```
* E041-001-6803 (LDR-APM)    3C NO. 180005000          REV A
*
*
*
* COMPUTER: DDP-116,416,516
*
*
* PROGRAM CATEGORY: LOADER
*
*
* PROGRAM TITLE: LDR-APM
*                RELOCATING LOADER FOR THE DDP-116,416,516 COMPUTERS
*
*
*
*
*
*
*
*
*
*
*
*
*                              APPROVAL                       DATE
*
*
*                              PROG----------------------     -----------
*
*                              SUPR----------------------     -----------
*
*                              QUAL--P.V. [illegible]-----    3. 10 . 68
*
*                                       NO. OF PAGES          ----86-----
```

Form 4536

```
*
*
*
*
* AUTHOR
*
*    HONEYWELL, INC., COMPUTER CONTROL DIVISION.
*
*
* PURPOSE
*
*          LDR-APM LOADS FROM THE ASR, HIGH SPEED READER,
*    MAGNETIC TAPE, AND DISC ON THE DDP-116,416,516 ABSOLUTE AND
*    RELOCATABLE MAIN PROGRAMS AND SUBPROGRAMS PRODUCED BY THE
*    DAP-116 ASSEMBLER, THE NEW DAP-16 ASSEMBLER IN
*    EITHER ONE-PASS OR TWO-PASS MODES, AND THE DDP-116 FORTRAN
*    IV COMPILER.
*
*
* STORAGE
*
*    1919  (DECIAML)
*    3577  (OCTAL)
*
*
* METHOD
*
*          THE BINARY OUTPUT OF THE DAP-116 ASSEMBLER AND THE
*    FORTRAN IV COMPILER INCLUDES A 14-BIT ADDRESS FOR EACH
*    MEMORY REFERENCE.  EACH INSTRUCTION IS CHECKED TO SEE IF ITS
*    REFERENCE IS IN THE SAME SECTOR AS THE INSTRUCTION.  IF SO,
*    THE 14-BIT ADDRESS IS TRUNCATED TO 9 BITS AND THE SECTOR
*    BIT OF THE INSTRUCTION WORD IS SET TO 1.  OTHERWISE, AN
*    INDIRECT WORD IS MADE IN THE CROSS-SECTOR INDIRECT WORD
*    TABLE CONSISTING OF THE INSTRUCTION'S 14-BIT ADDRESS, TAG,
*    AND FLAG.  THE INSTRUCTION IS LOADED WITH ITS INDIRECT FLAG
*    SET TO 1, ITS TAG AND ITS SECTOR BIT RESET TO 0, AND ITS ADDRESS
```

```
*    SET TO THE LOCATION OF THE INDIRECT WORD.
*
*         THE BINARY OUTPUT OF THE DAP-16 ASSEMBLER INCLUDES A 14-
*    BIT, 15-BIT OR 16-BIT ADDRESS FOR EACH MEMORY REFERENCE SO THAT
*    PROGRAMS MAY BE WRITTEN FOR COMPUTERS WITH THE EXTENDED
*    MEMORY OPTION.  NINE-BIT ADDRESS INSTRUCTIONS ARE DESECTORED
*    BY THE LOADER EITHER FOR EXECUTION IN THE EXTENDED-MEMORY-MODE
*    (TAGS ARE NOT MOVED TO INDIRECT WORDS) OR FOR EXECUTION NOT IN
*    THE EXTENDED-MEMORY-MODE (TAGS ARE MOVED TO INDIRECT WORDS),
*    AS SELECTED BY THE PROGRAMMER AT ASSEMBLY TIME VIA THE PSEUDO-
*    OPERATIONS 'ENTER EXTEND-MODE DESECTORING' (EXD) AND 'LEAVE
*    EXTEND-MODE DESECTORING' (LXD).  IT IS THE PROGRAMMER'S
*    RESPONSIBILITY TO DIRECT THAT CODE BE LOADED IN THE MODE IN
*    WHICH IT WILL BE EXECUTED.  THE DDP-516 INSTRUCTION 'LOAD
*    INDEX' (LDX) IS AN EXCEPTION AND IS ALWAYS DESECTORED IN THE
*    EXTENDED-MEMORY-MODE (THE TAG IS NOT MOVED TO THE INDIRECT WORD).
*
*         INDIRECT REFERENCE ACROSS SECTOR BOUNDARIES IS DONE
*    AUTOMATICALLY AND ONLY WHEN NEEDED.  THIS ALLOWS PROGRAMS WRITTEN
*    IN DAP-116 OR DAP-16 TO IGNORE THE SECTOR CONSTRAINTS ON
*    ADDRESSING.  IN ADDITION, WITH THE EXTENDED MEMORY OPTION,
*    DAP-16 PROGRAMS WHICH EXECUTE IN THE EXTENDED-MEMORY-MODE MAY
*    IGNORE THE 16-K BANK CONSTRAINTS.
*
*
*
*
*
* ********************************************************************
* *                          LOADING THE LOADER                      *
* ********************************************************************
*
*   1.    THE LOADER IS PROVIDED IN THE FORM OF : (A) A PAL-AP
*         FORMAT TAPE FOR 4K, WHICH ALLOWS LOADING ABOVE ITSELF, AND
*         (B) AN OBJECT TAPE WHICH MAY BE LOADED AS DESIRED.
*
*   2.    LOCATIONS 1-17 SHOULD BE LOADED AS FOLLOWS:
```

```
*
*            LOCATION  INSTRUCTION    ASR    DIGITRONICS
*
*                1      STA '57      010057    010057
*                2      OCP  '000X   030004    030001
*                3      INA  '100X   131004    131001
*                4      JMP  *-1     002003    002003
*                5      SNZ          101040    101040
*                6      JMP  *-3     002003    002003
*                7      STA  0       010000    010000
*               10      INA  '100X   131004    131001
*               11      JMP  *-1     002010    002010
*               12      LGL  8       041470    041470
*               13      INA  '000X   130004    130001
*               14      JMP  *-1     002013    002013
*               15      STA  *0      110000    110000
*               16      IRS 0        024000    024000
*               17      SZE          100040    100040
*
*    THIS WILL PERMIT THE LOADING OF SELF LOADING TAPE
*
*    3.    POSITION THE LEADER PORTION OF THE 4K SELF-LOADING TAPE
*          IN THE READER, MASTER CLEAR, SET THE PROGRAM COUNTER TO
*          '00001, AND PRESS 'START'. THE 4K LOADER WILL BE LOADED INTO
*          CORE.
*
*    4.    IT WILL BE USEFUL TO MAKE A SELF-LOADING VERSION THAT
*          LOADS ITSELF INTO HIGHER LOCATIONS, TO DO THIS,
*
*                (A) MOUNT THE OBJECT TAPE IN THE READER.
*                (B) MASTER CLEAR, SET THE A REGISTER TO 'YY000,
*                    WHERE YY IS THE FIRST OF FIVE SECTORS IN WHICH
*                    THE NEW LOADER IS TO BE PLACED.
*                (C) SET P TO 7000 AND PRESS START.  THE COMPUTER
*                    WILL HALT
*                    SET THE A REGISTER TO THE PROPER INPUT
*                    DEVICE CODE, BIT 15 FOR ASR OR BIT 16
```

```
*                 FOR HIGH SPEED READER.
*
*             (D) PRESS START.  ON A DDP-416, THERE WILL BE
*                 A SECOND HALT, FOR SETTING THE BASE SECTOR.
*                 PRESS START AGAIN AND THE OBJECT TAPE OF THE LOADER
*                 WILL LOAD INTO MEMORY. 'MR' WILL BE TYPED
*                 AND THE COMPUTER WILL HALT.
*             (E)
*                 (1) IF LDR-APM IS TO BE USED FOR LOADING FROM
*                     MAGNETIC TAPE AS WELL AS PAPER TAPE:
*                     MOUNT THE OBJECT TAPE OF I$MA IN THE
*                     PAPER TAPE READER.  SET THE A REGISTER TO
*                     'YY000, WHERE YY IS THE FIFTH SECTOR IN
*                     WHICH THE LOADER WILL RESIDE.  SET THE P
*                     REGISTER TO 'XX003 WHERE XX IS THE FOURTH
*                     SECTOR IN WHICH THE LOADER WILL RESIDE.
*                     PRESS START.  THE COMPUTER WILL HALT.  SET
*                     THE INPUT DEVICE CODE IN THE A REGISTER.
*                     AND CONTINUE AS IN (C) ABOVE.
*                     THE TAPE WILL BE LOADED, 'MR' WILL BE
*                     TYPED AND THE COMPUTER WILL HALT.
*                     MOUNT THE OBJECT TAPE OF M$UNIT-U IN
*                     THE READER, SET THE INPUT DEVICE CODE
*                     IN THE A REGISTER AND CONTINUE AS IN (C) ABOVE
*                     'LC' WILL BE TYPED AND THE COMPUTER
*                     WILL HALT
*
*                 (2) IF LDR-APM IS TO BE USED FOR LOADING FROM
*                     PAPER TAPE AND/OR DISC ONLY:
*
*                     MOUNT THE OBJECT TAPE OF DUMY-X16 IN THE
*                     PAPER TAPE READER.  SET THE INPUT DEVICE
*                     CODE IN THE A REGISTER, AS DESCRIBED IN (C)
*                     ABOVE AND PRESS START.  'LC WILL BE
*                     TYPED AND THE COMPUTER WILL HALT.
*             (F) AT THIS POINT THE DEFAULT INPUT DEVICE
*                 MAY BE SET.  INSERT THE DESIRED OCTAL
```

```
*                        DEVICE CODE IN LOCATION XX000
*                        WHERE XX IS THE SECOND SECTOR IN WHICH
*                        THE LOADER WILL RESIDE.  THE OCTAL
*                        DEVICE CODE OCCUPIES BITS 14-16.
*                        WHERE:
*                                 1 = HIGH SPEED PAPER TAPE READER
*                                 2 = ASR PAPER TAPE READER
*                                 3 = MAGNETIC TAPE
*                                 4 = DISC INPUT
*
*                        IF MAGNETIC TAPE IS SELECTED, BITS 11-13
*                        SHOULD BE SET WITH THE PHYSICAL TAPE NUMBER
*                        FROM WHICH LOADING WILL TAKE PLACE.  AT
*                        LOAD TIME THE TAPE MUST BE POSITIONED
*                        AT THE BEGINNING OF THE FILE.
*
*                    (G) LOAD PAL-AP IN 'ZZ000, WHERE ZZ IS ANY CONVENIENT
*                        SECTOR. CHANGE THE CONTENTS OF 'ZZ575 FROM 'ZZ600
*                        TO 'XX600 WHERE XX IS THE LAST OF THE FOUR
*                        SECTORS IN WHICH THE LOADER WAS PLACED. (THIS
*                        WILL BE IN THE STARTING LOCATION OF A '177 WORD
*                        BOOTSTRAP PROGRAM)
*
*                    (H) EXECUTE PAL-AP, DUMPING BETWEEN 'YY000 AND
*                        'XX577. THE OUTPUT TAPE IS YOUR NEW SELF-
*                        LOADING LOADER.
*
*
*
* *****************************************************************
* *                      LOADER OPERATING PROCEDURE               *
* *****************************************************************
*
*    1.      ONCE THE LOADER IS IN MEMORY, SET THE PROGRAM COUNTER TO
*            'XX000 OR XX006, WHERE 'XX IS THE FOURTH SECTOR OCCUPIED
*            BY THE LOADER, SET THE A-REGISTER TO THE STARTING LOCA-
*            TION AT WHICH LOADING IS TO BEGIN. IF THE A-REGISTER IS
```

```
*           CLEAR, A STARTING LOCATION OF '01000 WILL BE ASSUMED.
*           TO FORCE A STARTING LOCATION OF ZERO, TURN ON THE SIGN
*           BIT OF THE A-REGISTER. IF 'XX000 IS CHOSEN THEN 'NORMAL'
*           MODE LOADING IS TAKEN (IMPLIED LXD). IF 'XX006 IS CHOSEN,
*           THE EXTENDED MODE (IMPLIED EXD) LOADING IS TAKEN.
*
*     2.    ON THE DDP-116 OR -516..
*           SET THE B-REGISTER WITH THE LOCATION AT WHICH
*           THE CROSS-SECTOR INDIRECT WORD TABLE IS TO BEGIN.  IF THE
*           B-REGISTER IS CLEAR, A STARTING LOCATION OF '00100 WILL
*           BE ASSUMED. THIS WILL BE '40100 IF NORMAL LOADING
*           (LXD) INTO UPPER 16K.
*
*
*           NOTE..  ON COMPUTERS WITHOUT THE MEMORY LOCKOUT OPTION,
*           THE CROSS-SECTOR INDIRECT WORD TABLE MUST BE LOCATED IN
*           SECTOR ZERO OR IN THE SAME SECTOR AS THE INSTRUCTION. ON
*           COMPUTERS WITH THE MEMORY LOCKOUT OPTION,THE TABLE
*           MAY BE IN ANY SECTOR. THE CROSS SECTOR INDIRECT WORD
*           TABLE LOCATION MAY BE CHANGED DURING THE LOAD BY
*           OCCURRENCES OF THE DAP-16 PSEUDO-OPERATION 'SET BASE
*           SECTOR' (SETB). IF THE PROGRAM LOAD DOES NOT INCLUDE
*           A 'SET BASE SECTOR' PSEUDO-OPERATIONS PRIOR TO THE FIRST
*           INSTRUCTIONS WHICH REQUIRES DESECTORING, THE PROGRAMMER MUST
*           DIRECT THE LOADER TO SELECT THE PROPER SECTOR INITIALLY. IF
*           THIS IS A 'NORMAL' LOAD, START AT 'XX000 AND THE BASE
*           SECTOR WILL INITIALLY BE IN THE SAME 16K AS THE PROGRAM.
*           IF THE PROGRAM BEING LOADED IS TO RESIDE ABOVE THE
*           LOADER, CHANGE LOCATION 'SS000 FROM 'XX700 TO 'VV700
*           WHERE SS IS THE NEXT TO THE LAST SECTOR OCCUPIED BY THE
*           LOADER, XX IS THE LAST SECTOR OCCUPIED BY THE LOADER
*           AND 'VV IS THE LAST SECTOR IN CORE. FOR EXAMPLE, THE
*           LOADER RESIDES IN LOCATIONS '4000-7577 AND A PROGRAM IS
*           TO BE LOADED AT LOCATIONS 12000-14000 IN AN 8K CORE.
*           THE CONTENTS OF LOCATION '6000 OF THE LOADER MUST BE
*           CHANGED FROM '7700 TO '17700.
*
```

```
*    3.    PRESS 'START', THE COMPUTER WILL HALT.  SET BITS 14-16 OF THE
*          A-REGISTER WITH THE OCTAL DEVICE CODE.  WHERE:
*                    1=HSR(HIGH SPEED PAPER TAPE READER)
*                    2=ASR(ASR PAPER TAPE READER)
*                    3=MAGNETIC TAPE
*                    4=DISC INPUT
*
*          IF MAGNETIC TAPE (3 IN BITS 14-16) IS SELECTED, BITS 11-13
*          OF THE A-REGISTER SHOULD BE SET WITH THE PHYSICAL TAPE
*          NUMBER FROM WHICH THE PROGRAM WILL BE LOADED.  THE TAPE MUST
*          BE POSITIONED AT THE BEGINNING OF THE FILE.
*
*          DISC INPUT(4 IN BITS 14-16) IS ONLY MEANINGFUL WHEN
*          USED IN CONJUNCTION WITH "DOP", PROGRAM NO. 180021000.
*          IF DISC INPUT IS SELECTED BY THE USER, LDR-APM
*          GIVES CONTROL TO DOP AND DOP ASKS THE USER FOR
*          THE PROGRAM NAME TO BE LOADED(THE PROGRAM NAME
*          MUST BE PRESENT IN THE DOP DIRECTORY). THE
*          LOADER THEN LOADS THE REQUESTED PROGRAM.
*
*          ON THE DDP-416, THE COMPUTER WILL HALT AGAIN.
*          SET THE A-REGISTER WITH THE LOCATION AT WHICH
*          THE CROSS-SECTOR INDIRECT WORD TABLE IS TO BEGIN.  IF THE
*          A-REGISTER IS CLEAR, A STARTING LOCATION OF '100 WILL BE
*          ASSUMED.
*
*
*    4.    PRESS 'START'.  LOADING WILL PROCEED UNTIL A MESSAGE IS
*          TYPED.
*
*
*
* ********************************************************************
* *                        LOADER MESSAGES                           *
* ********************************************************************
*
*          A MESSAGE IS TYPED WHENEVER THE LOADER HALTS FOR ANY REASON.
```

```
*    UNLESS OTHERWISE SPECIFIED, UPON SUCH A HALT THE OPERATOR HAS
*    THE FOLLOWING OPTIONS..
*
*    1.    TO BEGIN A FRESH LOAD SET THE PROGRAM COUNTER TO 'XX000
*          OR 'XX006, SET THE A AND B REGISTERS AS DESCRIBED ABOVE IN
*          LOADER OPERATING PROCEDURE AND PRESS 'START'. THE LOADER
*          STATUS IS COMPLETELY RESET BEFORE PROCEEDING.  ALL SYMBOL
*          AND BASE SECTOR ENTRIES ARE CLEARED.  THE PROGRAM AND
*          COMMON COUNTERS ARE RESET. 'XX000 INITIALIZED THE LOADER FOR
*          'NORMAL' DESECTORIZING WHILE 'XX006' INITIALIZES IT FOR
*          EXTENDED DESECTORIZING.
*
*    2.    IF THE TAPE RUNS OUT OF THE TAPE READER
*          (BECAUSE OF A MISSING 'END-OF-TAPE' BLOCK), STOP THE
*          COMPUTER BY MOMENTARILY PLACING THE CPU IN SINGLE-
*          INSTRUCTION MODE, SET THE PROGRAM COUNTER TO 'XX001,
*          AND PRESS 'START'.
*
*    3.    IF A MEMORY MAP IS DESIRED, SET THE PROGRAM COUNTER TO
*          'XX002 AND PRESS 'START'.
*
*    4.    TO CONTINUE LOADING, BUT WITH A NEW ORIGIN, SET THE PROGRAM
*          COUNTER TO 'XX003, SET THE A REGISTER TO THE LOCATION WITH
*          WHICH LOADING IS TO CONTINUE AND PRESS 'START'. IF THE
*          A-REGISTER IS CLEAR, THE ORIGIN WILL REMAIN UNCHANGED.
*
*    5.    TO FORCE LOAD THE NEXT SUBPROGRAM (EVEN IF IT HAS NOT
*          BEEN CALLED) SET THE PROGRAM COUNTER TO 'XX004 AND
*          PRESS 'START'.
*
*    6.    TO FORCE EXECUTION TO BEGIN, SET THE PROGRAM COUNTER
*          TO 'XX005 AND PRESS 'START'.  THE PROGRAM WILL BE ENTERED
*          IN EXTENDED MODE (EXA) IF LOADED WITH EXTENDED DESECTORIZING
*          AND IN NORMAL MODE (DXA) IF LOADED WITH NORMAL DESECTORIZING
*
*          THE MESSAGES ARE..
*
```

```
*    LC..  LOADING COMPLETE.  PRESS 'START' TO BEGIN EXECUTION.
*
*    MR..  MORE SUBPROGRAMS REQUIRED.  RESET A REGISTER
*          WITH INPUT DEVICE AND PRESS 'START' TO CONTINUE LOADING.
*
*    CK..  CHECKSUM ERROR IN LAST BLOCK READ.  REPOSITION THE TAPE
*          AND PRESS 'START' TO IGNORE THE FAULTY READ AND CONTINUE
*          LOADING.
*
*    BL..  BLOCK TOO LARGE OR IN IMPROPER FORMAT.  REPOSITION THE
*          TAPE AND PRESS 'START' TO IGNORE THE FAULTY READ AND
*          CONTINUE LOADING.
*
*    MO..  MEMORY OVERFLOW.  A SUBPROGRAM BEING LOADED IS ATTEMPTING
*          TO OVERWRITE THE LOADER, OR A FORWARD-REFERENCE STRING
*          HAS BEEN OVERWRITTEN.  PRESS 'START' TO OBTAIN A
*          MEMORY MAP.  NO RECOVERY IS POSSIBLE FROM THIS ERROR.
*
*          IN GENERAL, THE LOADER CANNOT DETECT EITHER ASSEMBLED
*    CODE OVERLAYING THE INDIRECT WORD TABLES OR TABLE OVERLAYING
*    CODE.  IF THE PROGRAMMER DESIRES TO LOAD CODE INTO A BASE
*    SECTOR, HE MUST CHECK THE '*BASE' LINE OF THE MEMORY MAP TO
*    SEE IF TABLE AND PROGRAM OVERLAP.  THE LOADER WILL DETECT
*    OVERFLOW OF THE INDIRECT TABLE PAST THE END OF THE BASE
*    SECTOR..  IT TAKES AN 'MO' HALT IF SO.
*
*
*
* ********************************************************************
* *                           MEMORY MAP                             *
* ********************************************************************
*
*    THE MEMORY MAP IS ORGANIZED AS FOLLOWS..
*
*          *START   01005    THE ENTRY LOCATION OF THE MAIN PROGRAM.
*          *HIGH    04372    THE FIRST LOCATION NOT LOADED.
*          *NAMES   16014   BOTTOM OF THE LOADER
```

```
*            *COMN     07700   THE FIRST LOCATION AVAILABLE FOR COMMON
*            *BASE     00364   FIRST LOCATION NOT USED IN BASE SECOTR
*            SUB1      04012    ENTRY LOCATION OF SUBPROGRAM.
*             ..        ...
*            SUBX      03266**  SUBPROGRAM CALLED BUT NOT LOADED.
*             ..        ...
*
*     NOTES ABOUT MAP..
*
*     1.     UNUSED MEMORY IN THE ABOVE EXAMPLE IS 00364...00777
*            AND 04372...16013 AND NO COMMON HAS BEEN USED
*
*     2.     AT ANY TIME MEMORY MAP IS BEING PRINTED, IT
*            MAY BE TERMINATED BY HALTING THE COMPUTER,
*            SETTING THE PROGRAM COUNTER TO 'XX002 AND PRESSING 'START'.
*
*     3.     SUBPROGRAMS CALLED BUT NOT LOADED WILL BE FLAGGED WITH
*            ASTERISKS IN THE MAP.  THE LOCATION SHOWN IS THAT OF ITS MOST
*            RECENT CALL, AND PREVIOUS CALLS ARE STRUNG THERETO.
*
*     4.     THE BASE SECOTR ENTRY WILL BE MADE ONLY IF CROSS-SECTOR
*            REFERENCES WERE MADE, OR IF THE SETB PSEUDE-OPERATION
*            WAS USED.  IN THE LATTER CASE, THERE MAY BE SEVERAL "*BASE"
*            ENTRIES.
*
*     5.     THE ADDRESS OF COMN IS CONTAINED IN LOCATION AA000 WHERE
*            AA IS THE NEXT TO LAST SECTOR OF THE LOADER.
*
*
* *********************************************************************
* *                       BINARY INPUT FORMAT                         *
* *********************************************************************
*
*            EACH INPUT BLOCK IS COMPOSED OF A START CHARACTER (201),
*     FOLLOWED BY A VARIABLE LENGTH SEQUENCE OF 16-BIT WORDS
*     PUNCHED IN THE INVISIBLE FORMAT, THE LAST OF WHICH IS A 16-BIT
*     ERA CHECKSUM, FOLLOWED BY AN X-OFF (223) AND A RUBOUT (377).
```

```
*
*        AN INPUT OF AN END CHARACTER (203) FOLLOWED BY AN X-OFF
*   (223) IS CONSIDERED AN END-OF-TAPE MARK.  A SUBPROGRAM MAY NOT BE
*   SPLIT BETWEEN TAPES.
*
*        THERE ARE EIGHT TYPES OF BLOCKS, IDENTIFIED BY BITS 1...4 OF
*   THE FIRST WORD OF THE BLOCK.  IN ADDITION, BLOCK TYPE 0 IS FURTHER
*   SUBDIVIDED BY BITS 5...10 OF THE FIRST WORD.
*
*   BLOCK TYPE 0-0..  SUBPROGRAM NAME BLOCK.
*
*        0000000000000000  (000000)
*        00NNNNNN00000000
*        AAAAAAAABBBBBBBB
*        CCCCCCCCDDDDDDDD
*        EEEE........QQQQ
*        ZZZZZZZZZZZZZZZZ
*
*        N = NUMBER OF 16-BIT WORDS IN THE BLOCK, INCLUDING CONTROL
*            WORDS AND CHECKSUM.
*        A = FIRST CHARACTER OF THE FIRST ENTRY POINT NAME OF THIS
*            SUBPROGRAM IN ASCII.
*        B = SECOND CHARACTER OF THE FIRST ENTRY POINT NAME OF THIS
*            SUBPROGRAM.
*            ...
*        Q = SIXTH CHARACTER OF THE LAST ENTRY POINT NAME OF THIS
*            SUBPROGRAM.
*        Z = 16-BIT ERA CHECKSUM OF ALL THE 16-BIT WORDS IN THIS BLOCK
*            EXCEPT THE CHECKSUM ITSELF.
*
*        THE PROGRAM BREAK WILL BE SET SO THAT THE NEXT
*   SUBPROGRAM LOADS STARTING IN AN EVEN LOCATION.  IF ANY OF THE
*   SUBPROGRAM NAMES OF THE BLOCK HAVE BEEN CALLED BY PREVIOUSLY
*   LOADED PROGRAMS, THE FOLLOWING SUBPROGRAM WILL BE LOADED, OTHERWISE
*   IT WILL BE SKIPPED.
*        ENTRY POINT NAMES ARE SIX CHARACTERS, LEFT-JUSTIFIED, AND
*   FILLED WITH BLANKS (240).
```

```
*
*    BLOCK TYPES 0-1, 0-2, AND 0-3..  SPECIAL ACTION BLOCKS.
*
*           00000000NN000000
*           ----------------
*           ZZZZZZZZZZZZZZZZ
*
*           N = 1..  TURN OFF NON-LOAD FLAG (FORCE LOADING OF NEXT
*                    SUBPROGRAM).
*             = 2..  TURN ON CHAIN FLAG (USE WITH FORTRAN).
*             = 3..  END-OF-JOB (CAUSE BY $0 OR $3 IN FORTRAN).
*         --- = IGNORED, BUT INCLUDED IN CHECKSUM.
*           Z = CHECKSUM.
*
*    BLOCK TYPE 0-4..  DATA BLOCK.
*
*           0000000100000000  (000400)
*           00NNNNNN00000000
*           0LLLLLLLLLLLLLLL
*           AAAAAAAAAAAAAAAA
*           AAAAAAAABBBBBBBB
*           BBBBBBBBBBBBBBBB
*           CCCC........QQQQ
*           ZZZZZZZZZZZZZZZZ
*
*           N = NUMBER OF WORDS IN THIS BLOCK.
*           L = 15-BIT LOCATION INTO WHICH THE FIRST DATA WORD OF
*               THIS BLOCK IS TO BE LOADED.  SUCCESSIVE WORDS ARE
*               PLACED IN L+1, L+2, ETC.
*               THIS LOCATION IS TAKEN AS ABSOLUTE OR RELOCATABLE
*               ACCORDING TO THE CURRENT LOAD MODE (SEE BLOCK TYPES
*               0-24 AND 0-30).
*           A = FIRST DATA WORD IN 24-BIT FORMAT.
*           B = SECOND DATA WORD.
*               ...
*           Z = CHECKSUM.
*
```

```
*          EACH 24-BIT DATA WORD IS IN ONE OF THE FOLLOWING FORMATS..
*
*          GGGGGGGGGGGGGGGG-----000   DATA OR GENERIC.
*          FTIIII-MKKKKKKKKKKKKK010   FORWARD 9-BIT.
*          FT-RRVVVVVVVVVVVVVVVV100   KNOWN DAC.
*          FT-----MKKKKKKKKKKKKK110   FORWARD DAC.
*          FTIIIIVVVVVVVVVVVVVVRR1    KNOWN 9-BIT.
*
*          F = INDIRECT FLAG.
*          T = TAG.
*          I = INSTRUCTION.
*          V = 15- OR 16-BIT ADDRESS.
*          R = RELOCATION BITS.  00 = ABSOLUTE.
*                                01 = POSITIVELY RELOCATABLE.
*                                11 = NEGATIVELY RELOCATABLE.
*          G = 16-BIT UNMODIFIED DATA, GENERIC, OR SHIFT.
*          K = SYMBOL NUMBER.
*          M = 0..  THIS IS THE LAST SYMBOL NUMBER ASSOCIATED
*                   WITH THE ADDRESS.
*            = 1..  THE FOLLOWING SYMBOL NUMBER IS ALSO
*                   ASSOCIATED WITH THIS ADDRESS.  THE FOLLOWING
*                   SYMBOL NUMBER MAY APPEAR IN THE NEXT DATA BLOCK
*                   IF THE CURRENT WORD IS THE LAST WORD OF THE
*                   CURRENT DATA BLOCK.  THE FOLLOWING DATA WORD
*                   WILL BE IN THE FORMAT..
*
*                        -------MKKKKKKKKKKKKK---
*
*          KNOWN DAC..      ADDRESS KNOWN, DO NOT DESECTOR.
*          KNOWN 9-BIT..    ADDRESS KNOWN, DESECTOR.
*          FORWARD DAC..    SYMBOLIC ADDRESS, NOT TO BE DESECTORED
*                           WHEN ADDRESS BECOMES KNOWN.  UNIQUE SYMBOL
*                           NUMBERS WILL BE ASSIGNED TO INSTANCES OF
*                           THIS TYPE SO THAT THE REFERENCES TO THIS
*                           SYMBOL NEED NOT BE STRUNG BY THE LOADER.
*          FORWARD 9-BIT..  SYMBOLIC ADDRESS, TO BE DESECTORED WHEN
*                           ADDRESS BECOMES KNOWN.
```

```
*
*         NOTE ON DAC..  IN DAP-16, THE DAC PSEUDO-OPERATION
*         GENERATES A 16 BIT CONSTANT.  THE LOADER TRUNCATES
*         THIS CONSTANT TO 14 BITS IF IN THE LXD MODE, 15
*         BITS IF IN THE EXD MODE, OR DOES NOT TRUNCATE IT IF
*         ABSOLUTE OR NEGATIVELY RELOCATED.  THE LOADER MERGES THE
*         FLAG AND TAG BITS WITH THE ADDRESS CONSTANT, IGNORING THE
*         TAG IF IN EXTENDED DESECTORIZING.
*
*     BLOCK TYPE 0-10..  SYMBOL NUMBER DEFINITION BLOCK.
*
*         0000001000000000  (001000)
*         00NNNNNN00000000
*         CRRKKKKKKKKKKKKK
*         VVVVVVVVVVVVVVVV
*         CRRKKKKKKKKKKKKK
*         VVVVVVVVVVVVVVVV
*         ................
*         ZZZZZZZZZZZZZZZZ
*
*         N = NUMBER OF WORDS IN THIS BLOCK.
*         C = 1..  THERE MAY HAVE BEEN MORE THAN ONE
*                  REFERENCE TO THIS SYMBOL.
*             0..  THIS SYMBOL HAS OCCURRED IN ONLY ONE REFERENCE.
*         R = RELOCATION BITS.  00 = ABSOLUTE.
*                               01 = POSITIVELY RELOCATABLE.
*                               11 = NEGATIVELY RELOCATABLE.
*         K = 13-BIT SYMBOL NUMBER.
*         V = 16-BIT VALUE OF THIS SYMBOL (MAY BE NEGATIVE).
*         Z = CHECKSUM.
*
*     BLOCK TYPE 0-14..  END BLOCK.
*
*         0000001100000000  (001400)
*         00NNNNNN00000000  (002000)
*         0LLLLLLLLLLLLLLL
*         ZZZZZZZZZZZZZZZZ
```

```
*
*          N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 4).
*          L = 0 IF THIS IS THE END OF A SUBPROGRAM,
*            = 15-BIT STARTING LOCATION IF THIS IS THE END OF
*              A MAIN PROGRAM.
*              THIS LOCATION IS TAKEN AS ABSOLUTE OR RELOCATABLE
*              ACCORDING TO THE CURRENT LOAD MODE (SEE BLOCK TYPES
*              0-24 AND 0-30).
*          Z = CHECKSUM.
*
*          THIS IS THE END OF THIS SUBPROGRAM.
*
*     BLOCK TYPE 0-20..  ILLEGAL.  USED INTERNALLY BY DAP-16.
*
*     BLOCK TYPE 0-24..  RELOCATABLE MODE BLOCK.
*
*          0000010100000000  (002400)
*          00NNNNNN00000000  (001400)
*          ZZZZZZZZZZZZZZZZ  (003000)
*
*          N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 3).
*          Z = CHECKSUM
*
*          SET THE LOAD MODE TO RELOCATABLE.  ABSOLUTE IS THE ASSUMED
*     MODE, AND THE LOADER WILL RETURN TO THE ABSOLUTE MODE AT THE END
*     OF THE CURRENT SUBPROGRAM.
*
*     BLOCK TYPE 0-30..  ABSOLUTE MODE BLOCK.
*
*          0000011000000000  (003000)
*          00NNNNNN00000000  (001400)
*          ZZZZZZZZZZZZZZZZ  (002400)
*
*          N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 3).
*          Z = CHECKSUM.
*
*          SET THE LOAD MODE TO ABSOLUTE.  ABSOLUTE IS THE ASSUMED MODE.
```

```
*
*    BLOCK TYPES 0-34 AND 0-40..  ILLEGAL.  USED INTERNALLY BY DAP-16.
*
*    BLOCK TYPE 0-44..  SUBPROGRAM CALL BLOCK.
*
*          0000100100000000  (004400)
*          00NNNNNN00000000  (003400)
*          AAAAAAAABBBBBBBB
*          CCCCCCCCDDDDDDDD
*          EEEEEEEEFFFFFFFF
*          000000000000000P
*          ZZZZZZZZZZZZZZZZ
*
*          N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 7).
*          A = FIRST CHARACTER OF ENTRY POINT NAME IN ASCII.
*          B = SECOND CHARACTER OF ENTRY POINT NAME.
*              ...
*          F = SIXTH CHARACTER OF ENTRY POINT NAME.
*          P = 0..  PREVIOUS DATA WORD IS A NINE-BIT MEMORY REFERENCE.
*            = 1..  PREVIOUS DATA WORD IS A DAC.
*          Z = CHECKSUM.
*
*          THE LAST DATA WORD LOADED FROM THE PRECEDING DATA BLOCK IS
*    A REFERENCE TO THIS ENTRY POINT NAME (AND IT WAS LOADED WITH AN
*    ADDRESS OF ABSOLUTE ZERO).  ENTRY POINT NAMES ARE SIX CHARAC-
*    TERS, LEFT-JUSTIFIED, AND FILLED WITH BLANKS (240).  ONLY ONE NAME
*    WILL BE FOUND IN A SUBPROGRAM CALL BLOCK.
*
*    BLOCK TYPE 0-50..  SUBPROGRAM ENTRY POINT DEFINITION BLOCK.
*
*          0000101000000000  (005000)
*          00NNNNNN00000000
*          AAAAAAAABBBBBBBB
*          CCCCCCCCDDDDDDDD
*          EEEE........QQQQ
*          ZZZZZZZZZZZZZZZZ
*
```

```
*             N = NUMBER OF WORDS IN THIS BLOCK.
*             A = FIRST CHARACTER OF THE FIRST NAME FOR THIS ENTRY POINT
*                 IN ASCII.
*             B = SECOND CHARACTER OF THE FIRST NAME FOR THIS ENTRY POINT.
*                 ...
*             Q = SIXTH CHARACTER OF THE LAST NAME FOR THIS ENTRY POINT.
*             Z = CHECKSUM.
*
*             THE CURRENT VALUE OF THE LOADER LOCATION COUNTER HAS
*        THE INDICATED ENTRY POINT NAMES ASSOCIATED WITH IT.  IT MAY HAVE
*        MORE THAN ONE NAME.
*
*        BLOCK TYPE 0-54..  ENTER EXTENDED-MEMORY-MODE DESECTORING.
*
*             0000101100000000  (005400)
*             00NNNNNN00000000  (001400)
*             ZZZZZZZZZZZZZZZZ  (004000)
*
*             N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 3).
*             Z = CHECKSUM.
*
*             IN THE EXTEND-MODE OF DESECTORING, INDIRECT WORDS CONSIST
*        OF THE OBJECT WORD'S FLAG AND 15-BIT ADDRESS.  THE OBJECT WORD
*        IS ENTERED WITH FLAG SET AND TAG UNCHANGED.
*
*        BLOCK TYPE 0-60..  LEAVE EXTENDED-MEMORY-MODE DESECTORING.
*
*             0000110000000000  (006000)
*             00NNNNNN00000000  (001400)
*             ZZZZZZZZZZZZZZZZ  (007400)
*
*             N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 3).
*             Z = CHECKSUM.
*
*             OUT OF THE EXTEND-MODE OF DESECTORING, INDIRECT WORDS
*        CONSIST OF THE OBJECT WORD'S FLAG, TAG, AND 14-BIT ADDRESS.
*        THE OBJECT WORD IS ENTERED WITH FLAG SET AND TAG RESET.  THE
```

```
*    DDP-516 INSTRUCTION 'LOAD INDEX' (LDX) IS AN EXCEPTION..  THE
*    TAG OF THE INDIRECT WORD WILL BE ZERO AND THE TAG OF THE OBJECT
*    WORD WILL REMAIN SET.
*
*    THE LOADER IS INITIALLY NOT IN EXTENDED DESECTORIZING IF THE LOADER
*    IS STARTED AT 'XX000. IT IS INITIALLY IN EXTENDED DESECTORIZING
*    IF THE LOADER IS STARTED AT 'XX006. THE EXTEND-
*    MODE IS NOT ALTERED AT THE END OF THE CURRENT SUBPROGRAM.
*
*    BLOCK TYPE 0-64..  SET BASE SECTOR.
*
*         0000110100000000  (006400)
*         00NNNNNN00000000  (002000)
*         RLLLLLLLLLLLLLLL
*         ZZZZZZZZZZZZZZZZ
*
*         N = NUMBER OF WORDS IN THIS BLOCK (ALWAYS 4).
*         L = 15-BIT LOCATION AT WHICH THE CROSS-SECTOR INDIRECT
*             WORD TABLE IS TO BEGIN.
*         R = 0..  LOCATION IS ABSOLUTE.
*           = 1..  LOCATION IS RELOCATABLE.
*         Z = CHECKSUM.
*
*         THE OCCURENCE OF THE FIRST 'SETB' PSEUDO-OPERATION FOR
*    EACH BASE SECTOR DETERMINES THE LOCATION WITH WHICH THE INDIRECT
*    WORD TABLE WILL BEGIN IN THAT SECTOR.  THE TABLE THEN GROWS
*    UPWARD IN SUCCESSIVELY HIGHER LOCATIONS.  'SETB' PSEUDO-OPERATIONS
*    REFERENCING A SECTOR WHICH HAS BEEN PREVIOUSLY USED AS A BASE
*    SECTOR DO NOT RE-ORIGIN THE TABLE FOR THAT SECTOR--THE TABLE
*    FILLING IS RESUMED WHERE IT LEFT OFF.  THE B-REGISTER SETTING
*    AT THE START OF THE LOAD IS TREATED EXACTLY LIKE A 'SETB'
*    PSEUDO-OPERATION WHICH PRECEEDS THE FIRST WORD TO BE LOADED.
*
*         THE DEFAULT VALUE OF THE BASE SECTOR IS '00100, AT THE
*    INITIALIZATION OF THE LOADER.  THE LOADER WILL RETAIN KNOWLEDGE
*    OF THE LAST LOCATION USED IN EACH BASE SECTOR.  THE BASE SECTOR
*    IS NOT ALTERED AT THE END OF THE CURRENT SUBPROGRAM.
```

```
*
*    BLOCK TYPES 0-70 AND 0-74..  NOT ASSIGNED.
*
*    BLOCK TYPE 1            (ABSOLUTE PROGRAM WORDS)
*    BLOCK TYPE 2            (RELATIVE PROGRAMS WORDS)
*
*          CCCCNNNNNNAAAAAA
*          AAAAAAAA11111111
*          1111111111111111
*          2222222222222222
*          2222222233333333
*          3333333333333333
*          444.......KKKKKK
*          ZZZZZZZZZZZZZZZZ
*
*             C=1=STARTING ADDRESS IS ABSOLUTE
*               2=STARTING ADDRESS IS RELATIVE
*             N=NO. OF 16-BIT WORDS IN THIS BLOCK (INCLUDING CHECKSUM)
*             A=ADDRESS WHERE THE FIRST DATA WORD IN THIS BLOCK IS TO
*               BE STORED.  ALL OTHER DATA WORDS ARE STORED
*               SEQUENTIALLY.
*             1=FIRST DATA WORD
*             2...K=OTHER DATA WORDS
*             Z=16-BIT CHECKSUM
*
*          EACH DATA WORD IS 24 BITS LONG AND HAS ONE OF THE
*          FOLLOWING FORMATS.....
*
*          GGGGGGGGGGGGGGGG00000000
*                          OR
*          FTIIIIAAAAAAAAAAAAAA0D01
*                          OR
*          FTIIIIBBBBBBBBBBBBBB0D10
*                          OR
*          FTIIIICCCCCCCCCCCCCC0D11
*                          OR
*          111111VVVVVVVVVVVVVV0100
```

```
*
*                G=UNMODIFIED DATA OR GENERIC INSTRUCTION
*                F=INDIRECT FLAG
*                T=INDEX TAG
*                I=INSTRUCTION CODE
*                A=INSTRUCTION'S ABSOLUTE ADDRESS
*                B=INSTRUCTION'S RELATIVE ADDRESS
*                C=INSTRUCTION'S COMPLEMENTED RELATIVE ADDRESS
*                D=0=9-BIT ADDRESS.
*                D=1=14-BIT ADDRESS (DAC).
*                V=RELATIVE ADDRESS OF THE LAST OF A STRING OF INSTRUCTIONS
*                  WHICH ARE TO HAVE THE CURRENT LOCATION FILLED IN AS
*                  THEIR ADDRESSES.  THE STRING MAY CONTAIN DACS (WHICH
*                  ARE TO ACCEPT A FULL 14-BIT ADDRESS), WHICH ARE
*                  DISTINGUISHED FROM NINE-BITS BY HAVING AN OPERATION-
*                  CODE OF ZERO.  THE DACS IN THE STRING MUST BE STRUNG
*                  IN THE SAME MANNER AS NINE-BITS.  NO INSTRUCTION
*                  IN A STRING MAY HAVE BEEN DESECTORED INTO A BASE SECTOR
*                  OTHER THAN THE BASE SECTOR IN EFFECT AT THE TIME THE
*                  STRING VALUE IS DEFINED.  NO ERROR INDICATION CAN BE
*                  GIVEN IF THIS RESTRICTION IS VIOLATED, AND THE RESULTS
*                  ARE UNPREDICTABLE.  SINCE THE DDP-116 FORTRAN IV
*                  COMPILER DOES NOT EMIT 'SETB' BLOCKS, THIS NEED NOT
*                  NORMALLY BE OF CONCERN.
*
*             THE LOADER WILL REDUCE ADDRESS TYPES A,B, OR C TO THE
*             PROPER 9 BIT FORM (EXCEPT FOR DAC) MAKING ANY NECESSARY
*             INDIRECT REFERENCE ASSIGNMENTS AUTOMATICALLY.
*
*
*       BLOCK TYPE 3            (ABSOLUTE END JUMP)
*       BLOCK TYPE 4            (RELATIVE END JUMP)
*
*             CCCC000000AAAAAA
*             AAAAAAAA--------
*             ----------------
*             ZZZZZZZZZZZZZZZZ
```

```
*
*              C=3=TRANSFER ADDRESS IS ABSOLUTE
*                4=TRANSFER ADDRESS IS RELATIVE
*              A=TRANSFER ADDRESS (PROGRAM'S ENTRY LOCATION)
*              Z=CHECKSUM
*
*
*    BLOCK TYPE 5           (SUBROUTINE CALL)
*    BLOCK TYPE 7           (REFERENCE TO ITEM IN COMMON)
*
*          CCCCFTIIIIAAAAAA
*          AAAAAAAA11111111
*          2222222233333333
*          4444444455555555
*          66666666--------
*          ----------------
*          ZZZZZZZZZZZZZZZZ
*
*             C=5=SUBROUTINE CALL
*               7=REFERENCE TO ITEM IN COMMON
*             F=INDIRECT ADDRESS FLAG
*             T=INDEX TAG
*             I=INSTRUCTION (IF C=5, INST.=JST*)
*               IF C=7 AND INST=0000, THIS REFERENCE IS A DAC AND
*               IS NOT TO BE DESECTORED.
*             A=INSTRUCTION'S ADDRESS (IF C=5, ADDR.=0)
*               (IF C=7, ADDR. IS RELATIVE TO COMMON BASE)
*             1=SUBROUTINE OR COMMON NAME'S FIRST CHAR.
*             2=SECOND CHAR (OR SPACE)
*             3...6=OTHER CHAR. (OR SPACES)
*             Z=CHECKSUM
*
*    BLOCK TYPE 6           (SUBROUTINE OR COMMON BLOCK DEFINITION)
*
*          0110000000AAAAAA
*          AAAAAAAA11111111
*          2222222233333333
```

```
*         4444444455555555
*         66666666SS------
*         ----------------
*         ZZZZZZZZZZZZZZZZ
*
*            A=ENTRY POINT RELATIVE TO BEGINNING OF SUBROUTINE
*              (IF S=0 OR 2), OR SIZE OF COMMON BLOCK (IF S=1 OR 3).
*            1=FIRST LETTER OF SUBR. OR COMMON BLOCK NAME
*            2...6=OTHER NAME CHARACTERS (OR SPACES)
*            S=0=SUBROUTINE DEFINITION
*              1=COMMON BLOCK DEFINITION.
*              2=SUBROUTINE DEFINITION.
*              3=DATA STORAGE INTO COMMON FOLLOWS THIS BLOCK
*            Z=CHECKSUM
*
*         A 'BLOCK DATA' BLOCK (S=3) MUST HAVE A=0 AND
*   ITS NAME MUST HAVE BEEN PREVIOUSLY DEFINED (BY A BLOCK
*   WHICH HAS S=1).
*
*
*
* **********************************************************************
* *                            RESTRICTIONS                            *
* **********************************************************************
*
*   1.     THE LOADER ASSIGNS ALL COMMON BLOCKS STARTING AT 'XX700
*          DOWNWARD.  IF A 'BLOCK DATA' SUBPROGRAM IS PRESENT THAT WILL
*          STORE DATA INTO COMMON, IT IS POSSIBLE THAT THIS DATA WILL
*          ATTEMPT TO OVERWRITE THE LOADER, AND AN 'MO' MESSAGE
*          WILL BE TYPED.  TO PREVENT THIS, THE QUANTITY IN SYMBOLIC
*          LOCATION 'FIL7' ('2000 REL) SHOULD BE RAISED OR LOWERED TO
*          AVOID THE LOADER.
*
*   2.     THE DAP-116 OPTION 'SECT' IS NOT SUPPORTED, SUCH SUB-
*          PROGRAMS WILL BE DESECTORED NORMALLY.
*
*   3.     ON COMPUTERS WITHOUT THE MEMORY LOCKOUT OPTION, THE CROSS-
```

```
*             SECTOR INDIRECT WORD TABLE MUST BE LOCATED IN SECTOR ZERO.
*             OR WITHIN THE CURRENT SECTOR BEING LOADED.
*             ON COMPUTERS WITH THE MEMORY LOCKOUT OPTION, THE TABLE MAY
*             BE IN ANY SECTOR, AND ITS LOCATION MAY BE CHANGED DURING
*             THE LOAD BY OCCURRENCES OF THE DAP-16 PSEUDO-OPERATION 'SET
*             BASE SECTOR' (SETB).  IF THE PROGRAM LOAD DOES NOT INCLUDE
*             A 'SET BASE SECTOR' PSEUDO-OPERATION PRIOR TO THE FIRST
*             INSTRUCTION WHICH WILL REQUIRE DESECTORING, THE PROGRAMMER
*             MUST DIRECT THE LOADER TO SELECT THE PROPER SECTOR
*             INITIALLY, FORTRAN COMPILED PROGRAMS MAY NOT USE
*             LOCATION ZERO OF A BASE SECTOR FOR AN INDIRECT WORD IN
*             A STRING.  THIS ERROR CANNOT BE DETECTED, AND WILL RESULT
*             IN PREMATURE TRUNCATION OF THE STRING.
*
*     4.      IN GENERAL, THE LOADER CANNOT DETECT EITHER ASSEMBLED CODE
*             OVERLAYING THE INDIRECT WORD TABLES OR TABLE OVERLAYING
*             CODE.  IF THE PROGRAMMER DESIRES TO LOAD CODE INTO A BASE
*             SECTOR, HE MUST CHECK THE '*BASE' LINE OF THE MEMORY MAP TO
*             SEE IF TABLE AND PROGRAM OVERLAP.
*
*     5.      THE LOADER IS NORMALLY LOADED TO OCCUPY THE LAST
*             FOUR SECTORS OF MEMORY (FIVE SECTORS IF MAGNETIC
*             TAPE COMPATIBILITY REQUIRED.)  HOWEVER, IT CAN BE LOADED
*             WITH LOWER ORIGIN TO GIVE UPPER LOADING CAPABILITY.
*             SINCE NO PROGRAM MAY BE LOADED ABOVE COMMON, THE COMMON
*             ORIGIN (SYMBOLIC LOCATION "FIL7") (XX000+2000) SHOULD
*             BE RAISED WHEN THIS IS DONE.
*
*     6.      A SUBPROGRAM MAY NOT BE SPLIT BETWEEN TAPES.
*
*     7.      IN A STRING OF INSTRUCTIONS GENERATED BY THE DDP-116 FORTRAN
*             IV COMPILER, NO INSTRUCTION IN THE STRING MAY HAVE BEEN
*             DESECTORED INTO A BASE SECTOR OTHER THAN THE BASE SECTOR
*             IN EFFECT WHEN THE STRING VALUE IS ENCOUNTERED.  NO ERROR
*             INDICATION WILL BE GIVEN IF THIS RESTRICTION IS VIOLATED, AND
*             THE RESULT IS UNPREDICTABLE.
*
```

```
                  *     8.    COMMON MAY NOT BE USED BEFORE IT IS DEFINED.
                  *
                  *     9.    SUBPROGRAM CALLS ARE NOT NECESSARILY DESECTORED. IF THE
                  *           SUBPROGRAM ENTRY POINT IS IN THE SAME SECTOR AS THE CALL,
                  *           THE CALL WILL BE LINKED DIRECTLY TO IT, AND NO INDIRECT
                  *           WORD WILL BE CREATED IN THE BASE SECTOR.
                  *
                  *    10.    FORTRAN COMPILED PROGRAMS MAY NOT STRING INTO CODE IN
                  *           A BASE SECTOR.  THIS ERROR CANNOT BE DETECTED, AND IF IT
                  *           OCCURS, THE RESULT IS UNPREDICTABLE.
                  *
                  *
                  *
                  *
                  *
                  *
                  *
                  *
                  *
                  *
                  *
                  ***********************************************************************
                       EXD                      ENTER EXTENDED ADDRESSING
                       REL                      RELOCATE ONLY IN UNITS OF SECTORS.
         000000   ORG  EQU    *                 MAXIMUM ORIGIN FOR THIS LOADER IS '74000.
00000                  BSS    '130              SYMBOL TABLE AREA
                  *
                  *
                  *
         000122   BOT  EQU    *-6               GERM OF SYMBOL TABLE.
                  *
                  *
                  *
                       SETB   STB0              DESECTOR IN CURRENT SECTOR. .
                  *
                  *
                  *
```

```
                        * *****************************************************************
                        * *                     INPUT/OUTPUT ROUTINES                     *
                        * *****************************************************************
                        *
                        *
                        *
                        *    JST   R1K      READ ONE BLOCK, PLACE IT IN "BUFF" ... "BUFF+59".
                        *                   CHECKS CHECKSUM AND BLOCK LENGTH.  LOCATION 'R1KN'
                        *                   WILL CONTAIN WORD COUNT MINUS SIXTY.
                        *                   CONTROL JUMPS TO LOCATION 'EOT' IF END-OF-TAPE BLOCK
                        *                   ENCOUNTERED (203,223).
                        *
00130    0 003700            DAC   BUFF         R1K-1  POINTER TO INPUT BUFFER.
                        *
00131    0 000000       R1K  DAC   **           R1K+0  HERE TO READ A BLOCK.
00132    0 02 00130          LDA   R1K-1           +1  SET UP POINTER TO FIRST LOCATION
00133    0 04 03673          STA   T               +2  IN THE INPUT BUFFER.
00134    0 02 00712          LDA   ='177704        +3  SET THE BUFFER COUNT TO -60.
00135    0 04 00656          STA   R1KN            +4
00136    140040              CRA                   +5  INITIALIZE THE CHECKSUM.
00137    0 04 03674          STA   T+1             +6  C(T+1) = CHECKSUM.
                        *                              SKIP BLANK TAPE LOOP.
00140   -0 10 00660     MR1K JST*  SW                  TEST DEVICE SWITCH
00141    0 05 00711          ERA   ='000203        +8  IS IT 'ETX'.Q
00142    101040              SNZ                   +9  SKIP IF NOT.
00143    0 01 03031          JMP   EOT            +10  YES.  GO SERVICE END-OF-TAPE CONDITION.
00144    0 05 00710          ERA   ='000002       +11  NO.  IS IT 'SOH' (201).Q
00145    100040              SZE                  +12  SKIP IF SO.
00146    0 01 00140          JMP   R1K+7          +13  NO.  CONTINUE HUNT FOR GOOD TAPE.
                        *                              OK.  READ WORDS OF BLOCK.
00147    140040              CRA               R1K+14  HERE TO READ NEXT WORD.
00150    0 10 00164          JST   R1K+27         +15  READ AND TRANSLATE ONE CHARACTER.
00151    0 10 00164          JST   R1K+27         +16  READ AND TRANSLATE ANOTHER CHARACTER.
00152    0 10 00164          JST   R1K+27         +17  READ AND TRANSLATE STILL ANOTHER.
00153   -0 04 03673          STA*  T              +18  PLACE TOTAL WORD IN BUFFER.
00154    0 05 03674          ERA   T+1            +19  INCLUDE IT IN CHECKSUM.
00155    0 04 03674          STA   T+1            +20  UPDATE CHECKSUM.
```

```
00156   0 12 03673      IRS   T                 +21  INCREMENT BUFFER POINTER.
00157   0 12 00656      IRS   R1KN              +22  INCREMENT BUFFER COUNT.
00160   0 01 00147      JMP   R1K+14            +23  GO READ NEXT WORD.
00161   0 02 00707      LDA   ='141314 (BL)+24  OOPS.  BLOCK LENGTH OVERFLOW.
00162   0 10 00276      JST   TYER              +25  GRIPE TO USER.  BAD BLOCK.
00163   0 01 00132      JMP   R1K+1             +26  ON RESTART, TRY AGAIN FROM THE TOP.
                  *
00164   0 000000        DAC   **            R1K+27   HERE TO READ AND TRANSLATE ONE CHARACTER.
00165   0415 72         ALS   6                 +28  POSITION PREVIOUS CHARACTERS.
00166   0 04 03675      STA   T+2               +29  SAVE THEM AWAY WHILE WE GET NEXT ONE.
00167   0 10 00237      JST   R1K+70            +30  READ RAW CODE FROM TAPE.
00170   0 07 00706      SUB   ='000223          +31  IS IT AN 'X-OFF' CHARACTER.Q
00171   101040          SNZ                     +32  SKIP IF NOT.
00172   0 01 00231      JMP   R1K+64            +33  YES.  END OF BLOCK.  GO CHECK CHECKSUM.
00173   0 06 00706      ADD   ='000223          +34  NO.  RESTORE CODE.
00174   0406 71         ARR   7                 +35  SHIFT CODE BIT 2 TO A-REGISTER SIGN BIT.
00175   100400          SPL                     +36  IF BIT 2 IS NOT ZERO,
00176   0 01 00205      JMP   R1K+44            +37  GO TRANSLATE IT.
00177   0406 77         ARR   1                 +38  OTHERWISE, MOVE CODE BIT 1 TO SIGN.
00200   100400          SPL                     +39  IF IT IS A ONE,
00201   0 05 00705      ERA   ='120000          +40  FLIP IT AND BIT 3.
00202   0405 70         ARS   8                 +41  REPOSITION CHARACTER CORRECTLY.
00203   0 06 03675      ADD   T+2           R1K+42   COMBINE WITH PREVIOUS CHARACTERS.
00204  -0 01 00164      JMP*  R1K+27            +43  RETURN TO CALLER OF 'R1K+27'.
                  *
00205   0415 76         ALS   2             R1K+44   HERE FOR FUDGING FUNNY CODES.
00206   0 06 00704      ADD   ='000034          +45  MOVE HIGH BIT OVER TO BIT 11.
00207   0 03 00703      ANA   ='177740          +46  MASK OUT BITS 12...16.
00210   0 04 03676      STA   T+3               +47  SAVE HIGH ORDER INFO.
00211   0404 65         LGR   11                +48  NOW WORK ON LOW ORDER FIVE.
00212   0 07 00702      SUB   ='000037          +49  IF CODE IS 177,
00213   101040          SNZ                     +50
00214   0 02 00701      LDA   ='000020          +51  THEN MAKE IT 023.
00215   0 06 00700      ADD   ='000001          +52  OR IF CODE IS 176,
00216   101040          SNZ                     +53
00217   0 02 00677      LDA   ='000017          +54  THEN MAKE IT 021.
00220   0 06 00700      ADD   ='000001          +55  OR IF CODE IS 175,
```

```
00221   101040        SNZ                    +56
00222   0 02 00676    LDA   ='000011         +57  THEN MAKE IT 012.
00223   0 06 00700    ADD   ='000001         +58  OR IF CODE IS 174,
00224   101040        SNZ                    +59
00225   0 02 00675    LDA   ='000005         +60  THEN MAKE IT 005.
00226   0 06 03676    ADD   T+3              +61  RESTORE HIGH-ORDER AGAIN.
00227   0 03 00674    ANA   ='000077         +62  MASK DOWN TO SIX BITS.
00230   0 01 00203    JMP   R1K+42                +63  GO COMBINE WITH OTHER CHARACTERS
                    *
00231   0 02 03674    LDA   T+1          R1K+64  HERE WHEN BLOCK DONE.
00232   101040        SNZ                    +65  CHECK CHECKSUM.
00233  -0 01 00131    JMP*  R1K              +66  OK.  RETURN TO CALLER OF 'R1K'.
00234   0 02 00673    LDA   ='141713 (CK)+67  UHOH.  CHECKSUM ERROR.
00235   0 10 00276    JST   TYER             +68  GRIPE TO USER AND HALT.
00236   0 01 00132    JMP   R1K+1            +69  ON RESTART, TRY AGAIN FROM THE TOP.
                    *
00237   0 000000      DAC   **                     R1K+70 READ RAW CODE FROM TAPE
00240   0 02 03457    LDA   PTIN                   TEST FOR ASR(=2) OR HSR(=1)
00241   0 03 00700    ANA   =1                        +72 TEST FOR BIT 16 (HI-SPEED)
00242   101040        SNZ                             +73 SKIP IF HIGH SPEED
00243   0 01 00251    JMP   R1K+80                    +74 GO TO ASR
00244   14 0001       OCP   '0001                     +75 TURN ON HI-SPEED READER
00245   54 1001       INA   '1001                     +76 CLEAR A-REG AND READ CHARACTER
00246   0 01 00245    JMP   *-1                       +77 WAIT FOR IT
00247   14 0101       OCP   '0101                     +78 TURN OFF HI-SPEED
00250  -0 01 00237    JMP*  R1K+70                    +79 RETURN TO CALLER
00251   0 02 00672    LDA   ='017777              R1K+80 LOWSPEED-PREPARE TO WAIT
00252   0 04 03676    STA   T+3                       +81 FOR X16 DELAY CYCLE
00253   34 0004       SKS   '0004                     +82 NEXT CHARACTER READY. Q
00254   0 01 00260    JMP   *+4                       +83 NO. GO CHECK TIME
00255   54 1004       INA   '1004                     +84 YES, CLEAR A-REG AND READ CHARACTER
00256   0 01 00260    JMP   *+2                       +85 NOT READY, GO CHECK TIME.
00257  -0 01 00237    JMP*  R1K+70                    +86 RETURN TO CALLER
00260   0 12 03676    IRS   T+3                       +87 NOT READY.  COUNT THIS.
00261   0 01 00253    JMP   R1K+82                    +88 STILL TIME. GO CHECK AGAIN
00262   34 0104       SKS   '0104                     +89 TIME OUT. TURN ON READER
00263   0 01 00262    JMP   *-1                       +90 WAIT FOR ASR NOT BUSY
```

```
00264   14 0104          OCP   '0104               +91 SELECT OUTPUT MODE
00265   0 02 00671       LDA   ='000221         R1K+92 OUTPUT AN(X-ON) CHARACTER
00266   74 0004          OTA   '0004               +93 TO TURN ON ASR READER
00267   0 01 00266       JMP   *-1                 +94 WAIT FOR IT
00270   34 0104          SKS   '0104               +95 NOW WAIT TILL NOT BUSY
00271   0 01 00270       JMP   *-1                 +96 WAITING
00272   14 0004          OCP   '0004               +97 SELECT INPUT MODE
00273   34 0104          SKS   '0104            +98  TEST FOR BUSY
00274   0 01 00273       JMP   *-1              +99  WAIT FOR IT
00275   0 01 00251       JMP   R1K+80           R1K +100   GO BACK FOR NEXT CHARACTER
                      *
                      *
                      *
                      *  JST   TYER     TYPE TWO CHARACTERS IN A-REGISTER AND HALT.
                      *                 IF 'START' PRESSED, RETURN TO JST+1.
                      *
00276   0 000000      TYER DAC   **                    HERE FOR ERROR GRIPE AND HALT.
00277   0 04 03673       STA   T                       SAVE TWO-CHARACTER ARGUMENT.
00300   0 10 00313       JST   TTYN                    TURN THE TELETYPE ON.
00301   0 10 01650       JST   CRLF                    TYPE CARRIAGE-RETURN, LINE-FEED.
00302   0 02 03673       LDA   T                       PICK UP TWO-CHARACTER ARGUMENT.
00303   0 10 01643       JST   TYP2                    TYPE THE TWO CHARACTERS.
00304   0 10 01650       JST   CRLF                    CARRIAGE-RETURN, LINE-FEED.
00305   0400 40          LRL   32                      CLEAR THE A- AND B-REGISTERS.
00306   34 0104          SKS   '0104                   TEST FOR READY
00307   0 01 00306       JMP   *-1                     NO, WAIT TILLL IT IS
00310   14 0004          OCP   '0004                   THEN SET TO INPUT MODE
00311   000000           HLT                           HALT FOR USER ACTION.
00312  -0 01 00276       JMP*  TYER                    UPON RESTART, RETURN.
                      *
                      *
                      *
                      *  JST   TTYN     TURN TELETYPE ON IN OUTPUT MODE.
                      *
00313   0 000000      TTYN DAC   **                    HERE TO TURN ON ASR-33 OR -35.
00314   34 0104       TYNL SKS   '0104                 WAIT TILL THE ASR IS NOT BUSY.
00315   0 01 00314       JMP   *-1
```

```
00316    0 02 00672        LDA    ='017777           PREPARE TO WAIT FOR
00317    0 04 03676        STA    T+3                DELAY CYCLE
00320    34 0104      TYNK SKS    '0104              ASR MUST BE CONTINUOUSLY NOT BUSY FOR
00321    0 01 00314        JMP    TYNL               THAT INTERVAL.  IF NOT, HOLD TILL IT IS.
00322    0 12 03676        IRS    T+3                NOT BUSY.  CHECK THE TIME.
00323    0 01 00320        JMP    TYNK               STILL LONGER TO WAIT.
00324    14 0104           OCP    '0104              OK.  IT'S REALLY FREE.  SET OUTPUT MODE.
00325    140040            CRA                       TYPE TWO NULL CHARACTERS TO CLEAN IT OUT.
00326    0 10 01643        JST    TYP2
00327   -0 01 00313        JMP*   TTYN               RETURN TO CALLER.
                      *
                      *
                      * *****************************************************************
                      * *                          MAIN CONTROL                          *
                      * *****************************************************************
                      *
                      *
                      *
                      *           BLOK    RETURN HERE TO READ NEW BLOCK AND DISPATCH ON BLOCK
                      *
                      *                   TYPE.
00330    0 02 00707   BLER LDA    ='141314 (BL)      HERE ON BLOCK TYPE ERROR.
00331    0 10 00276        JST    TYER               GRIPE AND HALT.  CONTINUE IF PRESS 'START'.
00332    000013       BLOK EXA                       ALLOW ACCESS TO ALL OF CORE
00333    0 10 00131        JST    R1K                HERE TO INPUT NEXT BLOK.  READ A BLOCK
00334    0 02 03700        LDA    BUFF               FIRST WORD IS CONTROL WORD.
00335    0 03 00670        ANA    ='170377           WIPE OUT BITS 5...8.
00336    100040            SZE                       REST WILL BE ZERO IF NEW STYLE.
00337    0 01 01157        JMP    OLD                OLD STYLE BLOCK.  GO TO IT.
00340    0 02 03701        LDA    BUFF+1             WORD COUNT TO A-REGISTER, 'N'B8.
00341    0404 70           LGR    8                  'N'B16 = N.
00342    0 05 00667        ERA    ='177777           -N-1.
00343    0 06 00666        ADD    ='000003           -N+2.
00344    0 04 00654        STA    BUFN               SET 'WORDS LEFT PLUS ONE' COUNTER.
00345    140040            CRA
00346    0 04 01657        STA    R24S               SET THE PEG-LEG SWITCH EVEN.
00347    0 02 00374        LDA    BUFD =(DAC BUFF+1) SET BUFFER POINTER TO ONE BEFORE
```

```
00350   0 04 00655        STA   BUFP              FIRST USABLE WORD.
00351   0 02 03700        LDA   BUFF              RECOVER CONTROL WORD AGAIN.
00352   0404 70           LGR   16-8              BLOCK TYPE TO C(A)13-16.
00353   0 10 03217        JST   SWCH              DISPATCH ON BLOCK TYPE TO SERVICE ROUTINE.
00354   0 000375          DAC   NS00              00..  SUBPROGRAM NAME BLOCK.
00355   0 000413          DAC   NS04              04..  DATA BLOCK.
00356   0 000603          DAC   NS10              10..  SYMBOL NUMBER DEFINITION.
00357   0 001001          DAC   NS14              14..  END BLOCK.
00360   0 000330          DAC   BLER              20..  ILLEGAL.  USED INTERNALLY BY DAP-16.
00361   0 001015          DAC   NS24              24..  SET RELOCATABLE MODE.
00362   0 001017          DAC   NS30              30..  SET ABSOLUTE MODE.
00363   0 000330          DAC   BLER              34..  ILLEGAL.  USED INTERNALLY BY DAP-16.
00364   0 000330          DAC   BLER              40..  ILLEGAL.  USED INTERNALLY BY DAP-16.
00365   0 001022          DAC   NS44              44..  SUBPROGRAM CALL BLOCK.
00366   0 001117          DAC   NS50              50..  SUBPROGRAM ENTRY DEFINITION.
00367   0 001131          DAC   NS54              54..  ENTER EXTEND-MODE DESECTORING.
00370   0 001134          DAC   NS60              60..  LEAVE EXTEND-MODE DESECTORING.
00371   0 001140          DAC   NS64              64..  SET BASE SECTOR.
00372   0 000330          DAC   BLER              70 AND 74..  NOT ASSIGNED.
00373   0 000330          DAC   BLER
                   *
00374   0 003701   BUFD   DAC   BUFF+1            POINTER TO FIRST INFO WORD, LESS ONE.
                   *
                   *
                   *
                   * *********************************************************************
                   * *                       NEW STYLE BLOCK ROUTINES                    *
                   * *********************************************************************
                   *
                   *
                   *
                   *           NS00    NEW STYLE 0..  SUBPROGRAM NAME BLOCK.
                   *
00375   0 02 00665 NS00   LDA   ='100005          SET KEY TO INDICATE SUBPROGRAM
00376   0 04 03665        STA   KEY               NAME RATHER THAN COMMON NAME.
00377   0 10 01556 SNNF   JST   R48B              READ SUBPROGRAM NAME.
00400   0 10 02356        JST   SSS               SET UP SYMBOL TABLE SEARCH.
```

```
00401   0 02 00674 SNNL LDA   ='000077            LOOK AT ALL ENTRIES OF TYPE 5.
00402   0 10 02362      JST   SSN                 LOOK AT NEXT ENTRY.
00403   0 01 00377      JMP   SNNF                NOT FOUND.  IGNORE.
00404   0 10 02412      JST   MACH                DOES THIS ENTRY MATCH.Q
00405   0 01 00401      JMP   SNNL                NO.  CONTINUE HUNT.
00406   101040          SNZ                       FOUND.  IS IT CALLED OR LOADED.Q
00407   0 01 00377      JMP   SNNF                LOADED.  IGNORE IT.
00410   140040     FORS CRA                       CALLED.  WE MUST LOAD IT NOW.
00411   0 04 03671      STA   SKIF                TURN OFF SKIP FLAG.
00412   0 01 00332      JMP   BLOK                READ ON
                   *
                   *
                   *
                   *          NS04    NEW STYLE BLOCK 4..  DATA BLOCK.
                   *
00413   0 10 03024 NS04 JST   SKIP                IGNORE IF SKIPPING THIS SUBPROGRAM.
00414   0 10 01552      JST   LEFT                COUNT ORIGIN WORD.
00415   0 12 00655      IRS   BUFP                MOVE TO IT.
00416  -0 02 00655      LDA*  BUFP                PICK UP BLOCK ORIGIN.
00417   0 04 03677      STA   VALU                STORE IT FOR POSSIBLE RELOCATION.
00420   0 02 03667      LDA   MODE                GET ABSOLUTE OR RELOCATABLE SWITCH.
00421   0 10 03177      JST   RELO                RELOCATE ADDRESS IF NECESSARY.
00422   0 10 02653      JST   MRG2                MERGE BIT 2 OF PBRK IF NORMAL MODE
00423   0 04 03777      STA   WORD                SET LOCATION COUNTER.
00424   0 10 02607      JST   FITS                CHECK FOR MEMORY OVERFLOW.
00425   0 02 00443      LDA   NSDR =(JMP R24B+1) FIRST TIME, PASS THROUGH.
00426   101000     NSDS NOP                       OTHER TIMES, RETURN VIA 'R24B.
00427   0 04 00426      STA   *-1
                   *
                   *                  HERE TO PROCESS NEXT DATA WORD.
                   *
00430   0 10 01515 NSDL JST   R24B                READ NEXT 24-BIT DATA WORD.
00431   0 02 03775      LDA   DATA+1              WORD TYPE TO C(A-REG)6-8.
00432   0416 71         ALR   8-1                 C(W)24 TO C(A)1, C(W)22,23 TO C(A)15,16.
00433   100400          SPL                       CHECK FOR KNOWN NINE-BIT.
00434   0 01 00570      JMP   NSD4                YES.  GO TO IT.
00435   0 03 00666      ANA   ='000003            NO.  ISOLATE TYPE CODE.
```

```
00436  0 10 03217       JST   SWCH                 SWITCH JUMP, DEPENDS ON A-REG. CONTENTS.
00437  0 000444         DAC   NSD0                 0..  DATA OR GENERIC.
00440  0 000451         DAC   NSD1                 1..  9-BIT SYMBOLIC ADDRESS.
00441  0 000523         DAC   NSD2                 2..  DAC, ADDRESS KNOWN.
00442  0 000540         DAC   NSD3                 3..  DAC, SYMBOLIC ADDRESS.
                  *
00443  0 01 01516 NSDR JMP   R24B+1               AFTER FIRST TIME, TAKE EXISTING RETURN.
                  *
                  *
                  *
                  *          NSD0    HERE FOR DATA OR GENERIC.
                  *
00444  0 02 03774 NSD0 LDA   DATA                 GET WORD.
00445 -0 04 03777      STA*  WORD                 PLACE IT.
00446  0 12 03777 NSDT IRS   WORD                 BUMP THE LOCATION COUNTER.
00447  0 10 02607      JST   FITS                 CHECK FOR MEMORY OVERFLOW.
00450  0 01 00430      JMP   NSDL                 GO TO NEXT DATA WORD.
                  *
                  *
                  *
                  *                  HERE FOR FORWARD-REFERENCE NINE-BIT.
                  *
00451  0 02 03774 NSD1 LDA   DATA                 PLACE FLAG, TAG, AND
00452  0 03 00664      ANA   ='176000             OP-CODE INTO STORAGE.
00453 -0 04 03777      STA*  WORD
00454  0 10 01147      JST   EXT                  EXTRACT SYMBOL NUMBER.
00455  0 03 00672      ANA   ='017777             RETAIN ONLY 13 BITS.
00456  0 04 00657      STA   K
00457  0 10 02402      JST   CUT                  FORM KEY FOR CURRENT BASE AND
00460  0 05 00700      ERA   ='000001             MEMORY MODE.
00461  0 04 03665      STA   KEY
00462  0 02 03774      LDA   DATA                 GET THE 'MORE' BIT.
00463  0 03 00663      ANA   ='000400             IF ON, THIS SYMBOL HAS NOT
00464  100040          SZE                        BEEN USED BEFORE.
00465  0 01 00503      JMP   NSDN                 ON.  OMIT TABLE SEARCH.
00466  0 10 02356      JST   SSS                  OFF.  SET UP SYMBOL SEARCH.
00467  0 02 00667 NSDP LDA   ='177777             KEY MUST MATCH EXACTLY.
```

```
00470    0 10 02362        JST   SSN                    MOVE TO NEXT ENTRY.
00471    0 01 00503        JMP   NSDN                   END OF TABLE.  NOT FOUND.
00472    0 05 00657        ERA   K                      IS IT SAME SYMBOL.Q
00473   100040             SZE                          SKIP IF SO.
00474    0 01 00467        JMP   NSDP                   NO.  CONTINUE SEARCH.
00475    0 10 02646        JST   GET2                   YES.  GET LAST LOC USED.
00476    0 04 03677        STA   VALU                   MUST STRING TO IT.
00477    0 02 03777        LDA   WORD                   UPDATE USAGE POINTER IN
00500   -0 04 03673        STA*  T                      SYMBOL TABLE TO CURRENT LOC.
00501    0 10 02207        JST   DESS                   DESECTOR FOR STRING.
00502    0 01 00446        JMP   NSDT                   GO TO NEXT DATA WORD.
                     *
00503    0 10 02305  NSDN  JST   GEN3                   NOT IN TABLE.  PUT IT IN.
00504    0 02 03665        LDA   KEY                    PLACE THE KEY IN FIRST.
00505    0 10 03020        JST   PUT
00506    0 02 00657        LDA   K                      THEN THE SYMBOL NUMBER.
00507    0 10 03020        JST   PUT
00510    0 02 03777        LDA   WORD                   FINALLY THE CURRENT LOCATION.
00511   -0 04 03673        STA*  T
00512    0 02 03774        LDA   DATA                   GET THE 'MORE' BIT AGAIN.
00513    0 03 00663        ANA   ='000400               ISOLATE IT.
00514   101040             SNZ                          ARE THERE MORE SYMBOLS.Q
00515    0 01 00446        JMP   NSDT                   NO.  GO TO NEXT DATA WORD.
00516    0 10 01515        JST   R24B                   YES.  READ NEXT WORD.
00517    0 10 01147        JST   EXT                    EXTRACT THE SYMBOL NUMBER.
00520    0 03 00672        ANA   ='017777               RETAIN ONLY 13 BITS.
00521    0 04 00657        STA   K
00522    0 01 00503        JMP   NSDN                   GO PUT IT INTO TABLE TOO.
                     *
                     *
                     *
                     *                 HERE FOR DAC, ADDRESS KNOWN.
                     *
00523    0 10 01147  NSD2  JST   EXT                    EXTRACT ADDRESS.
00524    0 04 03677        STA   VALU                   SAVE IT FOR RELOCATION.
00525    0 02 03774        LDA   DATA                   GET RELOCATION BITS.
00526    0414 75           LGL   3                      LEFT ADJUST.
```

```
00527    0 10 03177       JST   RELO              RELOCATE ADDRESS.
                     *
                     *    SAVE 'VALU' WHERE 'MMOD' IS ONE.  MERGE 'VALU' WITH 'DATA' WHERE
                     *    'MMOD' IS ZERO.  THIS ALWAYS MERGES THE FLAG BIT FROM 'DATA', AND
                     *     THE TAG BIT(IF NOT IN EXTENDED MODE) WITH 'VALU'.
                     *
00530    0 05 03774       ERA   DATA              **********************
00531    0 03 03774       ANA   DATA              VALU 0 0 0 0 1 1 1 1
00532   -0 04 03777       STA*  WORD              DATA 0 0 1 1 0 0 1 1
00533    0 05 03666       ERA   MMOD              MMOD 0 1 0 1 0 1 0 1
00534   -0 03 03777       ANA*  WORD              WORD 0 0 1 0 1 1 1 1   RESULT
00535    0 05 03677       ERA   VALU              **********************
00536   -0 04 03777       STA*  WORD              LOAD WORD.
00537    0 01 00446       JMP   NSDT              GO TO NEXT DATA WORD.
                     *
                     *
                     *
                     *              HERE FOR FORWARD-REFERENCE DAC.
                     *
00540    0 02 03774 NSD3  LDA   DATA              PLACE THE FLAG AND TAG BITS.
00541    0 03 00662       ANA   ='140000
00542   -0 04 03777       STA*  WORD
00543    0 10 01147 NSDM  JST   EXT               EXTRACT THE SYMBOL NUMBER.
00544    0 03 00672       ANA   ='017777          RETAIN ONLY 13 BITS.
00545    0 04 00657       STA   K
00546    0 10 02305       JST   GEN3              GET SPACE IN SYMBOL TABLE.
00547    0 02 00700       LDA   ='000001          KEY FOR UNDEFINED DAC.
00550    0 10 03020       JST   PUT               PLACE KEY IN SYMBOL TABLE.
00551    0 02 00657       LDA   K                 GET SYMBOL NUMBER.
00552    0 10 03020       JST   PUT               PLACE IT IN TABLE.
00553    0 02 03777       LDA   WORD              GET POINTER TO CURRENT LOCATION.
00554   -0 04 03673       STA*  T                 PLACE IT IN TABLE
00555    0 02 03774       LDA   DATA              GET THE 'MORE' BIT.
00556    0 03 00663       ANA   ='400             ISOLATE IT
00557    101040           SNZ                     ARE THERE MORE SYMBOLS.Q
00560    0 01 00446       JMP   NSDT              NO.  GO GET NEXT DATA WORD.
00561    0 10 01515       JST   R24B              YES.  READ NEXT SYMBOL.
```

```
00562  -0 02 03777       LDA*  WORD              GET PREVIOUS FLAG/TAG
00563   0 05 03774       ERA   DATA              OLD OR NEW FLAG/TAG, NOT BOTH
00564  -0 03 03777       ANA*  WORD              OLD FLAG/TAG WHERE NOT NEW
00565   0 05 03774       ERA   DATA              BOTH OLD AND NEW
00566   0 03 00662       ANA   ='140000          WIPE OUT LOW ORDER BITS
00567   0 01 00542       JMP   NSDM-1            *
                   *
                   *
                   *
                   *           HERE FOR NINE-BIT KNOWN ADDRESS.
                   *
00570   0 02 03774 NSD4  LDA   DATA              PLACE THE FLAG, TAG, AND
00571   0 03 00664       ANA   ='176000          OP-CODE INTO STORAGE.
00572  -0 04 03777       STA*  WORD
00573   0 10 01147       JST   EXT               EXTRACT 15-BIT VALUE.
00574   0 03 00661       ANA   ='077777
00575   0 04 03677       STA   VALU              SAVE IT FOR RELOCATION.
00576   0 02 03775       LDA   DATA+1            GET RELOCATION BITS.
00577   0414 73          LGL   5                 PLACE THEM PROPERLY.
00600   0 10 03177       JST   RELO              RELOCATE ADDRESS.
00601   0 10 02065       JST   DESE              DESECTOR WORD.
00602   0 01 00446       JMP   NSDT              DONE.  GO GET NEXT DATA WORD.
                   *
                   *
                   *
                   *           NS10    NEW STYLE BLOCK 10..  SYMBOL NUMBER DEFINITION.
                   *
00603   0 10 03024 NS10  JST   SKIP              IGNORE DEFINITIONS IF SKIPPING.
00604   0 10 01552 N10L  JST   LEFT              SEE IF TWO WORDS LEFT
00605   0 10 01552       JST   LEFT              IN INPUT BUFFER.
00606   0 12 00655       IRS   BUFP              YES.  MOVE TO FIRST.
00607  -0 02 00655       LDA*  BUFP              PICK UP SYMBOL NUMBER, FLAGS.
00610   0 04 03774       STA   DATA              SAVE FLAGS.
00611   0 03 00672       ANA   ='017777          ISOLATE SYMBOL NUMBER.
00612   0 04 00657       STA   K                 SAVE IT.
00613   0 12 00655       IRS   BUFP              MOVE TO VALUE.
00614  -0 02 00655       LDA*  BUFP              GET IT.
```

```
00615   0 04 03677        STA    VALU           SAVE IT FOR POSSIBLE RELOCATION.
00616   0 02 03774        LDA    DATA           GET RELOCATION BITS.
00617   0414 77           LGL    1              POSITION THEM.
00620   0 10 03177        JST    RELO           RELOCATE ADDRESS.
00621   0 10 02002        JST    SAVE           SAVE LOADER FOR POSSIBLE UNSTRING.
00622   0 10 02356        JST    SSS            SET UP SEARCH OF SYMBOL TABLE.
00623   0 02 00700        LDA    ='000001       SET KEY TO INDICATE SYMBOL NUMBER.
00624   0 04 03665        STA    KEY
00625   0 02 00674  N10M  LDA    ='000077       LOOK AT ALL ENTRIES OF THIS TYPE.
00626   0 10 02362        JST    SSN            MOVE TO NEXT ENTRY.
00627   0 01 00652        JMP    N10X           ALL ENTRIES PROCESSED.  GO EXIT.
00630   0 05 00657        ERA    K              IS THIS ENTRY A MATCH.Q
00631   100040            SZE                   SKIP IF MATCHES SYMBOL NUMBER.
00632   0 01 00625        JMP    N10M           NO.  CONTINUE HUNT.
00633   0 02 03774        LDA    DATA           YES.  NOW, IS SYMBOL
00634   101400            SMI                   STRINGABLE OR ONE-SHOT.
00635   0 01 00644        JMP    N10S           ONE-SHOT.  GO SERVICE IT.
00636   0 10 01620        JST    SETS           STRINGABLE.  SET BASE, MMOD.
00637   140040            CRA                   WIPE OUT THIS ENTRY
00640  -0 04 03672        STA*   SSSP           FROM THE SYMBOL TABLE.
00641   0 10 02646        JST    GET2           GET LAST LOCATION USED.
00642   0 10 02232        JST    UNST           RUN THE STRING.
00643   0 01 00625        JMP    N10M           GO BACK FOR MORE USAGES.
                    *
00644  -0 02 03672  N10S  LDA*   SSSP           ONE-SHOT SYMBOL NUMBER.
00645   0 06 00700        ADD    ='1            CHANGE ENTRY KEY
00646  -0 04 03672        STA*   SSSP           ADDRESS-ARITHMETIC TYPE.
00647   0 12 03672        IRS    SSSP           REPLACE SYMBOL NUMBER
00650   0 02 03677        LDA    VALU           BY VALUE.
00651  -0 04 03672        STA*   SSSP           *
00652   0 10 02055  N10X  JST    REST           RESTORE THE LOADER
00653   0 01 00604        JMP    N10L           GO GET NEXT SYMBOL NUMBER.
                    *
00654               BUFN  BSS    1              BUFFER COUNTER , FOR 'BLOK'.
00655               BUFP  BSS    1              BUFFER POINTER, FOR 'BLOK'.
00656               R1KN  BSS    1              BLOCK WORD COUNT, FOR 'R1K'
00657               K     BSS    1              STORAGE FOR SYMBOL NUMBER
```

```
1370 00660    0 000000   SW   DAC    **                                                      1370
1371                     *                                                                   1371
1372                     *                                                                   1372
1373 00661    077777          FIN                      END OF SECTOR.  FLUSH THE LITERALS.   1373
     00662    140000
     00663    000400
     00664    176000
     00665    100005
     00666    000003
     00667    177777
     00670    170377
     00671    000221
     00672    017777
     00673    141713
     00674    000077
     00675    000005
     00676    000011
     00677    000017
     00700    000001
     00701    000020
     00702    000037
     00703    177740
     00704    000034
     00705    120000
     00706    000223
     00707    141314
     00710    000002
     00711    000203
     00712    177704
1374          000713     STB0 EQU    *                 DESECTORING AREA NO. 1                1374
1375                     *                                                                   1375
1376                     *                                                                   1376
1377                     *                                                                   1377
1378                          EJCT                                                           1378
```

```
                                ORG   ORG+'1000           START NEW SECTOR.
                                SETB  STB1                DESECTOR IN CURRENT SECTOR.
                           *
     01000   000001        DFLT OCT   1                   DEFAULT INPUT SWITCH
                           *
                           *          NS14    NEW STYLE BLOCK 14..  END BLOCK.
                           *
     01001   0 02 03702    NS14 LDA   BUFF+2              END BLOCK.  GET JUMP ADDRESS.
     01002   0 04 03677         STA   VALU                SAVE FOR RELOCATION.
     01003   0 02 03340    N14J LDA   SYM                 STORE THE CURRENT SYMBOL TABLE LENGTH
     01004   0 04 03662         STA   BACK                AND COMMON COUNTER IN THE BACK-UP CELLS
     01005   0 02 03345         LDA   COMN                BECAUSE END CLEAN-UP WILL RESTORE THEM
     01006   0 04 03663         STA   BACK+1              FROM THERE.
     01007   0 02 03667    N14K LDA   MODE                JOIN HERE IF OLD-STYLE SKIPPED.
     01010   0 10 03153         JST   SENT                GO SET ENTRY LOCATION.
     01011   0 10 03056         JST   CLEN                PERFORM INTER-PROGRAM CLEAN-UP.
     01012   0 02 01012         LDA   *                   SET SKIP FLAG TO SKIP
     01013   0 04 03671         STA   SKIF                NEXT SUBPROGRAM.
     01014   0 01 00332         JMP   BLOK                GO READ ON.
                           *
                           *
                           *
                           *          NS24    NEW STYLE BLOCK 24..  SET RELOCATABLE MODE.
                           *
     01015   0 02 01705    NS24 LDA   ='040000            GET RELOCATABLE FLAG.
     01016   0 01 01020         JMP   NS30+1              SET RELOCATABLE FLAG. . .
                           *
                           *
                           *
                           *          NS30    NEW STYLE BLOCK 30..  SET ABSOLUTE MODE.
                           *
     01017   140040        NS30 CRA                       GET ABSOLUTE FLAG.
     01020   0 04 03667         STA   MODE                SET MODE OF LOADER.
     01021   0 01 00332         JMP   BLOK                EXIT TO READ NEXT BLOCK.
                           *
                           *
```

```
                          *
                          *           NS44    NEW STYLE 44..  SUBPROGRAM CALL.
                          *
01022    0 10 03024 NS44 JST   SKIP                     SKIP CALL BLOCKS IF NOT LOADING.
01023    0 02 03777      LDA   WORD                     BACK UP THE LOCATION COUNTER
01024    0 07 01707      SUB   ='000001                 TO THE LAST WORD LOADED.
01025    0 04 03777      STA   WORD                     IT IS THE CALL LOCATION.
01026    0 10 01556      JST   R48B                     READ THE SUBPROGRAM NAME.
01027    0 12 00655      IRS   BUFP                     MOVE TO DAC/NINE-BIT FLAG.
01030    140040          CRA                            INITIALIZE OFFSET
01031    0 04 03677      STA   VALU                     TO ZERO.
01032   -0 07 00655      SUB*  BUFP                     FORM DAC FLAG, '000000 MEANS
01033    0 04 01656      STA   DACF                     NINE-BIT, '177777 MEANS NEW STYLE DAC.
01034    0 10 02402      JST   CUT                      CUT A KEY FOR THIS CALL.
01035    0406 71         ARR   7                        MOVE NINE-BIT TO LOW ORDER.
01036    0 06 01656      ADD   DACF                     WIPE IT OUT IF WE ARE
01037    0416 71         ALR   7                        PROCESSING A DAC.
01040    0 05 01704 N44J ERA   ='100405                 UNLOADED SUBPROGRAM NAME.
01041    0 04 03665 N44K STA   KEY                      SAVE THE KEY.
01042    0 10 02356      JST   SSS                      SET UP FOR SYMBOL TABLE SEARCH.
01043    0 02 01703 N44L LDA   ='100077                 ACCEPT ANY SUBPROGRAM ENTRY.
01044    0 10 02362      JST   SSN                      MOVE TO NEXT ENTRY.
01045    0 01 01075      JMP   N44E                     NOT FOUND.  GO MAKE ENTRY.
01046    0 10 02412      JST   MACH                     COMPARE NAME WITH TABLE.
01047    0 01 01043      JMP   N44L                     NO MATCH.  CONTINUE HUNT.
01050    100040          SZE                            MATCH.  IS IT CALLED OR LOADED.Q
01051    0 01 01103      JMP   N44S                     CALLED.  GO STRING TO IT.
01052    0 02 03672      LDA   SSSP                     LOADED.
01053    0 06 01702      ADD   ='000004
01054    0 10 02642      JST   GET                      GET ENTRY LOCATION.
01055    0 06 03677      ADD   VALU                     BUMP BY OFFSET.
01056    0 04 03677      STA   VALU                     THAT IS TRUE ADDRESS.
01057    0 02 03664      LDA   FLAG                     ARE WE PROCESSING A 'BLOCK DATA'.Q
01060    101400          SMI                            SKIP IF NOT.
01061    0 01 01475      JMP   OS6R                     YES, RETURN TO SET COUNTER.
01062    0 02 01656      LDA   DACF                     NO.  RECOVER DAC FLAG.
01063    101040          SNZ                            IS IT A NINE-BIT OR A DAC.Q
```

```
01064   0 01 01071       JMP   N44N           NINE-BIT.  GO DESECTOR IT.
01065   0 03 03677       ANA   VALU           DAC.  WIPE VALUE TO SIZE.
01066  -0 06 03777       ADD*  WORD
01067  -0 04 03777       STA*  WORD           PLACE FINAL WORD.
01070   100000           SKP                  SKIP AROUND NINE-BIT PROCESSING.
01071   0 10 02065  N44N JST   DESE           NINE-BIT.  DESECTOR IT.
01072   0 12 03777  N44X IRS   WORD           RESTORE THE LOCATION COUNTER.
01073   0 10 02607       JST   FITS           CHECK IF LOAD FITS.
01074   0 01 00332       JMP   BLOK           GO READ NEXT BLOCK.
                    *
01075   0 10 02327  N44E JST   GEN6           MUST MAKE ENTRY.
01076   0 02 03665       LDA   KEY            GET SPACE IN TABLE.
01077   0 10 02434       JST   SPUT           PLACE KEY AND NAME.
01100   0 02 03777       LDA   WORD           PLACE VALUE.
01101  -0 04 03673       STA*  T
01102   0 01 01072       JMP   N44X           DONE.  GO BUMP COUNTER AND EXIT.
                    *
01103  -0 02 03672  N44S LDA*  SSSP           CALLED.  IN ORDER TO STRING, KEY MUST
01104   0 05 03665       ERA   KEY            MATCH EXACTLY.
01105   100040           SZE                  SKIP IF SO.
01106   0 01 01043       JMP   N44L           NO.  CONTINUE HUNT.
01107   0 02 03672       LDA   SSSP           YES.  GET LAST LOCATION
01110   0 06 01702       ADD   ='000004       IT WAS USED.
01111   0 10 02642       JST   GET
01112   0 04 03677       STA   VALU           FORM POINTER TO IT.
01113   0 02 03777       LDA   WORD           UPDATE TABLE ENTRY TO POINT
01114  -0 04 03673       STA*  T              HERE AS LAST LOC USED.
01115   0 10 02207       JST   DESS           DESECTOR FOR STRING.
01116   0 01 01072       JMP   N44X           BUMP COUNTER AND EXIT.
                    *
                    *
                    *
                    *          NS50   NEW STYLE BLOCK 50..  SUBPROGRAM DEFINITION.
                    *
01117   0 10 03024  NS50 JST   SKIP           IGNORE DEFINITION IF NOT LOADING..
01120   0 02 03777       LDA   WORD           SUBPROGRAM WILL BE DEFINED
01121   0 04 03677       STA   VALU           AS CURRENT LOCATION.
```

```
01122   0 10 02002      JST  SAVE          SAVE LOADER FOR POSSIBLE UNSTRING.
01123   0 02 01701      LDA  ='100005      GET KEY FOR LOADED SUBPROGRAM.
01124   0 04 03665      STA  KEY           SET IT UP.
01125   0 10 01556 N50L JST  R48B          READ FIRST NAME.
01126   0 10 02445      JST  DEFS          GO DEFINE IT.
01127   0 10 02055      JST  REST          RESTORE LOADER IN CASE DONE.
01130   0 01 01125      JMP  N50L          LOOP FOR ALL NAMES.
                   *
                   *
                   *
                   *           NEW STYLE BLOCK 54..  ENTER EXTEND-MODE DESECTORING.
                   *
01131   0 10 03024 NS54 JST  SKIP          IGNORE IF NOT LOADING
01132   0 02 01700      LDA  ='077777      GET EXTENDED MODE FLAG
01133   0 01 01136      JMP  NS60+2        READ ON . . .
                   *
                   *
                   *
                   *           NEW STYLE BLOCK 60..  LEAVE EXTEND-MODE DESECTORING.
                   *
01134   0 10 03024 NS60 JST  SKIP          IGNORE IF NOT LOADING
01135   0 02 01706      LDA  ='037777      GET LXD FLAG (NORMAL)
01136   0 04 03666      STA  MMOD          SET IT UP.
01137   0 01 00332      JMP  BLOK          EXIT TO READ NEXT BLOCK.
                   *
                   *
                   *
                   *           NEW STYLE BLOCK 64..  SET BASE SECTOR.
                   *
01140   0 10 03024 NS64 JST  SKIP          IGNORE IF NOT LOADING
01141   0 02 03702      LDA  BUFF+2        GET DESIRED BASE
01142   100400          SPL                SKIP IF ABSOLUTE.
01143   0 06 03670      ADD  PBRK          RELOCATE.  BUMP IT.
01144   0 10 02653      JST  MRG2          PBRK
01145   0 10 02015      JST  SETB          GO SET LOADER BASE SECTOR.
01146   0 01 00332      JMP  BLOK          EXIT TO NEXT BLOCK.
                   *
```

```
                         *
                         *
01147   0 000000   EXT   DAC    **              HERE TO EXTRACT SYMBOL VALUE
01150   0 02 03775       LDA    DATA+1          C(W)17-21 = V12-16 TO C(A)1-5.
01151   0404 65          LGR    11              NOW TO C(A)12-16.
01152   0 04 03673       STA    T               SAVE.
01153   0 02 03774       LDA    DATA            C(W)6-16 = V1-11 TO C(A)6-16.
01154   0414 73          LGL    5               NOW TO C(A)1-11.
01155   0 05 03673       ERA    T               COMBINE WITH OTHER HALF.
01156  -0 01 01147       JMP*   EXT             EXIT, ANSWER IN A-REGISTER.
                         *
                         *
                         *
                         * *************************************************************
                         * *                     OLD STYLE BLOCK ROUTINES                *
                         * *************************************************************
                         *
                         *
                         *
                         *               HERE TO PROCESS OLD STYLE BLOCKS.
                         *
01157   0 02 03700   OLD LDA    BUFF            OLD STYLE BLOCK.
01160   100400           SPL                    BLOCKS HIGHER THAN SEVEN
01161   0 01 00330       JMP    BLER            ARE ILLEGAL.
01162   0414 66          LGL    10              GET ADDRESS FIELD, BITS 3-8.
01163   0404 76          LGR    2               ISOLATE THEM.
01164   0 04 03673       STA    T               SAVE TEMPORARILY.
01165   0 02 03701       LDA    BUFF+1          GET ADDRESS FIELD, BITS 9-16.
01166   0404 70          LGR    8               POSITION THEM.
01167   0 05 03673       ERA    T               FORM COMPLETE ADDRESS.
01170   0 04 03677       STA    VALU            SAVE IT AWAY.
01171   0 02 03700       LDA    BUFF            BLOCK TYPE AGAIN.
01172   0404 64          LGR    12              POSITION RIGHT.
01173   0 10 03217       JST    SWCH            JUMP TO PROCESSOR FOR BLOCK.
01174   0 001204         DAC    OS0             0..  SPECIAL ACTION.
01175   0 001213         DAC    OS1             1..  ABSOLUTE DATA WORDS.
01176   0 001216         DAC    OS2             2..  RELOCATABLE DATA WORDS.
```

```
01177   0 001406          DAC   OS3            3..  ABSOLUTE END JUMP.
01200   0 001410          DAC   OS4            4..  RELOCATABLE END JUMP.
01201   0 001420          DAC   OS5            5..  SUBPROGRAM CALL.
01202   0 001433          DAC   OS6            6..  SUBPROGRAM OR COMMON DEFINITION.
01203   0 001500          DAC   OS7            7..  COMMON REFERENCE.
                    *
                    *
                    *
                    *             OLD STYLE BLOCK 0..  SPECIAL ACTION.
                    *
01204   0 02 03700  OS0   LDA   BUFF           BLOCK TYPE AGAIN.
01205   0404 72           LGR   6              POSITION SUB-TYPE RIGHT.
01206   0 10 03217        JST   SWCH           JUMP TO PROCESSOR.
01207   0 000330          DAC   BLER           0-0..  ILLEGAL.
01210   0 000410          DAC   FORS           0-1..  FORCE LOAD NEXT SUBPROGRAM.
01211   0 003031          DAC   EOT            0-2..  CHAIN.  TREAT LIKE EOT.
01212   0 003031          DAC   EOT            0-3..  END OF JOB.  TREAT LIKE EOT.
                    *
                    *
                    *
                    *             OLD STYLE BLOCK 1..  ABSOLUTE DATA BLOCK.
                    *
01213   0 10 03024  OS1   JST   SKIP           IGNORE DATA BLOCKS IF SKIPPING.
01214   0 02 03677        LDA   VALU           GET THE BLOCK ORIGIN.
01215   0 01 01221        JMP   OSDJ           GO SET IT UP AND PROCESS BLOCK.
                    *
                    *
                    *
                    *             OLD STYLE BLOCK 2..  RELOCATABLE DATA BLOCK.
                    *
01216   0 10 03024  OS2   JST   SKIP           IGNORE DATA BLOCKS IF SKIPPING.
01217   0 02 03677        LDA   VALU           GET THE BLOCK ORIGIN.
01220   0 06 03670        ADD   PBRK           RELOCATE IT.
01221   0 04 03777  OSDJ  STA   WORD           SET LOCATION COUNTER.
01222   0 02 03700        LDA   BUFF           BLOCK CONTROL WORD.
01223   0 03 01677        ANA   ='007700       ISOLATE WORD COUNT.
01224   0404 72           LGR   6              RIGHT ADJUST.
```

```
01225   0 05 01676        ERA   ='177777           -N-1.
01226   0 06 01675        ADD   ='000003           -N+2.
01227   0 04 00654        STA   BUFN               INITIALIZE BUFFER COUNTER.
01230   0 02 00374        LDA   BUFD =(DAC BUFF+1) OK.  SET UP BUFFER POINTER.
01231   0 04 00655        STA   BUFP
01232   0 10 02607        JST   FITS               CHECK VALIDITY OF ORIGIN.
01233   0 02 01674        LDA   ='100000           SET THE 'R24B' SWITCH ODD,
01234   0 04 01657        STA   R24S               SINCE OLD STYLE BLOCKS OTHER PARITY.
01235   0 10 01515 OSDL   JST   R24B               ALL SET.  READ A DATA WORD.
01236   0 02 03775        LDA   DATA+1             EXTRACT ADDRESS PORTION.
01237   0404 64           LGR   12                 GET BITS 13-16 OF ADDRESS.
01240   0 04 03677        STA   VALU               SAVE TEMPORARILY.
01241   0 02 03774        LDA   DATA               GET BITS 3-12 OF ADDRESS.
01242   0414 72           LGL   6                  ISOLATE THEM.
01243   0404 76           LGR   2                  POSITION THEM.
01244   0 05 03677        ERA   VALU               FORM WHOLE ADDRESS.
01245   0 04 03677        STA   VALU               SAVE IT AWAY.
01246   0 02 03775        LDA   DATA+1             GET WORD TYPE CODE.
01247   0404 70           LGR   8                  POSITION RIGHT.
01250   0 03 01673        ANA   ='000007           MASK TO SIZE.
01251   0 10 03217        JST   SWCH               JUMP TO PROCESSOR FOR TYPE.
01252   0 001262          DAC   OSD0               0..  ABSOLUTE DATA OR GENERIC.
01253   0 001267          DAC   OSD1               1..  ABSOLUTE NINE-BIT.
01254   0 001274          DAC   OSD2               2..  RELOCATABLE NINE-BIT.
01255   0 001300          DAC   OSD3               3..  NEG RELOC NINE-BIT.
01256   0 001306          DAC   OSD4               4..  STRING DEFINITION.
01257   0 001370          DAC   OSD5               5..  ABSOLUTE DAC.
01260   0 001376          DAC   OSD6               6..  RELOCATABLE DAC.
01261   0 001401          DAC   OSD7               7..  NEG RELOC DAC.
                   *
                   *
                   *
                   *            HERE FOR ABSOLUTE DATA OR GENERIC.
                   *
01262   0 02 03774 OSD0   LDA   DATA               GET 16-BIT WORD.
01263  -0 04 03777        STA*  WORD               PLACE IT.
01264   0 12 03777 OSDT   IRS   WORD               BUMP THE LOCATION COUNTER.
```

```
01265   0 10 02607        JST   FITS                CHECK IF WE STILL FIT.
01266   0 01 01235        JMP   OSDL                YES.  MOVE TO NEXT DATA WORD.
                     *
                     *
                     *
                     *                HERE FOR ABSOLUTE ADDRESS NINE-BIT.
                     *
01267   0 02 03774   OSD1 LDA   DATA                ABSOLUTE NINE-BIT.
01270   0 03 01672        ANA   ='176000            PLACE FLAG, TAG,
01271  -0 04 03777        STA*  WORD                AND OPERATION IN STORAGE.
01272   0 10 02065        JST   DESE                DESECTOR IT.
01273   0 01 01264        JMP   OSDT                GO TO NEXT DATA WORD.
                     *
                     *
                     *
                     *                HERE FOR RELOCATABLE ADDRESS NINE-BIT.
                     *
01274   0 02 03670   OSD2 LDA   PBRK                GET THE PROGRAM BREAK.
01275   0 06 03677        ADD   VALU                RELOCATE THE ADDRESS.
01276   0 04 03677        STA   VALU
01277   0 01 01267        JMP   OSD1                NOW HANDLE LIKE ABSOLUTE.
                     *
                     *
                     *
                     *                HERE FOR NEGATIVE RELOCATABLE NINE-BIT.
                     *
01300   0 02 03670   OSD3 LDA   PBRK                GET THE PROGRAM BREAK.
01301   0 06 03677        ADD   VALU                RELOCATE THE ADDRESS.
01302   0 05 01676        ERA   ='177777            ONE'S COMPLICATE IT.
01303   0 06 01707        ADD   ='000001            TWO'S COMPLICATE IT.
01304   0 04 03677        STA   VALU                SAVE COMPLICATED VALUE.
01305   0 01 01267        JMP   OSD1                NOW HANDLE LIKE ABSOLUTE.
                     *
                     *
                     *
                     *                HERE FOR FORTRAN STRING DEFINITION.
                     *
```

```
01306   0 10 02002 OSD4 JST   SAVE          SAVE THE LOADER LOCATION.
01307   0 02 03677      LDA   VALU          VALUE IN DATA WORD IS END
01310   0 06 03670      ADD   PBRK          OF STRING TO RUN.
01311   0 04 02663      STA   NEXT          RELOCATE IT AND SAVE IT.
01312   0 02 03777      LDA   WORD          CURRENT LOCATION IS THE
01313   0 04 03677      STA   VALU          VALUE OF THE STRING.
01314   0 02 02663      LDA   NEXT          GET NEXT LOCATION IN STRING.
01315   0 04 03777 OUSL STA   WORD          GO TO IT.
01316  -0 02 03777      LDA*  WORD          PICK UP WORD THERE.
01317   0 03 01671      ANA   ='001000      CHECK ITS SECTOR BIT.
01320   101040          SNZ
01321   0 01 01347      JMP   OUSZ          ZERO.  GO SERVICE.
01322  -0 02 03777      LDA*  WORD          SAME SECTOR.  FORM COMPLETE
01323   0 05 03777      ERA   WORD          ADDRESS BY JOINING INSTRUCTION
01324   0 03 01670      ANA   ='000777      ADDRESS AND LOCATION COUNTER.
01325   0 05 03777      ERA   WORD
01326   0 04 02663 OUSD STA   NEXT          SAVE NEXT LOCATION IN STRING.
01327  -0 02 03777      LDA*  WORD          SEE IF CURRENT LOCATION IS MEANT
01330   0 03 01667      ANA   ='036000      TO BE A DAC.
01331   100040          SZE                 DAC WILL HAVE ZERO OPERATION CODE.
01332   0 01 01341      JMP   OUSN          NO.  NINE-BIT.  GO DESECTOR IT.
01333   0 02 03677      LDA   VALU          YES.  DROP IN 14-BIT ADDRESS
01334  -0 05 03777      ERA*  WORD          BEHIND FLAG AND TAG.
01335   0 03 01706      ANA   ='037777
01336  -0 05 03777      ERA*  WORD
01337  -0 04 03777      STA*  WORD          PLACE FINAL WORD.
01340   100000          SKP                 GO RESUME STRING RUN.
01341   0 10 02065 OUSN JST   DESE          NINE-BIT.  DESECTOR THE CURRENT LOCATION.
01342   0 02 02663      LDA   NEXT          RESTORE POINTER TO NEXT LOCATION.
01343   100040          SZE                 CHECK FOR END OF STRING.
01344   0 01 01315      JMP   OUSL          NO.  CARRY ON DOWN STRING.
01345   0 10 02055      JST   REST          YES.  RESTORE TRUE LOCATION.
01346   0 01 01235      JMP   OSDL          GO TO NEXT DATA WORD.
                   *
01347  -0 02 03777 OUSZ LDA*  WORD          SECTOR BIT ZERO.
01350   0 03 01666      ANA   ='100777      IF ABSOLUTE ZERO AND NOT FLAGGED,
01351   101040          SNZ                 END OF STRING.
```

```
01352   0 01 01326       JMP   OUSD           YES.  GO WIND IT UP.
01353   0 05 02660       ERA   BASE           NO.  MUST BE BASE SECTOR
01354   0 03 01670       ANA   ='000777       REFERENCE.
01355   0 05 02660       ERA   BASE           FORM ADDRESS IN BASE SECTOR.
01356   0 10 02642       JST   GET            PICK UP POINTER THERE.
01357  -0 05 03777       ERA*  WORD           RECOVER FLAG (AND POSSIBLY TAG,
01360   0 03 03666       ANA   MMOD           DEPENDING UPON MEMORY MODE) BACK
01361  -0 05 03673       ERA*  T              TO OBJECT INSTRUCTION.
01362  -0 04 03777       STA*  WORD           INSTRUCTION WITH CORRECT FLAG AND TAG.
01363  -0 02 03673       LDA*  T              GET POINTER WORD AGAIN.
01364   0 05 03777       ERA   WORD           TACK ON HIGH-ORDER BIT FROM
01365   0 03 01706       ANA   ='037777       THE LOCATION COUNTER, WHICH WAS
01366   0 05 03777       ERA   WORD           LOST ON LOADING.
01367   0 01 01326       JMP   OUSD           IT IS NEXT LOCATION IN STRING.
                   *
                   *
                   *
                   *             HERE FOR ABSOLUTE DAC.
                   *
01370   0 02 03677 OSD5  LDA   VALU           ABSOLUTE DAC.
01371   0 05 03774 OD5J  ERA   DATA           DROP 14-BIT ADDRESS IN
01372   0 03 01706       ANA   ='037777       BEHIND FLAG AND TAG.
01373   0 05 03774       ERA   DATA
01374  -0 04 03777       STA*  WORD           PLACE WORD.
01375   0 01 01264       JMP   OSDT           GO ON THE NEXT DATUM.
                   *
                   *
                   *
                   *             HERE FOR RELOCATABLE DAC.
                   *
01376   0 02 03677 OSD6  LDA   VALU           RELOCATE ADDRESS.
01377   0 06 03670       ADD   PBRK
01400   0 01 01371       JMP   OD5J           NOW HANDLE LIKE ABSOLUTE.
                   *
                   *
                   *
                   *             HERE FOR NEGATIVE RELOCATABLE DAC.
```

```
                             *
01401    0 02 03670 OSD7 LDA    PBRK             RELOCATE ADDRESS.
01402    0 06 03677      ADD    VALU
01403    0 05 01676      ERA    ='177777         ONE'S COMPLICATE IT.
01404    0 06 01707      ADD    ='000001         TWO'S COMPLICATE IT.
01405    0 01 01371      JMP    OD5J             GO DROP IN COMPLICATED ADDRESS.
                    *
                    *
                    *
                    *                  OLD STYLE BLOCK 3..  ABSOLUTE END JUMP BLOCK.
                    *
01406    140040     OS3  CRA                     GET ABSOLUTE FLAG.
01407    0 01 01411      JMP    OS4J             GO JOIN COMMON PROCESSING.
                    *
                    *
                    *
                    *                  OLD STYLE BLOCK 4..  RELOCATABLE END JUMP BLOCK.
                    *
01410    0 02 01705 OS4  LDA    ='040000         GET RELOCATABLE FLAG.
01411    0 04 03667 OS4J STA    MODE             SAVE MODE FLAG.
01412    0 02 03671      LDA    SKIF             CHECK WHETHER LAST SUBPROGRAM.
01413    101040          SNZ                     WAS LOADED OR SKIPPED.
01414    0 01 01003      JMP    N14J             LOADED.  GO SAVE NEW SYMTAB AND COMMON.
01415    0 02 03670      LDA    PBRK             SKIPPED.  BACK UP THE LOCATION COUNTER.
01416    0 04 03777      STA    WORD
01417    0 01 01007      JMP    N14K             GO SET ENTRY LOCATION.
                    *
                    *
                    *
                    *                  OLD STYLE BLOCK 5..  SUBPROGRAM CALL BLOCK.
                    *
01420    0 10 03024 OS5  JST    SKIP             IGNORE CALL BLOCKS IF SKIPPING.
01421    0 02 03700      LDA    BUFF             GET THE INSTRUCTION WORD.
01422    0414 74         LGL    4                ADJUST CORRECTLY.
01423    0 03 01665      ANA    ='076000         RETAIN TAG AND OP-CODE.
01424   -0 04 03777      STA*   WORD             PLACE IN STORAGE.
01425    0 10 01574      JST    R6C              READ SUBPROGRAM NAME.
```

```
01426   140040            CRA              SET FLAGS FOR 'NS44'.
01427   0 04 01656        STA   DACF       INDICATE NINE-BIT REFERENCE.
01430   0 04 03677        STA   VALU       ZERO OUT THE OFFSET.
01431   0 10 02402        JST   CUT        FORM A KEY FOR CURRENT LOADER.
01432   0 01 01040        JMP   N44J       GO JOIN NEW STYLE PROCESSING.
                    *
                    *
                    *
                    *           OLD STYLE BLOCK 6..  SUBPROGRAM OR COMMON DEFINITION.
                    *
01433   0 10 01574  OS6   JST   R6C        READ SIX-CHARACTER NAME.
01434  -0 02 00655        LDA*  BUFP       GET FLAGS.
01435   0416 67           ALR   9          MOVE SUBPROGRAM BIT TO SIGN.
01436   100400            SPL              SUBPROGRAM OR COMMON.Q
01437   0 01 01451        JMP   ISCO       COMMON.  GO TO IT.
01440   0 02 03677        LDA   VALU       SUBPROGRAM.
01441   0 06 03670        ADD   PBRK       RELOCATE ENTRY POINT.
01442   0 04 03677        STA   VALU       SET VALUE FOR DEFINITION.
01443   0 02 01701        LDA   ='100005   GET KEY FOR DEFINED
01444   0 04 03665        STA   KEY        SUBPROGRAM.
01445   0 10 02002        JST   SAVE       SAVE LOADER FOR POSSIBLE UNSTRING.
01446   0 10 02445        JST   DEFS       GO DEFINE NAME.
01447   0 10 02055        JST   REST       RESTORE AFTER UNSTRING.
01450   0 01 00332        JMP   BLOK       EXIT TO NEXT BLOCK.
                    *
01451   0406 77     ISCO  ARR   1          MOVE 'BLOCK DATA' FLAG TO SIGN BIT.
01452   100400            SPL              SKIP IF NOT 'BLOCK DATA'.
01453   0 01 01464        JMP   OS6C       'BLOCK DATA'.  GO HANDLE IT.
01454   0 02 01664        LDA   ='100006   NOT 'BLOCK DATA'.  GET KEY FOR
01455   0 04 03665        STA   KEY        COMMON NAME.
01456   0 02 03345        LDA   COMN       GET COMMON COUNTER.
01457   0 07 03677        SUB   VALU       BUMP BY SIZE OF BLOCK.
01460   0 10 02653        JST   MRG2       *
01461   0 04 03677        STA   VALU       SET AS VALUE OF THIS BLOCK.
01462   0 10 02445        JST   DEFS       DEFINE NAME.
01463   0 01 00332        JMP   BLOK       EXIT TO NEXT BLOCK.
                    *
```

```
01464   0 02 03664 OS6C LDA   FLAG           HERE FOR 'BLOCK DATA'.
01465   101400          SMI                  IF WE HAVE ALREADY SAVED
01466   0 01 01471      JMP   *+3            THE PROGRAM BREAK,
01467   0 02 03670      LDA   PBRK           DO NOT SAVE IT AGAIN.
01470   0 04 03664      STA   FLAG           NOT YET.  SAVE IT, SET FLAG.
01471   140040          CRA                  SET THE SKIP FLAG TO LOAD THIS SUBPROGRAM.
01472   0 04 03671      STA   SKIF           ZERO MEANS LOAD.
01473   0 04 03677      STA   VALU           SET OFFSET TO ZERO
01474   0 01 01513      JMP   OS7J           EXIT TO LOOK UP COMMON NAME.
                   *
01475   0 02 03677 OS6R LDA   VALU           BACK HERE WHEN NAME FOUND.
01476   0 04 03670      STA   PBRK           SET COUNTER FOR COMING DATA.
01477   0 01 00332      JMP   BLOK           GO GET DATA.
                   *
                   *
                   *
                   *                 OLD STYLE BLOCK 7..  COMMON REFERENCE.
                   *
01500   0 10 03024 OS7  JST   SKIP           IGNORE COMMON BLOCKS IF SKIPPING.
01501   0 02 03700      LDA   BUFF           GET THE INSTRUCTION WORD.
01502   0414 74         LGL   4              LEFT JUSTIFY IT.
01503   0 03 01672      ANA   ='176000       ISOLATE FLAG, TAG, OP-CODE.
01504  -0 04 03777      STA*  WORD           PLACE IN STORAGE.
01505   0 03 01667      ANA   ='036000       ISOLATE INSTRUCTION ITSELF.
01506   100040          SZE                  ZERO SIGNIFIES DAC.
01507   0 02 01706      LDA   ='037777       SET DACF TO '000000 IF NINE-BIT,
01510   0 05 01706      ERA   ='037777       '037777 IF OLD STYLE DAC.
01511   0 04 01656      STA   DACF
01512   0 10 01574      JST   R6C            READ COMMON NAME.
01513   0 02 01664 OS7J LDA   ='100006       GET KEY FOR COMMON REFERENCE.
01514   0 01 01041      JMP   N44K
                   *
                   *
                   *
                   * ****************************************************************
                   * *                 READ FROM BLOCK ROUTINES                     *
                   * ****************************************************************
```

```
                          *
                          *
                          *
                          *    JST   R24B    READ 24 BITS.  THE NEXT 24 BITS OF THIS DATA BLOCK
                          *                  ARE TRANSFERRED TO LOCATIONS 'DATA' AND 'DATA+1'.
                          *                  IF THERE ARE NOT 24 BITS LEFT, THE RETURN WILL BE
                          *                  SAVED WHILE SUCCEEDING BLOCKS ARE READ.  WHEN A DATA
                          *                  BLOCK IS READ IN, ITS FIRST 24 BITS WILL BE SUPPLIED
                          *                  AND THE OLD RETURN TAKEN.
                          *
01515   0 000000     R24B DAC   **                      HERE TO READ 24 BITS
01516   0 02 01657        LDA   R24S                    PICK UP THE EVEN-ODD SWITCH.
01517   0 05 01674        ERA   ='100000                FLIP IT.
01520   100400            SPL                           IS THIS EVEN OR ODD CALL.Q
01521   0 01 01540        JMP   R24E                    EVEN.  GO TO IT.
01522   0 04 01657        STA   R24S                    ODD.  UPDATE SWITCH.
01523   0 10 01552        JST   LEFT                    MAKE SURE A WORD IS LEFT.
01524  -0 02 00655        LDA*  BUFP                    YES.  GET WORD.
01525   0414 70           LGL   8                       WANT LAST HALF THIS ONE.
01526   0 04 03774        STA   DATA                    SAVE PARTIAL WORD.
01527   0 12 00655        IRS   BUFP                    MOVE TO NEXT WORD.
01530  -0 02 00655        LDA*  BUFP                    PICK IT UP.
01531   0404 70           LGR   8                       NEXT EIGHT BITS RIGHT.
01532   0 05 03774        ERA   DATA                    COMBINE WITH FIRST EIGHT.
01533   0 04 03774        STA   DATA                    SAVE COMPLETE WORD.
01534  -0 02 00655        LDA*  BUFP                    GET LAST EIGHT BITS.
01535   0414 70           LGL   8                       POSITION THEM.
01536   0 04 03775   R24X STA   DATA+1                  SAVE SECOND WORD.
01537  -0 01 01515        JMP*  R24B                    EXIT.
                          *
01540   0 10 01552   R24E JST   LEFT                    EVEN CALL.
01541   0 10 01552        JST   LEFT                    MAKE SURE TWO WORDS LEFT.
01542   0 04 01657        STA   R24S                    OK.  UPDATE THE SWITCH SETTING.
01543   0 12 00655        IRS   BUFP                    MOVE TO FIRST WORD.
01544  -0 02 00655        LDA*  BUFP                    GET IT.
01545   0 04 03774        STA   DATA                    PLACE IT.
01546   0 12 00655        IRS   BUFP                    MOVE TO NEXT WORD.
```

```
01547  -0 02 00655        LDA*  BUFP               GET IT.
01550   0 03 01663        ANA   ='177400           WANT ONLY EIGHT BITS.
01551   0 01 01536        JMP   R24X               GO PLACE THEM AND EXIT.
                    *
01552   0 000000     LEFT DAC   **                 HERE TO SEE IF A WORD IS LEFT.
01553   0 12 00654        IRS   BUFN               BUMP THE BUFFER COUNT.
01554  -0 01 01552        JMP*  LEFT               OK.  WORD LEFT.
01555   0 01 00332        JMP   BLOK               NOT OK, MUST GO REFILL BUFFER.
                    *
                    *
                    *
                    *     JST   R48B    READ 48 BITS.  THE NEXT 48 BITS OF THIS BLOCK WILL
                    *                   BE TRANSFERRED TO LOCATIONS 'DATA'...'DATA+2'.  IF
                    *                   THERE ARE NOT 48 BITS LEFT, NO RETURN WILL BE MADE.
                    *                   INSTEAD, CONTROL WILL PROCEED TO 'BLOK' TO READ THE
                    *                   NEXT BLOCK.
                    *
01556   0 000000     R48B DAC   **                 HERE TO READ 48 BITS.
01557   0 10 01552        JST   LEFT               MAKE SURE THERE ARE THREE WORDS
01560   0 10 01552        JST   LEFT               LEFT IN THE BUFFER.
01561   0 10 01552        JST   LEFT
01562   0 12 00655        IRS   BUFP               OK.  MOVE TO FIRST WORD.
01563  -0 02 00655        LDA*  BUFP               GET IT.
01564   0 04 03774        STA   DATA               PLACE IT.
01565   0 12 00655        IRS   BUFP               NEXT WORD.
01566  -0 02 00655        LDA*  BUFP               GET IT.
01567   0 04 03775        STA   DATA+1             PLACE IT.
01570   0 12 00655        IRS   BUFP               LIKEWISE FOR LAST WORD.
01571  -0 02 00655        LDA*  BUFP
01572   0 04 03776        STA   DATA+2
01573  -0 01 01556        JMP*  R48B               RETURN TO CALLER.
                    *
                    *
                    *
                    *     JST   R6C     READ SUBPROGRAM OR COMMON NAME.  C(BUFF+1)9 THROUGH
                    *                   C(BUFF+4)8 ARE READ INTO 'DATA'...'DATA+2'.  'BUFP'
                    *                   IS LEFT POINTING TO 'BUFF+4'.
```

```
                              *
      01574    0 000000   R6C  DAC    **                HERE TO READ SIX CHARACTERS.
      01575    0 02 00374      LDA    BUFD =(DAC BUFF+1) SET UP BUFFER POINTER.
      01576    0 04 00655      STA    BUFP
      01577    0 02 01617      LDA    R6CD              SET UP POINTER TO 'DATA'.
      01600    0 04 03673      STA    T
      01601    0 10 01605      JST    R6C2              READ LAST TWO CHARACTERS.
      01602    0 10 01605      JST    R6C2              READ TWO MORE CHARACTERS.
      01603    0 10 01605      JST    R6C2              READ TWO CHARACTERS.
      01604   -0 01 01574      JMP*   R6C               EXIT TO CALLER.
                          *
                          *
                          *
      01605    0 000000   R6C2 DAC    **                READ TWO CHARACTERS, FOR 'R6C'.
      01606   -0 02 00655      LDA*   BUFP              GET CURRENT BUFFER WORD.
      01607    0414 70         LGL    8                 GET ITS SECOND CHARACTER.
      01610   -0 04 03673      STA*   T                 IT IS FIRST CHARACTER OF OUTPUT.
      01611    0 12 00655      IRS    BUFP              MOVE TO NEXT BUFFER WORD.
      01612   -0 02 00655      LDA*   BUFP              GET ITS FIRST CHARACTER.
      01613    0404 70         LGR    8                 POSITION RIGHT.
      01614   -0 05 03673      ERA*   T                 COMBINE WITH FIRST CHARACTER.
      01615    0 10 03020      JST    PUT               PLACE OUTPUT WORD.
      01616   -0 01 01605      JMP*   R6C2              EXIT TO 'R6C' AGAIN.
                          *
      01617    0 003774   R6CD DAC    DATA              POINTER TO 'DATA'.
                          *
                          *
                          *
                          * ************************************************************
                          * *                      SECTOR ROUTINES                     *
                          * ************************************************************
                          *
                          *
                          *
      01620    0 000000   SETS DAC    **                HERE TO SET BASE, MMOD BY KEY.
      01621   -0 02 03672      LDA*   SSSP              GET OLD BASE, MMOD FLAG.
      01622    0414 70         LGL    8                 POSITION MMOD FLAG.
```

```
01623    0 03 01705         ANA    ='040000           ISOLATE IT.
01624    0 05 01700         ERA    ='077777           FLIP IT TO PROPER FORM.
01625    0 04 03666         STA    MMOD               SET MEMORY MODE.
01626   -0 02 03672         LDA*   SSSP               RESTORE KEY AGAIN.
01627    0 03 01662         ANA    ='000200           ISOLATE DAC/NINE-BIT INDICATOR.
01630    0 05 01662         ERA    ='000200           FLIP IT TO PROPER FORM.
01631    0 04 01656         STA    DACF               SET DACF NON-ZERO IF A DAC.
01632   -0 02 03672         LDA*   SSSP               GET BASE AGAIN.
01633    0 03 01661         ANA    ='77000            *
01634    0 10 02015         JST    SETB               SET BASE SECTOR.
01635   -0 01 01620         JMP*   SETS               EXIT TO CALLER.
                       *
                       *
                       *
                       *    JST    TYP1               TYPE CHARACTER IN C(A-REGISTER) 9-16,
                       *                              THEN INTERCHANGE CHARACTERS IN A-REGISTER
                       *
01636    0 000000     TYP1  DAC    **                 TYPE C(A-REG. 9-16 ON ASR-33 OR 35
01637    74 0004            OTA    '4                 OUTPUT RIGHT HALF A REG.
01640    0 01 01637         JMP    *-1                *
01641    0406 70            ARR    8                  INTERCHANGE CHARACTERS
01642   -0 01 01636         JMP*   TYP1               EXIT
                       *
                       *
                       *    JST    TYP2               TYPE THE TWO CHARACTERS IN THE A REG.
                       *
                       *
01643    0 000000     TYP2  DAC    **                 HERE TO TYPE TWO CHARS IN A REG.
01644    0406 70            ARR    8                  INTERCHANGE CHARACTERS.
01645    0 10 01636         JST    TYP1               TYPE THE FIRST ONE, POSITION SECOND
01646    0 10 01636         JST    TYP1               TYPE THE SECOND ONE
01647   -0 01 01643         JMP*   TYP2               EXIT
                       *
                       *
                       *
                       *    JST    CRLF               TYPE A CARRIAGE-RETURN LINE-FEED
                       *
```

```
01650    0 000000  CRLF DAC    **            TYPE NEW LINE ON ASR-33/35
01651    0 02 01660     LDA    ='106612      C/R - L/F
01652    0 10 01643     JST    TYP2          TYPE THEM
01653    140040         CRA                  NULL CHARACTER
01654    0 10 01636     JST    TYP1          TYPE NULL FOR RETURN DELAY
01655   -0 01 01650     JMP*   CRLF          EXIT
                   *
                   *
                   *
                   *
01656              DACF BSS    1             DAC FLAG, FOR 'NS44', 'SETS', AND 'USTR'.
01657              R24S BSS    1             EVEN/ODD SWITCH, FOR 'R24B'
                   *
                   *
                   *
01660    106612         FIN                  END OF SECTOR.  FLUSH THE LITERALS.
01661    077000
01662    000200
01663    177400
01664    100006
01665    076000
01666    100777
01667    036000
01670    000777
01671    001000
01672    176000
01673    000007
01674    100000
01675    000003
01676    177777
01677    007700
01700    077777
01701    100005
01702    000004
01703    100077
01704    100405
01705    040000
```

```
01706   037777
01707   000001

        001710    STB1 EQU   *            DESECTORING AREA NO. 2
                  *
                  *
                  *
                       EJCT
```

```
                              ORG    ORG+'2000      START NEW SECTOR.
02000    0 003700     FIL7 DAC    COMM
02001    0 000125     DAC1 DAC    BOT+3          POINTER TO SYMBOL TALBE
                           SETB   STB2           DESECTOR IN CURRENT SECTOR. .
                      *
                      *
02002    0 000000     SAVE DAC    **             HERE TO PREPARE FOR UNSTRINGING.
02003    0 02 03777        LDA    WORD           SAVE THE LOCATION COUNTER.
02004    0 04 02012        STA    SAVL           *
02005    0 02 03666        LDA    MMOD           SAVE THE MEMORY MODE.
02006    0 04 02013        STA    SAVL+1         *
02007    0 02 02660        LDA    BASE           SAVE THE CURRENT BASE SECTOR.
02010    0 04 02014        STA    SAVL+2         *
02011   -0 01 02002        JMP*   SAVE           EXIT TO CALLER.
02012                 SAVL BSS    3              SPACE TO SAVE LOADER, FOR 'SAVE'
                      *
02015    0 000000     SETB DAC    **             HERE TO CHANGE BASE SECTOR.
02016    0 04 02660        STA    BASE           NEW BASE ORIGIN.
02017    0 03 02706        ANA    ='077000       ISOLATE BASE SECTOR NUMBER.
02020    0 06 02707        ADD    ='000004       FORM BASE INFORMATION KEY.
02021    0 04 02661        STA    BASE+1         SAVE IT TEMPORARILY.
02022    0 02 03340        LDA    SYM            SET UP FOR A SEARCH OF THE
02023    0 04 03673        STA    T              SYMBOL TABLE.
02024    0 02 03673   SETL LDA    T              MOVE TO FIRST OR NEXT ENTRY.
02025    0 10 02341        JST    NEXE           IF TABLE IS EXHAUSTED, GO
02026    0 01 02042        JMP    SETN           CREATE NEW ENTRY FOR THE SECTOR.
02027    0 10 02642        JST    GET            GET ENTRY KEY.
02030    0 05 02661        ERA    BASE+1         IS IT THE KEY WE WANT.Q
02031    100040            SZE
02032    0 01 02024        JMP    SETL           NO.  CONTINUE SEARCH.
02033    0 12 03673        IRS    T              YES.  MOVE TO FIRST WORD POINTER.
02034   -0 02 03673        LDA*   T              GET START OF INDIRECT TABLE.
02035    0 04 02660        STA    BASE           SET IT UP IN CURRENT BASE.
02036    0 12 03673        IRS    T              MOVE TO COUNT.
02037    0 02 03673   SETJ LDA    T
02040    0 04 02661        STA    BASE+1         REMEMBER LOCATION OF THE COUNT.
```

```
02041  -0 01 02015        JMP*  SETB          EXIT TO CALLER.
                    *
02042   0 02 02660  SETN  LDA   BASE          PICKUP BASE SECTOR
02043   101040            SNZ                 *
02044  -0 01 02015        JMP*  SETB          IT'S ALREADY SET - EXIT
02045   0 10 02305        JST   GEN3          FIRST TIME THIS SECTOR USED
02046   0 02 02661        LDA   BASE+1        CREATE AN ENTRY FOR IT IN TABLE.
02047   0 10 03020        JST   PUT           PLACE THE KEY.
02050   0 02 02660        LDA   BASE          PLACE THE ORIGIN OF THE TABLE.
02051   0 10 03020        JST   PUT
02052   0 02 02705        LDA   ='177777      INITIALIZE (-COUNT-1) TO SAY
02053  -0 04 03673        STA*  T             NO ENTRIES MADE.
02054   0 01 02037        JMP   SETJ          GO COPY TABLE ENTRY, AND EXIT.
                    *
                    *
                    *
02055   0 000000    REST  DAC   **            HERE TO RESTORE AFTER UNSTRINGING.
02056   0 02 02012        LDA   SAVL          RESTORE THE LOCATION COUNTER.
02057   0 04 03777        STA   WORD
02060   0 02 02013        LDA   SAVL+1        RESTORE THE MEMORY MODE.
02061   0 04 03666        STA   MMOD
02062   0 02 02014        LDA   SAVL+2        RESTORE THE BASE SECTOR.
02063   0 10 02015        JST   SETB
02064  -0 01 02055        JMP*  REST          EXIT TO CALLER.
                    *
                    *
                    *
02065   0 000000    DESE  DAC   **            HERE TO DESECTOR 'WORD' TO 'VALU'.
02066   0 02 03677        LDA   VALU          PICK UP DESIRED ADDRESS.
02067   0 04 03674        STA   T+1           ASSUME IT WILL BE DIRECT ADDRESS.
02070   0 05 03777        ERA   WORD          COMPARE WITH LOC OF INSTRUCTION.
02071   0 03 02706        ANA   ='077000      ISOLATE THE SECTOR NUMBERS.
02072   0 03 03666        ANA   MMOD
02073   101040            SNZ                 IS ADDRESS IN SAME SECTOR.Q
02074   0 01 02150        JMP   DESC          YES.  GO HANDLE IT.
02075   0 02 03677        LDA   VALU          NO.  MAYBE ADDRESS IS IN THE
02076   0 03 02706        ANA   ='077000
```

```
02077    100040            SZE
02100    0 05 02660        ERA   BASE          CURRENT BASE SECTOR.
02101    0 03 02706        ANA   ='077000      ISOLATE SECTOR NUMBERS.
02102    101040            SNZ                 IS ADDRESS IN BASE SECTOR.Q
02103    0 01 02145        JMP   DESB          YES.  GO SERVICE IT.
02104    0 02 02660        LDA   BASE          PICK UP BASE SECTOR
02105    100040            SZE                 *
02106    0 01 02112        JMP   *+4           *
02107    0 02 02704        LDA   ='100         SET TO DEFAULT
02110    0 10 02653        JST   MRG2          *
02111    0 10 02015        JST   SETB          SET THE BASE
02112   -0 02 03777        LDA*  WORD          NO.  MUST CREATE INDIRECT WORD.
02113    0 05 02703        ERA   ='032000      CHECK FOR OBJECT INSTRUCTION
02114    0 03 02702        ANA   ='036000      'LOAD-INDEX', WHICH MUST BE
02115    101040            SNZ                 INDIRECTED DIFFERENTLY.
02116    0 01 02135        JMP   DESL          YES.  GO SERVICE 'LDX'.
02117    0 02 03677        LDA   VALU          NO.  GET TARGET ADDRESS.
02120   -0 05 03777        ERA*  WORD          COMBINE WITH FLAG (AND ALSO
02121    0 03 03666        ANA   MMOD          TAG IF 16K MEMORY MODE) OF OP.
02122   -0 05 03777        ERA*  WORD          CREATE INDIRECT WORD.
02123    0 04 03673        STA   T             SAVE IT TEMPORARILY.
02124    0 02 03666        LDA   MMOD          TURN ON THE INDIRECT BIT OF
02125   -0 03 03777  DESF  ANA*  WORD          THE OBJECT WORD (AND ALSO
02126    0 05 02241        ERA   SIGN          TURN OFF ITS TAG IN 16K MODE
02127   -0 04 03777        STA*  WORD          REPLACE CORRECTED WORD.
02130    0 02 02660        LDA   BASE          PREPARE FOR SEARCH OF
02131    0 04 03674        STA   T+1           BASE SECTOR INDIRECT TABLE.
02132   -0 02 02661        LDA*  BASE+1        SET POINTER AND COUNTER.
02133    0 04 03675        STA   T+2
02134    0 01 02165        JMP   DEST          GO ENTER SEARCH SAFE LOOP.
                     *
02135    0 02 03677  DESL  LDA   VALU          HERE TO HANDLE 'LDX' CASE.
02136    0 03 03666        ANA   MMOD          TAG DOES NOT MOVE, EVEN IN
02137   -0 05 03777        ERA*  WORD          16K MODE, BUT ADDRESS MUST
02140    0 03 02701        ANA   ='077777      BE TRUNCATED TO 14 BITS IF SO.
02141   -0 05 03777        ERA*  WORD          CREATE INDIRECT WORD.
02142    0 04 03673        STA   T             SAVE IT AWAY.
```

```
02143    0 02 02701       LDA    ='077777         SET MASK TO RETAIN TAG IN OP.
02144    0 01 02125       JMP    DESF             GO JOIN STANDARD PROCESSING.
                    *
02145    0 02 03674 DESB  LDA    T+1              DESECTOR TO BASE SECTOR.
02146    0 03 02700       ANA    ='000777         WIPE ADDRESS TO NINE BITS.
02147    0 01 02153       JMP    DESJ             SECTOR BIT OFF.
                    *
02150    0 02 03674 DESC  LDA    T+1              DESECTOR TO CURRENT SECTOR.
02151    0 03 02700       ANA    ='000777         WIPE ADDRESS TO NINE BITS,
02152    0 05 02677       ERA    ='001000         SECTOR BIT ON.
02153   -0 05 03777 DESJ  ERA*   WORD             DROP INTO OBJECT WORD.
02154    0 03 02676       ANA    ='001777
02155   -0 05 03777       ERA*   WORD
02156   -0 04 03777       STA*   WORD             REPLACE UPDATED WORD.
02157   -0 01 02065       JMP*   DESE             EXIT TO CALLER.
                    *
02160   -0 02 03674 DESU  LDA*   T+1              CONTINUE SEARCH OF INDIRECT TABLE.
02161    0 05 03673       ERA    T                GET NEXT ENTRY.
02162    101040           SNZ                     IS IT SAME AS THE ONE WE WANT.Q
02163    0 01 02201       JMP    DESG             YES.  GO USE IT OVER.
02164    0 12 03674       IRS    T+1              NO.  BUMP COUNTERS.
02165    0 12 03675 DEST  IRS    T+2
02166    0 01 02160       JMP    DESU             CONTINUE SEARCH.
02167    0 02 03674       LDA    T+1              TABLE EXHAUSTED, GET NEXT
02170    0 05 02660       ERA    BASE             AVAILABLE LOCATION.
02171    0 03 02706       ANA    ='077000         CHECK THAT IT IS STILL IN
02172    100040           SZE                     THE BASE SECTOR.
02173    0 10 02636       JST    MO               NO.  BASE SECTOR OVERFLOW.
02174    0 02 03673       LDA    T                YES.  PLACE THE INDIRECT
02175   -0 04 03674       STA*   T+1              WORD IN IT.
02176   -0 02 02661       LDA*   BASE+1           INCREMENT THE BASE SECTOR
02177    0 06 02705       ADD    ='177777         USAGE COUNT
02200   -0 04 02661       STA*   BASE+1
02201    0 02 03777 DESG  LDA    WORD             SEE IF THE CURRENT SECTOR
02202    0 05 02660       ERA    BASE             IS THE BASE SECTOR.
02203    0 03 02706       ANA    ='077000
02204    100040           SZE
```

```
02205    0 01 02145        JMP   DESB              NO.  DESECTOR WITHOUT SECTOR BIT.
02206    0 01 02150        JMP   DESC              YES.  DESECTOR WITH SECTOR BIT.
                      *
                      *
                      *
02207    0 000000     DESS DAC   **                HERE TO DESECTOR FOR STRING.
02210    0 02 02207        LDA   *-1               SET UP FOR A RETURN.
02211    0 04 02065        STA   DESE              THROUGH 'DESE'.
02212    0 02 03677        LDA   VALU              SET ASSUMED DIRECT ADDRESS.
02213    0 04 03674        STA   T+1               IS TARGET ADDRESS IN SAME
02214    0 05 03777        ERA   WORD              SECTOR AS INSTRUCTION.Q
02215    0 03 02706        ANA   ='077000          ISOLATE SECTOR NUMBERS.
02216    101040            SNZ
02217    0 01 02150        JMP   DESC              SAME SECTOR.  GO SERVICE IT.
02220    0 10 02305        JST   GEN3              NO.  MUST CREATE POINTER WORD.
02221    0 02 02675        LDA   ='000003          GET SPACE IN SYMBOL TABLE.
02222    0 10 03020        JST   PUT               INSERT POINTER WORD KEY.
02223    0 02 03777        LDA   WORD              INSERT POINTER BACK TO CURRENT
02224    0 10 03020        JST   PUT               LOCATION.
02225    0 02 03677        LDA   VALU              FINALLY GET DATA POINTER.
02226    0 03 02701        ANA   ='077777          MAKE SURE IT IS ONLY 15-BITS.
02227   -0 04 03673        STA*  T                 PLACE IT.
02230    0 02 02700        LDA   ='000777          FLAG STRING POINTER BY ADDRESS
02231    0 01 02153        JMP   DESJ              OF ABSOLUTE '00777.
                      *
                      *
                      *
02232    0 000000     UNST DAC   **                HERE TO UNSTRING TO 'VALU'.
02233    0 10 02247   UNSL JST   WALK              GENERATE NEXT LOCATION IN STRING.
02234    0 04 02663        STA   NEXT              SAVE IT.
02235    0 02 01656        LDA   DACF              ARE WE WORKING DACS OR NINE-BITS.Q
02236    101040            SNZ                     SKIP IF DAC.
02237    0 01 02242        JMP   UNSN              NINE-BIT.  GO DESECTOR IT.
02240    0 10 03165        JST   PROP              *
                      * THE NEXT WORD IS USED AS BOTH AN INST. AND CONST.
02241    100000       SIGN SKP                     SKIP NINE BIT PROCESSING
02242    0 10 02065   UNSN JST   DESE              DESECTOR CURRENT WORD.
```

```
02243    0 02 02663       LDA   NEXT          GET NEXT LOCATION.
02244    100040           SZE                 CHECK FOR END OF STRING.
02245    0 01 02233       JMP   UNSL          NO.  GO CONTINUE SEARCH.
02246   -0 01 02232       JMP*  UNST          YES.  EXIT TO CALLER.
                    *
                    *
                    *
02247    0 000000   WALK  DAC   **            HERE TO MOVE DOWN STRING ONE POSITION.
02250    0 04 03777       STA   WORD          SAVE NEXT LOCATION IN STRING.
02251   -0 02 03777       LDA*  WORD          GET THE WORD.
02252    0 03 02677       ANA   ='001000      CHECK THE SECTOR BIT.
02253    101040           SNZ                 DOES IT POINT TO THE SAME SECTOR.Q
02254    0 01 02262       JMP   UNSZ          NO.  JUMP.
02255   -0 02 03777       LDA*  WORD          YES.  FORM COMPLETE ADDRESS
02256    0 05 03777       ERA   WORD          BY WIPING IN LOCATION ADDRESS.
02257    0 03 02700       ANA   ='000777
02260    0 05 03777       ERA   WORD
02261   -0 01 02247       JMP*  WALK          RESULT IS NEXT LOCATION.  EXIT.
                    *
02262   -0 02 03777 UNSZ  LDA*  WORD          SECTOR BIT ZERO.
02263    0 03 02700       ANA   ='000777      IF ADDRESS IS COMPLETELY ZERO,
02264    101040           SNZ                 THIS IS END OF STRING.
02265   -0 01 02247       JMP*  WALK          END.  GIVE FINAL EXIT.
02266    0 02 03340       LDA   SYM           NOT END.  MUST WORK THROUGH
02267    0 04 03674       STA   T+1           POINTER IN SYMBOL TABLE.
02270    0 02 03674 UNSK  LDA   T+1           SET UP SEARCH OF TABLE.
02271    0 10 02341       JST   NEXE          MOVE TO NEXT ENTRY.
02272    0 10 02636       JST   MO            NOT FOUND.  STRING WAS OVERWRITTEN.
02273    0 04 03674       STA   T+1           SAVE POINTER.
02274    0 07 02705       SUB   ='177777      MOVE TO BACK POINTER
02275    0 10 02642       JST   GET           GET IT.
02276    0 05 03777       ERA   WORD          DOES IT POINT TO US.Q
02277    100040           SZE                 SKIP IF SO.
02300    0 01 02270       JMP   UNSK          NO.  RESUME SEARCH.
02301   -0 04 03674       STA*  T+1           YES.  WIPE OUT ENTRY.
02302    0 12 03673       IRS   T             MOVE TO DATA POINTER.
02303   -0 02 03673       LDA*  T             GET IT.
```

```
02304  -0 01 02247        JMP*  WALK                IT WILL BE NEXT STRING LOCATION.
                   *
                   *
                   *
                   * ******************************************************************
                   * *                     SYMBOL TABLE ROUTINES                      *
                   * ******************************************************************
                   *
                   *
                   *
02305   0 000000   GEN3 DAC   **                    HERE TO OBTAIN 3-WORD ENTRY.
02306   0 02 03340        LDA   SYM                 SET UP POINTER FOR SEARCH
02307   0 04 03673        STA   T                   OF SYMBOL TABLE.
02310   0 02 03673 GENL LDA   T                     MOVE TO THE FIRST OR
02311   0 10 02341        JST   NEXE                NEXT ENTRY.
02312   0 01 02317        JMP   GENE                TABLE EXHAUSTED.  GO EXTEND IT.
02313   0 10 02642        JST   GET                 SAVE POINTER, GET KEY FOR THIS ENTRY.
02314   100040            SZE                       CHECK FOR AVAILABLE SLOT.
02315   0 01 02310        JMP   GENL                NO.  CONTINUE HUNT.
02316  -0 01 02305        JMP*  GEN3                YES.  EXIT TO CALLER, ANSWER IN 'T'.
                   *
02317   0 02 03340 GENE LDA   SYM                   NEED TO EXTEND TABLE.
02320   0 04 03673        STA   T                   SAVE POINTER TO NEW SLOT.
02321   0 07 02675        SUB   ='000003            EXTEND TABLE BY THREE WORDS.
02322   0 04 03340        STA   SYM                 SAVE NEW TABLE INDEX.
02323   0 10 02607        JST   FITS                SEE IF BIGGER TABLE FITS IN CORE.
02324   140040            CRA                       YES.  WIPE OUT EXTENDED
02325  -0 04 03340        STA*  SYM                 AREA KEY WORD.
02326  -0 01 02305        JMP*  GEN3                RETURN TO CALLER.
                   *
                   *
                   *
02327   0 000000   GEN6 DAC   **                    HERE TO OBTAIN 6-WORD ENTRY.
02330   0 02 03340        LDA   SYM                 GET THE TABLE INDEX.
02331   0 07 02675        SUB   ='000003            EXTEND IT BY 3 WORDS.
02332   0 04 03673        STA   T                   SAVE POINTER FOR CALLER.
02333   0 07 02675        SUB   ='000003            EXTEND THE TABLE A TOTAL OF 6 WORDS.
```

```
02334   0 04 03340        STA   SYM            SAVE NEW TABLE INDEX.
02335   0 10 02607        JST   FITS           SEE IF TABLE FITS IN CORE.
02336   140040            CRA                  CLEAR OUT KEY WORD OF
02337  -0 04 03340        STA*  SYM            NEW AREA.
02340  -0 01 02327        JMP*  GEN6           RETURN TO CALLER.
                    *
                    *
                    *
02341   0 000000   NEXE   DAC   **             HERE TO ADVANCE POINTER TO NEXT ENTRY.
02342   0 10 02642        JST   GET            PICK UP KEY WORD, STORE POINTER.
02343   0 03 02241        ANA   SIGN           CHECK 3/6 INDICATOR, SIGN BIT
02344   100400            SPL                  PLUS MEANS 3 WORDS, MINUS MEANS 6.
02345   0 02 02675        LDA   ='000003       SIX.  MOVE AN EXTRA THREE.
02346   0 06 03673        ADD   T              BUMP THE POINTER.
02347   0 07 02354        SUB   NEX3           COMPARE WITH END OF TABLE.
02350   100400            SPL                  SKIP IF EXHAUSTED.
02351   0 12 02341        IRS   NEXE           OK.  INCREMENT RETURN.
02352   0 06 02355        ADD   NEX0           RESTORE POINTER TO NEXT ENTRY.
02353  -0 01 02341        JMP*  NEXE           RETURN TO CALLER.
                    *
02354   0 000125   NEX3   DAC   BOT+3          BES OF SYMBOL TABLE MINUS THREE.
02355   0 000130   NEX0   DAC   BOT+6          BES OF SYMBOL TABLE.
                    *
                    *
                    *
02356   0 000000   SSS    DAC   **             HERE TO SET UP SYMBOL TABLE SEARCH.
02357   0 02 03340        LDA   SYM            GET SYMBOL TABLE POINTER.
02360   0 04 03672        STA   SSSP           SAVE IT FOR SEARCH.
02361  -0 01 02356        JMP*  SSS            RETURN TO CALLER.
                    *
                    *
                    *
02362   0 000000   SSN    DAC   **             HERE TO LOOK AT NEXT ENTRY.
02363   0 04 03674        STA   T+1            SAVE KEY MASK.
02364   0 02 03672 SSNL   LDA   SSSP           GET THE SEARCH POINTER.
02365   0 10 02341        JST   NEXE           ADVANCE TO NEXT ENTRY.
02366  -0 01 02362        JMP*  SSN            END OF TABLE.  JST+1 RETURN.
```

```
02367  0 04 03672        STA   SSSP               SAVE ADVANCED POINTER.
02370  0 10 02642        JST   GET                GET KEY.
02371  0 05 03665        ERA   KEY                COMPARE WITH DESIRED KEY.
02372  0 03 03674        ANA   T+1                ONLY IN BITS DESIRED.
02373  100040            SZE                      IS IT A MATCH.Q
02374  0 01 02364        JMP   SSNL               NO.  GO TO NEXT ENTRY.
02375  0 02 03672        LDA   SSSP               YES.  GET POINTER.
02376  0 07 02705        SUB   ='177777           MOVE TO FIRST DATA WORD
02377  0 10 02642        JST   GET                GET IT IN A-REGISTER.
02400  0 12 02362        IRS   SSN                JST+2 RETURN.
02401 -0 01 02362        JMP*  SSN                EXIT TO CALLER.
                   *
                   *
                   *
02402  0 000000    CUT   DAC   **                 HERE TO FORM CURRENT KEY.
02403  0 02 03666        LDA   MMOD               GET THE MEMORY MODE FLAG.
02404  0 05 02705        ERA   ='177777           FLIP, SO '100000 = 32K, '140000 = 16K.
02405  0404 70           LGR   8                  POSITION TO BIT 10, SET BIT 9.
02406  0 05 02660        ERA   BASE               INSERT CURRENT BASE SECTOR.
02407  0 03 02674        ANA   ='100777           LEAVE TYPE-CODE OPEN.
02410  0 05 02660        ERA   BASE
02411 -0 01 02402        JMP*  CUT                EXIT, RESULT IN A-REGISTER.
                   *
                   *
                   *
02412  0 000000    MACH  DAC   **                 HERE TO COMPARE SIX-CHAR NAMES.
02413  0 05 03774        ERA   DATA               CHECK FIRST WORD.
02414  100040            SZE
02415 -0 01 02412        JMP*  MACH               NOT THE SAME. EXIT.
02416  0 10 02646        JST   GET2               GET NEXT WORD.
02417  0 05 03775        ERA   DATA+1
02420  100040            SZE
02421 -0 01 02412        JMP*  MACH               NOT THE SAME.  EXIT.
02422  0 02 03672        LDA   SSSP
02423  0 06 02675        ADD   ='000003
02424  0 10 02642        JST   GET                GET THE THIRD WORD.
02425  0 05 03776        ERA   DATA+2
```

```
02426   100040          SZE
02427  -0 01 02412      JMP*  MACH         NOT THE SAME.  EXIT.
02430   0 12 02412      IRS   MACH         OK.  MATCHES.  INCREMENT RETURN.
02431  -0 02 03672      LDA*  SSSP         RETURN WITH LOAD/NOLOAD
02432   0 03 02673      ANA   ='000400     BIT IN A-REGISTER.
02433  -0 01 02412      JMP*  MACH
                  *
                  *
                  *
02434   0 000000  SPUT  DAC   **           PUT AWAY A-REG FOLLOWED BY 6-CHAR NAME.
02435   0 10 03020      JST   PUT          PUT AWAY THE KEY.
02436   0 02 03774      LDA   DATA         GET THE FIRST WORD.
02437   0 10 03020      JST   PUT          PLACE IT AWAY.
02440   0 02 03775      LDA   DATA+1       GET THE SECOND WORD.
02441   0 10 03020      JST   PUT          PLACE IT AWAY.
02442   0 02 03776      LDA   DATA+2       GET THIRD WORD
02443   0 10 03020      JST   PUT          PLACE IT.
02444  -0 01 02434      JMP*  SPUT         EXIT, 'T' SET FOR NEXT LOC.
                  *
                  *
                  *
02445   0 000000  DEFS  DAC   **           HERE TO DEFINE SUBPROG OR COMMON.
02446   140040          CRA                SET FLAG TO LEAVE ONE ENTRY
02447   0 04 02662      STA   DEFF         IN SYMBOL TABLE.
02450   0 10 02356      JST   SSS          SET UP SEARCH OF SYMBOL TABLE.
02451   0 02 02672  DEFL LDA  ='000077     GET ALL ENTRIES OF SAME TYPE.
02452   0 10 02362      JST   SSN          MOVE TO NEXT ENTRY.
02453   0 01 02511      JMP   DEFE         END OF TABLE.  GO ENTER NAME.
02454   0 10 02412      JST   MACH         COMPARE THIS ENTRY.
02455   0 01 02451      JMP   DEFL         NOT THE SAME.  RESUME SEARCH.
02456   101040          SNZ                SAME.  CALLED OR LOADED.Q
02457  -0 01 02445      JMP*  DEFS         ALREADY LOADED.  IGNORE
02460   0 10 01620      JST   SETS         CALLED.  SET BASE, MMOD, AND DACF.
02461   140040          CRA                MAKE SURE THE LOAD FLAG IS ON.
02462   0 04 03671      STA   SKIF
02463   0 02 02662      LDA   DEFF         IS THIS THE FIRST ENTRY SEEN
02464   100040          SZE                FOR THIS NAME.Q
```

```
02465   0 01 02503        JMP   DEFW        NO.  GO WIPE OUT ENTRY.
02466  -0 02 03672        LDA*  SSSP        YES.  SET FLAG NOW SEEN.
02467   0 04 02662        STA   DEFF        CHANGE CALL ENTRY
02470   0 03 02671        ANA   ='100077    TO LOADED ENTRY.
02471  -0 04 03672 DEFJ   STA*  SSSP
02472   0 02 03672        LDA   SSSP
02473   0 06 02707        ADD   ='000004
02474   0 10 02642        JST   GET         GET LAST REFER LOC IN A, REPLACE WITH
02475   0 04 03674        STA   T+1         DEFINED VALUE (FOR FUTURE CALLS).
02476   0 02 03677        LDA   VALU
02477  -0 04 03673        STA*  T
02500   0 02 03674        LDA   T+1
02501   0 10 02232        JST   UNST        RUN THE REFERENCE STRING.
02502   0 01 02451        JMP   DEFL        CHECK FOR MORE REFERENCES.
                   *
02503   0 02 03672 DEFW   LDA   SSSP        SEEN AN ENTRY BEFORE.
02504   0 06 02675        ADD   ='000003    BOTH HALVES MUST BE SET
02505   0 04 03673        STA   T           TO ZERO TO AVOID FOULING
02506   140040            CRA               LATER TABLE SEARCHES.
02507  -0 04 03673        STA*  T
02510   0 01 02471        JMP   DEFJ        GO JOIN REGULAR PROCESSING.
                   *
02511   0 02 02662 DEFE   LDA   DEFF        NO MORE ENTRIES FOUND.
02512   100040            SZE               HAVE WE SEEN ANY.Q
02513  -0 01 02445        JMP*  DEFS        YES.  ALL DONE.  EXIT.
02514   0 10 02327        JST   GEN6        NO.  WE MUST MAKE ENTRY.
02515   0 02 03665        LDA   KEY         GET THE KEY.
02516   0 03 02671        ANA   ='100077    MAKE SURE IT SAYS LOADED.
02517   0 10 02434        JST   SPUT        FILL IN KEY AND NAME.
02520   0 02 03677        LDA   VALU        FILL IN VALUE.
02521  -0 04 03673        STA*  T
02522   0 02 03665        LDA   KEY         GET THE KEY FOR THIS NAME.
02523   0 05 02670        ERA   ='100006    CHECK FOR COMMON NAME.
02524   100040            SZE               SKIP IF SO.
02525  -0 01 02445        JMP*  DEFS        NO.  EXIT TO CALLER.
02526   0 02 03677        LDA   VALU        YES.  GET THIS ASSIGNMENT.
02527   0 04 03345        STA   COMN        UPDATE THE COMMON COUNTER.
```

```
02530   0 10 02607        JST   FITS                   MAKE SURE CORE LOAD STILL FITS.
02531  -0 01 02445        JMP*  DEFS                   EXIT TO CALLER.
                    *
                    *
                    *
                    * ******************************************************************
                    * *                        MISCELLANEOUS ROUTINES                  *
                    * ******************************************************************
                    *
                    *
                    *
                    *           STRT    HERE TO COMPLETELY INITIALIZE LOADER.
02532   0 04 00122  STRT  STA   BOT                    CLEAR OUT ADVANCE ENTRY
02533   0 04 03671        STA   SKIF                   SET TO LOAD NEXT SUBPROGRAM
02534   0 04 03667        STA   MODE                   SET MODE OF LOADING (ABS)
02535   0 04 03777        STA   WORD                   SET LOCATION COUNTER TO ZERO
02536   0 04 03333        STA   HIGH                   SET HIGHEST LOC. LOADED
02537   0 04 02660        STA   BASE                   SET TO EQUAL ZERO
02540   0 04 02636        STA   MO
02541   0 04 01657        STA   R24S                   RESET THE EVEN-ODD SWITCH
02542   0 02 02001        LDA   DAC1 =(DAC BOT)        INITIALIZE THE SYMBOL TABLE POINTER
02543   0 04 03340        STA   SYM                    TO POINT TO GERM OF SYMBOL TABLE.
02544   0 04 03662        STA   BACK                   SAVE ALSO FOR BACKUP (OLD SKIPPED).
02545   0 02 03673        LDA   T                      RECALL THE USER'S A-REGISTER.
02546   101040            SNZ                          CLEAR REGISTER MEANS USE DEFAULT.
02547   0 02 02677        LDA   ='001000               DEFAULT PROGRAM BREAK.
02550   0 07 02705        SUB   ='177777               THE PROGRAM BREAK
02551   0 03 02667        ANA   ='077776               MUST BE EVEN.
02552   0 04 03670        STA   PBRK                   SET PROGRAM BREAK.
02553   0 02 02000        LDA   FIL7                   SET THE COMMON ORIGIN.
02554   0 04 03345        STA   COMN
02555   0 04 03663        STA   BACK+1                 SAVE IT ALSO FOR BACKUP.
02556   0 02 02666        LDA   ='101000 (NOP)         RESET THE ENTRY SETTER AND THE MAP
02557   0 04 03154        STA   SENT+1                 PRINTER.
02560   0 04 03230        STA   MAP
02561   0 04 03664        STA   FLAG                   ALSO THE 'BLOCK DATA' FLAG
02562   0 04 00426        STA   NSDS                   AND THE NEW DATA SWITCH.
```

```
02563  0 01 03431         JMP   APM                    SELECT I/O OPTIONS AND RETURN TO (STRN)
02564  140040       STRN  CRA                          PREPARE FOR POSSIBLE HALT, IF DDP-416
02565  000201             IAB                          GET BASE SECTOR LOCATION.
02566  140600             SCB                          CHECK FOR DDP-416 CPU.
02567  101001             SSC                          C-BIT NEVER ON IN DDP-416.
02570  000000             HLT                          HA - ITS A 416 SET THE BASE
02571  101040             SNZ                          *
02572  0 01 02575         JMP   *+3
02573  0 10 02653         JST   MRG2
02574  0 10 02015         JST   SETB
02575  0 10 02607         JST   FITS
02576  0 01 00332         JMP   BLOK
                    *
                    *
                    *
                    *
                    *           RSTR    HERE TO SET PROGRAM BREAK.
                    *
02577  14 0004      RSTR  OCP   '0004                  SET THE ASR TO INPUT MODE
02600  101040             SNZ                          CHECK FOR DEFAULT REQUEST.
02601  0 02 03670         LDA   PBRK                   YES.  GET PRESENT VALUE.
02602  0 07 02705         SUB   ='177777               THE PROGRAM BREAK MUST BE EVEN
02603  0 03 02667         ANA   ='077776               WIPE IT DOWN TO SIZE.
02604  0 04 03777         STA   WORD                   SET THE LOCATION COUNTER.
02605  0 10 03056         JST   CLEN                   PERFORM INTER-PROGRAM CLEAN-UP.
02606  0 01 03431         JMP   APM
                    *
                    *
                    *
02607  0 000000     FITS  DAC   **                     HERE TO TEST FOR CORE OVERFLOW.
02610  0 02 03664         LDA   FLAG                   GET THE 'BLOCK-DATA' FLAG.
02611  101400             SMI                          IF ON, WE ARE LOADING COMMON,
02612  0 01 02621         JMP   FITK                   SO DO NOT UPDATE 'HIGH'.
02613  0 02 03777         LDA   WORD                   NOT BLOCK DATA.  CHECK THE
02614  0 07 03333         SUB   HIGH                   LOCATION COUNTER AGAINST THE
02615  100400             SPL                          HIGHEST LOCATION LOADED SO FAR.
02616  0 01 02621         JMP   FITK                   NOT EXCEEDED.
```

```
02617   0 02 03777        LDA   WORD                  EXCEEDED.
02620   0 04 03333        STA   HIGH                  UPDATE HIGH COUNTER.
02621   0 02 03333  FITK  LDA   HIGH                  CHECK FOR OVERLAP OF
02622   0 07 03345        SUB   COMN                  PROGRAM AND COMMON.
02623   101400            SMI                         SKIP IF OK.
02624   0 10 02636        JST   MO                    OVERLAP.  GO GRIPE.
02625   0 02 03777        LDA   WORD                  COMMON OK.  CHECK FOR OVERLAP
02626   0 07 03340        SUB   SYM                   OF PROGRAM AND LOADER.
02627   100400            SPL                         COMPARE WITH BOTTOM OF LOADER.
02630  -0 01 02607        JMP*  FITS                  BELOW LOADER.  OK.
02631   0 02 03777        LDA   WORD                  ABOVE BOTTOM.  SEE IF ABOVE
02632   0 07 03534        SUB   TOP                   THE LOADER.
02633   100400            SPL
02634   0 10 02636        JST   MO                    NO.  WANTS TO DESTROY LOADER.
02635  -0 01 02607        JMP*  FITS                  YES.  OK.  RETURN.
                    *
                    *
                    *
                    *           MO      HERE TO TAKE 'MO' HALT.
                    *
02636   0 000000    MO    DAC   **                    MEMORY OVERFLOW.
02637   0 02 02665        LDA   ='146717 (MO)         GRIPE TO USER.
02640   0 10 00276        JST   TYER                  HALT.
02641   0 01 03230        JMP   MAP                   UPON RESTART, GIVE USER A MAP.
                    *
                    *
                    *
02642   0 000000    GET   DAC   **                    HERE FOR 'LDA 0,A'.
02643   0 04 03673        STA   T                     SAVE POINTER.
02644  -0 02 03673        LDA*  T                     LOAD THROUGH IT.
02645  -0 01 02642        JMP*  GET                   EXIT TO CALLER.
                    *
                    *
                    *
02646   0 000000    GET2  DAC   **                    HERE TO GET C(C(SSSP)+2).
02647   0 02 03672        LDA   SSSP                  GET SYMBOL TABLE POINTER.
02650   0 06 02664        ADD   ='000002              MOVE UP TWO LOCATIONS.
```

```
02651   0 10 02642        JST   GET                GET CONTENTS.
02652  -0 01 02646        JMP*  GET2               RETURN TO CALLER.
                    *
02653   0 000000     MRG2 DAC   **                 ENTRY POINT
02654   0 05 03670        ERA   PBRK               PREPARE TO MERGE PBRK UNDER MASK
02655   0 03 03666        ANA   MMOD               0 BITS WILL MATCH PBRK, 1 BITS REMAIN
02656   0 05 03670        ERA   PBRK               REMOVE PBRK BITS FROM MASKED AREA
02657  -0 01 02653        JMP*  MRG2               RETURN TO CALLER
                    *
02660                BASE BSS   2                  CURRENT BASE SECTOR, POINTER TO COUNTER
02662                DEFF BSS   1                  DEFINED FLAG, FOR 'DEFS'.
02663                NEXT BSS   1                  NEXT ITEM LOC WHILE STRINGING
                    *
                    *
                    *
                    *     (MISCELLANEOUS ROUTINES CONTINUED AFTER LOADER ENTRY BLOCK)
                    *
                    *
                    *
02664   000002            FIN                      END OF SECTOR.  FLUSH THE LITERALS.
02665   146717
02666   101000
02667   077776
02670   100006
02671   100077
02672   000077
02673   000400
02674   100777
02675   000003
02676   001777
02677   001000
02700   000777
02701   077777
02702   036000
02703   032000
02704   000100
02705   177777
```

```
    02706   077000
    02707   000004

            002710   STB2 EQU   *                 DESECTORING AREA NO. 3
                     *
                     *
                          EJCT
```

```
                          * ********************************************************************
                          * *                      LOADER ENTRY BLOCK                          *
                          * ********************************************************************
                          *
                          *
                          *
                                ORG   ORG+'3000          START NEW SECTOR.
                                SETB  XSA4               DESECTOR AFTER STARTS
                          *
                          *
                          *
     03000    0 01 03011  GO    JMP   JWB                'XX000..  START LOADING(NORMAL MODE)
     03001    0 01 03031        JMP   EOT                'XX001..  RECOVER FROM MISSING 'EOT'.
     03002    0 01 03230        JMP   MAP                'XX002..  PRINT MEMORY MAP.
     03003    0 01 02577        JMP   RSTR               'XX003..  SET PROGRAM BREAK.
     03004    0 01 03427        JMP   FRCE               'XX004,,  FORCE LOAD NEXT PROGRAM
     03005    0 01 03416        JMP   JWB3               'XX005..BEGIN EXECUTION
     03006    0 04 03673  JWB1  STA   T                  'XX006..  START LOADING (EXTENDED MODE)
     03007    0 02 03521        LDA   ='77777            INITIALIZE TO DESECTOR IN EXTENDED MODE
     03010    0 01 03013        JMP   JWB2
     03011    0 04 03673  JWB   STA   T
     03012    0 02 03520        LDA   ='37777            INITIALIZE TO DESECTOR IN NORMAL MODE
     03013    0 04 03666  JWB2  STA   MMOD
     03014    140040            CRA
     03015    0 04 00000        STA   0
     03016    14 0004           OCP   '0004              SET ASR TO INPUT MODE
     03017    0 01 02532        JMP   STRT
                          *
                          *
                          *
                          *
                          *
                          *
                          * ********************************************************************
                          * *              MISCELLANEOUS ROUTINES (CONTINUED)                  *
                          * ********************************************************************
```

```
                         *
                         *
                         *
03020    0 000000   PUT  DAC   **          HERE TO STORE THROUGH 'T' AND BUMP.
03021   -0 04 03673      STA*  T           PLACE A-REGISTER THROUGH POINTER.
03022    0 12 03673      IRS   T           INCREMENT POINTER.
03023   -0 01 03020      JMP*  PUT         EXIT TO CALLER.
                         *
                         *
                         *
03024    0 000000   SKIP DAC   **          HERE TO TEST SKIP FLAG.
03025    0 02 03671      LDA   SKIF        GET THE SKIP FLAG.
03026    100040          SZE               ARE WE SKIPPING OR LOADING.Q
03027    0 01 00332      JMP   BLOK        SKIPPING.  GO READ NEXT BLOCK.
03030   -0 01 03024      JMP*  SKIP        LOADING.  RETURN TO CALLER.
                         *
                         *
                         *
03031    14 0101    EOT  OCP   '0101       MAKE SURE THE READER IS OFF.
03032    0 10 03056      JST   CLEN        PERFORM INTER-PROGRAM CLEAN-UP.
03033    0 02 03033      LDA   *           SET THE SKIP FLAG TO
03034    0 04 03671      STA   SKIF        SKIP THE NEXT SUBPROGRAM.
03035    0 02 02636      LDA   MO          CHECK IF THERE HAS ALREADY
03036    100040          SZE               BEEN A MEMORY OVERFLOW.
03037    0 01 02637      JMP   MO+1        YES.  TAKE 'MO' HALT AGAIN.
03040    0 02 03522      LDA   ='100405    NO.  SET UP KEY TO LOOK FOR
03041    0 04 03665      STA   KEY         MISSING SUBPROGRAMS.
03042    0 10 02356      JST   SSS         SET UP A SEARCH OF THE SYMBOL TABLE.
03043    0 02 03523      LDA   ='100477    MASK FOR KEYS DURING SEARCH.
03044    0 10 02362      JST   SSN         ARE ANY MISSING.Q
03045    0 01 03051      JMP   LCER        NO.  TAKE 'LC' HALT.
03046    0 02 03517      LDA   ='146722 (MR)  YES.  TAKE 'MR' HALT.
03047    0 10 00276      JST   TYER        GRIPE TO USER AND HALT.
03050    0 01 03433      JMP   APM+2       TEST INPUT DEVICE
                         *
03051    140040     LCER CRA               HERE WHEN THE LOAD IS COMPLETE.
03052    0 04 03671      STA   SKIF        FORCE LOAD THE NEXT SUBPROGRAM
```

```
03053   0 02 03516        LDA   ='146303 (LC)    IF THE USER CONTINUES LOADING.
03054   0 10 00276        JST   TYER             CONGRATULATE USER AND HALT.
03055   0 01 03416        JMP   JWB3             UPON RESTART,EXECUTE PROGRAM
                    *
                    *
03056   0 000000    CLEN  DAC   **               HERE TO PERFORM END CLEAN-UP.
03057   0 02 03664        LDA   FLAG             CHECK FOR 'BLOCK DATA'.
03060   101400            SMI
03061   0 04 03777        STA   WORD             RESTORE LOCATION IF 'BLOCK DATA'.
03062   0 02 03777        LDA   WORD             CHECK FOR EVEN LOCATION.
03063   0 07 03515        SUB   ='177777
03064   0 03 03514        ANA   ='077776
03065   0 04 03777        STA   WORD             UPDATE THE LOCATION COUNTER.
03066   0 04 03670        STA   PBRK             UPDATE THE PROGRAM BREAK.
03067   0 10 02607        JST   FITS             MAKE SURE IT STILL FITS.
03070   0 02 03225        LDA   NOP              RESET THE NEW STYLE BLOCK FOUR
03071   0 04 00426        STA   NSDS             INITIALIZATION SWITCH.
03072   0 04 03664        STA   FLAG             RESET 'BLOCK DATA' FLAG.
03073   0 10 02002        JST   SAVE             SAVE LOADER WHILE STRINGING.
03074   0 10 02356        JST   SSS              SET UP FOR SEARCH OF TABLE.
03075   0 02 03513        LDA   ='000002         SET UP KEY FOR ADDRESS-ARITHMETIC.
03076   0 04 03665        STA   KEY
03077   0 02 03512  CLEL  LDA   ='000077         LOOK AT ALL ENTRIES OF THIS TYPE.
03100   0 10 02362        JST   SSN              MOVE TO NEXT ENTRY.
03101   0 01 03142        JMP   CLEX             NO MORE ENTRIES.  GO RESTORE LOADER.
03102   0 04 03677        STA   VALU             INITIALIZE VALUE ACCUMULATOR.
03103   0 02 03672        LDA   SSSP             SAVE SYMBOL SEARCH
03104   0 04 03774        STA   DATA             POINTER.
03105   0 10 02646        JST   GET2             GET LOCATION WHICH USES
03106   0 04 03777        STA   WORD             THIS VALUE.
03107   0 02 03512  CLEM  LDA   ='000077         SEARCH FOR
03110   0 10 02362        JST   SSN              OTHER REFERENCES
03111   0 01 03125        JMP   CLEY             TO SAME LOCATION.
03112   0 10 02646        JST   GET2             GET LOCATION ENTRY.
03113   0 05 03777        ERA   WORD             COMPARE TO LOCATION.
03114   100040            SZE                    IS IT THE SAME.Q
03115   0 01 03107        JMP   CLEM             NO.  CONTINUE SEARCH.
```

```
03116  -0 04 03672        STA*   SSSP              YES, WIPE IT OUT
03117   0 02 03672        LDA    SSSP
03120   0 07 03515        SUB    ='177777
03121   0 10 02642        JST    GET               GET VALUE.
03122   0 06 03677        ADD    VALU              ADD IT IN THE ADDRESS
03123   0 04 03677        STA    VALU              ACCUMULATOR.
03124   0 01 03107        JMP    CLEM              GO LOOK FOR MORE REFERENCES.
                     *
03125   0 02 03774   CLEY LDA    DATA              ALL ENTRIES FOR THIS ADDRESS
03126   0 04 03672        STA    SSSP              COMPLETE.  RESTORE SEARCH
03127  -0 02 03672        LDA*   SSSP              POINTER.
03130   0 03 03511        ANA    ='000200          IS WORD A DAC OR A NINE-BIT.Q
03131   101040            SNZ                      CHECK BIT FROM KEY.
03132   0 01 03140        JMP    CLED              DAC.  GO SERVICE.
03133   0 10 01620        JST    SETS              NINE-BIT.  SET BASE, MMOD.
03134   0 10 02065        JST    DESE              DESECTOR TO VALUE.
03135   140040       CLEJ CRA                      WIPE OUT THE FINAL
03136  -0 04 03672        STA*   SSSP              ADDRESS ARITHMETIC ENTRY.
03137   0 01 03077        JMP    CLEL              NOW MOVE TO NEXT LOCATION.
                     *
03140   0 10 03165   CLED JST    PROP              *
03141   0 01 03135        JMP    CLEJ              GO WIPE ENTRY AND PROCEED.
                     *
03142   0 10 02055   CLEX JST    REST              ALL DONE WITH CLEAN-UP.  RESTORE LOADER.
03143   0 02 03662        LDA    BACK              BACK UP THE SYMBOL TABLE POINTER AND
03144   0 04 03340        STA    SYM               THE COMMON COUNTER IN CASE LAST SUBPROG
03145   0 02 03663        LDA    BACK+1            WAS AN OLD-STYLE SKIPPED.
03146   0 04 03345        STA    COMN
03147   140040            CRA
03150  -0 04 03340        STA*   SYM               CLEAN OUT THE ADVANCED ENTRY.
03151   0 04 03667        STA    MODE              SET ABSOLUTE MODE.
03152  -0 01 03056        JMP*   CLEN              RETURN TO CALLER.
                     *
                     *
                     *
                     *    JST    SENT    SET ENTRY LOCATION FOR MAIN PROGRAM.  DISABLES ITSELF
                     *                   AFTER FIRST CALL.
```

```
                          *
03153    0 000000   SENT DAC   **              HERE TO SET MAIN PROG ENTRY.
03154    101000          NOP                   OVERWRITE THIS LOC TO DISABLE.
03155    0 10 03177      JST   RELO            RELOCATE LOCATION IF NECESSARY.
03156    101040          SNZ                   *
03157    0 02 03670      LDA   PBRK            *
03160    0 10 02653      JST   MRG2            ASSURE BIT 2 SET OK IF NORMAL MODE
03161    0 04 03326      STA   NTRY            SAVE IT AWAY.
03162    0 02 03164      LDA   *+2             DISABLE THIS ROUTINE.
03163    0 04 03154      STA   SENT+1          FUTURE CALLS IGNORED.
03164   -0 01 03153      JMP*  SENT            RETURN TO CALLER.
                    *
                    *
                    *    PROP - - BITS OF VALUE PLUS BITS FROM (INDIRECT WORD) WHERE
                    *             MMOD IS ZERO, PUT INTO (INDIRECT WORD)
                    *
03165    0 000000   PROP DAC   **              SUBROUTINE
03166   -0 02 03777      LDA*  WORD            *
03167    0 05 03677      ERA   VALU            * WORD      // //1 111
03170   -0 03 03777      ANA*  WORD            * MMOD      00 110 011
03171   -0 04 03777      STA*  WORD            * VALUE      01 010 101
03172    0 05 03666      ERA   MMOD            *
03173   -0 03 03777      ANA*  WORD            * RESULT
03174    0 05 03677      ERA   VALU            * IN  *WORD 01 011 101
03175   -0 04 03777      STA*  WORD            *
03176   -0 01 03165      JMP*  PROP            EXIT WITH GREAT SPEED
                    *
                    *
                    *
                    *
                    *    JST   RELO    RELOCATE ADDRESS IN 'VALU' BY THE RELOCATION CONSTANT
                    *                  'PBRK' ACCORDING TO THE RELOCATION BITS C(A-REG)1,2.
                    *                  RESULT IN 'VALU' AND A-REGISTER.
                    *                                 00..  ABSOLUTE.
                    *                                 01..  POSITIVELY RELOCATABLE.
                    *                                 10..  NEGATIVELY RELOCATABLE.
                    *                                 11..  NEGATIVELY RELOCATABLE.
```

```
                           *
03177   0 000000    RELO DAC   **                   HERE TO RELOCATE 'VALU'
03200   0 03 03510       ANA   ='140000             ISOLATE BITS
03201   101040           SNZ                        TEST FOR NO RELOCATION
03202   0 01 03207       JMP   RABS                 ABSOLUTE
03203   100400           SPL                        RELOCATABLE. PLUS OR MINUS
03204   0 01 03212       JMP   RELN                 NEGATIVE RELOCATION
03205   0 02 03670       LDA   PBRK                 GET RELOCATION CONSTANT
03206   0 03 03666       ANA   MMOD                 MAKE 14 OR 15 BIT
03207   0 06 03677  RABS ADD   VALU                 ADD 16 BIT VALU
03210   0 04 03677       STA   VALU
03211  -0 01 03177       JMP*  RELO                 EXIT
03212   0 02 03670  RELN LDA   PBRK                 GET RELOCATION CONSTANT
03213   0 03 03666       ANA   MMOD                 MAKE 14 OR 15 BIT
03214   0 05 03515       ERA   ='177777             ONE'S COMPLEMENT
03215   0 07 03515       SUB   ='177777             TWO'S COMPLEMENT
03216   0 01 03207       JMP   RABS                 GO GET VALU
                    *
                    *
                    *
                    *    JST   SWCH    SWITCH JUMP.  RETURNS INDIRECTLY THROUGH 'JST+1+N',
                    *                  WHERE 'N' IS IN THE A-REGISTER UPON THE CALL.
                    *                  CALL MUST BE FOLLOWED BY A SUFFICIENT NUMBER OF DACS.
                    *
03217  -0 000000    SWCH DAC*  **                   INDIRECT RETURN.
03220   0 06 03217       ADD   SWCH                 ADD A-REGISTER TO RETURN LOCATION.
03221   0 04 03217       STA   SWCH                 SET NEW RETURN.
03222  -0 01 03217       JMP*  SWCH                 RETURN THROUGH 'JST+1+N'.
                    *
                    *
                    *
                    * *******************************************************************
                    * *                    SUBPROGRAM STORAGE MAP                       *
                    * *******************************************************************
                    *
                    *
                    *
```

```
03223   0 01 03224 MAPF JMP   *+1          HERE TO TURN OFF MAP.
03224   0 02 03225      LDA   *+1          RESET MAP ENTRY TO
03225   101000     NOP  NOP                PRINT MAP AGAIN NEXT TIME
03226   0 04 03230      STA   MAP
03227   0 01 03031      JMP   EOT          GO PERFORM END CLEAN-UP.
                   *
                   *
                   *
03230   101000     MAP  NOP                HERE TO PRINT MEMORY MAP.
03231   0 02 03223      LDA   MAPF         SET JUMP TO TERMINATE
03232   0 04 03230      STA   MAP          MAP IF USER DESIRES.
03233   0 10 00313      JST   TTYN         TURN ON THE TELETYPE.
03234   0 10 01650      JST   CRLF         SPACE UP A LINE.
03235   0 10 02356      JST   SSS          SET FOR ENTRY WITH CLEAR 'D' BIT.
03236   0 02 03322      LDA   MAP1         TYPE THE PROGRAM ENTRY
03237   0 10 03353      JST   LINE         LOCATION.
03240   0 02 03327      LDA   MAP2         TYPE THE HIGHEST LOCATION        8
03241   0 10 03353      JST   LINE         LOADED.
03242   0 02 03334      LDA   MAP3         TYPE THE SIZE OF THE
03243   0 10 03353      JST   LINE         SYMBOL TABLE.
03244   0 02 03341      LDA   MAP4         TYPE THE COMMON BREAK.
03245   0 10 03353      JST   LINE
03246   0 04 03774 MAPP STA   DATA         CLEAR CURRENT LOCATION COUNTER.
03247   0 10 02356      JST   SSS          SET UP A SYMBOL TABLE SEARCH.
03250   0 02 03507      LDA   ='100000     SET NEXT PASS OFFSET TO BE
03251   0 04 03775 MAPK STA   DATA+1       AS LARGE AS POSSIBLE.
03252   140040     MAPL CRA                LOOK AT ENTRIES OF ALL KEYS.
03253   0 10 02362      JST   SSN          MOVE TO NEXT ENTRY.
03254   0 01 03315      JMP   MAPE         END OF TABLE.  GO CLOSE THIS PASS.
03255  -0 02 03672      LDA*  SSSP         GET KEY FOR THIS ENTRY.
03256   100400          SPL                CHECK FOR NAME ENTRY (MINUS).
03257   0 01 03277      JMP   MAPS         YES.  SEE IF NOW IS TIME TO PRINT IT.
03260   0 03 03512      ANA   ='000077     NO.  MAYBE IT IS A BASE SECTOR
03261   0 05 03506      ERA   ='000004     ENTRY (KEY=4).
03262   0 06 03774      ADD   DATA         PRINT THEM ONLY ON FIRST PASS.
03263   100040          SZE                SKIP IF BASE ENTRY ON FIRST PASS.
03264   0 01 03252      JMP   MAPL         NO.  RESUME SEARCH.
```

```
03265  -0 02 03673        LDA*  T              YES.  GET BASE TABLE ORIGIN.
03266   0 12 03673        IRS   T              MOVE TO USAGE.
03267  -0 07 03673        SUB*  T              CALCULATE LAST LOCATION
03270   0 06 03515        ADD   ='177777       USED PLUS ONE.
03271   0 04 03352        STA   MA55           SAVE IT FOR PRINTING.
03272   0 02 03346        LDA   MAP5           GET POINTER FOR PRINTER.
03273   100000            SKP                  SKIP AROUND NAME JOINT.
03274   0 02 03673 MAPC   LDA   T              GET POINTER TO PRINT NAME.
03275   0 10 03353        JST   LINE           TYPE SYMBOL TABLE LINE.
03276   0 01 03252        JMP   MAPL           RESUME SEARCH LOOP.
                   *
03277   0 02 03672 MAPS   LDA   SSSP           HERE FOR NAME ENTRY.
03300   0 06 03506        ADD   ='000004       FIND THE VALUE OF THIS NAME.
03301   0 04 03674        STA   T+1
03302  -0 02 03674        LDA*  T+1            VALUE TO A-REGISTER.
03303   0 07 03774        SUB   DATA           CHECK IF IT MATCHES VALUES BEING
03304   101040            SNZ                  PRINTED ON THIS PASS.
03305   0 01 03274        JMP   MAPC           YES.  GO PRINT THIS NAME.
03306   100400            SPL                  NO.  IF IT IS LESS THAN THE CURRENT
03307   0 01 03252        JMP   MAPL           VALUE, IGNORE IT.
03310   0 07 03775        SUB   DATA+1         NO, MORE.  COMPARE IT WITH THE
03311   101400            SMI                  OFFSET FOR THE NEXT PASS.
03312   0 01 03252        JMP   MAPL           TOO HIGH.  IGNORE IT.
03313   0 06 03775        ADD   DATA+1         LOWER THAN PRESENT OFFSET.
03314   0 01 03251        JMP   MAPK           GO DROP OFFSET.
                   *
03315   0 02 03775 MAPE   LDA   DATA+1         END OF PASS.  CHECK OFFSET.
03316   100400            SPL                  IF STILL SET TO '100000, THERE IS
03317   0 01 03223        JMP   MAPF           NO NEXT PASS.  EXIT FROM MAP.
03320   0 06 03774        ADD   DATA           NO, BUMP VALUE AND GO FOR MORE.
03321   0 01 03246        JMP   MAPP           START NEXT PASS.
                   *
03322   0 003323   MAP1   DAC   *+1            PROGRAM ENTRY ADDRESS.
03323   125323            BCI   3,*START
03324   152301
03325   151324
03326   0 000000   NTRY   DAC   **
```

```
                          *
     03327    0 003330    MAP2 DAC    *+1            HIGHEST LOCATION LOADED.
     03330    125310           BCI    3,*HIGH
     03331    144707
     03332    144240
     03333    0 000000    HIGH DAC    **
                          *
     03334    0 003335    MAP3 DAC    *+1            BOTTOM OF LOADER.
     03335    125316           BCI    3,*NAMES
     03336    140715
     03337    142723
     03340    0 000000    SYM  DAC    **
                          *
     03341    0 003342    MAP4 DAC    *+1            COMMON BREAK.
     03342    125303           BCI    3,*COMN
     03343    147715
     03344    147240
     03345    0 000000    COMN DAC    **
                          *
     03346    0 003347    MAP5 DAC    *+1            BASE SECTOR USAGE.
     03347    125302           BCI    3,*BASE
     03350    140723
     03351    142640
     03352    0 000000    MA55 DAC    **
                          *
                          *
                          *
     03353    0 000000    LINE DAC    **             HERE TO TYPE LINE AND CR/LF.
     03354    0 04 03673       STA    T              SAVE INPUT POINTER.
     03355    0 10 03401       JST    LIN2           PRINT TWO CHARACTERS.
     03356    0 10 03401       JST    LIN2           PRINT NEXT TWO CHARACTERS.
     03357    0 10 03401       JST    LIN2           PRINT LAST TWO CHARACTERS.
     03360    0 02 03505       LDA    ='000240       PRINT A BLANK.
     03361    0 10 01636       JST    TYP1
     03362   -0 02 03673       LDA*   T              NOW GET NUMBER.
     03363    0416 74          ALR    4              POSITION IT FOR PRINTING.
     03364    0 10 03406       JST    LIN8           PRINT OCTAL DIGIT.
```

```
03365    0 10 03406        JST   LIN8
03366    0 10 03406        JST   LIN8
03367    0 10 03406        JST   LIN8
03370    0 10 03406        JST   LIN8                 PRINT LAST OCTAL DIGIT.
03371   -0 02 03672        LDA*  SSSP                 GET KEY WORD FOR THIS ENTRY.
03372    0 03 03504        ANA   ='000400             ISOLATE 'DEFINED' BIT.
03373    101040            SNZ                        SKIP IF UNDEFINED.
03374    0 01 03377        JMP   LINX                 DEFINED.  GO TO CR/LF.
03375    0 02 03503        LDA   ='125252 (**)        UNDEFINED.
03376    0 10 01643        JST   TYP2                 TYPE TWO ASTERISKS.
03377    0 10 01650  LINX  JST   CRLF                 CARRIAGE-RETURN/LINE-FEED.
03400   -0 01 03353        JMP*  LINE                 EXIT TO CALLER OF 'LINE'.
                   *
03401    0 000000    LIN2  DAC   **                   HERE TO ADD TWO CHARS TO LINE.
03402   -0 02 03673        LDA*  T                    GET TWO CHARACTERS.
03403    0 10 01643        JST   TYP2                 TYPE THEM.
03404    0 12 03673        IRS   T                    BUMP POINTER.
03405   -0 01 03401        JMP*  LIN2                 RETURN TO CALLER.
                   *
03406    0 000000    LIN8  DAC   **                   HERE TO TYPE OCTAL DIGIT.
03407    0 04 03673        STA   T                    SAVE WORD.
03410    0 03 03502        ANA   ='000007             ISOLATE DIGIT.
03411    0 06 03501        ADD   ='000260             CONVERT TO ASCII.
03412    0 10 01636        JST   TYP1                 PRINT IT.
03413    0 02 03673        LDA   T                    RESTORE WORD.
03414    0416 75           ALR   3                    SHIFT NEXT DIGIT TO POSITION.
03415   -0 01 03406        JMP*  LIN8                 RETURN.
03416    0 04 03673  JWB3  STA   T                    SAVE THE A REGISTER FOR EXECUTION
03417    0 02 03666        LDA   MMOD                 GET THE INTENDED MODE (SAME AS LOAD)
03420    0404 63           LGR   13                   SHIFT OFF ALL BUT FIRST THREE BITS
03421    0 06 03500        ADD   ='10                 GENERATE THE EXA OR DXA
03422    0 04 03425        STA   DEXA                 SAVE APPROPRIATE MODE OF EXECUTION
03423    0 02 03673        LDA   T                    RESTORE THE A REGISTER
03424    000013            EXA                        ALLOW ENTRY TO ANYPLACE IN CORE
03425    0 00 00000  DEXA  PZE                        EITHER A DXA (11) OR EXA(13)
03426   -0 01 03326        JMP*  NTRY
03427    140040      FRCE  CRA                        FORCE LOAD
```

```
03430   0 04 03671        STA   SKIF        TURN OFF SKIP FLAG
03431   140040       APM  CRA               ZERO A FOR DEVICE CODE
03432   000000            HLT               HALT FOR INPUT DEVICE
03433   0 04 03460        STA   IO          SAVE FOR POSSIBLE MTU NO.
03434   101040            SNZ               TEST FOR DEFAULT (A=0)
03435   0 02 01000        LDA   DFLT        PICKUP DEFAULT DEVICE
03436   0 04 03457        STA   PTIN        SAVE IN ASR/HSR SWITCH
03437   0 03 03502        ANA   ='7         EXTRACT DEVICE CODE
03440   0 07 03477        SUB   =3          TEST FOR MAG TAPE
03441   100400            SPL               *
03442   0 01 03452        JMP   PAP         ASR OR HSR
03443   100040            SZE
03444   0 01 03463        JMP   DISC        DISC INPUT
03445   0 02 03460        LDA   IO          MAG TAPE
03446   0404 75           LGR   3           RIGHT ADJUST UNIT NO.
03447   0 04 03527        STA   MGRD+3      INSERT IN CALLING SEQ.
03450   0 02 03462        LDA   MSW         PICK UP MAG BRANCH
03451   0 01 03453        JMP   *+2
03452   0 02 03461   PAP  LDA   PSW         PICK UP PAPER BRANCH
03453   0 04 00660        STA   SW          SET DEVICE SWITCH
03454   0 01 02564        JMP   STRN        GO SET BASE
03455   0 000000     MAG  DAC   **          ENTRY FOR MAG READ
03456   0 01 03524        JMP   MGRD        GET A RECORD
03457                PTIN BSS   1           ASR/ASR SWITCH
03460                IO   BSS   1           STORAGE FOR INPUT DEVICE
03461   0 000237     PSW  DAC   R1K+70      BRANCH FOR PAPER
03462   0 003455     MSW  DAC   MAG         BRANCH FOR MAG
                     *
        000770       DOPI EQU   '770        DISC INITIALIZATION
        000772       DOPO EQU   '772        OBJECT INPUT FROM DISC VIA DOP
                     *
03463   0 02 03476   DISC LDA   ='100002    BITS 1 AND 15 ON FOR DOP
03464   101000            NOP               FOR USED IN SPECIAL SYSTEMS  (OCP '40)
03465   000013            EXA               *
03466  -0 10 00770        JST*  DOPI        GET PROGRAM NAME
03467   0 02 03471        LDA   DSW         DISC ENTRY
03470   0 01 03453        JMP   PAP+1
```

```
03471    0 003472   DSW  DAC   DISI              ENTRY FOR DISC INPUT
                    *     .
03472    0 000000   DISI DAC   **
03473   -0 10 00772      JST*  DOPO              GET BLOCK FROM DISC
03474    0 003700        DAC   BUFF              BUFFER
03475   -0 01 00131      JMP*  R1K
03476    100002          FIN                     FLUSH THE LITERALS
03477    000003
03500    000010
03501    000260
03502    000007
03503    125252
03504    000400
03505    000240
03506    000004
03507    100000
03510    140000
03511    000200
03512    000077
03513    000002
03514    077776
03515    177777
03516    146303
03517    146722
03520    037777
03521    077777
03522    100405
03523    100477

                    * ************************************************************
                    * *                       STORAGE                            *
                    * ************************************************************
         003662     BACK EQU   ORG+'3662         SPACE FOR BACKUP SYMBOL COUNT,COMMON BRK.
         003664     FLAG EQU   BACK+2            'BLOCK DATA'FLAG, STORE 'PBRK'
         003665     KEY  EQU   FLAG+1            STORAGE FOR SYMBOL TABLE KEY WORD
         003666     MMOD EQU   KEY+1             MEM MODE, 16K(LXD)='037777,32K(EXD)=077777
         003667     MODE EQU   MMOD+1            ABS ('000000) OR REL ('040000).
```

```
           003670     PBRK EQU   MODE+1            PROGRAM BREAK
           003671     SKIF EQU   PBRK+1            SKIP FLAG.  ZERO MEANS LOAD
           003672     SSSP EQU   SKIF+1
           003673     T    EQU   SSSP+1            FOUR CELLS FOR TEMPORARIES
           003677     VALU EQU   T+4               ADDRESS OF INSTRUCTION
           003700     BUFF EQU   ORG+'3700         BLOCK INPUT BUFFER FOR 'R1K'.
           003774     DATA EQU   BUFF+60           CURREDT INPUT DATA, BLOCK ROUTINES
           003777     WORD EQU   DATA+3            LOCATION COUNTER.  END OF SECTOR
           003700     COMM EQU   ORG+'3700
                      *
     03524 0 10 00000 MGRD CALL  I$MC              CALL MAG READ ROUTINE
     03525 0 003700        DAC   BUFF              LOCATION OF BUFFER
     03526 000074          DEC   60                WORD COUNT
     03527 000000          DEC   0                 TAPE UNIT NUMBER
     03530 000000          HLT                     TAPE ERROR
     03531 000000          HLT                     END OF TAPE RETURN
     03532 0 01 03031      JMP   EOT               END OF FILE RETURN
     03533 0 01 00231      JMP   R1K+64            PROCESS RECORD
                      *
     03534 0 003777   TOP  DAC   WORD
                      *
           003535     XSA4 EQU   *                 DESECTORING AREA NO. 4
                      *
                      *    FROM HERE TO 3577 CAN BE USED FOR PATCHING AREA
                      *    FROM THERE ON WILL BE OVERLAID BY THE BOOTSTRAP, WHICH IN TURN WILL
                      *    BE USED FOR TEMPORARY STORAGE AFTER LOADING THE LOADER.
                      *
                      *
                           ORG   ORG+'3577              FORCE MEMORY MAP TO GIVE
     03577                 BSS   1                 HIGH ABOVE DESECTORING AREA
                      *
                      *
                           END   GO                THAT'S ALL FOLKS.
NO ERRORS IN ABOVE ASSEMBLY.
DAP-16 INHOUSE 1.7.68 PVP
```