAP22       GET LINE POSITION POINTER                        FT124120
                SLW     LIN10,4     STORE THIS LOCATION IN OUTPUT LINE               FT124130
                LXA     MAP22,4     SEE IF LINE IS FULL                              FT124140
                TXH     MAP24-1,4,12 TRANSFER IF LINE IS FULL                        FT124150
                TXI     *+1,4,3     OTHERWISE BUMP POINTER FOR NEXT ENTRY            FT124160
MAP23           SXA     MAP22,4     SAVE UPDATED LINE POSITION POINTER               FT124170
                CAL     POOL-1,2    IS THERE ANPTHER POOL ENTRY                      FT124180
                PAX     ,2          X                                                FT124190
                TXL     MAP26,2,0   NO, GO FINISH THIS SECTION                       FT124200
                TRA     MAP22-4     YES, GO PROCESS NEXT ENTRY                       FT124210
                TXI     *+3,4,-2    GET PROPER WORD COUNT                            FT124220
                TXI     *+2,4,1     D I T T O                                        FT124230
MAP24           LXA     MAP22,4     PREPARE TO OUTPUT A LINE THIS SECTION            FT124240
                SXD     MAP24+5,4   SAVE LINE WORD COUNT                             FT124250
                NZT     ERASE       IS THIS FIRST LINE THIS SECTION                  FT124260
                TRA     MAP25       YES, GO PUT OUT HEADINGS                         FT124270
                TSX     LIST,4      OUPUT LINE OF THIS SECTION                       FT124280
                PZE     LIN10,,..   SINGLE SPACED (ALWAYS)                          FT124290
                ZET     ERASE1      IS THIS LAST LINE THIS SECTION                   FT124300
                TRA     MAP27       YES, GO GET NEXT SECTION                         FT124310
                AXT     2,4         NO, PREPARE FOR NEXT LINE                        FT124320
                SXA     MAP22,4     X                                                FT124330
                TRA     MAP23       GO GET NEXT LINE                                 FT124340
MAP25           STL     ERASE       SET 'NO LINE THIS SECTION YET' SWITCH            FT124350
                TSX     LIST,4      PUT OUT A BLANK LINE                             FT124360
                PON     BLANKS,,1   DOUBLE SPACED                                    FT124370
                LXA     MAP27+2,4   GET CURRENT SECTION POINTER                      FT124380
                CAL     GARB21,4    SET-UP LIST CALL FOR HEADING                     FT124390
                SLW     *+2         X                                                FT124400
                TSX     LIST,4      OUTPUT THIS SECTION HEADING                      FT124410
                PON     ..,,..      ADDR. AND DECR. SET ABOVE                        FT124420
                LXD     MAP24+5,4   GET CURRENT LINE LENGTH                          FT124430
                SXD     *+5,4       SAVE IN LIST CALL                                FT124440
```

```
        LXA     MAP27+2,4       GET CURRENT SECTION POINTER                     FT124450
        CAL     GARB22,4        GET LOCATION OF SECONDARY HEADING               FT124460
        STA     *+2             SAVE MESSAGE LOCATION IN LIST CALL              FT124470
        TSX     LIST,4          OUTPUT SECONDARY HEADING                        FT124480
        PON     ..,,..          X                                               FT124490
        TRA     MAP24+4         GO OUTPUT LINE THIS SECTION                     FT124500
MAP26   STL     ERASE1          SET 'EXIT THIS SECTION' SWITCH                  FT124510
        LXA     MAP22,4         GET CURRENT LINE POSITION                       FT124520
        TXH     MAP24-2,4,2     TRANSFER IF LAST LINE NOT FULL                  FT124530
MAP27   STZ     ERASE           RESET 'NO LINE THIS SECTION YET'                FT124540
        STZ     ERASE1          RESET 'EXIT THIS SECTION' SWITCH                FT124550
        AXT     ..,1            XR1 = CURRENT SECTION NUMBER                    FT124560
        TXH     MAP31,1,4       GO TO M A P 3 1 IF LAST (THE SIXTH)             FT124570
        TXI     *+1,1,1          SECTION IS OUT. OTHERWISE GO GET               FT124580
        SXA     *-3,1            NEXT SECTION                                   FT124590
        AXT     2,2             RESTORE MAP22                                   FT124600
        SXA     MAP22,2         X                                               FT124610
        TRA     MAP21+4         GO GET NEXT SECTION                             FT124620
```

```
        EJECT                                                       FT124630
        REM     ... TO OUTPUT SYMBOL TABLE MESSAGE IF NECESSARY     FT124640
        REM                                                         FT124650
MAP31   NZT     STOUT           WAS SYMBOL TABLE PUNCHED            FT124660
        TRA     FINISH          NO, GO TO F I N I S H               FT124670
        TSX     LIST,4          OUPUT SYMBOL TABLE PUNCHED MESSAGE  FT124680
        PON     BLANKS,1        X                                   FT124690
        TSX     LIST,4          X                                   FT124700
        PON     LIN19,,16       X                                   FT124710
        TRA     FINISH          FINAL M A P EXIT                    FT124720
```

```
        EJECT                                                       FT124730
        REM     ... GARBAGE AREA FOR MAP                            FT124740
        REM                                                         FT124750
MAPORG                                                              FT124760
 GARB1  BCI     2,MAIN PROGRAM                                      FT124770
 GARB2  BCI     1, 00000                                            FT124780
 GARB3  BCI     1,00000                                             FT124790
        BCI     1,1,2,4                                             FT124800
        BCI     1,2,4                                               FT124810
        BCI     1,1,4                                               FT124820
        BCI     1,4                                                 FT124830
        BCI     1,1,2                                               FT124840
        BCI     1,2                                                 FT124850
        BCI     1,1                                                 FT124860
 GARB4  BCI     1,NONE                                              FT124870
                GARB6+4,2                                           FT124880
                GARB7+4,2                                           FT124890
                GARB7A+4,2                                          FT124900
                GARB8+4,2                                           FT124910
                GARB10+4,2                                          FT124920
                GARB11+4,2                                          FT124930
                GARB12+4,2                                          FT124940
                GARB13+4,2                                          FT124950
 GARB5  BSS     0                                                   FT124960
 GARB6  BCI     4,  TRANSFER VECTOR                                 FT124970
 GARB7  BCI     4,  PROLOGUE                                        FT124980
GARB7A  BCI     4,  ARITH. STMNT. FCNS.                             FT124990
 GARB8  BCI     4,  TEXT                                            FT125000
GARB10  BCI     4,  CONSTANTS                                       FT125010
GARB11  BCI     4,  FORMATS                                         FT125020
GARB12  BCI     4,  WORKING STORAGE                                 FT125030
GARB13  BCI     4,  LOW-CORE VARIABLES                              FT125040
        PON     LIN18,,9                                            FT125050
        PON     LIN17,,9                                            FT125060
        PON     LIN16,,9                                            FT125070
        PON     LIN14,,10                                           FT125080
        PON     LIN13,,7                                            FT125090
GARB21  PON     LIN11,,9                                            FT125100
                LIN15                                               FT125110
                LIN15                                               FT125120
                LIN15                                               FT125130
                LIN15                                               FT125140
                LIN12                                               FT125150
GARB22          LIN12                                               FT125160
                TOPCOM                                              FT125170
                TOPCOM                                              FT125180
                LTHRUW                                              FT125190
                LTHRUW                                              FT125200
                LTPTSC                                              FT125210
GARB23          LTVPRL                                              FT125220
  TEMP                                                              FT125230
```

```
      EJECT                                                                 FT125240
      REM     ... BCI'S FOR MAP                                             FT125250
      REM                                                                   FT125260
 LIN1 BCI     4,                                                            FT125270
      BCI     7,      FASTRAN STORAGE MAP FOR MAIN PROGRAM                  FT125280
 LIN2 BCI ,   STORAGE NOT USED BY PROGRAM- XXXXX  TO  XXXXX  (OCTAL)        FT125290
 LIN3 BCI ,                             XXXXX  TO  XXXXX  (DECIMAL)        FT125300
 LIN4 BCI     8,  INDEX REGISTERS USED BY THIS PROGRAM-   XXXXXX            FT125310
 LIN5 BCI     6,                                                            FT125320
      BCI     7,     ORIGIN                          LENGTH                 FT125330
 LIN6 BCI     6,                                                            FT125340
      BCI     8,OCTAL  --  DECIMAL              OCTAL  --  DECIMAL         FT125350
 LIN7 BCI     6,                                                            FT125360
      BCI     8,                                                            FT125370
 LIN8 BCI     9,  COMMON STORAGE                      XXXXX       XXXXX     FT125380
      BCI     5,             XXXXX       XXXXX                              FT125390
 LIN9 BCI     3,  TRANSFER VECTOR-                                          FT125400
LIN10 BCI     8,                                                            FT125410
      BCI     9,                                                            FT125420
LIN11 BCI     9,  OCTAL LOCATIONS OF NUMBERED STATEMENTS IN TEXT-           FT125430
LIN12 BCI     9,        ST NO - LOC       ST NO - LOC       ST NO - LOC   FT125440
      BCI     6,        ST NO - LOC       ST NO - LOC                     FT125450
LIN13 BCI     7,  OCTAL LOCATIONS OF FORMAT STATEMENTS-                     FT125460
LIN14 BCI ,   OCTAL LOCATIONS OF NON-DIMENSIONED NON-COMMON VARIABLES-      FT125470
LIN15 BCI     9,         NAME - LOC        NAME - LOC        NAME - LOC    FT125480
      BCI     6,         NAME - LOC        NAME - LOC                      FT125490
LIN16 BCI     9,  OCTAL LOCATIONS OF DIMENSIONED NON-COMMON VARIABLES-      FT125500
LIN17 BCI     9,  OCTAL LOCATIONS OF NON-DIMENSIONED COMMON VARIABLES-      FT125510
LIN18 BCI     9,  OCTAL LOCATIONS OF DIMENSIONED COMMON VARIABLES-          FT125520
LIN19 BCI ,   * * * SYMBOL TABLE (LABELLED ST000000 IN COLS. 73-80) PREF    T125530
      BCI     6,CEDES BINARY OBJECT DECK * * *                              FT125540
```

```
         TTL     P O S T  P A S S  2  -  F I N I S H                    FT125550
*        THE FOLLOWING ROUTINE IS TEMPORARY AND AUTOMATICALLY SENDS      FT125560
*        CONTROL BACK TO THE DRIVER IF THE SYSTEM INPUT TAPE IS NOT AT   FT125570
*        AN END-OF-FILE.                                                 FT125580
         REM                                                             FT125590
*        THIS WILL SOON BE REPLACED WITH REGARD TO FASTRAN RESIDING IN   FT125600
*        A MONITOR SYSTEM.                                               FT125610
         SPACE   3                                                       FT125620
FINISH   ZET     MONTOR          IF UNDER IU MONITOR, GO TO F I N 3      FT125630
         TRA     FIN3            X                                       FT125640
         ZET     ENFILF          ARE WE AT END-OF-FILE                   FT125650
         TRA     FIN1            YES, GO STOP                            FT125660
IOS38    EQU     *                                                       FT125670
         BACKR   MINTAP,1        BACKSPACE OVER LAST RECORD (THIS WAS    FT125680
         STL     IOEX                                                    FT125690
         BRA     BACKR,,*+2                                              FT125700
         MZE     MINTAP,,1                                               FT125710
END38    REM                                                             FT125720
         TRA     START           ORIGINALLY READ TO HANDLE CON-          FT125730
         REM                     TINUATION CARDS.                        FT125740
         REM                                                             FT125750
IOS39    EQU     *                                                       FT125760
  FIN1   WEOF    *MLSTAP         WRITE END-OF-FILE ON OUTPUT TAPE        FT125770
FIN1     STL     IOEX                                                    FT125780
         BRA     WEOF,,*+2                                               FT125790
         MTW     MLSTAP                                                  FT125800
END39    REM                                                             FT125810
IOS40    EQU     *                                                       FT125820
         BACKR   *MLSTAP,1       BACKSPACE OVER THIS FILE MARK IN CASE   FT125830
         STL     IOEX                                                    FT125840
         BRA     BACKR,,*+2                                              FT125850
         MTW     MLSTAP,,1                                               FT125860
END40    REM                                                             FT125870
         REM                     ANOTHER INPUT TAPE IS COMING ALONG      FT125880
         SWT     6               PRINT THE FINAL TIME IF SS 6 DOWN       FT125890
         TRA     *+3                                                     FT125900
IOS41    EQU     *                                                       FT125910
         RCLOCK                  PRINT THE TIME                          FT125920
         STL     IOEX                                                    FT125930
         BRA     RCLOCK,,*+1                                             FT125940
END41    REM                                                             FT125950
         REM                                                             FT125960
IOS42    EQU     *                                                       FT125970
  FIN4   CLOSE   MINTAP          CLOSE SYSTEM INPUT TAPE                 FT125980
FIN4     STL     IOEX                                                    FT125990
         BRA     CLOSE,,*+2                                              FT126000
         MZE     MINTAP                                                  FT126010
END42    REM                                                             FT126020
IOS43    EQU     *                                                       FT126030
         CLOSE   *MLSTAP         CLOSE SYSTEM OUTPUT TAPE                FT126040
         STL     IOEX                                                    FT126050
         BRA     CLOSE,,*+2                                              FT126060
         MTW     MLSTAP                                                  FT126070
END43    REM                                                             FT126080
IOS44    EQU     *                                                       FT126090
         CLOSE   *SYSBIN                                                 FT126100
         STL     IOEX                                                    FT126110
         BRA     CLOSE,,*+2                                              FT126120
         MTW     SYSBIN                                                  FT126130
END44    REM                                                             FT126140
```

```
IOS58   EQU     *
        REWIND  SYSUT1          REWIND FASTRAN'S SCRATCH TAPE
        STL     IOEX
        BRA     REWIND,,*+2
        MZE     SYSUT1
*       THIS INSTRUCTION IS REFERRING TO AN IOS LOCATION*****
        STZ     BUUT1           CLEAR SYSUT1'S BUFTBL ENTRY
BUUT1   BOOL    233
END58   REM
        REM
        NZT     MONTOR          WERE WE UNDER IU MONITOR
        TRA     FIN2            NO.
IOS45   EQU     *
        LOAD    ..,1,10         LOAD IOS/FMS-II MONITOR
        STL     IOEX
        BRA     LOAD,,*+3
        PZE     **,,1
        MZE     10
END45   REM
        REM
*       ... FOR NON-MONITOR OPERATION (SWITCH 5 UP)
        REM
IOS46   EQU     *
FIN2    PRINT   START4,4        GIVE OPERATOR DISMOUNT INSTRUCTIONS
FIN2    STL     IOEX
        BRA     PRINT,,*+2
        MZE     START4,,4
END46   REM
        LDQ     ALLSVN          ARE LAMPS IN WORKING ORDER
        CAL     ALLSVN
        HTR     START           HALT... WITH RETURN TO INITIALIZATION
        REM                     SECTION IF THERE IS ANOTHER DATA TAPE.
        SPACE   3
START4  BCI     4,0DISMOUNT AND LIST A-5.
        SPACE   3
  FIN3  LDI     LASTLA          IS NEXT CARD AN IU MONITOR CC
        IIR     410000          RETURN TO MONITOR IFF NEXT CARD IS
        LFT     77               NOT BCD OR IF NEXT CARD IS AN IU/FMS
        TRA     IOS47            CONTROL CARD (COL. 1 = 12-6-8)
        RNT     770000
        TRA     IOS38           ALLOW FASTRAN TO RETAIN CONTROL
IOS47   EQU     *
        BACKR   MINTAP,2        PREPARE TO RETURN TO MONITOR
        STL     IOEX
        BRA     BACKR,,*+2
        MZE     MINTAP,,2
END47   REM
        TRA     FIN4            X
```

```
FT126150
FT126160
FT126170
FT126180
FT126190
FT126200
FT126210
FT126220
FT126230
FT126240
FT126250
FT126260
FT126270
FT126280
FT126290
FT126300
FT126310
FT126320
FT126330
FT126340
FT126350
FT126360
FT126370
FT126380
FT126390
FT126400
FT126410
FT126420
FT126430
FT126440
FT126450
FT126460
FT126470
FT126480
FT126490
FT126500
FT126510
FT126520
FT126530
FT126540
FT126550
FT126560
FT126570
FT126580
FT126590
FT126600
FT126610
FT126620
```

```
        TTL     P O S T   P A S S   2   -   S U B R O U T I N E S          FT126630
*       POST PASS 2 HAS BUT ONE SIMPLE MINDED SUBROUTINE.  IT IS USED      FT126640
*       DURING THE MAPPING OF THE OBJECT PROGRAM AND IS USED TO RIGHT      FT126650
*       ADJUST A LEFT-ADJUSTED BCD SYMPOL IN THE ACCUMULATOR AND INSERT    FT126660
*       LEADING BLANKS.                                                    FT126670
        SPACE   3                                                          FT126680
RTJUST  STI     ERASE2          SAVE INDICATORS                            FT126690
        PAI                     PUT ARGUMENT INTO INDICATORS               FT126700
        IIR     17              LOOK AT LOW ORDER CHARACTER                FT126710
        RNT     77              IS IT A BLANK                              FT126720
        TRA     RESET           NO, GO EXIT THIS SUBROUTINE                FT126730
        ARS     6               YES, MOVE 1 CHAR. TO RIGHT                 FT126740
        ORA     GARB2            AND PUT IN LEADING BLANK                  FT126750
        TRA     RTJUST+1        GO BACK FOR NEXT CHARACTER                 FT126760
RESET   LDI     ERASE2          RESTORE INDICATORS                         FT126770
        TRA     1,4             FINAL SUBROUTINE EXIT                      FT126780
```

```
        TTL     U T I L I T Y  R O U T I N E S  -  L I S T                FT126790
        REM           SYSTEM OUTPUT TAPE LISTER -- LIST                  FT126800
        REM                                                              FT126810
        REM     THIS ROUTINE IS USED TO WRITE LISTING OUTPUT ON THE      FT126820
        REM      SYSTEM OUTPUT TAPE MLSTAP.  PAGE TITLING AND LINE AND   FT126830
        REM      PAGE COUNTING IS HANDLED BY THIS ROUTINE.               FT126840
        REM                                                              FT126850
        REM          CALLING SEQUENCE,                                   FT126860
        REM                                                              FT126870
        REM                        TSX   LIST,4                          FT126880
        REM                        (OP)  LOC,,COUNT                      FT126890
        REM                           (RETURN)                           FT126900
        REM                                                              FT126910
        REM      WHERE,                                                  FT126920
        REM           LOC = ORIGIN OF MESSAGE TO GO ON MLSTAP            FT126930
        REM           COUNT = LENGTH OF MESSAGE (WORDS)                  FT126940
        REM           (OP) = PZE IFF THIS LINE TO BE SINGLE SPACED       FT126950
        REM                = PON IFF THIS LINE TO BE DOUBLE SPACED       FT126960
        REM                = PTW IFF PAGE EJECT BEFORE THIS LINE GOES OUTFT126970
        REM                                                              FT126980
        REM     WHEN MLSTAP IS PRINTED, THE FIRST CHARACTER OF EACH      FT126990
        REM      MESSAGE IS PRINTED BEGINNING WITH PRINT POSITION SEVEN. FT127000
        REM                                                              FT127010
        REM     IF LOCATION PRIQQ IS NON-ZERO THE MESSAGE GOES OUT       FT127020
        REM      ON-LINE AS WELL AS ON MLSTAP.  ON-LINE PRINTING HAS     FT127030
        REM      FIRST CHARACTER OF THE MESSAGE IN PRINT POSITION ONE ANDFT127040
        REM      THUS PRINT CONTROL IS HANDLED BY THE USER.              FT127050
        SPACE   2                                                        FT127060
LIST    SXA     EXLIST,4        SAVE XR4 FOR RETURN                      FT127070
        CAL     1,4             SET ACC TO ARGUMENT                      FT127080
        SLW     LIS5            SAVE 1,4 IN TEMPORARY SAVE CELL          FT127090
        ARS     15              BRING (OP) INTO DECREMENT                FT127100
        PDX     ,4              PUT (OP) IN XR4                          FT127110
        TXH     LIS11,4,2       GO GIVE ERROR MESSAGE IF INVALID (OP)    FT127120
        TRA*    LISTV,4         GO PROCESS THIS (OP)                     FT127130
        REM                                                              FT127140
        REM     ... SINGLE SPACE                                         FT127150
        REM                                                              FT127160
  LIS1  LDQ     BLANKS          SET WORD ONE TO BLANKS                   FT127170
        CLA     LIS7            GET CURRENT LINE COUNT                   FT127180
        ADD     LIS6,4          BUMP BY PROPER AMOUNT                    FT127190
        CAS     LPERPG          LPERPG = MAX. LINES PER PAGE             FT127200
        TRA     LIS4            GO TO LIS4 IF THIS WON'T FIT             FT127210
        NOP                     O.K.                                     FT127220
        STO     LIS7            O.K., RESTORE LINE COUNT                 FT127230
  LIS2  STQ     LIS9            STORE FIRST WORD                         FT127240
        CLA     LIS5            GET ARGUMENT                             FT127250
        PDX     ,4              TAKE COUNT MOD 21                        FT127260
        TIX     *,4,21           X                                       FT127270
        STA     *+6             SET-UP WRITE MACRO                       FT127280
        SXD     *+5,4            X                                       FT127290
IOS48   EQU     *                                                        FT127300
        WRITE   *MLSTAP,DEC,LIS9,1,,,,,    WRITE THIS MESSAGE            FT127310
        STL     IOEX                                                     FT127320
        BRA     WRITE,,*+4                                               FT127330
        PTW     MLSTAP,,DEC                                              FT127340
        PZE     LIS9,,1                                                  FT127350
        MZE     **,,**                                                   FT127360
END48   REM                                                              FT127370
        NZT     PRIQQ           IS ON-LINE LISTING REQUESTED             FT127380
```

```
        TRA     EXLIST          NO, EXIT THE SCENE                      FT127390
        LDQ     *-3             SET-UP PRINT MACRO                      FT127400
        STQ     *+3              X                                      FT127410
IOS49   EQU     *                                                       FT127420
        PRINT   ..,..       PRINT THIS MESSAGE ON-LINE                  FT127430
        STL     IOEX                                                    FT127440
        BRA     PRINT,,*+2                                              FT127450
        MZE     **,,**                                                  FT127460
END49   REM                                                             FT127470
EXLIST  AXT     ..,4            RESTORE XR4 FOR RETURN                  FT127480
        TRA     2,4             DEPART                                  FT127490
        REM                                                             FT127500
        REM     ... DOUBLE SPACE                                        FT127510
        REM                                                             FT127520
  LIS3  LDQ     LIS10           SET FIRST CHARACTER OF LINE TO ZERO     FT127530
        TRA     LIS1+1          GO PUT THIS LINE OUT                    FT127540
        REM                                                             FT127550
        REM     ... EJECT                                               FT127560
        REM                                                             FT127570
LIS4    CAL     =3          RESET LINE COUNT TO 3 (START PRINTING       FT127580
        SLW     LIS7        SOURCE LISTING ON 3RD LINE OF PAGE).        FT127590
        SXA     LIS1X,4          SAVE XR4                               FT127600
        CLA     LIS8             BUMP PAGE COUNT BY ONE                 FT127610
        ADD     =1                X                                     FT127620
        STO     LIS8              X                                     FT127630
        TSX     BINBCD,4         CONVERT PAGE COUNT TO BCD              FT127640
        LDQ     BLANKS      TRAILING BLANKS FOR PAGE COUNT              FT127650
        LGL     24                                                      FT127660
        SLW     TITLE+18                                                FT127670
IOS50   EQU     *                                                       FT127680
        WRITE   *MLSTAP,DEC,TITLE,19                                    FT127690
        STL     IOEX                                                    FT127700
        BRA     WRITE,,*+3                                              FT127710
        PTW     MLSTAP,,DEC                                             FT127720
        MZE     TITLE,,19                                               FT127730
END50   REM                                                             FT127740
        REM                                                             FT127750
        NZT     SBTTL           IS THERE A SUBTITLE.                    FT127760
        TRA     LIS12           NO                                      FT127770
IOS51   EQU     *                                                       FT127780
        WRITE   *MLSTAP,DEC,SUBTTL,15       WRITE THE SUBTITLE          FT127790
        STL     IOEX                                                    FT127800
        BRA     WRITE,,*+3                                              FT127810
        PTW     MLSTAP,,DEC                                             FT127820
        MZE     SUBTTL,,15                                              FT127830
END51   REM                                                             FT127840
        CAL     =4              SET LINE COUNT TO 4 INSTEAD OF 3.       FT127850
        SLW     LIS7                                                    FT127860
        REM                                                             FT127870
IOS52   EQU     *                                                       FT127880
LIS12   WRITE   *MLSTAP,DEC,BLANKS,1                                    FT127890
LIS12   STL     IOEX                                                    FT127900
        BRA     WRITE,,*+3                                              FT127910
        PTW     MLSTAP,,DEC                                             FT127920
        MZE     BLANKS,,1                                               FT127930
END52   REM                                                             FT127940
        NZT     PRIQQ                IS ON-LINE LISTING REQUESTED       FT127950
        TRA     LIS1X                NO, EXIT THIS SECTION              FT127960
IOS53   EQU     *                                                       FT127970
        PRINT   TITLE,20,BLANKS,1  EJECT, TITLE AND SKIP ON-LINE        FT127980
```

```
        STL     IOEX                                                    FT127990
        BRA     PRINT,,*+3                                              FT128000
        PZE     TITLE,,20                                               FT128010
        MZE     BLANKS,,1                                               FT128020
END53   REM                                                             FT128030
 LIS1X  AXT     ..,4            RESTORE XR4                             FT128040
        LDQ     BLANKS          PREPARE FOR SINGLE SPACE                FT128050
        TRA     LIS2            GO PUT OUT THE ORIGINAL MESSAGE         FT128060
        SPACE   2                                                       FT128070
  LIS5                                                                  FT128080
                2                                                       FT128090
  LIS6          1                                                       FT128100
  LIS7                          CURRENT LINE COUNT                      FT128110
  LIS8                          CURRENT PAGE COUNT                      FT128120
  LIS9                                                                  FT128130
                                                                        FT128140
 LIS10  BCI     1,0                                                     FT128150
                LIS4                                                    FT128160
                LIS3                                                    FT128170
 LISTV          LIS1                                                    FT128180
LPERPG  PZE     59              ALLOWABLE LINES PER PAGE OF OUTPUT      FT128190
        REM                                                             FT128200
        REM     ... TO HANDLE INVALID (OP)                              FT128210
        REM                                                             FT128220
LIS11   MACER                   MACHINE ERROR                           FT128230
```

```
        TTL     U T I L I T Y   R O U T I N E S  -  S O R T                  FT128240
        REM                   HIGH-SPEED TABLE SORT                         FT128250
        REM                                                                 FT128260
        REM     THIS ROUTINE CAN BE USED TO SORT 1- OR 2-ELEMENT            FT128270
        REM      TABLES (WITH ELEMENT 1 OR 2 BEING THE KEY),                FT128280
        REM      IN ASCENDING OR DESCENDING SEQUENCE, LOGICAL               FT128290
        REM      OR ALGEBRAIC.                                              FT128300
        REM     CALLING SEQUENCE,                                           FT128310
        REM                                                                 FT128320
        REM          CALL SORT (A, B, I, J, K, L, M)                        FT128330
        REM       WHERE,                                                    FT128340
        REM             A = LOCATION OF FIRST KEY (LOW CORE)                FT128350
        REM             B = LOCATION OF LAST KEY (HIGH CORE)                FT128360
        REM             I = NO. ELEMENTS PER TABLE ENTRY (1 OR 2)           FT128370
        REM             J = ELEMENT WHICH IS KEY (1 OR 2)                   FT128380
        REM             K = 0 FOR ASCENDING, 1 FOR DESCENDING               FT128390
        REM             L = 0 FOR LOGICAL, 1 FOR ALGEBRAIC SORT             FT128400
        REM             M = LOCATION OF 'MASK' - ZERO IMPLIES NO MASKING    FT128410
        REM                 THE ARGUMENT 'M' MAY BE OMITTED FROM            FT128420
*                           THE CALLING SEQUENCE.                           FT128430
        REM                                                                 FT128440
        REM                                                                 FT128450
        REM     * * *  N O T E  * * *                                        FT128460
        REM     THIS SORT ROUTINE HAS BEEN SPECIALIZED (TO CONSERVE         FT128470
        REM      SPACE) TO HANDLE THE CASE WHERE,                           FT128480
        REM                                                                 FT128490
        REM             I = 1   (1 ELEMENT/ENTRY)                           FT128500
        REM             J = 1   (KEY IS ELEMENT 1)                          FT128510
        REM             K = 0   (ASCENDING SORT)                            FT128520
        REM             L = 0   (LOGICAL SORT)                              FT128530
        REM             M = 0   (NO MASKING)                                FT128540
        REM                                                                 FT128550
        REM                                                                 FT128560
        REM     THE CALLING SEQUENCE TO THIS SPECIALIZED VERSION IS,        FT128570
        REM                                                                 FT128580
-       REM             CALL   SORT,A,B                                     FT128590
        REM                                                                 FT128600
        REM     * * *  N O T E  * * *                                        FT128610
        REM                                                                 FT128620
SORT    SXA     EXSORT-3,1                                                  FT128630
        SXA     EXSORT-2,2                                                  FT128640
        SXA     EXSORT-1,4                                                  FT128650
        CAL     1,4                                                         FT128660
        PAX     0,1                                                         FT128670
        SXD     SRT1,1                                                      FT128680
        SXD     SRT10,1                                                     FT128690
        SXA     SRT18,1                                                     FT128700
        SXA     SRT19,1                                                     FT128710
        SXA     SRT21,1                                                     FT128720
        PAC     ,1                                                          FT128730
        SXD     SRT2,1                                                      FT128740
        AXT     1,1                                                         FT128750
        REM                                                                 FT128760
SRT1    TXI     *+1,1,**                                                    FT128770
        SXA     SRT22,1                                                     FT128780
        SXA     SRT25,1                                                     FT128790
        CAL     2,4                                                         FT128800
        PAX     0,1                                                         FT128810
SRT2    TXI     *+1,1,..                                                    FT128820
        TXL     EXSORT-3,1,0                                                FT128830
```

```
         SXA     SRT12,1                                          FT128840
         AXT     1,2                                              FT128850
         AXT     1,4                                              FT128860
SRT36    TXI     *+1,1,1                                          FT128870
SRT3     SXA     SRT31,1                                          FT128880
         SXD     SRT29,2                                          FT128890
SRT9     CLA     SRT31                                            FT128900
         ARS     2,4                                              FT128910
         ALS     1,4                                              FT128920
         STO     SRT31                                            FT128930
         TZE     EXSORT-3                                         FT128940
         PAC     0,1                                              FT128950
         SXD     SRT13,1                                          FT128960
         SXD     SRT27,1                                          FT128970
         PAX     0,1                                              FT128980
         SXD     SRT28,1                                          FT128990
SRT10    TXI     *+1,1,**                                         FT129000
         SXA     SRT17,1                                          FT129010
         SXA     SRT20,1                                          FT129020
         NOP                                                      FT129030
         TXH     SRT12,4,0                                        FT129040
SRT11    TXI     *+1,1,1                                          FT129050
         SXA     SRT23,1                                          FT129060
         SXA     SRT24,1                                          FT129070
         REM                                                      FT129080
SRT12    AXT     **,1                                             FT129090
SRT13    TXI     *+1,1,**                                         FT129100
         SXD     SRT30,1                                          FT129110
SRT15    AXT     0,1                                              FT129120
SRT16    PXA     0,1                                              FT129130
         PAC     0,2                                              FT129140
SRT17    CAL     ..,2                                             FT129150
SRT18    LAS     ..,2                                             FT129160
         TRA     SRT29                                            FT129170
         TRA     SRT29                                            FT129180
         REM                                                      FT129190
SRT19    LDQ     **,2                                             FT129200
SRT20    STQ     **,2                                             FT129210
SRT21    SLW     ..,2                                             FT129220
         TXH     SRT26,4,0                                        FT129230
         REM                                                      FT129240
SRT22    CAL     **,2                                             FT129250
SRT23    LDQ     **,2                                             FT129260
SRT24    SLW     **,2                                             FT129270
SRT25    STQ     **,2                                             FT129280
         REM                                                      FT129290
SRT26    TXL     SRT29,2,0                                        FT129300
SRT27    TXH     SRT29,2,**                                       FT129310
SRT28    TXI     SRT17,2,**                                       FT129320
         REM                                                      FT129330
SRT29    TXI     *+1,1,**                                         FT129340
SRT30    TXH     SRT9,1,**                                        FT129350
         TRA     SRT16                                            FT129360
SRT31                                                             FT129370
         REM                                                      FT129380
         AXT     **,1                                             FT129390
         AXT     **,2                                             FT129400
         AXT     **,4                                             FT129410
EXSORT   TRA     3,4                                              FT129420
```

```
        TTL     U T I L I T Y  -  S O R T 1                                 FT129430
*       THIS ROUTINE IS USED BY P S T (PUNCH SYMBOL TABLE) AND              FT129440
*       SPECIALIZES ROUTINE S O R T FOR A DESCENDING 2-ELEMENT SORT WITH    FT129450
*       SECOND ELEMENT THE KEY.  AT END, S O R T 1 RESTORES S O R T TO      FT129460
*       ITS ORIGINAL STATE.  CALLING SEQUENCE,                              FT129470
                                                                            FT129480
        REM                                                                 FT129490
*          CALL SORT1,A,B                                                   FT129500
        REM                                                                 FT129510
*       WHERE,                                                              FT129520
*          A = LOCATION OF FIRST KEY                                        FT129530
*          B = LOCATION OF LAST KEY                                         FT129540
        SPACE   3                                                           FT129550
 SORT1  SXA     EXSRT1,4        SAVE XR4 FOR RETURN                         FT129560
        AXT     2,1             SET UP FOR 2-ELEMENT TABLE ENTRIES          FT129570
        SXD     SRT36,1          X                                          FT129580
        AXT     0,1             X                                           FT129590
        SXA     SRT36-1,1       X                                           FT129600
        AXT     2,1             X                                           FT129610
        SXA     SRT36-2,1        X                                          FT129620
        AXT     -1,1            SET UP FOR KEY AS SECOND ELEMENT            FT129630
        SXA     SRT1-1,1         X                                          FT129640
        SXD     SRT11,1          X                                          FT129650
        AXT     3,1             SET UP FOR DESCENDING SORT                  FT129660
        CAL     1SRT1+3,1        X                                          FT129670
        SLW     SRT1-3,1         X                                          FT129680
        CAL     1SRT2+3,1        X                                          FT129690
        SLW     SRT10+4,1        X                                          FT129700
        TIX     *-4,1,1          X                                          FT129710
        REM                                                                 FT129720
        CAL     1,4             PREPARE TO GO TO S O R T                    FT129730
        SLW     *+4              X                                          FT129740
        CAL     2,4              X                                          FT129750
        SLW     *+3              X                                          FT129760
        CALL    SORT,*,*,*                                                  FT129770
        REM                                                                 FT129780
        AXT     1,1             AT END OF SORT, RESTORE S O R T             FT129790
        SXA     SRT1-1,1         X                                          FT129800
        SXD     SRT36,1          X                                          FT129810
        SXD     SRT11,1          X                                          FT129820
        SXA     SRT36-1,1        X                                          FT129830
        SXA     SRT36-2,1        X                                          FT129840
        AXT     3,1              X                                          FT129850
        CAL     1SRT2+3,1        X                                          FT129860
        SLW     SRT1-3,1         X                                          FT129870
        CAL     1SRT1+3,1        X                                          FT129880
        SLW     SRT10+4,1        X                                          FT129890
        TIX     *-4,1,1          X                                          FT129900
        REM                                                                 FT129910
EXSRT1  AXT     *,4             RESTORE XR4 FOR FINAL RETURN                FT129920
        TRA     3,4             EXIT S O R T 1                              FT129930
        SPACE   2                                                           FT129940
1SRT1   SXA     SRT17,1                                                     FT129950
        SXA     SRT20,1                                                     FT129960
        NOP                                                                 FT129970
1SRT2   SXA     SRT18,1                                                     FT129980
        SXA     SRT19,1                                                     FT129990
        SXA     SRT21,1
```

```
        TTL     U T I L I T Y  R O U T I N E S  -  B C D F I X          FT130000
        REM           BCD-TO-DECREMENT BINARY INTEGER --- BCDFIX        FT130010
        REM                                                             FT130020
        REM                                                             FT130030
        REM     THIS SUBROUTINE CONVERTS THE BCD INTEGER IN ACC (P,1-35) FT130040
        REM      TO A DECREMENT BINARY INTEGER.  THE BCD INTEGER IN THE FT130050
        REM      ACCUMULATOR IS LEFT-JUSTIFIED WITH TRAILING BLANKS.    FT130060
        REM      THE RESULTING BINARY INTEGER IS PLACED IN ACC (1-17)   FT130070
        REM      ON EXIT FROM BCDFIX.                                   FT130080
        REM                                                             FT130090
        REM           CALLING SEQUENCE,                                 FT130100
        REM                                                             FT130110
        REM                    .                                        FT130120
        REM                    .                                        FT130130
        REM                    .                                        FT130140
        REM               CAL  (ARGUMENT)                               FT130150
        REM               TSX  BCDFIX,4                                 FT130160
        REM                 (RETURN)                                    FT130170
        REM                    .                                        FT130180
        REM                    .                                        FT130190
        REM                    .                                        FT130200
        SPACE   2                                                       FT130210
BCDFIX  SXA     EXBDFX,4        SAVE XR4                                FT130220
        XCL                     PUT ARGUMENT IN MQ                      FT130230
        AXT     6,4             SET CHARACTER COUNTER                   FT130240
        STZ     BCDF3           INITIALIZE RESULT TO ZERO               FT130250
BCDF1   ZAC                     CLEAR ACCUMULATOR FOR NEXT DIGIT        FT130260
        TQP     BCDF2           IF NEXT CHAR. NUMERIC GO TO BCDF2       FT130270
        CLA     BCDF3           OTHERWISE GET RESULT                    FT130280
        ALS     18               AND SHIFT IT INTO ACC (1-17)           FT130290
EXBDFX  AXT     **,4             AND RESTORE XR4                        FT130300
        TRA     1,4              AND EXIT                               FT130310
        REM                                                             FT130320
        REM     ... THE FOLLOWING INSTRUCTIONS GENERATE THE BINARY      FT130330
        REM      RESULT DIGIT-BY-DIGIT.                                 FT130340
        REM                                                             FT130350
BCDF2   LGL     6               SAVE NEXT DIGIT IN BCDF4                FT130360
        STO     BCDF4           X                                       FT130370
        CLA     BCDF3           GET ACCUMULATED RESULT SO FAR           FT130380
        ALS     2               MULTIPLY ACCUMULATED RESULT BY 10       FT130390
        ADD     BCDF3           X                                       FT130400
        ALS     1               X                                       FT130410
        ADD     BCDF4           ADD THE NEW DIGIT                       FT130420
        STO     BCDF3           STORE AS NEW ACCUMULATED RESULT         FT130430
        TIX     BCDF1,4,1       GO GET NEXT CHARACTER                   FT130440
        TRA     BCDF1+2         RETURN TO CALLER                        FT130450
BCDF3                                                                   FT130460
BCDF4
```

```
         TTL     U T I L I T Y  R O U T I N E S  -  B I N B C D          FT130470
         REM     BINARY (INTEGER) TO BCD, LESS THAN 1,000,000 (10)      FT130480
         REM     ENTRY...        TSX   BINBCD,4  NUMBER IN 21-35 ACC    FT130490
         REM                     RETURN                                 FT130500
         REM     EXIT...         ANSWER IN AC WITH LEADING BLANKS       FT130510
         REM                                                            FT130520
BINBCD   LDQ     =0                                                     FT130530
         TZE     BX1                                                    FT130540
         LRS     6                                                      FT130550
         VDP     =100000,,6                                             FT130560
         VDP     =10000B29,,6                                           FT130570
         VDP     =1000B23,,6                                            FT130580
         VDP     =100B17,,6                                             FT130590
         VDP     =10B11,,6                                              FT130600
         VDP     =1B5,,6                                                FT130610
         SXA     BX,4            FIX TO READ LEADING BLANKS             FT130620
         AXT     0,4                                                    FT130630
         TNZ     *+3                                                    FT130640
         LGL     6                                                      FT130650
         TXI     *-2,4,6                                                FT130660
         ORA     =H      0                                              FT130670
         LGL     36,4                                                   FT130680
BX       AXT     **,4                                                   FT130690
         TRA     1,4                                                    FT130700
BX1      CAL     =H      0       IF ZERO                                FT130710
         TRA     1,4                                                    FT130720
```

```
         TTL      U T I L I T Y  R O U T I N E S  -  O C T B C D          FT130730
         REM                       OCTAL TO BCD CONVERSION               FT130740
         REM                                                             FT130750
         REM                                                             FT130760
OCTBCD   SXA      OCTX,4           OCTAL TO BCD.                        FT130770
         AXT      5,4              ANSWER AHS TRAILING BLANK            FT130780
         LGR      15                                                    FT130790
         ZAC                                                            FT130800
         ALS      3                                                     FT130810
         LGL      3                                                     FT130820
         TIX      *-2,4,1                                               FT130830
         TZE      OCTY                                                  FT130840
         LGR      36                                                    FT130850
         AXT      0,4                                                   FT130860
         TNZ      *+3                                                   FT130870
         LGL      6                                                     FT130880
         TXI      *-2,4,6                                               FT130890
         ORA      =H      0                                             FT130900
         LGL      36,4                                                  FT130910
         ALS      6                                                     FT130920
         ORA      =H00000                                               FT130930
OCTX     AXT      **,4             RESTORE INDEX REGISTER               FT130940
         TRA      1,4              AND EXIT                             FT130950
         REM                                                            FT130960
  OCTY   CAL      =H      0                                             FT130970
         TRA      OCTX
```

```
         TTL     U T I L I T Y  R O U T I N E S  -  C I N S T R          FT130980
         REM                        - CINSTR -                          FT130990
         REM     ROUTINE TO CONVERT 12 DIGIT OCTAL NUMBER TO BCD         FT131000
         REM     NO BLANKS ANYWHERE                                      FT131010
         REM     ENTRY...        LDQ   OCTAL.NUMBER                      FT131020
         REM                     TSX   CINSTR,4                          FT131030
         REM                     RETURN                                  FT131040
         REM     ANSWER IN AC,MQ IN BCD   ( LOGICAL)                     FT131050
CINSTR   SXA     CINX,1                                                  FT131060
         SXA     CINX+1,2                                                FT131070
         AXT     2,1                                                     FT131080
         AXT     6,2                                                     FT131090
         ALS     3                                                       FT131100
         LGL     3                                                       FT131110
         TIX     *-2,2,1                                                 FT131120
         SLW     CIN1+2,1                                                FT131130
         TIX     *-5,1,1                                                 FT131140
         XCL                                                             FT131150
         CAL     CIN1                                                    FT131160
CINX     AXT     **,1                                                    FT131170
         AXT     **,2                                                    FT131180
         TRA     1,4                                                     FT131190
CIN1     BSS     2                                                       FT131200
```

```
        TTL     UTILITY ROUTINES - STASH                    FT131210
STASH   SXA     XSTASH,1        SAVE XR'S                   FT131220
        SXA     XSTASH+1,2      X                           FT131230
        SXA     XSTASH+2,4      X                           FT131240
        AXT     *+3,1           SET-UP 2STASH FOR RETURN    FT131250
        SXA     EXSTSH,1        X                           FT131260
        TRA     2STASH          GO STASH THIS CARD          FT131270
        SPACE   3                                           FT131280
        AXT     GETFLG,1        RESET 2STASH                FT131290
        SXA     EXSTSH,1        X                           FT131300
XSTASH  AXT     **,1            RESTORE XR'S                FT131310
        AXT     **,2            X                           FT131320
        AXT     **,4            X                           FT131330
        TRA     1,4             RETURN                      FT131340
```

```
        TTL     E R R O R  P R O C E S S O R  -  E R R O R                FT131350
*       CONTROL IS GIVEN TO THIS ROUTINE ON EXECUTION OF ANY E R R O R   FT131360
*       MACRO.  THIS ROUTINE WILL OUTPUT THE INDICATED DIAGNOSTIC ALONG  FT131370
*       WITH THE LOCATION OF THE E R R O R MACRO IN QUESTION.  ON THE    FT131380
*       FIRST ENTRY TO THIS ROUTINE AN ON-LINE MESSAGE IS PRINTED        FT131390
*       INDICATING THAT A SOURCE ERROR WAS ENCOUNTERED DURING            FT131400
*       COMPILATION.                                                     FT131410
        REM                                                              FT131420
*       STANDARD COMMUNICATION IS VIA THE ERROR MACRO...                 FT131430
*               ERROR  XX,YY          WHICH IS EQUIVALENT TO             FT131440
*               STR    FORM'XX,,YY                                       FT131450
*       WHERE FORM'XX IDENTIFIES THE PERTINENT ERROR MESSAGE TO          FT131460
*       BE OUTPUT, AND YY GIVES THE POINT OF RETURN AFTER THE            FT131470
*       ERROR MESSAGE HAS BEEN ISSUED.                                   FT131480
        REM                                                              FT131490
*       THE ERROR MESSAGES HANDLED BY THIS ROUTINE MUST NOT EXCEED       FT131500
*       19 WORDS IN LENGTH.  THE WORD PRECEEDING THE MESSAGE             FT131510
*       (I.E. THE WORD SPECIFIED BY THE ADDRESS.FIELD-1 OF THE STR)      FT131520
*       MUST HAVE THE NUMBER OF WORDS IN THE ERROR MESSAGE (ADDRESS      FT131530
*       INTEGER).                                                        FT131540
        REM                                                              FT131550
*       THE ENTRY INTO THE ERROR ROUTINE (CELL 2) IS SET DURING THE      FT131560
*       PRE-INITIALIZATION SECTION OF THE COMPILER.                      FT131570
        SPACE   3                                                        FT131580
ERROR   TRA     *+1             STR ENTRY.  SET DURING PRE-INITIALIZATION FT131590
        SXA     ERR5,4                                                   FT131600
        CAL     0                 GET TRAPPED LOCATION                   FT131610
        SUB     =1                                                       FT131620
        STA     *+1                                                      FT131630
ERR2    CAL     **              GET INFORMATION WORD                     FT131640
        PDX     ,4                                                       FT131650
        TXH     *+2,4,0         GO TO MACHINE ERROR IF NO RETURN ADDRESS FT131660
        AXT     LOAD4,4                                                  FT131670
        SXA     EREX,4                                                   FT131680
        SUB     =1                                                       FT131690
        STA     ERR7                                                     FT131700
        STA     *+1                                                      FT131710
        LXA     **,4            GET COUNT                                FT131720
        PXA     ,4                                                       FT131730
        TXI     *+1,4,3                                                  FT131740
        SXD     ERR4,4                                                   FT131750
        PAC     ,4                                                       FT131760
        SXD     ERR8,4                                                   FT131770
        AXC     1,4                                                      FT131780
ERR7    LDQ     **,4                                                     FT131790
        STQ     HISS+2,4                                                 FT131800
ERR8    TXL     *+2,4,**                                                 FT131810
        TXI     *-3,4,-1                                                 FT131820
        REM                                                              FT131830
        LXA     ERR2,4                                                   FT131840
        PXA     ,4                                                       FT131850
        TSX     OCTBCD,4                                                 FT131860
        SLW     HISS+1                                                   FT131870
        TSX     LIST,4                                                   FT131880
ERR4    BCI     HISS,,**                                                 FT131890
        STL     ERHERE          FLAG FOR ERROR IN CURRENT STATEMENT      FT131900
ERR1    ZET     NOCODE          HAS 'NO EXECUTION' BEEN NOTED YET.       FT131910
        TRA     ERR5              YES                                    FT131920
        CAL     SBNAME          GET NAME OF ROUTINE                      FT131930
        TNZ     *+2             SKIP IF IT IS A SUBPROGRAM               FT131940
```

```
        CAL     =HMAIN          GET 'MAIN' LABEL                                FT131950
        SLW     SCER+3          PUT IN ERROR MESSAGE                            FT131960
        LDI     PRGBOX               TURN ON 'ERROR IN PROCESSOR' BIT           FT131970
        SIR     4                                                               FT131980
        STI     PRGBOX                                                          FT131990
IOS54   EQU     *                                                               FT132000
        PRINT   SCER,6                                                          FT132010
        STL     IOEX                                                            FT132020
        BRA     PRINT,,*+2                                                      FT132030
        MZE     SCER,,6                                                         FT132040
END54   REM                                                                     FT132050
IOS55   EQU     *                                                               FT132060
        PRINT   STDMP2,1                                                        FT132070
        STL     IOEX                                                            FT132080
        BRA     PRINT,,*+2                                                      FT132090
        MZE     STDMP2,,1                                                       FT132100
END55   REM                                                                     FT132110
        REM                                                                     FT132120
        STL     NOCODE          SET 'NO EXECUTION' FLAG ON                      FT132130
ERR5    AXT     **,4                                                            FT132140
EREX    TRA     **                                                              FT132150
        REM                                                                     FT132160
LOAD4   EQU     *                                                               FT132170
IOS56   EQU     *                                                               FT132180
        PRINT   QQ,5                                                            FT132190
        STL     IOEX                                                            FT132200
        BRA     PRINT,,*+2                                                      FT132210
        MZE     QQ,,5                                                           FT132220
END56   REM                                                                     FT132230
IOS57   EQU     *                                                               FT132240
        PRINT   EJECT,1                                                         FT132250
        STL     IOEX                                                            FT132260
        BRA     PRINT,,*+2                                                      FT132270
        MZE     EJECT,,1                                                        FT132280
END57   REM                                                                     FT132290
        HTR     *                                                               FT132300
        REM                                                                     FT132310
EJECT   BCI     1,1                                                             FT132320
HISS    BCI     3, ****         *****                                           FT132330
        BSS     19                                                              FT132340
        REM                                                                     FT132350
```

```
         EJECT                                                          FT132360
*        ERROR TRANSFER CELLS                                           FT132370
*                                                                       FT132380
ERROR1   ERROR   1,SKEND                                                FT132390
ERROR3   ERROR   3,SKEND                                                FT132400
ERROR5   ERROR   5,SKEND                                                FT132410
ERROR7   ERROR   7,SKEND                                                FT132420
ERROR8   ERROR   8,SKEND                                                FT132430
ERRO11   ERROR   11,SKEND                                               FT132440
ERRO15   ERROR   15,SKEND                                               FT132450
ERRO17   ERROR   17,SKEND                                               FT132460
ERRO18   ERROR   18,SKEND                                               FT132470
ERRO71   ERROR   71,SKEND                                               FT132480
ERRO79   ERROR   79,SKEND                                               FT132490
```

```
        TTL     E R R O R  P R O C E S S O R  -  E R R O R Y                 FT132500
*       THIS ROUTINE IS ENTERED WHENEVER AN E R R O R Y MACRO IS             FT132510
*       EXECUTED.  THE ONLY MACROS OF THIS TYPE RESIDE IN S Y P A S S.      FT132520
*       THE MACRO E R R O R Y CAUSES THE FOLLOWING INSTRUCTION TO BE        FT132530
*       GENERATED,                                                          FT132540
        REM                                                                 FT132550
        REM                     TSX   ERRORY,4,FLAG                         FT132560
        REM                                                                 FT132570
*       THIS ROUTINE EXAMINES F L A G TO DETERMINE THE SPECIFIC ERROR       FT132580
*       ENCOUNTERED ACCORDING TO THE FOLLOWING,                             FT132590
        REM                                                                 FT132600
        REM     FLAG = 0                UNDEFINED STATEMENT LABEL           FT132610
        REM     FLAG = 1                UNDEFINED STATEMENT STRING          FT132620
        REM     FLAG = 2                UNDEFINED STATEMENT VARIABLE        FT132630
        REM     FLAG = 3                STATEMENT WITH NO PATH OF FLOW      FT132640
        REM     FLAG = GREATER THAN 3   INVALID E Q U I V ENTRY             FT132650
        REM                                                                 FT132660
*       FOR FLAG 0 THRU 3 THE OFFENDING SYMBOL IS DUMPED INTO A POOL        FT132670
*       OF OTHER SIMILARLY OFFENDING SYMBOLS (UP TO A MAXIMUM OF 24 OF      FT132680
*       EACH).  THESE SYMBOLS ARE LATER DUMPED ONTO M L S T A P IN          FT132690
*       ROUTINE O U T U N D.                                                FT132700
        REM                                                                 FT132710
*       FOR FLAGS GREATER THAN 3 THE DIAGNOSTIC,                            FT132720
        REM                                                                 FT132730
        REM     XXXXXX HAS AN INVALID EQUIV MODE                            FT132740
        REM                                                                 FT132750
*          IS OUTPUT (XXXXXX = OFFENDING SYMBOL).                           FT132760
        REM                                                                 FT132770
*       IN ANY CASE, CONTROL IS RETURNED TO S Y P A S S FOR CONTINUED       FT132780
*       S Y M T A B / E Q U I V TESTING.                                    FT132790
        SPACE   3                                                           FT132800
ERRORY  RIL     700000          SET PREFIX TO 4                             FT132810
        SIL     400000          X                                           FT132820
        STI     EQUIV,1         X                                           FT132830
        LDQ     SYMTAB,1                                                    FT132840
        CAL     0,4                                                         FT132850
        ANA     =077000000                                                  FT132860
        PDX     ,2                                                          FT132870
        TXH     ERRY1,2,3                                                   FT132880
        CLA     TBUND,2                                                     FT132890
        PDX     ,4                                                          FT132900
        TXH     SPASS1,4,23                                                 FT132910
        STQ*    TBUND,2                                                     FT132920
        TXI     *+1,4,1                                                     FT132930
        PXD     ,4                                                          FT132940
        STD     TBUND,2                                                     FT132950
        TRA     SPASS1                                                      FT132960
        REM                                                                 FT132970
ERRY1   STQ     FORM39                                                      FT132980
        ERROR   39,SPASS1                                                   FT132990
        REM                                                                 FT133000
                NOPTH,4,..                                                  FT133010
                UNVAR,4,..                                                  FT133020
                UNSTR,4,..                                                  FT133030
TBUND           UNLBL,4,..                                                  FT133040
```

```
         TTL    D I A G N O S T I C  M E S S A G E S                    FT133050
*        THESE MESSAGES ARE OUTPUT ON M L S T A P BY THE EXECUTION OF AN FT133060
*        E R R O R MACRO.  TO OUTPUT THE MESSAGE FORM(XX), THE FOLLOWING FT133070
*        MACRO IS EXECUTED,                                              FT133080
         REM                                                             FT133090
         REM         ERROR   (XX),RETURN                                 FT133100
         SPACE   2                                                       FT133110
*        AFTER THE MESSAGE IS OUT, CONTROL IS GIVEN TO 'RETURN' IF IT IS FT133120
*        NON-ZERO.  OTHERWISE A CATASTROPHIC MESSAGE IS PRINTED ON-LINE  FT133130
*        AND THE COMPILER STOPS.                                         FT133140
         SPACE   3                                                       FT133150
*                                                                        FT133160
                 4                                                       FT133170
FORM1    BCI     4,INVALID STATEMENT NUMBER                              FT133180
                 4                                                       FT133190
FORM2    BCI     4,COMPILER EXPECTS COMMA                                FT133200
                 6                                                       FT133210
FORM3    BCI     6,COMPILER EXPECTS END OF STATEMENT                     FT133220
                 5                                                       FT133230
FORM4    BCI     5,COMPILER EXPECTS OCTAL NUMBER                         FT133240
                 3                                                       FT133250
FORM5    BCI     3,IMPROPER DIGIT                                        FT133260
                 6                                                       FT133270
FORM6    BCI     6,STATEMENT MAY TRANSFER TO ITSELF                      FT133280
                 5                                                       FT133290
FORM7    BCI     5,INVALID FORM OF STATEMENT                             FT133300
                 4                                                       FT133310
FORM8    BCI     4,INVALID VARIABLE NAME                                 FT133320
                 4                                                       FT133330
FORM9    BCI     4,COMPILER EXPECTS PAREN                                FT133340
                 5                                                       FT133350
FORM10   BCI     5,SUBPROGRAM MAY NOT CALL ITSELF                        FT133360
                 7                                                       FT133370
FORM11   BCI     7,STATEMENT STARTS WITH INVALID CHARACTER                FT133380
                 7                                                       FT133390
FORM12   BCI     7,COMPILER EXPECTS FIXED POINT VARIABLE                 FT133400
                 5                                                       FT133410
FORM13   BCI     5,MORE THAN FIVE OCTAL DIGITS                           FT133420
                 7                                                       FT133430
FORM14   BCI     7,VARIABLE FORMAT NAME NOT DECLARED AS ARRAY             FT133440
                 3                                                       FT133450
FORM15   BCI     3,SYMBOL TOO LONG                                       FT133460
                 12                                                      FT133470
FORM16   BCI     6,ARITHMETIC STATEMENT FUNCTION NAME                    FT133480
         BCI     1,XXXXXX      *****THIS CELL FILLED IN WITH NAME        FT133490
         BCI     5, APPEARS WITHOUT TERMINAL F                           FT133500
                 3                                                       FT133510
FORM17   BCI     3,VARIABLE SUBSCRIPT                                    FT133520
                 5                                                       FT133530
FORM18   BCI     5,MORE THAN THREE DIMENSIONS                            FT133540
                 15                                                      FT133550
FORM19   BCI     6,DIMENSION DECLARATION OF THE ARRAY                    FT133560
         BCI     1,XXXXXX      *****THIS CELL IS FILLED IN FOR EACH ERROR FT133570
         BCI     8, MUST PRECEDE APPEARANCE IN EXECUTABLE STATEMENT      FT133580
                 5                                                       FT133590
FORM20   BCI     5,ARRAY ILLEGALLY ENDS IN -F                            FT133600
                 10                                                      FT133610
FORM21   BCI     1,XXXXXX      *****THIS CELL FILLED IN FOR EACH ERROR   FT133620
         BCI     9, APPEARS MORE THAN ONCE IN DIMENSION DECLARATIONS      FT133630
                 3                                                       FT133640
```

```
FORM22 BCI      3,ARRAY TOO BIG                                                 FT133650
               9                                                                FT133660
FORM23 BCI      3,THE FUNCTION NAME                                             FT133670
       BCI      1,XXXXXX       *****THIS CELL FILLED IN WITH NAME               FT133680
       BCI      5, APPEARS WITHOUT AN ARGUMENT                                  FT133690
               10                                                               FT133700
FORM24 BCI ,FORMAT STATEMENT NUMBER REFERS TO EXECUTABLE STATEMENT              FT133710
               11                                                               FT133720
FORM25 BCI      4,THE SUBPROGRAM ARGUMENT                                       FT133730
       BCI      1,XXXXXX       *****THIS CELL FILLED IN FOR EACH ERROR          FT133740
       BCI      6, APPEARS IN EQUIVALENCE STATEMENT                             FT133750
               11                                                               FT133760
FORM26 BCI ,INCONSISTENCY IN EQUIVALENCE DECLARATIONS INVOLVING SYMBOL          FT133770
       BCI      1,XXXXXX       *****THIS CELL FILLED IN FOR EACH ERROR          FT133780
               9                                                                FT133790
FORM27 BCI      1,XXXXXX       *****THIS CELL FILLED IN FOR EACH ERROR          FT133800
       BCI      8, APPEARS MORE THAN ONCE IN COMMON DECLARATIONS                 FT133810
               15                                                               FT133820
FORM28 BCI      2,THE SYMBOL                                                    FT133830
       BCI      1,XXXXXX       *****THIS CELL FILLED IN FOR EACH ERROR          FT133840
       BCI , IN ASSIGN OR ASSIGNED GO TO STATEMENT APPEARS IN DIMENSION         FT133850
       BCI      2,STATEMENT                                                     FT133860
               14                                                               FT133870
FORM29 BCI ,STATEMENT NUMBERS BELOW ARE NEEDED IN ASSIGNED GO TO LISTS FFT133880
       BCI      3,OR THE VARIABLE                                               FT133890
       BCI      1,XXXXXX       *****THIS CELL IS FILLED IN FOR EACH ERROR       FT133900
               10                                                               FT133910
FORM30 BCI      3,ASSIGN VARIABLE                                               FT133920
       BCI      1,XXXXXX       *****THIS CELL IS FILLED IN FOR EACH ERROR       FT133930
       BCI      6, NOT IN ASSIGNED GO TO STATEMENT                              FT133940
               9                                                                FT133950
FORM31 BCI      4,ASSIGNED GO TO VARIABLE                                       FT133960
       BCI      1,XXXXXX       *****THIS CELL IS FILLED IN FOR EACH ERROR       FT133970
       BCI      4, NOT IN ASSIGN STATEMENT                                      FT133980
               3                                                                FT133990
FORM32 BCI      3,PROGRAM TOO LONG                                              FT134000
               8                                                                FT134010
FORM33 BCI      8,LAST EXECUTABLE STATEMENT MUST BE A TRANSFER                  FT134020
               12                                                               FT134030
FORM34 BCI ,FIXED POINT INTERNAL OR LIBRARY FUNCTION INVALID IN DP OR CAFT134040
       BCI      2, STATEMENT                                                    FT134050
               9                                                                FT134060
FORM35 BCI      9,WRONG COLUMN 1 MODE FOR ARITHMETIC STATEMENT FUNCTION         FT134070
               11                                                               FT134080
FORM36 BCI ,WRONG NUMBER OR MODE OF ARGUMENTS IN ARITHMETIC STATEMENT FUFT134090
       BCI      1,NCTION                                                        FT134100
               3                                                                FT134110
FORM37 BCI      3,MISSING END CARD                                              FT134120
               7                                                                FT134130
FORM38 BCI      7,NO SYMBOL TABLE PUNCHED -- POOL OVERFLOW                      FT134140
               6                                                                FT134150
FORM39 BCI      6,XXXXXX HAS AN INVALID EQUIV MODE$                             FT134160
FORM40                                                                          FT134170
FORM41                                                                          FT134180
               7                                                                FT134190
FORM42 BCI      6,INVALID FORM OF SUBSCRIPT FOR ARRAY                           FT134200
       BCI      1,XXXXXX       *****THIS CELL IS FILLED IN WITH NAME            FT134210
               4                                                                FT134220
FORM43 BCI      4,INVALID COMPLEX CONSTANT                                      FT134230
               3                                                                FT134240
```

```
FORM45 BCI      3,INVALID EQUAL SIGN                                          FT134250
                5                                                             FT134260
FORM46 BCI      5,INVALID INDEXING IN IO LIST                                 FT134270
                6                                                             FT134280
FORM47 BCI      6,INVALID LIST ITEM(S) IN IO LIST                             FT134290
                5                                                             FT134300
FORM48 BCI      5,INVALID PUNCTUATION IN IO LIST                              FT134310
                9                                                             FT134320
FORM49 BCI      9,SINGLE PRECISION ARRAY USED IN DP OR CA STATEMENT           FT134330
                7                                                             FT134340
FORM50 BCI      7,COMPILER EXPECTS COMMA OR RIGHT PAREN                       FT134350
                5                                                             FT134360
FORM51 BCI      5,INVALID FORTRAN FUNCTION NAME                               FT134370
                12                                                            FT134380
FORM52 BCI ,FUNCTION OR SUBROUTINE STATEMENT NOT FIRST STATEMENT IN THE       FT134390
       BCI      2,PROGRAM                                                     FT134400
                7                                                             FT134410
FORM53 BCI      7,RETURN STATEMENT INVALID IN MAIN PROGRAM                    FT134420
                9                                                             FT134430
FORM54 BCI      9,INCONSISTENT MODE DESIGNATIONS ON RETURN STATEMENTS         FT134440
                4                                                             FT134450
FORM55 BCI      4,RETURN STATEMENT MISSING                                    FT134460
                12                                                            FT134470
FORM56 BCI ,FUNCTION NAME DOES NOT APPEAR ON LEFT SIDE OF SUBSTITUTION SFT134480
       BCI      2,TATEMENT                                                    FT134490
                7                                                             FT134500
FORM57 BCI      7,NO PAREN FOLLOWS TERMINAL -F FUNCTION                       FT134510
                9                                                             FT134520
FORM58 BCI      9,STATEMENT NUMBER APPEARS ON MORE THAN ONE STATEMENT         FT134530
                8                                                             FT134540
FORM59 BCI      8,SUBPROGRAM ARGUMENT USED AS INDEX OF DO LOOP                FT134550
                4                                                             FT134560
FORM60 BCI      4,COMPILER EXPECTS = SIGN                                     FT134570
                6                                                             FT134580
FORM61 BCI      6,COMPILER EXPECTS STATEMENT NUMBER                           FT134590
                7                                                             FT134600
FORM62 BCI      7,TOO MANY NESTED DOS.  SIMPLIFY PROGRAM                      FT134610
                5                                                             FT134620
FORM63 BCI      5,INVALID USE OF FORMAT NUMBER                                FT134630
                11                                                            FT134640
FORM65 BCI ,NON-EXECUTABLE STATEMENT MAY NOT BE FIRST STATEMENT OF A DO       FT134650
       BCI      1,LOOP                                                        FT134660
                14                                                            FT134670
FORM66 BCI ,TERMINAL STATEMENT OF A DO MAY NOT BE NON-EXECUTABLE, A TRANFT134680
       BCI      4,SFER, OR ANOTHER DO                                         FT134690
                6                                                             FT134700
FORM67 BCI      6,NO PATH OF FLOW TO ABOVE STATEMENT                          FT134710
                7                                                             FT134720
FORM68 BCI      7,FORMAT STATEMENT HAS NO STATEMENT NUMBER                    FT134730
                9                                                             FT134740
FORM69 BCI      1,XXXXXX      *****THIS CELL FILLED IN FOR EACH ERROR         FT134750
       BCI      8, IS USED AS BOTH A VARIABLE AND SUBPROGRAM NAME             FT134760
                5                                                             FT134770
FORM70 BCI      5,STATEMENT NUMBER TOO LARGE                                  FT134780
                7                                                             FT134790
FORM71 BCI      7,ILLEGAL FORM OF FLOATING POINT CONSTANT                     FT134800
                6                                                             FT134810
FORM72 BCI      6,NUMBER EXCEEDS CAPACITY OF COMPUTER                         FT134820
                8                                                             FT134830
FORM73 BCI      8,TERMINUS OF THIS DO HAS ALREADY BEEN ENCOUNTERED            FT134840
```

```
                4                                                                FT134850
FORM74 BCI      4,CONSECUTIVE ** ILLEGAL                                         FT134860
                3                                                                FT134870
FORM75 BCI      3,INVALID COMMA                                                  FT134880
                8                                                                FT134890
FORM76 BCI      8,SUBPROGRAM NAME IS ARGUMENT OF LIBRARY FUNCTION                FT134900
                3                                                                FT134910
FORM77 BCI      3,ILLEGAL CHARACTER                                              FT134920
                9                                                                FT134930
FORM78 BCI      9,ARITHMETIC STATEMENT FUNCTION DEFINED MORE THAN ONCE           FT134940
                3                                                                FT134950
FORM79 BCI      3,MISPLACED PERIOD                                               FT134960
                5                                                                FT134970
FORM80 BCI      5,INVALID CHARACTER IN COLUMN 1                                  FT134980
                10                                                               FT134990
FORM81 BCI      ,COMPILER EXPECTS STATEMENT NUMBER, INTEGER OR VARIABLE           FT135000
                8                                                                FT135010
FORM82 BCI      8,0 IS NOT VALID AS STATEMENT NUMBER OR SUBSCRIPT                 FT135020
                7                                                                FT135030
FORM83 BCI      7,NO STATEMENT ASSOCIATED WITH ABOVE LABEL                       FT135040
                5                                                                FT135050
FORM84 BCI      5,LEFT SIDE OF STATEMENT INVALID                                 FT135060
                8                                                                FT135070
FORM85 BCI      8,FUNCTION NAME USED AS ARRAY OR SUBPROGRAM NAME                 FT135080
                8                                                                FT135090
FORM86 BCI      8,ABOVE DO NESTS WITHIN A DO HAVING SAME DO INDEX                FT135100
                9                                                                FT135110
FORM87 BCI      9,DO INDEX APPEARS WITHIN THE DO ON LEFT OF EQUAL SIGN           FT135120
                8                                                                FT135130
FORM88 BCI      8,BUILT-IN FUNCTION APPEARS ON LEFT OF EQUAL SIGN                FT135140
                11                                                               FT135150
FORM89 BCI      ,ARITHMETIC STATEMENT FUNCTIONS MUST PRECEDE EXECUTABLE STATE    FT135160
       BCI      1,MENTS                                                          FT135170
                9                                                                FT135180
FORM90 BCI      9,SUBSCRIPTS ILLEGAL IN ARITHMETIC STATEMENT FUNCTIONS           FT135190
                3                                                                FT135200
FORM91 BCI      3,UNMATCHED PAREN                                                FT135210
FORM92                                                                           FT135220
FORM93                                                                           FT135230
FORM94                                                                           FT135240
FORM95                                                                           FT135250
FORM96                                                                           FT135260
FORM97                                                                           FT135270
FORM98                                                                           FT135280
FORM99                                                                           FT135290
       REM                                                                       FT135300
                8                                                                FT135310
MSG100 BCI      8,COMPILER CAPACITY EXCEEDED.  SIMPLIFY PROGRAM.                 FT135320
                3                                                                FT135330
MCR    BCI      3,MACHINE ERROR ***                                              FT135340
       REM                                                                       FT135350
QQ     BCI      5,OTROUBLE.  TAKE CORE DUMP.                                     FT135360
SCER   BCI      3, SOURCE ERROR IN                                               FT135370
       BCI      1,XXXXXX                                                         FT135380
       BCI      2, PROGRAM.                                                      FT135390
```

```
      TTL    T A B L E S                                                FT135400
*                                                                       FT135410
      REM                                                               FT135420
      SPACE  9                                                          FT135430
*       *   *   *   *   *   *   *   *   *   *   *   *                   FT135440
      REM                                                               FT135450
*                 F A S T R A N  C O M P I L E R                        FT135460
      REM                                                               FT135470
*     R E S E A R C H  C O M P U T I N G  C E N T E R                   FT135480
      REM                                                               FT135490
*               I N D I A N A  U N I V E R S I T Y                      FT135500
      REM                                                               FT135510
*           B L O O M I N G T O N ,  I N D I A N A                      FT135520
      SPACE  3                                                          FT135530
*       *   *   *   *   *   *   *   *   *   *   *   *                   FT135540
      REM                                                               FT135550
*                       T A B L E S                                     FT135560
      REM                                                               FT135570
*       *   *   *   *   *   *   *   *   *   *   *   *                   FT135580
```

```
      TTL   T A B L E S                                                  FT135600
*     THE FOLLOWING ARE THE VARIOUS TABLES USED OR GENERATED BY FASTRANFT135610
*     DURING COMPILATION WITH THE EXCEPTION OF P O O L, S Y M T A B    FT135620
*     AND E Q U I V (THESE ARE DESCRIBED LATER).                       FT135630
```

```
         TTL    T A B L E S  -  O P E R A T O R  R A N K  T A B L E      FT135640
*        EACH ARITHMETIC OPERATOR IS ASSIGNED A RANK IN THIS TABLE.      FT135650
*        ALTHOUGH AT PRESENT EACH OPERATOR HAS A UNIQUE RANK, FACILITY IS FT135660
*        BUILT INTO THE TABLE FOR ASSIGNING ARBITRARY RANKS.  THE RANK   FT135670
*        TABLE IS USED BY THE COMPIL ROUTINE TO DETERMINE IF AN OPERATOR FT135680
*        MAY BE INSERTED INTO THE ARITHMETIC STACK (ASTACK), OR IF CODE  FT135690
*        MUST BE GENERATED FIRST.  THE OPERATORS ARE RANKED AS FOLLOWS,  FT135700
*        WITH THE LOWEST RANK (LEAST CODE-FORCING) OPERATOR FIRST...     FT135710
         REM                                                             FT135720
*           UNARY MINUS, EXPONENTIATION, DIVIDE, MULTIPLY, BINARY MINUS, FT135730
*           PLUS, COMMA, EQUALS, LEFT PAREN, RIGHT PAREN, ARITHMETIC     FT135740
*           STATEMENT FUNCTION EQUAL SIGN.                               FT135750
         REM                                                             FT135760
*        FOR BOOLEAN STATEMENTS, THE RANKS ARE...                        FT135770
         REM                                                             FT135780
*           AND (EQUIVALENT TO MULTIPLY), OR (EQUIVALENT TO PLUS),       FT135790
*           COMPLEMENT (EQUIVALENT TO UNARY MINUS).                      FT135800
         REM                                                             FT135810
*        EACH RANK KEY WORD CONTAINS, AMONG OTHER INFORMATION, THE       FT135820
*        RANK AND THE ADDRESS OF THE APPROPRIATE PROCESSOR IN THE        FT135830
*        COMPIL ROUTINE.  BIT 16 IS ON IN EACH RANK KEY WORD, INDICATING FT135840
*        TO THE COMPIL ROUTINE THAT THIS WORD IS AN OPERATOR.            FT135850
         SPACE  2                                                        FT135860
RANKTB   EQU    *                                                       FT135870
         RANK   NEG,NEG,UNEGKY,,MINUSP          -  UNARY MINUS          FT135880
         RANK   EXPNL,EXPNL,EXPKEY,,EXPONP      ** EXPONENTIAL          FT135890
         RANK   DIV,DIV,DIVKEY,,DIVIDP          /  DIVIDE               FT135900
         RANK   TIMES,TIMES,MPYKEY,,OTHROP      *  MULTIPLY             FT135910
         RANK   BNEG,BNEG,NEGKEY                -  BINARY MINUS         FT135920
         RANK   PLUS,PLUS,PLUSKY,,PLUSP         +  PLUS                 FT135930
         RANK   COMMA,COMMA,CMAKEY,,COMMAP      ,  COMMA                FT135940
         RANK   EQUALS,EQUALS,EQLKEY,,EQUALP    =  EQUALS               FT135950
         RANK   LPAREN,LPAREN,LPNKEY,,LPARNP    (  LEFT PAREN           FT135960
         RANK   RPAREN,RPAREN,RPNKEY,,RPARNP    )  RIGHT PAREN          FT135970
         RANK   ARSTFN,ARSTFN,ASFKEY            =  ARITH. STATE. FN. EQUALSFT135980
         REM                                                             FT135990
LRANKT   EQU    *-RANKTB        LENGTH OF RANK TABLE                    FT136000
         REM                                                             FT136010
         REM                                                             FT136020
         REM                  BOOLEAN OPERATOR EQUIVALENCES              FT136030
AND      EQU    TIMES                    *  BOOLEAN AND                 FT136040
OR       EQU    PLUS                     +  BOOLEAN OR                  FT136050
COMP     EQU    NEG                      -  BOOLEAN COMPLEMENT          FT136060
```

```
      TTL    T A B L E S - G E N E R A T O R  K E Y  T A B L E S      FT136070
*     THE GENERATOR KEY TABLES CONTAIN CODED DESCRIPTIONS OF ALL      FT136080
*     ALLOWED ARITHMETIC OPERATIONS.  FOR AN OPERATION OF THE FORM    FT136090
*     A.OPERATOR.B, EITHER OF THE ELEMENTS A OR B MAY CURRENTLY BE IN FT136100
*     THE ACCUMULATOR, IN THE MQ, OR IN STORAGE.                      FT136110
*     A TYPICAL KEY WORD IN THE TABLE MIGHT CONTAIN THE FOLLOWING     FT136120
*     INFORMATION (SEE THE KEY MACRO FOR THE BIT PATTERN)...          FT136130
      REM                                                             FT136140
*        MODE OF A-OPERAND, LOCATION (AC,MQ,VS) OF A-OPERAND,         FT136150
*        OPERATOR, MODE OF B-OPERAND, LOCATION OF B-OPERAND.          FT136160
      REM                                                             FT136170
*     IN ADDITION, THE KEY WORD CONTAINS THE ADDRESS OF THE           FT136180
*     APPROPRIATE CODE GENERATOR.  FOR UNARY OPERATIONS (LIKE -B),    FT136190
*     THE INFORMATION FOR THE A-OPERAND IS MISSING.                   FT136200
      REM                                                             FT136210
*     THERE ARE FOUR GENERATOR KEY TABLES, ONE FOR EACH PERTINENT     FT136220
*     COLUMN 1 MODE... FIXED/FLOATING POINT, BOOLEAN, DOUBLE          FT136230
*     PRECISION, AND COMPLEX ARITHMETIC.  EACH OF THESE TABLES CONTAINSFT136240
*     ENTRIES FOR ALL ALLOWABLE OPERATIONS, INCLUDING BUILT-IN        FT136250
*     FUNCTIONS AND CERTAIN OTHER CONVENIENT OPERATIONS INTERNAL TO    FT136260
*     THE COMPILER.  THE FOUR TABLES ARE SORTED IN THE PRE-INITIALIZA- FT136270
*     TION PHASE OF THE COMPILER, AND REMAIN SORTED THROUGHOUT THE     FT136280
*     COMPILER'S STAY IN CORE.                                        FT136290
      REM                                                             FT136300
*     THE COMPIL ROUTINE ESTABLISHES THE CORRECT TABLE TO USE ON      FT136310
*     THE BASIS OF THE CURRENT COLUMN 1 MODE.                         FT136320
*     THE GEN ROUTINE CREATES A KEY WORD (MINUS THE ADDRESS OF THE    FT136330
*     CODE GENERATOR) FROM THE INFORMATION IT FINDS IN ASTACK, AND    FT136340
*     DOES A MASKED BINARY SEARCH ON THE APPROPRIATE TABLE.           FT136350
*     IF AN EQUALITY IS FOUND WITH THE GENERATED KEY, THEN CODE       FT136360
*     IS PRODUCED BY THE ROUTINE DESIGNATED IN THE ADDRESS OF THE     FT136370
*     LOCATED TABLE WORD.  MOST OF THE SYNTACTICAL ERRORS IN          FT136380
*     ARITHMETIC STATEMENTS ARE DISCOVERED BY NOT FINDING A KEY IN    FT136390
*     THE TABLE.                                                      FT136400
```

```
        EJECT                                                          FT136410
KEYTAB  EQU     *                                                      FT136420
KEYTBO  EQU     *               REAL AND INTEGER KEYS                  FT136430
        REM                                                            FT136440
        KEY1    INT,REAL,INTO                                          FT136450
        KEY2    REAL,SIGN,REAL,SIGNO                                   FT136460
        KEY2    INTEGR,XSIGN,INTEGR,XSIGN                              FT136470
        KEY2    REAL,DIM,REAL,DIMO                                     FT136480
        KEY2    INTEGR,XDIM,INTEGR,XDIMO                               FT136490
        KEY2    INTEGR,MAXO,INTEGR,MXMIN                               FT136500
        KEY2    REAL,MAX1,REAL,MXMIN                                   FT136510
        KEY2    INTEGR,XMAXO,INTEGR,MXMIN                              FT136520
        KEY2    REAL,XMAX1,REAL,MXMIN                                  FT136530
        KEY2    INTEGR,MINO,INTEGR,MXMIN                               FT136540
        KEY2    REAL,MIN1,REAL,MXMIN                                   FT136550
        KEY2    INTEGR,XMINO,INTEGR,MXMIN                              FT136560
        KEY2    REAL,XMIN1,REAL,MXMIN                                  FT136570
        KEY2    REAL,MOD,REAL,MODO                                     FT136580
        KEY2    INTEGR,XMOD,INTEGR,XMODO                               FT136590
        KEY     INTEGR,VS,EQUALS,INTEGR,VS,IEQI01                      FT136600
        KEY     INTEGR,VS,EQUALS,INTEGR,AC,IEQI02                      FT136610
        KEY     INTEGR,VS,EQUALS,INTEGR,MQ,IEQI03                      FT136620
        KEY     INTEGR,VS,EQUALS,REAL,VS,IEQR01                        FT136630
        KEY     INTEGR,VS,EQUALS,REAL,AC,IEQR02                        FT136640
        KEY     INTEGR,VS,EQUALS,REAL,MQ,IEQR03                        FT136650
        KEY     REAL,VS,EQUALS,INTEGR,VS,REQI01                        FT136660
        KEY     REAL,VS,EQUALS,INTEGR,AC,REQI02                        FT136670
        KEY     REAL,VS,EQUALS,INTEGR,MQ,REQI03                        FT136680
        KEY     REAL,VS,EQUALS,REAL,VS,REQR01                          FT136690
        KEY     REAL,VS,EQUALS,REAL,AC,REQR02                          FT136700
        KEY     REAL,VS,EQUALS,REAL,MQ,REQR03                          FT136710
        KEY1    ABS,REAL,ABSO                                          FT136720
        KEY1    XABS,INTEGR,XABSO                                      FT136730
        KEY1    XINT,REAL,XINTO                                        FT136740
        KEY1    FLOAT,INTEGR,FLOAT                                     FT136750
        KEY1    XFIX,REAL,XINTO                                        FT136760
        KEY1    NEG,INTEGR,INEGO                                       FT136770
        KEY1    NEG,REAL,RNEGO                                         FT136780
        KEY2    INTEGR,PLUS,INTEGR,IADDO                               FT136790
        KEY2    INTEGR,BNEG,INTEGR,ISUBO                               FT136800
        KEY2    INTEGR,TIMES,INTEGR,IMPYO                              FT136810
        KEY2    INTEGR,DIV,INTEGR,IDIVO                                FT136820
        KEY2    INTEGR,EXPNL,INTEGR,IEXPO                              FT136830
        KEY2    REAL,PLUS,REAL,RADDO                                   FT136840
        KEY2    REAL,BNEG,REAL,RSUBO                                   FT136850
        KEY2    REAL,TIMES,REAL,RMPYO                                  FT136860
        KEY2    REAL,DIV,REAL,RDIVO                                    FT136870
        KEY2    REAL,EXPNL,REAL,REXPO                                  FT136880
        KEY2    REAL,EXPNL,INTEGR,REXPI                                FT136890
        KEY     ,,AUXIL,REAL,VS,AUXIL1                                 FT136900
        KEY     ,,AUXIL,REAL,AC,AUXIL6                                 FT136910
        KEY     ,,AUXIL,REAL,MQ,AUXIL2                                 FT136920
        KEY     ,,AUXIL,INTEGR,VS,AUXIL1                               FT136930
        KEY     ,,AUXIL,INTEGR,AC,AUXIL6                               FT136940
        KEY     ,,AUXIL,INTEGR,MQ,AUXIL2                               FT136950
        KEY     ,,XLOC,INTEGR,VS,XLOC01                                FT136960
        KEY     ,,XLOC,REAL,VS,XLOC01                                  FT136970
        KEY     REAL,,ARSTFN,REAL,VS,ASFRR1                            FT136980
        KEY     REAL,,ARSTFN,REAL,AC,ASFRR2                            FT136990
        KEY     REAL,,ARSTFN,REAL,MQ,ASFRR3                            FT137000
```

```
        KEY     REAL,,ARSTFN,INTEGR,VS,ASFRI1                    FT137010
        KEY     REAL,,ARSTFN,INTEGR,AC,ASFRI2                    FT137020
        KEY     REAL,,ARSTFN,INTEGR,MQ,ASFRI3                    FT137030
        KEY     INTEGR,,ARSTFN,REAL,VS,ASFIR1                    FT137040
        KEY     INTEGR,,ARSTFN,REAL,AC,ASFIR2                    FT137050
        KEY     INTEGR,,ARSTFN,REAL,MQ,ASFIR3                    FT137060
        KEY     INTEGR,,ARSTFN,INTEGR,VS,ASFII1                  FT137070
        KEY     INTEGR,,ARSTFN,INTEGR,AC,ASFII2                  FT137080
        KEY     INTEGR,,ARSTFN,INTEGR,MQ,ASFII3                  FT137090
        REM                                                      FT137100
LKYTBO  EQU     *-KEYTBO        LENGTH OF REAL AND INTEGER KEYS  FT137110
```

```
        EJECT                                                          FT137120
KEYTB1  EQU     *               BOOLEAN KEYS                           FT137130
        REM                                                            FT137140
        KEY4    ABS,REAL,ABSO                                          FT137150
        KEY4    FLOAT,INTEGR,FLOAT                                     FT137160
        KEY4    INT,REAL,INTO                                          FT137170
        KEY3    REAL,MOD,REAL,MODO                                     FT137180
        KEY3    REAL,SIGN,REAL,SIGNO                                   FT137190
        KEY3    REAL,DIM,REAL,DIMO                                     FT137200
        KEY3    INTEGR,MAXO,INTEGR,MXMIN                               FT137210
        KEY3    REAL,MAX1,REAL,MXMIN                                   FT137220
        KEY3    INTEGR,MINO,INTEGR,MXMIN                               FT137230
        KEY3    REAL,MIN1,REAL,MXMIN                                   FT137240
        KEY3    REAL,AND,REAL,BANDO                                    FT137250
        KEY3    REAL,OR,REAL,BORO                                      FT137260
        KEY     0,0,COMP,REAL,VS,BCOM01                                FT137270
        KEY     0,0,COMP,REAL,AC,BCOM02                                FT137280
        KEY     REAL,VS,EQUALS,REAL,VS,BEQB01                          FT137290
        KEY     REAL,VS,EQUALS,REAL,AC,BEQB02                          FT137300
        KEY     ,,AUXIL,REAL,VS,AUXIL3                                 FT137310
        KEY     ,,AUXIL,REAL,AC,AUXIL7                                 FT137320
        KEY     ,,AUXIL,INTEGR,AC,AUXIL7                               FT137330
        KEY     REAL,,ARSTFN,REAL,VS,ASFRR4                            FT137340
        KEY     REAL,,ARSTFN,REAL,AC,ASFRR5                            FT137350
        REM                                                            FT137360
LKYTB1  EQU     *-KEYTB1        LENGTH OF BOOLEAN KEYS                 FT137370
```

```
         EJECT                                                          FT137380
KEYTB2   EQU      *              DOUBLE PRECISION KEYS                  FT137390
         REM                                                           FT137400
         KEY1     4FLOAT,INTEGR,DFLOT                                  FT137410
         KEY1     4ABS,REAL,DABSO                                      FT137420
         KEY1     4FIX,REAL,DFIXO                                      FT137430
         KEY2     REAL,4SIGN,REAL,DSIGN                                FT137440
         KEY      REAL,VS,EQUALS,REAL,VS,DEQDO1                        FT137450
         KEY      REAL,VS,EQUALS,REAL,AC,DEQDO2                        FT137460
         KEY      REAL,VS,EQUALS,REAL,MQ,DEQDO3                        FT137470
         KEY      REAL,VS,EQUALS,INTEGR,VS,DEQIO1                      FT137480
         KEY      REAL,VS,EQUALS,INTEGR,AC,DEQIO2                      FT137490
         KEY      REAL,VS,EQUALS,INTEGR,MQ,DEQIO3                      FT137500
         KEY      INTEGR,VS,EQUALS,REAL,VS,IEQDO1                      FT137510
         KEY      INTEGR,VS,EQUALS,REAL,AC,IEQDO2                      FT137520
         KEY      INTEGR,VS,EQUALS,REAL,MQ,IEQDO3                      FT137530
         KEY      INTEGR,VS,EQUALS,INTEGR,VS,IEQIO1                    FT137540
         KEY      INTEGR,VS,EQUALS,INTEGR,AC,IEQIO2                    FT137550
         KEY      INTEGR,VS,EQUALS,INTEGR,MQ,IEQIO3                    FT137560
         KEY2     REAL,PLUS,REAL,DADDO                                 FT137570
         KEY1     NEG,REAL,DNEGO                                       FT137580
         KEY2     REAL,BNEG,REAL,DSUBO                                 FT137590
         KEY2     REAL,TIMES,REAL,DMPYO                                FT137600
         KEY2     REAL,DIV,REAL,DDIVO                                  FT137610
         KEY2     REAL,EXPNL,INTEGR,DEXPI                              FT137620
         KEY2     REAL,EXPNL,REAL,DEXPR                                FT137630
         KEY1     NEG,INTEGR,INEGO                                     FT137640
         KEY2     INTEGR,PLUS,INTEGR,IADDO                             FT137650
         KEY2     INTEGR,BNEG,INTEGR,ISUBO                             FT137660
         KEY2     INTEGR,TIMES,INTEGR,IMPYO                            FT137670
         KEY2     INTEGR,DIV,INTEGR,IDIVO                              FT137680
         KEY2     INTEGR,EXPNL,INTEGR,IEXPO                            FT137690
         KEY      ,,AUXIL,REAL,VS,AUXIL1                               FT137700
         KEY      ,,AUXIL,REAL,AC,AUXIL4                               FT137710
         KEY      ,,AUXIL,REAL,MQ,AUXIL5                               FT137720
         KEY      ,,AUXIL,INTEGR,VS,AUXIL1                             FT137730
         KEY      ,,AUXIL,INTEGR,AC,AUXIL6                             FT137740
         KEY      ,,AUXIL,INTEGR,MQ,AUXIL2                             FT137750
         KEY      REAL,,ARSTFN,REAL,VS,ASFDD1                          FT137760
         KEY      REAL,,ARSTFN,REAL,AC,ASFDD2                          FT137770
         KEY      REAL,,ARSTFN,REAL,MQ,ASFDD3                          FT137780
         KEY      REAL,,ARSTFN,INTEGR,VS,ASFDI1                        FT137790
         KEY      REAL,,ARSTFN,INTEGR,AC,ASFDI2                        FT137800
         KEY      REAL,,ARSTFN,INTEGR,MQ,ASFDI3                        FT137810
         REM                                                           FT137820
LKYTB2   EQU      *-KEYTB2       LENGTH OF DOUBLE PRECISION KEYS       FT137830
```

```
         EJECT                                                          FT137840
KEYTB3   EQU   *              COMPLEX ARITHMETIC KEYS                   FT137850
         REM                                                            FT137860
         KEY1  1FLOAT,INTEGR,DFLOT                                      FT137870
         KEY1  1FIX,REAL,DFIXO                                          FT137880
         KEY2  REAL,1SIGN,REAL,DSIGN                                    FT137890
         KEY   REAL,VS,EQUALS,REAL,VS,DEQDO1                            FT137900
         KEY   REAL,VS,EQUALS,REAL,AC,DEQDO2                            FT137910
         KEY   REAL,VS,EQUALS,REAL,MQ,DEQDO3                            FT137920
         KEY   REAL,VS,EQUALS,INTEGR,VS,DEQIO1                          FT137930
         KEY   REAL,VS,EQUALS,INTEGR,AC,DEQIO2                          FT137940
         KEY   REAL,VS,EQUALS,INTEGR,MQ,DEQIO3                          FT137950
         KEY   INTEGR,VS,EQUALS,REAL,VS,IEQDO1                          FT137960
         KEY   INTEGR,VS,EQUALS,REAL,AC,IEQDO2                          FT137970
         KEY   INTEGR,VS,EQUALS,REAL,MQ,IEQDO3                          FT137980
         KEY   INTEGR,VS,EQUALS,INTEGR,VS,IEQIO1                        FT137990
         KEY   INTEGR,VS,EQUALS,INTEGR,AC,IEQIO2                        FT138000
         KEY   INTEGR,VS,EQUALS,INTEGR,MQ,IEQIO3                        FT138010
         KEY2  REAL,PLUS,REAL,CADDO                                     FT138020
         KEY1  NEG,REAL,DNEGO                                           FT138030
         KEY2  REAL,BNEG,REAL,CSUBO                                     FT138040
         KEY2  REAL,TIMES,REAL,CMPYO                                    FT138050
         KEY2  REAL,DIV,REAL,CDIVO                                      FT138060
         KEY2  REAL,EXPNL,INTEGR,CEXPI                                  FT138070
         KEY1  NEG,INTEGR,INEGO                                         FT138080
         KEY2  INTEGR,PLUS,INTEGR,IADDO                                 FT138090
         KEY2  INTEGR,BNEG,INTEGR,ISUBO                                 FT138100
         KEY2  INTEGR,TIMES,INTEGR,IMPYO                                FT138110
         KEY2  INTEGR,DIV,INTEGR,IDIVO                                  FT138120
         KEY2  INTEGR,EXPNL,INTEGR,IEXPO                                FT138130
         KEY   ,,AUXIL,REAL,VS,AUXIL1                                   FT138140
         KEY   ,,AUXIL,REAL,AC,AUXIL4                                   FT138150
         KEY   ,,AUXIL,REAL,MQ,AUXIL5                                   FT138160
         KEY   ,,AUXIL,INTEGR,VS,AUXIL1                                 FT138170
         KEY   ,,AUXIL,INTEGR,AC,AUXIL6                                 FT138180
         KEY   ,,AUXIL,INTEGR,MQ,AUXIL2                                 FT138190
         KEY   REAL,,ARSTFN,REAL,VS,ASFDD1                              FT138200
         KEY   REAL,,ARSTFN,REAL,AC,ASFDD2                              FT138210
         KEY   REAL,,ARSTFN,REAL,MQ,ASFDD3                              FT138220
         KEY   REAL,,ARSTFN,INTEGR,VS,ASFDI1                            FT138230
         KEY   REAL,,ARSTFN,INTEGR,AC,ASFDI2                            FT138240
         KEY   REAL,,ARSTFN,INTEGR,MQ,ASFDI3                            FT138250
         REM                                                            FT138260
LKYTB3   EQU   *-KEYTB3       LENGTH OF COMPLEX ARITHMETIC KEYS         FT138270
LKEYTB   EQU   *-KEYTAB                                                 FT138280
```

```
         TTL    T A B L E S  -  B U I L T - I N  F C N .  N A M E S      FT138290
*        THE OPEN FUNCTION TABLE CONTAINS TWO-WORD ENTRIES FOR EACH      FT138300
*        BUILT-IN FUNCTION IN FASTRAN.  THE FIRST WORD OF THE ENTRY IS   FT138310
*        A KEY WORD FOR USE BY THE COMPIL AND OPNGEN ROUTINES.  THE      FT138320
*        KEY CONSISTS MAINLY OF A 'PSEUDO-RANK', SIMILAR TO AN           FT138330
*        OPERATOR RANK, BUT DESIGNED TO PERMIT EASY RECOGNITION AS       FT138340
*        A BUILT-IN FUNCTION OPERATOR, AND DIFFERENTIATION BETWEEN       FT138350
*        THE DIFFERENT FUNCTIONS.  THE SECOND WORD OF THE ENTRY IS THE   FT138360
*        BCD NAME OF THE FUNCTION.                                       FT138370
         REM                                                             FT138380
*        THE SCAN ROUTINE, WHEN IT ENCOUNTERS A FUNCTION NAME            FT138390
*        ENDING IN -F, SEARCHES THIS TABLE (LINEARLY) TO DETECT A        FT138400
*        BUILT-IN FUNCTION.  IF ONE IS FOUND, THE KEY WORD (FIRST WORD   FT138410
*        IN THE ENTRY) IS USED IN ASTACK.                                FT138420
         REM                                                             FT138430
*        IN ORDER TO DIFFERENTIATE BETWEEN CERTAIN SINGLE, DOUBLE,       FT138440
*        AND COMPLEX FUNCTIONS, THE SCAN ROUTINE WILL APPEND A NUMBER    FT138450
*        TO THE BEGINNING OF A DOUBLE PRECISION OR COMPLEX FUNCTION NAME.FT138460
*        THIS NUMBER IS 4 FOR DOUBLE AND 1 FOR COMPLEX.                  FT138470
*        THESE FUNCTIONS APPEAR THUS IN THE TABLE.                       FT138480
         REM                                                             FT138490
*        THE FUNCTION XLOC HAS BEEN MADE A BUILT-IN FUNCTION IN FASTRAN. FT138500
*        NOTICE THAT THE IBM DOUBLE PRECISION AND COMPLEX 'FIX' FUNCTIONS FT138510
*        GENERATE INCORRECT CODE, BUT THE NAMES ARE INCLUDED IN THIS TABLEFT138520
*        FOR COMPATABILITY.                                              FT138530
         SPACE  2                                                        FT138540
LIBTAB   EQU    *          OPEN ROUTINE NAMES                           FT138550
         TAB3   MIN1     THESE EIGHT MAX AND MIN ENTRIES                 FT138560
         TAB3   MIN0        MUST BE FIRST IN LIBTAB, AND                 FT138570
         TAB3   MAX0        IN THIS ORDER.  THIS IS NECESSARY            FT138580
         TAB3   MAX1        SO THAT THE CODE GENERATORS                  FT138590
         TAB3   XMIN0       WILL WORK PROPERLY.                          FT138600
         TAB3   XMIN1                                                    FT138610
         TAB3   XMAX1                                                    FT138620
         TAB3   XMAX0                                                    FT138630
         TAB3   ABS                                                      FT138640
         TAB3   XABS                                                     FT138650
         TAB3   INT                                                      FT138660
         TAB3   XINT                                                     FT138670
         TAB3   FLOAT                                                    FT138680
         TAB3   XFIX                                                     FT138690
         TAB3   MOD                                                      FT138700
         TAB3   XMOD                                                     FT138710
         TAB3   SIGN                                                     FT138720
         TAB3   XSIGN                                                    FT138730
         TAB3   DIM                                                      FT138740
         TAB3   XDIM                                                     FT138750
         TAB3   4ABS     DABSF                                           FT138760
         TAB3   4FLOAT   DFLOATF                                         FT138770
         TAB3   4FIX     DFIXF                                           FT138780
         TAB3   4SIGN    DSIGNF                                          FT138790
         TAB3   1FLOAT   IFLOATF                                         FT138800
         TAB3   1FIX     IFIXF                                           FT138810
         TAB3   1SIGN    ISIGNF                                          FT138820
         TAB3   XLOC                                                     FT138830
LOPENT   EQU    *-LIBTAB   LENGTH OF OPEN ROUTINE NAME TABLE            FT138840
```

```
         TTL    T A B L E S  -  L I B R A R Y  N A M E  T A B L E          FT138850
*        THE RESERVED LIBRARY NAME TABLE CONTAINS THE NAMES OF             FT138860
*        FMS II LIBRARY ROUTINES REFERRED TO BY COMPILER-GENERATED         FT138870
*        CODE.  (NONE OF THESE NAMES ARE LEGAL FORTRAN SYMBOLS).           FT138880
*        THE TWO-WORD ENTRY CONSISTS OF A FLAG WORD AND THE BCD NAME       FT138890
*        OF THE ROUTINE.  THE FLAG WORDS ARE ALL ZEROED OUT DURING THE     FT138900
*        INITIALIZATION OF THE COMPILER.  WHEN A ROUTINE IS USED IN THE    FT138910
*        GENERATED CODE, THE FLAG WORD IS SET NON-ZERO.  LATER THE         FT138920
*        POST PASS 1 ROUTINE TVECTL AND THE PASS 2 ROUTINE STVECT          FT138930
*        DETERMINE WHICH ROUTINES HAVE BEEN USED, AND INSERT THE BCD       FT138940
*        NAMES INTO THE TRANSFER VECTOR.                                   FT138950
         SPACE  3                                                          FT138960
SLNTAB   EQU    *              SPECIAL LIBRARY NAME TABLE.  INCLUDES ALL  FT138970
         REM                   NAMES OF LIBRARY ROUTINES CALLED BY THE    FT138980
         REM                   PROCESSORS.                                FT138990
(FPT)                                                                      FT139000
         BCI    1,(FPT)                                                   FT139010
EXP(1                                                                      FT139020
         BCI    1,EXP(1                                                   FT139030
EXP(2                                                                      FT139040
         BCI    1,EXP(2                                                   FT139050
EXP(3                                                                      FT139060
         BCI    1,EXP(3                                                   FT139070
(DFAD)                                                                     FT139080
         BCI    1,(DFAD)                                                  FT139090
(DFSB)                                                                     FT139100
         BCI    1,(DFSB)                                                  FT139110
(DFMP)                                                                     FT139120
         BCI    1,(DFMP)                                                  FT139130
(DFDP)                                                                     FT139140
         BCI    1,(DFDP)                                                  FT139150
DEXP(2                                                                     FT139160
         BCI    1,DEXP(2                                                  FT139170
DEXP(3                                                                     FT139180
         BCI    1,DEXP(3                                                  FT139190
(IFMP)                                                                     FT139200
         BCI    1,(IFMP)                                                  FT139210
(IFDP)                                                                     FT139220
         BCI    1,(IFDP)                                                  FT139230
IEXP(2                                                                     FT139240
         BCI    1,IEXP(2                                                  FT139250
(TSH)                                                                      FT139260
         BCI    1,(TSH)                                                   FT139270
(STH)                                                                      FT139280
         BCI    1,(STH)                                                   FT139290
(CSH)                                                                      FT139300
         BCI    1,(CSH)                                                   FT139310
(SCH)                                                                      FT139320
         BCI    1,(SCH)                                                   FT139330
(SPH)                                                                      FT139340
         BCI    1,(SPH)                                                   FT139350
(STB)                                                                      FT139360
         BCI    1,(STB)                                                   FT139370
(WLR)                                                                      FT139380
         BCI    1,(WLR)                                                   FT139390
(TSB)                                                                      FT139400
         BCI    1,(TSB)                                                   FT139410
(RLR)                                                                      FT139420
         BCI    1,(RLR)                                                   FT139430
(BST)                                                                      FT139440
```

```
         BCI     1,(BST)                                                 FT139450
(EFT)                                                                    FT139460
         BCI     1,(EFT)                                                 FT139470
(RWT)                                                                    FT139480
         BCI     1,(RWT)                                                 FT139490
(SDR)                                                                    FT139500
         BCI     1,(SDR)                                                 FT139510
(DRS)                                                                    FT139520
         BCI     1,(DRS)                                                 FT139530
(SLO)                                                                    FT139540
         BCI     1,(SLO)                                                 FT139550
(SLI)                                                                    FT139560
         BCI     1,(SLI)                                                 FT139570
(FIL)                                                                    FT139580
         BCI     1,(FIL)                                                 FT139590
(RTN)                                                                    FT139600
         BCI     1,(RTN)                                                 FT139610
         REM                                                             FT139620
LSLNTB   EQU     *-SLNTAB      LENGTH OF SPECIAL LIBRARY NAME TABLE      FT139630
LLIBTB   EQU     *-LIBTAB        LENGTH OF OPEN + CLOSED ROUTINE NAME TABLFT139640
```

```
         TTL    T A B L E S  -  I N S T R U C T I O N  T A B L E          FT139650
*        THIS TABLE CONSISTS OF ONE WORD INSTRUCTIONS THAT ARE USED IN    FT139660
*        GENERATING THE OBJECT CODE.                                      FT139670
         SPACE  3                                                        FT139680
OPTAB    EQU    *                                                        FT139690
         REM                                                             FT139700
ADD      ADD    0                                                        FT139710
ALS17    ALS    17                                                       FT139720
ALS18    ALS    18                                                       FT139730
ANA      ANA    0                                                        FT139740
ANA)2    ANA    **                                                       FT139750
AXT1     AXT    **,1                                                     FT139760
AXT2     AXT    **,2                                                     FT139770
AXT4     AXT    ,4                                                       FT139780
CAL      CAL    0                                                        FT139790
CAL4     CAL    ,4                                                       FT139800
CHS      CHS                                                             FT139810
CLA      CLA    0                                                        FT139820
CLA774   CLA    -4                                                       FT139830
CLA775   CLA    -3                                                       FT139840
CLA776   CLA    -2                                                       FT139850
CLA777   CLA    -1                                                       FT139860
CLS      CLS    0                                                        FT139870
CLS774   CLS    -4                                                       FT139880
CLS775   CLS    -3                                                       FT139890
CLS776   CLS    -2                                                       FT139900
CLS777   CLS    -1                                                       FT139910
COM      COM                                                             FT139920
DCT      DCT                                                             FT139930
DVP      DVP    0                                                        FT139940
DVP777   DVP    -1                                                       FT139950
FAD      FAD    0                                                        FT139960
FAD774   FAD    -4                                                       FT139970
FAD775   FAD    -3                                                       FT139980
FAD776   FAD    -2                                                       FT139990
FAD777   FAD    -1                                                       FT140000
FAD)1    FAD    **                                                       FT140010
FDP      FDP    0                                                        FT140020
FDP777   FDP    -1                                                       FT140030
FMP      FMP    0                                                        FT140040
FMP777   FMP    -1                                                       FT140050
FRN      FRN                                                             FT140060
FSB      FSB    0                                                        FT140070
FSB775   FSB    -3                                                       FT140080
FSB777   FSB    -1                                                       FT140090
LDA      LDA    0                                                        FT140100
LDC4     LDC    0,4                                                      FT140110
LDQ      LDQ    0                                                        FT140120
LDQ2     LDQ    2                                                        FT140130
LDQ774   LDQ    -4                                                       FT140140
LDQ775   LDQ    -3                                                       FT140150
LDQ776   LDQ    -2                                                       FT140160
LDQ777   LDQ    -1                                                       FT140170
LGR74    LGR    74                                                       FT140180
LLS      LLS    0                                                        FT140190
LLS18    LLS    18                                                       FT140200
LRS      LRS    0                                                        FT140210
LRS18    LRS    18                                                       FT140220
LRS35    LRS    35                                                       FT140230
LXD4     LXD    ,4                                                       FT140240
```

```
MPY     MPY     0                                                       FT140250
NOP     NOP                                                             FT140260
ORA     ORA     0                                                       FT140270
ORA)1   ORA     **                                                      FT140280
PAX1    PAX     ,1                                                      FT140290
PAX4    PAX     ,4                                                      FT140300
PXA4    PXA     ,4                                                      FT140310
PXD4    PXD     ,4                                                      FT140320
PZE     PZE                                                             FT140330
SLF     SLF                                                             FT140340
SLT     SLT     0                                                       FT140350
SLW     SLW     0                                                       FT140360
SLW8    SLW     8                                                       FT140370
SSP     SSP     0                                                       FT140380
STA     STA     0                                                       FT140390
STO     STO     0                                                       FT140400
STO775  STO     -3                                                      FT140410
STO776  STO     -2                                                      FT140420
STO777  STO     -1                                                      FT140430
STQ     STQ     0                                                       FT140440
STQ2    STQ     2                                                       FT140450
STQ774  STQ     -4                                                      FT140460
STQ775  STQ     -3                                                      FT140470
STQ776  STQ     -2                                                      FT140480
STQ777  STQ     -1                                                      FT140490
STR     STR                                                             FT140500
STZ     STZ     0                                                       FT140510
STZOV   STZ     OVFLOW                                                  FT140520
SUB     SUB     0                                                       FT140530
SWT     SWT     0                                                       FT140540
SXA1    SXA     ..,1                                                    FT140550
SXA2    SXA     ..,2                                                    FT140560
SXA4    SXA     ,4                                                      FT140570
SXD4    SXD     ,4                                                      FT140580
SXD44   SXD     4,4                                                     FT140590
TLQ2    TLQ     2                                                       FT140600
TMI     TMI     0                                                       FT140610
TNZ     TNZ     0                                                       FT140620
TNZ2    TNZ     2                                                       FT140630
TPL     TPL     0                                                       FT140640
TPL2    TPL     2                                                       FT140650
TRA     TRA     0                                                       FT140660
TRA04   TRA     ,4                                                      FT140670
TRA2    TRA     2                                                       FT140680
TRA4    TRA     1,4                                                     FT140690
TRA1    TRA*    0                                                       FT140700
TSX     TSX     ,,                                                      FT140710
TSX4    TSX     0,4                                                     FT140720
TXH4    TXH     ,4,0                                                    FT140730
TZE     TZE     0                                                       FT140740
TXI11   TXI     1,1,..                                                  FT140750
TXI14   TXI     1,4,..                                                  FT140760
TXL4    TXL     ..,4,..                                                 FT140770
TXI4    TXI     0,4,0                                                   FT140780
UFA)1   UFA     **                                                      FT140790
XCA     XCA                                                             FT140800
ZAC     ZAC                                                             FT140810
        REM                                                             FT140820
LOPTAB  EQU     *-OPTAB                                                 FT140830
```

```
                TTL     T A B L E S  -  T R A I L I N G  B L A N K  I N S E R T  FT140840
*               TABLE OF CONSTANTS USED BY THE SCANNER TO INSERT TRAILING       FT140850
*               BLANKS ONTO A SYMBOL.                                           FT140860
                REM                                                             FT140870
                OCT     777777777777                                            FT140880
                OCT     777777777700                                            FT140890
                OCT     777777770000                                            FT140900
                OCT     777777000000                                            FT140910
                OCT     777700000000                                            FT140920
                OCT     770000000000                                            FT140930
SCANT1          PZE                                                             FT140940
                OCT     60                                                      FT140950
                OCT     6060                                                    FT140960
                OCT     606060                                                  FT140970
                OCT     60606060                                                FT140980
                OCT     6060606060                                              FT140990
SCANT2          OCT     606060606060                                            FT141000
```

```
                TTL     T A B L E S  -  P O W E R S  O F  1 0                    FT141010
*               THE FOLLOWING FOUR TABLES ARE USED BY THE NUMBER PROCESSOR      FT141020
*               IN THE SCAN ROUTINE.                                            FT141030
                SPACE   2                                                       FT141040
*               DOUBLE PRECISION FLOATING POINT POWERS OF 10                    FT141050
*                       LOW ORDER PARTS                                         FT141060
                SPACE   3                                                       FT141070
                OCT     344413241541    38                                      FT141080
                OCT     340653551066    37                                      FT141090
                OCT     335674440705    36                                      FT141100
                OCT     332543515403    35                                      FT141110
                OCT     326554174006    34                                      FT141120
                OCT     323443311470    33                                      FT141130
                OCT     320202556055    32                                      FT141140
                OCT     314004260106    31                                      FT141150
                OCT     311320214722    30                                      FT141160
                OCT     306563327102    29                                      FT141170
                OCT     303617422401    28                                      FT141180
                OCT     277177204002    27                                      FT141190
                OCT     274631003151    26                                      FT141200
                OCT     271024002440    25                                      FT141210
                OCT     265354635550    24                                      FT141220
                OCT     262760512755    23                                      FT141230
                OCT     257446725444    22                                      FT141240
                OCT     253561357240    21                                      FT141250
                OCT     250132614200    20                                      FT141260
                OCT     245110475000    19                                      FT141270
                OCT     241647310000    18                                      FT141280
                OCT     236354240000    17                                      FT141290
                OCT     233760200000    16                                      FT141300
                OCT     227432000000    15                                      FT141310
                OCT     224510000000    14                                      FT141320
                OCT     221240000000    13                                      FT141330
                OCT     215400000000    12                                      FT141340
                OCT     212000000000    11                                      FT141350
                OCT     207000000000    10                                      FT141360
                OCT     203000000000    09                                      FT141370
                OCT     200000000000    08                                      FT141380
                OCT     175000000000    07                                      FT141390
                OCT     171000000000    6                                       FT141400
                OCT     166000000000    05                                      FT141410
                OCT     163000000000    04                                      FT141420
                OCT     157000000000    03                                      FT141430
                OCT     154000000000    002                                     FT141440
                OCT     151000000000    001                                     FT141450
P10L            OCT     146000000000    00                                      FT141460
                OCT     142314631463    -1                                      FT141470
                OCT     137727024365    -2                                      FT141480
                OCT     134570651767    -3                                      FT141490
                OCT     130616103131    -4                                      FT141500
                OCT     125161550741    -5                                      FT141510
                OCT     122132755432    -6                                      FT141520
                OCT     116536257220    -7                                      FT141530
                OCT     113430214163    -8                                      FT141540
                OCT     110023326451    -9                                      FT141550
                OCT     104353675565    -10                                     FT141560
                OCT     101274544452    -11                                     FT141570
                OCT     076226752042    -12                                     FT141580
                OCT     072044566404    -13                                     FT141590
                OCT     067520453466    -14                                     FT141600
```

OCT	064563526054	-15	FT141610
OCT	060754211571	-16	FT141620
OCT	055443324455	-17	FT141630
OCT	052351103530	-18	FT141640
OCT	046733322531	-19	FT141650
OCT	043111102107	-20	FT141660
OCT	040407233237	-21	FT141670
OCT	034013536714	-22	FT141680
OCT	031011262242	-23	FT141690
OCT	026324216517	-24	FT141700
OCT	022355175664	-25	FT141710
OCT	017127313050	-26	FT141720
OCT	014571074040	-27	FT141730
OCT	010450140066	-28	FT141740
OCT	005523263220	-29	FT141750
OCT	002565702500	-30	FT141760
OCT	000000000000	-31	FT141770
OCT	000000000000	-32	FT141780
OCT	0	-33	FT141790
OCT	0	-34	FT141800
OCT	0	-35	FT141810
OCT	0	-36	FT141820
OCT	0	-37	FT141830
OCT	0	-38	FT141840

```
      EJECT                                                          FT141850
*     DOUBLE PRECISION FLOATING POINT POWERS OF 10                   FT141860
*           HIGH ORDER PARTS                                         FT141870
      SPACE 3                                                        FT141880
      OCT   377454732312  38                                         FT141890
      OCT   373741367020  37                                         FT141900
      OCT   370601137163  36                                         FT141910
      OCT   365464114134  35                                         FT141920
      OCT   361755023372  34                                         FT141930
      OCT   356612334310  33                                         FT141940
      OCT   353473426555  32                                         FT141950
      OCT   347770675742  31                                         FT141960
      OCT   344623713116  30                                         FT141970
      OCT   341503074076  29                                         FT141980
      OCT   336402374713  28                                         FT141990
      OCT   332635456171  27                                         FT142000
      OCT   327512676455  26                                         FT142010
      OCT   324410545213  25                                         FT142020
      OCT   320647410336  24                                         FT142030
      OCT   315522640261  23                                         FT142040
      OCT   312417031701  22                                         FT142050
      OCT   306661534465  21                                         FT142060
      OCT   303532743536  20                                         FT142070
      OCT   300425434430  19                                         FT142080
      OCT   274674055531  18                                         FT142090
      OCT   271543212741  17                                         FT142100
      OCT   266434157115  16                                         FT142110
      OCT   262706576511  15                                         FT142120
      OCT   257553630407  14                                         FT142130
      OCT   254443023471  13                                         FT142140
      OCT   250721522450  12                                         FT142150
      OCT   245564416672  11                                         FT142160
      OCT   242452013710  10                                         FT142170
      OCT   236734654500  09                                         FT142180
      OCT   233575360400  08                                         FT142190
      OCT   230461132000  07                                         FT142200
      OCT   224750220000  06                                         FT142210
      OCT   221606500000  05                                         FT142220
      OCT   216470400000  04                                         FT142230
      OCT   212764000000  03                                         FT142240
      OCT   207620000000  02                                         FT142250
      OCT   204500000000  01                                         FT142260
PLUM  OCT   201400000000  00                                         FT142270
      OCT   175631463146  -1                                         FT142280
      OCT   172507534121  -2                                         FT142290
      OCT   167406111564  -3                                         FT142300
      OCT   163643334272  -4                                         FT142310
      OCT   160517426524  -5                                         FT142320
      OCT   155414336750  -6                                         FT142330
      OCT   151655376246  -7                                         FT142340
      OCT   146527461670  -8                                         FT142350
      OCT   143422701372  -9                                         FT142360
      OCT   137667633766  -10                                        FT142370
      OCT   134537657770  -11                                        FT142380
      OCT   131431363140  -12                                        FT142390
      OCT   125702270232  -13                                        FT142400
      OCT   122550223341  -14                                        FT142410
      OCT   117440165747  -15                                        FT142420
      OCT   113715126245  -16                                        FT142430
      OCT   110560736521  -17                                        FT142440
```

OCT	105447113564	-18	FT142450
OCT	101730171123	-19	FT142460
OCT	076571624103	-20	FT142470
OCT	073456166402	-21	FT142480
OCT	067743444004	-22	FT142490
OCT	064602666320	-23	FT142500
OCT	061465370246	-24	FT142510
OCT	055757132075	-25	FT142520
OCT	052614110061	-26	FT142530
OCT	047474723215	-27	FT142540
OCT	043773036657	-28	FT142550
OCT	040625513677	-29	FT142560
OCT	035504411377	-30	FT142570
OCT	032403472631	-31	FT142580
OCT	026637304365	-32	FT142590
OCT	023514235135	-33	FT142600
OCT	020411660744	-34	FT142610
OCT	014651264555	-35	FT142620
OCT	011524220444	-36	FT142630
OCT	006420163520	-37	FT142640
OCT	002663437347	-38	FT142650

```
         EJECT                                                          FT142660
*        DOUBLE PRECISION FIXED POINT POWERS OF 10                      FT142670
*                LOW ORDER PARTS                                        FT142680
         SPACE   3                                                      FT142690
         OCT     314631463146    -1                                     FT142700
         OCT     256050753412    -2                                     FT142710
         OCT     136152375747    -3                                     FT142720
         OCT     326161031312    -4                                     FT142730
         OCT     142161550741    -5                                     FT142740
         OCT     275013275543    -6                                     FT142750
         OCT     371232571275    -7                                     FT142760
         OCT     230734214106    -8                                     FT142770
         OCT     134057501155    -9                                     FT142780
         OCT     157467754727    -10                                    FT142790
         OCT     127753776057    -11                                    FT142800
         OCT     010627463004    -12                                    FT142810
         OCT     000702270232    -13                                    FT142820
         OCT     000055022334    -14                                    FT142830
         OCT     000004401657    -15                                    FT142840
         OCT     000000346453    -16                                    FT142850
         OCT     000000027035    -17                                    FT142860
         OCT     000000002234    -18                                    FT142870
         OCT     000000000166    -19                                    FT142880
         OCT     000000000013    -20                                    FT142890
         OCT     000000000001    -21                                    FT142900
FIOL     EQU     *                                                      FT142910
```

```
        EJECT                                                          FT142920
*       DOUBLE PRECISION FIXED POINT POWERS OF 10                      FT142930
*               HIGH ORDER PARTS                                       FT142940
        SPACE   3                                                      FT142950
        OCT     031463146314    -1                                     FT142960
        OCT     002436560507    -2                                     FT142970
        OCT     000203044672    -3                                     FT142980
        OCT     000015066705    -4                                     FT142990
        OCT     000001237055    -5                                     FT143000
        OCT     000000103067    -6                                     FT143010
        OCT     000000006553    -7                                     FT143020
        OCT     000000000527    -8                                     FT143030
        OCT     000000000042    -9                                     FT143040
        OCT     000000000003    -10                                    FT143050
        OCT     0               -11                                    FT143060
        OCT     0               -12                                    FT143070
        OCT     0               -13                                    FT143080
        OCT     0               -14                                    FT143090
        OCT     0               -15                                    FT143100
        OCT     0               -16                                    FT143110
        OCT     0               -17                                    FT143120
        OCT     0               -18                                    FT143130
        OCT     0               -19                                    FT143140
        OCT     0               -20                                    FT143150
        OCT     0               -21                                    FT143160
F10M    EQU     *                                                      FT143170
```

```
         TTL    T A B L E S  -  B C I  C I T  F L A G S                  FT143180
*        THIS TABLE IS USED BY C I T B L D WHEN THE LIST PASS 1 OPTION    FT143190
*        IS IN EFFECT.                                                   FT143200
         SPACE  3                                                        FT143210
CITBL1   BCI    1, N                                                    FT143220
         BCI    1, P                                                    FT143230
         BCI    1, R                                                    FT143240
         BCI    1, V                                                    FT143250
         BCI    1, T                                                    FT143260
         BCI    1, F                                                    FT143270
         BCI    1, C                                                    FT143280
         BCI    1, S                                                    FT143290
         BCI    1, X                                                    FT143300
         BCI    1, W                                                    FT143310
         BCI    1, E                                                    FT143320
         BCI    1, Q                                                    FT143330
         BCI    1, D                                                    FT143340
         BCI    1, G                                                    FT143350
         BCI    1, H                                                    FT143360
         BCI    1, I                                                    FT143370
         BCI    1, J                                                    FT143380
         BCI    1, K                                                    FT143390
```

```
         TTL     T A B L E S  -  C O N V T B                          FT143400
*        THIS TABLE IS USED BY C D L A B L TO CONVERT TITLE (FIXED    FT143410
*        ALPHABETIC PART) OF BINARY CARD LABEL TO COLUMN BINARY.      FT143420
         SPACE   3                                                    FT143430
CONVTB   OCT     020004001000          02        01        00         FT143440
         OCT     002000400100          05        04        03         FT143450
         OCT     000200040010          08        07        06         FT143460
         OCT     010200000001          11(=)     -         09         FT143470
         OCT     000000000042          -         -         12(')      FT143480
         OCT     440040000000          17(A)     16(+)     -          FT143490
         OCT     404041004200          20(D)     19(C)     18(B)      FT143500
         OCT     400440104020          23(G)     22(F)     21(E)      FT143510
         OCT     000040014002          -         25(I)     24(H)      FT143520
         OCT     000040424102          -         28())     27(.)      FT143530
         OCT     200000000000          32(-)     -         -          FT143540
         OCT     210022002400          35(L)     34(K)     33(J)      FT143550
         OCT     201020202040          38(O)     37(N)     36(M)      FT143560
         OCT     200120022004          41(R)     40(Q)     39(P)      FT143570
         OCT     204221020000          44(*)     43($)     -          FT143580
TMPTL    OCT     0                     -         -         -          FT143590
         OCT     120014000000          50(S)     49(/)     48( )      FT143600
         OCT     102010401100          53(V)     52(U)     51(T)      FT143610
         OCT     100210041010          56(Y)     55(X)     54(W)      FT143620
         OCT     110200001001          59(,)     -         57(Z)      FT143630
         OCT     000000001042          -         -         60(()      FT143640
```

```
*       THIS TABLE IS USED BY C D L A B L TO CONVERT BINARY CARD SEQUENCEFT143660
*       NUMBER TO COLUMN BINARY.                                        FT143670
        SPACE   3                                                       FT143680
BCDCNV  VFD     06/00,015/00000,15/BCDCNV                               FT143690
        VFD     06/01,015/00000,15/BCDCNV                               FT143700
        VFD     06/02,015/00000,15/BCDCNV                               FT143710
        VFD     06/03,015/00000,15/BCDCNV                               FT143720
        VFD     06/04,015/00000,15/BCDCNV                               FT143730
        VFD     06/05,015/00000,15/BCDCNV                               FT143740
        VFD     06/06,015/00000,15/BCDCNV                               FT143750
        VFD     06/07,015/00000,15/BCDCNV                               FT143760
        VFD     06/10,015/00000,15/BCDCNV                               FT143770
        VFD     06/11,015/00000,15/BCDCNV                               FT143780
        VFD     06/00,015/00000,15/BCDCNV+1                             FT143790
        VFD     06/01,015/00000,15/BCDCNV+1                             FT143800
        VFD     06/02,015/00000,15/BCDCNV+1                             FT143810
        VFD     06/03,015/00000,15/BCDCNV+1                             FT143820
        VFD     06/04,015/00000,15/BCDCNV+1                             FT143830
        VFD     06/05,015/00000,15/BCDCNV+1                             FT143840
        VFD     06/06,015/00000,15/BCDCNV+1                             FT143850
        VFD     06/07,015/00000,15/BCDCNV+1                             FT143860
        VFD     06/10,015/00000,15/BCDCNV+1                             FT143870
        VFD     06/11,015/00000,15/BCDCNV+1                             FT143880
```

```
         TTL     T A B L E S  -  M O N T H                          FT143890
*        THIS TABLE IS USED BY G E T C H  TO CONVERT A SIX-CHARACTER DATE FT143900
*        TO A READABLE FORM FOR PAGE HEADINGS.                      FT143910
         SPACE   3                                                  FT143920
         BCI     1,DEC 19                                           FT143930
         BCI     1,NOV 19                                           FT143940
         BCI     1,OCT 19                                           FT143950
         BCI     1,SEP 19                                           FT143960
         BCI     1,AUG 19                                           FT143970
         BCI     1,JUL 19                                           FT143980
         BCI     1,JUN 19                                           FT143990
         BCI     1,MAY 19                                           FT144000
         BCI     1,APR 19                                           FT144010
         BCI     1,MAR 19                                           FT144020
         BCI     1,FEB 19                                           FT144030
         BCI     1,JAN 19                                           FT144040
MONTH    EQU     *                                                  FT144050
```

```
        TTL     M I S C E L L A N E O U S   C O N S T A N T S                   FT144060
                                                                                FT144070
 ADREL  OCT     2                                                               FT144080
 OCT76  OCT     760000000000                                                    FT144090
        REM                                                                     FT144100
*       THE ORDER OF THE FOLLOWING THREE CARDS IS CRITICAL                      FT144110
        REM                                                                     FT144120
  PCWD1 OCT     400504000000                                                    FT144130
PCWD2                                                                           FT144140
PCWD3                                                                           FT144150
        REM                                                                     FT144160
ARGDEL  OCT     777777700000    TO MASK OUT ALL BUT ARG. NO. AND DELTA          FT144170
ARGBIT  OCT     770000000000    TO MASK OUT ALL BUT ARG. NO.                    FT144180
DELBIT  OCT     007777700000    TO MASK OUT ALL BUT DELTA                       FT144190
TOPCOM  OCT     77461           ORIGIN COMMON                                   FT144200
TG1BIT          ,1                                                              FT144210
TG7BIT          ,7                                                              FT144220
   ONE          1                                                               FT144230
        REM                                                                     FT144240
ALLSVN  OCT     777777777777  ALL BITS ON
```

```
      TTL    U N I T  D E S I G N A T I O N S  A N D  B U F F E R S   FT144250
*                                                                     FT144260
      REM                                                             FT144270
      SPACE  9                                                        FT144280
*        *   *   *   *   *   *   *   *   *   *   *   *                FT144290
      REM                                                             FT144300
*              F A S T R A N  C O M P I L E R                         FT144310
      REM                                                             FT144320
*     R E S E A R C H  C O M P U T I N G  C E N T E R                 FT144330
      REM                                                             FT144340
*           I N D I A N A  U N I V E R S I T Y                        FT144350
      REM                                                             FT144360
*              B L O O M I N G T O N ,  I N D I A N A                 FT144370
      SPACE  3                                                        FT144380
*        *   *   *   *   *   *   *   *   *   *   *                   FT144390
      REM                                                             FT144400
*                    T A P E  B U F F E R S                           FT144410
      REM                                                             FT144420
*        *   *   *   *   *   *   *   *   *   *   *                   FT144430
```

```
        EJECT                                                           FT144440
*       THE LOGICAL/PHYSICAL TAPE EQUIVALENCES ARE SET-UP HERE.         FT144450
        REM                                                             FT144460
*       BUFFER SPACE FOR THE FOLLOWING FOUR TAPES IS ALLOTTED HERE,     FT144470
        REM                                                             FT144480
*          MINTAP = SYSTEM INPUT TAPE                                   FT144490
*          MLSTAP = SYSTEM OUTPUT TAPE                                  FT144500
*          SYSBIN = SYSTEM (BINARY) LOAD TAPE                           FT144510
*          SYSUT1 = SYSTEM SCRATCH TAPE NO. 1                           FT144520
        SPACE   2                                                       FT144530
*       EACH OF THESE TAPES IS ASSIGNED TWO BUFFERS.                    FT144540
        SPACE   3                                                       FT144550
SYSTP1  EQU     1               SYSTEM TAPE                             FT144560
MINTAP  EQU     5               SYSTEM INPUT TAPE                       FT144570
MLSTAP  PZE     9               SYSTEM OUTPUT TAPE                      FT144580
SYSBIN  PZE     9               SYSTEM (BINARY) LOAD TAPE               FT144590
SYSUT1  EQU     2               SYSTEM UTILITY TAPE NO. 1               FT144600
        SPACE   2                                                       FT144610
BUFMIT  BSS     604             MINTAP- 2 BUFFERS OF 302 WORDS EACH     FT144620
        REM                             (MAX. ALLOWABLE REC. SIZE AT    FT144630
        REM                              1401 = 300 WORDS)              FT144640
        REM                                                             FT144650
BUFMLT  BSS     604             MLSTAP- 2 BUFFERS OF 302 WORDS EACH     FT144660
        REM                             (MAX. ALLOWABLE REC. SIZE AT    FT144670
        REM                              1401 = 300 WORDS)              FT144680
        REM                                                             FT144690
BUFBIN  BSS     492             SYSBIN- 2 BUFFERS OF 246 WORDS EACH     FT144700
        REM                             (8 BINARY CARDS/BLOCK)          FT144710
        REM                                                             FT144720
BUFUT1  BSS     358             SYSUT1- 2 BUFFERS OF 179 WORDS EACH     FT144730
        REM                             (CIT BLOCK (150) + CITFLG       FT144740
        REM                                  BLOCK (25) = 175)          FT144750
        SPACE   2                                                       FT144760
*       A COUPLE OF FMS II MONITOR CELLS ...                            FT144770
        REM                                                             FT144780
LINECT  EQU     97              MONITOR LINE COUNT                      FT144790
DATEBX  EQU     98              MONITOR DATE CELL                       FT144800
```

```
      TTL   C E L L S  R E Q U I R I N G  I N I T I A L I Z A T I O NFT144810
*                                                                   FT144820
*                                                                   FT144830
      REM                                                           FT144840
      SPACE  9                                                      FT144850
*        *  *  *  *  *  *  *  *  *  *  *                            FT144860
      REM                                                           FT144870
*           F A S T R A N  C O M P I L E R                          FT144880
      REM                                                           FT144890
*     R E S E A R C H  C O M P U T I N G  C E N T E R               FT144900
      REM                                                           FT144910
*           I N D I A N A  U N I V E R S I T Y                      FT144920
      REM                                                           FT144930
*              B L O O M I N G T O N ,  I N D I A N A              FT144940
      SPACE  3                                                      FT144950
*        *  *  *  *  *  *  *  *  *  *  *                            FT144960
      REM                                                           FT144970
*C E L L S  R E Q U I R I N G  I N I T I A L I Z A T I O N         FT144980
      REM                                                           FT144990
*        *  *  *  *  *  *  *  *  *  *  *
```

```
        EJECT                                                           FT145000
*       ALL CELLS LYING BETWEEN TRIGR1 AND TRIGR2 ARE ZEROED OUT        FT145010
*       IN A LOOP IN THE INITIALIZATION ROUTINE.                        FT145020
        REM                                                             FT145030
TRIGR1  EQU     *               DEMARCATE START OF INITIALIZATION BLOCK FT145040
 STOUT  BSS     1                    FLAG FOR SYMBOL TABLE              FT145050
GTSWCH  BSS     1                                                       FT145060
NOARGS  BSS     1               NUMBER OF SUBROUTINE ARGUMENTS          FT145070
DOTRMF  BSS     1                                                       FT145080
DOPNTR  BSS     1               POINTER FOR DO STACK                    FT145090
DOFLAG  BSS     1               DO FLAG CELL. TURNED ON BY A DO STATEMENT FT145100
FIRSTF  BSS     1                 STATEMENT COUNTER                     FT145110
MODERR  BSS     1               MODE ERROR FLAG                         FT145120
ENFILF  BSS     1               END-OF-FILE FLAG.                       FT145130
LSTSYM  BSS     1                 ON TO CAUSE DUMP OF SYMBOL TABLE      FT145140
LPASS1  BSS     1                 ON TO FORCE LISTING OF PASS1 OUTPUT   FT145150
PRIQQ   BSS     1                 ON TO CAUSE ON-LINE PRINTING OF       FT145160
        REM                       COMPILER OUTPUT.                      FT145170
SBTTL   BSS     1               ON IF SUBTITLING IS IN EFFECT           FT145180
SBNAME  BSS     1               NAME OF SUBPROGRAM BEING COMPILED       FT145190
SUBR    EQU     SBNAME                                                  FT145200
SNAME   EQU     SBNAME                                                  FT145210
COMCNT  BSS     1               NUMBER OF ITEMS IN THE COMMON TABLE     FT145220
NOCODE  BSS     1                 ON TO TURN OFF EXECUTION AND CODING   FT145230
TRAFLG  BSS     1               TRANSFER COMMAND FLAG CELL              FT145240
ERHERE  BSS     1               FLAG FOR ERROR IN CURRENT STATEMENT     FT145250
WRKCEL  BSS     1               WORKING CELL DIRECTOR WORD              FT145260
PCOUNT  BSS     1                 TEXT COUNTER                          FT145270
PGMCTR  BSS     1               PROGRAM COUNTER                         FT145280
LABEL   BSS     1               HOLDS THE LABELING FOR THE BINARY DECK  FT145290
LABFLG  BSS     1               ON WHEN NO MORE LABELING POSSIBLE       FT145300
ROUNDF  BSS     1               ON IF FLOATING ROUND OPTION IS IN EFFECT FT145310
MODFLG  BSS     1               MODE FLAG CELL.  ADDRESS HAS COLUMN 1 MODE FT145320
        REM                                                             FT145330
*       THE FOLLOWING 'LENGTH' CELLS MUST RETAIN THEIR RELATIVE ORDER.  FT145340
LTVECT  BSS     1               LENGTH OF TRANSFER VECTOR               FT145350
LPROLG  BSS     1               LENGTH OF PROLOGUE                      FT145360
LARSTS  BSS     1               LENGTH OF ARITHMETIC STATEMENT FUNCTIONS FT145370
LTEXTS  BSS     1               LENGTH OF TEXT                          FT145380
LCONST  BSS     1               LENGTH OF CONSTANTS                     FT145390
NOCONT  EQU     LCONST          NUMBER OF CONSTANTS IN CONSTANT TABLE   FT145400
LSTRIN  BSS     1               LENGTH OF STRINGS (FORMAT STATEMENTS)   FT145410
LWORKS  BSS     1                    LENGTH OF WORKING STORAGE          FT145420
LVARBS  BSS     1               LENGTH OF VARIABLES                     FT145430
LCOMMN  BSS     1               LENGTH OF COMMON STORAGE                FT145440
        REM                                                             FT145450
CITBL6  BSS     1                                                       FT145460
CITBL7  BSS     1                                                       FT145470
CISWTH  BSS     1                                                       FT145480
CITSW1  BSS     1               SWITCH TO INDICATE NO RECORDS ON SYSUT1 FT145490
CITSW2  BSS     1               LAST FLAG SWITCH                        FT145500
STSW1   BSS     1                                                       FT145510
TVSW1   BSS     1                                                       FT145520
TVSW2   BSS     1                                                       FT145530
TLSWTH  BSS     1                                                       FT145540
OVERFG  BSS     1               USED IN STASHING VARIABLES              FT145550
XRFLAG  BSS     1                 INDEX REGISTER FLAG CELL              FT145560
NOXRS   BSS     1               NUMBER OF XRS USED IN OBJECT PROGRAM    FT145570
        REM                                                             FT145580
9LEFT   BSS     24              BINARY OUTPUT IMAGE                     FT145590
```

```
SUM     EQU     9LEFT+1                                                 FT145600
8LEFT   EQU     9LEFT+2                                                 FT145610
8RIGHT  EQU     9LEFT+3                                                 FT145620
        REM                                                             FT145630
IFFLAG  BSS     1       ON IF AN ARRAY NAMED 'IF' HAS BEEN FOUND        FT145640
FORFLG  BSS     1       ON IF ARRAY NAMED 'FORMAT' HAS BEENFOUND        FT145650
CPOINT  BSS     1       COMMON POSITION POINTER.  DECREMENT INTEGER     FT145660
SCAN14  BSS     1                                                       FT145670
ASNFLG  BSS     1       POINTS TO FIRST ASSIGN STACK                    FT145680
ISTKFG  BSS     1       TELLS BEGINNING AND END OF ISTACK ENTRIES       FT145690
TERMFG  BSS     1       CURRENT BOTTOM OF EQUIVALENCE STACKS            FT145700
DBLMOD  BSS     1       INDICATES EITHER DOUBLE OR COMPLEX STATE.       FT145710
FLAG)1  BSS     1       ON AFTER CONSTANT HAS BEEN PUT IN STORAGE       FT145720
FLAG)2  BSS     1       ON IF 0377777 HAS BEEN PUT INTO STORAGE         FT145730
        REM                                                             FT145740
*       THE ORDER OF THE FOLLOWING SIX CELLS IS CRITICAL -- USED BY M A PFT145750
GARB20  BSS     1       DIMENSIONED, COMMON                             FT145760
GARB19  BSS     1       NON-DIMENSIONED, COMMON                         FT145770
GARB18  BSS     1       DIMENSIONED, NON-COMMON                         FT145780
GARB17  BSS     1       NON-DIMENSIONED, NON-COMMON                     FT145790
GARB16  BSS     1       STRINGS                                         FT145800
GARB15  BSS     1       LABELS                                          FT145810
        REM                                                             FT145820
GETRFL  BSS     1       FLAG CELL GOVERNING READING OF NEXT CARD        FT145830
TRIGR2  EQU     *       DEMARCATE END OF INITIALIZATION BLOCK           FT145840
```

```
         REM                                                        FT145860
CMNBRK   BSS   1          LOCATION - 1 OF LOWEST COMMON USED       FT145870
PROCSW   BSS   1        VARIABLE PROCESSOR SWITCH                  FT145880
ENTRY    BSS   1                                                   FT145890
FNCTFL   BSS   1          ON FOR FUNCTION SUBPROGRAM               FT145900
RTNFLG   BSS   1          NUMBER OF RETURN STATEMENTS IN PROGRAM   FT145910
PNFLAG   BSS   1        HOLDS CURRENT PAREN TYPE                   FT145920
CXFLAG   BSS   1        INDICATES POSSIBLE COMPLEX CONSTANT SITUAT.FT145930
PCNT     BSS   1        PAREN LEVEL COUNT CELL                     FT145940
SCAN5B   BSS   1        SCRATCH FOR THE ACANNER                    FT145950
ITRAN2   BSS   1        DATA CELL FOR THE TRANSFER-TYPE PROCESSORS FT145960
ADDIN    BSS   1        HOLDS CONSTANT ADDEND AT STASH TIME        FT145970
         REM                                                       FT145980
NXTLOC   BSS   1        NEXT AVAILABLE LOCATION IN POOL            FT145990
DPOINT   EQU   NXTLOC                                              FT146000
EPOINT   EQU   NXTLOC                                              FT146010
         REM                                                       FT146020
         REM           THESE CELLS HOLD EQUIVALENTS OF            FT146030
         REM           OPERANDS AT CODE GENERATION TIME.          FT146040
A        BSS   2       HOLDS EQUIV FOR LEFT-HAND FACTOR           FT146050
B        BSS   2       HOLDS EQUIV FOR RIGHT-HAND FACTOR          FT146060
         REM           SECOND WORD HOLDS ASTACK LOCATION          FT146070
COL73    BSS   3                                                  FT146080
SQSUM    BSS   1                                                  FT146090
         REM                                                      FT146100
CITB15   BSS   1        TEMPORARY CELL                           FT146110
SAVIND   BSS   1                                                 FT146120
CITBL5   BSS   1        BUMP PCOUNT ONLY IF THIS CELL IS ZERO    FT146130
STSW2    BSS   1                                                 FT146140
LTVPRL   BSS   1        LTVECT + LPROLG                          FT146150
LTVPRT   BSS   1                                                 FT146160
LTVPTS   EQU   LTVPRT                                            FT146170
LTPTSC   BSS   1                                                 FT146180
LTHRUS   BSS   1                                                 FT146190
LTHRUW   BSS   1                                                 FT146200
PGMBRK   BSS   1                                                 FT146210
IVARBS   BSS   1                                                 FT146220
         REM                                                     FT146230
ERASE    BSS   1                                                 FT146240
GETFGA   EQU   ERASE                                             FT146250
ERASE1   BSS   1                                                 FT146260
ERASE2   BSS   1                                                 FT146270
ERASE3   BSS   1                                                 FT146280
ERASE4   BSS   1                                                 FT146290
LPBODY   BSS   1        LENGTH OF BODY OF PROLOGUE               FT146300
         REM                                                     FT146310
LOCEND   BSS   1        THIS CELL SET TO PCOUNT WHEN END STATEMENT FT146320
         REM            IS ENCOUNTERED                          FT146330
RTNMOD   BSS   1        HOLDS MODE OF FUNCTION AS DEFINED BY COLUMNFT146340
         REM            1 ON RETURN STATEMENTS                  FT146350
PEND10   BSS   1                                                 FT146360
RASCAL   BSS   1        ON WHEN LAST PREVIOUS EXECUTABLE STATEMENT FT146370
         REM            WAS A CALL                              FT146380
DUARG2   BSS   1                                                 FT146390
GET1B    BSS   1        TITLE CARD LIST FLAG CELL               FT146400
IOLT11   BSS   1                                                 FT146410
IOLT12   BSS   1                                                 FT146420
IOLT13   BSS   1                                                 FT146430
IOLT14   BSS   1                                                 FT146440
```

```
          REM                                                                  FT146450
CURPTR    BES       3          ASTACK POINTERS FOR EACH LEVEL                  FT146460
          REM                                                                  FT146470
IOLVL     BSS       1                                                          FT146480
NODOAR    BSS       1                                                          FT146490
APTR      BSS       1                                                          FT146500
PPTR      BSS       1                                                          FT146510
SAVE      BSS       1                                                          FT146520
M1FLAG    BSS       1                                                          FT146530
M2FLAG    BSS       1                                                          FT146540
M1VAR     BSS       1                                                          FT146550
M2VAR     BSS       1                                                          FT146560
M3VAR     BSS       1                                                          FT146570
DUTRM3    BSS       1                                                          FT146580
DVSTFG    BSS       1          USED BY ROUTINES STZTST AND STZTSD              FT146590
GEN6      BSS       1                                                          FT146600
          REM                                                                  FT146610
REGIST    BSS       2          HOLDS REGISTER LOCATION OF RESULT OF            FT146620
          REM                  A CODE GENERATION.                              FT146630
          REM                  00 = VS, 01 = AC, 10 = MQ                       FT146640
          REM                  SECOND WORD ADDRESS GETS ASTACK POINTER.        FT146650
REGFLG    BSS       1          ON IF PSEUDO REG RESULT IS IN AC+MQ             FT146660
AUXFLG    BSS       1          INDICATES IF 'PLACE' OR 'STORE' ROUTINE         FT146670
AUXFG1    BSS       1                                                          FT146680
COD4      BSS       1                                                          FT146690
COD6      BSS       1                                                          FT146700
COD31     BSS       1                                                          FT146710
SPCDFG    BSS       1          DP AND CA ROUTING FLAG IN CODE ROUTINE          FT146720
LOCAT1    BSS       1                                                          FT146730
SYMST1    BSS       1                                                          FT146740
STOLB1    BSS       1            TEMPORARY                                     FT146750
GETWR7    BSS       1                                                          FT146760
SCANHT    BSS       1          TEMPORARY STORAGE FOR MQ                        FT146770
HFLAG     BSS       1          MUST BE ZEROED AT BEGINNING OF EACH STATE.      FT146780
STOSAV    BSS       1          TEMPORARY CELL USED BY STORE ROUTINE            FT146790
          REM                                                                  FT146800
*         CELLS USED BY THE SCAN ROUTINE                                       FT146810
SYMBOL    BSS       4          SCRATCH LOCATIONS FOR SCANNER                   FT146820
GETR4     BSS       1                                                          FT146830
EXPON     BSS       1                                                          FT146840
INTG      BSS       1          HOLDS INTEGER PART OF A CONSTANT                FT146850
FRAC      BSS       2          HOLDS HIGH + LOW PARTS OF FLOATING CONSTANT     FT146860
EXPONA    BSS       1            ADJUSTIVE EXPONENT IF LEADING ZEROS           FT146870
CHAR      BSS       1                                                          FT146880
DECPT     BSS       1          INDICATES DECIMAL POINT IN CONSTANT             FT146890
CX1       BSS       1          USED TO HOLD FIRST PART OF COMPLEX CONSTANT     FT146900
CX2       BSS       1          HOLDS SECOND PART OF COMPLEX CONSTANT           FT146910
ASNLCF    BSS       1          START OF LAST-BUILT ASSIGN STACK IN POOL        FT146920
SETUP2    BSS       1          CONTAINS 'NAME' OF CURRENT ROUTINE              FT146930
SAVCL2    BSS       1          HOLDS WORKING CELL POINTER FOR CELL 2 SAVE      FT146940
SPCELL    BSS       1          HOLDS EQUIV (POOL) POINTER FOR START            FT146950
          REM                  OF CALLING SEQUENCE.                            FT146960
ARGCNT    BSS       1          NUMBER OF ARGUMENTS IN CALLING SEQUENCE         FT146970
STOCEL    BSS       1          HOLDS CURRENT HI-CORE ARGUMENT LOCATION         FT146980
CMPS1     BSS       1          WORKING STORAGE                                 FT146990
          REM                                                                  FT147000
*         CELLS USED BY THE EQUIVALENCE PROCESSOR                              FT147010
INITFG    BSS       1          BEGINNING OF EQUTBL DIRECTOR STACK              FT147020
PRIOR     BSS       1          ON IF NEW STACK HAS BEEN STARTED                FT147030
LINKER    BSS       1          ON IF LINK TO OLD STACK HAS BEEN FOUND          FT147040
```

```
HI      BSS     1               TOP OF CURRENT CHAIN OF STACKS                  FT147050
LO      BSS     1               BUTTOM OF CURRENT CHAIN OF STACKS               FT147060
TOP     BSS     1               TOP OF NEW STACK NOW BEING BUILT                FT147070
DEL     BSS     1               SUBSCRIPT DIFFERENCE USED IN OLD STACKS         FT147080
DEL1    BSS     1               SUBSCRIPT INCREMENT FOR CURRENT STACK           FT147090
T1      BSS     1                                                               FT147100
T3      BSS     1                                                               FT147110
        REM                                                                     FT147120
*       WORKING CELLS FOR THE SUBSCRIPT PROCESSOR                               FT147130
*       THE ORDER OF THESE CELLS IS CRITICAL                                    FT147140
.STT    EQU     *               START OF SUBSCRIPT WORKING CELLS                FT147150
        BSS     3                                                               FT147160
CMLT    EQU     *-1                                                             FT147170
        BSS     3                                                               FT147180
CADD    EQU     *-1                                                             FT147190
        BSS     3                                                               FT147200
VARB    EQU     *-1                                                             FT147210
        BSS     3                                                               FT147220
VIND    EQU     *-1                                                             FT147230
DIM1    BSS     1                                                               FT147240
DIM2    BSS     1                                                               FT147250
DIMN    BSS     1                                                               FT147260
ASIGN   BSS     1                                                               FT147270
SAV1    BSS     1                                                               FT147280
SAV2    BSS     1                                                               FT147290
SAV3    BSS     1                                                               FT147300
TEMPX   BSS     1                                                               FT147310
CONADD  BSS     1                                                               FT147320
.CNT    EQU     *               END OF SUBSCRIPT WORKING CELLS                  FT147330
        REM                                                                     FT147340
*       CELLS USED BY THE COMPIL ROUTINE                                        FT147350
EXITFG  BSS     1               NON-ZERO FOR IF AND CALL TYPE STATEMENTS        FT147360
CALLFG  BSS     1               ON FOR CALL AND FORTRAN FUNCTION                FT147370
SUBFLG  BSS     1               ON IF A SUBSCRIPT APPEARS IN THIS STATEMENT     FT147380
ARITFN  BSS     1               ON IF THIS IS AN ARITH. STATEMENT FUNCTION      FT147390
ARFLAG  BSS     1               ON WHEN PROCESSING A CALL TO AN INTERNAL FN     FT147400
EQULFG  BSS     1               FLAG FOR INDICATING = SIGN HAS BEEN             FT147410
        REM                     ENCOUNTERED                                     FT147420
SYMTS   BSS     1               CONTAINS EQUIVALENT OF OPERAND OR OPERATOR      FT147430
        REM                     BEFORE STORING IN THE ARITHMETIC STACK.         FT147440
        REM                                                                     FT147450
        REM                                                                     FT147460
LDOSTK  EQU     75              LENGTH OF THE DO STACK. NESTING LEVEL A MAX     FT147470
        REM                     OF 25                                           FT147480
DOSTAK  BES     LDOSTK          DO STACK                                        FT147490
LASTAK  EQU     200             LENGTH OF THE ARITHMETIC STACK                  FT147500
        BSS     LASTAK                                                          FT147510
ASTACK  BSS     1               START OF THE ARITHMETIC STACK (RUNS BACK-       FT147520
        REM                     WARS IN CORE)                                   FT147530
        BSS     1               ASTACK+1 MUST BE ALL ONES FOR PROPER            FT147540
        REM                     FUNCTIONING OF THE COMPILE ROUTINE.             FT147550
        REM                     INITIALIZATION OF THE COMPILER SHOULD SET       FT147560
        REM                     THE CELL.                                       FT147570
        REM                                                                     FT147580
        BSS     25                                                              FT147590
PSTACK  BSS     1               PAREN STACK                                     FT147600
        REM                                                                     FT147610
PRLNGH  EQU     180             LENGTH OF PROLOGUE TABLE                        FT147620
PRLTBL  BSS     PRLNGH          PROLOGUE TABLE                                  FT147630
        REM                                                                     FT147640
```

```
CARD    BSS     115             STATEMENT BUFFER                        FT147650
UNLBL   EQU     CARD+24                                                 FT147660
UNSTR   EQU     UNLBL+25                                                FT147670
UNVAR   EQU     UNSTR+25                                                FT147680
NOPTH   EQU     UNVAR+25                                                FT147690
        REM                                                             FT147700
CITLNG  EQU     150                                                     FT147710
CIT1    BSS     CITLNG                                                  FT147720
CIT2    BSS     CITLNG                                                  FT147730
CTFLG1  BSS     CITLNG/6                                                FT147740
CTFLG2  BSS     CITLNG/6                                                FT147750
        REM                                                             FT147760
CPATCH  EQU     *               USE END OF CONTAB FOR OCTAL PATCHING    FT147770
NOCNST  EQU     300             LENGTH OF CONSTANT TABLE                FT147780
        BSS     NOCNST                                                  FT147790
CONTAB  EQU     *-1                                                     FT147800
        REM                                                             FT147810
        REM                                                             FT147820
.P      BSS     10                                                      FT147830
COMFLG  EQU     .P                                                      FT147840
HANGMX  EQU     .P+1                                                    FT147850
DELMAX  EQU     .P+2                                                    FT147860
CURRFG  EQU     .P+3                                                    FT147870
CTCELL  EQU     .P+4                                                    FT147880
IOFLAG  EQU     .P+6            NON-ZERO FOR INPUT LIST, ZERO FOR OUTPUT. FT147890
DEFINE  EQU     .P+7                                                    FT147900
ORIGIN  EQU     .P+8                                                    FT147910
MONTOR                          NON-ZERO IMPLIES MONITORED RUN          FT147920
MAXNLC  PZE     ,,POOL-COLUMN+1 MAXIMUM POOL AREA AVAILABLE             FT147930
LASTLA                          COTAINS LOOK-AHEAD OF LAST CARD         FT147940
   STW1 OCT     600526000000                                            FT147950
```

```
      TTL A B S O L U T E   S Y M B O L   D E F I N I T I O N S      FT147960
*                                                                    FT147970
      REM                                                            FT147980
      SPACE  9                                                       FT147990
*        *  *  *  *  *  *  *  *  *  *  *  *                          FT148000
      REM                                                            FT148010
*              F A S T R A N   C O M P I L E R                       FT148020
      REM                                                            FT148030
*     R E S E A R C H   C O M P U T I N G   C E N T E R              FT148040
      REM                                                            FT148050
*           I N D I A N A   U N I V E R S I T Y                      FT148060
      REM                                                            FT148070
*            B L O O M I N G T O N ,   I N D I A N A                 FT148080
      SPACE  3                                                       FT148090
*        *  *  *  *  *  *  *  *  *  *  *  *                          FT148100
      REM                                                            FT148110
*    A B S O L U T E   S Y M B O L   D E F I N I T I O N S           FT148120
      REM                                                            FT148130
*        *  *  *  *  *  *  *  *  *  *  *  *                          FT148140
```

```
        EJECT                                                                   FT148150
*       SYMBOL TABLE MODE CONFIGURATIONS                                        FT148160
*                                                                               FT148170
MLABL   BOOL    000000          LABEL                                           FT148180
MSTRG   BOOL    100000          STRING                                          FT148190
MINTG   BOOL    200000          INTEGER                                         FT148200
MREAL   BOOL    300000          REAL                                            FT148210
        REM                                                                     FT148220
        REM                                                                     FT148230
        REM                                                                     FT148240
*       FLAGS FOR EQUIV ENTRY IN THE SYMBOL TABLE SCHEME                        FT148250
*                                                                               FT148260
BINTF   BOOL    10              INTERNAL FUNCTION NAME (ARITHMETIC              FT148270
        REM                     STATEMENT FUNCTION)                             FT148280
BEXTF   BOOL    20              EXTERNAL NAME                                   FT148290
BARRY   BOOL    40              VARIABLE IN DIMENSION STATEMENT                 FT148300
BASSN   BOOL    100             VARIABLE IN ASSIGN OR ASSIGNED GO TO            FT148310
        REM                     STATEMENT                                       FT148320
BLIBF   BOOL    100             IF BEXTF IS ON, EXTERNAL FUNCTION IS            FT148330
        REM                     LIBRARY TYPE.                                   FT148340
BARGT   BOOL    200             SUBPROGRAM PARAMETER                            FT148350
BCOMN   BOOL    400             VARIABLE IN COMMON STORAGE                      FT148360
BEQUV   BOOL    1000            VARIABLE IN EQUIVALENCE STATEMENT               FT148370
BLHSX   BOOL    10000           VARIABLE OR LABEL OR FORMAT LABEL IS            FT148380
        REM                     DEFINED (LEFT-HAND VALUE)                       FT148390
BVARB   BOOL    20000           SYMBOL APPEARS IN VARIABLE CONTEXT IN           FT148400
        REM                     AN EXECUTABLE STATEMENT, I.E. IT MUST BE        FT148410
        REM                     DEFINED.                                        FT148420
BPATH   BOOL    40000           LABEL IS IN PATH OF FLOW                        FT148430
BDOUB   BOOL    40000           SUPPLY DOUBLE-LENGTH STORAGE FOR VARIABLE       FT148440
        REM                                                                     FT148450
        REM                                                                     FT148460
        REM                                                                     FT148470
*       FLAGS FOR ASTACK FORM OF EQUIV WORD                                     FT148480
*                                                                               FT148490
BCNST   BOOL    1               USED TO FLAG ASTACK ENTRIES AS CONSTANTS        FT148500
BOPRX   BOOL    2               USED TO FLAG ASTACK ENTRIES AS OPERATORS        FT148510
BARIT   BOOL    4               USED TO FLAG ASTACK ENTRIES AS ARGUMENTS        FT148520
        REM                          OF ARITHMETIC STATEMENT FUNCTIONS          FT148530
BOPNF   BOOL    100             DENOTES OPEN (BUILT-IN) FUNCTION                FT148540
        REM                     RATHER THAN ARITHMETIC STATEMENT FUNCTION       FT148550
        REM                                                                     FT148560
        REM                                                                     FT148570
        REM                                                                     FT148580
*       INDEXING INFORMATION FLAGS FOR EQUIV                                    FT148590
*                                                                               FT148600
        REM                     FOLLOWING TWO FLAGS ARE NOT ACTIVE NOW.         FT148610
BINDX   BOOL    0               VARIABLE IS IN AN INDEX OF SOME KIND            FT148620
BIXXR   BOOL    0               INDEX IS IN AN INDEX REGISTER                   FT148630
        REM                                                                     FT148640
BDOIX   BOOL    4000            INDEX OF AN OPEN DO LOOP                        FT148650
        REM                                                                     FT148660
        REM                                                                     FT148670
        REM                                                                     FT148680
TASSN   BOOL    100000          FLAG FOR ASSIGN STACK. IN ASSIGN STATEMENT      FT148690
TGOTO   BOOL    200000          IN ASSIGNED GO TO STATEMENT                     FT148700
        SPACE   3                                                               FT148710
*       DECREMENT FLAGS TO CITBLD                                               FT148720
*                                                                               FT148730
N       EQU     0                 NO RELOCATION, ABSOLUTE                       FT148740
```

```
          REM                                                                  FT148750
P         EQU     1         INSTRUCTION ADDRESS CONTAINS A VALUE OF            FT148760
          REM               PCOUNT.  TEXT RELOCATION                           FT148770
          REM                                                                  FT148780
R         EQU     2         INSTRUCTION ADDRESS CONTAINS POINTER               FT148790
          REM               TO EQUIV. TEXT RELOCATION OF EQUIV                 FT148800
          REM               ADDRESS FIELD.                                     FT148810
          REM                                                                  FT148820
V         EQU     3         INSTRUCTION ADDRESS IS POINTER TO EQUIV.           FT148830
          REM               IF BCOMN FLAG IS ON, COMMON RELOCATION IS          FT148840
          REM               DONE ON EQUIV ADDRESS FIELD.  IF BCOMN             FT148850
          REM               IS OFF, VARIABLE RELOCATION IS DONE ON EQ          FT148860
          REM               EQUIV ADDRESS FIELD.                               FT148870
          REM                                                                  FT148880
T         EQU     4         INSTRUCTION ADDRESS CONTAINS POINTER               FT148890
          REM               TO EQUIV. EXTERNAL SYMBOL RELOCATION.              FT148900
          REM                                                                  FT148910
F         EQU     5         INSTRUCTION ADDRESS CONTAINS POINTER TO            FT148920
          REM               EQUIV.  FORMAT STRING RELOCATION                   FT148930
          REM                                                                  FT148940
C         EQU     6         INSTRUCTION ADDRESS CONTAINS POINTER               FT148950
          REM               TO CONSTANT TABLE.  CONSTANT RELOCATION.           FT148960
          REM               SPECIAL SYMBOL TABLE                               FT148970
          REM                                                                  FT148980
S         EQU     7         INSTRUCTION ADDRESS CONTAINS ADDRESS OF            FT148990
          REM               LIBRARY SUBPROGRAM IN SLNTAB                       FT149000
          REM                                                                  FT149010
X         EQU     8       CAUSES A TRANSFER TO BE GENERATED TO                 FT149020
          REM             THE RESTORE AND RETURN SECTION OF A                  FT149030
          REM             SUBPROGRAM.                                          FT149040
          REM                                                                  FT149050
W         EQU     9       INSTRUCTION ADDRESS CONTAINS NUMBER                  FT149060
          REM               OF WORKING CELL USED.  WORKING CELL                FT149070
          REM               RELOCATION.                                        FT149080
          REM                                                                  FT149090
E         EQU     10        INSTRUCTION ADDRESS CONTAINS REFERENCE             FT149100
          REM               TO EXTERNAL (NON-LIBRARY) SUBPROGRAM ENTRY         FT149110
          REM                                                                  FT149120
Q         EQU     11        CONSTANT ADDEND FLAG                               FT149130
          REM                                                                  FT149140
D         EQU     12        ADDEND = 1 AND VARIABLE RELOCATION                 FT149150
          REM                                                                  FT149160
G         EQU     13        ADDEND = -1 AND VARIABLE RELOCATION                FT149170
          REM                                                                  FT149180
H         EQU     14        BEGINNING OF TEXT                                  FT149190
          REM                                                                  FT149200
          REM                                                                  FT149210
I         EQU     15            O P E N                                        FT149220
          REM                                                                  FT149230
J         EQU     16            O P E N                                        FT149240
          REM                                                                  FT149250
END       BOOL    77        FINAL CIT FLAG DEFINITION                          FT149260
          REM                                                                  FT149270
          REM                                                                  FT149280
          REM                                                                  FT149290
*         BCD CHARACTER EQUIVALENTS                                            FT149300
*                                                                              FT149310
EQCHAR    BOOL    13        = IN OCTAL                                         FT149320
FCHAR     BOOL    26        F IN OCTAL                                         FT149330
RPCHAR    BOOL    34        ) IN OCTAL                                         FT149340
```

```
CMCHAR  BOOL    73              COMMA IN OCTAL                                  FT149350
LPCHAR  BOOL    74              ( IN OCTAL                                      FT149360
EOS     BOOL    77              END-OF-STATEMENT                                FT149370
        REM                                                                     FT149380
        REM                                                                     FT149390
        REM                                                                     FT149400
*       PROCSW (PROCESSOR FLAG WORD FORMED BY SPRSW MACRO) BITS                 FT149410
*                                                                               FT149420
BXEQ    BOOL    400000                                                          FT149430
BTRANS  BOOL    200000                                                          FT149440
BDO     BOOL    100000                                                          FT149450
BCALL   BOOL    40000           PROCSW FLAG INDICATING CALL STATEMENT           FT149460
BEND    BOOL    20000           PROCSW FLAG INDICATING END STATEMENT            FT149470
        REM                                                                     FT149480
XEQ     EQU     1                                                               FT149490
NOXEQ   EQU     0                                                               FT149500
TRANS   EQU     1                                                               FT149510
NOTRA   EQU     0                                                               FT149520
DO      EQU     1                                                               FT149530
NODO    EQU     0                                                               FT149540
        SPACE   3                                                               FT149550
*       TAPE MODE DEFINITIONS                                                   FT149560
*                                                                               FT149570
DEC     EQU     0                                                               FT149580
BINARY  EQU     1                                                               FT149590
        REM                                                                     FT149600
        REM                                                                     FT149610
        REM                                                                     FT149620
*       DECREMENT FLAGS USED BY RESULT ROUTINE                                  FT149630
*                                                                               FT149640
VS      EQU     0               VARIABLE STORAGE                                FT149650
AC      EQU     1               AC                                              FT149660
MQ      EQU     2               MQ                                              FT149670
PAC     EQU     5               PSEUDO-AC AND REAL AC+MQ                        FT149680
PMQ     EQU     6               PSEUDO-MQ AND REAL AC+MQ                        FT149690
        REM                                                                     FT149700
        REM                                                                     FT149710
        REM                                                                     FT149720
*       DEFINITIONS USED BY THE GENERATOR KEY TABLES                            FT149730
*                                                                               FT149740
INTEGR  EQU     2                                                               FT149750
REAL    EQU     3                                                               FT149760
        REM                                                                     FT149770
AUXIL   BOOL    377             'OP' CODE FOR PLACE AND STORE ROUTINES          FT149780
        REM                                                                     FT149790
        REM                                                                     FT149800
        REM                                                                     FT149810
     .. EQU     0                                                               FT149820
TOPCOR  BOOL    77777           TOP OF CORE                                     FT149830
OVFLOW  BOOL    77462           OVERFLOW CELL (COMMON+1)                        FT149840
  STCC  BOOL    20                    PRGBOX BIT FOR SYMBOL TABLE CARD          FT149850
PRGBOX  EQU     18                    MONITOR CELL                              FT149860
```

```
        L ,   E Q U I V ,   S Y M T A B
                                                                    FT149880
         REM                                                        FT149890
         SPACE   9                                                  FT149900
*        *   *   *   *   *   *   *   *   *   *   *   *              FT149910
         REM                                                        FT149920
*             F A S T R A N   C O M P I L E R                       FT149930
         REM                                                        FT149940
*        R E S E A R C H   C O M P U T I N G   C E N T E R          FT149950
         REM                                                        FT149960
*             I N D I A N A   U N I V E R S I T Y                   FT149970
         REM                                                        FT149980
*           B L O O M I N G T O N ,   I N D I A N A                 FT149990
         SPACE   3                                                  FT150000
*        *   *   *   *   *   *   *   *   *   *   *   *              FT150010
         REM                                                        FT150020
*            P O O L ,   E Q U I V ,   S Y M T A B                  FT150030
         REM                                                        FT150040
*        *   *   *   *   *   *   *   *   *   *   *   *              FT150050
```

```
         EJECT
LSYMTB   EQU     4096            LENGTH OF THE SYMBOL TABLE                FT150060
SYMTAB   EQU     TOPCOR          ORIGIN OF SYMBOL TABLE                    FT150070
                                                                           FT150080
         REM                                                               FT150090
EQUIV    EQU     TOPCOR-LSYMTB-1 ORIGIN OR EQUIVALENT TABLE                FT150100
                                                                           FT150110
         REM                                                               FT150120
POOL     EQU     EQUIV-LSYMTB-1  ORIGIN OF  STORAGE POOL                   FT150130
DIMTBL   EQU     POOL                                                      FT150140
EQUTBL   EQU     POOL
```

```
         TTL     E N D   O F   F A S T R A N   C O M P I L E R                FT150150
INDREG   TRA     1,4              *****TEMPORARY RETURN FOR INDREG ROUTINE     FT150160
         SPACE   6                                                             FT150170
*****    PLEASE NOTE  *****                                                    FT150180
*        TO ADVANTAGEOUSLY UTILIZE SPACE, THE BSS ARRAY 'COLUMN'               FT150190
*        (LENGTH LCOLMN=666) IS PLACED 'ABOVE' THE LITERALS AND BELOW          FT150200
*        POOL.  THE DEFINITION OF 'COLUMN' (ON THE END CARD) IS                FT150210
*        THE LAST LITERAL.  TO AVOID CLOBBERING A NECESSARY LITERAL, AND       FT150220
*        TO AVOID HAVING TO REFER TO THE COLUMN REGION BY 'COLUMN+1',          FT150230
*        THE HIGHEST POSSIBLE LITERAL (0777777777777) IS NOW GENERATED.        FT150240
*        THIS IS JUST A DUMMY USAGE TO PUT A SCRATCH CELL AT THE LOCATION      FT150250
*        CALLED 'COLUMN'.                                                      FT150260
         REM                                                                   FT150270
*****    ANY ACTUAL REFERENCES TO THE BIT CONFIGURATION                        FT150280
*****    OCTAL 777777777777 MUST BE MADE BY USING THE CELL 'ALLSVN'.           FT150290
*****    DO NOT USE THE LITERAL. *******                                       FT150300
         REM                                                                   FT150310
*        THERE ARE TWO PURPOSES FOR THIS MANEUVER.  FIRST, SINCE POOL          FT150320
*        OVERFLOW IS TESTED FOR ONLY AT THE END OF EACH STATEMENT,             FT150330
*        THE POOL MAY OVERFLOW SIGNIFICANTLY BEFORE THE TEST IS MADE.          FT150340
*        THIS ARRAY COLUMN PROVIDES A BUFFER TO KEEP FROM DESTROYING           FT150350
*        THE LITERALS (AND THUS NECESSITATING THE RESTORING OF THE             FT150360
*        COMPILER).                                                            FT150370
*        SECOND, ALL SECTIONS OF THE COMPILER AFTER PASS1 CAN MAKE USE OF      FT150380
*        AN EXPANDED POOL REGION WHICH INCLUDES THE 666 COLUMN CELLS.          FT150390
         SPACE   2                                                             FT150400
         PZE     =0777777777777     FORCE OUT THE ALL-BIT LITERAL              FT150410
         ORG     *-1             RE-USE THE DUMMY CELL                         FT150420
         REM                                                                   FT150430
LCOLMN   EQU     666             LENGTH OF THE COLUMN ARRAY                    FT150440
         REM                                                                   FT150450
COLUMN   END     BEGINX          END CARD AND DEFINITION OF COLUMN ARRAY       FT150460
*     END TAPE                                                                (((((((((
```