```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                   ASM H V 05 22.10 05/07/81

                                      2 *   THIS FILE IS AN ALTERATION OF ORIGINAL SOURCE TO CONFORM TO
                                      3 *     MORE SOPHISTICATED ASSEMBLER.
                                      4 *   SPECIFICALLY, USING THE CONDITIONAL ASSEMBLY TO PERFORM THE
                                      5 *     FUNCTION OF THE EDITOR PROGRAM SUPPLIED WITH THE PACKAGE.
                                      6 *
                                      7 *   FURTHER MODIFICATIONS MADE TO ALLOW RUNNING UNDER CMS
                                      8 *     1. CARD I/O VIA CMS AND CP SPOOL ONLY
                                      9 *     2. ELIMINATE THE 'B  *'  WAIT LOOPS
                                     10 *     3. 2 POSITION ADCONS NOW FULLWORDS.  I.E. LOTS OF LH CHANGED
                                     11 *          TO L  INSNS.
                                     12 *     4. STAND-ALONE LOADER STUFF REMOVED
                                     13 *     5. MACHINE CHECK STUFF REMOVED
                                     14 *     6. TERMINAL I/O VIA WRTERM AND LINEDIT
                                     15 *
                                     16 *  HERE ARE THE CONDITIONAL ASSEMBLY TRANSLATIONS
                                     17 *
                                     18 *  NOTE: THE SIMULATED 1620 IS ALWAYS ASSUMED TO HAVE
                                     19 *        A 2540 READER/PUNCH,  1403 PRINTER, AND A CONSOLE
                                     20 *
                                     21 *  ALL OTHER I/O DEVICES ARE SPECIFIED BY SETTING BITS ON IN
                                     22 *    APPROPRIATE BINARY VARIABLES.
                                     23 *
                                     24          GBLA  &CORE         VALUE 2,4,6 FOR 20K, 40K, 60K 1620
                                     25          GBLA  &MODEL        VALUE 1 OR 2 FOR MODEL OF 1620
                                     26          GBLB  &R1621P       SIMULATING A PAPER TAPE PUNCH
                                     27          GBLB  &R1621R       SIMULATING A PAPER TAPE READER
                                     28          GBLB  &INDEX        INDEXING INSTRUCTIONS ALLOWED
                                     29          GBLB  &INDAD        INDIRECT ADDRESSING ALLOWED
                                     30          GBLB  &TRANS
                                     31          GBLB  &DIVIDE       FIXED POINT DIVISION ALLOWED
                                     32          GBLB  &FLOAT        FLOATING POINT INSNS. ALLOWED
                                     33          GBLB  &DISK         ON IF DISK(S) ARE BEING SIMULATED
                                     34          ACTR  100
                                     35 *
                                     36 *
                                     37 *    SET VALUES FOR TRIAL ASSEMBLY
                                     38 &CORE    SETA  2     20K MACHINE
                                     39 &MODEL   SETA  2     DELUXE MODEL
                                     40 &INDEX   SETB  1
                                     41 &TRANS   SETB  1
                                     42 &DIVIDE  SETB  1
                                     43 &FLOAT   SETB  1
                                     44 *
                                     45 *
                                     46 *
                                     47 * STATUS CHANGE  VERSION 1,LEVEL 2                          V1L2
000000                               48 CONTPR   START 0                *
                               00000 49          USING *,0
                               01000 50          USING *+4096,3
                               02000 51          USING *+8192,4
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
***********************************************************************
*                                                                     *
*                                SIM20                                *
*               IBM SYSTEM/360 SIMULATOR FOR THE IBM 1620             *
*                                                                     *
*                            INTRODUCTION                             *
*                                                                     *
*  THE SIMULATOR CONTAINS THE FOLLOWING MAJOR SECTIONS =              *
*                                                                     *
*                  - CONSOLE SIMULATION                               *
*                  - BASIC INTERPRETIVE ROUTINE  (BIR)                *
*                  - CPU SIMULATION                                   *
*                  - I/O SIMULATION                                   *
*                  - DISK SIMULATION                                  *
*                                                                     *
*  EACH SECTION OF THE SIMULATOR, WITH THE EXCEPTION OF THE BASIC IN- *
*  TERPRETIVE ROUTINE, CONTAINS ROUTINES WHICH SIMULATE  THE  CORRE-  *
*  SPONDING FUNCTIONS OF A 1620.                                      *
*                                                                     *
*  THE FOLLOWING COMMON INFORMATION IS USED BY MORE THAN ONE SECTION  *
*                                                                     *
*        - SIMULATED 1620 CORE STORAGE                                *
*                                                                     *
*        - SIMULATED INSTRUCTION COUNTER                              *
*                                                                     *
*        - SIMULATED REGISTERS                                        *
*                                                                     *
*        - SYSTEM/360 REGISTER ALLOCATION                             *
*                                                                     *
*        - SIMULATED 1620 INDICATORS                                  *
*                                                                     *
*                                                                     *
***********************************************************************
```

```
LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                   ASM H V 05 22.10 05/07/81

                                  88 **********************************************************************
                                  89 *                                                                    *
                                  90 *                    GENERAL REGISTER ASSIGNMENT                     *
                                  91 *                                                                    *
                                  92 **********************************************************************

00001                             94 R1       EQU   1                         REGISTER 1
00002                             95 R2       EQU   2                         REGISTER 2
00003                             96 SIMB1    EQU   3                         BASE REGISTER 1 (4096 TO 8192)
00004                             97 SIMB2    EQU   4                         BASE REGISTER 2 (8192 TO 12288)
00005                             98 CNTR     EQU   5                         INSTRUCTION COUNTER
00006                             99 MAPORG   EQU   6                         GL.REGISTER CONTAINING THE ABSO-
                                 100 *                                        LUTE ADDRESS OF MAP ORIGIN
00007                            101 SIZE     EQU   7                         GL.REGISTER CONTAINING THE MAXI-
                                 102 *                                        MUM ABSOLUTE ADDRESS OF SIMULAT-
                                 103 *                                        ED 1620 CORE STORAGE
00008                            104 RP       EQU   8                         REGISTER P
00009                            105 RQ       EQU   9                         REGISTER Q
0000A                            106 WR1      EQU   10                        WORKING REGISTER 1
0000B                            107 WR2      EQU   11                        WORKING REGISTER 2
0000C                            108 WR3      EQU   12                        WORKING REGISTER 3
0000D                            109 WR4      EQU   13                        WORKING REGISTER 4
0000E                            110 WR5      EQU   14                        WORKING REGISTER 5
0000F                            111 WR6      EQU   15                        WORKING REGISTER 6

                                 113 **********************************************************************
                                 114 *                                                                    *
                                 115 *          CONDITION CODE MNEMONICS FOR BRANCH INSTRUCTIONS          *
                                 116 *                                                                    *
                                 117 **********************************************************************

0000F                            119 BR       EQU   15                        * UNCONDITIONAL BRANCH (REGISTER
00000                            120 NOP      EQU   0                         * NO OPERATION
00005                            121 BOM      EQU   5                         * BRANCH ON OVERFLOW OR MINUS
00008                            122 BZ       EQU   8                         * BRANCH ON ZERO
0000C                            123 BZM      EQU   12                        * BRANCH ON ZERO OR MINUS
00080                            124 CONVTB   EQU   128                       ABSOLUTE ADDRESS OF CODE CONVER-
                                 125 *                                        SION TABLE FOR I/O OPERATIONS
```

```
LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                      ASM H V 05 22.10 05/07/81

                                  127 ***********************************************************************
                                  128 *                                                                     *
                                  129 *                          CONSOLE SIMULATION                         *
                                  130 *                                                                     *
                                  131 *  THIS SECTION OF THE SIMULATOR CONTAINS ALL ROUTINES,  SUBROUTINES  *
                                  132 *  AND TABLES NEEDED FOR SIMULATION OF THE 1620 CONSOLE  OPERATIONS.  *
                                  133 *                                                                     *
                                  134 *  CONSOLE SIMULATION CONSISTS OF =                                   *
                                  135 *                                                                     *
                                  136 *  - SIMULATION OF CONSOLE KEYS, SWITCHES AND INDICATORS              *
                                  137 *                                                                     *
                                  138 *  - BASIC SUBROUTINES                                                *
                                  139 *                                                                     *
                                  140 *  - MESSAGES FROM THE SIMULATOR                                      *
                                  141 *                                                                     *
                                  142 *  - INSERT AND AUTOMATIC CARD LOAD OPERATIONS                        *
                                  143 *  ALTHOUGH THESE ROUTINES AND THEIR CORRESPONDING  SUBROUTINES  ARE  *
                                  144 *  RELATED TO CPU OPERATIONS, THEY ARE INCLUDED IN THE CONSOLE SIMUL- *
                                  145 *  ATION SINCE THEY MAY BE USED IN CONSOLE KEY OPERATIONS.            *
                                  146 *                                                                     *
                                  147 ***********************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   149 ********************************************************************************
                                   150 *                                                                              *
                                   151 *                                SECTION 1                                     *
                                   152 *                CONSOLE KEYS, SWITCHES AND INDICATORS                         *
                                   153 *                                                                              *
                                   154 *  AN 8-BYTE FIELD, CONTAINING THE BYTES MENTIONED IN THE  FOLLOWING           *
                                   155 *  TABLE,  IS USED TO SIMULATE THE 1620 CONSOLE KEYS,  SWITCHES  AND           *
                                   156 *  INDICATORS. EACH BIT OF A GIVEN BYTE CORRESPONDS TO A KEY, SWITCH           *
                                   157 *  OR INDICATOR.                                                               *
                                   158 *                                                                              *
                                   159 ********************************************************************************
000000 80                          161 KEYBIT   DC    X'80'                    (BYTE FOR KEY INTERLOCK)
                                   162 *               'KEYBIT' BIT 0 = 'STOP'  KEY WHEN ONE
                                   163 *                              = 'START' KEY WHEN ZERO
                                   164 *                        BIT 1 = 'SAVE'  KEY WHEN ONE
                                   165 *                        BIT 2 = 'CHECK RESET' KEY WHEN ONE
                                   166 *                        BIT 3 = 'RESET' KEY WHEN ONE
                                   167 *                        BIT 4 = 'INSERT'KEY WHEN ONE
                                   168 *                        BIT 5 = 'MODIFY'KEY WHEN ONE
                                   169 *                        BIT 6 = 'AUTOMATIC LOAD' KEY WHEN ONE
                                   170 *                        BIT 7 = MAP DUMP FEATURE (FOR MAINTENANCE
                                   171 *                                                  PURPOSES ONLY)
000001 00                          173 PROSWI   DC    X'0'                        SWITCHES
                                   174 *               'PROSWI' BIT 0 = PROGRAM SWITCH 1
                                   175 *                        BIT 1 = PROGRAM SWITCH 2
                                   176 *                        BIT 2 = PROGRAM SWITCH 3
                                   177 *                        BIT 3 = PROGRAM SWITCH 4
                                   178 *                        BIT 4 = DISK CHECK SWITCH
                                   179 *                        BIT 5 = WRITE ADDRESS SWITCH FOR 1311 DISK
                                   180 *                        BIT 6 = I/O CHECK SWITCH
                                   181 *                        BIT 7 = O'FLOW SWITCH

000002 00                          183 NYCHEK   DC    X'0'                        INDICATORS
                                   184 *               'NYCHEK' BIT 0 = READ  CHECK INDICATOR 06
                                   185 *                        BIT 1 = WRITE CHECK INDICATOR 07
                                   186 *                        BIT 2 = NOT USED EXCEPT FOR PAPER TAPE
                                   187 *                        BIT 3 = NOT USED
                                   188 *                        BIT 4 = PRINTER CHECK INDICATOR 25
                                   189 *                        BIT 5 = ADDRESS CHECK INDICATOR 36
                                   190 *                        BIT 6 = WLR,RBC       INDICATOR 37
                                   191 *                        BIT 7 = CYL. OVERFLOW INDICATOR 38

000003 00                          193 HPEZ     DC    X'0'                        H/P AND E/Z INDICATORS
                                   194 *               'HPEZ'   BITS 0,1,2 NOT USED
                                   195 *                        BIT 3 = HIGH/POSITIVE INDICATOR 11
                                   196 *                        BITS 4,5,6 NOT USED
                                   197 *                        BIT 7 = EQUAL / ZERO  INDICATOR 12

000004 00                          199 OFLOW    DC    X'0'                        ARITH. AND EXP. CHECK INDICATORS
                                   200 *               'OFLOW'  BITS 0,1,2 NOT USED
                                   201 *                        BIT 3 = ARITHMETIC CHECK INDICATOR 14
                                   202 *                        BITS 4,5,6 NOT USED
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   203 *                       BIT 7 = EXPONENT   CHECK INDICATOR 15

000005 00                          205 LASTCD   DC    X'0'                    LAST CARD INDICATOR
                                   206 *               'LASTCD' BITS 0 TO 6 NOT USED
                                   207 *                       BIT 7 = LAST CARD        INDICATOR 09

000006 00                          209 IXBAND   DC    X'0'                    BAND INDICATORS
                                   210 *               'IXBAND' BITS 0 TO 4 NOT USED
                                   211 *                       BIT 5 = NO BAND          INDICATOR 30
                                   212 *                       BIT 6 = BAND 1           INDICATOR 31
                                   213 *                       BIT 7 = BAND 2           INDICATOR 32

000007 00                          215 TAPE43   DC    X'0'                    1443 PRINTER TAPE INDICATORS
                                   216 *               'TAPE43' BITS 0 TO 4 NOT USED
                                   217 *                       BIT 5 = CHANNEL 9     INDICATOR 33
                                   218 *                       BIT 6 = CHANNEL 12    INDICATOR 34
                                   219 *                       BIT 7 = PRINTER BUSY  INDICATOR 35
                                   220          AIF   (NOT &DISK).NODSK1
                                   221 .NODSK1  ANOP
000008                             222 SATT     DS    0H                      ENABLE INTERRUPTIONS
000008 47F0 00D4         000D4     223          B       MESSWI+4              AND STOP SIMULATION
00000C 45F0 00F4         000F4     224 BRCC     BAL   15,MESTOP
000010 000001E2                    225          DC    A(READY)              SEND MESSAGE SIM20 READY
000014 00003002                    226 CORest   DC  A(CORES1)                ADDRESS OF 1620 CORE STORAGE


                                   228          AIF   (&CORE GT 2).CORE1          1620 = 20K
000018 00004E1F                    229 CAPACT   DC    F'19999'                CAPACITY OF 1620
                                   230          AGO   .COREX
                                   231 .COREX   ANOP
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                      ASM H V 05 22.10 05/07/81

                                    234 **********************************************************************
                                    235 *                                                                    *
                                    236 *                              SECTION 2                             *
                                    237 *                BASIC CONSOLE SIMULATION  SUBROUTINES               *
                                    238 *                                                                    *
                                    239 **********************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   241 ********************************************************************************
                                   242 *                                                                              *
                                   243 *                                'TYPIO'                                       *
                                   244 *                                                                              *
                                   245 *  THIS SUBROUTINE SUBMITS ALL I/O REQUESTS RELATED TO THE 1052 PRIN-          *
                                   246 *  TER-KEYBOARD TO THE CONTROL PROGRAM. IN ADDITION, IT SUBMITS  I/O           *
                                   247 *  REQUESTS RELATED TO THE PAPER TAPE READER/PUNCH, WHEN THIS OPTION           *
                                   248 *  IS PRESENT.                                                                 *
                                   249 *                                                                              *
                                   250 *  OPERATION                                                                   *
                                   251 *  THE TRAILER RECORD IMMEDIATELY FOLLOWING THE  INSTRUCTION   WHICH           *
                                   252 *  CALLS THE 'TYPIO' SUBROUTINE CONTAINS THE ADDRESS OF THE  DEVICE,           *
                                   253 *  (1052 OR PAPER TAPE READER/PUNCH), THE NUMBER OF BYTES TO BE PRO-           *
                                   254 *  CESSED,  AND THE BUFFER ADDRESS.  IT IS TRANSMITTED TO 'TYPIO'              *
                                   255 *  SUBROUTINE WHICH USES IT TO BUILD UP THE I/O REQUEST.                       *
                                   256 *                                                                              *
                                   257 *  PROCESSING OF THE INSTRUCTION IS STOPPED AS SOON AS THE   I/O RE-           *
                                   258 *  QUEST AND CONTINUE IS ACCEPTED AND RESTARTS AT 'DEVICE END'.                *
                                   259 *                                                                              *
                                   260 *  THERE ARE TWO EXITS =                                                       *
                                   261 *  - A NORMAL EXIT (DEVICE END + CHANNEL END) WHICH RETURNS  TO  THE           *
                                   262 *    CALLING SEQUENCE (+ 6 BYTES)                                              *
                                   263 *  - AN ERROR EXIT (DEVICE END ONLY) WHICH RETURNS TO THE CALLING SE-          *
                                   264 *    QUENCE (+ 10 BYTES)                                                       *
                                   265 *                                                                              *
                                   266 ********************************************************************************


00001C                             268 TYPIOXX  DS    0H
00001C 5010 005C             0005C 269          ST    R1,TYPIOSV
000020 0000 0000             00000 270          ICM   R12,B'1111',0(R1)
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
000024 0000                        271          SR    R11,R11           CLEAR LENGTH REG
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
000026 0000 0000             00000 272          IC    R11,0(R1)         GET THE TYPEOUT LENGTH
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
                                  273          WRTERM  (R12),(R11)
00002A 0700                        274+         CNOP  0,4                                                    01-WRTER
00002C                             275+         DS    0H                                                     01-WRTER
00002C 0000 0000             00000 276+         ST    R12,DMS0001B       STORE MESSAGE-ADDRESS               01-WRTER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
000030 9201 0044       00044       277+         MVI   DMS0001B,X'01'     RESTORE FLAG                        01-WRTER
000034 0000 0000             00000 278+         STH   R11,DMS0001C+2           STORE LENGTH IN PLIST          01-WRTER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
000038 4510 004C             0004C 279+         BAL   1,DMS0001E      POINT R1 TO PLIST                    01-WRTER
00003C E3E8D7D3C9D54040            280+DMS0001A     DC    CL8'TYPLIN'                                        01-WRTER
000044 01000000                    281+DMS0001B     DC    X'01',AL3(R12)                                     01-WRTER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81

000048 C2000001                   282+DMS0001C     DC   C'B',X'00',AL2(01)                              01-WRTER
00004C                            283+DMS0001E     DS   0H                                          01-WRTER
00004C 0ACA                       284+         SVC   202           CALL CMS TO TYPE                 01-WRTER
00004E 00000052                   285+         DC    AL4(*+4)                                       01-WRTER
000052 5810 005C           0005C  286          L     R1,TYPIOSV
000056 47F1 0004           00004  287          B     4(R1)
00005A 0000
00005C 00000000                   288 TYPIOSV  DC    F'0'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   291 ********************************************************************************
                                   292 *                                                                              *
                                   293 *                                'ALARM'                                       *
                                   294 *                                                                              *
                                   295 *  THIS SUBROUTINE ISSUES ALL ALARM COMMANDS REQUESTED DURING  SIMU-          *
                                   296 *  LATION EACH TIME THE 1052 IS IN THE 'INTERVENTION REQUIRED'STATE.           *
                                   297 *                                                                              *
                                   298 ********************************************************************************

000060 5010 0084           00084   300 ALARM    ST    R1,ALAREG              SAVE REGISTER R1
                                   301 *        BAL   R1,TYPIO               SEND AN ALARM REQUEST
000064 00090B01                    302          DC    X'00090B01'            *
000068 00000000                    303          DC    A(0)                 *                                       A
00006C 4700 0000           00000   304          BC    NOP,0                  *
000070 4110 0001           00001   305          LA    R1,1                   *
000074 8B10 0013           00013   306          SLA   R1,19                  *
000078 4610 0078           00078   307          BCT   R1,*                   WAIT FOR 5 SECONDS
00007C 5810 0084           00084   308          L     R1,ALAREG              RESTORE REGISTER R1
000080 07F1                        309          BR      R1                   RETURN TO CALLER

000084                             311 ALAREG   DS    1F                     USED TO SAVE REGISTER R1
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   313 ********************************************************************************
                                   314 *                                                                              *
                                   315 *                                SECTION 3                                     *
                                   316 *                       MESSAGES FROM THE SIMULATOR                            *
                                   317 *                         PROCESSING  OF MESSAGES                              *
                                   318 *                                                                              *
                                   319 ********************************************************************************


                                   321 ********************************************************************************
                                   322 *                                                                              *
                                   323 *                                'MESSAG'                                      *
                                   324 *                                                                              *
                                   325 *  THIS SUBROUTINE ISSUES ALL OUTPUT  MESSAGES REQUIRED BY THE SIMU-           *
                                   326 *  LATOR. THESE MESSAGES ARE WRITTEN ON THE 1052. 'MESSAG'  USES THE           *
                                   327 *  'TYPIO' AND 'ALARM' SUBROUTINES.                                            *
                                   328 *                                                                              *
                                   329 *  THIS SEQUENCE HAS THREE ENTRIES = 'MESSAG', 'MESTOP' AND 'MESTIR'.          *
                                   330 *                                                                              *
                                   331 *  'MESSAG' = THIS SEQUENCE WRITES A GIVEN MESSAGE ON THE 1052   AND           *
                                   332 *             THEN RETURNS TO THE CALLING SEQUENCE.                            *
                                   333 *                                                                              *
                                   334 *  'MESTOP' = THIS SEQUENCE  WRITES THE MESSAGE  AND THEN  SETS THE            *
                                   335 *             'STOP' BIT ON SO THAT SIMULATION STOPS UPON RETURN TO            *
                                   336 *             'BIR'                                                            *
                                   337 *                                                                              *
                                   338 *  'MESTIR' = THIS SEQUENCE  IS ONLY  USED WHEN  OPERATOR  INTERVEN-           *
                                   339 *             TION IS REQUIRED ON AN I/O DEVICE.THE MESSAGE IS WRIT-           *
                                   340 *             TEN, BUT THE INSTRUCTION COUNTER 'CNTR' IS NOT  INCRE-           *
                                   341 *             MENTED SO THAT SIMULATION RESUMES AT THE SAME POINT AS           *
                                   342 *             SOON AS THE DEVICE IS READY.                                     *
                                   343 *                                                                              *
                                   344 *  THE SEQUENCE WHICH IS COMMON TO THESE THREE ENTRY POINTS SENDS AN           *
                                   345 *  OUTPUT REQUEST TO THE TYPEWRITER THROUGH THE 'TYPIO'  SUBROUTINE.           *
                                   346 *  RECOVERY PROCEDURES ARE PROVIDED FOR EXCEPTIONAL CONDITIONS IN THE          *
                                   347 *  TYPEWRITER OPERATION (SEE ALSO 'EXCRET' AND 'ALARM' SUBROUTINES).           *
                                   348 *                                                                              *
                                   349 ********************************************************************************

000088 9209 00B0       000B0       351 MESSAG   MVI   MESSIO+6,X'09'          SET COMMAND = 'WRITE WITH AUTO-
                                   352 *                                        MATIC CARRIAGE RETURN'
00008C 96F0 00D1       000D1       353 MESSA2   OI    MESSWI+1,X'F0'          SET SWITCH 'MESSWI' TO BRANCH
000090 5010 0108             00108 354          ST    R1,MESSAV               SAVE REGISTER R1
000094 940F 00C1       000C1       355          NI    ALARSW+1,X'0F'          SET ALARM SWITCH OFF
000098 481F 0000             00000 356          LH    R1,0(15)                LOAD ADDRESS OF MESSAGE INTO R1
00009C D200 00B1 1000 000B1 00000 357          MVC   MESSIO+7(1),0(R1)       STORE COUNT IN MESSAGE REQUEST
0000A2 4111 0001             00001 358          LA    R1,1(R1)                STORE ADDRESS OF MESSAGE IN
0000A6 4010 00B2             000B2 359          STH   R1,MESSIO+8             IN MESSAGE REQUEST
0000AA                             360 MESSIO   DS    0H
                                   361 *        BAL   R1,TYPIO                ENTER MESSAGE REQUEST
0000AA 00090900                    362          DC    X'00090900'
0000AE 0000
0000B0 00000000                    363          DC    A(0)                *                                        A
0000B4 47F0 00CC             000CC 364          B       MESSOK                MESSAGE OK,EXIT
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

0000B8 4700 0000            00000   365          BC    NOP,0                  UE  SHOULD NOT OCCUR, ALARM
0000BC 47F0 0100            00100   366          B       ALAMES               INT.REQ.,GO TO ALARM
0000C0 4700 00BC            000BC   367 ALARSW   BC    NOP,*-4                SENSE EXIT +ALARM SWITCH
0000C4 96F0 00C1      000C1         368          OI    ALARSW+1,X'F0'         TRY AGAIN, THEN ALARM
0000C8 47F0 00AA            000AA   369          B       MESSIO               (SEE E R P)
0000CC 5810 0108            00108   370 MESSOK   L     R1,MESSAV              RESTORE REGISTER R1
0000D0 47FF 0004            00004   371 MESSWI   B     4(15)                OK,RETURN TO CALLER
0000D4 9680 0000      00000         372          OI    KEYBIT,X'80'           SET 'STOP' BIT FOR SEQUENCE
0000D8 47F0 07F6            007F6   373          B       ENTRY1               *
0000DC 50F0 0108            00108   374 MESTIR   ST    15,MESSAV              SAVE REGISTER R15
0000E0 0000 0000            00000   375          L     15,SAVUCB              GET ADDRESS OF DEVICE NAME
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0000E4 D203 012D F000 0012D 00000   376          MVC   INTREQ+12(4),0(15)     MOVE DEVICE NAME TO MESSAGE
0000EA 41F0 000C            0000C   377          LA    15,12                  R15=12
0000EE 185F                         378          SR    CNTR,15                CNTR=CNTR-12
0000F0 58F0 0108            00108   379          L     15,MESSAV              RELOAD REGISTER R15
0000F4 940F 00D1      000D1         380 MESTOP   NI    MESSWI+1,X'0F'         SPECIAL ENTRY WHEN MESSAGE STOPS
0000F8 9209 00B0      000B0         381          MVI   MESSIO+6,X'09'         SET COMMAND = 'WRITE WITH AUTO-
                                    382 *                                        MATIC CARRIAGE RETURN'
                                    383          AIF   (NOT &DISK).NODSK2       DISKS ARE PRESENT
                                    384 .NODSK2  ANOP
0000FC 47F0 0090            00090   385          B       MESSAG+8             *
000100 4510 0060            00060   386 ALAMES   BAL   R1,ALARM                SEND AN ALARM REQUEST
000104 47F0 0098            00098   387          B        MESSAG+16           AND RETRY


000108                              389 MESSAV   DS    1F                     *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   391 ********************************************************************************
                                   392 *                                                                              *
                                   393 *                    TABLE OF MESSAGES FROM THE SIMULATOR                      *
                                   394 *                                                                              *
                                   395 *  THE FOLLOWING MESSAGES CONSIST OF =                                         *
                                   396 *                                                                              *
                                   397 *  1. MESSAGES ISSUED BY THE SIMULATOR  PROPER  WHEN  IT  ENCOUNTERS           *
                                   398 *     CONDITIONS WHICH PREVENT CONTINUATION OF THE OPERATION                   *
                                   399 *                                                                              *
                                   400 *  2. NORMAL MESSAGES WHICH SIMULATE THE 1620 LIGHTS  OR  INDICATORS         *
                                   401 *     WHEN THEIR DISPLAY IS REQUIRED ON THE 1052 PRINTER-KEYBOARD              *
                                   402 *                                                                              *
                                   403 ********************************************************************************

00010C 14                          405 ERRTYP   DC    X'14'
00010D C1F2F5F1C140                406           DC    C'A251A '              CATASTROPHIC DAMAGE ON DEVICE
000113 C4C1D4C1C7C540D6            407           DC    C'DAMAGE ON XXXX'      CATASTROPHIC DAMAGE ON DEVICE
000121 0F                          408 INTREQ   DC    X'0F'
000122 C9D5E34BD9C5D840            409           DC    C'INT.REQ ON XXXX'     INTERVENTION REQUIRED ON DEVICE
000131 0C                          410 INVDEV   DC    X'0C'
000132 C9D5E540F1F6F2F0            411           DC    C'INV 1620 DEV'        INVALID 1620 DEVICE
00013E 0B                          412 RDRNOF   DC    X'0B'
00013F D9C4D940D5D640C6            413           DC    C'RDR NO FEED'         READER NO FEED
00014A 09                          414 LASCAR   DC    X'09'
00014B D3C1E2E340C3C1D9            415           DC    C'LAST CARD'           LAST CARD
000154 09                          416 PDLOK    DC    X'09'
000155 D761C440C9D5E3D3            417           DC    C'P/D INTLK'           PUNCH/DISK INTERLOCK
00015E 0C                          418 CHKDIS   DC    X'0C'
00015F C4C9E2D240C3C8C5            419           DC    C'DISK CHECK 0'        DISK CHECK
00016B 08                          420 UNTCHK   DC    X'08'
00016C C1F2F5F3C140                421           DC    C'A253A '              *
000172 E4C3                        422           DC    C'UC'
000174 16                          423           DC    X'16'
000175 C1F2F5F2C140                424           DC    C'A252A '              *
00017B F2F3F1F140C5D8E4            425           DC    C'2311 EQUIP CHECK'    EQUIPMENT CHECK
00018B 07                          426 EXP      DC    X'07'
00018C C5E7D740C3C8D2              427           DC    C'EXP CHK'             EXPONENT  OVERFLOW
000193 09                          428 AR       DC    X'09'
000194 C1D9C9E3C840C3C8            429           DC    C'ARITH CHK'           ARITHMETIC CHECK OVERFLOW
00019D 07                          430 NOBRAD   DC    X'07'
00019E D4C1D940C3C8D2              431           DC    C'MAR CHK'             MAR CHECK
0001A5 0B                          432 HLT      DC    X'0B'
0001A6 15                          433           DC    X'15'
0001A7 C8C1D3E36BF0F0F0            434           DC    C'HALT,00000'          HALT
0001B1 07                          435 INVXOP   DC    X'07'
0001B2 C9D5E540E7D6D7              436           DC    C'INV XOP'             INVALID INDEX OPERATION
0001B9 06                          437 WRICHK   DC    X'06'
0001BA E6D940C3C8D2                438           DC    C'WR CHK'              WRITE CHECK
0001C0 06                          439 PRICHK   DC    X'06'
0001C1 D7D940C3C8D2                440           DC    C'PR CHK'              PRINTER CHECK
0001C7 13                          441 CHK360   DC    X'13'                                         V1L2
0001C8 C1F2F5F4E640                442           DC    C'A254W '                                     V1L2
0001CE D7D9D6C7D9C1D440            443           DC    C'PROGRAM ERROR'                              V1L2
                         0019D     444 INVCOD   EQU   NOBRAD
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

                            0019D   445 INVP     EQU   NOBRAD
                            0019D   446 INVQ     EQU   NOBRAD
                            0019D   447 CHKSTP   EQU   NOBRAD
0001DB 06                           448 REDCHK   DC    X'06'
0001DC D9C440C3C8D2                 449          DC    C'RD CHK'           READ CHECK


0001E2 11                           451 READY    DC    FL1'17'
0001E3 C1F2F5F0C140                 452          DC    C'A250A '           *
0001E9 E2C9D4F2F040D9C5             453          DC    C'SIM20 READY'
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                          ASM H V 05 22.10 05/07/81

                                   455 ***********************************************************************
                                   456 *                                                                     *
                                   457 *                               SECTION  4                            *
                                   458 *           INSERT, AUTOMATIC CARD LOAD AND  READ OPERATIONS          *
                                   459 *                                                                     *
                                   460 ***********************************************************************



                                   463 ***********************************************************************
                                   464 *                                                                     *
                                   465 *                        INSERT KEY  OPERATIONS                       *
                                   466 *             AND READ NUMERICALLY  OPERATIONS ON TYPEWRITER          *
                                   467 *                                                                     *
                                   468 *  THIS IS AN INPUT OPERATION WHICH IS PERMANENTLY IN CORE  STORAGE,  *
                                   469 *  SINCE IT MAY BE USED BY AN 'INSERT' OPERATION.                     *
                                   470 *                                                                     *
                                   471 *  OPERATION                                                          *
                                   472 *  ADDRESSES ARE PROVIDED FOR THE 'MASK' SUBROUTINE TO  PREPARE  THE  *
                                   473 *  CODE CONVERSION TABLE.  THE FIELD EXPANSION SWITCH IN THE 'VALIN'  *
                                   474 *  SUBROUTINE IS SET OFF TO ALLOW DIRECT NUMERIC INPUT  INSTEAD   OF  *
                                   475 *  ALPHAMERIC  EXPANSION. THE COMMAND MASK CHARACTER  CAN BE  RECOG-  *
                                   476 *  NIZED BY 'MASK'.                                                   *
                                   477 *                                                                     *
                                   478 *  THE 'TYPIO' SUBROUTINE SUBMITS A REQUEST TO THE  CONTROL  PROGRAM  *
                                   479 *  FOR A READ OPERATION WITH A COUNT OF 100 BYTES. 'TYPIO'   RETURNS  *
                                   480 *  CONTROL TO 'RNTY' IN CASE OF =                                     *
                                   481 *                                                                     *
                                   482 *  - AN ERROR CONDITION WHICH BRANCHES TO THE  'EXCRET'  SUBROUTINE.  *
                                   483 *    'EXCRET' DISTINGUISHES BETWEEN 'UNIT EXCEPTION' ('CANCEL' CAUS-  *
                                   484 *    ING THE ENTIRE SEQUENCE TO BE RESUMED), 'INTERVENTION REQUIRED'  *
                                   485 *    (CAUSING AN 'ALARM' REQUEST TO BE SENT TO THE 1052),OR ANY SEN-  *
                                   486 *    SE INDICATION WHICH TESTS IF READ CHECK INDICATOR 06 IS INVOLV-  *
                                   487 *    ED.                                                              *
                                   488 *  - A NORMAL CONDITION.                                              *
                                   489 *                                                                     *
                                   490 *  THE SEQUENCE WHICH HANDLES THE NORMAL CONDITION COMPUTES THE  RE-  *
                                   491 *  SIDUAL COUNT TO DETERMINE IF THE READ OPERATION HAS  ENTERED  100  *
                                   492 *  DATA BYTES. IF LESS, THE EXIT SWITCH CONSIDERS THE READ OPERATION  *
                                   493 *  AS TERMINATED. IF 100 BYTES HAVE BEEN ENTERED WITHOUT AN 'EOB' IN- *
                                   494 *  DICATION, THEY WILL BE CONVERTED BY THE 'VALIN' SUBROUTINE,  THEN  *
                                   495 *  TRANSFERRED TO SIMULATED 1620 CORE STORAGE. THE ENTIRE READ OPER-  *
                                   496 *  ATION WILL BE RESUMED.                                             *
                                   497 *                                                                     *
                                   498 *  'INSERT' BIT - THE READ OPERATION ON THE 1052 MAY BE DUE TO AN IN- *
                                   499 *  SERT OPERATION. THEREFORE, THE EXIT SWITCH TESTS THE PRESENCE  OF  *
                                   500 *  AN 'INSERT' BIT.  IF PRESENT, THE EXIT SWITCH TERMINATES THE READ  *
                                   501 *  OPERATION AFTER 100 INPUT DATA BYTES AND RETURNS TO 'BIR' THAT IS  *
                                   502 *  IN THE 'STOPPED' STATE.                                            *
                                   503 *                                                                     *
                                   504 ***********************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

0001F4 41B0 058A            0058A   506 RNTY     LA    WR2,TABLE1          LOAD ADDRESS OF CODE CONVERSION
0001F8 41A0 05AE            005AE   507          LA    WR1,TBRNTY          TABLE
0001FC 9200 04B3      004B3         508          MVI   FLEXSW+1,X'00'       SET FIELD EXPANSION SWITCH = OFF
000200 9201 0511      00511         509          MVI   MASK+1,X'01'         SET COMMAND MASK = 01
000204 940F 0485      00485         510          NI    VAL1+1,X'0F'        SET SWITCH FOR CARRIAGE RETURN
000208                              511 RNTYGO   DS    0H
                                    512 *        BAL   R1,TYPIO            CALL TYPEWRITER
000208 00090AD2                     513          DC    X'00090AD2'         READ 210 BYTES INTO IOBUFF
00020C 000003A8                     514          DC  A(IOBUFF)
000210 47F0 0228            00228   515          B       RNTYOK            OK,EXIT
000214 47F0 0208            00208   516          B       RNTYGO            UE,  CANCEL, RETRY
000218 47F0 026A            0026A   517          B       RNTYAL            INT.REQ., SEND ALARM
00021C 0000 0000      00000         518          TM    TYPERR+1,X'30'      SENSE,TEST BITS 2+3
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
000220 4780 026A            0026A   519          BZ      RNTYAL            ZERO,CATASTROPHIC DAMAGE
000224 9680 0002      00002         520          OI    NYCHEK,X'80'        SET INDICATOR 06 = ON
000228 45F0 0510            00510   521 RNTYOK   BAL   15,MASK              LOOK FOR CODE CONVERSION TABLE
00022C 9200 025F      0025F         522          MVI   TYPSWI+1,X'00'       SET EXIT TO NORMAL
000230 41D0 00D2            000D2   523          LA    WR4,210             SET L= 210
000234 0000 0000            00000   524          LH    WR5,TYPCSW+6        LOAD RESIDUAL COUNT OF C S W
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
000238 19DE                         525          CR    WR4,WR5             IS RESIDUAL COUNT = 100
00023A 4780 024E            0024E   526          BE      TYPEND+4          YES,NOTHING TYPED,EXIT
00023E 12EE                         527          LTR   WR5,WR5             NO,IS RESIDUAL COUNT = 0
000240 4720 0248            00248   528          BP      *+8               NO, L IS LESS THAN 100,PROCESS
000244 92F0 025F      0025F         529          MVI   TYPSWI+1,X'F0'       YES,L=210,MODIFY EXIT
000248 1BDE                         530          SR    WR4,WR5             L=210-(CSW) IN WR4
00024A 45F0 047E            0047E   531 TYPEND   BAL   15,VALIN+4          CHECK VALIDITY OF INPUT DATA
00024E 9108 0000      00000         532          TM    KEYBIT,X'08'        IS 'INSERT' BIT ON
000252 4780 025E            0025E   533          BZ      TYPSWI            NO,CONTINUE
000256 94F7 0000      00000         534          NI    KEYBIT,X'F7'        YES,CLEAR 'INSERT BIT' AND STOP
00025A 47F0 0880            00880   535          B       KAUTO             READING
00025E 47F0 0208            00208   536 TYPSWI   B       RNTYGO            SWITCH 'TYPSWI', BRANCH FOR
                                    537 *                                    TERMINATION
000262 4510 07B0            007B0   538          BAL   R1,ENTRY3           INTERROGATE I/O INDICATORS
                                    539 *        B       ENTRY1            RETURN TO INT.LOOP(DELETED) V1L1
000266 47F0 07F6            007F6   540 RNCD2    B       ENTRY1            RETURN TO INTERPRETIVE LOOP V1L1
00026A 4510 0060            00060   541 RNTYAL   BAL   R1,ALARM            GO TO SEND AN ALARM REQUEST
00026E 47F0 0208            00208   542          B       RNTYGO            AND RETRY
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   544 ********************************************************************************
                                   545 *                                                                              *
                                   546 *                      AUTOMATIC CARD LOAD  KEY OPERATIONS                     *
                                   547 *                    AND READ NUMERICALLY OPERATIONS ON CARDS                  *
                                   548 *                                                                              *
                                   549 *  THIS IS ALSO AN INPUT OPERATION WHICH IS PERMANENTLY IN CORE STOR-           *
                                   550 *  AGE SINCE IT MAY BE USED BY AN 'AUTOMATIC LOAD' OPERATION.                  *
                                   551 *                                                                              *
                                   552 *  OPERATION                                                                   *
                                   553 *  ADDRESSES ARE PROVIDED FOR THE 'MASK' SUBROUTINE TO  PREPARE  THE           *
                                   554 *  CODE CONVERSION TABLE.  THE FIELD EXPANSION SWITCH IN THE 'VALIN'           *
                                   555 *  SUBROUTINE IS SET OFF TO ALLOW DIRECT NUMERIC INPUT  INSTEAD   OF           *
                                   556 *  ALPHAMERIC  EXPANSION. THE COMMAND MASK CHARACTER  CAN BE  RECOG-           *
                                   557 *  NIZED BY 'MASK'.                                                            *
                                   558 *                                                                              *
                                   559 *  THE SVC 1 CALLING SEQUENCE SUBMITS A REQUEST TO THE CONTROL PROG-           *
                                   560 *  RAM FOR A READ OPERATION WITH A COUNT OF 80 BYTES. AS SOON AS THE           *
                                   561 *  REQUEST IS ACCEPTED,  CONTROL IS GIVEN TO THE  'MASK' SUBROUTINE.           *
                                   562 *  THE FREE TIME IS USED TO PREPARE THE CODE CONVERSION TABLE IF  SO            *
                                   563 *  REQUIRED.  THEN, PROCESSING STOPS AT SWITCH 'RNLOCK'.                       *
                                   564 *                                                                              *
                                   565 *  AT CHANNEL END, THE CONTROL PROGRAM RETURNS CONTROL TO   'NRMRNC'           *
                                   566 *  WHICH SETS SWITCH 'RNLOCK' OFF AND RETURNS TO THE POINT OF INTER-           *
                                   567 *  RUPTION.                                                                    *
                                   568 *  THE 'VALIN' SUBROUTINE TRANSLATES THE INPUT DATA BYTES,    CHECKS           *
                                   569 *  THEM AND MOVES THEM TO CORE STORAGE.                                        *
                                   570 *                                                                              *
                                   571 *  IN CASE OF AN ERROR CONDITION, THE 'EXCRET' SUBROUTINE IS ENTERED=          *
                                   572 *  - 'INTERVENTION REQUIRED' WILL CAUSE THE MESSAGE 'RDR NO FEED' TO           *
                                   573 *    BE TYPED ON THE 1052.                                                     *
                                   574 *  - 'UNIT EXCEPTION' WILL CAUSE INDICATOR 09 TO BE SET ON  AND  THE           *
                                   575 *    MESSAGE 'LAST CARD' TO BE TYPED ON THE 1052.                              *
                                   576 *                                                                              *
                                   577 ********************************************************************************

000272 4180 058A           0058A   579 RNCD     LA    WR2,TABLE1          LOAD ADDRESS OF CODE CONVERSION
000276 41A0 0358           00358   580          LA    WR1,TBRNCD          TABLE
00027A 9200 04B3     004B3         581          MVI   FLEXSW+1,X'00'      SET FIELD EXPANSION SWITCH = OFF
00027E 9205 0511     00511         582          MVI   MASK+1,X'05'        SET COMMAND MASK = 05
000282 9200 02CD     002CD         583 RNCD1A   MVI   RNCSWI+1,X'00'       SET SWITCH OFF                V1L1
000286 070007000700                584          CNOP  4,8                 *
00028C                             585 RNCD1    DS    0H   SVC 1          I/O REQUEST AND INTERRUPT AT
00028C 000C                        586          DC    X'000C'               CHANNEL END FOR CARD READER
00028E 0000
000290 00000350                    587          DC    A(RCDCCW)           *
000294                             588 RNCERR   DS    C                   *
000295                             589          DS    3C                  *
000298                             590          DS    D                   *
0002A0                             591 RNCPSW   DS    D                   *
0002A8 000002E0                    592          DC    A(NRMRNC)           *
0002AC 000002F0                    593          DC    A(EXCRNC)           *
0002B0 92F0 02B9     002B9         594          MVI   RNLOCK+1,X'F0'      SET LOCK SWITCH ON
0002B4 45F0 0510           00510   595          BAL   15,MASK             ACCEPTED, LOOK FOR COD.CONV.TAB.
0002B8 47F0 02B8           002B8   596 RNLOCK   B       *                 LOCK SWITCH (INTERRUPT AT
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                    ASM H V 05 22.10 05/07/81

                                    597 *                                               CHANNEL END)
0002BC 957F 03A8       003A8        598          CLI   IOBUFF,X'7F'                    TEST IF LAST CARD
0002C0 4780 0342             00342  599          BE       RNCDUE+10                    YES,BRANCH                 V1L2
                                    600 *        BE       RNCDUE+6                     YES,BRANCH       (DELETED) V1L2
0002C4 41D0 0050             00050  601          LA    WR4,80                          SET COUNTER TO 80
0002C8 45F0 047A             0047A  602          BAL   15,VALIN                        CHECK VALIDITY AND MOVE DATA
0002CC 4700 0266             00266  603 RNCSWI   BC    0,RNCD2                         SWITCH TO SPLIT MESSAGE    V1L1
                                    604 *        B        TYPSWI+4                     INTERROG.I/O INDIC(DELETED) V1L1
0002D0 9180 0002       00002        605          TM    NYCHEK,X'80'                    IS INDICATOR 06 ON         V1L1
0002D4 4780 0266             00266  606          BC    8,RNCD2                         NO, BRANCH                 V1L1
0002D8 4110 0266             00266  607          LA    R1,RNCD2                        YES                        V1L1
0002DC 47F0 031C             0031C  608          B        EXCRN2                       MESSAGE 'READ CHECK'       V1L1

0002E0 9200 02B9       002B9        610 NRMRNC   MVI   RNLOCK+1,X'00'                  SET LOCK SWITCH OFF AND RETURN
0002E4 92FF 02A0       002A0        611          MVI   RNCPSW,X'FF'                    TO POINT OF INTERRUPTION
0002E8 0700                         612          CNOP  2,4                             *
                                    613 *        SVC   3                               RETURN TO POINT OF INTERRUPTION
0002EA 0000
0002EC 000002A0                     614          DC    A(RNCPSW)                       *

0002F0 5010 034C             0034C  616 EXCRNC   ST    R1,RNCSAV                       SAVE R1                    V1L2
0002F4 47F0 0338             00338  617          B       RNCDUE                        UE, LAST CARD INDICATION
0002F8 47F0 032C             0032C  618          B       RNCDIR                        INT.REQ.,GO TO SEND MESSAGE
0002FC 9108 0295       00295        619          TM    RNCERR+1,X'08'                  TEST SENSE BIT 4           V1L1
000300 4780 0308             00308  620          BC    8,EXCRN3                        IF ZERO BY PASS            V1L1
000304 47F0 0314             00314  621          B        EXCRN4                       MESSAGE 'READ CHECK'       V1L1
000308 9130 0295       00295        622 EXCRN3   TM    RNCERR+1,X'30'                  TEST SENSE BIT 2+3         V1L1
00030C 0000 0000             00000  623          BZ       EXCR3                        IF NONE,CATASTROPHIC DAMAGE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
000310 9680 0002       00002        624          OI    NYCHEK,X'80'                    IF SOME,SET INDICATOR 06 ON
                                    625 *        BAL   R1,ENTRY3                                        (DELETED) V1L1
                                    626 *        B       NRMRNC                                         (DELETED) V1L1
000314 4110 0324             00324  627 EXCRN4   LA    R1,EXCRN1                       RETURN ADDRESS             V1L1
000318 92F0 02CD       002CD        628          MVI   RNCSWI+1,X'F0'                  SET SWITCH ON              V1L1
00031C 41F0 01DB             001DB  629 EXCRN2   LA    15,REDCHK                       TYPE MESSAGE 'READ CHECK' + V1L1
000320 47F0 07E2             007E2  630          B        DISIO                        SET BIT STOP ON            V1L1
000324 5810 034C             0034C  631 EXCRN1   L     R1,RNCSAV                       RESTORE R1                 V1L2
000328 47F0 02E0             002E0  632          B       NRMRNC                                                   V1L2
                                    633 *EXCRN1  B       NRMRNC                                         (DELETED) V1L2

00032C 5810 034C             0034C  635 RNCDIR   L     R1,RNCSAV                       RESTORE R1                 V1L2
000330 45F0 00DC             000DC  636          BAL   15,MESTIR                        SEND MESSAGE READER NO FEEDV1L2
                                    637 *RNCDIR  BAL   15,MESTIR                        SEND MESSAGE READ(DELETED) V1L2
000334 0000013E                     638          DC  A(RDRNOF)                         TO OPERATOR AND STOP          A

000338 5810 034C             0034C  640 RNCDUE   L     R1,RNCSAV                       RESTORE R1                 V1L2
00033C 45F0 0088             00088  641          BAL   15,MESSAG                       SEND MESSAGE 'LAST CARD'   V1L2
                                    642 *RNCDUE  BAL   15,MESSAG                       SEND MESSAGE 'LAST(DELETED) V1L2
000340 0000014A                     643          DC  A(LASCAR)                         TO OPERATOR                   A
000344 9601 0005       00005        644          OI    LASTCD,X'01'                    SET LAST CARD INDICATOR 09 ON
000348 47F0 0282             00282  645          B        RNCD1A                       LOOP ON READ
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
00034C                             646 RNCSAV   DS    F                                                    V1L2

000350 020003AB20000050            648 RCDCCW   CCW   X'02',IOBUFF,X'20',80   READ 80 BYTES (SLI FLAG)

000358 4000F0                      650 TBRNCD   DC    X'4000F0'
00035B 4B05FBFCAAAADFF0            651          DC    X'4B05FBFCAAAADFF0'
000363 5A02DADBDC                  652          DC    X'5A02DADBDC'
000368 6001D0F1                    653          DC    X'6001D0F1'
00036C 6B04FBFCAAAAFF              654          DC    X'6B04FBFCAAAAFF'
000373 7B01FBFC                    655          DC    X'7B01FBFC'
000377 C009F0F1F2F3F4F5            656          DC    X'C009F0F1F2F3F4F5F6F7F8F9'
000383 D000D0                      657          DC    X'D000D0'
000386 DA05AAAAAAAAAAAA            658          DC    X'DA05AAAAAAAAAAAA'
00038E E009FAAAF2F3F4F5            659          DC    X'E009FAAAF2F3F4F5F6F7F8F9'
00039A FA05AAAAAAAAAAAA            660          DC    X'FA05AAAAAAAAAAAA'
0003A2 EF                          661          DC    X'EF'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                ASM H V 05 22.10 05/07/81
                                   664 ********************************************************************
                                   665 *                                                                  *
                                   666 *                        INPUT/OUTPUT BUFFER                       *
                                   667 *                                                                  *
                                   668 ********************************************************************

0003A8                             670 IOBUFF   DS    0D
0003A8 0000000000000000            671          DC    210X'0'
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
                                673 ********************************************************************************
                                674 *                                                                              *
                                675 *                          'VALIN' SUBROUTINE                                  *
                                676 *                                                                              *
                                677 *  THIS SUBROUTINE PERFORMS CODE CONVERSION AND VALIDITY CHECKING IN           *
                                678 *  ALL INPUT OPERATIONS (INCLUDING 'INSERT' AND 'AUTOMATIC LOAD').             *
                                679 *  AT THE REQUEST OF THE CALLING SEQUENCE, IT CAN ALSO EXPAND   EACH           *
                                680 *  INPUT DATA BYTE INTO TWO INTERNAL STORAGE BYTES,  AS  REQUIRED IN           *
                                681 *  ALPHAMERIC READ OPERATIONS.                                                 *
                                682 *                                                                              *
                                683 *  OPERATION                                                                   *
                                684 *  THE INPUT BUFFER 'IOBUFF' IS SCANNED FOR TRANSLATION OF EACH  IN-           *
                                685 *  PUT DATA BYTE, USING THE CODE CONVERSION TABLE LOCATED AT ADDRESS            *
                                686 *  128.  AFTER CONVERSION,  EACH BYTE IS CHECKED FOR VALIDITY,  THEN           *
                                687 *  STORED AT THE P ADDRESS WHICH IS PROGRESSIVELY INCREMENTED.                 *
                                688 *                                                                              *
                                689 *  THE WRAP-AROUND FEATURE IS INCLUDED IN THIS SUBROUTINE.                     *
                                690 *                                                                              *
                                691 *  ALPHAMERIC READ OPERATIONS INVOLVE ADDITIONAL PROCESSING. THE CON-          *
                                692 *  VERTED BYTE IS STORED  IN A TWO-BYTE BUFFER,  THEN UNPACKED.  THE           *
                                693 *  TWO NUMERICS THUS OBTAINED ARE MOVED TO THE CURRENT P AND P+1 AD-           *
                                694 *  DRESSES TO SAVE POSSIBLE FLAGS.                                             *
                                695 *                                                                              *
                                696 *  IN CASE OF AN INVALID BYTE CODE, THIS SUBROUTINE SETS THE READ              *
                                697 *  CHECK INDICATOR 06 ON AND CONTINUES CONVERSION.                             *
                                698 *                                                                              *
                                699 *                                                                              *
                                700 ********************************************************************************

00047A 96F0 0485  00485       702 VALIN   OI    VAL1+1,X'F0'        SET SWITCH FOR CARRIAGE RETURN
00047E 41A0 03A8        003A8 703         LA    WR1,IOBUFF          LOAD ADDRESS OF I/O BUFFER
000482 1880                   704         LR    WR2,0               CLEAR REGISTER WR2
000484 4700 049E        0049E 705 VAL1    BC    NOP,VALAAB          CARRIAGE RETURN SWITCH
000488 9108 0000  00000       706         TM    KEYBIT,X'08'        IS INSERT BIT PRESENT
00048C 4780 0496        00496 707         BC    8,*+10              NO ,BRANCH
000490 4980 04F4        004F4 708         CH    RP,LIMITX           YES,TEST RP EQUAL TO MAPORG+100
000494 07AF                   709         BCR   10,15               YES,RETURN TO CALLER
000496 9515 A000  00000       710         CLI   0(WR1),X'15'        NO,IS A CARRIAGE RETURN
00049A 4780 04C4        004C4 711         BE      VAL2              YES,BRANCH
00049E 43BA 0000        00000 712 VALAAB  IC    WR2,0(WR1)          NO,INSERT DATA BYTE IN WR2
0004A2 43BB 0080        00080 713         IC    WR2,CONVTB(WR2)     CONVERT IT TO INTERNAL CODE
0004A6 42B0 04F1        004F1 714         STC   WR2,BUFF2+1         STORE IN RIGHTMOST POSITION
0004AA 95AA 04F1  004F1       715         CLI   BUFF2+1,X'AA'       IS DATA VALID
0004AE 4780 04E8        004E8 716         BE      CODINV            YES,BRANCH CODINV
0004B2 4700 04D4        004D4 717 FLEXSW  BC    NOP,EXPAND          NO, FIELD EXPANSION SWITCH
0004B6 42B8 0000        00000 718         STC   WR2,0(RP)           *
0004BA 41B8 0001        00001 719         LA    RP,1(RP)            RP = RP + 1
0004BE 1987                   720         CR    RP,SIZE             IS RP GREATER THAN SIZE
0004C0 4720 04CE        004CE 721         BH      VAWR              YES,BRANCH TO VAWR
0004C4 41AA 0001        00001 722 VAL2    LA    WR1,1(WR1)          NO,WR1=WR1+1
0004C8 46D0 0484        00484 723         BCT   WR4,VAL1            LOOP IF COUNT NOT EQUAL TO ZERO
0004CC 07FF                   724         BR      15                RETURN TO CALLER
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
0004CE 1886                        726 VAWR     LR    RP,MAPORG             LOAD RP WITH MAPORG
0004D0 47F0 04C4            004C4  727          B       VAL2                RETURN TO VAL2
0004D4 F111 04F0 04F0 004F0 004F0  728 EXPAND   MVO   BUFF2(2),BUFF2(2)    SHIFT DATA BYTE TO THE LEFT
0004DA D101 8000 04F0 00000 004F0  729          MVN   0(2,RP),BUFF2         MOVE 2 NUMERICS TO CORE STORAGE
0004E0 4188 0001            00001  730          LA    RP,1(RP)              RP = RP + 1
0004E4 47F0 04BA            004BA  731          B       FLEXSW+8            *


0004E8 9680 0002      00002        733 CODINV   OI    NYCHEK,X'80'          SET READ CHECK INDICATOR ON
0004EC 47F0 04B2            004B2  734          B       FLEXSW              AND RETURN


0004F0 0000                        736 BUFF2    DC    X'0000'               TWO-BYTE BUFFER
0004F2 0000
0004F4 00003066                    737 LIMITX   DC  A(CORES1+100)          MAPORG+100                          A


0004F8 45F0 00F4            00F4   739 ERROR    BAL   15,MESTOP             BRANCH TO MESTOP
0004FC 0000019D                    740          DC  A(CHKSTP)           *                                       A
000500 47F0 0500            00500  741          B       *                                                     V1L2
000504 45F0 0088            00088  742 PRGERR   BAL   15,MESSAG             PROGRAM CHECK 360                   V1L2
000508 000001C7                    743          DC  A(CHK360)           STOP SIMULATION                     V1L2 A
00050C 47F0 050C            0050C  744          B       *                   AND WAIT                          V1L2
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                   746 ********************************************************************************
                                   747 *                                                                              *
                                   748 *                          'MASK' SUBROUTINE                                   *
                                   749 *                                                                              *
                                   750 *  THIS SUBROUTINE HANDLES ANY REQUEST FOR A CODE  CONVERSION  TABLE           *
                                   751 *  COMING FROM AN I/O OPERATION. THERE IS ONLY  ONE  CODE CONVERSION           *
                                   752 *  TABLE LOCATED AT ABSOLUTE ADDRESS 128 TO 384 (256 BYTES).HOWEVER,           *
                                   753 *  UP TO 18 DIFFERENT CODE CONVERSION TABLES CAN BE REQUIRED BY  THE           *
                                   754 *  1620 CONFIGURATION. CONSEQUENTLY, THE UNIQUE CODE CONVERSION TAB-           *
                                   755 *  LE MUST BE LOADED AS SOON AS REQUESTED BY THE 'MASK' SUBROUTINE.            *
                                   756 *                                                                              *
                                   757 *  OPERATION                                                                   *
                                   758 *  A SPECIFIC MASK BYTE IS PROVIDED TO THE 'MASK' SUBROUTINE  BY THE           *
                                   759 *  CALLING I/O OPERATION. THIS MASK IS COMPARED WITH A COMMAND CHECK           *
                                   760 *  BYTE TO CHECK IF THE CALLING I/O OPERATION IS DIFFERENT FROM  THE           *
                                   761 *  PRECEDING ONE.                                                              *
                                   762 *                                                                              *
                                   763 *  IF DIFFERENT, 'MASK' LOADS THE CODE CONVERSION  TABLES  WITH  THE           *
                                   764 *  DATA CORRESPONDING TO THE I/O OPERATION REQUESTED.  'MASK' STORES            *
                                   765 *  IN ITS COMMAND CHECK BYTE THE SPECIFIC MASK PROVIDED BY THE   I/O           *
                                   766 *  OPERATION.  THUS, IF THE FOLLOWING I/O OPERATION IS THE SAME, THE           *
                                   767 *  IDENTITY BETWEEN THE SPECIFIC MASK BYTE AND  THE  COMMAND   CHECK           *
                                   768 *  BYTE  IS ESTABLISHED AND LOADING OF THE TABLE WILL BE BYPASSED.             *
                                   769 *                                                                              *
                                   770 *  TWO ARGUMENTS  SUPPLIED IN REGISTERS WR1 AND WR2 BY THE   CALLING          *
                                   771 *  I/O OPERATION ARE REQUIRED FOR LOADING.  THE FIRST ONE IS THE AB-           *
                                   772 *  SOLUTE ADDRESS OF THE BASIC TABLE (TABLE 1, 2 OR 3). THE   SECOND          *
                                   773 *  ONE IS THE ABSOLUTE ADDRESS OF AN AUXILIARY TABLE  (FOR  INSTANCE           *
                                   774 *  'TBRNTY', 'TBRNCD'', 'TBPRN', ETC.)                                         *
                                   775 *                                                                              *
                                   776 *  THE BASIC TABLE CONTAINS THE MAXIMUM NUMBER OF BYTES COMMON  TO A           *
                                   777 *  GIVEN GROUP OF I/O OPERATIONS (EXAMPLE = 'TABLE1' FOR READ, WRITE           *
                                   778 *  AND DUMP NUMERICALLY, 'TABLE2' FOR READ AND WRITE ALPHAMERICALLY,           *
                                   779 *  'TABLE 3' FOR PRINT OPERATIONS).                                            *
                                   780 *                                                                              *
                                   781 *  THE AUXILIARY TABLE CONTAINS A VARIABLE NUMBER OF BYTES WHICH ARE           *
                                   782 *  ADDED TO THE PREVIOUSLY LOADED FIELDS OR THEIR SCRATCH PAD PARTS.           *
                                   783 *                                                                              *
                                   784 *  THE BASIC AND AUXILIARY TABLES CONSIST OF VARIABLE LENGTH FIELDS,           *
                                   785 *  THE LAST ONE BEING A HEXADECIMAL 'EE' BYTE FOR BASIC TABLES,  AND           *
                                   786 *  A HEXADECIMAL 'EF' BYTE FOR AUXILIARY TABLES.  ALL FIELDS, EXCEPT           *
                                   787 *  THE LAST ONE, CONTAIN =                                                     *
                                   788 *                                                                              *
                                   789 *  - ONE BYTE WHICH PROVIDES THE RELATIVE ADDRESS IN THE CODE CONVER-          *
                                   790 *    SION  TABLE AT WHICH LOADING OF THE DATA FIELD STARTS                     *
                                   791 *  - ONE BYTE WHICH CONTAINS THE ACTUAL LENGTH  (L - 1) OF THE  DATA           *
                                   792 *    FIELD TO BE LOADED                                                        *
                                   793 *  - 'L' BYTES WHICH FORM THE DATA FIELD PROPER.                               *
                                   794 *                                                                              *
                                   795 ********************************************************************************

000510 9500 0588       00588       797 MASK     CLI   COMDB,X'00'            IS COMMAND BYTE = MASK
000514 078F                        798          BER      15                  YES,RETURN TO CALLER
000516 92AA 0080       00080       799 MVTAB    MVI   CONVTB,X'AA'           RESET CODE CONVERSION TABLE
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81

00051A D2FE 0081 0080 00081 00080   800          MVC   CONVTB+1(255),CONVTB      . TO 'AA'                        ..
000520 90CE 0570            00570   801          STM   WR3,WR5,MASKRG           SAVE REGISTERS WR3 TO WR5
000524 18D0                         802          LR    WR4,0                    CLEAR WR4 AND WR5
000526 18E0                         803          LR    WR5,0                    *
000528 41C0 0080            00080   804          LA    WR3,CONVTB               WR3= A(TABLE TO BE BUILT)
00052C 95EE B000      00000         805 AL       CLI   0(WR2),X'EE'             *
000530 4780 0558            00558   806          BE      AP                     *
000534 95EF B000      00000         807          CLI   0(WR2),X'EF'             *
000538 4780 055E            0055E   808          BE      AQ                     *
00053C 43DB 0001            00001   809 AU       IC    WR4,1(WR2)               INSERT LENGTH OF MOVED FIELD
000540 43EB 0000            00000   810          IC    WR5,0(WR2)               INSERT ADDRESS OF IDENTIFICATION
000544 42E0 056D            0056D   811          STC   WR5,MLDT+3               STORE IT IN FIRST OPERAND
000548 41BB 0002            00002   812          LA    WR2,2(WR2)               WR2 = WR2 + 2
00054C 44D0 056A            0056A   813          EX    WR4,MLDT                 *
000550 41BD B001            00001   814          LA    WR2,1(WR4,WR2)           *
000554 47F0 052C            0052C   815          B       AL                     *
000558 18BA                         816 AP       LR    WR2,WR1                  *
00055A 47F0 053C            0053C   817          B       AU                     *
00055E D200 0588 0511 00588 00511   818 AQ       MVC   COMDB(1),MASK+1          SET COMMAND BYTE = MASK
000564 98CE 0570            00570   819          LM    WR3,WR5,MASKRG           LOAD REGISTERS WR3 TO WR5
000568 07FF                         820          BR      15                     RETURN TO CALLER

00056A D200 C000 B000 00000 00000   822 MLDT     MVC   0(1,WR3),0(WR2)

000570                              824 MASKRG   DS    6F
000588 0000                         825 COMDB    DC    X'0000'                  COMMAND CHECK BYTE



                                    828 ************************************************************************
                                    829 *                                                                      *
                                    830 *                DEFINITION OF CODE CONVERSION TABLES                  *
                                    831 *                                                                      *
                                    832 ************************************************************************

00058A D00A60D1D2D3D4D5             834 TABLE1   DC    X'D00A60D1D2D3D4D5D6D7D8D95A'
000597 DF004F                       835          DC    X'DF004F'
00059A F00AF0F1F2F3F4F5             836          DC    X'F00AF0F1F2F3F4F5F6F7F8F9E0'
0005A7 FC0040                       837          DC    X'FC0040'
0005AA FF006F                       838          DC    X'FF006F'
0005AD EE                           839          DC    X'EE'
0005AE 4000F0                       840 TBRNTY   DC    X'4000F0'                *
0005B1 4F00DF                       841          DC    X'4F00DF'                *
0005B4 5A00DA                       842          DC    X'5A00DA'
0005B7 6000D0                       843          DC    X'6000D0'
0005BA 6F00FF                       844          DC    X'6F00FF'
0005BD 7B01FAFC                     845          DC    X'7B01FAFC'
0005C1 D000AA                       846          DC    X'D000AA'
0005C4 DA05AAAAAAAAAAAA             847          DC    X'DA05AAAAAAAAAAAA'
0005CC FA05AAAAAAAAAAAA             848          DC    X'FA05AAAAAAAAAAAA'
0005D4 EF                           849          DC    X'EF'
0005D5 4B050304AAAA5F10             850 TABLE2   DC    X'4B050304AAAA5F10'
0005DD 5A075A131404AAAA             851          DC    X'5A075A131404AAAA2021'
0005E7 6B012324                     852          DC    X'6B012324'
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
0005EB 6F000F                      853          DC    X'6F000F'
0005EE 7B013334                    854          DC    X'7B013334'
0005F2 C009704142434445            855          DC    X'C009704142434445464748 49'
0005FE D009505152535455            856          DC    X'D009505152535455565758 59'
00060A E0090AAA62636465            857          DC    X'E0090AAA62636465666768 69'
000616 F009707172737475            858          DC    X'F009707172737475767778 79'
000622 EE                          859          DC    X'EE'
000623 000040                      860 TABLE3   DC    X'000040'
000626 03014B4C                    861          DC    X'03014B4C'
00062A 0A00E0                      862          DC    X'0A00E0'
00062D 0F016F50                    863          DC    X'0F016F50'
000631 13015B5C                    864          DC    X'13015B5C'
000635 1F054F6061AA6B6C            865          DC    X'1F054F6061AA6B6C'
00063D 33017B7C                    866          DC    X'33017B7C'
000641 4108C1C2C3C4C5C6            867          DC    X'4108C1C2C3C4C5C6C7C8C9'
00064C 500AD0D1D2D3D4D5            868          DC    X'500AD0D1D2D3D4D5D6D7D8D95A'
000659 6207E2E3E4E5E6E7            869          DC    X'6207E2E3E4E5E6E7E8E9'
000663 7009F0F1F2F3F4F5            870          DC    X'7009F0F1F2F3F4F5F6F7F8F9'
00066F EE                          871          DC    X'EE'
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
873 ********************************************************************************
874 *                                                                              *
875 *                                I/O SIMULATION                                *
876 *                                                                              *
877 *                               CODE  CONVERSION                               *
878 *                                                                              *
879 *  SIMULATION OF A 1620 I/O DEVICE REQUIRES CODE CONVERSION OF INPUT           *
880 *  DATA, SINCE THE BIT CONFIGURATION, AS REPRESENTED  IN  1620  CORE           *
881 *  STORAGE, CANNOT BE DIRECTLY ENTERED INTO SYSTEM/360 MAIN STORAGE.           *
882 *  INPUT DATA ARE THEREFORE FIRST ENTERED INTO AN INPUT BUFFER  AREA           *
883 *  FOR CONVERSION INTO A FORMAT ACCEPTABLE TO THE SIMULATOR.  OUTPUT           *
884 *  DATA WILL BE ENTERED INTO AN OUTPUT BUFFER AREA  FOR  CONVERSION            *
885 *  INTO AN APPROPRIATE FORMAT BEFORE BEING WRITTEN OR PUNCHED.                 *
886 *                                                                              *
887 ********************************************************************************


889 ********************************************************************************
890 *                                                                              *
891 *                            I/O OPERATION ENTRIES                             *
892 *                                                                              *
893 *  THIS SEQUENCE IS USED FOR ALL CONTROL, READ, WRITE AND DUMP OPERA-          *
894 *  TIONS.  IT IS ENTERED FROM THE BASIC INTERPRETIVE ROUTINE VIA THE           *
895 *  OPERATION CODE TABLE.                                                       *
896 *                                                                              *
897 *  ENTRY POINTS                                                                *
898 *  THERE ARE SIX ENTRIES CORRESPONDING TO THE I/O OPERATION CODES  =           *
899 *                                                                              *
900 *          K-34    DN-35    WN-38    RA-37    WA-39    RN-36                   *
901 *                                                                              *
902 *  AT EACH ENTRY POINT, CONTROL IS FIRST GIVEN TO  THE  'OUTIN' SUB-           *
903 *  ROUTINE TO TEST THE AVAILABILITY OF THE I/O SEQUENCE TO BE  USED.           *
904 *  THEN, THE ADDRESS OF THIS ROUTINE IS OBTAINED FROM THE  FOLLOWING           *
905 *  HALFWORD TABLE (SIMILAR TO THE OPERATION CODE TABLE), AND THE SE-           *
906 *  QUENCE BRANCHES TO THIS ADDRESS.                                            *
907 *                                                                              *
908 ********************************************************************************
```

```
                                912 ********************************************************************************
                                913 *                                                                              *
                                914 *                                  CONTROL                                     *
                                915 *                                                                              *
                                916 ********************************************************************************

000670 45E0 0780        00780   918 K34     BAL   WR5,OUTIN              COMMAND ANALYSIS
000674 47F0 04F8        004F8   919         B       SEE1                  BRANCH TO DISK OPERATIONS
000678 43DD 06EC        006EC   920         IC    WR4,WRITAB(WR4)         INSERT IN WR4 ADDRESS
00067C 48DD 070C        0070C   921         LH    WR4,KTABA(WR4)          OF 1620 INSTRUCTION
000680 07FD                     922         BR      WR4                   BRANCH TO INSTRUCTION



                                924 ********************************************************************************
                                925 *                                                                              *
                                926 *                            READ NUMERICALLY                                  *
                                927 *                                                                              *
                                928 ********************************************************************************

000682 45E0 0780        00780   930 RN36    BAL   WR5,OUTIN              COMMAND ANALYSIS
000686 47F0 04F8        004F8   931         B       RCDI1                 BRANCH TO DISK OPERATIONS
00068A 43DD 06FC        006FC   932         IC    WR4,RDTAB(WR4)          INSERT IN WR4 ADDRESS
00068E 48DD 0720        00720   933         LH    WR4,RNTAB(WR4)          OF 'READ NUMERICALLY'
000692 07FD                     934         BR      WR4                   BRANCH TO READ
                                935 ********************************************************************************
                                936 *                                                                              *
                                937 *                           WRITE NUMERICALLY                                  *
                                938 *                                                                              *
                                939 ********************************************************************************

000694 45E0 0780        00780   941 WN38    BAL   WR5,OUTIN              COMMAND ANALYSIS
000698 47F0 04F8        004F8   942         B       WDI1                  BRANCH TO DISK OPERATIONS
00069C 43DD 06EC        006EC   943         IC    WR4,WRITAB(WR4)         INSERT IN WR4 ADDRESS
0006A0 48DD 0730        00730   944         LH    WR4,WNTAB(WR4)          OF 'WRITE NUMERICALLY'
0006A4 07FD                     945         BR      WR4                   BRANCH TO WRITE
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                         ASM H V 05 22.10 05/07/81

                                    948 ***********************************************************************
                                    949 *                                                                     *
                                    950 *                       READ ALPHAMERICALLY                           *
                                    951 *                                                                     *
                                    952 ***********************************************************************

0006A6 45E0 0780            00780   954 RA37     BAL   WR5,OUTIN                 COMMAND ANALYSIS
0006AA 47F0 04F8            004F8   955          B        ERROR                  GO TO ERROR
0006AE 43DD 06FC            006FC   956          IC    WR4,RDTAB(WR4)            INSERT IN WR4 ADDRESS OF
0006B2 48DD 0744            00744   957          LH    WR4,RATAB(WR4)            OF 'READ ALPHAMERICALLY'
0006B6 4280 0F58            00F58   958 RA1      STC   RP,STC                    *
0006BA 9101 0F58      00F58         959          TM    STC,X'01'                 IS P ADDRESS EVEN
0006BE 4780 09F0            009F0   960          BZ       ERRORP                 NO,ERROR
0006C2 0680                         961          BCTR  RP,0                      RP=RP-1
0006C4 07FD                         962          BR       WR4                    YES,GO TO READ OR WRITE

                                    964 ***********************************************************************
                                    965 *                                                                     *
                                    966 *                       WRITE ALPHAMERICALLY                          *
                                    967 *                                                                     *
                                    968 ***********************************************************************

0006C6 45E0 0780            00780   970 WA39     BAL   WR5,OUTIN                 COMMAND ANALYSIS
0006CA 47F0 04F8            004F8   971          B        ERROR                  GO TO ERROR
0006CE 43DD 06EC            006EC   972          IC    WR4,WRITAB(WR4)           INSERT IN WR4 ADDRESS OF
0006D2 48DD 0758            00758   973          LH    WR4,WATAB(WR4)            'WRITE ALPHAMERICALLY'
0006D6 47F0 06B6            006B6   974          B        RA1                    BRANCH TO WRITE

                                    976          AIF (NOT &DISK).NODSK6
                                    977 .NODSK6  ANOP
                            004F8   978 SEE1     EQU   ERROR                     DISKS NOT EXISTING
                            004F8   979 WDI1     EQU   ERROR                     *
                            004F8   980 RCDI1    EQU   ERROR                     *
                                    981 .DSKX1   ANOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   983 ********************************************************************************
                                   984 *                                                                              *
                                   985 *                                DUMP                                          *
                                   986 *                                                                              *
                                   987 ********************************************************************************


0006DA 45E0 0780           00780   989 DN35     BAL   WR5,OUTIN              COMMAND ANALYSIS
0006DE 47F0 04F8           004F8   990          B        ERROR                GO TO ERROR
0006E2 43DD 06EC           006EC   991          IC    WR4,WRITAB(WR4)         INSERT IN WR4 ADDRESS OF
0006E6 48DD 076C           0076C   992          LH    WR4,DNTAB(WR4)          'DUMP'
0006EA 07FD                        993          BR       WR4                  BRANCH TO DUMP


0006EC 0002040006                  995 WRITAB   DC    X'0002040006'           TABLE FOR WRITE
0006F1 0000000008                  996          DC    X'0000000008'           *
0006F6 000000000000                997          DC    X'000000000000'         *
0006FC 000200040006                998 RDTAB    DC    X'000200040006'         TABLE FOR READ
000702 0000000000                  999          DC    X'0000000000'           *
000707 0000000000                 1000          DC    X'0000000000'           *
00070C 000004F8                   1001 KTABA    DC  A(ERROR)                TABLE FOR CONTROL                A
000710 00001F64                   1002          DC  A(KTYPE)                *                                A
000714 000004F8                   1003          DC  A(ERROR)                *                                A
000718 000004F8                   1004          DC  A(ERROR)                *                                A
00071C 00002290                   1005          DC  A(KPRINT)               *                                A
000720 000004F8                   1006 RNTAB    DC  A(ERROR)                TABLE FOR READ                   A
000724 000001F4                   1007          DC  A(RNTY)                 NUMERICALLY                      A
000728 00000A00                   1008          DC    A(RNPT)               *                                A
00072C 00000272                   1009          DC  A(RNCD)                 *                                A
000730 000004F8                   1010 WNTAB    DC  A(ERROR)                TABLE FOR WRITE                  A
000734 00001E7A                   1011          DC  A(WNTY)                 NUMERICALLY                      A
000738 00000A00                   1012          DC  A(WNPT)                 *                                A
00073C 00001FBA                   1013          DC  A(WNCD)                 *                                A
000740 000020CC                   1014          DC  A(PRN)                  *                                A
000744 000004F8                   1015 RATAB    DC  A(ERROR)                TABLE FOR READ                   A
000748 00001E44                   1016          DC  A(RATY)                 ALPHAMERICALLY                   A
00074C 00000A00                   1017          DC    A(RAPT)               *                                A
000750 00001FA2                   1018          DC  A(RACD)                 *                                A
000754 000004F8                   1019          DC  A(ERROR)                *                                A
000758 000004F8                   1020 WATAB    DC  A(ERROR)                TABLE FOR WRITE                  A
00075C 00001EF6                   1021          DC  A(WATY)                 ALPHAMERICALLY                   A
000760 00000A00                   1022          DC  A(WAPT)                 *                                A
000764 0000207C                   1023          DC  A(WACD)                 *                                A
000768 00002224                   1024          DC  A(PRA)                  *                                A
00076C 000004F8                   1025 DNTAB    DC  A(ERROR)                TABLE FOR DUMP                   A
000770 00001F2A                   1026          DC  A(DNTY)                 *                                A
000774 00000A00                   1027          DC  A(DNPT)                 *                                A
000778 0000209C                   1028          DC  A(DNCD)                 *                                A
00077C 00002258                   1029          DC  A(PRD)                  *                                A
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1031 *  DISKV SECTION DELETED
                                  1032           AGO   .NODISKV
                                  1033 .NODISKV ANOP     *-*      END
                                  1034 *   NISKV SECTION DELETED TOO
                                  1035           AGO   .NONISKV
                                  1036 .NONISKV ANOP    *-*      END
                                  1037 *+* NODSK                                  COMMON PART OF SIM20
                                  1038 ************************************************************************
                                  1039 *                                                                      *
                                  1040 *                          'OUTIN' SUBROUTINE                          *
                                  1041 *                                                                      *
                                  1042 *  OPERATION                                                           *
                                  1043 *  'OUTIN' CHECKS THE TYPE OF I/O OPERATION BLOCK PRESENT IN CORE      *
                                  1044 *  STORAGE. IT GIVES DIRECT CONTROL TO IT.                             *
                                  1045 *                                                                      *
                                  1046 ************************************************************************

000780 D100 07A9 5009 007A9 00009  1048 OUTIN    MVN   ACOMA+1(1),9(CNTR)      YES,INSERT IN WR4 BINARY
000786 48D0 07A8           007A8  1049           LH    WR4,ACOMA               EQUIVALENT OF Q9
00078A 910B 5001      00001       1050           TM    1(CNTR),X'0B'           TEST IF CONTROL COMMAND
00078E 4780 079E           079E   1051           BZ       DPROG                YES,BRANCH
000792 90AE 0570           00570  1052           STM   WR1,WR5,MASKRG          *
000796 45F0 0906           00906  1053           BAL   15,CONVP                NO,CONVERT P ADDRESS
00079A 98AE 0570           00570  1054           LM    WR1,WR5,MASKRG          *
00079E 9107 5009      00009       1055 DPROG    TM    9(CNTR),X'07'           IS Q9 = X'07'
0007A2 47CE 0004           00004  1056           BC    BZM,4(WR5)              NO,RETURN
0007A6 07FE                       1057           BR       WR5                  RETURN TO CALLER
                                  1058 *-*       END
                                  1059 *   NASKV SECTION DELETED
                                  1060           AGO .NONASKV
                                  1061 .NONASKV ANOP
0007A8 0000                       1062 ACOMA    DC    X'0000'                 *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1064 ********************************************************************************
                                  1065 *                                                                              *
                                  1066 *                          BASIC INTERPRETIVE ROUTINE                          *
                                  1067 *                                                                              *
                                  1068 *  THE BASIC INTERPRETIVE ROUTINE ('BIR') CONSISTS OF =                        *
                                  1069 *                                                                              *
                                  1070 *  - SWITCH INTERROGATION ROUTINE                                              *
                                  1071 *  - 'BIR' PROPER REQUIRED IN ALL CASES OF SIMULATION FOR   DECODING           *
                                  1072 *    OF THE 1620  INSTRUCTION                                                  *
                                  1073 *  - ROUTINES FOR CONSOLE KEY SIMULATION                                       *
                                  1074 *  - ADDRESS CONVERSION ROUTINES                                               *
                                  1075 *  - ERROR ROUTINES                                                            *
                                  1076 *                                                                              *
                                  1077 ********************************************************************************



                                  1079 ********************************************************************************
                                  1080 *                                                                              *
                                  1081 *                          SWITCH INTERROGATION                                *
                                  1082 *                                                                              *
                                  1083 *  THIS SEQUENCE CONTAINS THREE SUBROUTINES ('ENTRY4', 'ENTRY3', AND           *
                                  1084 *  'ENTRY0') WHICH INTERROGATE THE CONDITIONS= DISK CHECK, I/O CHECK            *
                                  1085 *  AND ARITHMETIC CHECK.                                                       *
                                  1086 *                                                                              *
                                  1087 ********************************************************************************


0007AA 1B43                       1089 ENTRY4   SR    SIMB2,SIMB1             DECREMENT BASE REGISTERS
0007AC 9200 0829      00829       1090          MVI   KSAVE+1,X'00'           SET KSAVE + 1 = X'00'
                                  1091          AIF (NOT &DISK).NODSK12
                                  1092 .NODSK12 ANOP


0007B0                            1094 ENTRY3   DS 0H SVC   9                        ENABLE INTERRUPTIONS
0007B0 9102 0001      00001       1095          TM    PROSWI,X'02'            TEST IF I/O SWITCH IS ON STOP
0007B4 4711 0000            00000 1096          BO       0(R1)                NO, RETURN TO CALLER
0007B8 9180 0002      00002       1097          TM    NYCHEK,X'80'            TEST IF 'READ CHECK'
0007BC 4780 07C8            007C8 1098          BZ       *+12                 IF NO, BY-PASS  3 INSTRUCTIONS
0007C0 41F0 01DB            001DB 1099          LA    15,REDCHK               IF YES,LOAD R15=A(MESSAGE)
0007C4 47F0 07E2            007E2 1100          B        DISIO                *
0007C8 9140 0002      00002       1101          TM    NYCHEK,X'40'            TEST IF 'WRITE CHECK'
0007CC 4780 07D8            007D8 1102          BZ       *+12                 *
0007D0 41F0 01B9            001B9 1103          LA    15,WRICHK               IF YES,LOAD R15=A(MESSAGE)
0007D4 47F0 07E2            007E2 1104          B        DISIO                *
0007D8 9108 0002      00002       1105          TM    NYCHEK,X'08'            TEST IF PRINTER CHECK
0007DC 0781                       1106          BZR      R1                   IF NO,RETURN TO CALLER
0007DE 41F0 01C0            001C0 1107          LA    15,PRICHK               IF YES,LOAD R15=A(MESSAGE)
0007E2 40F0 07EA            007EA 1108 DISIO    STH   15,*+8                  STORE ADDRESS OF MESSAG FOR
                                  1109 *                                          BRANCH AND LINK
0007E6 45F0 0088            00088 1110          BAL   15,MESSAG               DISPLAY INDICATOR
0007EA 0000
0007EC 00000000                   1111          DC    A(0)                *                                        A
0007F0 9680 0000      00000       1112 IOSWIT   OI    KEYBIT,X'80'              SET STOP BIT AND RETURN
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                ASM H V 05 22.10 05/07/81
0007F4 07F1                       1113          BR       R1                TO CALLER
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1116 ********************************************************************************
                                  1117 *                                                                              *
                                  1118 *                                    'BIR'                                     *
                                  1119 *                                                                              *
                                  1120 *  'BIR'  DECODES  EACH 1620 INSTRUCTION  LOCATED IN SIMULATED  CORE           *
                                  1121 *  STORAGE. THE ABSOLUTE ADDRESS OF THIS INSTRUCTION IS GIVEN BY RE-           *
                                  1122 *  GISTER 'CNTR'  WHICH  FUNCTIONS AS AN  INSTRUCTION  COUNTER.  THE           *
                                  1123 *  OPERATION CODE IS PACKED TOGETHER WITH THE P-ADDRESS OF THE  CUR-           *
                                  1124 *  RENT INSTRUCTION, THEN INSERTED IN REGISTER 1  AND MULTIPLIED  BY           *
                                  1125 *  2. THE VALUE THUS OBTAINED IS USED AS AN INDEX FOR THE  OPERATION           *
                                  1126 *  CODE TABLE 'OPTBL'.                                                         *
                                  1127 *                                                                              *
                                  1128 *  'BIR' HAS TWO ENTRY POINTS =                                                *
                                  1129 *                                                                              *
                                  1130 *  -ENTRY 1 IS USED TO INCREMENT THE INSTRUCTION COUNTER BY 12                 *
                                  1131 *                                                                              *
                                  1132 *  -ENTRY 2 IS USED TO BYPASS INCREMENTING (PRECEDING INSTRUCTION IS           *
                                  1133 *   AN UNCONDITIONAL BRANCH OR A CONDITIONAL  BRANCH WHOSE CONDITION           *
                                  1134 *   HAS BEEN SATISFIED).                                                       *
                                  1135 *   IT ALSO TESTS  THE PRESENCE OF  THE 'STOP' BIT  IN  THE 'KEYBIT'          *
                                  1136 *   BYTE. IF THIS BIT IS ABSENT (ZERO), THE NORMAL DECODING FUNCTION           *
                                  1137 *   OF 'BIR' IS EXECUTED.  IF PRESENT (ONE),  THE  'START'  EXIT  IS           *
                                  1138 *   TRANSFORMED TO NO OPERATION ('NOP').                                       *
                                  1139 *                                                                              *
                                  1140 *  NOTE= THE 'STOP' FUNCTION IS DIRECTLY PERFORMED BY A MANUAL OPER-           *
                                  1141 *        ATION ON THE  CONTROL PANEL  (SEE ALSO 'KEYBIT' BYTE  UNDER           *
                                  1142 *        CONSOLE SIMULATION.)                                                  *
                                  1143 *                                                                              *
                                  1144 *  OUTPUT      OPERATION CODE + P-ADDRESS, IN PACKED DECIMAL FORMAT,           *
                                  1145 *              IN 'BUFFP'                                                      *
                                  1146 *                                                                              *
                                  1147 ********************************************************************************

0007F6 4155 000C           0000C  1149 ENTRY1   LA    CNTR,12(CNTR)          INCREMENT INSTR. COUNTER BY 12
                                  1150 *ENTRY2  TM    KEYBIT,X'01'           TEST IF 'DUMP' BIT(DELETED) V1L2
                                  1151 *        BZ      *+8                  NO,BY-PASS2 INST  (DELETED) V1L2
                                  1152 *        BAL   15,DUMP1               YES,PRINT DUMP OFF(DELETED) V1L2
0007FA 9180 0000     00000        1153 ENTRY2   TM    KEYBIT,X'80'           TEST IF 'STOP' BIT          V1L2
                                  1154 *        TM    KEYBIT,X'80'           TEST IF 'STOP' BIT(DELETED  V1L2
0007FE 4780 0806           00806  1155          BZ      *+8                  NO, BY-PASS 2 INSTRUCTIONS
000802 9200 0819     00819        1156          MVI   START+1,X'00'          YES, STOP SIMULATION
000806 F276 09D8 5000 009D8 00000  1157 RESTAR   PACK  BUFFP(8),0(7,CNTR)    PACK OPER.CODE + P-ADDRESS
00080C 1810                       1158          LR    R1,0                   CLEAR REGISTER R1
00080E 4310 09DC           009DC  1159          IC    R1,BUFFP+4             INSERT OPERATION CODE AND FORM
000812 1A11                       1160          AR    R1,R1                  A HALFWORD ADDRESS
000814 4811 0A08           00A08  1161          LH    R1,OPTBL(R1)           GET ADDRESS OF CORRESP. ROUTINE
000818 47F0 1000           00000  1162 START    B       0(0,R1)              BRANCH TO CORREPONDING ROUTINE
00081C 9201 0900     00900        1163          MVI   SW,X'01'                                     V1L2
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1165 ************************************************************************
                                  1166 *                                                                      *
                                  1167 *                          SIMULATION OF KEYS                          *
                                  1168 *                                                                      *
                                  1169 *  KEY SIMULATION IS PERFORMED BY MEANS OF A LOOP CONSISTING OF SEV-   *
                                  1170 *  ERAL SEQUENCES WHICH ARE CONTINUOUSLY SCANNED AS SOON AS  SIMULA-   *
                                  1171 *  TION STOPS.  EACH OF THESE SEQUENCES GIVES CONTROL TO THE FOLLOW-   *
                                  1172 *  ING ONE  WHETHER  OR NOT ITS FUNCTION HAS BEEN FULFILLED.THE LAST   *
                                  1173 *  SEQUENCE  ('START KEY')  RETURNS  CONTROL EITHER TO THE FIRST SE-   *
                                  1174 *  QUENCE ('SAVE KEY') OR TO 'BIR'.                                    *
                                  1175 *                                                                      *
                                  1176 ************************************************************************


                                  1178 ************************************************************************
                                  1179 *                             SAVE KEY                                 *
                                  1180 *                                                                      *
                                  1181 *  THE SAVE FUNCTION IS ONLY OPERATIVE WHEN THE 'STOP'BIT IS PRESENT   *
                                  1182 *  IN THE  'KEYBIT' BYTE. IN THIS CASE, THE INSTRUCTION  COUNTER RE-   *
                                  1183 *  GISTER 'CNTR' IS  STORED IN FIELD 'PR1'.  THE  SEQUENCE  TURNS ON   *
                                  1184 *  A SWITCH WHICH  SIMULATES THE 'BRANCH BACK-42' INSTRUCTION.  THIS    *
                                  1185 *  ALLOWS NORMAL OPERATION OF THE SEQUENCE. CONTROL IS THEN GIVEN TO    *
                                  1186 *  THE FOLLOWING SEQUENCE.                                             *
                                  1187 *                                                                      *
                                  1188 ************************************************************************

                                  1190 *+* DISKV    DISKS RESIDENT              ON DISK PACK
                                  1191 *KSAVE    TM    KEYBIT,X'40'            TEST IF SAVE BIT
                                  1192 *-*      END
                                  1193 *+* NISKV                                 NO DISKV VERSION
                                  1194 *KSAVE    SR    0,0                     DECREMENT BASE REGISTER
000820 9200 0829       00829      1195          MVI   KSAVE+1,X'00'           SET KSAVE + 1 = X'00'
000824 9140 0000       00000      1196          TM    KEYBIT,X'40'            TEST IF SAVE BIT
                                  1197 *-*      END
                                  1198 *+* NODSK                                 COMMON PART OF SIM20
                                  1199 *KSAVE    TM    KEYBIT,X'40'            TEST IF SAVE BIT
                                  1200 *-*      END
                                  1201 *+* NASKV
000828 9140 0000       00000      1202 KSAVE    TM    KEYBIT,X'40'            TEST IF SAVE BIT IS PRESENT
                                  1203 *-*      END
00082C 4780 083C             083C 1204          BZ      KCHECK                NO, GO TO NEXT SEQUENCE
000830 5050 08FC             08FC 1205          ST    CNTR,PR1                YES, STORE COUNTER IN PR-1
000834 92F0 3109       01109      1206          MVI   BB42+1,X'F0'            SET SWITCH FOR BB42 = ON
000838 94BF 0000       00000      1207          NI    KEYBIT,X'BF'            CLEAR 'SAVE' BIT


                                  1209 ************************************************************************
                                  1210 *                                                                      *
                                  1211 *                          CHECK RESET KEY                             *
                                  1212 *                                                                      *
                                  1213 *  THE CHECK FUNCTION IS ONLY OPERATIVE WHEN A 'CHECK'BIT IS PRESENT   *
                                  1214 *  IN THE 'KEYBIT' BYTE. IN THIS CASE, CONTROL IS GIVEN TO THAT PART   *
                                  1215 *  OF THE 'RESET' SEQUENCE WHICH RESETS ONLY THE DISK CHECK  AND I/O   *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1216 *  CHECK INDICATORS. CONTROL IS THEN GIVEN TO THE NEXT SEQUENCE.      *
                                  1217 *                                                                     *
                                  1218 ***********************************************************************

00083C 9120 0000      00000       1220 KCHECK   TM    KEYBIT,X'20'            TEST IF 'CHECK RESET' BIT
000840 4780 084C            0084C 1221          BZ      KRESET                NO, GO TO NEXT SEQUENCE
000844 94DF 0000      00000       1222          NI    KEYBIT,X'DF'            YES, CLEAR 'CHECK RESET' BIT AND
000848 47F0 0864            00864 1223          B       KRESIO                GO TO 'RESET I/O INDICATORS'
                                  1224 ***********************************************************************
                                  1225 *                                                                     *
                                  1226 *                          RESET KEY                                  *
                                  1227 *                                                                     *
                                  1228 *  THE  RESET  FUNCTION IS ONLY OPERATIVE WHEN A 'STOP' BIT IS FOUND  *
                                  1229 *  IN THE  'KEYBIT' BYTE. IN THIS CASE, A POSSIBLE 'MODIFY'  BIT  IS  *
                                  1230 *  CLEARED FROM THE 'KEYBIT' BYTE. THE INSTRUCTION COUNTER  REGISTER  *
                                  1231 *  'CNTR' IS SET TO THE MAP ORIGIN ADDRESS AND  ALL  INDICATORS  ARE  *
                                  1232 *  CLEARED. CONTROL IS THEN GIVEN TO THE NEXT SEQUENCE.               *
                                  1233 *                                                                     *
                                  1234 ***********************************************************************

00084C 9110 0000      00000       1236 KRESET   TM    KEYBIT,X'10'            TEST IF 'RESET' BIT
000850 4780 0870            00870 1237          BZ      KINSER                NO, GO TO NEXT SEQUENCE
000854 94EF 0000      00000       1238          NI    KEYBIT,X'EF'            YES,CLEAR RESET BIT
000858 9200 0005      00005       1239          MVI   LASTCD,X'00'            RESET ALL INDICATORS
00085C 9200 0003      00003       1240          MVI   HPEZ,X'00'              *
000860 9200 0004      00004       1241          MVI   OFLOW,X'00'             *
000864 9420 0002      00002       1242 KRESIO   NI    NYCHEK,X'20'            *
                                  1243 *        MVI   IXBAND,X'04'            RESET TO 'NOBAND' (DELETED) V1L2
000868 9200 0007      00007       1244          MVI   TAPE43,X'00'            *
00086C 47F0 084C            0084C 1245          B       KRESET                RETURN TO RESET             V1L2



                                  1248 ***********************************************************************
                                  1249 *                                                                     *
                                  1250 *                          INSERT KEY                                 *
                                  1251 *                                                                     *
                                  1252 *  THIS FUNCTION IS ONLY  OPERATIVE WHEN THE 'INSERT'  BIT  IS  PRE-  *
                                  1253 *  SENT IN  THE 'KEYBIT' BYTE.  THE INSTRUCTION COUNTER 'CNTR',  AND  *
                                  1254 *  THE P ADDRESS REGISTER 'RP' ARE SET TO THE MAP ORIGIN ADDRESS AND  *
                                  1255 *  CONTROL IS GIVEN TO THE SEQUENCE SIMULATING THE'READ NUMERICALLY'  *
                                  1256 *  (TYPEWRITER) INSTRUCTION.                                          *
                                  1257 *                                                                     *
                                  1258 *  AT THE END OF THIS SEQUENCE,  THE INSERT BIT IS TESTED.  IF 'ON',  *
                                  1259 *  THE 'READ NUMERICALLY' ( TYPEWRITER )  INSTRUCTION WAS  CALLED BY  *
                                  1260 *  THE INSERT FUNCTION AND NOT BY THE 'RNTY-36' INSTRUCTION.          *
                                  1261 *  THUS,  CONTROL IS RETURNED TO THE FOLLOWING  KEY  SIMULATION  SE-  *
                                  1262 *  QUENCE.                                                            *
                                  1263 *                                                                     *
                                  1264 ***********************************************************************

000870 9108 0000      00000       1266 KINSER   TM    KEYBIT,X'08'            TEST IF 'INSERT' BIT
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                ASM H V 05 22.10 05/07/81

000874 4780 0880            00880  1267          BZ       KAUTO          NO, GO TO NEXT SEQUENCE
000878 1856                        1268          LR     CNTR,MAPORG      YES, RESET INSTRUCTION COUNTER
00087A 1886                        1269          LR     RP,MAPORG        RESET P-ADDRESS REGISTER
00087C 47F0 01F4            001F4  1270          B        RNTY           GO TO 'READ NUMERIC.TYPEWRITER'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1272 ********************************************************************************
                                  1273 *                                                                              *
                                  1274 *                          AUTOMATIC CARD LOADING KEY                          *
                                  1275 *                                                                              *
                                  1276 *  THIS FUNCTION IS ONLY OPERATIVE WHEN THE 'STOP' BIT IS PRESENT IN           *
                                  1277 *  THE 'KEYBIT' BYTE. THE INSTRUCTION COUNTER 'CNTR' IS SET TO   THE           *
                                  1278 *  VALUE OF MAP ORIGIN MINUS 12, AND THE P-ADDRESS REGISTER 'RP'  IS           *
                                  1279 *  SET TO THE VALUE OF MAP ORIGIN. ALL BITS IN THE 'KEYBIT' BYTE ARE           *
                                  1280 *  CLEARED AND THE 'BIR' SWITCH IS TURNED ON TO RESUME SIMULATION.             *
                                  1281 *  FINALLY, CONTROL IS GIVEN TO THE SEQUENCE SIMULATING  THE  'READ            *
                                  1282 *  NUMERICALLY' (CARD) INSTRUCTION.                                            *
                                  1283 *                                                                              *
                                  1284 *  AT THE END OF THIS SEQUENCE, THE AUTO-LOAD BIT IS TESTED.IF 'ON',           *
                                  1285 *  THE 'READ NUMERICALLY' (CARD)  INSTRUCTION WAS CALLED BY THE AUTO           *
                                  1286 *  LOAD FUNCTION AND NOT BY THE 'RNCD-36' INSTRUCTION.                         *
                                  1287 *  CONTROL IS THUS RETURNED TO THE FOLLOWING KEY SIMULATION  SEQUEN-           *
                                  1288 *  CE.                                                                         *
                                  1289 *                                                                              *
                                  1290 ********************************************************************************

000880 9102 0000       00000      1292 KAUTO    TM    KEYBIT,X'02'            TEST IF 'AUTOMATIC LOAD' BIT
000884 4780 089E             089E 1293          BZ      KSTART                NO, GO TO NEXT SEQUENCE
000888 9401 0000       00000      1294          NI    KEYBIT,X'01'            YES, CLEAR ALL BITS EXCEPT DUMP
00088C 4150 000C             000C 1295          LA    CNTR,12                 LOAD INSTRUCTION COUNTER WITH
000890 1155                       1296          LNR   CNTR,CNTR               MAPORG-12
000892 1A56                       1297          AR    CNTR,MAPORG             *
000894 1806                       1298          LR    RP,MAPORG               SET P-ADDRESS = MAP ORIGIN
000896 92F0 0819       00819      1299          MVI   START+1,X'F0'           RESTART SIMULATION
00089A 47F0 0272             00272 1300          B       RNCD                GO TO READ A CARD

                                  1303 ********************************************************************************
                                  1304 *                                                                              *
                                  1305 *                                START KEY                                     *
                                  1306 *                                                                              *
                                  1307 *  THE START FUNCTION IS ONLY OPERATIVE WHEN THE 'STOP' BIT HAS BEEN           *
                                  1308 *  SET ON IN THE 'KEYBIT' BYTE. THE START FUNCTION ALSO INTERROGATES           *
                                  1309 *  THE 'MODIFY' BIT IN THE SAME BYTE. IF THIS BIT IS ON, CLEARING OF           *
                                  1310 *  THE 1620 CORE STORAGE WAS PREVIOUSLY REQUESTED AND A SEQUENCE PER-          *
                                  1311 *  FORMS THE CLEARING IN GROUPS OF 250 BYTES UNTIL THE UPPER BOUNDARY          *
                                  1312 *  HAS BEEN REACHED. IF THIS BIT IS OFF,  ALL  BITS IN THE  'KEYBIT'           *
                                  1313 *  BYTE ARE CLEARED. TO RESUME SIMULATION, THE 'BIR' SWITCH  HAS  TO           *
                                  1314 *  BE TURNED ON SO THAT CONTROL CAN BE RETURNED TO 'BIR'.                      *
                                  1315 *                                                                              *
                                  1316 ********************************************************************************

00089E 9104 0000       00000      1318 KSTART   TM    KEYBIT,X'04'            TEST IF 'MODIFY' BIT
0008A2 4780 08D6             08D6 1319          BZ      KSTAR1                NO, GO TO NEXT SEQUENCE
0008A6 94FB 0000       00000      1320          NI    KEYBIT,X'FB'            YES, CLEAR 'MODIFY' BIT
0008AA 92F0 6000       00000      1321          MVI   0(MAPORG),X'F0'         CLEAR FIRST BYTE OF MEMORY MAP
0008AE D2F8 6001 6000 00001 00000 1322          MVC   1(249,MAPORG),0(MAPORG) CLEAR FIRST GROUP OF 250 BYTES
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

0008B4 18A6                      1323          LR    WR1,MAPORG            LOAD POINTER FOR CORE STORAGE
0008B6 41BA 00FA           000FA 1324          LA    WR2,250(WR1)          LOAD POINTER FOR UPPER BOUND
0008BA D2F9 A0FA A000 000FA 00000 1325 KCLEAR   MVC   250(250,WR1),0(WR1)  CLEAR FOLLOWING 250 BYTE GROUP
0008C0 41BB 00FA           000FA 1326          LA    WR2,250(WR2)          INCREMENT POINTER
0008C4 41AA 00FA           000FA 1327          LA    WR1,250(WR1)          INCREMENT   POINTER
0008C8 19B7                      1328          CR    WR2,SIZE              IS IT HIGHER THAN UPPER BOUNDARY
0008CA 0000 0000           00000 1329          BC    BNH,KCLEAR            NO,LOOP
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
0008CE 98BE 34D8           014D8 1330          LM    WR2,WR5,MCLEAR
0008D2 9680 0000      00000      1331          OI    KEYBIT,X'80'          YES, RESTORE 'STOP' BIT


0008D6 9101 0900      00900      1333 KSTAR1   TM    SW,X'01'                                    V1L2
0008DA 4710 08EA           008EA 1334          BO      KSTAR2                                    V1L2
                                 1335 *KSTAR1  TM    KEYBIT,X'80'          TEST IF 'STOP' BIT(DELETED) V1L2
                                 1336 *        BO      KSAVE               YES,LOOP          (DELETED) V1L2
0008DE 9401 0000      00000      1337          NI    KEYBIT,X'01'          NO, CLEAR ALL BITS EXCEPT 'DUMP'
0008E2 92F0 0819      00819      1338          MVI   START+1,X'F0'         AND RESTART SIMULATION.
0008E6 47F0 0806           00806 1339          B       RESTAR              GO TO 'BIR'
0008EA 9180 0000      00000      1340 KSTAR2   TM    KEYBIT,X'80'                                V1L2
0008EE 4710 081C           081C 1341          BO      START+4                                   V1L2
0008F2 9200 0900      00900      1342          MVI   SW,X'00'                                    V1L2
0008F6 47F0 0828           00828 1343          B       KSAVE                                     V1L2


0008FA 0000
0008FC 00000000                  1345 PR1      DC    F'0'                  PR-1 REGISTER
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
000900 00                        1347 SW       DC    X'00'                                          V1L2
                                 1348 ********************************************************************************
                                 1349 *                                                                              *
                                 1350 *                          ADDRESS CONVERSION                                  *
                                 1351 *                                                                              *
                                 1352 *  ADDRESS  CONVERSION HAS BEEN  SEPARATED FROM 'BIR', SINCE NOT ALL           *
                                 1353 *  1620 INSTRUCTIONS NEED BOTH THE P ADDRESS AND THE Q ADDRESS.                *
                                 1354 *                                                                              *
                                 1355 *  THERE MAY BE THREE DIFFERENT ADDRESS CONVERSION ROUTINES =                   *
                                 1356 *                                                                              *
                                 1357 *  - THE STANDARD ADDRESS ROUTINE                                              *
                                 1358 *  - THE INDIRECT ADDRESSING SUBROUTINE ('INDAD')                              *
                                 1359 *  - THE ADDRESS INDEXING SUBROUTINE ('INDEX') FOR MODEL 2 ONLY                *
                                 1360 *                                                                              *
                                 1361 *  ENTRIES                                                                     *
                                 1362 *  'CONVP', 'CONVPQ' AND 'CONVQ' FOR P ADDRESS CONVERSION ONLY, P+Q            *
                                 1363 *  ADDRESS CONVERSION, OR Q ADDRESS CONVERSION ONLY.                           *
                                 1364 *                                                                              *
                                 1365 *  EXITS                                                                       *
                                 1366 *  ALL ADDRESS CONVERSION ROUTINES RETURN TO THE SEQUENCE  WHICH RE-           *
                                 1367 *  QUIRED ADDRESS CONVERSION.                                                  *
                                 1368 *                                                                              *
                                 1369 *  EACH ROUTINE HAS TWO ADDITIONAL EXITS FOR INVALID  ADDRESSES.               *
                                 1370 *                                                                              *
                                 1371 *  OUTPUT       REGISTER RP CONTAINS THE ABSOLUTE P ADDRESS IN BINARY          *
                                 1372 *               FORMAT                                                         *
                                 1373 *               REGISTER RQ CONTAINS THE ABSOLUTE Q ADDRESS IN BINARY          *
                                 1374 *               FORMAT                                                         *
                                 1375 *                                                                              *
                                 1376 ********************************************************************************

                                 1378          AIF   (NOT &INDEX).NOTIX1     INDEX INSTRUCTIONS
000901 00
000902 940F 091D      0091D      1379 CONVPQ   NI    RET+1,X'0F'               SET SWITCH ON


000906 9200 09DC      009DC      1381 CONVP    MVI   BUFFP+4,X'00'             CLEAR OPERATION CODE
00090A 960F 09DF      009DF      1382          OI    BUFFP+7,X'0F'             SET PLUS SIGN
00090E 4FD0 09D8           009D8 1383          CVB   WR4,BUFFP                 CONVERT P ADDRESS TO BINARY
000912 41A5 0002           00002 1384          LA    WR1,2(CNTR)               WR1=CNTR+2
000916 45E0 095C           0095C 1385          BAL   14,INDEX                  GO TO 'INDEX' SUBROUTINE
00091A 188D                      1386          LR    RP,WR4                    RP=WR4
00091C 07FF                      1387 RET      BR       15                     RETURN IF SWITCH OFF
00091E F274 09E0 5007 009E0 00007 1388 CONVQ    PACK  BUFFQ(8),7(5,CNTR)        PACK Q ADDRESS IN BUFFQ
000924 960F 09E7      009E7      1389          OI    BUFFQ+7,X'0F'             SET PLUS SIGN
000928 4FD0 09E0           009E0 1390          CVB   WR4,BUFFQ                 CONVERT Q ADDRESS TO BINARY
00092C 41A5 0007           00007 1391          LA    WR1,7(CNTR)               WR1=CNTR+7
000930 45E0 095C           0095C 1392          BAL   14,INDEX                  GO TO 'INDEX' SUBROUTINE
000934 189D                      1393          LR    RQ,WR4                    RQ=WR4
000936 96F0 091D      0091D      1394          OI    RET+1,X'F0'               SET SWITCH OFF
00093A 07FF                      1395          BR       15                     RETURN TO CALLER
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
00093C 070E                      1397 INDAD    BCR   0,14                 INDIRECT ADDRESSING SWITCH
00093E 9120 A004      00004      1398          TM    4(WR1),X'20'         TEST FLAG
000942 071E                      1399          BCR   1,14                 IF NONE, BRANCH
000944 18AD                      1400          LR    WR1,WR4              WR1=WR4
000946 4BA0 09D0            009D0 1401          SH    WR1,DEC4             WR1=WR1-4
00094A F274 09E0 A000 009E0 00000 1402          PACK  BUFFQ(8),0(5,WR1)    PACK NEW ADDRESS
000950 960F 09E7      009E7      1403          OI    BUFFQ+7,X'0F'        SET PLUS SIGN
000954 4FD0 09E0            009E0 1404          CVB   WR4,BUFFQ            CONVERT TO BINARY
000958 47F0 095C            0095C 1405          B       INDEX              GO TO INDEX
                                  1406 ********************************************************************************
                                  1407 *                                                                              *
                                  1408 *                              INDEX SUBROUTINE                                *
                                  1409 *                                                                              *
                                  1410 *  THIS SUBROUTINE  MODIFIES THE P AND Q ADDRESSES  IF ANY FLAGS ARE           *
                                  1411 *  PRESENT IN THE THOUSANDS, HUNDREDS OR TENS  POSITION  OF THE  AD-           *
                                  1412 *  DRESS FIELD OF THE INSTRUCTION.  IT  COMPUTES THE  INDEX REGISTER           *
                                  1413 *  ADDRESS, ACCORDING TO THE BAND SELECTED.  IT THEN ADDS  THIS  AD-           *
                                  1414 *  DRESS TO THE ADDRESS SUBMITTED TO THE ROUTINE. IF  THE  RESULTING           *
                                  1415 *  ADDRESS IS NEGATIVE,  NINES COMPLEMENTING TAKES PLACE.                      *
                                  1416 *                                                                              *
                                  1417 *  AT THE END OF INDEXING, CONTROL IS GIVEN TO THE 'INDAD' SEQUENCE.           *
                                  1418 *  THE FINAL ADDRESSES ARE CONTAINED IN REGISTERS RP AND RQ.                   *
                                  1419 *                                                                              *
                                  1420 *  NOTE =                                                                      *
                                  1421 *  WHEN AN ADDRESS CONTAINS BOTH THE INDEXING AND INDIRECT  ADDRESS-           *
                                  1422 *  ING FEATURE,  PRIORITY IS GIVEN TO THE INDEX SUBROUTINE.                    *
                                  1423 *                                                                              *
                                  1424 ********************************************************************************

00095C 9104 0006      00006      1426 INDEX    TM    IXBAND,X'04'         IS 'NO BAND' INDICATOR ON
000960 4710 09C2            009C2 1427          BC    1,INDX5              YES,BRANCH
000964 18B0                      1428          LR    WR2,0                CLEAR WR2
000966 9120 A003      00003      1429          TM    3(WR1),X'20'         IS 3(WR1) DIGIT FLAGGED
00096A 4770 0972            00972 1430          BC    7,INDX1              NO,BRANCH
00096E 41BB 0005            00005 1431          LA    WR2,5(WR2)           YES,ADD 5 TO WR2
000972 9120 A002      00002      1432 INDX1    TM    2(WR1),X'20'         IS 2(WR1) DIGIT FLAGGED
000976 4770 097E            0097E 1433          BC    7,INDX2              NO,BRANCH
00097A 41BB 000A            0000A 1434          LA    WR2,10(WR2)          YES,ADD 10 TO WR2
00097E 9120 A001      00001      1435 INDX2    TM    1(WR1),X'20'         IS 1(WR1) DIGIT FLAGGED
000982 4770 098A            0098A 1436          BC    7,INDX3              NO,BRANCH
000986 41BB 0014            00014 1437          LA    WR2,20(WR2)          YES,ADD 20 TO WR2
00098A 12BB                      1438 INDX3    LTR   WR2,WR2              IS WR2 NULL
00098C 4780 09C2            009C2 1439          BC    8,INDX5              YES,BRANCH
000990 9102 0006      00006      1440          TM    IXBAND,X'02'         NO,IS 'BAND 1' INDICATOR ON
000994 4730 099C            0099C 1441          BC    3,INDX4              YES, BRANCH
000998 41BB 0028            00028 1442          LA    WR2,40(WR2)          NO,ADD 40 TO WR2
00099C 41B6 B12C            0012C 1443 INDX4    LA    WR2,300(MAPORG,WR2)  COMPUTE INDEX REGISTER ADDRESS
0009A0 070E                      1444          BCR   0,14                 RETURN SWITCH FOR IX INSTRUC.
0009A2 F274 09E0 B000 009E0 00000 1445          PACK  BUFFQ(8),0(5,WR2)    PACK ADDRESS
0009A8 4FC0 09E0            009E0 1446          CVB   WR3,BUFFQ            CONVERT TO BINARY
0009AC 1ADC                      1447          AR    WR4,WR3              COMPUTE NEW ADDRESS
0009AE 47A0 09B6            009B6 1448          BC    10,INDX6             BRANCH IF POSITIVE OR EQUAL
0009B2 5AD0 09CC            009CC 1449          A     WR4,MAXVAL           COMPUTE 9'S COMPLEMENT
0009B6 59D0 09CC            009CC 1450 INDX6    C     WR4,MAXVAL           TEST OVERFLOW ON RESULTING ADD
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

0009BA 4740 09C2           009C2  1451          BC    4,INDX5              NO, BRANCH
0009BE 5BD0 09CC           009CC  1452          S     WR4,MAXVAL           YES, CLEAR OVERFLOW
0009C2 86D6 09F0           009F0  1453 INDX5    BXH   WR4,MAPORG,ERRORP    ADD BASE AND CHECK
0009C6 47F0 093C           0093C  1454 INDX8    B       INDAD              GO TO INDAD SUBROUTINE      V1L2
                                  1455 *         B       INDAD              GO TO INDAD SUBROU(DELETED) V1L2
0009CA 07FE                       1456          BR       14                RETURN TO CALLER            V1L2
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1458 .NOTIX1  ANOP
                                  1459          AIF   (NOT &INDAD).NOTIND1    INDIR ADDRESSING
                                  1460 .NOTIND1 ANOP
                                  1461          AIF   (&MODEL NE 1).NMOD1         MODEL 1
                                  1462 .NMOD1   ANOP

                           007F6  1464 ENTRY0   EQU   ENTRY1
0009CC 000186A0                   1465 MAXVAL   DC    F'100000'                  MAXIMUM VALUE FOR 9'S COMPLEMENT
0009D0 0004                       1466 DEC4     DC    H'4'                       DECIMAL 4



                                  1469 ************************************************************************
                                  1470 *                                                                      *
                                  1471 *            PARAMETERS USED IN ADDRESS CONVERSION ROUTINES            *
                                  1472 *                                                                      *
                                  1473 ************************************************************************

0009D8                            1475 BUFFP    DS    1D                         FOR P-ADDRESS PACKING
0009E0                            1476 BUFFQ    DS    1D                         FOR Q-ADDRESS PACKING
0009E8                            1477 INITP    DS    1F                         FOR STORING INITIAL P-ADDRESS
0009EC                            1478 INITQ    DS    1F                         FOR STORING INITIAL Q-ADDRESS
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1480 ************************************************************************
                                  1481 *                                                                      *
                                  1482 *                          ERROR ROUTINES                              *
                                  1483 *                                                                      *
                                  1484 *  THESE ROUTINES ARE ENTERED FROM THE SPECIAL EXITS OF THE  ADDRESS   *
                                  1485 *  CONVERSION ROUTINES IN THE FOLLOWING CASES =                        *
                                  1486 *                                                                      *
                                  1487 *  1.  THE  CONVERSION  ADDRESS IS GREATER  THAN THE  AVAILABLE CORE   *
                                  1488 *      STORAGE SIZE SPECIFIED FOR THE SIMULATOR.                       *
                                  1489 *                                                                      *
                                  1490 *  2.  ALPHABETIC OR SPECIAL  CHARACTERS ARE  ENCOUNTERED IN 'BUFFP'   *
                                  1491 *      OR 'BUFFQ' AT THE TIME OF BINARY CONVERSION.  AN INTERRUPTION   *
                                  1492 *      IS THEN REQUESTED BY THE CONTROL PROGRAM WHICH GIVES  CONTROL   *
                                  1493 *      TO THE ERROR ROUTINE. THIS SENDS A MESSAGE TO THE 1052 PRINT-   *
                                  1494 *      ER-KEYBOARD  ('INVP' OR 'INVQ')  AND SIMULATION  STOPS  UNTIL   *
                                  1495 *      OPERATOR ACTION (SEE 'BIR' AND SIMULATION OF KEYS).             *
                                  1496 *                                                                      *
                                  1497 ************************************************************************
                                  1498 *        SPACE 2                                    (DELETED) V1L2
                                  1499 *        CNOP  2,4                  *               (DELETED) V1L2

0009F0 45F0 00F4          000F4   1501 ERRORP   BAL   15,MESTOP             *
0009F4 0000019D                   1502          DC  A(INVP)               ERROR ON P ADDRESS              A
0009F8 45F0 00F4          000F4   1503 ERRORQ   BAL   15,MESTOP             *
0009FC 0000019D                   1504          DC  A(INVQ)               ERROR ON Q ADDRESS              A
000A00                            1505 CODERR   DS    0H
000A00 45F0 00F4          000F4   1506          BAL   15,MESTOP             *
000A04 0000019D                   1507          DC  A(INVCOD)             ERROR ON OPERATION CODE
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
                                1509 ************************************************************************
                                1510 *                                                                      *
                                1511 *                        OPERATION CODE TABLE                          *
                                1512 *                                                                      *
                                1513 *  THIS IS A 154 HALFWORD  TABLE WHICH CONTAINS  THE  RELATIVE   AD-  *
                                1514 *  SES IN DISPLACEMENT FORMAT OF THE SIMULATION ROUTINES. EACH HALF-  *
                                1515 *  WORD REFERS TO  THE STARTING ADDRESS OF THE ROUTINE  WHICH  SIMU-  *
                                1516 *  LATES THE EXECUTION OF THE 1620 INSTRUCTION INDICATED BY THE  OP-  *
                                1517 *  ERATION CODE.                                                       *
                                1518 *  SOME 1620 OPERATION CODES ARE NOT RELATED TO  ANY  EXISTING   IN-  *
                                1519 *  STRUCTION. THEY CONTAIN THE ADDRESS OF THE ERROR ROUTINE.           *
                                1520 *                                                                      *
                                1521 ************************************************************************

000A08 00000A00                 1523 OPTBL    DC  A(CODERR)              NOT USED
000A0C 00001AA8                 1524          DC  A(FADD01)
000A10 00001B80                 1525          DC  A(FSUB02)
000A14 0000176E                 1526          DC  A(FMUL03)
000A18 00000A00                 1527          DC  A(CODERR)              NOT USED
000A1C 0000199C                 1528          DC  A(FSL05)
000A20 00001932                 1529          DC  A(TFL06)
000A24 00001948                 1530          DC  A(BTFL07)
000A28 0000196A                 1531          DC  A(FSR08)
000A2C 00001804                 1532          DC  A(FDIV09)
000A30 00000A00                 1533          DC  A(CODERR)              NOT USED
000A34 00000A00                 1534          DC  A(CODERR)              NOT USED
000A38 00000A00                 1535          DC  A(CODERR)              NOT USED
000A3C 00000A00                 1536          DC  A(CODERR)              NOT USED
000A40 00000A00                 1537          DC  A(CODERR)              NOT USED
000A44 00000A00                 1538          DC  A(CODERR)              NOT USED
000A48 00001C78                 1539          DC  A(BTAM10)
000A4C 000011CA                 1540          DC  A(AM11)
000A50 000011EC                 1541          DC  A(SM12)
000A54 00001388                 1542          DC  A(MM13)
000A58 00000E80                 1543          DC  A(CM14)
000A5C 00000C70                 1544          DC  A(TDM15)
000A60 00000C8A                 1545          DC  A(TFM16)
000A64 000010FC                 1546          DC  A(BTM17)
000A68 00001762                 1547          DC  A(LDM18)
000A6C 0000171A                 1548          DC  A(DM19)
000A70 00000A00                 1549          DC  A(CODERR)              NOT USED
000A74 00000A00                 1550          DC  A(CODERR)              NOT USED
000A78 00000A00                 1551          DC  A(CODERR)              NOT USED
000A7C 00000A00                 1552          DC  A(CODERR)              NOT USED
000A80 00000A00                 1553          DC  A(CODERR)              NOT USED
000A84 00000A00                 1554          DC  A(CODERR)              NOT USED
000A88 00001C10                 1555          DC  A(BTA20)
000A8C 00001186                 1556          DC  A(A21)
000A90 000011D6                 1557          DC  A(S22)
000A94 000013A6                 1558          DC  A(M23)
000A98 00000DC2                 1559          DC  A(C24)
000A9C 00000C7C                 1560          DC  A(TD25)
000AA0 00000C96                 1561          DC  A(TF26)
000AA4 000010CC                 1562          DC  A(BT27)
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
000AA8 00001726                    1563          DC  A(LD28)
000AAC 00001700                    1564          DC  A(D29)
000AB0 00000A00                    1565          DC  A(CODERR)              NOT USED
000AB4 00000A00                    1566          DC  A(CODERR)              NOT USED
000AB8 00000A00                    1567          DC  A(CODERR)              NOT USED
000ABC 00000A00                    1568          DC  A(CODERR)              NOT USED
000AC0 00000A00                    1569          DC  A(CODERR)              NOT USED
000AC4 00000A00                    1570          DC  A(CODERR)              NOT USED
000AC8 00000A00                    1571          DC  A(TRNM30)
000ACC 00000CD0                    1572          DC  A(TR31)
000AD0 00001138                    1573          DC  A(SF32)
000AD4 00001144                    1574          DC  A(CF33)
000AD8 00000670                    1575          DC  A(K34)
000ADC 000006DA                    1576          DC  A(DN35)
000AE0 00000682                    1577          DC  A(RN36)
000AE4 000006A6                    1578          DC  A(RA37)
000AE8 00000694                    1579          DC  A(WN38)
000AEC 000006C6                    1580          DC  A(WA39)
000AF0 00000A00                    1581          DC  A(CODERR)              NOT USED
000AF4 00000A00                    1582          DC  A(CODERR)              NOT USED
000AF8 00000A00                    1583          DC  A(CODERR)              NOT USED
000AFC 00000A00                    1584          DC  A(CODERR)              NOT USED
000B00 00000A00                    1585          DC  A(CODERR)              NOT USED
000B04 00000A00                    1586          DC  A(CODERR)              NOT USED
000B08 00000A00                    1587          DC  A(CODERR)              NOT USED
000B0C 0000114C                    1588          DC  A(NOP41)
000B10 00001108                    1589          DC  A(BB42)
000B14 00000F32                    1590          DC  A(BD43)
000B18 00000F5A                    1591          DC  A(BNF44)
000B1C 00000F6A                    1592          DC  A(BNR45)
000B20 00000F7A                    1593          DC  A(BI46)
000B24 00000FD0                    1594          DC  A(BNI47)
000B28 00001150                    1595          DC  A(H48)
000B2C 00000F3E                    1596          DC  A(B49)
000B30 00000A00                    1597          DC  A(CODERR)              NOT USED
000B34 00000A00                    1598          DC  A(CODERR)              NOT USED
000B38 00000A00                    1599          DC  A(CODERR)              NOT USED
000B3C 00000A00                    1600          DC  A(CODERR)              NOT USED
000B40 00000A00                    1601          DC  A(CODERR)              NOT USED
000B44 00000A00                    1602          DC  A(CODERR)              NOT USED
000B48 00000A00                    1603          DC  A(CODERR)              NOT USED
000B4C 00000A00                    1604          DC  A(CODERR)              NOT USED
000B50 00000A00                    1605          DC  A(CODERR)              NOT USED
000B54 00000A00                    1606          DC  A(CODERR)              NOT USED
000B58 00000A00                    1607          DC  A(CODERR)              NOT USED
000B5C 00000A00                    1608          DC  A(BNG55)
000B60 00000A00                    1609          DC  A(CODERR)              NOT USED
000B64 00000A00                    1610          DC  A(CODERR)              NOT USED
000B68 00000A00                    1611          DC  A(CODERR)              NOT USED
000B6C 00000A00                    1612          DC  A(CODERR)              NOT USED
000B70 00000A00                    1613          DC  A(CODERR)              NOT USED
000B74 00000A00                    1614          DC  A(CODERR)              NOT USED
000B78 00000A00                    1615          DC  A(CODERR)              NOT USED
000B7C 00000A00                    1616          DC  A(CODERR)              NOT USED
000B80 00000A00                    1617          DC  A(CODERR)              NOT USED
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
000B84 00000A00                   1618          DC  A(CODERR)              NOT USED
000B88 00001C9A                   1619          DC  A(BS60)
000B8C 00001CFC                   1620          DC  A(BX61)
000B90 00001DAE                   1621          DC  A(BXM62)
000B94 00001DB6                   1622          DC  A(BCX63)
000B98 00001DBE                   1623          DC  A(BCXM64)
000B9C 00001DE4                   1624          DC  A(BLX65)
000BA0 00001E12                   1625          DC  A(BLXM66)
000BA4 00001E1E                   1626          DC  A(BSX67)
000BA8 00000A00                   1627          DC  A(CODERR)              NOT USED
000BAC 00000A00                   1628          DC  A(CODERR)              NOT USED
000BB0 00000A00                   1629          DC  A(CODERR)              NOT USED
000BB4 00000A00                   1630          DC  A(CODERR)              NOT USED
000BB8 00000A00                   1631          DC  A(CODERR)              NOT USED
000BBC 00000A00                   1632          DC  A(CODERR)              NOT USED
000BC0 00000A00                   1633          DC  A(CODERR)              NOT USED
000BC4 00000A00                   1634          DC  A(CODERR)              NOT USED
000BC8 00001C84                   1635          DC  A(MA70)
000BCC 0000116C                   1636          DC  A(MF71)
000BD0 00000D06                   1637          DC  A(TNS72)
000BD4 00000D72                   1638          DC  A(TNF73)
000BD8 00000A00                   1639          DC  A(CODERR)              NOT USED
000BDC 00000A00                   1640          DC  A(CODERR)              NOT USED
000BE0 00000A00                   1641          DC  A(CODERR)              NOT USED
000BE4 00000A00                   1642          DC  A(CODERR)              NOT USED
000BE8 00000A00                   1643          DC  A(CODERR)              NOT USED
000BEC 00000A00                   1644          DC  A(CODERR)              NOT USED
000BF0 00000A00                   1645          DC  A(CODERR)              NOT USED
000BF4 00000A00                   1646          DC  A(CODERR)              NOT USED
000BF8 00000A00                   1647          DC  A(CODERR)              NOT USED
000BFC 00000A00                   1648          DC  A(CODERR)              NOT USED
000C00 00000A00                   1649          DC  A(CODERR)              NOT USED
000C04 00000A00                   1650          DC  A(CODERR)              NOT USED
000C08 00000A00                   1651          DC  A(CODERR)              NOT USED
000C0C 00000A00                   1652          DC  A(CODERR)              NOT USED
000C10 00000A00                   1653          DC  A(CODERR)              NOT USED
000C14 00000A00                   1654          DC  A(CODERR)              NOT USED
000C18 00000A00                   1655          DC  A(CODERR)              NOT USED
000C1C 00000A00                   1656          DC  A(CODERR)              NOT USED
000C20 00000A00                   1657          DC  A(CODERR)              NOT USED
000C24 00000A00                   1658          DC  A(CODERR)              NOT USED
000C28 00000A00                   1659          DC  A(CODERR)              NOT USED
000C2C 00000A00                   1660          DC  A(CODERR)              NOT USED
000C30 00000A00                   1661          DC  A(CODERR)              NOT USED
000C34 00000A00                   1662          DC  A(CODERR)              NOT USED
000C38 00000A00                   1663          DC  A(CODERR)              NOT USED
000C3C 00000A00                   1664          DC  A(CODERR)              NOT USED
000C40 00000A00                   1665          DC  A(CODERR)              NOT USED
000C44 00000A00                   1666          DC  A(CODERR)              NOT USED
000C48 00000A00                   1667          DC  A(CODERR)              NOT USED
000C4C 00000A00                   1668          DC  A(CODERR)              NOT USED
000C50 00000A00                   1669          DC  A(CODERR)              NOT USED
000C54 00000A00                   1670          DC  A(CODERR)              NOT USED
000C58 00000A00                   1671          DC  A(CODERR)              NOT USED
000C5C 00000A00                   1672          DC  A(CODERR)              NOT USED
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
000C60 00000A00                   1673          DC  A(CODERR)            NOT USED
000C64 00000A00                   1674          DC  A(CODERR)            NOT USED
000C68 00000A00                   1675          DC  A(CODERR)            NOT USED
000C6C 00000A00                   1676          DC  A(CODERR)            NOT USED     LAST CODE=99
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
1678 ********************************************************************************
1679 *                                                                              *
1680 *                              CPU SIMULATION                                  *
1681 *                                                                              *
1682 *  THE  SIMULATED  CPU INSTRUCTIONS  ARE THOSE  OF THE 1620  MODEL 2           *
1683 *  EQUIPPED WITH  THE SPECIAL FEATURE 'INDEX REGISTERS'. THE FOLLOW-           *
1684 *  ING TABLE REPRESENTS, IN ALPHABETIC ORDER,  ALL  CPU   OPERATIONS           *
1685 *  WHICH CAN BE SIMULATED.                                                     *
1686 *                                                                              *
1687 *                                                                              *
1688 *  A              BTFL           CF             H              TD             *
1689 *  AM             BB             C              LD             TDM            *
1690 *  B              BCX            CM             LDM            TF             *
1691 *  BLX            BCXM           D              MA             TFM            *
1692 *  BLXM           BI             DM             MF             TFL            *
1693 *  BX             BNF            FADD           M                             *
1694 *  BXM            BNG            FDIV           MM             TR             *
1695 *  BS             BNI            FMUL           NOP            TRNM           *
1696 *  BSX            BNR            FSL            SF             TNF            *
1697 *  BT             BD             FSR            S              TNS            *
1698 *  BTM                           FSUB           SM                            *
1699 *                                                                              *
1700 *                                                                              *
1701 *  IF THE 1620 MODEL 1 IS SIMULATED, OR IF MODEL 2 IS SIMULATED WITH-          *
1702 *  OUT CERTAIN SPECIAL FEATURES, SOME OF THESE INSTRUCTIONS WILL  BE           *
1703 *  ELIMINATED DURING THE EDITING PROCESS.                                      *
1704 *                                                                              *
1705 *  CPU SIMULATION CONSISTS OF THE FOLLOWING SECTIONS =                         *
1706 *                                                                              *
1707 *  - INTERNAL DATA TRANSMISSION OPERATIONS                                     *
1708 *  - LOGIC OPERATIONS                                                          *
1709 *  - PROGRAM CONTROL OPERATIONS                                                *
1710 *  - ARITHMETIC OPERATIONS                                                     *
1711 *  - FLOATING-POINT OPERATIONS                                                 *
1712 *  - INDEX REGISTER OPERATIONS                                                 *
1713 *                                                                              *
1714 *                                                                              *
1715 *  THE CPU OPERATIONS MENTIONED IN THE ABOVE TABLE HAVE A NUMBER OF            *
1716 *  COMMON CHARACTERISTICS WHICH ARE=                                           *
1717 *                                                                              *
1718 *  1) ENTRY POINTS                                                             *
1719 *  THE ENTRY POINT FOR EACH OF THE SIMULATION ROUTINES   CORRESPONDS           *
1720 *  TO THE 1620 MNEMONICS OF THE INSTRUCTION PLUS ITS CODE NUMBER.              *
1721 *                                                                              *
1722 *              EXAMPLE =  TDM15 FOR 'TRANSMIT DIGIT IMMEDIATE'                 *
1723 *                                                                              *
1724 *  2) INPUT                                                                    *
1725 *  WHEN EACH OF THE ROUTINES IS ENTERED, THE GENERAL REGISTER 'CNTR'           *
1726 *  CONTAINS THE ADDRESS OF THE 1620 INSTRUCTION (IN MEMORY MAPPING).           *
1727 *  THE DOUBLE WORD 'BUFFP' CONTAINS THE OPERATION CODE AND THE P AD-           *
1728 *  DRESS IN PACKED DECIMAL FORMAT.                                             *
1729 *                                                                              *
1730 *                                                                  ./...       * ..
1731 *                                                                              *
1732 ********************************************************************************
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
1734 ************************************************************************
1735 *                                                                      *
1736 *    ./...                                                             * ..
1737 *                                                                      *
1738 *  THESE ARGUMENTS ARE USED FOR  ADDRESS CONVERSION,  IF SUCH IS RE-   *
1739 *  QUIRED,  AND THEN FOR THE EXECUTION OF THE  SIMULATED  1620   IN-   *
1740 *  STRUCTION.                                                          *
1741 *                                                                      *
1742 *  3) OUTPUT                                                           *
1743 *  THE OUTPUT FROM THE ROUTINES DEPENDS ON THE FUNCTION OF THE  SIM-   *
1744 *  ULATED INSTRUCTION.                                                 *
1745 *                                                                      *
1746 *  4) EXIT                                                             *
1747 *  AT THE END OF EACH ROUTINE, CONTROL IS GIVEN TO ONE OF THE TWO      *
1748 *  'BIR' ENTRIES.                                                      *
1749 *                                                                      *
1750 *  5) GENERAL SUBROUTINES                                              *
1751 *                                                                      *
1752 *  A) THE 'COMP' SUBROUTINE IS USED FOR 'COMPARE' AND 'COMPARE IMME-   *
1753 *     DIATE 'OPERATIONS.  IT IS ALSO USED FOR 'FLOATING POINT' OPER-   *
1754 *    ATIONS IF THIS OPTION IS INCLUDED.                                *
1755 *                                                                      *
1756 *  B) THE 'FIXADD' SUBROUTINE  IS USED FOR 'ADD' AND 'ADD IMMEDIATE'   *
1757 *    OPERATIONS.  IT IS ALSO USED FOR 'FLOATING ADD' AND 'INDEX  RE-   *
1758 *    GISTER' OPERATIONS IF THESE OPTIONS ARE INCLUDED.                 *
1759 *                                                                      *
1760 *  C) THE 'MULT' SUBROUTINE IS USED FOR 'MULTIPLY' AND 'MULTIPLY IM-   *
1761 *     MEDIATE' OPERATIONS.  IT IS ALSO USED FOR  THE 'FLOATING  MUL-   *
1762 *    TIPLY' OPERATION IF THIS OPTION IS INCLUDED.                      *
1763 *                                                                      *
1764 *  D) THE 'INDIC' SUBROUTINE IS USED FOR THOSE  OPERATIONS WHOSE RE-   *
1765 *     SULT MODIFIES THE HIGH/POSITIVE AND EQUAL/ZERO INDICATORS.       *
1766 *                                                                      *
1767 *  E) THE 'ARCHK' SUBROUTINE  IS USED FOR  OPERATIONS WHICH CAUSE AN   *
1768 *     ARITHMETIC CHECK (FOR INSTANCE OVERFLOW).                        *
1769 *                                                                      *
1770 *  F) THE  'EXCHK' SUBROUTINE IS USED FOR FLOATING-POINT  ARITHMETIC   *
1771 *    OPERATIONS WHEN EXPONENT OVERFLOW OR UNDERFLOW OCCURS.            *
1772 *                                                                      *
1773 *  G) THE 'FIXDIV' SUBROUTINE IS USED FOR 'DIVIDE' AND 'DIVIDE IMME-   *
1774 *     DIATE' OPERATIONS AND FOR 'FLOATING DIVIDE' OPERATIONS IF ONE    *
1775 *     OR BOTH OF THESE OPTIONS ARE INCLUDED.                           *
1776 *                                                                      *
1777 *  H) THE 'SHIFT' SUBROUTINE AND THE 'EXPOW' SUBROUTINE ARE USED FOR   *
1778 *     FLOATING-POINT ARITHMETIC OPERATIONS.                            *
1779 *                                                                      *
1780 *                                                                      *
1781 *                                                          ./...       * ..
1782 *                                                                      *
1783 ************************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1785 ********************************************************************************
                                  1786 *                                                                              *
                                  1787 *    ./...                                                                     * ..
                                  1788 *                                                                              *
                                  1789 *  NOTE=                                                                        *
                                  1790 *  WRAP-AROUND IS SIMULATED IN THE FOLLOWING OPERATIONS=                       *
                                  1791 *                                                                              *
                                  1792 *  -TRANSMIT FIELD  IMMEDIATE TFM-16                                           *
                                  1793 *  -TRANSMIT FIELD  TF-26                                                      *
                                  1794 *  -TRANSMIT RECORD  TR-31                                                     *
                                  1795 *  -TRANSMIT RECORD NO RECORD MARK  TRNM-30                                    *
                                  1796 *  -TRANSFER NUMERICAL STRIP TNS-72                                            *
                                  1797 *  -TRANSFER NUMERICAL FILL  TNF-73                                            *
                                  1798 *  -TRANSFER FLOATING  TFL-06                                                  *
                                  1799 *  -BRANCH AND TRANSMIT FLOATING  BTFL-07                                      *
                                  1800 *                                                                              *
                                  1801 *  IT IS NOT SIMULATED IN FIXED AND FLOATING-POINT  ARITHMETIC   AND           *
                                  1802 *  COMPARE OPERATIONS.                                                         *
                                  1803 *                                                                              *
                                  1804 *                                                                              *
                                  1805 *  THOSE OF THE FOLLOWING ROUTINES THAT  ARE  SELF-EXPLANATORY  WILL           *
                                  1806 *  NOT BE PRECEDED BY ANY GENERAL COMMENTS.                                    *
                                  1807 *                                                                              *
                                  1808 ********************************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1810 ***************************************************************************
                                  1811 *                                                                         *
                                  1812 *                                SECTION 1                                *
                                  1813 *                  INTERNAL DATA TRANSMISSION OPERATIONS                  *
                                  1814 *                                                                         *
                                  1815 ***************************************************************************



                                  1818 ***************************************************************************
                                  1819 *                                                                         *
                                  1820 *                             TRANSMIT DIGIT                              *
                                  1821 *                                                                         *
                                  1822 *  1. TRANSMIT DIGIT IMMEDIATE                                            *
                                  1823 *  2. TRANSMIT DIGIT                                                      *
                                  1824 *                                                                         *
                                  1825 ***************************************************************************

000C70 4195 000B           0000B  1827 TDM15    LA    RQ,11(CNTR)           Q = RIGHTMOST ADDRESS OF INSTR.
000C74 45F0 0906           00906  1828          BAL   15,CONVP              CONVERT P ADDRESS
000C78 47F0 0C80           00C80  1829          B       TD25+4              GO TO TRANSMIT DIGIT

000C7C 45F0 0902           00902  1831 TD25     BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
000C80 D200 8000 9000 00000 00000  1832          MVC   0(1,RP),0(RQ)         MOVE DIGIT + FLAG
000C86 47F0 07F6           007F6  1833          B       ENTRY1              RETURN TO 'BIR'



                                  1837 ***************************************************************************
                                  1838 *                                                                         *
                                  1839 *                             TRANSMIT FIELD                              *
                                  1840 *                                                                         *
                                  1841 *  1. TRANSMIT FIELD IMMEDIATE                                            *
                                  1842 *                                                                         *
                                  1843 *  2. TRANSMIT FIELD                                                      *
                                  1844 *     AFTER CONVERSION  OF THE  P AND Q ADDRESSES, THE FIRST BYTE IS      *
                                  1845 *     MOVED FROM THE P ADDRESS TO THE Q ADDRESS. A LOOP IS THEN  EN-      *
                                  1846 *     TERED TO TRANSMIT ONE BYTE AFTER THE OTHER FROM  RIGHT TO LEFT      *
                                  1847 *     (INCLUDING THE FLAG, IF ANY).  THE WRAP-AROUND FEATURE IS  IN-      *
                                  1848 *     CLUDED.                                                             *
                                  1849 *                                                                         *
                                  1850 ***************************************************************************

000C8A 4195 000B           0000B  1852 TFM16    LA    RQ,11(CNTR)           Q = RIGHTMOST ADDRESS OF INSTR.
000C8E 45F0 0906           00906  1853          BAL   15,CONVP              CONVERT P ADDRESS
000C92 47F0 0C9A           00C9A  1854          B       TF26+4              GO TO TRANSMIT FIELD

000C96 45F0 0902           00902  1856 TF26     BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
000C9A D200 8000 9000 00000 00000  1857           MVC   0(1,RP),0(RQ)          MOVE FIRST CHARACTER
000CA0 45F0 0CB8            0CB8   1858 TFA       BAL   15,TNSWR+2             TEST FOR WRAP-AROUND
000CA4 D200 8000 9000 00000 00000  1859           MVC   0(1,RP),0(RQ)          MOVE CURRENT CHARACTER
000CAA 9120 9000      00000        1860           TM    0(RQ),X'20'            TEST IF FLAGGED
000CAE 4710 0CA0            0CA0   1861           BO      TFA                  NO, LOOP
000CB2 47F0 07F6            07F6   1862           B     ENTRY1                 YES, EXIT


000CB6 0680                        1864 TNSWR     BCTR  RP,0                   RP=RP-1
000CB8 0680                        1865           BCTR  RP,0                   RP=RP-1
000CBA 1986                        1866           CR    RP,MAPORG              IS RP LOWER THAN MAPORG
000CBC 4740 0CCA            0CCA   1867           BL      TNSWA                YES,BRANCH
000CC0 0690                        1868 TNSWB     BCTR  RQ,0                   NO,RQ=RQ-1
000CC2 1996                        1869           CR    RQ,MAPORG              IS RQ LOWER THAN MAPORG
000CC4 07AF                        1870           BCR   10,15                  NO,BRANCH
000CC6 1897                        1871           LR    RQ,SIZE                YES,RQ=SIZE
000CC8 07FF                        1872           BR      15                   RETURN TO CALLER
000CCA 1887                        1873 TNSWA     LR    RP,SIZE                RP=SIZE
000CCC 47F0 0CC0            0CC0   1874           B       TNSWB                RETURN TO CALLER



                                   1877 ***************************************************************************
                                   1878 *                                                                         *
                                   1879 *                            TRANSMIT RECORD                              *
                                   1880 *                                                                         *
                                   1881 *  AFTER CONVERSION OF THE  P AND Q ADDRESSES,  THE FIRST  BYTE   IS      *
                                   1882 *  MOVED FROM THE  P ADDRESS TO THE  Q ADDRESS.  A LOOP IS THEN  EN-      *
                                   1883 *  TERED  TO TRANSMIT ONE BYTE AFTER  THE OTHER FROM  RIGHT  TO LEFT      *
                                   1884 *  (INCLUDING THE FLAG, IF ANY).                                          *
                                   1885 *                                                                         *
                                   1886 *  THE WRAP-AROUND FEATURE IS INCLUDED.                                   *
                                   1887 *                                                                         *
                                   1888 ***************************************************************************


000CD0 45F0 0902            0902   1890 TR31      BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
000CD4 D200 8000 9000 00000 00000  1891           MVC   0(1,RP),0(RQ)          MOVE ONE CHARACTER
000CDA 910A 9000      00000        1892           TM    0(RQ),X'0A'            TEST IF RECORD MARK
000CDE 4710 07F6            07F6   1893           BO      ENTRY1               YES, EXIT
000CE2 45F0 0CEA            0CEA   1894           BAL   15,TRWR                TEST FOR WRAP-AROUND
000CE6 47F0 0CD4            0CD4   1895           B       TR31+4               LOOP


000CEA 4188 0001            0001   1897 TRWR      LA    RP,1(RP)               RP=RP+1
000CEE 1987                        1898           CR    RP,SIZE                IS RP GREATER THAN SIZE
000CF0 4720 0D00            0D00   1899           BH      TR31B                YES,BRANCH
000CF4 4199 0001            0001   1900 TR31A     LA    RQ,1(RQ)               NO,RQ=RQ+1
000CF8 1997                        1901           CR    RQ,SIZE                IS RQ GREATER THAN SIZE
000CFA 0000                        1902           BCR   BNH,15                 NO,RETURN TO CALLER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
000CFC 1896                        1903           LR    RQ,MAPORG              YES,RQ=MAPORG
000CFE 07FF                        1904           BR      15                   RETURN TO CALLER
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
000D00 1886                       1905 TR31B    LR    RP,MAPORG             RP=MAPORG
000D02 47F0 0CF4           00CF4  1906          B       TR31A               *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1908          AIF   (NOT &DISK).NODSK7
                                  1909 .NODSK7  ANOP
                            00A00 1910 TRNM30   EQU   CODERR                  CODE OPERATION ERROR
                                  1911 .DSKX1A  ANOP
                                  1912          AIF   (NOT &TRANS).NOTRN1    ADDIT INSTRUCTIONS


                                  1914 ************************************************************************
                                  1915 *                                                                      *
                                  1916 *                        TRANSFER NUMERICAL STRIP                      *
                                  1917 *                                                                      *
                                  1918 *  THE P AND Q ADDRESSES ARE CONVERTED.  THE P ADDRESS IS TESTED. IF   *
                                  1919 *  THIS ADDRESS IS ODD,   THE FIRST NUMERIC DIGIT IS MOVED  FROM THE   *
                                  1920 *  P FIELD TO THE Q FIELD,  AND THE SECOND  NUMERIC  DIGIT  IS TEST-    *
                                  1921 *  ED.  IF IT IS DIFFERENT FROM 5,  2 OR 1, A 'D' IS MOVED INTO  THE   *
                                  1922 *  ZONE PART OF THE Q DIGIT, OTHERWISE AN 'F' IS MOVED INTO THE ZONE   *
                                  1923 *  PART OF THE Q DIGIT. THEN, A LOOP IS ENTERED TO MOVE EACH NUMERIC   *
                                  1924 *  DIGIT FROM THE P ADDRESS TO THE Q ADDRESS. A TEST FOR A FLAG TER-    *
                                  1925 *  MINATES THIS SEQUENCE.                                              *
                                  1926 *                                                                      *
                                  1927 ************************************************************************

000D06 45F0 0902                  1929 TNS72    BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
000D0A 4280 0F58            00F58 1930          STC   RP,STC                  TEST IF P ADDRESS IS ODD
000D0E 9101 0F58      00F58       1931          TM    STC,X'01'               *
000D12 4780 09F0            009F0 1932          BZ      ERRORP              NO,P IS INVALID
000D16 18A8                       1933          LR    WR1,RP                  YES,PUT P-1 IN WR1
000D18 06A0                       1934          BCTR  WR1,0                   *
000D1A D200 9000 8000 00000 00000 1935          MVC   0(1,RQ),0(RP)           MOVE LAST NUMERIC FROM P TO Q
000D20 9400 0F58      00F58       1936          NI    STC,X'00'               CLEAR STC
000D24 D100 0F58 A000 00F58 00000 1937          MVN   STC(1),0(WR1)           MOVE LAST ZONE FROM P
000D2A 9505 0F58      00F58       1938          CLI   STC,X'05'               COMPARE ZONE WITH 5
000D2E 4780 0D6A            00D6A 1939          BC    8,TNSB                  EQUAL,BRANCH
000D32 4720 0D3E            00D3E 1940          BC    2,TNSA                  HIGHER,BRANCH
000D36 9103 0F58      00F58       1941          TM    STC,X'03'               LOWER,TEST IF EQUAL TO 1 OR 2
000D3A 4740 0D5A            00D5A 1942          BC    4,TNSC                  YES,BRANCH
000D3E 45F0 0CB6            00CB6 1943 TNSA     BAL   15,TNSWR                TEST FOR WRAP-AROUND
000D42 D100 9000 8000 00000 00000 1944          MVN   0(1,RQ),0(RP)           MOVE NUMERIC FROM P TO Q
000D48 9120 9000      00000       1945          TM    0(RQ),X'20'             TEST IF Q FLAGGED
000D4C 4780 07F6            007F6 1946          BZ      ENTRY1                YES, EXIT TO 'BIR'
000D50 D300 9000 8000 00000 00000 1947          MVZ   0(1,RQ),0(RP)           NO, MOVE ZONE FROM P TO Q
000D56 47F0 0D3E            00D3E 1948          B       TNSA                  BRANCH TO TNSA
000D5A 9101 0F58      00F58       1949 TNSC     TM    STC,X'01'               TEST IF EQUAL 1
000D5E 4710 0D6A            00D6A 1950          BC    1,TNSB                  YES,BRANCH
000D62 910F 9000      00000       1951          TM    0(RQ),X'0F'             NO,TEST IF NUMERIC EQUAL 0
000D66 4770 0D3E            00D3E 1952          BC    7,TNSA                  NO,BRANCH
000D6A 94DF 9000      00000       1953 TNSB     NI    0(RQ),X'DF'             MOVE 'D' TO Q ZONE
000D6E 47F0 0D3E            00D3E 1954          B       TNSA                  BRANCH TO TNSA


                                  1957 ************************************************************************
                                  1958 *                                                                      *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  1959 *                    TRANSFER NUMERICAL FILL                    *
                                  1960 *                                                               *
                                  1961 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY.  THE P ADDRESS  IS  *
                                  1962 *  TESTED. IF IT IS ODD, THE Q DIGIT, WITH AN 'F' IN THE ZONE  PART,  *
                                  1963 *  IS MOVED TO THE P ADDRESS.  A TEST FOR A Q DIGIT FLAG IS MADE AND  *
                                  1964 *  A LOOP IS ENTERED TO EXTEND THE Q FIELD INTO THE  P  FIELD,  WITH  *
                                  1965 *  'F7' BYTES INTERLEAVED.                                          *
                                  1966 *                                                               *
                                  1967 ***************************************************************************

000D72 45F0 0902            00902 1969 TNF73    BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
000D76 4280 0F58            00F58 1970          STC   RP,STC                TEST IF P ADDRESS IS ODD
000D7A 9101 0F58      00F58       1971          TM    STC,X'01'             *
000D7E 4780 09F0            009F0 1972          BZ      ERRORP              NO,P IS INVALID
000D82 0680                       1973          BCTR  RP,0                  DECREMENT RP BY ONE
000D84 D100 8001 9000 00001 00000 1974          MVN   1(1,RP),0(RQ)         MOVE Q DIGIT TO P DIGIT
000D8A 96F0 8001      00001       1975          OI    1(RP),X'F0'           WITH'F'IN ZONE
000D8E 9120 9000      00000       1976          TM    0(RQ),X'20'           IS Q DIGIT FLAGGED
000D92 4780 0DBA            00DBA 1977          BZ      TNFB                YES, BRANCH
000D96 92F7 8000      00000       1978          MVI   0(RP),X'F7'           NO,PUT'F7'AT RP
000D9A 45F0 0CB8            00CB8 1979 TNFA     BAL   15,TNSWR+2            TEST FOR WRAP-AROUND
000D9E 0680                       1980          BCTR  RP,0                  *
000DA0 D100 8001 9000 00001 00000 1981          MVN   1(1,RP),0(RQ)         MOVE NUMERIC FROM Q TO P
000DA6 96F0 8001      00001       1982          OI    1(RP),X'F0'           WITH'F'IN ZONE
000DAA 92F7 8000      00000       1983          MVI   0(RP),X'F7'           PUT'F7'AT RP ADDRESS
000DAE 9120 9000      00000       1984          TM    0(RQ),X'20'           IS Q DIGIT FLAGGED
000DB2 4710 0D9A            00D9A 1985          BO      TNFA                NO, LOOP
000DB6 47F0 07F6            007F6 1986          B     ENTRY1                YES, EXIT TO 'BIR'
000DBA 92F5 8000      00000       1987 TNFB     MVI   0(RP),X'F5'           PUT'F5'IN RP BYTE
000DBE 47F0 0D9A            00D9A 1988          B     TNFA                  *
                                  1989          AGO   .TRNSX1
                                  1990 .TRNSX1  ANOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  1992 ********************************************************************************
                                  1993 *                                                                              *
                                  1994 *                                SECTION 2                                     *
                                  1995 *                             LOGIC  OPERATIONS                                *
                                  1996 *                                                                              *
                                  1997 ********************************************************************************



                                  1999 ********************************************************************************
                                  2000 *                                                                              *
                                  2001 *                                COMPARE                                       *
                                  2002 *                                                                              *
                                  2003 *  AFTER CONVERSION OF THE P AND Q ADDRESSES,THE P FIELD AND Q FIELD            *
                                  2004 *  SIGNS ARE COMPARED.  UNLIKE  SIGNS CAUSE  THE  SETTING OF THE E/Z           *
                                  2005 *  AND  H/P INDICATORS.  A SCAN THEN  DETERMINES  IF THE P FIELD AND           *
                                  2006 *  THE Q FIELD ARE ZERO,  AND IF  THE  P FIELD IS  LONGER THAN   THE          *
                                  2007 *  Q FIELD.                                                                    *
                                  2008 *                                                                              *
                                  2009 *  IF THE P FIELD AND Q FIELD SIGNS ARE ALIKE, A COMPARISON ON ABSO-           *
                                  2010 *  LUTE VALUES IS PERFORMED BY THE 'COMP' SUBROUTINE.                          *
                                  2011 *                                                                              *
                                  2012 *  THE COMPARISON IS COMPLETED  WHEN THE  NUMBER OF BYTES  IN  THE P           *
                                  2013 *  FIELD IS  GREATER THAN OR EQUAL TO THE NUMBER OF BYTES  IN  THE Q           *
                                  2014 *  FIELD*.  OTHERWISE, THE ARITHMETIC CHECK INDICATOR IS SET TO OVER-          *
                                  2015 *  FLOW.                                                                       *
                                  2016 *  THE MINIMUM LENGTH OF THE TWO COMPARED FIELDS IS TWO DIGITS.                *
                                  2017 *  *THE COMPARISON IS CORRECT UP TO THE END OF  THE SMALLER  ONE  OF           *
                                  2018 *  THE COMPARED FIELDS.                                                        *
                                  2019 *                                                                              *
                                  2020 ********************************************************************************

000DC2 45F0 0902          00902   2022 C24      BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
000DC6 5080 09E8          009E8   2023          ST    RP,INITP               SAVE P ADDRESS
000DCA 95F0 8000    00000         2024          CLI   0(RP),X'F0'            IS P FIELD POSITIVE
000DCE 47A0 0DE2          00DE2   2025          BC    10,C24A                YES,GO TO  C24A
000DD2 95F0 9000    00000         2026          CLI   0(RQ),X'F0'            NO,TEST Q SIGN POSITIVE
000DD6 4740 0E4C          00E4C   2027          BL      C24B                 NO,BRANCH
000DDA 9200 0003    00003         2028          MVI   HPEZ,X'00'             YES,SET H/P,E/Z = OFF
000DDE 47F0 0DEE          00DEE   2029          B       C24C                 *
000DE2 95F0 9000    00000         2030 C24A     CLI   0(RQ),X'F0'            IS Q FIELD POSITIVE
000DE6 47A0 0E4C          00E4C   2031          BC    10,C24B                YES,GO TO C24B
000DEA 9210 0003    00003         2032          MVI   HPEZ,X'10'             NO,SET H/P=ON,E/Z=OFF
000DEE 910F 8000    00000         2033 C24C     TM    0(RP),X'0F'            IS P FIELD EQUAL TO ZERO
000DF2 4750 0E5C          00E5C   2034          BC    BOM,C24ARC             GO TO TEST IF OVERFLOW OCCURV1L2
                                  2035 *        BC    BOM,ENTRY0             NO,RETURN TO 'BIR'(DELETED) V1L2
000DF6 910F 9000    00000         2036          TM    0(RQ),X'0F'            YES,IS Q FIELD EQUAL TO ZERO
000DFA 4750 0E5C          00E5C   2037          BC    BOM,C24ARC             GO TO TEST IF OVERFLOW      V1L2
                                  2038 *        BC    BOM,ENTRY0             NO RETURN TO 'BIR'(DELETED) V1L2
000DFE 0680                       2039 C24E     BCTR  RP,0                   YES,RP=RP-1
000E00 0690                       2040          BCTR  RQ,0                   RQ=RQ-1
000E02 910F 8000    00000         2041          TM    0(RP),X'0F'            IS P FIELD EQUAL TO ZERO
000E06 4750 0E60          00E60   2042          BC    BOM,C24AR1             GO TO TEST IF OVERFLOW      V1L2
                                  2043 *        BC    BOM,ENTRY0             NO,RETURN TO 'BIR'(DELETED) V1L2
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
000E0A 910F 9000      00000        2044           TM    0(RQ),X'0F'          YES,IS Q FIELD EQUAL TO ZERO
000E0E 4750 0E60            00E60  2045           BC    BOM,C24AR1           GO TO TEST IF OVERFLOW      V1L2
                                   2046 *         BC    BOM,ENTRY0           NO,RETURN TO 'BIR'(DELETED) V1L2
000E12 9120 9000      00000        2047           TM    0(RQ),X'20'          YES,IS Q FIELD FLAGGED
000E16 4780 0E2E            00E2E  2048           BZ      C24D               YES,BRANCH
000E1A 9120 8000      00000        2049           TM    0(RP),X'20'          NO,IS P FIELD FLAGGED
000E1E 4750 0DFE            00DFE  2050           BC    BOM,C24E             YES,LOOP
000E22 9201 0003      00003        2051           MVI   HPEZ,X'01'           NO, SET H/P=OFF,E/Z=ON
000E26 45F0 357C            0157C  2052 C24I      BAL   15,ARCHK             ARITHMETIC CHECK TESTING
000E2A 47F0 07F6            007F6  2053           B       ENTRY0             RETURN TO 'BIR'
000E2E 9120 8000      00000        2054 C24D      TM    0(RP),X'20'          IS P FIELD FLAGGED
000E32 4780 0E44            00E44  2055           BZ      C24F               YES,BRANCH
000E36 0680                        2056           BCTR  RP,0                 NO,RP=RP-1
000E38 910F 8000      00000        2057           TM    0(RP),X'0F'          IS P FIELD EQUAL TO ZERO
000E3C 4750 07F6            007F6  2058           BC    BOM,ENTRY0           NO,RETURN TO 'BIR'
000E40 47F0 0E2E            00E2E  2059           B       C24D               YES,LOOP
000E44 9201 0003      00003        2060 C24F      MVI   HPEZ,X'01'           SET H/P=OFF,E/Z=ON
000E48 47F0 07F6            007F6  2061           B       ENTRY0             RETURN TO 'BIR'
000E4C 9200 0E55      00E55        2062 C24B      MVI   NOQEND+1,X'00'       SET SWITCH NOQEND OFF
000E50 45F0 0E8C            00E8C  2063           BAL   15,COMP              COMPARE SUBROUTINE
000E54 4700 0E26            00E26  2064 NOQEND    BC    NOP,C24I             SWITCH NOQEND
000E58 47F0 07F6            007F6  2065           B       ENTRY0             RETURN TO 'BIR'
000E5C 0680                        2066 C24ARC    BCTR  RP,0                 RP=RP-1                     V1L2
000E5E 0690                        2067           BCTR  RQ,0                 RQ=RQ-1                     V1L2
000E60 9120 8000      00000        2068 C24AR1    TM    0(RP),X'20'          IS P FIELD FLAGGED          V1L2
000E64 4780 0E74            00E74  2069           BC    8,C24AR2             YES,BRANCH                  V1L2
000E68 9120 9000      00000        2070           TM    0(RQ),X'20'          NO,IS Q FIELD FLAGGED       V1L2
000E6C 4780 07F6            007F6  2071           BC    8,ENTRY0             YES RETURN TO 'BIR'         V1L2
000E70 47F0 0E5C            00E5C  2072           B       C24ARC             NO,LOOP                     V1L2
000E74 9120 9000      00000        2073 C24AR2    TM    0(RQ),X'20'          IS Q FIELD FLAGGED          V1L2
000E78 4780 07F6            007F6  2074           BC    8,ENTRY0             YES RETURN TO 'BIR'         V1L2
000E7C 47F0 0E26            00E26  2075           B       C24I               NO GO TO ARITH CHECK SUBR   V1L2


                                   2078 ************************************************************************
                                   2079 *                                                                      *
                                   2080 *                          COMPARE IMMEDIATE                           *
                                   2081 *                                                                      *
                                   2082 *  OPERATION IS AS FOR THE ABOVE 'COMPARE' INSTRUCTION EXCEPT   THAT   *
                                   2083 *  ONLY THE P ADDRESS IS CONVERTED. THE Q ADDRESS IS GIVEN  THE VAL-   *
                                   2084 *  UE  CORRESPONDING TO THE RIGHTMOST BOUNDARY OF THE 1620 INSTRUCT-   *
                                   2085 *  ION.                                                                *
                                   2086 *                                                                      *
                                   2087 ************************************************************************


000E80 45F0 0906            00906  2089 CM14      BAL   15,CONVP             CONVERT P ADDRESS
000E84 4195 000B            0000B  2090           LA    RQ,11(CNTR)          RQ = CNTR + 11
000E88 47F0 0DC6            00DC6  2091           B       C24+4              GO TO COMPARE
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2094 ************************************************************************
                                  2095 *                                                                      *
                                  2096 *                          'COMP' SUBROUTINE                           *
                                  2097 *                                                                      *
                                  2098 *  THIS SUBROUTINE IS USED BY THE FIXED-POINT  'COMPARE'  OPERATION,   *
                                  2099 *  AND THE FLOATING-DIVIDE OPERATION.                                  *
                                  2100 *                                                                      *
                                  2101 *  THE COMPARISON OF THE P AND Q FIELDS IS MADE,DIGIT BY DIGIT, FROM   *
                                  2102 *  RIGHT TO LEFT.  ALL THE INDICATORS ARE SUSEQUENTLY MODIFIED. NEI-   *
                                  2103 *  THER THE P FIELD NOR THE Q FIELD IS ALTERED.                        *
                                  2104 *                                                                      *
                                  2105 *  OUTPUT      UPDATING OF H/P AND E/Z INDICATORS                      *
                                  2106 *                                                                      *
                                  2107 ************************************************************************

000E8C 41A0 0001          00001   2109 COMP     LA    WR1,1                     WR1=1
000E90 D100 0E9D 9000 00E9D 00000 2110          MVN   COMPA+1(1),0(RQ)          COMPARE P NUMERIC
000E96 D100 0F30 8000 00F30 00000 2111          MVN   ACOMP(1),0(RP)            WITH Q NUMERIC
000E9C 9500 0F30      00F30       2112 COMPA    CLI   ACOMP,X'00'               *
000EA0 4740 0F10          00F10   2113          BL      COMPB                   LOWER
000EA4 4780 0F18          00F18   2114          BE      COMPE                   EQUAL
000EA8 9210 0003      00003       2115          MVI   HPEZ,X'10'                SET H/P=ON,E/Z=OFF
000EAC 0680                       2116 COMPF    BCTR  RP,0                      RP=RP-1
000EAE 0690                       2117          BCTR  RQ,0                      RQ=RQ-1
000EB0 41AA 0001          00001   2118          LA    WR1,1(WR1)                WR1=WR1+1
000EB4 D100 0EC1 9000 00EC1 00000 2119          MVN   COMPC+1(1),0(RQ)          COMPARE P NUMERIC
000EBA D100 0F30 8000 00F30 00000 2120          MVN   ACOMP(1),0(RP)            WITH Q NUMERIC
000EC0 9500 0F30      00F30       2121 COMPC    CLI   ACOMP,X'00'               *
000EC4 4720 0F20          00F20   2122          BH      COMPB1                  HIGHER
000EC8 4740 0F28          00F28   2123          BL      COMPE1                  LOWER
000ECC 9120 9000      00000       2124 COMPD    TM    0(RQ),X'20'               IS Q FIELD FLAGGED
000ED0 4780 0EE4          00EE4   2125          BZ      COMP1                   YES,BRANCH
000ED4 9120 8000      00000       2126          TM    0(RP),X'20'               IS P FIELD FLAGGED
000ED8 4710 0EAC          00EAC   2127          BO      COMPF                   NO,LOOP
000EDC 92F0 0E55      00E55       2128          MVI   NOQEND+1,X'F0'            SET SWITCH NOQEND ON
000EE0 47F0 0EFA          00EFA   2129          B       COMP2


000EE4 9120 8000      00000       2131 COMP1    TM    0(RP),X'20'               IS P FIELD FLAGGED
000EE8 4780 0EFA          00EFA   2132          BZ      COMP2                   YES,BRANCH
000EEC 0680                       2133          BCTR  RP,0                      NO,RP=RP-1
000EEE 910F 8000      00000       2134          TM    0(RP),X'0F'               IS P FIELD NUMERIC = 0
000EF2 4780 0EE4          00EE4   2135          BZ      COMP1                   YES,LOOP
000EF6 9210 0003      00003       2136          MVI   HPEZ,X'10'                SET H/P=ON,E/Z=OFF


000EFA 5880 09E8          009E8   2138 COMP2    L     RP,INITP                  RESTORE P ADDRESS
000EFE 95F0 8000      00000       2139          CLI   0(RP),X'F0'               IS P FIELD POSITIVE
000F02 07AF                       2140          BCR   10,15                     *

000F04 9501 0003      00003       2142          CLI   HPEZ,X'01'                NO,IS E/Z ON
000F08 078F                       2143          BZR     15                      YES,RETURN TO CALLER
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                ASM H V 05 22.10 05/07/81
000F0A 9710 0003      00003        2145           XI    HPEZ,X'10'        NO,EXCLUSIVE OR WITH H/P
000F0E 07FF                        2146           BR      15              RETURN TO CALLER


000F10 9200 0003      00003        2148 COMPB     MVI   HPEZ,X'00'        SET H/P,E/Z = OFF
000F14 47F0 0EAC            00EAC  2149           B       COMPF           *
000F18 9201 0003      00003        2150 COMPE     MVI   HPEZ,X'01'        SET H/P=OFF,E/Z=ONN
000F1C 47F0 0EAC            00EAC  2151           B       COMPF           *
000F20 9210 0003      00003        2152 COMPB1    MVI   HPEZ,X'10'        SET HP ON,EZ OFF
000F24 47F0 0ECC            00ECC  2153           B       COMPD           *
000F28 9200 0003      00003        2154 COMPE1    MVI   HPEZ,X'00'        SET H/P, E/Z = OFF
000F2C 47F0 0ECC            00ECC  2155           B       COMPD           *


000F30 0000                        2157 ACOMP     DC    X'0000'           BYTE FOR COMPARING P AND Q
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2159 ********************************************************************************
                                  2160 *                                                                              *
                                  2161 *                              BRANCH ON DIGIT                                 *
                                  2162 *                                  BRANCH                                      *
                                  2163 *                                                                              *
                                  2164 ********************************************************************************

000F32 45F0 091E           0091E  2166 BD43     BAL   15,CONVQ              CONVERT Q ADDRESS
000F36 910F 9000     00000        2167          TM    0(RQ),X'0F'           TEST IF DIGIT
000F3A 4780 07F6           007F6  2168          BZ      ENTRY1              NO,GO TO 'BIR'
000F3E 45F0 0906           00906  2169 B49      BAL   15,CONVP              CONVERT P ADDRESS
000F42 1858                       2170          LR    CNTR,RP               SET NEW INSTR. COUNTER
000F44 4250 0F58           00F58  2171 TESTBR   STC   CNTR,STC              TEST IF BRANCH ADDRESS CORRECT
000F48 9101 0F58     00F58        2172          TM    STC,X'01'             *
000F4C 4780 07FA           007FA  2173          BC    8,ENTRY2              YES,BRANCH
000F50 45F0 00F4           000F4  2174          BAL   15,MESTOP             NO,ERROR,STOP SIMULATION
000F54 0000019D                   2175          DC  A(NOBRAD)             *
000F58                            2176 STC      DS    H                     *

                                  2179 ********************************************************************************
                                  2180 *                                                                              *
                                  2181 *                                BRANCH NO FLAG                                *
                                  2182 *                                                                              *
                                  2183 ********************************************************************************

000F5A 45F0 091E           0091E  2185 BNF44    BAL   15,CONVQ              CONVERT Q ADDRESS
000F5E 9120 9000     00000        2186          TM    0(RQ),X'20'           TEST IF FLAG
000F62 4780 07F6           007F6  2187          BZ      ENTRY1              YES,GO TO 'BIR'
000F66 47F0 0F3E           00F3E  2188          B       B49                 NO,BRANCH TO B49

                                  2191 ********************************************************************************
                                  2192 *                                                                              *
                                  2193 *                           BRANCH NO RECORD MARK                              *
                                  2194 *                                                                              *
                                  2195 ********************************************************************************

000F6A 45F0 091E           0091E  2197 BNR45    BAL   15,CONVQ              CONVERT Q ADDRESS
000F6E 910A 9000     00000        2198          TM    0(RQ),X'0A'           TEST IF RECORD MARK
000F72 4710 07F6           007F6  2199          BO      ENTRY1              YES,GO TO 'BIR'
000F76 47F0 0F3E           00F3E  2200          B       B49                 NO,GO TO B49
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2202 ************************************************************************
                                  2203 *                                                                      *
                                  2204 *                          BRANCH INDICATOR                            *
                                  2205 *                                                                      *
                                  2206 *  THE DIGITS Q8 AND Q9 ARE CONVERTED INTO BINARY IN ORDER TO OBTAIN   *
                                  2207 *  AN ADDRESS FOR THE HALFWORD TABLE 'INDICA' AND FOR THE BYTE  TAB-   *
                                  2208 *  LES 'INDICM' AND INDICX'.                                           *
                                  2209 *                                                                      *
                                  2210 *  THE HALFWORD TABLE 'INDICA' PROVIDES THE INDICATOR ADDRESS IN THE   *
                                  2211 *  DATA FIELD OF THE SIMULATOR (SEE UNDER 'CONSOLE SIMULATION' - BE-   *
                                  2212 *  GINNING OF SIMULATION).                                             *
                                  2213 *                                                                      *
                                  2214 *  THE BYTE TABLE 'INDICM' PROVIDES THE MASK WHICH ALLOWS TESTING OF   *
                                  2215 *  THE INDICATOR BIT.                                                  *
                                  2216 *                                                                      *
                                  2217 *  THE BYTE TABLE 'INDICX' PROVIDES AN EXCLUSIVE-OR MASK THAT ALLOWS   *
                                  2218 *  CLEARING OF THE INDICATOR BIT AFTER TESTING,  IF SO REQUIRED.       *
                                  2219 *                                                                      *
                                  2220 ************************************************************************

000F7A 9200 0FBD       00FBD      2222 BI46     MVI   BNIS1+1,0             SET BNIS1 SWITCH OFF
000F7E 9200 0FC9       00FC9      2223          MVI   BNIS2+1,0             SET BNIS2 SWITCH OFF
000F82 4195 0008             00008 2224 BIA      LA    RQ,8(CNTR)            LOAD ADDRESS OF Q8
000F86 910C 9000       00000      2225          TM    0(RQ),X'0C'           TEST INDICATOR VALIDITY
000F8A 4740 0FBC             00FBC 2226          BC    4,BNIS1               BRANCH IF INVALID
000F8E F271 09E0 9000 009E0 00000 2227          PACK  BUFFQ(8),0(2,RQ)      PACK Q8 AND Q9
000F94 960F 09E7       009E7      2228          OI    BUFFQ+7,X'0F'         SET + SIGN
000F98 4FA0 09E0             009E0 2229          CVB   WR1,BUFFQ             CONVERT INTO BINARY
000F9C 18B0                        2230          LR    WR2,0                 CLEAR WR2 REGISTER
000F9E 43BA 0FDC             00FDC 2231          IC    WR2,INDICM(WR1)       GET TESTING MASK AND STORE IT
000FA2 42B0 0FB5             00FB5 2232          STC   WR2,BIC+1             IN TEST UNDER MASK
000FA6 43BA 30A4             010A4 2233          IC    WR2,INDICX(WR1)       GET XOR MASK AND STORE IT  IN
000FAA 42B0 0FC5             00FC5 2234          STC   WR2,BID+1             EXCLUSIVE OR INSTRUCTION
000FAE 1AAA                        2235          AR    WR1,WR1               *
000FB0 48CA 3004             01004 2236          LH    WR3,INDICA(WR1)       LOAD ADDRESS OF TESTED BYTE
000FB4 9100 C000       00000      2237 BIC      TM    0(WR3),X'00'          TEST INDICATOR BYTE
000FB8 4770 0FC4             00FC4 2238          BC    7,BID                 BRANCH IF INDICATOR ON
000FBC 4700 0F3E             00F3E 2239 BNIS1    BC    NOP,B49               SWITCH 1 FOR BNI47
000FC0 47F0 07F6             007F6 2240          B        ENTRY1             *
000FC4 9700 C000       00000      2241 BID      XI    0(WR3),X'00'          SET INDICATOR OFF (IF REQUIRED)
000FC8 4700 07F6             007F6 2242 BNIS2    BC    NOP,ENTRY1            SWITCH 2 FOR BNI47
000FCC 47F0 0F3E             00F3E 2243          B        B49                BRANCH TO B49
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2245 ********************************************************************************
                                  2246 *                                                                              *
                                  2247 *                          BRANCH NO INDICATOR                                 *
                                  2248 *                                                                              *
                                  2249 *  SAME COMMENTS AS FOR 'BRANCH INDICATOR'.                                    *
                                  2250 *                                                                              *
                                  2251 ********************************************************************************

000FD0 92F0 0FBD      00FBD       2253 BNI47    MVI   BNIS1+1,X'F0'          SET BNIS1 SWITCH ON
000FD4 92F0 0FC9      00FC9       2254          MVI   BNIS2+1,X'F0'          SET BNIS2 SWITCH ON
000FD8 47F0 0F82            00F82 2255          B        BIA                 GO TO BI-46



                                  2258 ********************************************************************************
                                  2259 *                                                                              *
                                  2260 *                      TABLES FOR INDICATOR TESTING                            *
                                  2261 *                                                                              *
                                  2262 ********************************************************************************

000FDC 0080402010008040           2264 INDICM   DC    X'0080402010008040'   MASK TABLE
000FE6 0010011110010000           2265          DC    X'0010011110010000'
000FF0 0000000000080000           2266 INDERR   DC    X'0000000000800000'
000FFA 0402010402010402           2267          DC    X'0402010402010402'
001004 00000FF0                   2268 INDICA   DC  A(INDERR)               ERROR
001008 00000001                   2269          DC  A(PROSWI)               INDICATOR 01
00100C 00000001                   2270          DC  A(PROSWI)               INDICATOR 02
001010 00000001                   2271          DC  A(PROSWI)               INDICATOR 03
001014 00000001                   2272          DC  A(PROSWI)               INDICATOR 04
001018 00000FF0                   2273          DC  A(INDERR)               ERROR
00101C 00000002                   2274          DC  A(NYCHEK)               INDICATOR 06
001020 00000002                   2275          DC  A(NYCHEK)               INDICATOR 07
001024 00000FF0                   2276          DC  A(INDERR)               ERROR
001028 00000005                   2277          DC  A(LASTCD)               INDICATOR 09
00102C 00000FF0                   2278          DC  A(INDERR)               ERROR
001030 00000003                   2279          DC  A(HPEZ)                 INDICATOR 11
001034 00000003                   2280          DC  A(HPEZ)                 INDICATOR 12
001038 00000003                   2281          DC  A(HPEZ)                 INDICATOR 13
00103C 00000004                   2282          DC  A(OFLOW)                INDICATOR 14
001040 00000004                   2283          DC  A(OFLOW)                INDICATOR 15
001044 00000FF0                   2284          DC  A(INDERR)               ERROR
001048 00000FF0                   2285          DC  A(INDERR)               ERROR
00104C 00000FF0                   2286          DC  A(INDERR)               ERROR
001050 00000002                   2287          DC  A(NYCHEK)               INDICATOR 19
001054 00000FF0                   2288          DC  A(INDERR)               ERROR
001058 00000FF0                   2289          DC  A(INDERR)               ERROR
00105C 00000FF0                   2290          DC  A(INDERR)               ERROR
001060 00000FF0                   2291          DC  A(INDERR)               ERROR
001064 00000FF0                   2292          DC  A(INDERR)               ERROR
001068 00000002                   2293          DC    A(NYCHEK)             INDICATOR 25
00106C 00000FF0                   2294          DC  A(INDERR)               ERROR
001070 00000FF0                   2295          DC    A(INDERR)             ERROR
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
001074 00000FF0                   2296           DC   A(INDERR)             ERROR
001078 00000FF0                   2297           DC   A(INDERR)             ERROR
00107C 00000006                   2298           DC  A(IXBAND)              INDICATOR 30
001080 00000006                   2299           DC  A(IXBAND)              INDICATOR 31
001084 00000006                   2300           DC  A(IXBAND)              INDICATOR 32
001088 00000007                   2301           DC  A(TAPE43)              INDICATOR 33
00108C 00000007                   2302           DC  A(TAPE43)              INDICATOR 34
001090 00000007                   2303           DC  A(TAPE43)              INDICATOR 35
001094 00000002                   2304           DC  A(NYCHEK)              INDICATOR 36
001098 00000002                   2305           DC  A(NYCHEK)              INDICATOR 37
00109C 00000002                   2306           DC  A(NYCHEK)              INDICATOR 38
0010A0 00000002                   2307           DC  A(NYCHEK)              INDICATOR 39
0010A4 0000000000008040           2308 INDICX    DC    X'0000000000080400001'
0010AE 0000000010010000           2309           DC    X'0000000010010000000'
0010B8 0000000000080000           2310           DC    X'0000000000800000000'
0010C2 0000000402000402           2311           DC    X'0000000402000402010'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2313 ********************************************************************************
                                  2314 *                                                                              *
                                  2315 *                          BRANCH AND TRANSMIT                                 *
                                  2316 *                                                                              *
                                  2317 ********************************************************************************

0010CC 45F0 0902           00902  2319 BT27     BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
0010D0 41A5 000C           0000C  2320 BT27A    LA    WR1,12(CNTR)           WR1=INST.COUNTER+12
0010D4 50A0 3134           01134  2321         ST    WR1,IR2                STORE WR1 IN IR-2
0010D8 1858                       2322         LR    CNTR,RP                INST. COUNTER= P ADDRESS
0010DA 0680                       2323         BCTR  RP,0                   RP=RP-1
0010DC D200 8000 9000 00000 00000 2324         MVC   0(1,RP),0(RQ)          MOVE FIRST BYTE FROM Q TO P
0010E2 0680                       2325 BT27B    BCTR  RP,0                   RP=RP-1
0010E4 0690                       2326         BCTR  RQ,0                   RQ=RQ-1
0010E6 D200 8000 9000 00000 00000 2327         MVC   0(1,RP),0(RQ)          MOVE ONE CHARACTER
0010EC 9120 9000      00000       2328         TM    0(RQ),X'20'            IS Q DIGIT FLAGGED
0010F0 4710 30E2           010E2  2329         BO      BT27B                NO,LOOP
0010F4 92F0 310D      0110D       2330         MVI   SWBB42+1,X'F0'         SET SWITCH IN BRANCH BACK ON
0010F8 47F0 0F44           00F44  2331         B       TESTBR               GO TO TEST BRANCH ADDRESS


                                  2334 ********************************************************************************
                                  2335 *                                                                              *
                                  2336 *                    BRANCH AND TRANSMIT IMMEDIATE                             *
                                  2337 *                                                                              *
                                  2338 ********************************************************************************

0010FC 45F0 0906           00906  2340 BTM17    BAL   15,CONVP               CONVERT P ADDRESS
001100 4195 000B           0000B  2341         LA    RQ,11(CNTR)            Q=RIGHTMOST ADDRESS OF INSTR.
001104 47F0 30D0           010D0  2342         B       BT27A                BRANCH TO BRANCH AND TRANSMIT
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2344 ********************************************************************************
                                  2345 *                                                                              *
                                  2346 *                              BRANCH BACK                                     *
                                  2347 *                                                                              *
                                  2348 *  THIS  SEQUENCE CONTAINS TWO PROGRAMMED SWITCHES =                           *
                                  2349 *                                                                              *
                                  2350 *  - IF THE SAVE KEY IS DEPRESSED, A BRANCH IS EXECUTED                        *
                                  2351 *  - IF SWITCH 'BT27' 'BTM17' IS DEPRESSED AFTER EXECUTION OF A 'BT'           *
                                  2352 *    INSTRUCTION, THE SEQUENCE ALSO EXECUTES A BRANCH.                         *
                                  2353 *                                                                              *
                                  2354 *  IF NONE OF THESE CONDITIONS IS FULFILLED,SIMULATION STOPS AND THE           *
                                  2355 *  MESSAGE 'MAR CHK' IS ISSUED.                                                *
                                  2356 *                                                                              *
                                  2357 ********************************************************************************

001108 4700 3118           01118  2359 BB42     BC    NOP,BB42A              'SAVE KEY' SWITCH ON, BRANCH
00110C 4700 3124           01124  2360 SWBB42   BC    NOP,BB42B              SWITCH IN BT27 AND BTM17 ON,
                                  2361 *                                        BRANCH
001110 45F0 00F4           000F4  2362          BAL   15,MESTOP              SEND MESSAGE 'NO BRANCH ADDRESS'
001114 0000019D                   2363          DC  A(NOBRAD)                 AND STOP SIMULATION
001118 5850 08FC           008FC  2364 BB42A    L     CNTR,PR1               LOAD INST. COUNTER WITH PR-1
00111C 9200 3109      01109        2365          MVI   BB42+1,X'00'           SET SWITCH 'SAVE KEY' OFF
001120 47F0 0F44           00F44  2366          B       TESTBR               GO TO TEST BRANCH ADDRESS
001124 5850 3134           01134  2367 BB42B    L     CNTR,IR2               LOAD INST. COUNTER WITH IR-2
001128 5000 3134           01134  2368          ST    0,IR2                  CLEAR INTERNAL REGISTER IR-2
00112C 9200 310D      0110D        2369          MVI   SWBB42+1,X'00'         SET SWITCH BT27,BTM17 OFF
001130 47F0 0F44           00F44  2370          B       TESTBR               GO TO TEST BRANCH ADDRESS

001134 00000000                   2372 IR2      DC    F'0'                   *
                                  2373          AIF   (NOT &DISK).NODSK8
                                  2374 .NODSK8  ANOP                             COMMON PART OF SIM20
                           00A00  2375 BNG55    EQU   CODERR                 NO DISK
                                  2376 .DSKX2   ANOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2378 ********************************************************************************
                                  2379 *                                                                              *
                                  2380 *                                SECTION 3                                     *
                                  2381 *                          PROGRAM CONTROL OPERATIONS                          *
                                  2382 *                                                                              *
                                  2383 ********************************************************************************



                                  2385 ********************************************************************************
                                  2386 *                                                                              *
                                  2387 *                                SET FLAG                                      *
                                  2388 *                                                                              *
                                  2389 *  AFTER  CONVERSION  OF  THE P  ADDRESS,  A FLAG IS  SET IN  THE  P           *
                                  2390 *  DIGIT ZONE. THIS IS IMPLEMENTED BY AN 'AND IMMEDIATE' INSTRUCTION           *
                                  2391 *  WITH 'DF'.                                                                  *
                                  2392 *                                                                              *
                                  2393 ********************************************************************************

001138 45F0 0906          00906   2395 SF32     BAL   15,CONVP              CONVERT P ADDRESS
00113C 94DF 8000    00000         2396          NI    0(RP),X'DF'           SET FLAG AT P ADDRESS
001140 47F0 07F6          007F6   2397          B       ENTRY1              GO TO 'BIR'



                                  2399 ********************************************************************************
                                  2400 *                                                                              *
                                  2401 *                        CLEAR FLAG AND NO OPERATION                           *
                                  2402 *                                                                              *
                                  2403 *  AFTER  CONVERSION OF THE  P ADDRESS,  AN  'OR IMMEDIATE' INSTRUC-           *
                                  2404 *  TION  WITH AN  'F0' MASK SETS  THE  ZONE PART OF THE P DIGIT TO A           *
                                  2405 *  HEXADECIMAL 'F'.                                                            *
                                  2406 *                                                                              *
                                  2407 *  THE 'NOP41' INSTRUCTION CAUSES A RETURN TO 'BIR' ENTRY1.                    *
                                  2408 *                                                                              *
                                  2409 ********************************************************************************

001144 45F0 0906          00906   2411 CF33     BAL   15,CONVP              CONVERT P ADDRESS
001148 96F0 8000    00000         2412          OI    0(RP),X'F0'           CLEAR FLAG AT P ADDRESS
00114C 47F0 07F6          007F6   2413 NOP41    B       ENTRY1              NO OPERATION, GO TO 'BIR'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2415 ********************************************************************************
                                  2416 *                                                                              *
                                  2417 *                                  HALT                                        *
                                  2418 *                                                                              *
                                  2419 *  THIS INSTRUCTION CAUSES A 'HALT' MESSAGE AND THE 1620  ADDRESS OF           *
                                  2420 *  THE 'HALT' TO BE PRINTED ON THE CONSOLE TYPEWRITER. THE SIMULATOR           *
                                  2421 *  THEN WAITS FOR OPERATOR ACTION.                                             *
                                  2422 *                                                                              *
                                  2423 *  EACH EVENT CAUSING A 'WAIT FOR OPERATOR ACTION'MESSAGE GIVES CON-           *
                                  2424 *  TROL TO A SPECIAL ROUTINE THAT SENDS THE PROPER MESSAGE,   ALTERS           *
                                  2425 *  THE BRANCH INSTRUCTION OF 'BIR' TO A 'NO OPERATION' AND   RETURNS           *
                                  2426 *  TO 'BIR'.                                                                   *
                                  2427 *                                                                              *
                                  2428 ********************************************************************************

001150 18A5                       2430 H48      LR    WR1,CNTR              WR1=CNTR
001152 1BA6                       2431          SR    WR1,MAPORG            WR1=WR1-MAPORG
001154 4EA0 09D8          009D8   2432          CVD   WR1,BUFFP             CONVERT ADDRESS INTO DECIMAL
001158 960F 09DF     009DF        2433          OI    BUFFP+7,X'0F'         SET F SIGN
00115C F342 01AC 09DD 001AC 009DD 2434          UNPK  HLT+7(5),BUFFP+5(3)   *
001162 45F0 00F4          000F4   2435          BAL   15,MESTOP             STOP SIMULATION
001166 0000
001168 000001A5                   2436          DC  A(HLT)                *

                                  2438          AIF   (NOT &TRANS).NOTRN2   ADDIT INSTRUCTIONS

                                  2440 ********************************************************************************
                                  2441 *                                MOVE FLAG                                     *
                                  2442 *                                                                              *
                                  2443 *  AFTER CONVERSION OF THE P AND Q ADDRESSES, THE Q  DIGIT  ZONE  IS           *
                                  2444 *  MOVED TO THE P DIGIT ZONE. THEN, THE Q DIGIT ZONE IS  CLEARED  BY           *
                                  2445 *  AN 'OR IMMEDIATE' WITH A HEXADECIMAL 'F0' UNLESS THE P AND Q  AD-           *
                                  2446 *  DRESSES ARE THE SAME.                                                       *
                                  2447 *                                                                              *
                                  2448 *  NOTE   'MOVE FLAG' IS A SPECIAL FEATURE FOR 1620 MODEL 1.                   *
                                  2449 *                                                                              *
                                  2450 ********************************************************************************

00116C 45F0 0902          00902   2452 MF71     BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
001170 D300 8000 9000 00000 00000 2453          MVZ   0(1,RP),0(RQ)         MOVE FLAG FROM Q TO P
001176 43A8 0000          00000   2454          IC    WR1,0(RP)             *
00117A 96F0 9000     00000        2455          OI    0(RQ),X'F0'           AND CLEAR THE FLAG AT Q ADDRESS
00117E 42A8 0000          00000   2456          STC   WR1,0(RP)             *
001182 47F0 07F6          007F6   2457          B       ENTRY1              GO TO 'BIR'
                                  2458          AGO   .TRNX2
                                  2459 .TRNX2   ANOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2461 ********************************************************************************
                                  2462 *                                                                              *
                                  2463 *                                SECTION 4                                     *
                                  2464 *                           ARITHMETIC OPERATIONS                              *
                                  2465 *                                                                              *
                                  2466 ********************************************************************************



                                  2468 ********************************************************************************
                                  2469 *                                                                              *
                                  2470 *                                   ADD                                        *
                                  2471 *                                                                              *
                                  2472 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY.  THE BINARY  VALUE           *
                                  2473 *  OF THE P AND Q ADDRESSES, AND THE P AND Q SIGNS ARE THEN SAVED IN           *
                                  2474 *  SIMULATED CORE STORAGE. CONTROL IS GIVEN TO THE 'FIXADD'  SUBROU-           *
                                  2475 *  TINE. INDICATORS ARE UPDATED BY MEANS OF THE 'INDIC'  SUBROUTINE.           *
                                  2476 *  THEN,  THE LAST CARRY IS TESTED.                                            *
                                  2477 *                                                                              *
                                  2478 *  -IF IT  IS ZERO AND THE SWITCH 'AQEND' IS 'ON', THE 'ADD' ROUTINE           *
                                  2479 *  RETURNS TO THE 'BIR' ENTRY POINT WHICH MAY BE USED FOR  INDICATOR           *
                                  2480 *  DISPLAY.                                                                    *
                                  2481 *                                                                              *
                                  2482 *  -IF THE LAST CARRY IS ONE,OR IF THE Q FIELD HAS NOT BEEN EXHAUST-           *
                                  2483 *  ED,  THE 'ARCHK' SUBROUTINE IS ENTERED.  THEN,  THE 'ADD' ROUTINE           *
                                  2484 *  EXITS TO THE 'BIR' ENTRY POINT USED FOR INDICATOR DISPLAY.                  *
                                  2485 *                                                                              *
                                  2486 ********************************************************************************

001186 45F0 0902           00902  2488 A21      BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
00118A 9200 31AB      011AB       2489 ADD6A    MVI   ADD4A+1,X'00'          SET SWITCH OFF
00118E D300 33A5 9000 013A5 00000 2490 ADD5A    MVZ   AQSIGN(1),0(RQ)        SAVE Q SIGN
001194 D300 33A4 8000 013A4 00000 2491 ADD1A    MVZ   APSIGN(1),0(RP)        SAVE P SIGN
00119A 5080 09E8           009E8  2492          ST    RP,INITP               SAVE P ADDRESS
00119E 18A8                       2493          LR    WR1,RP                 LOAD WR1 = RP
0011A0 1889                       2494          LR    WR2,RQ                 LOAD WR2=RQ
0011A2 45F0 31F8           011F8  2495          BAL   15,FIXADD              BRANCH TO FIXADD
0011A6 4700 0000           00000  2496          BC    NOP,0                  *
                                  2497          AIF   (NOT &INDEX).NOTIX2    INDEX INSTRUCTIONS
0011AA 4700 3D5C           01D5C  2498 ADD4A    BC    NOP,BXF                RETURN SWITCH FOR INDEX INSTRUCT
0011AE 58A0 09E8           009E8  2499          L     WR1,INITP              WR1= P ADDRESS
                                  2500          AGO   .IXX2
                                  2501 .IXX2    ANOP
0011B2 45F0 3548           01548  2502          BAL   15,INDIC               INDICATOR  UPDATING
0011B6 46E0 31BE           011BE  2503          BCT   WR5,ADD2A              TEST IF CARRY
0011BA 47F0 0E26           00E26  2504 ADD3A    B       C24I                 ARITHMETIC CHECK TESTING
0011BE 95F0 3231      01231       2505 ADD2A    CLI   AQEND+1,X'F0'          IS SWITCH AQEND ON
0011C2 4780 07F6           007F6  2506          BE      ENTRY0               YES,BRANCH TO ENTRY0
0011C6 47F0 31BA           011BA  2507          B     ADD3A                  NO,GO TO ARITH. CHECK
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2510 ************************************************************************
                                  2511 *                                                                      *
                                  2512 *                          ADD IMMEDIATE                               *
                                  2513 *                                                                      *
                                  2514 ************************************************************************

0011CA 45F0 0906           00906  2516 AM11     BAL   15,CONVP              CONVERT P ADDRESS
0011CE 4195 000B           0000B  2517          LA    RQ,11(CNTR)           LOAD RQ = CNTR+11
0011D2 47F0 318A           0118A  2518          B       ADD6A               GO TO ADD


                                  2521 ************************************************************************
                                  2522 *                                                                      *
                                  2523 *                             SUBTRACT                                 *
                                  2524 *                                                                      *
                                  2525 *  BOTH THE P AND Q ADDRESSES ARE CONVERTED TO BINARY. THE Q SIGN IS   *
                                  2526 *  SAVED IN 'AQSIGN',  THEN INVERTED.  THE SEQUENCE FINALLY BRANCHES   *
                                  2527 *  TO THE 'ADD' ROUTINE.                                               *
                                  2528 *                                                                      *
                                  2529 ************************************************************************

0011D6 45F0 0902           00902  2531 S22      BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
0011DA D300 33A5 9000 013A5 00000  2532 S22A     MVZ   AQSIGN(1),0(RQ)       *
0011E0 9720 33A5      013A5        2533          XI    AQSIGN,X'20'          INVERT Q SIGN
0011E4 9200 31AB      011AB        2534          MVI   ADD4A+1,X'00'
0011E8 47F0 3194           01194  2535          B       ADD1A               GO TO ADD


                                  2538 ************************************************************************
                                  2539 *                                                                      *
                                  2540 *                       SUBTRACT IMMEDIATE                             *
                                  2541 *                                                                      *
                                  2542 ************************************************************************

0011EC 45F0 0906           00906  2544 SM12     BAL   15,CONVP              CONVERT P ADDRESS
0011F0 4195 000B           0000B  2545          LA    RQ,11(CNTR)           RQ = CNTR+11
0011F4 47F0 31DA           011DA  2546          B       S22A                GO TO SUBTRACT
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2548          AIF (&MODEL NE 1).NMOD1A          FOR MODEL 1
                                  2549 .NMOD1A  ANOP
                                  2550          AIF (&MODEL NE 2).NMOD2           FOR MODEL 2
                                  2551 ************************************************************************
                                  2552 *                                                                      *
                                  2553 *                    FIXADD SUBROUTINE MODEL 2                         *
                                  2554 *                                                                      *
                                  2555 *  THIS SUBROUTINE IS USED BY THE ADD/SUBTRACT  AND   FLOATING-POINT  *
                                  2556 *  ADD/SUBTRACT OPERATIONS. IT PERFORMS THE ADDITION AND SUBTRACTION  *
                                  2557 *  (DIGIT BY DIGIT) OF TWO DATA FIELDS, USING THE SPECIAL TABLES BE-   *
                                  2558 *  LOW.                                                                *
                                  2559 *                                                                      *
                                  2560 *  EXITS                                                               *
                                  2561 *  THERE ARE TWO EXITS,  A NORMAL ONE, AND  A SPECIAL ONE USED AFTER   *
                                  2562 *  COMPLEMENTING. THE SPECIAL EXIT RETURNS TO THE CALLING   SEQUENCE   *
                                  2563 *                                                                      *
                                  2564 *  OPERATION                                                           *
                                  2565 *  THIS SEQUENCE STARTS WITH A TEST OF 'APSIGN' AND 'AQSIGN'.          *
                                  2566 *  IF THESE SIGNS ARE BOTH POSITIVE OR NEGATIVE,CVQ SWITCH IS SET ON   *
                                  2567 *  IF NOT CVQ SWITCH IS SET OFF FOR INSERTION OF THE NINES COMPLEM-    *
                                  2568 *  ENT OF Q DIGIT INTO A WORKING REGISTER.                             *
                                  2569 *                                                                      *
                                  2570 *  THEN, CONTROL IS GIVEN TO THE 'ADDPQ' SUBROUTINE TO  PROCESS  THE   *
                                  2571 *  FIRST P AND Q DIGITS.                                               *
                                  2572 *                                                                      *
                                  2573 *  FOR EASE OF UNDERSTANDING, THE 'FIXADD' SUBROUTINE HAS BEEN   DI-   *
                                  2574 *  VIDED INTO  FOUR LEVELS =                                           *
                                  2575 *                                                                      *
                                  2576 *  LEVEL 1   AQEND - APEND                                             *
                                  2577 *  A TEST IS MADE TO DETECT A FLAG INDICATING THE LEFTMOST  BOUNDARY   *
                                  2578 *  OF THE Q FIELD. IF THERE IS NO FLAG,  THE  Q ADDRESS  REGISTER IS   *
                                  2579 *  DECREMENTED BY ONE. IF THERE IS A FLAG, THE Q ADDRESS REGISTER IS   *
                                  2580 *  LOADED WITH THE ADDRESS OF A DIGIT 'F0'.  THEN, SWITCH 'AQEND' IS   *
                                  2581 *  SET ON TO BYPASS THE P FLAG TEST AND DECREMENTING  OF  THE  Q AD-   *
                                  2582 *  DRESS.                                                              *
                                  2583 *                                                                      *
                                  2584 *  P FIELD TEST - SAME COMMENTS.                                       *
                                  2585 *                                                                      *
                                  2586 *                                                          ./...       * ..
                                  2587 *                                                                      *
                                  2588 ************************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2590 ************************************************************************
                                  2591 *                                                                      *
                                  2592 *    ./...                                                             * ..
                                  2593 *                                                                      *
                                  2594 *  LEVEL 2   ADDC                                                      *
                                  2595 *  THE BYTE LOCATED AT ADDRESS 'WR1' ( (WHICH  MAY BE EQUAL TO THE P   *
                                  2596 *  ADDRESS) IS TESTED FOR A FLAG AND THE PRECEDING  ADD  RESULT   IS   *
                                  2597 *  STORED AT THE ADDRESS OF 'WR1' BEFORE THE BRANCH. IF THERE IS  NO   *
                                  2598 *  FLAG, 'WR1' IS DECREMENTED BY 1 AND CONTROL IS GIVEN BACK TO  LE-   *
                                  2599 *  VEL 1.  IF THERE IS A FLAG, THE ADD LOOP IS TERMINATED AND  CONT-   *
                                  2600 *  ROL IS GIVEN TO LEVEL 3.                                            *
                                  2601 *                                                                      *
                                  2602 *  LEVEL 3   ADDA                                                      *
                                  2603 *  THE P ADDRESS REGISTER IS RESTORED TO ITS INITIAL VALUE.IF COMPL-   *
                                  2604 *   EMENTING IS REQUIRED , A TEST IS MADE ON THE LAST CARRY.           *
                                  2605 *  -IF THE LAST CARRY IS ONE,  IT IS ADDED TO THE RESULTING  P FIELD   *
                                  2606 *  WHICH BEGINS AT THE RIGHTMOST BOUNDARY. THIS OPERATION IS REPEAT-   *
                                  2607 *  ED UNTIL EXHAUSTION OF THE CARRY. THE P SIGN IS RESTORED AND  THE   *
                                  2608 *  'FIXADD' SUBROUTINE EXITS TO THE CALLING SEQUENCE.                  *
                                  2609 *  -IF THE LAST CARRY IS ZERO, CONTROL IS GIVEN TO THE RECOMPLEMENT-   *
                                  2610 *  ING LOOP WHICH SCANS THE RESULTING P FIELD AND PROCEEDS TO SYSTE-   *
                                  2611 *  MATIC NINES COMPLEMENTING.THE P FLAG IS RESET AND A TEST IS DONE    *
                                  2612 *  ON THE VALUE OF RESULT,IF ZERO,P SIGN IS RESTORED,IF DIFFERENT      *
                                  2613 *  Q SIGN IS GIVEN TO P FIELD,AND FIXADD EXITS TO THE CALLING SEQ.     *
                                  2614 *  -IF COMPLEMENTING IS NOT REQUIRED,THE P FLAG IS RESET,AND P SIGN    *
                                  2615 *  IS RESTORED IN THE RESULTING FIELD AND FIXADD EXITS TO THE CALL-    *
                                  2616 *  ING SEQUENCE OF THE FOLLOWING INSTRUCTION.                          *
                                  2617 *                                                                      *
                                  2618 ************************************************************************
0011F8 9200 321F       0121F      2619 FIXADD   MVI   ADD5+1,X'00'          SET SWITCH  ADD5 OFF
0011FC D500 33A4 33A5  013A4 013A5 2620          CLC   APSIGN(1),AQSIGN      COMPARE P AND Q SIGNS
001202 4780 320E             0120E 2621          BE      ADD3                EQUAL BRANCH
001206 9200 32ED       012ED      2622 ADD1     MVI   CVQ+1,X'00'           SET SWITCH FOR COMPLEMENTATION
00120A 47F0 3212             01212 2623          B       ADD4                *
00120E 92F0 32ED       012ED      2624 ADD3     MVI   CVQ+1,X'F0'           SET SWITCH OFF


001212 9200 323F       0123F      2626 ADD4     MVI   APEND+1,X'00'          SET SWITCH APEND OFF
001216 9200 3231       01231      2627          MVI   AQEND+1,X'00'          SET SWITCH AQEND OFF
00121A 98CE 34D8             014D8 2628          LM    WR3,WR5,MCLEAR        SET WR3,WR4,WR5=0
00121E 4700 322C             0122C 2629 ADD5     BC    NOP,ADDF              SWITCH FOR USE OF FLOATING ADD
001222 4510 32E8             012E8 2630          BAL   R1,ADDPQ              BRANCH TO ADDPQ
001226 0680                        2631          BCTR  RP,0                  RP=RP-1
001228 06A0                        2632          BCTR  WR1,0                 WR1=WR1-1
00122A 06B0                        2633          BCTR  WR2,0                 WR2=WR2-1
00122C 4510 32E8             012E8 2634 ADDF     BAL   R1,ADDPQ              BRANCH TO ADDPQ
001230 4700 323E             0123E 2635 AQEND    BC    NOP,APEND             SWITCH AQEND
001234 9120 B000       00000      2636          TM    0(WR2),X'20'          IS Q DIGIT FLAGGED
001238 4780 32D0             012D0 2637          BZ      ADDB                YES,GO TO SET AQEND ON
00123C 06B0                        2638          BCTR  WR2,0                 NO,WR2=WR2-1
00123E 4700 324C             0124C 2639 APEND    BC    NOP,ADDC              SWITCH APEND
001242 9120 B000       00000      2640          TM    0(RP),X'20'           IS P FIELD FLAGGED
001246 4780 32DC             012DC 2641          BZ      ADDE                YES,GO TO SET SWITCH APEND ON
00124A 0680                        2642          BCTR  RP,0                  NO,RP=RP-1
00124C 9120 A000       00000      2643 ADDC     TM    0(WR1),X'20'          IS P FIELD FLAGGED
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
001250 4780 3258            01258  2644          BZ      ADDA               YES,GO TO LAST CARRY LOOP
001254 46A0 322C            0122C  2645          BCT   WR1,ADDF             NO,WR1=WR1-1 AND LOOP


001258 5880 09E8            009E8  2647 ADDA     L     RP,INITP             RESTORE P ADDRESS
00125C 95F0 32ED      012ED        2648          CLI   CVQ+1,X'F0'          WAS COMPLEMENTATION REQUIRED
001260 4780 327E            0127E  2649          BE      ADDG               NO,GO TO RESTORE P SIGN
001264 18C8                        2650          LR    WR3,RP               YES,WR3=RP
001266 46E0 3294            01294  2651          BCT   WR5,ADDH             TEST IF LAST CARRY
00126A 9109 C000      00000        2652 ADDM     TM    0(WR3),X'09'         ONE,IS P NUMERIC=9
00126E 4710 328C            0128C  2653          BC    1,ADDL               IF EQUAL NORMALISE
001272 43DC 0000            00000  2654          IC    WR4,0(WR3)           INSERT 0(WR3) IN WR4
001276 41DD 0001            00001  2655          LA    WR4,1(WR4)           ADD CARRY
00127A 42DC 0000            00000  2656          STC   WR4,0(WR3)           AND STORE RESULT
00127E D300 8000 33A4 00000 013A4  2657 ADDG     MVZ   0(1,RP),APSIGN       RESTORE P SIGN
001284 94DF A000      00000        2658          NI    0(WR1),X'DF'         RESET P FLAG
001288 47FF 0004            00004  2659          B       4(15)              RETURN TO CALLING SEQUENCE + 4


00128C 92F0 C000      00000        2661 ADDL     MVI   0(WR3),X'F0'         NORMALIZE RESULT
001290 46C0 326A            0126A  2662          BCT   WR3,ADDM             WR3=WR3-1, AND LOOP
001294 41EE 0001            00001  2663 ADDH     LA    WR5,1(WR5)           SET CARRY TO INITIAL VALUE
001298 DC00 C000 3264 00000 01264  2664          TR    0(1,WR3),ATAB3-208   COMPLEMENT P DIGIT
00129E 19AC                        2665          CR    WR1,WR3              IS WR3=WR1
0012A0 4780 32A8            012A8  2666          BE      ADDN               YES,RESTORE RQ
0012A4 46C0 3298            01298  2667          BCT   WR3,ADDH+4           NO,LOOP
0012A8 94DF C000      00000        2668 ADDN     NI    0(WR3),X'DF'         RESET P FLAG
0012AC 18C8                        2669          LR    WR3,RP               WR3 = RP
0012AE 910F C000      00000        2670 ADDT     TM    0(WR3),X'0F'         TEST IF P NUMERIC = 0
0012B2 4770 32C8            012C8  2671          BC    7,ADDQS              IF NOT,BRANCH
0012B6 19AC                        2672          CR    WR1,WR3              IS WR3 = WR1
0012B8 4780 32C0            012C0  2673          BE      ADDPS              YES,EXIT
0012BC 46C0 32AE            012AE  2674          BCT   WR3,ADDT             NO,LOOP
0012C0 D300 8000 33A4 00000 013A4  2675 ADDPS    MVZ   0(1,RP),APSIGN       MOVE APSIGN TO P SIGN
0012C6 07FF                        2676          BR      15                 RETURN TO CALLING SEQUENCE
0012C8 D300 8000 33A5 00000 013A5  2677 ADDQS    MVZ   0(1,RP),AQSIGN       MOVE AQSIGN TO P SIGN
0012CE 07FF                        2678          BR      15                 RETURN TO CALLING SEQUENCE


0012D0 92F0 3231      01231        2680 ADDB     MVI   AQEND+1,X'F0'        SET SWITCH AQEND ON
0012D4 41B0 3374            01374  2681          LA    WR2,ATAB2            WR2=ADDRESS OF A'F0' DIGIT
0012D8 47F0 323E            0123E  2682          B       APEND              *
0012DC 92F0 323F      0123F        2683 ADDE     MVI   APEND+1,X'F0'        SET SWITCH APEND ON
0012E0 4180 3374            01374  2684          LA    RP,ATAB2             RP=ADDRESS OF A 'F0'
0012E4 47F0 324C            0124C  2685          B       ADDC               *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2687 ************************************************************************
                                  2688 *                                                                      *
                                  2689 *  LEVEL 4 ADDPQ                                                       *
                                  2690 *  THE BINARY EQUIVALENT OF THE Q DIGIT IS DIRECTLY OBTAINED,OR ITS    *
                                  2691 *  NINES COMPLEMENT FROM ATAB3 IF COMPLEMENTATION REQUIRED,AND INS-    *
                                  2692 *  ERTED INTO A WORKING REGISTER.THE BINARY EQUIVALENT OF THE P DI-     *
                                  2693 *  GIT IS OBTAINED FROM TABLE MTAB1 AND INSERTED INTO A SECOND WOR-    *
                                  2694 *  KING REGISTER.                                                      *
                                  2695 *  THE FIRST WORKING REGISTER IS ADDED TO THE SECOND ONE AND THE RE-   *
                                  2696 *  SULT IS ADDED TO A POSSIBLE CARRY OCCURRING IN THE PRECEDING OPER-  *
                                  2697 *  ATION.A TEST IS DONE TO DETECT A POSSIBLE CARRY AND TO STORE IT     *
                                  2698 *  IN AN APPROPRIATE REGISTER.THEN THE NUMERIC OF THE RESULT IS MO -   *
                                  2699 *  VED TO THE ADDRESS OF THE P DIGIT.                                  *
                                  2700 *                                                                      *
                                  2701 ************************************************************************

0012E8 43DB 0000            00000 2703 ADDPQ    IC    WR4,0(WR2)            INSERT BINARY   .                ..
0012EC 4700 32F4            012F4 2704 CVQ      BC    NOP,CVQ+8             SWITCH FOR COMPLEMENTATION
0012F0 43DD 3264            01264 2705          IC    WR4,ATAB3-208(WR4)      .EQUIVALENT OF Q               ..
0012F4 54D0 34D4            014D4 2706 ADDPQX   N     WR4,CUTZ              CUT ZONE OF Q DIGIT
0012F8 43CA 0000            00000 2707          IC    WR3,0(WR1)            INSERT BINARY  .                 ..
0012FC 43CC 3434            01434 2708          IC    WR3,MTAB1-208(WR3)      .EQUIVALENT OF P               ..
001300 1ACD                       2709          AR    WR3,WR4               ADD DIGITS
001302 1ACE                       2710          AR    WR3,WR5               ADD CARRY
001304 18E0                       2711          LR    WR5,0                 RESTORE CARRY
001306 41D0 0009            00009 2712          LA    WR4,9                 *
00130A 19CD                       2713          CR    WR3,WR4               TEST IF RESULT LOWER THAN 9
00130C 47D0 3318            01318 2714          BC    13,ADDZ               YES,BRANCH
001310 41E0 0001            00001 2715          LA    WR5,1                 NO, SET CARRY
001314 1BCD                       2716          SR    WR3,WR4               SUBTRACT 10
001316 06C0                       2717          BCTR  WR3,0                 *
001318 42C0 0F58            00F58 2718 ADDZ     STC   WR3,STC               MOVE NUMERIC   .                 ..
00131C D100 A000 0F58 00000 00F58 2719          MVN   0(1,WR1),STC            .TO RESULT                     ..
001322 07F1                       2720          BR      R1                  *
                                  2721 .NMOD2   ANOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2723 ********************************************************************************
                                  2724 *                                                                              *
                                  2725 *                    TABLES FOR 'FIXADD' SUBROUTINE                            *
                                  2726 *                                                                              *
                                  2727 ********************************************************************************

001324 000A141E28323C46           2729 ATAB1    DC    X'000A141E28323C46505A'     *TABLE FOR 10*P
00132E 000A28322832               2730          DC    X'000A28322832'                 ID
001334 F9F8F7F6F5F4F3F2           2731 ATAB3    DC    X'F9F8F7F6F5F4F3F2F1F0'     COMPLEMENT TABLE
00133E F9F8F5F4F5F4               2732          DC    X'F9F8F5F4F5F4'                 ID
001344 000A141E28323C46           2733          DC    X'000A141E28323C46505A'     *TABLE FOR 10*P
00134E 000A28322832               2734          DC    X'000A28322832'                 ID
001354 F9F8F7F6F5F4F3F2           2735          DC    X'F9F8F7F6F5F4F3F2F1F0'
00135E F9F8F5F4F5F4               2736          DC    X'F9F8F5F4F5F4'
001364 0101010101010101           2737 ACARRY   DC    X'0101010101010101'         ***TABLE FOR CARRY
00136E 010101010101               2738          DC    X'010101010101'                 ID
001374 F0F1F2F3F4F5F6F7           2739 ATAB2    DC    X'F0F1F2F3F4F5F6F7F8F9'     NORMAL TABLE
00137E F0F1F4F5F4F5               2740          DC    X'F0F1F4F5F4F5'                 ID
001384 0000000000000000           2741          DC    X'00000000000000000000'     ***TABLE FOR CARRY
00138E 010101010101               2742          DC    X'010101010101'                 ID
001394 F0F1F2F3F4F5F6F7           2743          DC    X'F0F1F2F3F4F5F6F7F8F9'
00139E F0F1F4F5F4F5               2744          DC    X'F0F1F4F5F4F5'
0013A4 00                         2745 APSIGN   DC    X'00'                   P SIGN
0013A5 00                         2746 AQSIGN   DC    X'00'                   Q SIGN

                                  2748 *        NOTE = IN SUBTRACT OPERATION Q SIGN TAKES OPPOSITE VALUE
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2750 ************************************************************************
                                  2751 *                                                                      *
                                  2752 *                              MULTIPLY                                *
                                  2753 *                                                                      *
                                  2754 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY. THEN,  THE  'MULT'   *
                                  2755 *  SUBROUTINE IS ENTERED, FOLLOWED BY THE 'INDIC' SUBROUTINE.  AFTER   *
                                  2756 *  INDICATOR UPDATING, EXIT IS TO THE 'BIR' ENTRY POINT WHICH MAY BE   *
                                  2757 *  USED FOR INDICATOR DISPLAY (SEE 'ADD' ROUTINE).                     *
                                  2758 *                                                                      *
                                  2759 ************************************************************************

0013A6 45F0 0902          00902   2761 M23      BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
0013AA 45F0 33C4          013C4   2762 M23A     BAL   15,MULT               MULTIPLY P FIELD BY Q FIELD
0013AE 188A                       2763          LR    RP,WR1                LOAD RP = WR1 (ADDRESS 99)
0013B0 45F0 3548          01548   2764          BAL   15,INDIC              INDICATOR  UPDATING
0013B4 47F0 07F6          007F6   2765          B       ENTRY0              INTERPRETIVE LOOP (ENTRY0)
                          00570   2766 FDIC     EQU   MASKRG                *

                                  2769 ************************************************************************
                                  2770 *                                                                      *
                                  2771 *                          MULTIPLY IMMEDIATE                          *
                                  2772 *                                                                      *
                                  2773 ************************************************************************

0013B8 45F0 0906          00906   2775 MM13     BAL   15,CONVP              CONVERT P ADDRESS
0013BC 4195 000B          0000B   2776          LA    RQ,11(CNTR)           LOAD RQ WITH CNTR + 11
0013C0 47F0 33AA          013AA   2777          B       M23A                BRANCH TO MULTIPLY
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
                                2779 ************************************************************************
                                2780 *                                                                      *
                                2781 *                          'MULT' SUBROUTINE                           *
                                2782 *                                                                      *
                                2783 *  FUNCTION                                                            *
                                2784 *  THIS SUBROUTINE IS USED BY FIXED-POINT AND FLOATING-POINT MULTIPLY  *
                                2785 *  OPERATIONS.                                                         *
                                2786 ************************************************************************
                                2787 *                                                                      *
                                2788 *                                                                      *
                                2789 *  OPERATION                                                           *
                                2790 *  POSITIONS 80 TO 99 (PRODUCT AREA) ARE FIRST CLEARED. THE  P  SIGN   *
                                2791 *  AND THE P  AND Q ADDRESSES ARE SAVED.  THE  MULTIPLICATION  LOOP    *
                                2792 *  USES TWO SCANNERS. THE FIRST ONE POINTS TO THE RIGHTMOST  ADDRESS   *
                                2793 *  OF THE PARTIAL PRODUCT AND SHIFTS TO THE LEFT EACH TIME THE Q AD-   *
                                2794 *  DRESS IS DECREMENTED BY 1. THE SECOND SCANNER POINTS TO EACH PAR-    *
                                2795 *  TIAL PRODUCT FORMED BY THE ADDITION OF THE P FIELD DIGITS  MULTI-    *
                                2796 *  PLIED BY A GIVEN Q DIGIT. THE BINARY EQUIVALENT OF  THE  P  DIGIT   *
                                2797 *  MULTIPLIED BY 10 IS USED AS AN ADDRESS FOR THE  MULTIPLY  TABLE.    *
                                2798 *  (MAP ORIGIN + 100 + 10XQ).  THIS  BASE  ADDRESS REMAINS  CONSTANT   *
                                2799 *  DURING SCANNING OF THE P FIELD.THE TWO DIGITS RESULTING FROM EACH   *
                                2800 *  DIGIT-BY-DIGIT MULTIPLICATION ARE OBTAINED BY ADDING  THE  BINARY   *
                                2801 *  EQUIVALENT OF THE  P DIGIT TO THIS BASE ADDRESS.IF A CARRY OCCURS   *
                                2802 *  THIS IS CONSIDERED IN EACH PARTIAL PRODUCT.                         *
                                2803 *                                                                      *
                                2804 *  AN  EXTERNAL  LOOP IS USED  IN ORDER TO  SCAN THE  Q FIELD  UNTIL   *
                                2805 *  A FLAG OTHER THAN A SIGN FLAG IS ENCOUNTERED. THIS LOOP  INCLUDES   *
                                2806 *  A SECOND SMALLER LOOP SCANNING THE P FIELD UNTIL DETECTION OF   A   *
                                2807 *  FLAG. CARRIES RECURRING OVER THE WHOLE PRODUCT AREA ARE TAKEN IN-    *
                                2808 *  TO ACCOUNT.                                                         *
                                2809 *                                                                      *
                                2810 *  WHEN THE MULTIPLY LOOP IS TERMINATED, THE LEFTMOST FLAG IS STORED   *
                                2811 *  AND THE RESULT SIGN IS COMPUTED AND STORED  BEFORE  EXIT  TO  THE   *
                                2812 *  CALLING SEQUENCE.                                                   *
                                2813 *                                                                      *
                                2814 ************************************************************************

0013C4 92F0 6050      00050     2816 MULT     MVI   80(MAPORG),X'F0'       CLEAR PRODUCT AREA
0013C8 D212 6051 6050 00051 00050 2817        MVC   81(19,MAPORG),80(MAPORG)
0013CE D300 33A4 8000 013A4 00000 2818        MVZ   APSIGN(1),0(RP)        SAVE P SIGN
0013D4 96F0 8000      00000     2819        OI    0(RP),X'F0'            DELETE P SIGN
0013D8 9089 09E8            09E8 2820        STM   RP,RQ,INITP            SAVE RP AND RQ
0013DC 50F0 34D0            14D0 2821        ST    15,REG15               SAVE RETURN ADDRESS
0013E0 1810                      2822        LR    R1,0                   CLEAR REGISTER 1
0013E2 4126 0063            00063 2823        LA    R2,99(MAPORG)          LOAD R2 WITH ADDRESS OF CORE
                                2824 *                                        STORAGE LOCATION 99

0013E6 18A2                      2826 MULTF    LR    WR1,R2                 LOAD WR1 WITH REGISTER 2
0013E8 98CF 34D8            14D8 2827        LM    WR3,WR6,MCLEAR         CLEAR 4 WORKING REGISTERS
0013EC 43C9 0000            00000 2828        IC    WR3,0(RQ)              INSERT Q DIGIT IN WR3
0013F0 43CC 3424            01424 2829        IC    WR3,MQTAB-208(WR3)     FORM ADDRESS OF
0013F4 41C6 C000            00000 2830        LA    WR3,0(MAPORG,WR3)      MULTIPLY TABLE

0013F8 188A                      2832 MULTD    LR    WR2,WR1                LOAD SCANNER WR2 = WR1
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
0013FA 43D8 0000            00000 2833          IC    WR4,0(RP)           INSERT BINARY
0013FE 43DD 3254            01254 2834          IC    WR4,ATAB1-208(WR4)  EQUIVALENT OF 10*P
001402 431D C000            00000 2835          IC    R1,0(WR4,WR3)       GET FIRST DIGIT OF RESULT
001406 5610 34F0            014F0 2836          O     R1,RFLAG            RESET FLAG                  V1L2
00140A 43FB 0000            00000 2837          IC    WR6,0(WR2)          INSERT PRODUCT AREA DIGIT
00140E 43FF 3434            01434 2838          IC    WR6,MTAB1-208(WR6)  GET ITS BINARY EQUIVALENT
001412 1A1F                       2839          AR    R1,WR6              ADD THE TWO DIGITS
001414 1A1E                       2840          AR    R1,WR5              ADD CARRY
001416 4311 3444            01444 2841          IC    R1,MTAB2-240(R1)    *
00141A 5410 34EC            014EC 2842          N     R1,RCLEAR           CLEAR CARRY (IF ANY)
00141E 43E1 3294            01294 2843          IC    WR5,ACARRY-208(R1)  SET NEW CARRY(IF ANY)
001422 4311 32A4            012A4 2844          IC    R1,ATAB2-208(R1)    NORMALIZE RESULT
001426 421B 0000            00000 2845          STC   R1,0(WR2)           STORE RESULT IN PRODUCT AREA
00142A 06B0                       2846          BCTR  WR2,0               *
00142C 431D C001            00001 2847          IC    R1,1(WR4,WR3)       GET 2ND DIGIT OF RESULT
001430 5610 34F0            014F0 2848          O     R1,RFLAG            RESET FLAG                  V1L2
001434 4311 3434            01434 2849          IC    R1,MTAB1-208(R1)    INSERT PRODUCT AREA DIGIT
001438 43FB 0000            00000 2850          IC    WR6,0(WR2)          INSERT PRODUCT AREA DIGIT
00143C 43FF 3434            01434 2851          IC    WR6,MTAB1-208(WR6)  GET ITS BINARY EQUIVALENT
001440 1A1F                       2852          AR    R1,WR6              ADD TWO DIGITS

001442 1A1E                       2854 MULTB    AR    R1,WR5              ADD CARRY
001444 4311 3534            01534 2855          IC    R1,MTAB2(R1)        *
001448 43E1 3294            01294 2856          IC    WR5,ACARRY-208(R1)  SET NEW CARRY(IF ANY)
00144C 4311 32A4            012A4 2857          IC    R1,ATAB2-208(R1)    NORMALIZE RESULT
001450 421B 0000            00000 2858          STC   R1,0(WR2)           STORE IT IN PRODUCT AREA
001454 12EE                       2859          LTR   WR5,WR5             TEST LAST CARRY
001456 4780 3468            01468 2860          BZ      MULTA             IF NONE, BRANCH
00145A 06B0                       2861          BCTR  WR2,0               *
00145C 431B 0000            00000 2862          IC    R1,0(WR2)           INSERT FOLLOWING DIGIT
001460 4311 3434            01434 2863          IC    R1,MTAB1-208(R1)    GET ITS BINARY EQUIVALENT
001464 47F0 3442            01442 2864          B       MULTB             *

001468 9120 8000      00000       2866 MULTA    TM    0(RP),X'20'         IS P DIGIT FLAGGED
00146C 4780 3476            01476 2867          BZ      MULTC             YES,BRANCH
001470 0680                       2868          BCTR  RP,0                NO,RP=RP-1
001472 46A0 33F8            013F8 2869          BCT   WR1,MULTD           WR1=WR1-1 AND LOOP

001476 9120 9000      00000       2871 MULTC    TM    0(RQ),X'20'         IS Q DIGIT FLAGGED
00147A 4710 34B6            014B6 2872          BO      MULTE             NO,BRANCH
00147E 5990 09EC            009EC 2873          C     RQ,INITQ            YES, IS RQ = INITIAL VALUE
001482 4780 34B6            014B6 2874          BE      MULTE             YES,BRANCH
001486 94DF B000      00000       2875          NI    0(WR2),X'DF'        NO, SET FLAG ON PRODUCT AREA
00148A 41A6 0063            00063 2876          LA    WR1,99(MAPORG)      LOAD WR1=ADDR. OF PROD.AREA
00148E 5080 0570            00570 2877          ST    RP,FDIC             SAVE RP (FMUL)
001492 9689 09E8            009E8 2878          LM    RP,RQ,INITP         RESTORE P AND QADDRESSES
001496 D300 8000 33A4 00000 013A4 2879          MVZ   0(1,RP),APSIGN      RESTORE P SIGN
00149C 9120 8000      00000       2880          TM    0(RP),X'20'         IS P FIELD POSITIVE
0014A0 4710 34C0            014C0 2881          BO      MULT1             YES,TEST Q SIGN
0014A4 9120 9000      00000       2882          TM    0(RQ),X'20'         NO, IS Q FIELD POSITIVE
0014A8 4710 34C8            014C8 2883          BO      MULT2             YES,SET SIGN
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
0014AC 96F0 A000      00000        2885 MULT3    OI    0(WR1),X'F0'          SET + SIGN IN PROD. AREA
0014B0 58F0 34D0            014D0  2886 MULT4    L     15,REG15              RESTORE RETURN ADDR.
0014B4 07FF                        2887          BR      15                  RETURN TO CALLER

0014B6 5880 09E8            009E8  2889 MULTE    L     RP,INITP              RESTORE P ADDRESS
0014BA 0620                        2890          BCTR  R2,0                  R2=R2-1
0014BC 4690 33E6            013E6  2891          BCT   RQ,MULTF              RQ=RQ-1,LOOP
0014C0 9120 9000      00000        2892 MULT1    TM    0(RQ),X'20'           IS Q FIELD POSITIVE
0014C4 4710 34AC            014AC  2893          BO      MULT3               YES,GO TO SET + SIGN
0014C8 94DF A000      00000        2894 MULT2    NI    0(WR1),X'DF'          NO, SET MINUS SIGN
0014CC 47F0 34B0            014B0  2895          B       MULT4               *

0014D0                             2897 REG15    DS    F                     *
0014D4 0000000F                    2898 CUTZ     DC    X'0000000F'
0014D8 0000000000000000            2899 MCLEAR   DC    5F'0'                 *

0014EC 000000FF                    2901 RCLEAR   DC    X'000000FF'           *
0014F0 000000F0                    2902 RFLAG    DC    X'000000F0'                                       V1L2

                                   2904 ************************************************************************
                                   2905 *                                                                      *
                                   2906 *                    TABLES FOR 'MULT' SUBROUTINE                      *
                                   2907 *                                                                      *
                                   2908 ************************************************************************

0014F4 6466686A6CC8CACC            2910 MQTAB    DC    X'6466686A6CC8CACC'   *
0014FC CED064666CC86CC8            2911          DC    X'CED064666CC86CC8'   *
001504 0001020304050607            2912 MTAB1    DC    X'0001020304050607'   **
00150C 0809000104050405            2913          DC    X'0809000104050405'   **
001514 6466686A6CC8CACC            2914          DC    X'6466686A6CC8CACC'   *
00151C CED064666CC86CC8            2915          DC    X'CED064666CC86CC8'   *
001524 0001020304050607            2916          DC    X'0001020304050607'   **
00152C 0809000104050405            2917          DC    X'0809000104050405'   **
001534 F0F1F2F3F4F5F6F7            2918 MTAB2    DC    X'F0F1F2F3F4F5F6F7F8F9'
00153E D0D1D2D3D4D5D6D7            2919          DC    X'D0D1D2D3D4D5D6D7D8D9'
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2921 ********************************************************************************
                                  2922 *                                                                              *
                                  2923 *                          'INDIC' SUBROUTINE                                  *
                                  2924 *                                                                              *
                                  2925 *  FUNCTION                                                                    *
                                  2926 *  THIS SUBROUTINE IS, IN GENERAL, USED BY  BY ARITHMETIC AND FLOAT-           *
                                  2927 *  ING POINT INSTRUCTIONS AND UPDATES THE HIGH/POSITIVE AND   EQUAL/           *
                                  2928 *  ZERO INDICATORS UPON COMPLETION OF THE OPERATION.                           *
                                  2929 *                                                                              *
                                  2930 *  OPERATION                                                                   *
                                  2931 *  THIS SUBROUTINE STARTS WITH A PRELIMINARY SCAN OF THE P FIELD FOR           *
                                  2932 *  ZERO RESULT. IF THE SCAN IS ENTIRELY SUCCESSFUL,   THE  SIMULATED           *
                                  2933 *  EQUAL/ZERO INDICATOR IS SET  ON,  AND THE SIMULATED HIGH/POSITIVE           *
                                  2934 *  INDICATOR IS SET  OFF.  IF A NON-ZERO DIGIT IS ENCOUNTERED DURING           *
                                  2935 *  THE SCAN, THE SIMULATED EQUAL/ZERO INDICATOR IS SET OFF,  AND THE           *
                                  2936 *  SIMULATED HIGH/POSITIVE INDICATOR IS SET  ON  OR  OFF,  DEPENDING           *
                                  2937 *  ON THE PRECEDING SIGN TEST OF THE P FIELD.  THEN,  THIS  SEQUENCE           *
                                  2938 *  EXITS TO THE CALLING SUBROUTINE.                                            *
                                  2939 *                                                                              *
                                  2940 ********************************************************************************

001548 910F A000      00000       2942 INDIC    TM    0(WR1),X'0F'          IS FIRST WR1 NUMERIC = 0
00154C 4750 3568            01568 2943          BC    BOM,IND2              NO, BRANCH
001550 06A0                       2944 IND1     BCTR  WR1,0                 YES, WR1 = WR1 - 1
001552 910F A000      00000       2945          TM    0(WR1),X'0F'          IS CURRENT WR1 NUMERIC = 0
001556 4750 3568            01568 2946          BC    BOM,IND2              NO, BRANCH
00155A 9120 A000      00000       2947          TM    0(WR1),X'20'          YES, IS WR1 DIGIT FLAGGED
00155E 4710 3550            01550 2948          BO       IND1               NO, LOOP
001562 9201 0003      00003       2949          MVI   HPEZ,X'01'            YES, PUT EZ=ON, HP=OFF
001566 07FF                       2950          BR       15                 RETURN TO CALLER
001568 9120 8000      00000       2951 IND2     TM    0(RP),X'20'           IS P FIELD POSITIVE
00156C 4780 3576            01576 2952          BZ       IND3               NO, BRANCH
001570 9210 0003      00003       2953          MVI   HPEZ,X'10'            YES, PUT EZ=OFF, HP=ON
001574 07FF                       2954          BR       15                 RETURN TO CALLER
001576 9200 0003      00003       2955 IND3     MVI   HPEZ,X'00'            PUT EZ AND HP = OFF
00157A 07FF                       2956          BR       15                 RETURN TO CALLER
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                    ASM H V 05 22.10 05/07/81

                                   2958 ***********************************************************************
                                   2959 *                                                                     *
                                   2960 *                     'ARCHK' AND 'EXCHK' SUBROUTINES                 *
                                   2961 *                                                                     *
                                   2962 *  THIS SUBROUTINE IS USED TO PROCESS ARITHMETIC OVERFLOW AND UNDER-  *
                                   2963 *  FLOW CONDITIONS.                                                   *
                                   2964 *                                                                     *
                                   2965 *  ENTRIES                                                            *
                                   2966 *  'ARCHK' FOR ARITHMETIC CHECKS CAUSED BY ARITHMETIC OVERFLOW        *
                                   2967 *  'EXCHK' FOR EXPONENT CHECKS CAUSED BY EXPONENT OVERFLOW/UNDERFLOW  *
                                   2968 *                                                                     *
                                   2969 *  OPERATION                                                          *
                                   2970 *  THE ARITHMETIC CHECK INDICATOR IS TURNED ON. THEN, THIS INDICATOR  *
                                   2971 *  IS DISPLAYED ON THE 1052 PRINTER-KEYBOARD IF SO REQUIRED BY CONT-  *
                                   2972 *  ROL INFORMATION. THE SIMULATED OVERFLOW SWITCH IS INTERROGATED -   *
                                   2973 *  IF SET TO 'PROGRAM',THE SUBROUTINE RETURNS TO THE CALLING SEQUEN-  *
                                   2974 *  CE. IF SET TO 'STOP', A MESSAGE STATING THAT THERE IS AN  'ARITH-  *
                                   2975 *  METIC CHECK' AND INDICATING THE ADDRESS OF THE INSTRUCTION INVOLV- *
                                   2976 *  ED, IS SENT TO THE OPERATOR. SIMULATION IS SUSPENDED UNTIL OPERA-  *
                                   2977 *  TOR ACTION  IS TAKEN.                                              *
                                   2978 *                                                                     *
                                   2979 *  NOTE = SAME OPERATION FOR 'EXCHK'.                                 *
                                   2980 *                                                                     *
                                   2981 ***********************************************************************

00157C 9610 0004      00004        2983 ARCHK    OI    OFLOW,X'10'              SET ARITHM. CHECK INDIC. = ON
001580 41E0 0193            00193  2984          LA    WR5,AR                   PREPARE MESSAGE
001584 9101 0001      00001        2985 OFLSWI   TM    PROSWI,X'01'             TEST IF O'FLOW SWITCH = 'STOP'
001588 471F 0000            00000  2986          BO       0(15)                 NO, RETURN TO CALLER
00158C 40E0 3594            01594  2987          STH   WR5,*+8                  *
001590 45F0 00F4            000F4  2988          BAL   15,MESTOP                DISPLAY INDICATOR
001594 00000000                    2989          DC    A(0)                   *
001598 9601 0004      00004        2990 EXCHK    OI    OFLOW,X'01'              SET EXPONENT CHECK INDICATOR ON
00159C 41E0 0188            00188  2991          LA    WR5,EXP                  PREPARE MESSAGE
0015A0 47F0 3584            01584  2992          B       OFLSWI                 BRANCH
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  2994          AIF (NOT &DIVIDE).NODIV1
                                  2995 ********************************************************************************
                                  2996 *                                                                              *
                                  2997 *                          'FIXDIV' SUBROUTINE                                 *
                                  2998 *                                                                              *
                                  2999 *  THIS SUBROUTINE IS USED BY FIXED-POINT AND FLOATING-POINT  DIVIDE           *
                                  3000 *  OPERATIONS.                                                                 *
                                  3001 *                                                                              *
                                  3002 *  OPERATION                                                                   *
                                  3003 *  THE P AND Q ADDRESSES AND THE P FIELD SIGN ARE  SAVED.  REGISTERS           *
                                  3004 *  WR1 AND WR2 USED AS POINTERS ARE INITIALIZED. THEN,  THE  'ADDPQ'           *
                                  3005 *  SUBROUTINE IS ENTERED TO SUBTRACT THE Q FIELD FROM  THE  DIVIDEND           *
                                  3006 *  BY ADDING THE NINES COMPLEMENT OF EACH DIVISOR TO  EACH  DIVIDEND           *
                                  3007 *  (SEE ALSO FIXED-POINT SUBTRACTION). IF NECESSARY, A NEW  CARRY IS           *
                                  3008 *  SET.                                                                        *
                                  3009 *                                                                              *
                                  3010 *  WHEN A Q FLAG OTHER THAN THE SIGN FLAG IS DETECTED, AN EXTRA ZERO           *
                                  3011 *  DIGIT IS SUBTRACTED FROM THE DIVIDEND BEFORE ADDING THE P   DIGIT           *
                                  3012 *  OF WR1 TO THE Q DIGIT.                                                      *
                                  3013 *                                                                              *
                                  3014 *  A TEST IS PERFORMED ON THE LAST CARRY. A CARRY ONE INDICATES THAT           *
                                  3015 *  THE SUBTRACTION OF THE Q FIELD HAS NOT EXHAUSTED THE DIVIDEND.THE           *
                                  3016 *  LAST CARRY IS ADDED TO THE UNITS POSITION  OF THE   RESULTING   P           *
                                  3017 *  FIELD. THE ZERO-SET QUOTIENT DIGIT IS INCREMENTED BY ONE,   WHICH           *
                                  3018 *  PROVES THAT ONE SUBTRACTION HAS BEEN SUCCESSFULLY PERFORMED.  THE           *
                                  3019 *  QUOTIENT DIGIT IS CHECKED AGAINST 9, SINCE MORE THAN NINE    SUB-           *
                                  3020 *  TRACTIONS RESULT IN AN OVERFLOW. IN CASE OF OVERFLOW, THE 'FIXDIV'          *
                                  3021 *  SUBROUTINE RETURNS TO THE CALLING SEQUENCE THROUGH A SPECIAL OVER-          *
                                  3022 *  FLOW EXIT.                                                                  *
                                  3023 *                                                                              *
                                  3024 *  IF THE QUOTIENT DIGIT IS LOWER THAN, OR EQUAL TO 9, CONTROL IS GI-          *
                                  3025 *  VEN BACK TO THE START OF A NEW SUBTRACTION. THIS PROCESS  IS  RE-           *
                                  3026 *  PEATED UNTIL THE LAST CARRY IS ZERO. IN CASE OF A ZERO CARRY, THE           *
                                  3027 *  PARTIAL REMAINDER IS TESTED.  IF  NULL,  THE   PROGRAM RETURNS TO           *
                                  3028 *  'CARRY ONE'.  IF NOT  NULL,  THE   ZERO CARRY INDICATES AN EXCESS           *
                                  3029 *  SUBTRACTION AND THE RE-ADDITION LOOP 'ADDPQ' IS ENTERED IN  ORDER           *
                                  3030 *  TO ADD ONE DIGIT OF THE Q FIELD AFTER THE OTHER TO THE   PREVIOUS           *
                                  3031 *  FIELD.                                                                      *
                                  3032 *                                                                              *
                                  3033 *  FINALLY, THE QUOTIENT DIGIT IS NORMALIZED AND STORED. THE  P  AD-           *
                                  3034 *  DRESS IS THEN COMPARED TO THE 1620 ADDRESS 99 AND, IF LOWER,   IT           *
                                  3035 *  IS INCREMENTED BY 1. THE WHOLE  PROCESS  OF SUCCESSIVE  SUBTRACT-           *
                                  3036 *  IONS + RE-ADD IS STARTED ALL OVER AGAIN, UNTIL THE P ADDRESS   IS           *
                                  3037 *  EQUAL TO 99. THE LENGTH OF THE QUOTIENT IS COMPUTED AND THE  QUO-           *
                                  3038 *  TIENT FLAG IS SET. A FLAG IS SET ON THE REMAINDER, WHICH THEN RE-           *
                                  3039 *  CEIVES THE P SIGN.                                                          *
                                  3040 *                                                                              *
                                  3041 ********************************************************************************
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
0015A4 9201 0003      00003        3043 FIXDIV   MVI   HPEZ,X'01'
0015A8 9089 09E8            09E8   3044          STM   RP,RQ,INITP          SAVE RP AND RQ
0015AC D300 33A4 6063 013A4 00063  3045          MVZ   APSIGN(1),99(MAPORG) SAVE P SIGN
0015B2 D300 33A5 9000 013A5 00000  3046          MVZ   AQSIGN(1),0(RQ)      SAVE Q SIGN
0015B8 96F0 9000      00000        3047          OI    0(RQ),X'F0'          DELETE Q SIGN
0015BC 9200 32ED      012ED        3048 FIXD13   MVI   CVQ+1,X'00'          SET SWITCH FOR COMPLEMENTATION
0015C0 18C0                        3049 FIXDID   LR    WR3,0                WR3=0
0015C2 18D0                        3050          LR    WR4,0                WR4=0
0015C4 1820                        3051          LR    R2,0                 R2=0
0015C6 18A8                        3052 FIXDIA   LR    WR1,RP               WR1=RP
0015C8 1889                        3053          LR    WR2,RQ               WR2=RQ
0015CA 18E0                        3054          LR    WR5,0                WR5=0 (CARRY)
0015CC 4510 32E8            012E8  3055 FIXDA    BAL   R1,ADDPQ             SUBTRACT Q DIGIT FROM P DIGIT
0015D0 96F0 A000      00000        3056          OI    0(WR1),X'F0'         CLEAR POSSIBLE FLAG
0015D4 9120 B000      00000        3057          TM    0(WR2),X'20'         IS Q FIELD FLAGGED
0015D8 06A0                        3058          BCTR  WR1,0                WR1=WR1-1
0015DA 50A0 0580            00580  3059          ST    WR1,FDIC+16          SAVE RIGHT ADDRESS OF P FIELD+1
0015DE 4780 35E6            015E6  3060          BZ      FIXD1              YES,BRANCH
0015E2 46B0 35CC            015CC  3061          BCT   WR2,FIXDA            WR2=WR2-1,LOOP
0015E6 43D0 3334            01334  3062 FIXD1    IC    WR4,ATAB3            SET Q DIGIT =09
0015EA 4510 32F4            012F4  3063          BAL   R1,ADDPQX            SUBTRACT Q DIGIT FROM P DIGIT
0015EE 96F0 A000      00000        3064          OI    0(WR1),X'F0'         CLEAR POSSIBLE FLAG
0015F2 46E0 363A            0163A  3065 FIXDIE   BCT   WR5,FIXD2            TEST CARRY,ZERO BRANCH
0015F6 18B8                        3066          LR    WR2,RP               ONE,WR2=RP
0015F8 9109 B000      00000        3067 FIXD5    TM    0(WR2),X'09'         IS 0(WR2) NUMREIC = 09
0015FC 47C0 3610            01610  3068          BC    BZM,FIXD4            NO,BRANCH
001600 94F0 B000      00000        3069          NI    0(WR2),X'F0'         0(WR2)=0
001604 59B0 0580            00580  3070          C     WR2,FDIC+16          TEST IF P FIELD IS EXHAUSTED
001608 4780 3610            01610  3071          BC    8,FIXD4              YES,BRANCH
00160C 46B0 35F8            015F8  3072          BCT   WR2,FIXD5            WR2=WR2-1,LOOP
001610 41EE 0001            00001  3073 FIXD4    LA    WR5,1(WR5)           RESET CARRY
001614 43CB 0000            00000  3074          IC    WR3,0(WR2)           INSERT P DIGIT
001618 1ACE                        3075          AR    WR3,WR5              ADD CARRY
00161A 42CB 0000            00000  3076          STC   WR3,0(WR2)           STORE RESULT
00161E 4122 0001            00001  3077          LA    R2,1(R2)             R2=R2+1
001622 4110 0009            00009  3078          LA    R1,9                 *
001626 1921                        3079          CR    R2,R1                COMPARE R2 WITH 9
001628 4720 3630            01630  3080          BH      FIXD6              *
00162C 47F0 35C6            015C6  3081          B     FIXDIA               LOOP
001630 D300 9000 33A5 00000 013A5  3082 FIXD6    MVZ   0(1,RQ),APSIGN+1     RESTORE Q SIGN
001636 47FF 0004            00004  3083          B       4(15)              RETURN TO CALLER
00163A 18A8                        3084 FIXD2    LR    WR1,RP               WR1 = RP
00163C 18B9                        3085          LR    WR2,RQ               WR2 = RQ
00163E 95F9 A000      00000        3086 FIXD19   CLI   0(WR1),X'F9'          COMPARE P DIGIT WITH 9
001642 4770 3654            01654  3087          BC    7,FIXD18             BRANCH IF NO EQUAL
001646 9120 B000      00000        3088          TM    0(WR2),X'20'         IS Q DIGIT FLAGGED
00164A 4780 35F6            015F6  3089          BC    8,FIXDIE+4           YES,BRANCH
00164E 06A0                        3090          BCTR  WR1,0                WR1 = WR1-1
001650 46B0 363E            0163E  3091          BCT   WR2,FIXD19           WR2 = WR2-1,LOOP
001654 41EE 0001            00001  3092 FIXD18   LA    WR5,1(WR5)           RESET CARRY
001658 18A8                        3093          LR    WR1,RP               WR1 = RP
00165A 18B9                        3094          LR    WR2,RQ               RW2=RQ
00165C 92F0 32ED      012ED        3095          MVI   CVQ+1,X'F0'          SET SWITCH OFF
001660 4510 32E8            012E8  3096 FIXD9    BAL   R1,ADDPQ             ADD Q TO P DIGIT
001664 9120 B000      00000        3097          TM    0(WR2),X'20'         IS Q FIELD FLAGGED
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

001668 4780 3672            01672  3098          BZ      FIXD8               YES,BRANCH
00166C 06A0                        3099          BCTR  WR1,0                 WR1=WR1-1
00166E 46B0 3660            01660  3100          BCT   WR2,FIXD9            WR2=WR2-1,LOOP
001672 43D0 3374            01374  3101 FIXD8    IC    WR4,ATAB2             WR4=00(Q DIGIT)
001676 50A0 0580            00580  3102          ST    WR1,FDIC+16           SAVE RIGHT ADDRESS OF P FIELD
00167A 06A0                        3103          BCTR  WR1,0                 WR1=WR1-1
00167C 4510 32F4            012F4  3104          BAL   R1,ADDPQX             ADD Q TO P DIGIT
001680 422A 0000            00000  3105          STC   R2,0(WR1)             STORE R2 ATWR1 ADDRESS
001684 96F0 A000      00000        3106          OI    0(WR1),X'F0'          *
001688 18C8                        3107          LR    WR3,RP                WR3=RP
00168A 9109 C000      00000        3108 FIXD11   TM    0(WR3),X'09'          IS 0(WR3)=9
00168E 47C0 36A2            016A2  3109          BC    BZM,FIXD10            NO,BRANCH
001692 94F0 C000      00000        3110          NI    0(WR3),X'F0'          0(WR3) NUMREIC =0
001696 59C0 0580            00580  3111          C     WR3,FDIC+16           TEST IF P FIELD IS EXHAUSTED
00169A 4780 36AC            016AC  3112          BC    8,FIXD10+10           YES,BRANCH
00169E 46C0 368A            0168A  3113          BCT   WR3,FIXD11            NO,LOOP
0016A2 43DC 0000            00000  3114 FIXD10   IC    WR4,0(WR3)            *
0016A6 1ADE                        3115          AR    WR4,WR5               ADD CARRY
0016A8 42DC 0000            00000  3116          STC   WR4,0(WR3)            *
0016AC 4116 0063            00063  3117          LA    R1,99(MAPORG)         *
0016B0 1981                        3118          CR    RP,R1                 COMPARE RP WITH COREST+99
0016B2 4780 36BE            016BE  3119          BE      FIXD12              EQUAL,BRANCH
0016B6 4186 0001            00001  3120          LA    RP,1(RP)              RP = RP+1
0016BA 47F0 35BC            015BC  3121          B       FIXD13              *
0016BE 58D0 09E8            009E8  3122 FIXD12   L     WR4,INITP             RESTORE P ADDRESS
0016C2 1B8A                        3123          SR    RP,WR1                RP=RP-WR1=L
0016C4 1BD8                        3124          SR    WR4,RP                WR4=WR4-L
0016C6 94DF D000      00000        3125          NI    0(WR4),X'DF'          SET FLAG AT WR4-L ADDR.
0016CA 94DF A001      00001        3126          NI    1(WR1),X'DF'          SET REMAINDER FLAG
0016CE 5890 09EC            009EC  3127          L     RQ,INITQ              RESTORE Q ADDRESS
0016D2 D300 9000 33A5 00000 013A5  3128          MVZ   0(1,RQ),APSIGN+1      SET Q SIGN
0016D8 9120 33A4      013A4        3129          TM    APSIGN,X'20'          IS P FIELD POSITIVE
0016DC 4710 36F2            016F2  3130          BO      FIXD15              YES,BRANCH
0016E0 9720 6063      00063        3131          XI    99(MAPORG),X'20'      NO,REMAINDER NEGATIVE
0016E4 9120 33A5      013A5        3132          TM    APSIGN+1,X'20'        IS Q FIELD POSITIVE
0016E8 4710 36FA            016FA  3133          BO      FIXD17              YES,BRANCH
0016EC 96F0 A000      00000        3134 FIXD16   OI    0(WR1),X'F0'          SET PLUS SIGN
0016F0 07FF                        3135          BR      15                  RETURN TO CALLER
0016F2 9120 33A5      013A5        3136 FIXD15   TM    APSIGN+1,X'20'        IS Q FIELD POSITIVE
0016F6 4710 36EC            016EC  3137          BO      FIXD16              YES,BRANCH
0016FA 94DF A000      00000        3138 FIXD17   NI    0(WR1),X'DF'          SET MINUS SIGN
0016FE 07FF                        3139          BR      15                  RETURN TO CALLER
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                ASM H V 05 22.10 05/07/81
                                  3141 ********************************************************************************
                                  3142 *                                                                              *
                                  3143 *                                DIVIDE                                        *
                                  3144 *                                                                              *
                                  3145 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY BEFORE THE 'FIXDIV'           *
                                  3146 *  SUBROUTINE IS ENTERED. THE NORMAL SEQUENCE PROCEEDS TO THE 'INDIC'          *
                                  3147 *  SUBROUTINE FROM WHERE EXIT IS TO 'BIR' ENTRY0. HOWEVER, IF A  DI-           *
                                  3148 *  VIDE OVERFLOW OCCURS, THE 'ARCHK' SUBROUTINE IS  ENTERED   FOR  A           *
                                  3149 *  CHECK. THE SEQUENCE THEN EXITS TO 'BIR' ENTRY0.                             *
                                  3150 *                                                                              *
                                  3151 ********************************************************************************

001700 45F0 0902          00902  3153 D29      BAL   15,CONVPQ            CONVERT P AND Q ADDRESSES
001704 45F0 35A4          015A4  3154 D29A     BAL   15,FIXDIV            BRANCH TO FIXDIV
001708 47F0 3710          01710  3155          B       D29B               BRANCH TO NORMAL SEQUENCE
00170C 47F0 0E26          00E26  3156          B        C24I              ARITHMETIC CHECK TESTING
001710 188A                      3157 D29B     LR    RP,WR1               RP = WR1
001712 45F0 3548          01548  3158          BAL   15,INDIC             INDICATORS UPDATING
001716 47F0 07F6          007F6  3159          B       ENTRY0             GO TO INTERPRETIVE LOOP


                                  3162 ********************************************************************************
                                  3163 *                                                                              *
                                  3164 *                            DIVIDE IMMEDIATE                                  *
                                  3165 *                                                                              *
                                  3166 ********************************************************************************

00171A 45F0 0906          00906  3168 DM19     BAL   15,CONVP             CONVERT P ADDRESS
00171E 4195 000B          0000B  3169          LA    RQ,11(CNTR)          RQ = CNTR+11
001722 47F0 3704          01704  3170          B       D29A               GO TO DIVIDE
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                          ASM H V 05 22.10 05/07/81

                                   3172 ***********************************************************************
                                   3173 *                                                                     *
                                   3174 *                          LOAD DIVIDEND                              *
                                   3175 *                                                                     *
                                   3176 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY.A PRELIMINARY 'MOVE  *
                                   3177 *  CHARACTER' IS INITIATED TO FILL THE PRODUCT AREA WITH 'F0' BYTES.  *
                                   3178 *  THEN, THE Q FIELD SIGN IS TRANSMITTED TO POSITION 99. EXIT IS  TO  *
                                   3179 *  THE 'TRANSMIT FIELD' ROUTINE DESCRIBED UNDER 'INTERNAL DATA TRANS- *
                                   3180 *  MISSION'.                                                          *
                                   3181 ***********************************************************************

001726 45F0 0902             00902 3183 LD28     BAL   15,CONVPQ              CONVERT PANDQ ADDRESSES
00172A 92F0 6050       00050       3184 LD28A    MVI   80(MAPORG),X'F0'       CLEAR PRODUCT AREA
00172E D212 6051 6050 00051 00050  3185          MVC   81(19,MAPORG),80(MAPORG)
001734 96F0 8000       00000       3186          OI    0(RP),X'F0'            CLEAR FLAG IF ANY           V1L1
                                   3187 *        MVZ   99(1,MAPORG),0(RQ)     MOVE Q SIGN       (DELETED) V1L2
001738 D100 8000 9000 00000 00000  3188          MVN   0(1,RP),0(RQ)          MOVE NUMERIC PART OF Q      V1L2
00173E 5090 09E8             009E8 3189          ST    RQ,INITP               SAVE RQ FOR FURTHER USE     V1L2
001742 45F0 0CB8             00CB8 3190 LD28B    BAL   15,TNSWR+2             MOVE FIELD RQ TO RP         V1L2
001746 D200 8000 9000 00000 00000  3191          MVC   0(1,RP),0(RQ)          *                           V1L2
00174C 9120 9000       00000       3192          TM    0(RQ),X'20'            IS RQ FLAGGED               V1L2
001750 4710 3742             01742 3193          BC    1,LD28B                NO,LOOP                     V1L2
001754 5890 09E8             009E8 3194          L     RQ,INITP               YES,RESTORE RQ              V1L2
001758 D300 6063 9000 00063 00000  3195          MVZ   99(1,MAPORG),0(RQ)     MOVE Q SIGN TO 99           V1L2
00175E 47F0 07F6             007F6 3196          B       ENTRY1               RETURN TO 'BIR'             V1L2
                                   3197 *        B      TFA                   GO TO TRANSMIT FIE(DELETED) V1L2


                                   3200 ***********************************************************************
                                   3201 *                                                                     *
                                   3202 *                    LOAD DIVIDEND IMMEDIATE                          *
                                   3203 *                                                                     *
                                   3204 ***********************************************************************

001762 45F0 0906             00906 3206 LDM18    BAL   15,CONVP               CONVERT P ADDRESS
001766 4195 000B             0000B 3207          LA    RQ,11(CNTR)            RQ = CNTR + 11
00176A 47F0 372A             0172A 3208          B      LD28A                 *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3210          AGO   .DIVX1
                                  3211 .DIVX1   ANOP
                                  3212          AIF   (NOT &FLOAT).NOFLO1
                                  3213 ********************************************************************************
                                  3214 *                                                                              *
                                  3215 *                                SECTION 5                                     *
                                  3216 *                          FLOATING-POINT OPERATIONS                           *
                                  3217 *                                                                              *
                                  3218 *  SIMULATION OF FLOATING-POINT OPERATIONS USES THE FIVE SUBROUTINES            *
                                  3219 *  ALREADY DESCRIBED UNDER 'ARITHMETIC OPERATIONS', AND THE 'EXPOW'            *
                                  3220 *  SUBROUTINE REFERRED TO IN THE PRESENT SECTION.                              *
                                  3221 *  THE FLOATING-ADD AND FLOATING-SUBTRACT INSTRUCTIONS REQUIRE,   IN           *
                                  3222 *  ADDITION, THE 'SHIFT' SUBROUTINE ALSO DESCRIBED IN THIS SECTION.            *
                                  3223 *                                                                              *
                                  3224 ********************************************************************************



                                  3226 ********************************************************************************
                                  3227 *                                                                              *
                                  3228 *                             FLOATING MULTIPLY                                *
                                  3229 *                                                                              *
                                  3230 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY,  THEN  DECREMENTED           *
                                  3231 *  BY 2.  CONTROL IS GIVEN TO THE 'MULT' SUBROUTINE, FOLLOWED BY THE           *
                                  3232 *  'INDIC' SUBROUTINE.  AFTER INDICATOR UPDATING,  THE P AND Q EXPO-           *
                                  3233 *  NENTS ARE CONVERTED TO BINARY AND ADDED.  THE  LEFTMOST DIGIT  IS           *
                                  3234 *  TESTED AND, IF ZERO, THE DIGITS IN THE PRODUCT AREA ARE MOVED  TO           *
                                  3235 *  THE P FIELD MANTISSA.                                                       *
                                  3236 *                                                                              *
                                  3237 *  THE E/Z INDICATOR IS TESTED =                                               *
                                  3238 *                                                                              *
                                  3239 *  IF ON, THE RESULT OF THE MULTIPLICATION IS ZERO AND THE  P  EXPO-           *
                                  3240 *  NENT IS SET TO 99.                                                          *
                                  3241 *  IF OFF, CONTROL IS GIVEN TO THE 'EXPOW' SUBROUTINE WHICH NORMALLY           *
                                  3242 *  EXITS TO 'BIR'. IN CASE OF OVERFLOW, CONTROL IS DIRECTLY GIVEN TO           *
                                  3243 *  THE 'EXCHK' SUBROUTINE WHICH INTERROGATES THE OVERFLOW SWITCH. IN           *
                                  3244 *  CASE OF UNDERFLOW, THE E/Z INDICATOR IS SET ON.                             *
                                  3245 *                                                                              *
                                  3246 ********************************************************************************

00176E 45F0 0902           00902  3248 FMUL03   BAL   15,CONVPQ            CONVERT P AND Q ADDRESSES
001772 0680                       3249          BCTR  RP,0                 *
001774 0690                       3250          BCTR  RQ,0                 *
001776 0680                       3251          BCTR  RP,0                 RP=RP-2
001778 0690                       3252          BCTR  RQ,0                 RQ= RQ-2
00177A 45F0 33C4           013C4  3253          BAL   15,MULT              MULTIPLY P FIELD BY Q FIELD
00177E 58C0 0570           00570  3254          L     WR3,FDIC             WR3= ADDRESS OF FIRST P DIGIT
001782 4186 0063           00063  3255          LA    RP,99(MAPORG)        RP =COREST +99
001786 18A8                       3256          LR    WR1,RP               WR1 = COREST + 99
001788 45F0 3548           01548  3257          BAL   15,INDIC             INDICATORS UPDATING
00178C 5880 09E8           009E8  3258          L     RP,INITP             RESTORE P ADDRESS
001790 F271 09D8 8001 009D8 00001  3259          PACK  BUFFP(8),1(2,RP)     CONVERT EP TO BINARY
001796 4FD0 09D8           009D8  3260          CVB   WR4,BUFFP            *
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
00179A F271 09D8 9001 009D8 00001  3261          PACK  BUFFP(8),1(2,RQ)       CONVERT EQ TO  BINARY
0017A0 4FE0 09D8           009D8  3262          CVB   WR5,BUFFP               *
0017A4 1ADE                       3263          AR    WR4,WR5                 WR4=EP+EQ
0017A6 910F B000      00000       3264          TM    0(WR2),X'0F'            IS 0(WR2) = 0
0017AA 4770 37B8           017B8  3265          BC    7,FMUL1                 NO,BRANCH
0017AE 41BB 0001           00001  3266          LA    WR2,1(WR2)              YES,WR2 =WR2+1
0017B2 06D0                       3267          BCTR  WR4,0                   WR4=WR4-1
0017B4 94DF B000      00000       3268          NI    0(WR2),X'DF'            SET FLAG AT WR2ADDRESS
0017B8 D100 C000 B000 00000 00000 3269 FMUL1    MVN   0(1,WR3),0(WR2)         MOVE NUMERIC TO 0(WR3)
0017BE 41BB 0001           00001  3270 FMUL2    LA    WR2,1(WR2)              WR2=WR2+1
0017C2 41CC 0001           00001  3271          LA    WR3,1(WR3)              WR3=WR3+1
0017C6 D100 C000 B000 00000 00000 3272          MVN   0(1,WR3),0(WR2)         MOVE NUMERIC
0017CC 19C8                       3273          CR    WR3,RP                  IS WR3 LOWER THAN RP
0017CE 4740 37BE           017BE  3274          BL       FMUL2                YES,BRANCH
0017D2 D300 8000 6063 00000 00063 3275          MVZ   0(1,RP),99(MAPORG)      MOVE ZONE TO P FIELD
0017D8 9501 0003      00003       3276          CLI   HPEZ,X'01'              IS  E/Z ON
0017DC 4780 37F8           017F8  3277          BE       FMUL3                YES,BRANCH
0017E0 45F0 3B9A           01B9A  3278          BAL   15,EXPOW                NO,EXPOW SUBROUTINE
0017E4 47F0 07F6           007F6  3279          B       ENTRY0                RETURN TO INTER. LOOP
0017E8 47F0 37F0           017F0  3280          B       FMUL4                 O'FL
0017EC 9201 0003      00003       3281          MVI   HPEZ,X'01'              U'FL,SET E/Z ON
0017F0 45F0 3598           01598  3282 FMUL4    BAL   15,EXCHK                EXCHK  SUBROUTINE
0017F4 47F0 07F6           007F6  3283          B       ENTRY0                RETURN TO INTER. LOOP
0017F8 92D9 8001      00001       3284 FMUL3    MVI   1(RP),X'D9'             *
0017FC 92D9 8002      00002       3285          MVI   2(RP),X'D9'             SET EP= 'D9D9'
001800 47F0 07F6           007F6  3286          B       ENTRY0                RETURN TO INTERPRETIVE LOOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3288 ************************************************************************
                                  3289 *                                                                      *
                                  3290 *                          FLOATING DIVIDE                             *
                                  3291 *                                                                      *
                                  3292 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY,  THEN  DECREMENTED   *
                                  3293 *  BY 2.  CONTROL IS GIVEN TO THE 'COMP' SUBROUTINE.  THE   ABSOLUTE   *
                                  3294 *  VALUES OF THE TWO MANTISSAS ARE COMPARED AND ACCORDING TO THE RE-    *
                                  3295 *  SULT OF THE TEST  THE P MANTISSA IS TRANSFERRED TO THE PRODUCT A-    *
                                  3296 *  REA.                                                                 *
                                  3297 *  THEN,   CONTROL IS GIVEN TO THE 'FIXDIV' SUBROUTINE  WITH  NORMAL   *
                                  3298 *  EXIT TO THE 'INDIC' SUBROUTINE.                                      *
                                  3299 *  THE QUOTIENT IS MOVED TO  THE P MANTISSA FIELD AND THE  RESULTING    *
                                  3300 *  EXPONENT IS COMPUTED (EP - EQ).                                      *
                                  3301 *  THE 'EXPOW' SUBROUTINE IS ENTERED AND IN CASE  OF EXPONENT  OVER-    *
                                  3302 *  FLOW OR UNDERFLOW, THE SEQUENCE GOES TO THE 'EXCHK' 'ARCHK'  SUB-    *
                                  3303 *  ROUTINE INSTEAD OF RETURNING TO 'BIR'. THE P AND Q EXPONENTS  ARE    *
                                  3304 *  THEN COMPUTED.                                                       *
                                  3305 *                                                                       *
                                  3306 ************************************************************************

001804 45F0 0902           00902  3308 FDIV09   BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
001808 0690                       3309          BCTR  RQ,0                   *
00180A 0690                       3310          BCTR  RQ,0                   RQ= RQ-2
00180C 0680                       3311          BCTR  RP,0                   *
00180E 0680                       3312          BCTR  RP,0                   RP=RP-2
001810 9089 0570           00570  3313          STM   RP,RQ,FDIC             SAVE P AND Q ADDRESSES
001814 92F0 6050      00050        3314          MVI   80(MAPORG),X'F0'       CLEAR PRODUCT AREA
001818 D212 6051 6050 00051 00050  3315          MVC   81(19,MAPORG),80(MAPORG)
00181E 4120 3374           01374  3316          LA    R2,ATAB2               LOAD A 'F0' ADDRESS
001822 5020 09E8           009E8  3317          ST    R2,INITP               STORE THIS ADDRESS
001826 45F0 0E8C           00E8C  3318          BAL   15,COMP                COMP SUBROUTINE
00182A 9111 0003      00003        3319          TM    HPEZ,X'11'             IS HP AND EZ OFF
00182E 4780 38E4           018E4  3320          BZ      FDIA                 YES,BRANCH
001832 41B6 0064           00064  3321          LA    WR2,100(MAPORG)        NO,WR2=COREST+100
001836 D201 3AA2 3524 01AA2 01524  3322          MVC   EPH+2(2),MTAB1+32      EPH+2=0001
00183C 1BBA                       3323 FDIB     SR    WR2,WR1                WR2=WR3-WR1
00183E 9859 0570           00570  3324          LM    RP,RQ,FDIC             RP,RQ=INITIAL ADDRESSES
001842 50A0 0578           00578  3325          ST    WR1,FDIC+8             SAVE WR1
001846 D100 B000 8000 00000 00000  3326          MVN   0(1,WR2),0(RP)         MOVE NUMERIC TO 0(WR2)
00184C D300 6063 8000 00063 00000  3327          MVZ   99(1,MAPORG),0(RP)     MOVE ZONE TO COREST+99
001852 0680                       3328 FDIE     BCTR  RP,0                   RP=RP-1
001854 06B0                       3329          BCTR  WR2,0                  WR2=WR2-1
001856 D200 B000 8000 00000 00000  3330          MVC   0(1,WR2),0(RP)         MOVE BYTE TO 0(WR2)
00185C 9120 8000      00000        3331          TM    0(RP),X'20'            IS P FIELD FLAGGED
001860 4710 3852           01852  3332          BO      FDIE                 NO,LOOP
001864 4186 0064           00064  3333          LA    RP,100(MAPORG)         YES,RP=COREST+100
001868 1B8A                       3334          SR    RP,WR1                 RP =RP-WR1
00186A 45F0 35A4           015A4  3335          BAL   15,FIXDIV              FIXDIV SUBROUTINE
00186E 47F0 3876           01876  3336          B       *+8                  *
001872 47F0 38F2           018F2  3337          B       FIXOFL               OVERFLOW
001876 188A                       3338          LR    RP,WR1                 RP =WR1
001878 45F0 3548           01548  3339          BAL   15,INDIC               INDICATORS UPDATING
00187C 988A 0570           00570  3340          LM    RP,WR1,FDIC            RELOAD RP,RQ,WR1
001880 41D6 0063           00063  3341          LA    WR4,99(MAPORG)         WR4 =COREST+99
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
001884 1BDA                          3342          SR    WR4,WR1           WR4=WR4-WR1
001886 D200 8000 D000 00000 00000    3343          MVC   0(1,RP),0(WR4)    MOVE FIRST BYTE TO 0(RP)
00188C 0680                          3344 FDIV6    BCTR  RP,0              *
00188E 06D0                          3345          BCTR  WR4,0             WR4 =WR4-1
001890 D200 8000 D000 00000 00000    3346          MVC   0(1,RP),0(WR4)    MOVE ONE BYTE TO 0(RP)
001896 9120 D000       00000         3347          TM    0(WR4),X'20'      IS 0(WR4) FLAGGED
00189A 4710 388C             0188C   3348          BO      FDIV6           NO, LOOP
00189E 5880 0570             00570   3349          L     RP,FDIC           YES,RP=INITIAL ADDRESS
0018A2 9501 0003       00003         3350          CLI   HPEZ,X'01'        IS E/Z ON
0018A6 4780 38D8             018D8   3351          BE      FDIV7           YES,BRANCH
0018AA F271 09D8 8001 009D8 00001    3352          PACK  BUFFP(8),1(2,RP)  NO,CONVERT EP TO BINARY
0018B0 4FD0 09D8                     009D8 3353    CVB   WR4,BUFFP         *
0018B4 F271 09D8 9001 009D8 00001    3354          PACK  BUFFP(8),1(2,RQ)  CONVERT EQ TO BINARY
0018BA 4FB0 09D8                     009D8 3355    CVB   WR2,BUFFP         *
0018BE 1BDB                          3356          SR    WR4,WR2           WR4=WR4-WR2
0018C0 4AD0 3AA2             01AA2   3357          AH    WR4,EPH+2         WR4=WR4+EPH
0018C4 45F0 3B9A             01B9A   3358          BAL   15,EXPOW           EXPOW SUBROUTINE
0018C8 47F0 07F6             007F6   3359          B       ENTRY0          RETURN TO INTERPRETIVE ROUTINE
0018CC 47F0 38D0             018D0   3360          B       *+4             OFL
0018D0 45F0 3598             01598   3361          BAL   15,EXCHK          UFL,EXCHK SUBROUTINE
0018D4 47F0 07F6             007F6   3362          B       ENTRY0          RETURN TO INTERPRETIVE ROUTINE
0018D8 92D9 8001       00001         3363 FDIV7    MVI   1(RP),X'D9'       *
0018DC 92D9 8002       00002         3364          MVI   2(RP),X'D9'       EP='D9D9'
0018E0 47F0 07F6             007F6   3365          B       ENTRY0          RETURN TO INTERPRETIVE LOOP
0018E4 41B6 0063             00063   3366 FDIA     LA    WR2,99(MAPORG)    WR2=COREST + 99
0018E8 D201 3AA2 34D8 01AA2 014D8    3367          MVC   EPH+2(2),MCLEAR   EPH+2=0000
0018EE 47F0 383C             0183C   3368          B       FDIB            BRANCH
0018F2 9889 0570             00570   3369 FIXOFL   LM    RP,RQ,FDIC        RELOAD RP,RQ
0018F6 F271 09D8 8001 009D8 00001    3370          PACK  BUFFP(8),1(2,RP)  CONVERT EP TO BINARY
0018FC 4FC0 09D8                     009D8 3371    CVB   WR3,BUFFP         *
001900 5890 09EC             009EC   3372          L     RQ,INITQ          CONVERT EQ TO BINARY
001904 F271 09D8 9001 009D8 00001    3373          PACK  BUFFP(8),1(2,RQ)  *
00190A 4FD0 09D8                     009D8 3374    CVB   WR4,BUFFP         *
00190E 1BCD                          3375          SR    WR3,WR4           WR3=WR3-WR4
001910 4EC0 09D8                     009D8 3376    CVD   WR3,BUFFP         CONVERT WR3 TO DECIMAL
001914 9101 09DF       009DF         3377          TM    BUFFP+7,X'01'     TEST SIGN
001918 4710 3920             01920   3378          BC    1,*+8             IF D,BRANCH
00191C 960F 09DF       009DF         3379          OI    BUFFP+7,X'0F'     IF C,CONVERT TO F
001920 F311 8001 09DE 00001 009DE    3380          UNPK  1(2,RP),BUFFP+6(2) UNPACK IT IN P FIELD
001926 94DF 8001       00001         3381          NI    1(RP),X'DF'       SET FLAG ON EXPONENT
00192A 45F0 3598             01598   3382          BAL   15,EXCHK          EXCHK SUBROUTINE
00192E 47F0 0E26             00E26   3383          B       C24I            ARITHMETIC CHECK
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  3386 ************************************************************************
                                  3387 *                                                                      *
                                  3388 *                          TRANSMIT FLOATING                           *
                                  3389 *                                                                      *
                                  3390 *  THE P AND Q ADDRESSES ARE CONVERTED TO BINARY,  AND THREE  DIGITS   *
                                  3391 *  ARE MOVED FROM THE Q FIELD TO THE P FIELD.                          *
                                  3392 *  CONTROL IS THEN GIVEN TO TFA IN THE 'TRANSMIT FIELD' ROUTINE.       *
                                  3393 *                                                                      *
                                  3394 ************************************************************************

001932 45F0 0902           00902  3396 TFL06    BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
001936 0680                       3397          BCTR  RP,0                  *
001938 0680                       3398          BCTR  RP,0                  RP=RP-2
00193A 0690                       3399          BCTR  RQ,0                  *
00193C 0690                       3400          BCTR  RQ,0                  RQ=RQ-2
00193E D202 8000 9000 00000 00000 3401          MVC   0(3,RP),0(RQ)         MOVE 3 CHARACTERS FROM Q TO P
001944 47F0 0CA0           00CA0  3402          B       TFA                 GO TO TRANSMIT FIELD


                                  3404 ************************************************************************
                                  3405 *                                                                      *
                                  3406 *                        BRANCH AND TRANSMIT                           *
                                  3407 *                                                                      *
                                  3408 ************************************************************************

001948 45F0 0902           00902  3410 BTFL07   BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
00194C 41A5 000C           0000C  3411          LA    WR1,12(CNTR)          WR1=CNTR+12
001950 50A0 3134           01134  3412          ST    WR1,IR2               STORE WR1 IN IR-2
001954 1858                       3413          LR    CNTR,RP               CNTR =RP
001956 0680                       3414          BCTR  RP,0                  *
001958 0680                       3415          BCTR  RP,0                  *
00195A 0680                       3416          BCTR  RP,0                  RP=RP-3
00195C 0690                       3417          BCTR  RQ,0                  *
00195E 0690                       3418          BCTR  RQ,0                  RQ=RQ-2
001960 D202 8000 9000 00000 00000 3419          MVC   0(3,RP),0(RQ)         MOVE 3 CHARACTERS FROM Q TO P
001966 47F0 30E2           010E2  3420          B       BT27B               GO TO BRANCH AND TRANSMIT
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3422 ************************************************************************
                                  3423 *                                                                      *
                                  3424 *                          FLOATING SHIFT RIGHT                        *
                                  3425 *                                                                      *
                                  3426 ************************************************************************

00196A 45F0 0902           00902  3428 FSR08    BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
00196E 45E0 3976           01976  3429          BAL   14,*+8                 *
001972 47F0 07F6           007F6  3430          B       ENTRY1               *
001976 D100 8000 9000 00000 00000 3431          MVN   0(1,RP),0(RQ)        MOVE NUMERIC FROM Q TO P
00197C 0680                       3432 FSRA     BCTR  RP,0                 RP=RP-1
00197E 0690                       3433          BCTR  RQ,0                 RQ=RQ-1
001980 D200 8000 9000 00000 00000 3434 FSRC     MVC   0(1,RP),0(RQ)        MOVE ONE BYTE FROM Q TO P
001986 9120 9000      00000       3435          TM    0(RQ),X'20'          IS Q FIELD FLAGGED
00198A 4710 397C           0197C  3436          BO      FSRA               NO,LOOP
00198E 1989                       3437 FSRB     CR    RP,RQ                YES, IS RP EQUAL TO RQ
001990 078E                       3438          BER     14                 RETURN
001992 0680                       3439          BCTR  RP,0                 RP=RP-1
001994 92F0 8000      00000       3440          MVI   0(RP),X'F0'          MOVE 'F0' TO P FIELD
001998 47F0 398E           0198E  3441          B       FSRB               LOOP


                                  3444 ************************************************************************
                                  3445 *                                                                      *
                                  3446 *                          FLOATING SHIFT LEFT                         *
                                  3447 *                                                                      *
                                  3448 ************************************************************************

00199C 45F0 0902           00902  3450 FSL05    BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
0019A0 18A9                       3451          LR    WR1,RQ               WR1=RQ
0019A2 06A0                       3452 FSLA     BCTR  WR1,0                WR1=WR1-1
0019A4 9120 A000      00000       3453          TM    0(WR1),X'20'         IS WR1 FIELD FLAGGED
0019A8 4710 39A2           019A2  3454          BO      FSLA               NO,LOOP
0019AC D200 8000 A000 00000 00000 3455 FSLB     MVC   0(1,RP),0(WR1)       YES,MOVE ONE BYTE FROM WR1 TO P
0019B2 19A9                       3456          CR    WR1,RQ               IS WR1=RP
0019B4 4780 39C4           019C4  3457          BE      FSLC               YES,BRANCH
0019B8 41AA 0001           00001  3458          LA    WR1,1(WR1)           NO,WR1=WR1+1
0019BC 4188 0001           00001  3459          LA    RP,1(RP)             RP=RP+1
0019C0 47F0 39AC           019AC  3460          B       FSLB               BRANCH
0019C4 1989                       3461 FSLC     CR    RP,RQ                TEST IF RP=RQ
0019C6 4780 07F6           007F6  3462          BC    8,ENTRY1             YES,RETURN TO INTER.LOOP
0019CA 96F0 8000      00000       3463          OI    0(RP),X'F0'          NO,CLEAR POSSIBLE FLAG
0019CE 4188 0001           00001  3464 FSLE     LA    RP,1(RP)             RP=RP+1
0019D2 1989                       3465          CR    RP,RQ                TEST IF RP=RQ
0019D4 4780 39E0           019E0  3466          BE      FSLD               YES,BRANCH
0019D8 92F0 8000      00000       3467          MVI   0(RP),X'F0'          NO,MOVE 'F0' TO P ADDRESS
0019DC 47F0 39CE           019CE  3468          B       FSLE               BRANCH
0019E0 94F0 9000      00000       3469 FSLD     NI    0(RQ),X'F0'          MOVE FLAG FROM Q TO P
0019E4 47F0 07F6           007F6  3470          B       ENTRY1             RETURN TO INTER. LOOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3472 ********************************************************************************
                                  3473 *                                                                              *
                                  3474 *                          'SHIFT' SUBROUTINE                                  *
                                  3475 *                                                                              *
                                  3476 *  THIS SUBROUTINE IS REQUIRED FOR SHIFT OPERATIONS IN  FLOATING-ADD           *
                                  3477 *  AND FLOATING-SUBTRACT INSTRUCTIONS,  WHEN THE P AND  Q  EXPONENTS           *
                                  3478 *  ARE DIFFERENT.                                                              *
                                  3479 *                                                                              *
                                  3480 ********************************************************************************

0019E8 0690                       3482 SHIFT    BCTR  RQ,0                RQ=RQ-2
0019EA 0690                       3483          BCTR  RQ,0                RQ=RQ-2
0019EC D300 33A5 9000 013A5 00000 3484          MVZ   AQSIGN(1),0(RQ)     SAVE Q SIGN
0019F2 0680                       3485 SHIFTS   BCTR  RP,0                RP=RP-2
0019F4 0680                       3486          BCTR  RP,0                RP=RP-2
0019F6 4120 31F8          011F8   3487          LA    R2,FIXADD           LOAD FIXADD ADDRESS
0019FA D300 33A4 8000 013A4 00000 3488          MVZ   APSIGN(1),0(RP)     SAVE P SIGN
001A00 18A8                       3489          LR    WR1,RP              WR1=RP
001A02 F271 09D8 8001 009D8 00001 3490          PACK  BUFFP(8),1(2,RP)    CONVERT EP TO BINARY
001A08 4FB0 09D8          009D8   3491          CVB   WR2,BUFFP           *
001A0C F271 09D8 9001 009D8 00001 3492          PACK  BUFFP(8),1(2,RQ)    CONVERT EQ TO BINARY
001A12 4FC0 09D8          009D8   3493          CVB   WR3,BUFFP           *
001A16 90BC 3AA0          01AA0   3494          STM   WR2,WR3,EPH         STORE EP AND EQ
001A1A 1BBC                       3495          SR    WR2,WR3             WR2=WR2-WR3=EP-EQ
001A1C 9089 09E8          009E8   3496          STM   RP,RQ,INITP         SAVE RP AND RQ
001A20 18C8                       3497          LR    WR3,RP              WR3=RP
001A22 41D0 0001          00001   3498          LA    WR4,1               WR4=1=L
001A26 06C0                       3499 SHIFTA   BCTR  WR3,0               WR3=WR3-1
001A28 41DD 0001          00001   3500          LA    WR4,1(WR4)          WR4=WR4+1
001A2C 9120 C000      00000       3501          TM    0(WR3),X'20'        IS 0(WR3) FLAGGED
001A30 4710 3A26          01A26   3502          BO      SHIFTA            NO,LOOP
001A34 12BB                       3503          LTR   WR2,WR2             YES,TEST WR2
001A36 078F                       3504          BER     15                ZERO,RETURN
001A38 4720 3A8A          01A8A   3505          BH      SHIFTB            HIGH,BRANCH
001A3C D201 8001 9001 00001 00001 3506          MVC   1(2,RP),1(RQ)       LOWER,MOVE EQ TO EP
001A42 D203 3AA0 3AA4 01AA0 01AA4 3507          MVC   EPH(4),EQH          MOVE EQ TO EP
001A48 10BB                       3508          LPR   WR2,WR2             LOAD POSITIVE D
001A4A 19BD                       3509          CR    WR2,WR4             IS WR2 LOWER THAN L (WR4 = L)
001A4C 4740 3A72          01A72   3510          BL      SHIFTC            YES,BRANCH
001A50 D200 8000 9000 00000 00000 3511          MVC   0(1,RP),0(RQ)       NO,MOVE Q DIGIT TO P DIGIT
001A56 D300 8000 33A5 00000 013A5 3512          MVZ   0(1,RP),AQSIGN      SET P SIGN
001A5C 0680                       3513 SHIFTE   BCTR  RP,0                RP=RP-1
001A5E 0690                       3514          BCTR  RQ,0                RQ=RQ-1
001A60 D200 8000 9000 00000 00000 3515          MVC   0(1,RP),0(RQ)       MOVE Q DIGIT TO P DIGIT
001A66 9120 9000      00000       3516          TM    0(RQ),X'20'         IS Q FIELD FLAGGED
001A6A 4710 3A5C          01A5C   3517          BO      SHIFTE            NO,LOOP
001A6E 47F0 3B62          01B62   3518          B     SHIFTF              YES,GO TO NORMALIZATION
001A72 1898                       3519 SHIFTC   LR    RQ,RP               RQ=RP
001A74 1B9B                       3520          SR    RQ,WR2              RQ=RP-D
001A76 45E0 3980          01980   3521          BAL   14,FSRC             GO TO SHIFT RIGHT
001A7A 94D0 9000      00000       3522          NI    0(RQ),X'D0'         SET FLAG
001A7E 1A9B                       3523          AR    RQ,WR2              *
001A80 96F0 9000      00000       3524          OI    0(RQ),X'F0'         CLEAR FLAG
001A84 9889 09E8          009E8   3525          LM    RP,RQ,INITP         RESTORE RP,RQ ADDRESSES
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT
001A88 07FF                       3526          BR      15              RETURN
001A8A 19BD                       3527 SHIFTB   CR    WR2,WR4           IS WR2 LOWER THAN WR4
001A8C 47A0 3B62            01B62 3528          BC    10,SHIFTF         NO,GO TO NORMALIZATION
001A90 1B9B                       3529          SR    RQ,WR2            YES, RQ=RQ-WR2
001A92 4122 0004            00004 3530          LA    R2,4(R2)          R2=R2+4
001A96 96F0 321F      0121F       3531          OI    ADD5+1,X'F0'      SET SWITCH ON
001A9A 96F0 B000      00000       3532          OI    0(RP),X'F0'       DELETE P SIGN
001A9E 07FF                       3533          BR      15              RETURN


001AA0                            3535          DS    0D                *
001AA0                            3536 EPH      DS    F                 P EXPONENT
001AA4                            3537 EQH      DS    F                 Q EXPONENT
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  3539 ********************************************************************************
                                  3540 *                                                                              *
                                  3541 *                              FLOATING ADD                                    *
                                  3542 *                                                                              *
                                  3543 *  THE P AND Q ADDRESSES ARE CONVERTED  TO  BINARY BEFORE   ENTERING          *
                                  3544 *  THE 'SHIFT' SUBROUTINE. TABLES ARE THEN PREPARED FOR THE 'FIXADD'          *
                                  3545 *  SUBROUTINE.  ITS SPECIAL EXIT GIVES CONTROL TO NORMALIZATION  AND          *
                                  3546 *  ITS NORMAL EXIT TESTS THE LAST CARRY. IF ONE, A SHIFT OF ONE  PO-          *
                                  3547 *  SITION TO THE RIGHT OF THE RESULTING FIELD IS PERFORMED,  AND  A           *
                                  3548 *  FLAGGED ONE IS MOVED TO ITS LEFTMOST DIGIT.  IF ZERO,  CONTROL IS          *
                                  3549 *  GIVEN TO NORMALIZATION. HOWEVER,  BOTH EXITS LEAD TO THE  'EXPOW'          *
                                  3550 *  SUBROUTINE WHICH SETS THE INDICATORS TO THE PROPER VALUE  ACCORD-          *
                                  3551 *  ING TO THE RESULTING EXPONENT AND MANTISSA VALUES.                         *
                                  3552 *  IN THE NORMALIZATION SEQUENCE, THE RESULTING FIELD IS SCANNED FOR          *
                                  3553 *  HIGH-ORDER ZEROS.  THE MANTISSA IS SHIFTED  AND  THE EXPONENT  IS          *
                                  3554 *  MODIFIED.  IF EXPONENT MODIFICATION IS NOT REQUIRED,  CONTROL  IS          *
                                  3555 *  GIVEN TO THE 'INDIC' SUBROUTINE WHICH UPDATES THE H/P AND E/Z IN-          *
                                  3556 *  DICATORS AND THEN RETURNS TO THE CALLING SEQUENCE.                         *
                                  3557 *                                                                              *
                                  3558 ********************************************************************************

001AA8 45F0 0902           00902  3560 FADD01   BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
001AAC 45F0 39E6           019E8  3561          BAL   15,SHIFT              SHIFT SUBROUTINE
001AB0 18A8                       3562 FADD5    LR    WR1,RP                WR1=RP
001AB2 18B9                       3563          LR    WR2,RQ                WR2=RQ
001AB4 05F2                       3564          BALR  15,R2                 FIXADD SUBROUTINE
001AB6 47F0 3B0E           01B0E  3565          B       FADDA               SPECIAL EXIT,GO TO NORMALIZAT.
001ABA 12EE                       3566 FADD2    LTR   WR5,WR5               NORMAL EXIT,TEST LAST CARRY
001ABC 4780 3B0E           01B0E  3567          BE      FADDA               IF ZERO,GO TO NORMALIZATION
001AC0 58B0 09E8           009E8  3568          L     WR2,INITP             IF ONE, WR2=RP
001AC4 06B0                       3569          BCTR  WR2,0                 WR2=RP-1
001AC6 D100 B001 B000 00001 00000 3570 FADD1    MVN   1(1,WR2),0(WR2)       MOVE NUMERIC FROM 0(WR2)
001ACC 9120 B000      00000       3571          TM    0(WR2),X'20'          IS 0(WR2) FLAGGED
001AD0 4780 3AD8           01AD8  3572          BC    8,FADDC               YES BRANCH
001AD4 46B0 3AC6           01AC6  3573          BCT   WR2,FADD1             NO,LOOP
001AD8 92D1 B000      00000       3574 FADDC    MVI   0(WR2),X'D1'          MOVE 'D1' TO WR2
001ADC 18C0                       3575          LR    WR3,0                 *
001ADE 06C0                       3576          BCTR  WR3,0                 WR3=-1
001AE0 5860 09E8           009E8  3577 FADDB    L     RP,INITP              RESTORE P ADDRESS
001AE4 58D0 3AA0           01AA0  3578          L     WR4,EPH               *
001AE8 1BDC                       3579          SR    WR4,WR3               WR4=EP-WR3
001AEA 45F0 3B9A           01B9A  3580          BAL   15,EXPOW              EXPOW SUBROUTINE
001AEE 47F0 3B62           01B62  3581          B       SHIFTF              RETURN TO 'SHIFT' SUBROUTINE
001AF2 47F0 3B02           01B02  3582          B      OFL                  O'FLOW
001AF6 9201 0003      00003       3583 FADDX    MVI   HPEZ,X'01'            U'FLOW
001AFA 45F0 3598           01598  3584 FADD9    BAL   15,EXCHK              EXCHK SUBROUTINE
001AFE 47F0 07F6           007F6  3585          B      ENTRY0               RETURN TO INTERPRETIVE LOOP
001B02 58A0 09E8           009E8  3586 OFL      L     WR1,INITP             WR1=P ADDRESS
001B06 45F0 3548           01548  3587          BAL   15,INDIC              INDIC SUBROUTINE
001B0A 47F0 3AFA           01AFA  3588          B      FADD9                BRANCH

                                  3590 *              NORMALIZATION
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
001B0E 5860 09E8            009E8  3592 FADDA    L     RP,INITP            RESTORE P ADDRESS
001B12 18BA                        3593          LR    WR2,WR1             WR2=WR1
001B14 18C0                        3594          LR    WR3,0               WR3=0
001B16 910F B000      00000        3595 FADDU    TM    0(WR2),X'0F'        IS 0(WR2) NUMERIC =0
001B1A 4780 3B50            01B50  3596          BZ      FADD3             YES,BRANCH
001B1E 12CC                        3597          LTR   WR3,WR3             NO, IS WR3=0
001B20 4780 3B62            01B62  3598          BE      SHIFTF            YES,BRANCH
001B24 18DA                        3599          LR    WR4,WR1             NO,WR4=WR1
001B26 D100 D000 B000 00000 00000  3600 FADD6    MVN   0(1,WR4),0(WR2)     MOVE NUMERIC FROM WR2 TO WR4
001B2C 19B8                        3601          CR    WR2,RP              IS WR2= P ADDRESS
001B2E 4780 3B3E            01B3E  3602          BE      FADD4             YES,BRANCH
001B32 41BB 0001            00001  3603          LA    WR2,1(WR2)          NO,WR2=WR2+1
001B36 41DD 0001            00001  3604          LA    WR4,1(WR4)          WR4=WR4+1
001B3A 47F0 3B26            01B26  3605          B     FADD6               LOOP
001B3E 41DD 0001            00001  3606 FADD4    LA    WR4,1(WR4)          WR4=WR4+1
001B42 94F0 D000      00000        3607          NI    0(WR4),X'F0'        NI WITH 'F0' AT WR4 ADDRESS
001B46 19D8                        3608          CR    WR4,RP              IS WR4=RP
001B48 4780 3AE0            01AE0  3609          BE      FADDB             YES,BRANCH
001B4C 47F0 3B3E            01B3E  3610          B     FADD4               NO,LOOP
001B50 19B8                        3611 FADD3    CR    WR2,RP              IS WR2=RP
001B52 4780 3B62            01B62  3612          BE      SHIFTF            YES BRANCH
001B56 41CC 0001            00001  3613          LA    WR3,1(WR3)          NO,INCREMENT WR3 BY 1
001B5A 41BB 0001            00001  3614          LA    WR2,1(WR2)          INCREMENT WR2 BY 1
001B5E 47F0 3B16            01B16  3615          B     FADDU               LOOP



001B62 58A0 09E8            009E8  3617 SHIFTF   L     WR1,INITP           WR1= P ADDRESS
001B66 188A                        3618          LR    RP,WR1              RP=P ADDRESS
001B68 45F0 3548            01548  3619          BAL   15,INDIC            INDIC SUBROUTINE
001B6C 9501 0003      00003        3620          CLI   HPEZ,X'01'          IS E/Z ON
001B70 4770 07F6            007F6  3621          BNE     ENTRY0            NO,RETURN TO INTER. LOOP
001B74 92D9 8001      00001        3622          MVI   1(RP),X'D9'         YES,SET P EXPONENT = -99
001B78 92D9 8002      00002        3623          MVI   2(RP),X'D9'         *
001B7C 47F0 07F6            007F6  3624          B     ENTRY0              RETURN TO INTER. LOOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3626 ************************************************************************
                                  3627 *                                                                      *
                                  3628 *                          FLOATING SUBTRACT                           *
                                  3629 *                                                                      *
                                  3630 *  THE P AND Q ADDRESSES ARE CONVERTED  TO  BINARY BEFORE   ENTERING  *
                                  3631 *  THE 'SHIFT' SUBROUTINE.  THE Q SIGN  IS INVERTED  AND  CONTROL IS  *
                                  3632 *  THEN GIVEN TO THE FLOATING-ADD ROUTINE (FADD5).                     *
                                  3633 *                                                                      *
                                  3634 ************************************************************************

001B80 45F0 0902           00902  3636 FSUB02   BAL   15,CONVPQ              CONVERT P AND Q ADDRESSES
001B84 0690                       3637          BCTR  RQ,0                   RQ=RQ-2
001B86 0690                       3638          BCTR  RQ,0                   *
001B88 D300 33A5 9000 013A5 00000 3639          MVZ   AQSIGN(1),0(RQ)        SAVE Q SIGN
001B8E 9720 33A5      013A5       3640          XI    AQSIGN,X'20'           INVERT Q SIGN
001B92 45F0 39F2           019F2  3641          BAL   15,SHIFTS              SHIFT SUBROUTINE
001B96 47F0 3AB0           01AB0  3642          B       FADD5                *

                                  3645 ************************************************************************
                                  3646 *                                                                      *
                                  3647 *                          'EXPOW' SUBROUTINE                          *
                                  3648 *                                                                      *
                                  3649 *  THIS SUBROUTINE IS REQUIRED IN FLOATING-POINT OPERATIONS TO CHECK   *
                                  3650 *  THE VALUE OF THE RESULTING EXPONENT (OVERFLOW).                     *
                                  3651 *                                                                      *
                                  3652 *  INPUT       WR4 CONTAINS THE EXPONENT RESULTING FROM THE PREVIOUS   *
                                  3653 *                  OPERATION                                           *
                                  3654 *                                                                      *
                                  3655 ************************************************************************

001B9A 41A0 0063           00063  3657 EXPOW    LA    WR1,99                 WR1=99
001B9E 19DA                       3658          CR    WR4,WR1                IS WR4 GREATER THAN 99
001BA0 4720 3BC8           01BC8  3659          BH      EXP1AB               YES,BRANCH
001BA4 11AA                       3660          LNR   WR1,WR1                NO,SET WR1=-99
001BA6 19DA                       3661          CR    WR4,WR1                IS WR4 LESS THAN -99
001BA8 4740 3BF0           01BF0  3662          BL      EXP2AB               YES,BRANCH
001BAC 4ED0 3AA0           01AA0  3663          CVD   WR4,EPH                NO,CONVERT WR4 TO DECIMAL
001BB0 9101 3AA7      01AA7       3664          TM    EQH+3,X'01'            TEST SIGN
001BB4 4710 3BBC           01BBC  3665          BC    1,*+8                  IF D,BRANCH
001BB8 960F 3AA7      01AA7       3666          OI    EQH+3,X'0F'            IF C,CONVERT TO F
001BBC F311 8001 3AA6 00001 01AA6 3667          UNPK  1(2,RP),EQH+2(2)       UNPACK EP IN P FIELD
001BC2 94DF 8001      00001       3668          NI    1(RP),X'DF'            MOVE FLAG
001BC6 07FF                       3669          BR      15                   RETURN
001BC8 92D9 8001      00001       3670 EXP1AB   MVI   1(RP),X'D9'            *
001BCC 92F9 8002      00002       3671          MVI   2(RP),X'F9'            EP='D9F9'
001BD0 18A8                       3672          LR    WR1,RP                 WR1=RP
001BD2 94F0 A000      00000       3673          NI    0(WR1),X'F0'           *
001BD6 9609 A000      00000       3674          OI    0(WR1),X'09'           FILL MANTISSA WITH NINES
001BDA 06A0                       3675 EXP3     BCTR  WR1,0                  WR1=WR1-1
001BDC 94F0 A000      00000       3676          NI    0(WR1),X'F0'           *
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

001BE0 9609 A000      00000        3677           OI    0(WR1),X'09'          FILL MANTISSA WITH ZEROS
001BE4 9120 A000      00000        3678           TM    0(WR1),X'20'          IS 0(WR1) FLAGGED
001BE8 4710 3BDA            01BDA  3679           BO       EXP3               NO,LOOP
001BEC 47FF 0004            00004  3680           B       4(15)              YES,RETURN TO 4 (15)
001BF0 92D9 8001      00001        3681 EXP2AB    MVI   1(RP),X'D9'           *
001BF4 92D9 8002      00002        3682           MVI   2(RP),X'D9'           EP= 'D9D9'
001BF8 18A8                        3683           LR    WR1,RP                WR1=RP
001BFA 94F0 A000      00000        3684           NI    0(WR1),X'F0'          0(WR1)='F0'
001BFE 06A0                        3685 EXP4      BCTR  WR1,0                 WR1=WR1 -1
001C00 94F0 A000      00000        3686           NI    0(WR1),X'F0'          SET'F0' OR 'D0' IN MANTISSA
001C04 9120 A000      00000        3687           TM    0(WR1),X'20'          IS 0(WR1) FLAGGED
001C08 4710 3BFE            01BFE  3688           BO       EXP4               NO,LOOP
001C0C 47FF 0008            00008  3689           B       8(15)              YES,RETURN  TO  8(15)
                                   3690           AGO   .FLOX1
                                   3691 .FLOX1    ANOP
                                   3692           AIF   (NOT &INDEX).NOTIX3
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3694 ********************************************************************************
                                  3695 *                                                                              *
                                  3696 *                                SECTION 6                                     *
                                  3697 *                        INDEX REGISTER OPERATIONS                             *
                                  3698 *                                                                              *
                                  3699 *  AFTER MODIFICATION OF THE INDEX REGISTERS, THE NEW INDEX REGISTER            *
                                  3700 *  FIELD CONTAINS THE ALGEBRAIC SUM OF THE CONTENTS OF THE OLD FIELD           *
                                  3701 *  AND THE MODIFIER.                                                           *
                                  3702 *                                                                              *
                                  3703 ********************************************************************************



                                  3705 ********************************************************************************
                                  3706 *                                                                              *
                                  3707 *                        BRANCH AND TRANSMIT ADDRESS                           *
                                  3708 *                                                                              *
                                  3709 ********************************************************************************

001C10 45F0 0902                00902  3711 BTA20    BAL   15,CONVPQ           CONVERT P AND Q ADDRESSES
001C14 41A5 000C                0000C  3712 BTA20A   LA    WR1,12(CNTR)        WR1 = CNTR +12
001C18 50A0 3134                01134  3713          ST    WR1,IR2             STORE WR1 IN IR2
001C1C 1858                           3714          LR    CNTR,RP             CNTR= P ADDRESS
001C1E 41B0 0004                00004  3715          LA    WR2,4
                                      3716 *        SR    RQ,WR2              RQ=RQ-4           (DELETED) V1L1
                                      3717 *        LA    WR2,1(WR2)          *                 (DELETED) V1L1
                                      3718 *        SR    RP,WR2              RP=RP-5           (DELETED) V1L1
                                      3719 *        MVN   0(5,RP),0(RQ)       MOVE 5 NUMERICS   (DELETED) V1L1
001C22 9200 3C3D        01C3D         3720          MVI   BTA20D+1,X'00'      SET SWITCH OFF                V1L1
001C26 92BC 3C49        01C49         3721          MVI   BTA20E+1,X'BC'      SET BCTR ON                   V1L1
001C2A 41C0 3C2E                01C2E  3722          LA    WR3,BTA20B          ADDRESS FOR LOOP              V1L1
001C2E 0680                           3723 BTA20B  BCTR  RP,0                RP=RP-1                       V1L1
001C30 1986                           3724          CR    RP,MAPORG           TEST FOR WRAP AROUND          V1L1
001C32 4740 3C5A                01C5A  3725          BL      BTA20F            *                             V1L1
001C36 D200 8000 9000 00000 00000     3726 BTA20C  MVC   0(1,RP),0(RQ)        MOVE 1 CHARACTER              V1L1
001C3C 4700 3C6E                01C6E  3727 BTA20D  BC    0,BTA20J            IF ON, END OF TRANSMIT        V1L1
001C40 0690                           3728          BCTR  RQ,0                RQ=RQ-1                       V1L1
001C42 1996                           3729          CR    RQ,MAPORG           TEST FOR WRAP AROUND          V1L1
001C44 4740 3C60                01C60  3730          BL      BTA20G            *                             V1L1
001C48 06BC                           3731 BTA20E  BCTR  WR2,WR3              IF WR2=0 NO BRANCH            V1L1
001C4A 9400 3C49        01C49         3732          NI    BTA20E+1,X'00'      SET BCTR OFF                  V1L1
001C4E 9120 9000        00000         3733          TM    0(RQ),X'20'         TEST IF FLAG                  V1L1
001C52 4780 3C66                01C66  3734          BC    8,BTA20H            YES BRANCH                    V1L1
001C56 47F0 3C2E                01C2E  3735          B       BTA20B            NO LOOP                       V1L1
001C5A 1887                           3736 BTA20F  LR    RP,SIZE                                           V1L1
001C5C 47F0 3C36                01C36  3737          B       BTA20C                                          V1L1
001C60 1897                           3738 BTA20G  LR    RQ,SIZE                                           V1L1
001C62 47F0 3C48                01C48  3739          B       BTA20E                                          V1L1
001C66 96F0 3C3D        01C3D         3740 BTA20H  OI    BTA20D+1,X'F0'      SET SWITCH ON                 V1L1
001C6A 47F0 3C2E                01C2E  3741          B       BTA20B            GO TO MOVE LAST DIGIT         V1L1
001C6E 1B00                           3742 BTA20J  SR    0,0                                               V1L1
001C70 92F0 310D        0110D         3743          MVI   SWBB42+1,X'F0'      SET SWITCH BTM,BT,BTFL ON
001C74 47F0 0F44                00F44  3744          B       TESTBR            GO TO TEST BRANCH ADDRESS
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM H V 05 22.10 05/07/81

```
                                3747 ***************************************************************************
                                3748 *                                                                         *
                                3749 *                BRANCH AND TRANSMIT ADDRESS IMMEDIATE                    *
                                3750 *                                                                         *
                                3751 ***************************************************************************

001C78 4195 000B          0000B 3753 BTAM10   LA    RQ,11(CNTR)           RQ=CNTR+11
001C7C 45F0 0906          00906 3754          BAL   15,CONVP              CONVERT P ADDRESS
001C80 47F0 3C14          01C14 3755          B       BTA20A              BRANCH TO BTA-20
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3757 ********************************************************************************
                                  3758 *                                                                              *
                                  3759 *                              MOVE ADDRESS                                    *
                                  3760 *                                                                              *
                                  3761 ********************************************************************************

001C84 45F0 0902            00902 3763 MA70     BAL   15,CONVPQ             CONVERT P AND Q ADDRESSES
001C88 41B0 0004            00004 3764          LA    WR2,4                  *
001C8C 1B8B                       3765          SR    RP,WR2                 RP = RP-4
001C8E 1B9B                       3766          SR    RQ,WR2                 RQ = RQ-4
001C90 D104 8000 9000 00000 00000 3767          MVN   0(5,RP),0(RQ)          MOVE 5 NUMERICS FROM Q TO P
001C96 47F0 07F6            007F6 3768          B       ENTRY1               RETURN TO INTERPRETIVE LOOP
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3770 ************************************************************************
                                  3771 *                                                                      *
                                  3772 *                          BRANCH AND SELECT                           *
                                  3773 *                                                                      *
                                  3774 *  THE P ADDRESS IS CONVERTED TO BINARY AND DIGIT Q11  (CNTR+11)  IS   *
                                  3775 *  TESTED.  ACCORDING TO THE TEST RESULT, THE 'IXBAND' INDICATOR  IS   *
                                  3776 *  SET TO 'NO BAND', 'BAND 1 SELECTED', OR 'BAND 2 SELECTED'(SEE THE   *
                                  3777 *  'INDEX' SUBROUTINE), AND THE INDIRECT ADDRESSING SWITCH IS SET ON   *
                                  3778 *  OR OFF (SEE THE 'INDAD' SUBROUTINE).  THE INSTRUCTION COUNTER  IS   *
                                  3779 *  THEN LOADED WITH THE P ADDRESS, BEFORE CONTROL IS GIVEN TO 'BIR'.   *
                                  3780 *                                                                      *
                                  3781 *  NOTE = THE 'NO BAND' MODE IS SELECTED AT SIMULATOR INITIALIZATION   *
                                  3782 *                                                                      *
                                  3783 ************************************************************************

001C9A 45F0 0906           00906  3785 BS60     BAL   15,CONVP              CONVERT P ADDRESS
001C9E 4195 0008           0000B  3786          LA    RQ,11(CNTR)           RQ=DIGIT Q11
001CA2 95F0 9000      00000        3787          CLI   0(RQ),X'F0'           Q11=0
001CA6 4770 3CB2           01CB2  3788          BC    7,BS1                 NO,BRANCH
001CAA 9204 0006      00006        3789          MVI   IXBAND,X'04'          YES,SET INDICATOR 30 ON
001CAE 47F0 3CEE           01CEE  3790          B       BSE                 RETURN
001CB2 95F1 9000      00000        3791 BS1      CLI   0(RQ),X'F1'           Q11=1
001CB6 4770 3CC2           01CC2  3792          BC    7,BS2                 NO,BRANCH
001CBA 9202 0006      00006        3793          MVI   IXBAND,X'02'          YES,SET INDICATOR 31 ON
001CBE 47F0 3CEE           01CEE  3794          B       BSE                 RETURN
001CC2 95F2 9000      00000        3795 BS2      CLI   0(RQ),X'F2'           Q11=2
001CC6 4770 3CD2           01CD2  3796          BC    7,BS3                 NO,BRANCH
001CCA 9201 0006      00006        3797          MVI   IXBAND,X'01'          YES,SET INDICATOR 32 ON
001CCE 47F0 3CEE           01CEE  3798          B       BSE                 RETURN
001CD2 95F8 9000      00000        3799 BS3      CLI   0(RQ),X'F8'           Q11=8
001CD6 4770 3CE2           01CE2  3800          BC    7,BS4                 NO,BRANCH
001CDA 92FE 093D      0093D        3801          MVI   INDAD+1,X'FE'         YES,SET IND.ADD.SWITCH OFF
001CDE 47F0 3CEE           01CEE  3802          B       BSE                 RETURN
001CE2 95F9 9000      00000        3803 BS4      CLI   0(RQ),X'F9'           Q11=9
001CE6 4770 3CF4           01CF4  3804          BC    7,ERXOP               NO,INVALID OPERATION
001CEA 9200 093D      0093D        3805          MVI   INDAD+1,X'00'         YES,SET IND.ADD.SWITCH ON
001CEE 1858                     3806 BSE      LR    CNTR,RP               CNTR = P ADDRESS
001CF0 47F0 0F44           00F44  3807          B       TESTBR              GO TO TEST BRANCH ADDRESS
001CF4 45F0 00F4           000F4  3808 ERXOP    BAL   15,MESTOP             *
001CF8 000001B1                   3809          DC  A(INVXOP)             ERROR ON INDEX OPERATION
```

```
3811 ********************************************************************************
3812 *                                                                              *
3813 *                              BRANCH AND MODIFY                               *
3814 *                                                                              *
3815 *  THE CONDITION SWITCH IS SET OFF.                                            *
3816 *  THE Q ADDRESS IS FIRST CONVERTED TO  BINARY  WITHOUT  CONSIDERING           *
3817 *  ANY FLAGS, AND THEN THE P ADDRESS, WHICH MAY BE INDEXED  AND   AN           *
3818 *  INDIRECT ADDRESS,  IS CONVERTED.                                            *
3819 *  CONTROL IS THEN GIVEN TO THE 'INDEX' SUBROUTINE  WHICH   COMPUTES           *
3820 *  THE SYSTEM/360 EFFECTIVE ADDRESS OF THE INDEX REGISTER  SPECIFIED           *
3821 *  BY THE Q FLAGS. THIS ADDRESS IS LOADED INTO REGISTER WR2.                   *
3822 *  THE FIELD DESIGNATED BY THE Q ADDRESS  (RQ) IS  ADDED TO  THE SE-           *
3823 *  LECTED REGISTER BY LOADING RP WITH THE INDEX REGISTER  ADDRESS IN           *
3824 *  WR2. CONTROL IS GIVEN TO THE 'FIXADD' SUBROUTINE AND   INDICATORS          *
3825 *  HP/EZ ARE UPDATED IN THE 'INDIC' SUBROUTINE.                                *
3826 *  THE CONDITION SWITCH IS TESTED =                                            *
3827 *  IF ON, THE SELECTED INDEX REGISTER IS TESTED FOR THE THREE CONDI-           *
3828 *  TIONS 'SIGN CHANGE', 'SIGN VALUE NULL', 'LAST CARRY'. IF ONE   OF           *
3829 *  THESE THREE CONDITIONS IS PRESENT,  LOADING OF THE P ADDRESS INTO           *
3830 *  'CNTR' WILL BE  BYPASSED.  IF NONE IS PRESENT,  THE P ADDRESS  IS           *
3831 *  LOADED.                                                                     *
3832 *  IF OFF,THE THREE PRECEDING TESTS ARE NOT PERFORMED, AND THE COUN-           *
3833 *  TER IS DIRECTLY LOADED.                                                     *
3834 *  FINALLY, THE Q FIELD IS TESTED.  IF NOT  EXHAUSTED,  THE OVERFLOW           *
3835 *  INDICATOR IS SET ON BY THE 'ARCHK' SUBROUTINE.                              *
3836 *                                                                              *
3837 *  NOTE =                                                                      *
3838 *  THE INSTRUCTIONS BX-61,  BXM-62,  BCX-63,  BCXM-64 AND BLX-65 ARE           *
3839 *  BASICALLY THE  SAME ROUTINES.  THEREFORE,  ONLY BX-61 IS DESCRIB-           *
3840 *  ED IN DETAIL.                                                               *
3841 *                                                                              *
3842 ********************************************************************************

001CFC 92F0 3D63      01D63          3844 BX61     MVI   BXA+1,X'F0'          SET CONDITION SWITCH OFF
001D00 9200 3D45      01D45          3845          MVI   BXG+1,0              SET BXG SWITCH OFF
001D04 D200 34D0 0006 014D0 00006    3846 BXE      MVC   REG15(1),IXBAND      SAVE IXBAND INDICATOR
001D0A 9204 0006      00006          3847          MVI   IXBAND,X'04'         SET NO BAND INDICATOR ON
001D0E 9200 09C7      009C7          3848          MVI   INDX6+1,X'00'        SWITCH OFF, NO INDAD FOR Q  V1L2
001D12 45F0 091E            0091E    3849          BAL   15,CONVQ             CONVERT Q ADDRESS
001D16 92F0 09C7      009C7          3850          MVI   INDX8+1,X'F0'        RESTORE SWITCH              V1L2
001D1A D200 0006 34D0 00006 014D0    3851          MVC   IXBAND(1),REG15      RESTORE IXBAND INDICATOR
001D20 45F0 0906            00906    3852 BXD      BAL   15,CONVP            CONVERT P ADDRESS
001D24 41A5 0007            00007    3853          LA    WR1,7(CNTR)          WR1=CNTR+7
001D28 50C0 34D0            014D0    3854          ST    12,REG15             SAVE R12
001D2C 92FE 09A1      009A1          3855          MVI   INDX4+5,X'FE'        SET RETURN SWITCH ON
001D30 9200 098D      0098D          3856          MVI   INDX3+3,0            SET NOP
001D34 45E0 095C            0095C    3857          BAL   14,INDEX             GO TO INDEX SUPROUTINE
001D38 9280 098D      0098D          3858          MVI   INDX3+3,X'80'        CLEAR NOP
001D3C 9200 09A1      009A1          3859          MVI   INDX4+5,0            SET RETURN SWITCH OFF
001D40 58C0 34D0            014D0    3860          L     12,REG15             RESTORE R12
001D44 07FC                          3861 BXG      BR      12                 POSSIBLE RETURN TO BLX65/66
001D46 92FC 3D45      01D45          3862          MVI   BXG+1,X'FC'          SET BXG SWITCH ON
001D4A 5080 34D0            014D0    3863          ST    RP,REG15             SAVE P ADDRESS
001D4E 41BB 0004            00004    3864          LA    WR2,4(WR2)           TAKE RIGHT IX ADDRESS
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

001D52 188B                       3865          LR    RP,WR2                RP=WR2 INDEX ADDRESS
001D54 92F0 31AB      011AB       3866          MVI   ADD4A+1,X'F0'         SET SWITCH ON
001D58 47F0 318E            0118E 3867          B       ADD5A               GO TO ADD21
001D5C 18A8                       3868 BXF      LR    WR1,RP                RESTORE ADDRESS
001D5E 45F0 3548            01548 3869          BAL   15,INDIC              INDICATORS UPDATING
001D62 4700 3DA0            01DA0 3870 BXA      BC    NOP,BXB               CONDITION SWITCH
001D66 D300 33A5 8000 013A5 00000 3871          MVZ   AQSIGN(1),0(RP)       IS SAVED IX SIGN PRESENT
001D6C D500 33A4 33A5 013A4 013A5 3872          CLC   APSIGN(1),AQSIGN      IX SIGN
001D72 4770 3D82            01D82 3873          BC    7,BXC                 NO,BRANCH
001D76 9101 0003      00003       3874          TM    HPEZ,X'01'            IS E/Z ON
001D7A 4710 3D82            01D82 3875          BC    1,BXC                 YES,BRANCH
001D7E 46E0 3DA0            01DA0 3876          BCT   WR5,BXB               NO,TEST LAST CARRY
001D82 95F0 3231      01231       3877 BXC      CLI   AQEND+1,X'F0'         IS AQEND ON
001D86 4770 0E26            00E26 3878          BC    7,C24I                NO,ARITHMETIC CHECK TESTING
001D8A 4250 0F58            00F58 3879          STC   CNTR,STC              TEST IF BRANCH ADDRESS IS
001D8E 9101 0F58      00F58       3880          TM    STC,X'01'             CORRECT
001D92 4780 07F6            007F6 3881          BC    8,ENTRY0              YES,BRANCH
001D96 45F0 00F4            000F4 3882          BAL   15,MESTOP             ERROR, STOP SIMULATION
001D9A 0000
001D9C 0000019D                   3883          DC  A(NOBRAD)           *
001DA0 5850 34D0            014D0 3884 BXB      L     CNTR,REG15            CNTR=RP
001DA4 41C0 000C            0000C 3885          LA    WR3,12                SUBTRACT 12 FROM CNTR
001DA8 1B5C                       3886          SR    CNTR,WR3              *
001DAA 47F0 3D82            01D82 3887          B       BXC                 BRANCH
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                            ASM H V 05 22.10 05/07/81

                                   3891 ***********************************************************************
                                   3892 *                                                                     *
                                   3893 *                     BRANCH AND MODIFY IMMEDIATE                     *
                                   3894 *                                                                     *
                                   3895 *  THE CONDITION SWITCH IS SET OFF.                                   *
                                   3896 *  CONTROL IS THEN GIVEN TO BCXM-64 DESCRIBED BELOW.                  *
                                   3897 *                                                                     *
                                   3898 ***********************************************************************

001DAE 92F0 3D63      01D63        3900 BXM62    MVI   BXA+1,X'F0'            SET CONDITION SWITCH OFF
001DB2 47F0 3DC2            01DC2  3901          B       BCXM64+4             BRANCH

                                   3903 ***********************************************************************
                                   3904 *                                                                     *
                                   3905 *                    BRANCH CONDITIONALLY AND MODIFY                  *
                                   3906 *                                                                     *
                                   3907 *  THE CONDITION SWITCH IS SET ON.                                    *
                                   3908 *  CONTROL IS THEN GIVEN TO THE Q ADDRESS CONVERSION  SUBROUTINE  OF  *
                                   3909 *  BX-61.                                                             *
                                   3910 *                                                                     *
                                   3911 ***********************************************************************

001DB6 9200 3D63      01D63        3913 BCX63    MVI   BXA+1,X'00'            SET CONDITION SWITCH ON
001DBA 47F0 3D00            01D00  3914          B       BX61+4               BRANCH

                                   3916 ***********************************************************************
                                   3917 *                                                                     *
                                   3918 *              BRANCH CONDITIONALLY AND MODIFY IMMEDIATE              *
                                   3919 *                                                                     *
                                   3920 *  THE CONDITION SWITCH IS SET ON.                                    *
                                   3921 *  FIVE NUMERICS ARE MOVED FROM CNTR+7 TO  A SMALL  BUFFER.  CONTROL  *
                                   3922 *  IS THEN GIVEN  TO THE P ADDRESS CONVERSION  SUBROUTINE OF  BX-61.  *
                                   3923 *                                                                     *
                                   3924 ***********************************************************************

001DBE 9200 3D63      01D63        3926 BCXM64   MVI   BXA+1,X'00'            SET CONDITION SWITCH ON
001DC2 D303 0570 3374 00570 01374  3927          MVZ   FDIC(4),ATAB2          SET 4 ZONES TO 'F'
001DC8 D103 0570 5007 00570 00007  3928          MVN   FDIC(4),7(CNTR)        STORE 4 NUMERIC DIGITS FROM CNTR
001DCE D200 0574 500B 00574 0000B  3929          MVC   FDIC+4(1),11(CNTR)     STORE LAST DIGIT
001DD4 94DF 0570      00570        3930          NI    FDIC,X'DF'             SET FLAG
001DD8 4190 0574            00574  3931          LA    RQ,FDIC+4              RQ=FDIC+4 ADDRESS
001DDC 9200 3D45      01D45        3932          MVI   BXG+1,0                SET BXG SWITCH OFF
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
001DE0 47F0 3D20            01D20  3933          B       BXD                  BRANCH


                                   3935 ************************************************************************
                                   3936 *                                                                      *
                                   3937 *                          BRANCH AND LOAD                             *
                                   3938 *                                                                      *
                                   3939 *  CONTROL IS DIRECTLY GIVEN TO THE Q ADDRESS  CONVERSION ROUTINE OF   *
                                   3940 *  BX-61,  BUT IS RETURNED TO BLX1 AFTER ADDING THE Q FIELD  TO  THE   *
                                   3941 *  INDEX REGISTER.                                                     *
                                   3942 *  FIVE NUMERICS  ARE MOVED FROM THE Q ADDRESS FIELD (WITH SIGN)  TO   *
                                   3943 *  THE  INDEX REGISTER ADDRESS.  THE  INSTRUCTION  COUNTER IS LOADED   *
                                   3944 *  WITH THE P ADDRESS, AND THE ROUTINE PROCEEDS TO A VALIDITY  CHECK   *
                                   3945 *  OF THE ADDRESS.                                                     *
                                   3946 *                                                                      *
                                   3947 ************************************************************************

001DE4 45C0 3D04            01D04  3949 BLX65    BAL   12,BXE              CONVERT Q ADDRESS
001DE8 4B90 09D0            009D0  3950 BLX1     SH    RQ,DEC4              RQ=RQ-4
001DEC 41A0 0004            00004  3951          LA    WR1,4                *                              V1L2
001DF0 1ABA                        3952          AR    WR2,WR1              *                              V1L2
001DF2 96F0 B000      00000        3953 BLX651   OI    0(WR2),X'F0'         CLEAR RECEIVING AREA           V1L2
001DF6 06B0                        3954          BCTR  WR2,0                *                              V1L2
001DF8 46A0 3DF2            01DF2  3955          BCT   WR1,BLX651           *                              V1L2
001DFC D103 B000 9000 00000 00000  3956          MVN   0(4,WR2),0(RQ)       MOVE 5 NUMERIC> FROM Q ADDR    V1L2
001E02 D200 B004 9004 00004 00004  3957          MVC   4(1,WR2),4(RQ)       (WITH SIGN) TO WR2 ADDRESS
001E08 94DF B000      00000        3958          NI    0(WR2),X'DF'         SET IX FLAG
001E0C 1858                        3959 BLX2     LR    CNTR,RP              CNTR=P ADDRESS
001E0E 47F0 0F44            00F44  3960          B       TESTBR             GO TO TEST BRANCH ADDRESS


                                   3962 ************************************************************************
                                   3963 *                                                                      *
                                   3964 *                      BRANCH AND LOAD IMMEDIATE                       *
                                   3965 *                                                                      *
                                   3966 *  THE RQ REGISTER IS LOADED  WITH CNTR+11. THEN,  THE SEQUENCE CON-   *
                                   3967 *  TINUES AS FOR BLX-65 EXCEPT THAT IT STARTS WITH P ADDRESS CONVER-   *
                                   3968 *  SION.                                                               *
                                   3969 *                                                                      *
                                   3970 ************************************************************************

001E12 4195 000B            0000B  3972 BLXM66   LA    RQ,11(CNTR)          RQ=CNTR+11
001E16 45C0 3D20            01D20  3973          BAL   12,BXD               BRANCH
001E1A 47F0 3DE8            01DE8  3974          B       BLX1               BRANCH
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  3976 ************************************************************************
                                  3977 *                                                                      *
                                  3978 *                          BRANCH AND STORE                            *
                                  3979 *                                                                      *
                                  3980 *  THIS ROUTINE IS THE SAME AS THAT OF BLX-65,  EXCEPT THAT FIVE NU-   *
                                  3981 *  MERICS ARE MOVED FROM THE INDEX REGISTER TO THE Q ADDRESS.          *
                                  3982 *                                                                      *
                                  3983 ************************************************************************

001E1E 45C0 3D04            01D04 3985 BSX67    BAL   12,BXE                 *
                                  3986 *        SH    RQ,DEC4                RQ=RQ-4          (DELETED) V1L2
001E22 41A0 0004            00004 3987          LA    WR1,4                  *                          V1L2
001E26 96F0 9000      00000       3988 BSX67A   OI    0(RQ),X'F0'            CLEAR RECEIVING AREA       V1L2
001E2A 0690                       3989          BCTR  RQ,0                   *                          V1L2
001E2C 46A0 3E26            01E26 3990          BCT   WR1,BSX67A             *                          V1L2
001E30 D103 9000 B000 00000 00000 3991          MVN   0(4,RQ),0(WR2)         MOVE 5 NUMERICS FROM WR2 ADDRESS
001E36 D200 9004 B004 00004 00004 3992          MVC   4(1,RQ),4(WR2)         (WITH SIGN) TO RQ ADDRESS
001E3C 94DF 9000      00000       3993          NI    0(RQ),X'DF'            SET IX FLAG
001E40 47F0 3E0C            01E0C 3994          B       BLX2                 BRANCH
                                  3995          AGO   .IXX3
                                  3996 .IXX3    ANOP
                                  3997 *+* DISKV    DISK  VERSION
                                  3998 *        PRINT OFF
                                  3999          PUNCH ' SLC  006000'
                                  4000 *        PRINT ON
                                  4001 *-*      END
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4003 ************************************************************************
                                  4004 *                                                                      *
                                  4005 *                    READ ALPHAMERICALLY (TYPEWRITER)                  *
                                  4006 *                                                                      *
                                  4007 *  THIS SEQUENCE LOADS THE ADDRESSES OF THE CODE CONVERSION   FIELDS  *
                                  4008 *  FOR SUBSEQUENT USE BY THE 'MASK' SUBROUTINE.  THE FIELD EXPANSION  *
                                  4009 *  SWITCH OF THE 'VALIN'  SUBROUTINE IS SET ON.                        *
                                  4010 *  THEN, CONTROL IS GIVEN TO THE 'READ NUMERICALLY' (TYPEWRITER) SE-   *
                                  4011 *  QUENCE DESCRIBED UNDER 'CONSOLE SIMULATION'.                        *
                                  4012 *                                                                      *
                                  4013 ************************************************************************

001E44 41B0 05D5          005D5   4015 RATY     LA    WR2,TABLE2            LOAD ADDRESSES OF CODE CON-
001E48 41A0 3E58          01E58   4016          LA    WR1,TBRATY            VERSION TABLE
001E4C 92F0 04B3    004B3         4017          MVI   FLEXSW+1,X'F0'        SET FIELD EXPANSION SWITCH ON
001E50 9241 0511    00511         4018          MVI   MASK+1,X'41'          SET COMMAND MASK = 41
001E54 47F0 0204          00204   4019          B       RNTYGO-4            BRANCH TO WRITE

001E58 4E02105FAA                 4021 TBRATY   DC    X'4E02105FAA'         *
001E5D 7B030A34AA33               4022          DC    X'7B030A34AA33'       *
001E63 C000AA                     4023          DC    X'C000AA'             *
001E66 4D0024                     4024          DC    X'4D0024'
001E69 5D0004                     4025          DC    X'5D0004'
001E6C 4C00AA                     4026          DC    X'4C00AA'
001E6F 6C00AA                     4027          DC    X'6C00AA'
001E72 E000AA                     4028          DC    X'E000AA'             *
001E75 400000                     4029          DC    X'400000'             *
001E78 EF                         4030          DC    X'EF'                 *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4032 ************************************************************************
                                  4033 *                                                                      *
                                  4034 *                        WRITE NUMERICALLY (TYPEWRITER)                *
                                  4035 *                                                                      *
                                  4036 *  THIS SEQUENCE FIRST PREPARES THE ADDRESSES NECESSARY FOR CODE CON-  *
                                  4037 *  VERSION (SEE 'MASK' SUBROUTINE).  IT  SETS  THE FIELD COMPRESSION  *
                                  4038 *  SWITCH IN THE 'VALOUT' SUBROUTINE OFF.  WR3 IS GIVEN THE VALUE 0,  *
                                  4039 *  CORRESPONDING TO A STEP OF 1, TO AVOID  THE  2  BYTES-IN-ONE-BYTE  *
                                  4040 *  ALPHABETIC MODE, AND CONTROL IS GIVEN TO THE 'GETEOR'  SUBROUTINE  *
                                  4041 *  WHICH PROVIDES THE LENGTH OF THE FIELD TO BE WRITTEN.  THEN,  THE  *
                                  4042 *  SEQUENCE GOES TO THE 'MASK' SUBROUTINE  TO PREPARE THE  CODE CON-  *
                                  4043 *  VERSION TABLE AT ADDRESS 128 (IF NECESSARY).                       *
                                  4044 *                                                                      *
                                  4045 *  THE BYTE COUNT OBTAINED FROM THE 'GETEOR' SUBROUTINE IS THEN COM-  *
                                  4046 *  PUTED.  IF  THE BYTE  COUNT IS LOWER  THAN OR EQUAL TO  100,  ONE  *
                                  4047 *  SINGLE 'WRITE' OPERATION WILL BE EXECUTED.  IF IT IS GREATER THAN  *
                                  4048 *  100, A NEW LINE MUST BE WRITTEN WITH THE REMAINING BYTE COUNT. AN  *
                                  4049 *  EXIT SWITCH IS SET TO RESUME THE OUTPUT OPERATION ON THIS REMAIN-  *
                                  4050 *  ING FIELD.  THE LOOP WILL BE PERFORMED UNTIL THE FIELD HAS   BEEN  *
                                  4051 *  EXHAUSTED.                                                          *
                                  4052 *                                                                      *
                                  4053 *  WHEN THE COMPUTATION OF THE BYTE COUNT IS TERMINATED, CONTROL  IS  *
                                  4054 *  GIVEN TO THE 'VALOUT' SUBROUTINE WHICH CONVERTS THE BYTES TO OUT-  *
                                  4055 *  PUT CODES AND MOVES THEM TO BUFFER 'IOBUFF'.  CONTROL IS GIVEN TO  *
                                  4056 *  THE 'TYPIO' SUBROUTINE WHICH PERFORMS THE 'WRITE' OPERATION PROP-  *
                                  4057 *  ER.                                                                 *
                                  4058 *                                                                      *
                                  4059 *  NORMAL EXIT OF THE SEQUENCE IS TO 'BIR' THROUGH THE 'SWITCH INTER-  *
                                  4060 *  ROGATION' (ENTRY3) SUBROUTINE. HOWEVER, THE ERROR EXIT OF 'TYPIO'  *
                                  4061 *  GIVES CONTROL TO THE 'EXCRET' SUBROUTINE WHICH  DISTINGUISHES BE-  *
                                  4062 *  TWEEN =                                                             *
                                  4063 *                                                                      *
                                  4064 *  - 'INTERVENTION REQUIRED' WHICH GIVES CONTROL TO 'ALARM' AND THEN  *
                                  4065 *    TRIES TO RESUME THE ENTIRE OPERATION                              *
                                  4066 *  - 'UNIT EXCEPTION' WHICH IS CONSIDERED AS A CATASTROPHIC  FAILURE  *
                                  4067 *    OF THE 1052                                                       *
                                  4068 *  - 'SENSE INDICATION' WHICH ALLOWS SETTING THE WRITE CHECK INDICA-  *
                                  4069 *    TOR 07 ON                                                         *
                                  4070 *                                                                      *
                                  4071 ************************************************************************

001E79 00
001E7A 41B0 058A           0058A  4073 WNTY     LA    WR2,TABLE1           LOAD ADDRESSES OF CODE CON-
001E7E 41A0 3EE6           01EE6  4074           LA    WR1,TBWNTY           VERSION TABLE
001E82 9200 43A9      023A9       4075           MVI   FLCOSW+1,X'00'       SET FIELD COMPRESSION SWITCH OFF
001E86 9211 0511      00511       4076           MVI   MASK+1,X'11'         SET COMMAND MASK = 11
001E8A 41C0 0000           00000  4077           LA    WR3,0                *
001E8E 45F0 43F6           023F6  4078 WNTY1    BAL   15,GETEOR            SEARCH RECORD MARK
001E92 18ED                       4079           LR    WR5,WR4              LOAD LENGTH OF 'WRITE' INTO WR5
001E94 45F0 0510           00510  4080           BAL   15,MASK              PREPARE CODE CONVERSION TABLE
001E98 9200 3ED9      01ED9       4081 WNTYGO   MVI   WRISWI+1,X'00'       SET EXIT SWITCH TO NORMAL
001E9C 41F0 0064           00064  4082           LA    15,100               R15=100
001EA0 19EF                       4083           CR    WR5,15               IS WR5 GREATER THAN 100
001EA2 0000 0000           00000  4084           BC    BNH,WNTYG2           NO,BRANCH
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81

   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
001EA6 1BEF                        4085          SR    WR5,15                 YES,WR5=WR5-100
001EA8 18DF                        4086          LR    WR4,15                 WR4=100
001EAA 92F0 3ED9      01ED9        4087          MVI   WRISWI+1,X'F0'         SET RETRY SWITCH ON
001EAE 42D0 3EB9            01EB9  4088 WNTYG3   STC   WR4,WNTY3+3            STORE DATA LENGTH IN CCW COUNT
001EB2 45F0 43A2            023A2  4089          BAL   15,VALOUT              CHECK VALIDITY OF OUTPUT DATA
                                   4090 *        BAL   R1,TYPIO               CALL TYPEWRITER
001EB6 00090164                    4091 WNTY3    DC    X'00090164'            WRITE 100 BYTES FROM IOBUFF
001EBA 0000
001EBC 00003AB                     4092          DC  A(IOBUFF)            *
001EC0 47F0 3ED8            01ED8  4093          B       WRISWI               OK,EXIT
001EC4 4700 0000            00000  4094          BC    NOP,0                  U E,SHOULD NOT OCCUR,ALARM
001EC8 47F0 3F5C            01F5C  4095          B       WNTYAL               INT.REQ.,GO TO ALARM
001ECC 0000 0000      00000        4096          TM    TYPERR+1,X'30'         SENSE EXIT,TEST BITS 2+3
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
001ED0 4780 3F5C            01F5C  4097          BZ      WNTYAL               ZERO,EXIT
001ED4 9640 0002      00002        4098          OI    NYCHEK,X'40'           SET INDICATOR 07 ON
001ED8 4700 3E98            01E98  4099 WRISWI   BC    NOP,WNTYGO             'EXIT' OR 'RETRY' SWITCH
001EDC 47F0 0262            00262  4100          B       TYPSWI+4             INTERROGATE I/O INDICATOR
001EE0 18DE                        4101 WNTYG2   LR    WR4,WR5                WR4= WR5
001EE2 47F0 3EAE            01EAE  4102          B       WNTYG3               BRANCH
001EE6 DF00AA                      4103 TBWNTY   DC    X'DF00AA'
001EE9 DA00AA                      4104          DC    X'DA00AA'
001EEC FA00AA                      4105          DC    X'FA00AA'              *
001EEF FF00AA                      4106          DC    X'FF00AA'              *
001EF2 FC007C                      4107          DC    X'FC007C'              *
001EF5 EF                          4108          DC    X'EF'                  *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  4110 ***************************************************************************
                                  4111 *                                                                         *
                                  4112 *                  WRITE ALPHAMERICALLY (TYPEWRITER)                      *
                                  4113 *                                                                         *
                                  4114 *  THIS SEQUENCE PREPARES THE ADDRESSES FOR CODE CONVERSION SO  THAT      *
                                  4115 *  THEY CAN BE USED BY THE 'MASK' SUBROUTINE. IT SETS THE FIELD COM-      *
                                  4116 *  PRESSION SWITCH IN THE 'VALOUT' SUBROUTINE ON. WR3 IS LOADED WITH      *
                                  4117 *  1 WHICH CORRESPONDS TO A STEP OF TWO. THIS ALLOWS CONDENSING  TWO      *
                                  4118 *  ALPHAMERIC BYTES INTO ONE SINGLE OUTPUT CODE. THEN,  CONTROL   IS      *
                                  4119 *  GIVEN TO THE 'WRITE NUMERICALLY' (TYPEWRITER) SEQUENCE.                *
                                  4120 *                                                                         *
                                  4121 ***************************************************************************

001EF6 41B0 0623          00623   4123 WATY     LA    WR2,TABLE3            LOAD ADDRESSES OF CODE CON-
001EFA 41A0 3F0E          01F0E   4124          LA    WR1,TBWATY            VERSION TABLE
001EFE 92F0 43A9    023A9         4125          MVI   FLCOSW+1,X'F0'        SET FIELD COMPRESSION SWITCH ON
001F02 9251 0511    00511         4126          MVI   MASK+1,X'51'          SET COMMAND MASK = 51
001F06 41C0 0001          00001   4127          LA    WR3,1                 *
001F0A 47F0 3E8E          01E8E   4128          B       WNTY1               *

001F0E 0A007B                     4130 TBWATY   DC    X'0A007B'             *
001F11 33007E                     4131          DC    X'33007E'
001F14 24004D                     4132          DC    X'24004D'
001F17 22007B                     4133          DC    X'22007B'             *
001F1A 04005D                     4134          DC    X'04005D'
001F1D 1F00AA                     4135          DC    X'1F00AA'             *
001F20 10004E                     4136          DC    X'10004E'
001F23 500060                     4137          DC    X'500060'             *
001F26 5F004F                     4138          DC    X'5F004F'             *
001F29 EF                         4139          DC    X'EF'                 *

                                  4141 ***************************************************************************
                                  4142 *                                                                         *
                                  4143 *                     DUMP NUMERICALLY (TYPEWRITER)                       *
                                  4144 *                                                                         *
                                  4145 *  THIS SEQUENCE COMPUTES THE MAXIMUM ADDRESS OF THE 20K MODULE, AND      *
                                  4146 *  COMPARES IT TO THE P ADDRESS. THE BYTE COUNT TO BE WRITTEN IS THUS     *
                                  4147 *  OBTAINED.                                                              *
                                  4148 *  THEN, THE ADDRESSES FOR CODE CONVERSION ARE LOADED FOR USE BY THE      *
                                  4149 *  'MASK' SUBROUTINE. THE FIELD COMPRESSION SWITCH IN THE   'VALOUT'      *
                                  4150 *  SUBROUTINE IS SET OFF. WR3 IS LOADED WITH ZERO FOR OPERATION   IN      *
                                  4151 *  NUMERICAL MODE.  CONTROL IS GIVEN DIRECTLY TO THE  'WRITE  NUMER-      *
                                  4152 *  ICALLY' (TYPEWRITER) SEQUENCE.                                         *
                                  4153 *                                                                         *
                                  4154 *                                                                         *
                                  4155 ***************************************************************************

001F2A 18E6                       4157 DNTY     LR    WR5,MAPORG            COMPARE P ADDRESS WITH MAXIMUM
001F2C 5AE0 3F50          01F50   4158          A     WR5,DEC20K            ADDRESS OF THE 20K MODULE
001F30 198E                       4159          CR    RP,WR5                *
001F32 47A0 3F2C          01F2C   4160          BC    10,DNTY+2             *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                          ASM H V 05 22.10 05/07/81
001F36 18E8                      4161          SR    WR5,RP              *
001F38 41B0 058A           0058A 4162          LA    WR2,TABLE1          LOAD ADDRESSES OF CODE CON-
001F3C 41A0 3F54           01F54 4163          LA    WR1,TBDNTY          VERSION TABLE
001F40 9291 0511     00511       4164          MVI   MASK+1,X'91'         SET COMMAND MASK = 91
001F44 9200 43A9     023A9       4165          MVI   FLCOSW+1,X'00'       SET COMPRESSION SWITCH OFF
001F48 41C0 0000           00000 4166          LA    WR3,0               WR3=0
001F4C 47F0 3E94           01E94 4167          B       WNTY1+6           BRANCH TO WRITE

001F50 00004E20                  4169 DEC20K    DC    F'20000'            MAXIMUM ADDRESS OF 20K MODULE

001F54 FA007B                    4171 TBDNTY    DC    X'FA007B'           *
001F57 FC007C                    4172           DC    X'FC007C'           *
001F5A EF                        4173           DC    X'EF'               *

001F5B 00
001F5C 4510 0060           00060 4175 WNTYAL    BAL   R1,ALARM            *
001F60 47F0 3EB2           01EB2 4176          B       WNTY3-4           *
```

```
                                 4179 ************************************************************************
                                 4180 *                                                                      *
                                 4181 *                  CONTROL OPERATIONS ON TYPEWRITER                    *
                                 4182 *                                                                      *
                                 4183 *  THIS SEQUENCE USES THE MAJOR PART OF THE 'MESSAG' SUBROUTINE, BUT   *
                                 4184 *  IT FIRST CHANGES THE REQUESTED WRITE COMMAND TO A  'WRITE  INHIB-   *
                                 4185 *  IT CARRIAGE RETURN' COMMAND. THIS AVOIDS ANY POSSIBLE INTERFEREN-   *
                                 4186 *  CE BETWEEN THE 1620 CONTROL COMMAND AND THE 'WRITE AUTO  CARRIAGE   *
                                 4187 *  RETURN' OF THE NORMAL 'MESSAG' SUBROUTINE.                          *
                                 4188 *                                                                      *
                                 4189 *  THE 1052 CONTROL CHARACTER IS OBTAINED FROM DIGIT Q11 BY MEANS OF   *
                                 4190 *  TABLE 'KTAB'. IT IS SENT TO THE TYPEWRITER IN THE SAME WAY AS   A   *
                                 4191 *  ONE-BYTE MESSAGE.                                                   *
                                 4192 *                                                                      *
                                 4193 ************************************************************************
```

```
001F64 9201 00B0     000B0       4195 KTYPE    MVI   MESSIO+6,X'01'       SET WRITE INHIBIT CARRIAGE RE-
                                 4196 *                                       TURN COMMAND
001F68 18E0                      4197          LR    WR5,0               *
001F6A D100 07A9 500B 007A9 0000B 4198          MVN   ACOMA+1(1),11(CNTR) INSERT IN WR4 BINARY EQUIVALENT
001F70 48D0 07A8           007A8 4199          LH    WR4,ACOMA           OF Q11
001F74 43ED 3F92           01F92 4200          IC    WR5,KTAB(WR4)       INSERT IN WR5 CHARACTER TO BE
                                 4201 *                                       PRINTED
001F78 12EE                      4202          LTR   WR5,WR5             IS CHARACTER VALID
001F7A 4780 07F6           007F6 4203          BZ      ENTRY1            NO,RETURN TO INTERPRETIVE LOOP
001F7E 42E0 3F91           01F91 4204          STC   WR5,KMESS+1         YES,STORE CONTROL COMMAND IN
001F82 45F0 008C           0008C 4205          BAL   15,MESSA2           MESSAGE BUFFER
001F86 0000
001F88 00001F90                  4206          DC  A(KMESS)                *
001F8C 47F0 07F6           007F6 4207          B       ENTRY1            RETURN TO INTERPRETIVE ROUTINE
001F90 0100                      4208 KMESS    DC    X'0100'             CONTROL COMMAND (BUFFER)
001F92 0040151625000000          4209 KTAB     DC    X'0040151625000000'5' TABLE OF CORRESPONDANCE WITH
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
001F9B 00000000000000             4210          DC    X'00000000000000'          1052 CONTROL COMMANDS



                                  4213 ************************************************************************
                                  4214 *                                                                      *
                                  4215 *                    READ ALPHAMERICALLY (CARD)                        *
                                  4216 *                                                                      *
                                  4217 *  THIS SEQUENCE  LOADS THE ADDRESSES OF THE CODE CONVERSION  FIELDS   *
                                  4218 *  FOR SUBSEQUENT USE BY THE 'MASK' SUBROUTINE.  THE FIELD EXPANSION   *
                                  4219 *  SWITCH OF THE 'VALIN'  SUBROUTINE IS SET ON.                        *
                                  4220 *  THEN, CONTROL IS GIVEN TO THE 'READ NUMERICALLY' (CARD)  SEQUENCE   *
                                  4221 *  DESCRIBED UNDER 'CONSOLE SIMULATION'                                *
                                  4222 *                                                                      *
                                  4223 ************************************************************************

001FA2 41B0 05D5          005D5   4225 RACD     LA    WR2,TABLE2             LOAD ADDRESSES OF CODE CON-
001FA6 41A0 3FB6          01FB6   4226          LA    WR1,TBRACD             VERSION TABLE
001FAA 92F0 04B3    004B3         4227          MVI   FLEXSW+1,X'F0'         SET FIELD EXPANSION SWITCH ON
001FAE 9245 0511    00511         4228          MVI   MASK+1,X'45'           SET COMMAND MASK = 45
001FB2 47F0 0282          00282   4229          B       RNCD1A               *
001FB6 400000                     4230 TBRACD   DC    X'400000'              *
001FB9 EF                         4231          DC    X'EF'                  *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4233 ********************************************************************************
                                  4234 *                                                                              *
                                  4235 *                          WRITE NUMERICALLY (CARD)                            *
                                  4236 *                                                                              *
                                  4237 *  THIS SEQUENCE FIRST PREPARES THE ADDRESSES NECESSARY FOR CODE CON-          *
                                  4238 *  VERSION (SEE 'MASK' SUBROUTINE). IT SETS THE 'DUMP  CARD'  SWITCH          *
                                  4239 *  AND THE FIELD COMPRESSION SWITCH IN THE 'VALOUT' SUBROUTINE  OFF.           *
                                  4240 *  THE 'DUMP CARD' SWITCH IS USED BY THE 'DUMP  NUMERICALLY  (CARD)'           *
                                  4241 *  SEQUENCE  REQUIRING A RETRY OF CARD PUNCHING UNTIL  EXHAUSTION OF           *
                                  4242 *  THE DUMP. WR3 IS GIVEN THE VALUE ZERO CORRESPONDING TO A STEP  OF           *
                                  4243 *  1, TO AVOID THE TWO BYTES-IN-ONE-BYTE ALPHABETIC MODE *).                   *
                                  4244 *                                                                              *
                                  4245 *  THE SEQUENCE GOES TO THE 'MASK' SUBROUTINE AND, IF NECESSARY,PRE-           *
                                  4246 *  PARES THE CODE CONVERSION TABLE AT ADDRESS 128.                             *
                                  4247 *                                                                              *
                                  4248 *  THE 'VALOUT SUBROUTINE TRANSLATES THE OUTPUT DATA BYTES,   CHECKS           *
                                  4249 *  THEM, AND MOVES THEM FROM CORE STORAGE TO 'IOBUFF'.                         *
                                  4250 *                                                                              *
                                  4251 *  THE SUPERVISOR CALL SVC 1 EMITS A WRITE OPERATION FOR  80  BYTES.           *
                                  4252 *  AS SOON AS THE REQUEST IS ACCEPTED, PROCESSING STOPS AT 'WNLOCK'.           *
                                  4253 *  AT CHANNEL END, THE CONTROL PROGRAM RETURNS CONTROL TO   'NRMWNC'           *
                                  4254 *  WHICH UNLOCKS THE 'WAIT' SWITCH AND RETURNS TO THE POINT OF INTER-          *
                                  4255 *  RUPTION.                                                                    *
                                  4256 *                                                                              *
                                  4257 *  THEN THE SEQUENCE  EXITS TO 'ENTRY3' WHICH INTERROGATES AND  DIS-           *
                                  4258 *  PLAYS WRITE INDICATOR 07  AND RETURNS TO 'BIR'.                             *
                                  4259 *                                                                              *
                                  4260 *  IN CASE OF AN ERROR CONDITION,THE 'EXCRET' SUBROUTINE IS ENTERED=           *
                                  4261 *                                                                              *
                                  4262 *  - 'INTERVENTION REQUIRED' WILL CAUSE THE MESSAGE 'PUNCH/DISK  IN-           *
                                  4263 *    TERLOCK' TO BE TYPED ON THE 1052. CONTROL IS THEN GIVEN TO 'BIR'          *
                                  4264 *    WITHOUT INCREMENTING THE INSTRUCTION COUNTER. THE PUNCH  OPERA-           *
                                  4265 *    TION IS THUS RESUMED AS SOON AS THE DEVICE IS READY.                      *
                                  4266 *                                                                              *
                                  4267 *  - SENSE BITS 2 AND/OR 3 WILL CAUSE WRITE CHECK INDICATOR 07 TO BE           *
                                  4268 *    SET ON.                                                                   *
                                  4269 *                                                                              *
                                  4270 *  - 'UNIT EXCEPTION' WILL BE HANDLED AS ANY OTHER CATASTROPHIC  ER-           *
                                  4271 *    ROR WHICH SHOULD NOT OCCUR, I.E. A SPECIAL MESSAGE IS TYPED  ON           *
                                  4272 *    THE 1052 AND SIMULATION STOPS (SEE 'EXCRET' SUBROUTINE).                  *
                                  4273 *                                                                              *
                                  4274 *  *) THE 'GETEOR' SUBROUTINE IS NOT USED BECAUSE THE OUTPUT  OPERA-           *
                                  4275 *     TION ISSUES A COUNT OF 80 BYTES.                                         *
                                  4276 ********************************************************************************

001FBA 41B0 058A          0058A   4278 WNCD     LA    WR2,TABLE1              LOAD ADDRESSES OF CODE CON-
001FBE 41A0 4078          02078   4279          LA    WR1,TBWNCD              VERSION TABLE
001FC2 9200 4011  02011           4280          MVI   DUMCAR+1,X'00'          SET DUMP CARD SWITCH OFF
001FC6 9214 0511  00511           4281          MVI   MASK+1,X'14'            SET COMMAND MASK = 14
001FCA 41C0 0000          00000   4282          LA    WR3,0                   *
001FCE 9200 43A9  023A9           4283          MVI   FLCOSW+1,X'00'          SET FIELD COMPRESSION SWITCH OFF
001FD2 45F0 0510          00510   4284 WNCD1    BAL   15,MASK                 LOOK FOR COD.CONV.TABLE
001FD6 41D0 0050          00050   4285          LA    WR4,80                  *
001FDA 45F0 43A2          023A2   4286          BAL   15,VALOUT               CHECK VALIDITY OF OUTPUT DATA
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                         ASM H V 05 22.10 05/07/81

                                   4287 *                                                     AND MOVE THEM
001FDE 070007000700                4288          CNOP  4,8                             *
001FE4                             4289 WNCD2  DS 0H  SVC   1                           I/O REQUEST AND INTERRUPT AT
001FE4 000D                        4290          DC    X'000D'                        .FOR CARD PUNCH                    ..
001FE6 0000
001FE8 00002070                    4291          DC    A(WCDCCW)                       *
001FEC                             4292 WNCERR   DS    C                               *
001FED                             4293          DS    3C                              *
001FF0 0000000006000000            4294 CSWAAA   DC    X'0000000006000000'             *
001FF8                             4295 WNCPSW   DS    D                               *
002000 0000202C                    4296          DC    A(NRMWNC)                       *
002004 0000203C                    4297          DC    A(EXCWNC)                       *
002008 92F0 400D       0200D       4298          MVI   WNLOCK+1,X'F0'                  SET WAIT SWITCH ON
00200C 47F0 400C             0200C 4299 WNLOCK   B       *                             ACCEPTED,WAIT SWITCH
002010 4700 4018             02018 4300 DUMCAR   BC    NOP,*+8                         DUMP CARD SWITCH
002014 47F0 0262             00262 4301          B        TYPSWI+4                     INTERROGATE I/O INDICATOR
002018 198E                        4302          CR    RP,WR5                          IS RP GREATER THAN OR EQUAL  WR5
00201A 47A0 4014             02014 4303          BC    10,DUMCAR+4                     YES,BRANCH
00201E 41A6 0050             00050 4304          LA    WR1,80(MAPORG)                  NO,IS RP LESS THAN
002022 198A                        4305          CR    RP,WR1                          MAPORG+80
002024 4740 4014             02014 4306          BL       DUMCAR+4                     YES,BRANCH
002028 47F0 3FD6             01FD6 4307          B        WNCD1+4                      NO,LOOP

00202C 9200 400D       0200D       4309 NRMWNC   MVI   WNLOCK+1,X'00'                  SET WAIT SWITCH OFF
002030 92FF 3FF8       01FF8       4310          MVI   WNCPSW,X'FF'                    ENABLE INTERRUPTIONS
002034 0700                        4311          CNOP  2,4                             *
                                   4312 *        SVC   3                               RETURN TO POINT OF INTERRUPTION
002036 0000
002038 00001FF8                    4313          DC    A(WNCPSW)                       *
00203C 5010 034C             0034C 4314 EXCWNC   ST    R1,RNCSAV                       SAVE R1                      V1L2
002040 0000 0000             00000 4315          B       EXCR3                         UE,CATASTROPHIC DAMAGE
  IEV044  *** ERROR ***  UNDEFINED SYMBOL
002044 47F0 4060             02060 4316          B       WNCDIR                        INT.REQ.,GO TO SEND MESSAGE
002048 9130 3FED       01FED       4317          TM    WNCERR+1,X'30'                  SENSE EXIT,TEST BITS 2+3
00204C 0000 0000             00000 4318          BZ       EXCR3                        IF NONE,CATASTROPHIC DAMAGE
  IEV044  *** ERROR ***  UNDEFINED SYMBOL
002050 9640 0002       00002       4319          OI    NYCHEK,X'40'                    IF SOME,SET INDICATOR 07 ON
002054 4510 07B0             007B0 4320          BAL   R1,ENTRY3                       INTERROGATE I/O INDICATORS
002058 5810 034C             0034C 4321          L     R1,RNCSAV                       RESTORE R1                   V1L2
00205C 47F0 402C             0202C 4322          B       NRMWNC                        *
002060 5810 034C             0034C 4323 WNCDIR   L     R1,RNCSAV                       RESTORE R1                   V1L2
002064 45F0 00DC             000DC 4324          BAL   15,MESTIR                       SEND 'PUNCH/DISK INTERLOCK' V1L2
                                   4325 *WNCDIR  BAL   15,MESTIR                       SEND 'PUNCH/DISK I(DELETED) V1L2
002068 00000154                    4326          DC  A(PDLOK)                        MESSAGE TO OPERATOR AND STOP

                                   4328 *             2540 PUNCH PRESENT
00206C 00000000
002070 010003A820000050            4329 WCDCCW   CCW   X'01',IOBUFF,X'20',80           WRITE 80 BYTES (SLI FLAG)
002078 D000D0                      4330 TBWNCD   DC    X'D000D0'                       *
00207B EF                          4331          DC    X'EF'                           *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  4333 ********************************************************************************
                                  4334 *                                                                              *
                                  4335 *                          WRITE ALPHAMERICALLY (CARD)                         *
                                  4336 *                                                                              *
                                  4337 *  THIS SEQUENCE PREPARES THE ADDRESSES FOR CODE CONVERSION SO  THAT           *
                                  4338 *  THEY CAN BE USED BY THE 'MASK' SUBROUTINE. IT SETS THE FIELD COM-           *
                                  4339 *  PRESSION SWITCH IN THE 'VALOUT' SUBROUTINE ON. WR3 IS LOADED WITH           *
                                  4340 *  1 WHICH CORRESPONDS TO A STEP OF TWO. THIS ALLOWS CONDENSING  TWO           *
                                  4341 *  ALPHAMERIC BYTES INTO ONE SINGLE OUTPUT CODE. THEN,  CONTROL   IS           *
                                  4342 *  GIVEN TO THE 'WRITE NUMERICALLY' (CARD) SEQUENCE.                           *
                                  4343 *                                                                              *
                                  4344 ********************************************************************************

00207C 41B0 0623          00623   4346 WACD     LA    WR2,TABLE3            LOAD ADDRESSES OF CODE CON-
002080 41A0 4098          02098   4347          LA    WR1,TBWACD            VERSION TABLE
002084 9200 4011    02011         4348          MVI   DUMCAR+1,X'00'        SET DUMP CARD SWITCH OFF
002088 9254 0511    00511         4349          MVI   MASK+1,X'54'          SET COMMAND MASK = 54
00208C 41C0 0001          00001   4350          LA    WR3,1                 *
002090 92F0 43A9    023A9         4351          MVI   FLCOSW+1,X'F0'        SET FIELD COMPRESSION SWITCH ON
002094 47F0 3FD2          01FD2   4352          B       WNCD1               *
002098 160040                     4353 TBWACD   DC    X'160040'             *
00209B EF                         4354          DC    X'EF'                 *


                                  4357 ********************************************************************************
                                  4358 *                                                                              *
                                  4359 *                          DUMP NUMERICALLY (CARD)                             *
                                  4360 *                                                                              *
                                  4361 *                                                                              *
                                  4362 *  THIS SEQUENCE COMPUTES THE MAXIMUM ADDRESS OF THE 20K MODULE, AND           *
                                  4363 *  COMPARES IT TO THE P ADDRESS TO OBTAIN THE BYTE COUNT TO BE WRIT-           *
                                  4364 *  TEN.                                                                        *
                                  4365 *                                                                              *
                                  4366 *  THE CODE CONVERSION ADDRESSES ARE LOADED FOR USE  BY  THE  'MASK'           *
                                  4367 *  SUBROUTINE.  THE FIELD COMPRESSION SWITCH IN THE 'VALOUT' SUBROU-           *
                                  4368 *  TINE IS SET OFF. WR3 IS LOADED WITH ZERO  FOR NUMERIC MODE OPERA-           *
                                  4369 *  TION.                                                                       *
                                  4370 *                                                                              *
                                  4371 *  THE DUMP  CARD SWITCH  IN THE 'WRITE NUMERICALLY' (CARD)  SEQUEN-           *
                                  4372 *  CE IS SET ON IN ORDER TO SYSTEMATICALLY RETRY CARD PUNCHING UNTIL           *
                                  4373 *  THE DUMP OPERATION IS EXHAUSTED. THEN, CONTROL IS GIVEN  TO   THE           *
                                  4374 *  'WRITE NUMERICALLY' (CARD) SEQUENCE.                                        *
                                  4375 *                                                                              *
                                  4376 *  TO TERMINATE THE DUMP OPERATION, THE  80 COLUMNS OF THE LAST DUMP           *
                                  4377 *  CARD ARE COMPLETED BY ADDING THE BYTES FOLLOWING THE  DUMP  UPPER           *
                                  4378 *  BOUNDARY OR BY CORE STORAGE WRAP-AROUND.                                    *
                                  4379 *                                                                              *
                                  4380 ********************************************************************************

00209C 41B0 058A          0058A   4382 DNCD     LA    WR2,TABLE1            LOAD ADDRESSES OF CODE CON-
0020A0 41A0 40C8          020C8   4383          LA    WR1,TBDNCD            VERSION TABLE
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
0020A4 92F0 4011      02011        4384           MVI   DUMCAR+1,X'F0'       SET DUMCAR SWITCH ON
0020A8 9294 0511      00511        4385           MVI   MASK+1,X'94'         SET COMMAND MASK =94
0020AC 18D6                        4386           LR    WR4,MAPORG           WR4=MAPORG
0020AE 5AD0 3F50            01F50  4387 DNCD1     A     WR4,DEC20K           WR4=MAPORG+20K
0020B2 198D                        4388           CR    RP,WR4               IS RP GREATER THAN WR4
0020B4 47A0 40AE            020AE  4389           BC    10,DNCD1             YES,LOOP
0020B8 18E8                        4390           LR    WR5,RP               NO, COMPARE P ADDRESS + 80 TO
0020BA 41EE 0050            00050  4391 DNCD2     LA    WR5,80(WR5)          MAXIMUM ADDRESS OF 20K MODULE
0020BE 19ED                        4392           CR    WR5,WR4              IS WR5 GREATER THAN WR4
0020C0 4740 40BA            020BA  4393           BL       DNCD2             NO,LOOP
0020C4 47F0 3FCA            01FCA  4394           B      WNCD1-8             YES,BRANCH
0020C8 0000AA                      4395 TBDNCD    DC    X'0000AA'            *
0020CB EF                          4396           DC    X'EF'                *
                                   4397 *** 1443W    PRINT
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
                                  4399 ************************************************************************
                                  4400 *                                                                      *
                                  4401 *                          PRINT NUMERICALLY                           *
                                  4402 *                                                                      *
                                  4403 *  THIS SEQUENCE FIRST PREPARES THE ADDRESSES NECESSARY FOR CODE CON- *
                                  4404 *  VERSION (SEE 'MASK' SUBROUTINE). THE FIELD COMPRESSION SWITCH  IN  *
                                  4405 *  THE 'VALOUT' SUBROUTINE IS SET OFF. WR3 IS GIVEN THE VALUE 0,COR-  *
                                  4406 *  RESPONDING TO A STEP OF 1 FOR NUMERIC MODE OPERATION.   THE  DUMP  *
                                  4407 *  SWITCH IS SET OFF TO ALLOW GIVING CONTROL TO THE 'GETEOR' SUBROU-  *
                                  4408 *  TINE FOR DETECTION OF ANY RECORD MARK IN THE LINE TO BE PRINTED.   *
                                  4409 *                                                                      *
                                  4410 *  THEN, THE 1620 PRINT INSTRUCTION PROPER IS EXAMINED TO DEFINE THE  *
                                  4411 *  WRITE OPERATION (PRINT + SPACE AFTER PRINT, OR PRINT WITHOUT SPA-   *
                                  4412 *  CE -DIGIT Q11-).                                                    *
                                  4413 *                                                                      *
                                  4414 *  IF NECESSARY, THE CODE CONVERSION  TABLE IS PREPARED BY THE 'MASK' *
                                  4415 *  SUBROUTINE.  A SEARCH FOR RECORD MARK IS MADE BY THE 'GETEOR' SUB- *
                                  4416 *  ROUTINE, AND THE OUTPUT DATA BYTES ARE CONVERTED,   CHECKED,  AND  *
                                  4417 *  MOVED TO 'IOBUFF' BY THE 'VALOUT' SUBROUTINE.                       *
                                  4418 *                                                                      *
                                  4419 *  A SUPERVISOR CALL SVC 1 IS ISSUED. AS SOON AS THE REQUEST IS  AC-   *
                                  4420 *  CEPTED, 'PRLOCK' STOPS PROCESSING AND WAITS FOR CHANNEL END.        *
                                  4421 *                                                                      *
                                  4422 *  AT CHANNEL END, 'PRINTER BUSY' INDICATOR 35 IS SET ON AND CONTROL   *
                                  4423 *  IS  RETURNED TO 'BIR' BY THE INDICATOR  INTERROGATION  SUBROUTINE   *
                                  4424 *  'ENTRY3'. 'PRINTER BUSY' INDICATOR 35 IS SET OFF AS SOON AS  'DE-   *
                                  4425 *  VICE END' OCCURS.                                                   *
                                  4426 *                                                                      *
                                  4427 *  IN CASE OF AN ERROR CONDITION, CONTROL IS GIVEN TO  THE  'EXCRET'   *
                                  4428 *  SUBROUTINE.  THERE ARE 5 DIFFERENT EXITS =                          *
                                  4429 *                                                                      *
                                  4430 *  - 'INTERVENTION REQUIRED' WILL CAUSE THE MESSAGE 'INTERVENTION RE-  *
                                  4431 *    QUIRED ON PRINTER' TO BE TYPED AND SIMULATION TO BE TEMPORARILY   *
                                  4432 *    STOPPED,  SO AS TO ALLOW THE PRINT OPERATION TO RESTART AS SOON   *
                                  4433 *    AS THE PRINTER IS READY.                                          *
                                  4434 *  - 'UNIT EXCEPTION' WILL CAUSE CHANNEL 12 INDICATOR 34 TO BE   SET  *
                                  4435 *    ON.                                                               *
                                  4436 *  - 'SENSE BIT 7' WILL CAUSE CHANNEL 9 INDICATOR 33 TO BE SET ON.     *
                                  4437 *  - SENSE BITS 2 AND 3 WILL CAUSE 'PRINTER CHECK' INDICATOR  25  TO   *
                                  4438 *    SET ON.                                                           *
                                  4439 *  - ALL OTHER ERROR CONDITIONS WILL EXIT TO THE 1052 MESSAGE  INDI-   *
                                  4440 *    CATING A CATASTROPHIC DAMAGE.                                     *
                                  4441 *                                                                      *
                                  4442 ************************************************************************

0020CC 41B0 058A          0058A   4444 PRN      LA    WR2,TABLE1          LOAD ADDRESSES OF CODE CON-
0020D0 41A0 4210          02210   4445          LA    WR1,TBPRN           VERSION TABLE
0020D4 41C0 0000          00000   4446          LA    WR3,0               *
0020D8 9200 4121    02121         4447          MVI   DUMPRI+1,X'00'      SET DUMP SWITCH OFF
0020DC 9219 0511    00511         4448          MVI   MASK+1,X'19'        SET COMMAND MASK = X'19'
0020E0 9200 43A9    023A9         4449 PRNA     MVI   FLCOSW+1,X'00'      SET FIELD COMPRE SWITCH OFF V1L2
                                  4450 *        MVI   FLCOSW+1,X'00'      SET FIELD COMPRE S(DELETED) V1L2
0020E4 9101 500B    0000B         4451 PRNB     TM    11(CNTR),X'01'      PRINT SUPPRESS SPACE        V1L2
0020E8 4710 40F8          020F8   4452          BC    1,PRNDEL            YES BRANCH                  V1L2
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81

0020EC 47F0 4100           02100  4453 SPSKBT   B       PRN1                SPACE SKIP SWITCH
0020F0 9200 4288      02288        4454          MVI   PR1CCW,X'00'          MOVE 'KPRINT' COMMAND TO CCW
0020F4 47F0 4114           02114  4455          B       PRN2                *
0020F8 9201 4288      02288        4456 PRNDEL   MVI   PR1CCW,X'01'          SET WRITE WITHOUT SPACE     V1L2
0020FC 47F0 4118           02118  4457          B       PRN3                                            V1L2
002100 9101 500B      0000B        4458 PRN1     TM    11(CNTR),X'01'        IS Q11 = 1
002104 4710 4110           02110  4459          BO      *+12                YES, BRANCH
002108 9209 4288      02288        4460          MVI   PR1CCW,X'09'          NO, SET CCW = WRITE + SPACE
00210C 47F0 4114           02114  4461          B       PRN2                *
002110 9201 4288      02288        4462          MVI   PR1CCW,X'01'          SET CCW = WRITE WITHOUT SPACE
002114 92F0 40ED      020ED        4463 PRN2     MVI   SPSKBT+1,X'F0'        SET SPACE SKIP SWITCH = OFF
002118 45F0 0510           00510  4464 PRN3     BAL   15,MASK               PREPARE CODE CONVER TABLE   V1L2
                                    4465 *        BAL   15,MASK               PREPARE CODE CONVE(DELETED) V1L2
00211C 4110 0078           00078  4466          LA    R1,120          PRINT POSITIONS
002120 4700 41A0           021A0  4467 DUMPRI   BC    NOP,DUMP1R            DUMP SWITCH
002124 45F0 43F6           023F6  4468          BAL   15,GETEOR             SEARCH FOR RECORD MARK
002128 12DD                       4469          LTR   WR4,WR4               IS WR4=0
00212A 4780 41A6           021A6  4470          BC    8,MVBLAN              YES,BRANCH
00212E 19D1                       4471          CR    WR4,R1                NO,COMPARE WR4 AND R1
002130 4720 41A0           021A0  4472          BC    2,DUMP1R              HIGHER BRANCH
002134 42D0 428F           0228F  4473 DUMP2R   STC   WR4,PR1CCW+7           STORE COUNT IN CCW
002138 45F0 43A2           023A2  4474          BAL   15,VALOUT             *


00213C                            4476          CNOP  4,8                   *
00213C                            4477 PRNGO DS 0H    SVC   1                I/O REQUEST WAIT CHANNEL END
00213C 000E                       4478          DC    X'000E'                 CHANNEL END (FOR PRINTER)
00213E 0000
002140 00002288                   4479          DC    A(PR1CCW)             *
002144                            4480 PRNERR   DS    4C                    *
002148 0000000006000000           4481 PRNCSW   DC    X'0000000006000000'   *
002150                            4482 PRNPSW   DS    D                     *
002158 0000217E                   4483          DC    A(NRMPRN)             NORMAL RETURN (CE,THEN DE)
00215C 000021BC                   4484          DC    A(EXCPRN)             EXCEPT. RETURN
002160 92F0 4177      02177       4485          MVI   PRLOCK+1,X'F0'        SET WAIT SWITCH ON
002164 18BB                       4486          SR    WR2,WR2               WR2=0
002166 43B0 4288           02288  4487          IC    WR2,PR1CCW            INSERT COMMAND BYTE
00216A 41BB 0002           00002  4488          LA    WR2,2(WR2)            INCREMENT WR2 BY 2
00216E 45F0 42F8           022F8  4489          BAL   15,SPSKIP             BRANCH TO TEST SKIP
002172 9601 0007      00007       4490          OI    TAPE43,X'01'          SET PRINTER BUSY ON
002176 47F0 4176           02176  4491 PRLOCK   B       *                   ACCEPTED WAIT SWITCH
00217A 47F0 0262           00262  4492          B       TYPSWI+4            INTERROGATE I/O INDICATOR


00217E 9104 414C      0214C       4494 NRMPRN   TM    PRNCSW+4,X'04'        TEST DEVICE END
002182 5010 420C           0220C  4495          ST    R1,PRNSAV             SAVE R1                     V1L1
002186 4780 418E           0218E  4496          BZ      *+8                 NOT YET,BRANCH
00218A 94FE 0007      00007       4497 NRMPR4   NI    TAPE43,X'FE'          YES,SET BUSY INDICATOR 35 OFF
00218E 9200 4177      02177       4498          MVI   PRLOCK+1,X'00'        SET WAIT SWITCH OFF
002192 92FF 4150      02150       4499 NRMPR1   MVI   PRNPSW,X'FF'          ENABLE INTERRUPTIONS
002196 5810 420C           0220C  4500          L     R1,PRNSAV             RESTORE R1                  V1L1
00219A                            4501          CNOP  2,4                   *
                                  4502 *        SVC   3                     RETURN TO POINT OF INTERRUPTION
00219A 0000
00219C 00002150                   4503          DC    A(PRNPSW)             *
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
0021A0 18D1                       4504 DUMP1R   LR    WR4,R1                 WR4=R1
0021A2 47F0 4134            02134 4505          B       DUMP2R               BRANCH
0021A6 41D0 0001            00001 4506 MVBLAN   LA    WR4,R1                 WR4=R1
0021AA 9240 03A8      003A8       4507          MVI   IOBUFF,X'40'           SET BLANKS IN IOBUFF
0021AE D276 03A9 03A8 003A9 003A8 4508          MVC   IOBUFF+1(119),IOBUFF   *
0021B4 42D0 428F            0228F 4509          STC   WR4,PR1CCW+7           STORE COUNT
0021B8 47F0 413C            0213C 4510          B       PRNGO                GO TO WRITE

0021BC 5010 420C            0220C 4512 EXCPRN   ST    R1,PRNSAV              SAVE R1                      V1L1
0021C0 47F0 41F0            021F0 4513          B       PRNUE                U E,EXIT
0021C4 47F0 4204            02204 4514          B       PRNIR                INT.REQ.,GO TO SEND MESSAGE
                                  4515 *                    1403 PRINTER
0021C8 9131 4145      02145       4516          TM    PRNERR+1,X'31'         TEST SENSE BITS 2+3+7
0021CC 0000 0000            00000 4517          BZ      EXCR3                NONE,CATASTROPHIC DAMAGE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0021D0 4710 41FC            021FC 4518          BO      PXA2PR               ALL BRANCH
0021D4 9130 4145      02145       4519          TM    PRNERR+1,X'30'         TEST SENSE BITS 2+3
0021D8 4750 41FC            021FC 4520          BC    BOM,PXA2PR             SOME/ALL BRANCH
0021DC 9101 4145      02145       4521 PXA4PR   TM    PRNERR+1,X'01'         TEST SENSE BIT 7
0021E0 4780 41E8            021E8 4522          BZ      PXA5PR               BRANCH
0021E4 9604 0007      00007       4523          OI    TAPE43,X'04'           SET INDICATOR
0021E8 9101 414C      0214C       4524 PXA5PR   TM    PRNCSW+4,X'01'         TEST IF UE PRESENT
0021EC 4780 41F4            021F4 4525          BZ      NRMPR5               NO BRANCH
0021F0 9602 0007      00007       4526 PRNUE    OI    TAPE43,X'02'           SET INDICATOR UE IS PRESENT
0021F4 4510 07B0            007B0 4527 NRMPR5   BAL   R1,ENTRY3              GO TO INTERROGATE INDICATOR
0021F8 47F0 418A            0218A 4528          B       NRMPR4               RETURN
0021FC 9608 0002      00002       4529 PXA2PR   OI    NYCHEK,X'08'           SET PRINTER CHECK ON
002200 47F0 41DC            021DC 4530          B       PXA4PR               BRANCH

002204 45F0 00DC            000DC 4532 PRNIR    BAL   15,MESTIR              SEND MESSAGE 'INTERVENTION RE-
002208 00000121                   4533          DC  A(INTREQ)                 QUIRED ON PRINTER AND STOP
00220C                            4534 PRNSAV   DS    F                                                  V1L1

002210 DC0040                     4536 TBPRN    DC    X'DC0040'              TABLE FOR PRINT NUMERICALLY
002213 FC0040                     4537          DC    X'FC0040'              *
002216 FA00AA                     4538          DC    X'FA00AA'              *
002219 FF00AA                     4539          DC    X'FF00AA'              *
00221C DA00AA                     4540          DC    X'DA00AA'              *
00221F DF00AA                     4541          DC    X'DF00AA'              *
002222 EF                         4542          DC    X'EF'                  *

                                  4544 ************************************************************************
                                  4545 *                                                                      *
                                  4546 *                        PRINT ALPHAMERICALLY                          *
                                  4547 *                                                                      *
                                  4548 *  THIS SEQUENCE PREPARES THE ADDRESSES FOR CODE CONVERSION SO  THAT  *
                                  4549 *  THEY CAN BE USED BY THE 'MASK' SUBROUTINE. IT SETS THE FIELD COM-   *
                                  4550 *  PRESSION SWITCH IN THE 'VALOUT' SUBROUTINE ON. WR3 IS LOADED WITH   *
                                  4551 *  1 WHICH CORRESPONDS TO A STEP OF TWO. THIS ALLOWS CONDENSING  TWO  *
                                  4552 *  ALPHAMERIC BYTES INTO ONE SINGLE OUTPUT CODE. THEN,  CONTROL   IS  *
                                  4553 *  GIVEN TO THE 'WRITE NUMERICALLY' (CARD) SEQUENCE.                   *
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                              ASM H V 05 22.10 05/07/81

                                   4554 *                                                                       *
                                   4555 ************************************************************************


002223 00
002224 41B0 0623            00623  4557 PRA      LA    WR2,TABLE3              LOAD ADDRESSES OF CODE CON-
002228 41A0 4240            02240  4558          LA    WR1,TBPRA               VERSION TABLE
00222C 41C0 0001            00001  4559          LA    WR3,1                   *
002230 9200 4121      02121        4560          MVI   DUMPRI+1,X'00'          SET DUMP SWITCH OFF
002234 9259 0511      00511        4561          MVI   MASK+1,X'59'            SET COMMAND MASK = '59'
002238 92F0 43A9      023A9        4562          MVI   FLCOSW+1,X'F0'          SET COMPRESSION SWITCH ON
00223C 47F0 40E4            020E4  4563          B       PRNB                  *                           V1L2
                                   4564 *        B       SPSKBT                *                 (DELETED) V1L2


002240 0A00AA                      4566 TBPRA    DC    X'0A00AA'               *
002243 0F01AA4E                    4567          DC    X'0F01AA4E'             *
002247 1F00AA                      4568          DC    X'1F00AA'               *
00224A 500060                      4569          DC    X'500060'               *
00224D 33007E                      4570          DC    X'33007E'
002250 24004D                      4571          DC    X'24004D'
002253 04005D                      4572          DC    X'04005D'
002256 EF                          4573          DC    X'EF'                   *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4575 ************************************************************************
                                  4576 *                                                                      *
                                  4577 *                     DUMP NUMERICALLY ON PRINTER                      *
                                  4578 *                                                                      *
                                  4579 *                                                                      *
                                  4580 *  THE CODE CONVERSION ADDRESSES ARE LOADED FOR USE  BY  THE  'MASK'   *
                                  4581 *  SUBROUTINE.  THE FIELD COMPRESSION SWITCH IN THE 'VALOUT' SUBROU-   *
                                  4582 *  TINE IS SET OFF. WR3 IS LOADED WITH ZERO  FOR NUMERIC MODE OPERA-   *
                                  4583 *  TION.                                                               *
                                  4584 *                                                                      *
                                  4585 *  THE DUMP SWITCH IN THE 'PRINT NUMERICALLY' SEQUENCE IS SET ON  TO   *
                                  4586 *  BY-PASS THE SEARCH FOR RECORD MARK ('GETEOR' SUBROUTINE).           *
                                  4587 *                                                                      *
                                  4588 ************************************************************************

002257 00
002258 41B0 058A          0058A   4590 PRD      LA    WR2,TABLE1              LOAD ADDRESSES OF CODE CON-
00225C 41A0 4270          02270   4591          LA    WR1,TBPRD               VERSION TABLE
002260 41C0 0000          00000   4592          LA    WR3,0                   *
002264 92F0 4121    02121         4593          MVI   DUMPRI+1,X'F0'          SET DUMP SWITCH ON
002268 9299 0511    00511         4594          MVI   MASK+1,X'99'            SET COMMAND MASK  = 99
00226C 47F0 40E0          020E0   4595          B       PRNA                  *                          V1L2
                                  4596 *        B       SPSKBT-4              *                (DELETED) V1L2

002270 DA05E6AA5CAAAAE7           4598 TBPRD    DC    X'DA05E6AA5CAAAAE7'                              V1L2
                                  4599 *TBPRD   DC    X'DA05E6AA5CAAAAE6'                    (DELETED) V1L2
002278 FA057BAA7CAAAAC7           4600          DC    X'FA057BAA7CAAAAC7'                              V1L2
                                  4601 *        DC    X'FA057BAA7CAAAA7B'                    (DELETED) V1L2
002280 EF                         4602          DC    X'EF'                   *

002281 00000000000000
002288 000003AB20000000           4604 PR1CCW   CCW   X'00',IOBUFF,X'20',0    CCW FOR PRINTER
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4606 ********************************************************************************
                                  4607 *                                                                              *
                                  4608 *                        CONTROL OPERATIONS ON PRINTER                         *
                                  4609 *                                                                              *
                                  4610 *  THIS SEQUENCE PERFORMS TWO DIFFERENT TYPES OF PRINTER OPERATIONS=           *
                                  4611 *  IMMEDIATE COMMANDS AND DELAYED COMMANDS.                                    *
                                  4612 *                                                                              *
                                  4613 *  DIGIT Q10 IS FIRST TESTED TO DEFINE THE TYPE OF COMMAND =                   *
                                  4614 *  1) IMMEDIATE COMMANDS                                                       *
                                  4615 *                                                                              *
                                  4616 *  DIGIT Q11 IS USED TO GET A COMMAND BYTE DIRECTLY FROM A TABLE.THE           *
                                  4617 *  COMMAND BYTE IS PLACED INTO THE CCW OF THE OUTPUT OPERATION TO BE           *
                                  4618 *  PERFORMED ON THE PRINTER. THIS OPERATION IS REQUESTED BY AN SVC 2           *
                                  4619 *  WHICH HAS TWO EXITS (ACCEPTED AND NORMAL). BOTH CAUSE  SIMULATION           *
                                  4620 *  TO BE CONTINUED.                                                            *
                                  4621 *                                                                              *
                                  4622 *  IN CASE OF AN EXCEPTIONAL RETURN, THE 'EXCRET' SUBROUTINE DISTIN-           *
                                  4623 *  GUISHES BETWEEN THE ERROR INDICATIONS 'UNIT EXCEPTION' AND 'INTER-          *
                                  4624 *  VENTION REQUIRED' (SEE UNDER 'PRINT NUMERICALLY') ON THE ONE HAND           *
                                  4625 *  AND 'UNIT CHECK' ON THE OTHER HAND. THIS LATTER SENDS THE MESSAGE           *
                                  4626 *  'UNIT CHECK' TO THE 1052 AND STOPS SIMULATION.                              *
                                  4627 *                                                                              *
                                  4628 *  2) DELAYED COMMANDS                                                         *
                                  4629 *  A SPECIAL WRITE COMMAND IS COMPUTED FROM Q10 AND Q11 AND  DIRECT-           *
                                  4630 *  LY STORED IN THE 'PRINT NUMERICALLY' SEQUENCE FOR FURTHER USE. IT           *
                                  4631 *  WILL BE PROCESSED AS A COMMAND  MODIFIER DURING THE NEXT   'PRINT           *
                                  4632 *  NUMERICALLY/ALPHAMERICALLY' OPERATION.                                      *
                                  4633 *                                                                              *
                                  4634 ********************************************************************************

002290 18A0                       4637 KPRINT   LR    WR1,0                 CLEAR WR1
002292 18B0                       4638          LR    WR2,0                 CLEAR WR2
002294 D101 4370 500A 02370 0000A 4639          MVN   Q10(2),10(CNTR)       MOVE Q10 AND Q11 NUMER. TO BUFF.
00229A 43A0 4370            02370 4640          IC    WR1,Q10               INSERT Q10 NUMERIC AND USE IT AS
00229E 43AA 4372            02372 4641          IC    WR1,TABQ10(WR1)       INDEX FOR COMMAND ADDRESS
0022A2 41AA 437A            0237A 4642          LA    WR1,TABIMM(WR1)       *
0022A6 43B0 4371            02371 4643          IC    WR2,Q11               INSERT Q11 NUMERIC
0022AA 43BA B000            00000 4644          IC    WR2,0(WR1,WR2)        GET COMMAND BYTE
0022AE 9101 4370      02370       4645          TM    Q10,X'01'             TEST IF Q10 IS ODD
0022B2 4710 42C6            022C6 4646          BO       KPR1               YES, BRANCH TO IMMEDIATE COMMAND
0022B6 06B0                       4647          BCTR  WR2,0                 NO, DELAYED COMMAND, DECREMEN BY
0022B8 06B0                       4648          BCTR  WR2,0                 2 TO OBTAIN WRITE COMMAND
0022BA 42B0 40F1            020F1 4649          STC   WR2,SPSKBT+5          AND STORE IT FOR FURTHER USE.
0022BE 9200 40ED      020ED       4650          MVI   SPSKBT+1,X'00'        SET SPACE SKIP SWITCH ON
0022C2 47F0 07F6            007F6 4651          B       ENTRY1              RETURN TO INTERPRETIVE LOOP.
0022C6 42B0 4368            02368 4652 KPR1     STC   WR2,KPRCCW           STORE IMMEDIATE COMMAND IN CCW
0022CA 0700                       4653          CNOP  4,8                   *
0022CC                            4654 KPRGO DS 0H SVC   2                     I/O REQUEST AND CONTINUE
0022CC 000E                       4655          DC    X'000E'                (FOR PRINTER)
0022CE 0000
0022D0 00002368                   4656          DC    A(KPRCCW)             *
0022D4                            4657 KPRERR   DS    4C                    *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
0022D8 0000000006000000           4658 KPRCSW   DC    X'0000000006000000'     *
0022E0                            4659 KPRPSW   DS    D                       *
0022E8 00002328                   4660          DC    A(NRMKPR)               *
0022EC 00002334                   4661          DC    A(EXCKPR)               *
0022F0 45F0 42F8          022F8   4662          BAL   15,SPSKIP               GO TO TEST SKIP
0022F4 47F0 07F6          007F6   4663          B       ENTRY1                RETURN TO INTERPRETIVE LOOP

0022F8 1BCC                       4665 SPSKIP   SR    WR3,WR3                 WR3=0
0022FA 43C0 437F          0237F   4666          IC    WR3,SKIP1               INSERT SKIP TO CHAN 1
0022FE 19CB                       4667          CR    WR3,WR2                 TEST IF CHAN 1
002300 4770 430A          0230A   4668          BC    7,SKIP9A                NO,BRANCH
002304 94F9 0007    00007         4669          NI    TAPE43,X'F9'            RESET INDICATORS 33 AND 34
002308 07FF                       4670          BR      15                    RETURN TO CALLER
00230A 43C0 4387          02387   4671 SKIP9A   IC    WR3,SKIP9               INSERT CHAN 9
00230E 19CB                       4672          CR    WR3,WR2                 TEST IF CHANN 9
002310 4770 431A          0231A   4673          BC    7,SKIP1A                NO,BRANCH
002314 9604 0007    00007         4674          OI    TAPE43,X'04'            SET INDICATOR ON
002318 07FF                       4675          BR      15                    RETURN TO CALLER
00231A 43C0 438C          0238C   4676 SKIP1A   IC    WR3,SKIP12              INSERT CHAN 12
00231E 19CB                       4677          CR    WR3,WR2                 TEST IF CHAN 12
002320 077F                       4678          BCR   7,15                    NO,RETURN TO CALLER
002322 9602 0007    00007         4679          OI    TAPE43,X'02'            SET INDICATOR ON
002326 07FF                       4680          BR      15                    RETURN TO CALLER



002328 92FF 42E0    022E0         4683 NRMKPR   MVI   KPRPSW,X'FF'            ENABLE INTERRUPTIONS
00232C 0700                       4684          CNOP  2,4                     *
                                  4685 *        SVC   3                       RETURN TO POINT OF INTERRUPTION
00232E 0000
002330 000022E0                   4686          DC    A(KPRPSW)               *

002334 5010 034C          0034C   4688 EXCKPR   ST    R1,RNCSAV               SAVE R1                      V1L2
002338 47F0 4358          02358   4689          B       KPRUE                 U E, GO TO SET CHAN.12 INDIC ON
00233C 47F0 434C          0234C   4690          B       KPRIR                 INT.REQ.,GO TO SEND MESSAGE
002340 5810 034C          0034C   4691          L     R1,RNCSAV               RESTORE R1                   V1L2
002344 45F0 00F4          000F4   4692          BAL   15,MESTOP               SEND MESSAGE 'UNIT CHECK'
002348 0000016B                   4693          DC  A(UNTCHK)               AND STOP SIMULATION

00234C 5810 034C          0034C   4695 KPRIR    L     R1,RNCSAV               RESTORE R1                   V1L2
002350 45F0 00DC          000DC   4696          BAL   15,MESTIR               MESSAGE 'INTERV. REQUIRED'   V1L2
                                  4697 *KPRIR   BAL   15,MESTIR               MESSAGE 'INTERV. R(DELETED) V1L2
002354 00000121                   4698          DC  A(INTREQ)               ON PRINTER AND STOP
002358 5810 034C          0034C   4699 KPRUE    L     R1,RNCSAV               RESTORE R1                   V1L2
00235C 9602 0007    00007         4700          OI    TAPE43,X'02'            SET CHANNEL 12 INDICATOR ON V1L2
                                  4701 *KPRUE   OI    TAPE43,X'02'            SET CHANNEL 12 IND(DELETED) V1L2
002360 47F0 4328          02328   4702          B       NRMKPR                AND RETURN


002364 00000000
002368 0000236800000001           4704 KPRCCW   CCW   X'00',*,0,1             CONTROL CCW
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81


002370 00                          4706 Q10      DC    X'00'                 *
002371 00                          4707 Q11      DC    X'00'                 *


002372 0E00000E04000004            4709 TABQ10   DC    X'0E00000E04000004'   *

00237A 000B131BD38B939B            4711 TABIMM   DC    X'000B131BD38B939BA3AB' *
002384 B3BBC3CB000000DB            4712          DC    X'B3BBC3CB000000DBE3'   *
                         0237F     4713 SKIP1    EQU   TABIMM+5              *
                         02387     4714 SKIP9    EQU   TABIMM+13             *
                         0238C     4715 SKIP12   EQU   TABIMM+18             *
                                   4716          AIF   (NOT &R1621R).NOPT4    PAPER TAPE      READER
                                   4717 .NOPT4   ANOP
                         00A00     4718 RAPT     EQU   CODERR                 PAPER TAPE READER NOT EXISTING
                         00A00     4719 RNPT     EQU   CODERR                 *
                                   4720 .PTRX1   ANOP
                                   4721          AIF   (NOT &R1621P).NOPT5         PAPER TAPE PUNCH
                                   4722 .NOPT5   ANOP                              COMMON PART OF SIM20
                         00A00     4723 WNPT     EQU   CODERR                 PAPER TAPE DOES NOT EXIST
                         00A00     4724 WAPT     EQU   CODERR                 *
                         00A00     4725 PTIOR    EQU   CODERR                 *
                         00A00     4726 DNPT     EQU   CODERR                 *
                         00A00     4727 PTI01    EQU   CODERR                 *
                                   4728 .PTPX1   ANOP
00238D                             4729 PTSTAT   DS    20C                   ERROR STATUS
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4731 ************************************************************************
                                  4732 *                                                                      *
                                  4733 *                          'VALOUT' SUBROUTINE                         *
                                  4734 *                                                                      *
                                  4735 *  THIS SUBROUTINE IS ENTERED FROM ALL OUTPUT OPERATIONS AND PERFORMS  *
                                  4736 *  THE INVERSE FUNCTIONS OF THE 'VALIN' SUBROUTINE.                    *
                                  4737 *                                                                      *
                                  4738 *  OPERATION                                                           *
                                  4739 *  'VALOUT' CONVERTS SUCCESSIVELY ALL BYTES INTENDED FOR AN   OUTPUT   *
                                  4740 *  OPERATION AND PLACES THEM INTO 'IOBUFF'.   BYTE  CONVERSION  USES   *
                                  4741 *  CODE CONVERSION TABLE 'CNVTB' LOCATED AT ABSOLUTE ADDRESS 128. IT   *
                                  4742 *  INCLUDES A VALIDITY CHECK.                                          *
                                  4743 *                                                                      *
                                  4744 *  'VALOUT' MAY BE ENTERED FROM A 'WRITE NUMERICALLY' OR 'DUMP'  OP-   *
                                  4745 *  ERATION. IN THIS CASE, THE FIELD COMPRESSION SWITCH MUST BE  OFF,   *
                                  4746 *  AND THE CONTENTS OF WR3 MUST BE ZERO TO INCREMENT THE  P  ADDRESS   *
                                  4747 *  BY A STEP OF 1.                                                     *
                                  4748 *                                                                      *
                                  4749 *  IT MAY ALSO BE ENTERED  FROM A 'WRITE ALPHAMERICALLY'  OPERATION.   *
                                  4750 *  IN THIS CASE,  THE CONTENTS OF WR3 MUST BE 1 TO INCREMENT THE   P   *
                                  4751 *  ADDRESS BY A STEP OF TWO. THE FIELD COMPRESSION SWITCH MUST BE ON   *
                                  4752 *  SO AS TO ALLOW PACKING OF TWO DATA BYTES INTO ONE SINGLE   OUTPUT   *
                                  4753 *  CHARACTER.                                                          *
                                  4754 *                                                                      *
                                  4755 *  THE WRAP-AROUND FEATURE IS INCLUDED IN THIS SUBROUTINE.             *
                                  4756 *                                                                      *
                                  4757 *  WHEN AN INVALID ERROR CODE IS DETECTED, ERROR EXIT IS AS IN   THE   *
                                  4758 *  'VALIN' SUBROUTINE.                                                 *
                                  4759 *                                                                      *
                                  4760 ************************************************************************

0023A1 00
0023A2 41A0 03A8           003A8  4762 VALOUT   LA    WR1,IOBUFF             LOAD ADDRESS OF IOBUFF
0023A6 18B0                       4763         LR    WR2,0                  *
0023A8 4700 43D4           023D4  4764 FLCOSW   BC    NOP,PACK1              FIELD COMPRESSION SWITCH
0023AC 43B8 0000           00000  4765         IC    WR2,0(RP)              INSERT CURRENT CHARACTER
0023B0 43BB 0080           00080  4766 PACK2   IC    WR2,CONVTB(WR2)        TRANSLATE CURRENT CODE
0023B4 42BA 0000           00000  4767         STC   WR2,0(WR1)             STORE DATA
0023B8 95AA A000      00000        4768         CLI   0(WR1),X'AA'           IS DATA INVALID
0023BC 4780 43EE           023EE  4769         BE      CODINA               YES,BRANCH
0023C0 418C 8001           00001  4770 PACK3   LA    RP,1(WR3,RP)           NO,RP = RP + 1 (OR + 2)
0023C4 1987                       4771         CR    RP,SIZE                IS RP GREATER THAN SIZE
0023C6 4720 43E8           023E8  4772         BH      VALOU1               YES,BRANCH
0023CA 41AA 0001           00001  4773 VALOU2  LA    WR1,1(WR1)             NO,WR1=WR1+1
0023CE 46D0 43A8           023A8  4774         BCT   WR4,FLCOSW             LOOP IF COUNT NOT EQUAL TO 0
0023D2 07FF                       4775         BR      15                   RETURN TO CALLER
0023D4 F211 04F0 8000 004F0 00000 4776 PACK1   PACK  BUFF2(2),0(2,RP)       PACK 2 BYTES IN 1 BYTE
0023DA F111 04F0 04F0 004F0 004F0 4777         MVO   BUFF2(2),BUFF2(2)      *
0023E0 43B0 04F0           004F0  4778         IC    WR2,BUFF2              INSERT RESULT IN WR2
0023E4 47F0 43B0           023B0  4779         B       PACK2                *
0023E8 1886                       4780 VALOU1  LR    RP,MAPORG              RP=MAPORG
                                  4781         AIF   (NOT &R1621P).NOPT6  PAPER TAPE PUNCH
                                  4782 .NOPT6  ANOP                                                    V1L1
0023EA 47F0 43CA           023CA  4783         B       VALOU2                                         V1L1
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

0023EE 9640 0002      00002        4784 CODINA   OI    NYCHEK,X'40'          SET WRITE CHECK INDICATOR ON
0023F2 47F0 43C0            023C0  4785          B       PACK3               RETURN
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4787 ************************************************************************
                                  4788 *                                                                      *
                                  4789 *                          'GETEOR' SUBROUTINE                         *
                                  4790 *                                                                      *
                                  4791 *  THIS SUBROUTINE IS CALLED BY OUTPUT OPERATIONS  WHICH REQUIRE RE-   *
                                  4792 *  CORD MARK SEARCHING. STARTING AT THE P ADDRESS, CORE STORAGE   IS   *
                                  4793 *  SCANNED FROM LEFT TO RIGHT.  WHEN A RECORD MARK HAS BEEN   FOUND,   *
                                  4794 *  THE DIFFERENCE OF ITS ADDRESS AND THE INITIAL P ADDRESS  IS  COM-   *
                                  4795 *  PUTED.                                                              *
                                  4796 *                                                                      *
                                  4797 *  OPERATION IN NUMERIC OR ALPHAMERIC  MODE  REQUIRES  INCREMENTING.   *
                                  4798 *  THE STEP MAY BE 1 OR 2, DEPENDING ON THE VALUE GIVEN TO  REGISTER   *
                                  4799 *  WR3 BY THE CALLING SEQUENCE (1 FOR 'WRITE' OR 'DUMP NUMERICALLY',   *
                                  4800 *  2 FOR 'WRITE ALPHAMERICALLY').  THE RESULTING BYTE COUNT IS  CON-   *
                                  4801 *  TAINED IN REGISTER WR4 FOR SUBSEQUENT USE BY THE OUTPUT OPERATION.  *
                                  4802 *                                                                      *
                                  4803 *  THIS SUBROUTINE INCLUDES THE WRAP-AROUND FEATURE.                   *
                                  4804 *                                                                      *
                                  4805 *  SPECIAL CASE - WHEN A RECORD MARK IS FOUND AT THE P ADDRESS,  THE   *
                                  4806 *  'GETEOR' SUBROUTINE SENDS THE MESSAGE 'CHECK STOP' AND SIMULATION   *
                                  4807 *  STOPS.                                                              *
                                  4808 *                                                                      *
                                  4809 ************************************************************************

0023F6 41DC 8000          00000   4811 GETEOR   LA    WR4,0(WR3,RP)          INITIALIZE WR4 = P OR P+1
0023FA 18E1                       4812          LR    WR5,R1                 WR5 USED IF NO RM OR GM      V1L1
0023FC 1820                       4813          LR    R2,0                   *
                                  4814 *        TM    0(WR4),X'0A'           TEST IF RECORD MA (DELETED) V1L1
0023FE 910A D000    00000         4815 GETEOM   TM    0(WR4),X'0A'           TEST IF RM OR GM             V1L1
002402 4710 4426          02426   4816          BO      EOR1                 YES,BRANCH
002406 41DC D001          00001   4817          LA    WR4,1(WR3,WR4)         NO,WR4=WR4+1+WR3
00240A 19D7                       4818          CR    WR4,SIZE               IS WR4 GREATER THAN SIZE
00240C 4720 4418          02418   4819          BH      EOR2                 YES,BRANCH
                                  4820 *        B       GETEOR+6                                (DELETED) V1L1
                                  4821 * EOR2   BCTR  WR4,0                  NO,WR4=WR4-2 OR -1(DELETED) V1L1
002410 46E0 43FE          023FE   4822          BCT   WR5,GETEOM                                       V1L1
002414 18D1                       4823          LR    WR4,R1                 MAX BUFF TO BE PRINTED       V1L1
002416 07FF                       4824          BR      15                   RETURN TO CALLER             V1L1
002418 1BDC                       4825 EOR2     SR    WR4,WR3                                             V1L1
00241A 182D                       4826          LR    R2,WR4                 *
00241C 1B28                       4827          SR    R2,RP                  R2=WR4-RP
00241E 41DC 6000          00000   4828          LA    WR4,0(WR3,MAPORG)      WR4=A(MAPORG+WR3)
                                  4829 *        B       GETEOR+6             BRANCH             (DELETED) V1L1
002422 47F0 43FE          023FE   4830          B       GETEOM                                     V1L1
002426 19D8                       4831 EOR1     CR    WR4,RP                 IS WR4 GREATER THAN RP
002428 4720 444E          0244E   4832          BH      EOR3                 YES,BRANCH
00242C 4780 443E          0243E   4833          BE      EOR5                 NO,EQUAL(ERROR)
002430 1BD6                       4834          SR    WR4,MAPORG             WR4=WR4-MAPORG
002432 1AD2                       4835          AR    WR4,R2                 WR4=WR4+R2
002434 42C0 443B          0243B   4836 EOR4     STC   WR3,*+7                *
002438 8AD0 0000          00000   4837          SRA   WR4,0                  COMPUTE WR4=L
00243C 0000                       4838          BCR   BP,15                  IF POSITIVE, NORMAL EXIT
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV029  *** ERROR ***  INCORRECT REGISTER OR MASK SPECIFICATION
```

```
00243E 9109 5009      00009         4839 EOR5     TM    9(CNTR),X'09'      IS 9(CNTR) = 09
002442 4710 444A            0244A   4840          BO       EOR6A           YES,BRANCH
002446 47F0 09F0            009F0   4841          B        ERRORP          NO,SEND MESSAGE 'MAR CHK'
00244A 1BDD                         4842 EOR6A    SR    WR4,WR4            SET WR4 = 0
00244C 07FF                         4843          BR       15              RETURN TO CALLER
00244E 1BD8                         4844 EOR3     SR    WR4,RP             WR4=WR4-RP
002450 47F0 4434            02434   4845          B        EOR4            BRANCH
                                    4846          PRINT OFF
                                    4848          PRINT ON
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                  4850 *** DISKV                                    DISK VERSION
                           06000  4851 ADDFQ    EQU   X'6000'
                                  4852 *-*      END
                                  4853 *** 1620/0                                   COMMON  PART OF  SIM20
                           01E44  4854 ADDFQ    EQU   RATY
  IEV043  *** ERROR ***  PREVIOUSLY DEFINED SYMBOL
                                  4855 *-*      END
                                  4856 * 1622P                                     1622 CARD  PUNCH
                           01FE6  4857 WAA      EQU   WNCD2+2
                           001A2  4858 WAA1     EQU   WAA-RATY
                           061A2  4859 WAA2     EQU   WAA1+ADDFQ
002454 000061A2                   4860 WNCAP    DC  A(WAA2)
                                  4861 * 1620C                                     1052 TYPEWRITER
                           00072  4862 WNAA     EQU   WNTY3-RATY
                           06072  4863 WNAA1    EQU   WNAA+ADDFQ
002458 00006072                   4864 WCONS    DC  A(WNAA1)
                                  4865          AIF   (NOT &R1621P).NOPT6A     PAPER TAPE PUNCH
                                  4866 .NOPT6A  ANOP
                                  4867 * 1443W                                     1443 PRINTER
                           0213E  4868 PBB      EQU   PRNGO+2
                           002FA  4869 PBB1     EQU   PBB-RATY
                           062FA  4870 PBB2     EQU   PBB1+ADDFQ
00245C 000062FA                   4871 PNAP1    DC  A(PBB2)
                           022CE  4872 PNBB     EQU   KPRGO+2
                           0048A  4873 PNBB1    EQU   PNBB-RATY
                           0648A  4874 PNBB2    EQU   PNBB1+ADDFQ
002460 0000648A                   4875 PNAP2    DC  A(PNBB2)
                                  4876          AIF   (NOT &R1621R).NOPT7      PAPER TAPE READER
                                  4877 .NOPT7   ANOP
                                  4878 *** DISKV    'DISK VERSION
                                  4879 *        PRINT OFF
                                  4880          PUNCH ' SLC  006E00'
                                  4881          PRINT ON
002464                     01E44  4882          ORG   RATY
                                  4883 *-*      END
                                  4884 *** NISKV    NO    'DISKV'                   VERSION
001E44                     03000  4885          ORG   KEYBIT+12288
                           03000  4886          USING *,SIMB2
                                  4887 *-*      END
                                  4888          AIF   (NOT &DISK).NODSK8A
                                  4889 .NODSK8A ANOP
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
003000 0700                       4891          CNOP  2,4
003002 00                         4892 CORES1   DC    X'00'                 BEGINNING OF 1620 CORE STORAGE
                                  4893          AIF   (NOT &DISK).NODSK9
                                  4894 .NODSK9  ANOP
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81

003003                     04000  4896           ORG   KEYBIT+16384
                           04000  4897           USING *,8
                           05000  4898           USING *+4096,9
004000 4180 0800           00800  4899 DSKINT    LA    8,2048                LOAD BASE REGISTERS
004004 1A88                       4900           AR    8,8                   *
004006 1858                       4901           LR    5,8                   *
004008 1A88                       4902           AR    8,8                   *
00400A 1A88                       4903           AR    8,8                   *
00400C 4198 5000           00000  4904           LA    9,0(8,5)              *
004010 1B00                       4905           SR    0,0
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                            ASM H V 05 22.10 05/07/81

                                   4908 ************************************************************************
                                   4909 *                                                                      *
                                   4910 *                                                                      *
                                   4911 *                          BEGINNING OF SIMULATION                     *
                                   4912 *                                                                      *
                                   4913 *                                                                      *
                                   4914 ************************************************************************

004012 4130 0800             00800 4916 BEGIN    LA    SIMB1,2048
004016 1A33                        4917          AR    SIMB1,SIMB1            SIMB1 BASE = 4096
004018 4143 3000             00000 4918          LA    SIMB2,0(SIMB1,SIMB1)   SIMB2 BASE = 8192

                                   4920 *      ENABLE I/O INTERRUPTS
                                   4921 *

                                   4923 *      SET INPUT COMMAND RETURN
                                   4924 *
00401C 41A0 0008             00008 4925          LA    WR1,SATT
004020 0000 0000             00000 4926          ST    WR1,REQUST
   IEV044  *** ERROR ***  UNDEFINED SYMBOL

                                   4928 *      SIM-20 INITIALIZATION
                                   4929 *
004024 1B00                        4930          SR    0,0                    CLEAR REGISTER 0
004026 1810                        4931          LR    R1,0
004028 1620                        4932          LR    R2,0
00402A 4860 0014             00014 4933          LH    MAPORG,COREST          LOAD MAPORG REGISTER
00402E 5870 0018             00018 4934          L     SIZE,CAPACT            LOAD SIZE = MAXIMUM 1620 SIZE
004032 1A76                        4935          AR    SIZE,MAPORG
004034 18A0                        4936          LR    WR1,0
004036 1856                        4937          LR    CNTR,MAPORG            COUNTER = ADDRESS 00000
004038 18F0                        4938          LR    WR6,0
00403A 9200 0001       00001       4939          MVI   KEYBIT+1,X'00'         RESET SWITCHES
00403E 9294 0000       00000       4940          MVI   KEYBIT,X'94'           SET RESET AND MODIFY BITS
004042 9204 0006       00006       4941          MVI   IXBAND,X'04'           RESET TO 'NOBAND'          V1L2
004046 4700 000C             0000C 4942 BRCC2    BC    NOP,BRCC               SWITCH FOR BEGINNING OF SIM20
                                   4943 *+* DISKV    DISKV VERSION
00404A 41A0 3E44             01E44 4944          LA    WR1,RATY
00404E 48C0 0014             00014 4945          LH    WR3,COREST
004052 1BCA                        4946          SR    WR3,WR1                SET DATA LENGTH
004054 0000 0000             00000 4947          STH   WR3,CCWRD+6            STORE COUNT FOR FURTHER USE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004058 41B0 0100             00100 4948          LA    WR2,256
00405C 0000 0000             00000 4949          LH    WR4,IOKPR
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004060 19CB                        4950 VVV2     CR    WR3,WR2                IS LENGTH BELOW THAN 256
004062 4740 807A             0407A 4951          BL      VVV1
004066 D2FF A000 D000 00000 00000  4952          MVC   0(256,WR1),0(WR4)      MOVE
00406C 41AA 0100             00100 4953          LA    WR1,256(WR1)           *
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
004070 41DD 0100            00100  4954          LA    WR4,256(WR4)
004074 18CB                        4955          SR    WR3,WR2                 DECREMENT WR3 BY 256
004076 47F0 8060            04060  4956          B       VVV2
00407A 42C0 807F            0407F  4957 VVV1     STC   WR3,VV4+1
00407E D200 A000 D000 00000 00000  4958 VV4      MVC   0(0,WR1),0(WR4)          LAST MOVE
                                   4959 *-*      END
                                   4960          AIF   (NOT &DISK).NODSK10
                                   4961 .NODSK10 ANOP
                                   4962 *** DISKV    DISKV VERSION
004084 4110 0000            00000  4963          LA    R1,0                    RETURN 4
004088 0000 0000 0000 00000 00000  4964          MVC   WHDL(2),CCWRD+6         MOVE DATA LENGTH
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00408E 0000 0000              00000  4965          LH    WR3,CCWRD+6             WR3=DL
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004092 41CC 0008              00008  4966          LA    WR3,8(WR3)              WR3=DL+8
004096 0000 0000              00000  4967          STH   WR3,CCWRIT+6            STORE IT IN CCW FOR WRITE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00409A 0000 0000              00000  4968 OTHE2    ST    R1,REG1                 SAVE REGISTER R1
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00409E 0000 0000 0000 00000 00000  4969          MVC   CCWRIT+2(2),IOKPR       PLACE I/O DEVICE ADDRESS OF
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040A4 0000 0000              00000  4970          LH    WR1,WHVDSK(R1)          2311 UNIT IN WHDEV
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040A8 0000 0000              00000  4971          STH   WR1,WHDEV               *
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040AC 0000 0000               00000         4972          MVI   WHHH+1,X'00'            HEAD NUMBER = 0
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040B0 0000 0000               00000         4973          MVI   WHCC+1,X'01'            CYLINDER  NUMBER =01
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040B4 0000 0000               00000         4974          MVI   WHRO+2,X'01'            RECORD 0 = CYLINDER 01
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040B8 0000 0000               00000         4975          MVI   WHRO+4,X'00'            HEAD  0
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040BC 0000 0000              00000  4976 A9CD     LA    WR5,CCWINI              PLACE HOME ADDRESS AND RECORD 0
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040C0 0000 0000              00000  4977          ST    WR5,WHDEV+2             IN CCW ADDRESS
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040C4 0000 0000              00000  4978          LA    WR5,CCWIST              PLACE RESTORE COMMAND IN
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040C8 0000 0000              00000  4979          STH   WR5,WHESTC              'WHESTC' BUFFER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040CC 0000 0000              00000  4980          BAL   15,WHRW                 GO TO 'IWHW' SUBROUTINE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040D0 0000 0000              00000  4981          B       WHSKEW                GO TO DISK ERROR ANALYSIS ROUT.
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040D4 0000 0000              00000  4982          LA    WR5,CCWLD               STORE CCW ADDRESS IN IORW
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040D8 0000 0000              00000  4983          ST    WR5,WHDEV+2             *
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040DC 0000 0000              00000  4984          LA    WR5,CCWLDR              PLACE RESTORE COMMAND IN
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040E0 0000 0000              00000  4985          STH   WR5,WHESTC              'WHESTC' BUFFER
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
```

```
 LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                    ASM H V 05 22.10 05/07/81
0040E4 0000 0000            00000  4986          LH    WR2,CCWRIT+2
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040E8 41A0 0008            00008  4987          LA    WR1,8                 WR1 = 8
0040EC 1BBA                        4988          SR    WR2,WR1               WR2 = WR2 - WR1
0040EE 0000 0000            00000  4989          STH   WR2,CCWRIT+2          STORE WR2 IN CCW
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040F2 0000 0000 0000 00000 00000  4990          MVC   0(8,WR2),WHEK+2       MOVE DATA ADDRESS TO CCW
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040F8 0000 0000            00000  4991          BAL   15,WHRW               GO TO 'WHRW' SUBROUTINE
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
0040FC 0000 0000            00000  4992          B       WHSKEW              GO TO DISK ERROR ANALYSIS ROUT.
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004100 41BB 0008            00008  4993          LA    WR2,8(WR2)            INCREMENT REGISTER WR2 BY 8
004104 0000 0000            00000  4994          STH   WR2,CCWRIT+2          STORE WR2 IN CCW
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004108 1BBB                        4995          SR    WR2,WR2               CLEAR WR2
00410A 0000 0000            00000  4996          IC    WR2,WHHH+1            INCREMENT HEAD NUMBER BY 1
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00410E 41A0 000A            0000A  4997          LA    WR1,10                INCRMENT WR1 BY 10
004112 41BB 0001            00001  4998          LA    WR2,1(WR2)
004116 198A                        4999          CR    WR2,WR1               IS HEAD NUMBER = 10
004118 4780 8128            04128  5000          BC    8,OTHE                YES,BRANCH TO OTHE
00411C 0000 0000            00000  5001          STC   WR2,WHHH+1            NO,STORE HEAD NUMBER IN HA
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004120 0000 0000            00000  5002          STC   WR2,WHRO+4            AND IN RECORD ZERO
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004124 47F0 80BC            040BC  5003          B       A9CD                BRANCH TO A9CD
004128 0000 0000 0000 00000 00000  5004 OTHE     CLC   CCWRIT+2(2),DSKPR     COMPARE CCW ADDRESS WITH DISK
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
                                   5005 *                                    PROGRAM ADDRESS
00412E 4780 8150            04150  5006          BC    8,OTHE1               IF EQUAL,BRANCH TO OTHE1
004132 0000 0000      00000        5007          MVI   WHHH+1,X'00'          OTHERWISE, SET HEAD NBR = 0
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004136 0000 0000      00000        5008          MVI   WHCC+1,X'02'          SET CYLINDER = 02
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00413A 0000 0000      00000        5009          MVI   WHRO+2,X'02'          RECORD ZERO = CYLINDER 02
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00413E 0000 0000      00000        5010          MVI   WHRO+4,X'00'          HEAD 0
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004142 0000 0000            00000  5011          LH    WR2,DSKPR             LOAD DISK PROGRAM ADDRESS
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004146 0000 0000 0000 00000 00000  5012          MVC   CCWRIT+2(2),DSKPR     PLACE DISK PR.ADDRESS INTO CCW
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00414C 47F0 80BC            040BC  5013          B       A9CD                GO TO MESSAG SUBROUTINE
004150 0000 0000            00000  5014 OTHE1    L     R2,REG2               RESTORE REGISTERS R1 AND R2
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004154 0000 0000            00000  5015          L     R1,REG1               *
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
004158 1912                        5016          CR    R1,R2                 IS REGISTER R1 EQUAL TO REG. R2
00415A 0000 0000            00000  5017          BE      END1                YES,BRANCH TO END1
   IEV044  *** ERROR ***  UNDEFINED SYMBOL
00415E 4111 0002            00002  5018          LA    R1,2(R1)              NO,INCREMENT R1 BY 2
004162 47F0 809A            0409A  5019          B       OTHE2               BRANCH TO OTHE2
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                ASM H V 05 22.10 05/07/81
                                   5020 *-*       END
                                   5021           AIF   (NOT &DISK).NODSK11
                                   5022 .NODSK11  ANOP
004166 47F0 000C            0000C  5023           B       BRCC                BRANCH TO BEGINNING OF SIM20
                            04000  5024 BEGINA    EQU   DSKINT                *
004000                             5025           END   DSKINT
```

ASM H V 05 22.10 05/07/81

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000010
0001	0001	0C	000014
0001	0001	0C	00004E
0001	0001	0C	00020C
0001	0001	0C	000290
0001	0001	0C	0002A8
0001	0001	0C	0002AC
0001	0001	0C	0002EC
0001	0001	0C	000334
0001	0001	0C	000340
0001	0001	08	000351
0001	0001	0C	0004F4
0001	0001	0C	0004FC
0001	0001	0C	000508
0001	0001	0C	00070C
0001	0001	0C	000710
0001	0001	0C	000714
0001	0001	0C	000718
0001	0001	0C	00071C
0001	0001	0C	000720
0001	0001	0C	000724
0001	0001	0C	000728
0001	0001	0C	00072C
0001	0001	0C	000730
0001	0001	0C	000734
0001	0001	0C	000738
0001	0001	0C	00073C
0001	0001	0C	000740
0001	0001	0C	000744
0001	0001	0C	000748
0001	0001	0C	00074C
0001	0001	0C	000750
0001	0001	0C	000754
0001	0001	0C	000758
0001	0001	0C	00075C
0001	0001	0C	000760
0001	0001	0C	000764
0001	0001	0C	000768
0001	0001	0C	00076C
0001	0001	0C	000770
0001	0001	0C	000774
0001	0001	0C	000778
0001	0001	0C	00077C
0001	0001	0C	0009F4
0001	0001	0C	0009FC
0001	0001	0C	000A04
0001	0001	0C	000A08
0001	0001	0C	000A0C
0001	0001	0C	000A10
0001	0001	0C	000A14
0001	0001	0C	000A18
0001	0001	0C	000A1C
0001	0001	0C	000A20
0001	0001	0C	000A24
0001	0001	0C	000A28

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000A2C
0001	0001	0C	000A30
0001	0001	0C	000A34
0001	0001	0C	000A38
0001	0001	0C	000A3C
0001	0001	0C	000A40
0001	0001	0C	000A44
0001	0001	0C	000A48
0001	0001	0C	000A4C
0001	0001	0C	000A50
0001	0001	0C	000A54
0001	0001	0C	000A58
0001	0001	0C	000A5C
0001	0001	0C	000A60
0001	0001	0C	000A64
0001	0001	0C	000A68
0001	0001	0C	000A6C
0001	0001	0C	000A70
0001	0001	0C	000A74
0001	0001	0C	000A78
0001	0001	0C	000A7C
0001	0001	0C	000A80
0001	0001	0C	000A84
0001	0001	0C	000A88
0001	0001	0C	000A8C
0001	0001	0C	000A90
0001	0001	0C	000A94
0001	0001	0C	000A98
0001	0001	0C	000A9C
0001	0001	0C	000AA0
0001	0001	0C	000AA4
0001	0001	0C	000AA8
0001	0001	0C	000AAC
0001	0001	0C	000AB0
0001	0001	0C	000AB4
0001	0001	0C	000AB8
0001	0001	0C	000ABC
0001	0001	0C	000AC0
0001	0001	0C	000AC4
0001	0001	0C	000AC8
0001	0001	0C	000ACC
0001	0001	0C	000AD0
0001	0001	0C	000AD4
0001	0001	0C	000AD8
0001	0001	0C	000ADC
0001	0001	0C	000AE0
0001	0001	0C	000AE4
0001	0001	0C	000AE8
0001	0001	0C	000AEC
0001	0001	0C	000AF0
0001	0001	0C	000AF4
0001	0001	0C	000AF8
0001	0001	0C	000AFC
0001	0001	0C	000B00
0001	0001	0C	000B04

ASM H V 05 22.10 05/07/81

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000B08
0001	0001	0C	000B0C
0001	0001	0C	000B10
0001	0001	0C	000B14
0001	0001	0C	000B18
0001	0001	0C	000B1C
0001	0001	0C	000B20
0001	0001	0C	000B24
0001	0001	0C	000B28
0001	0001	0C	000B2C
0001	0001	0C	000B30
0001	0001	0C	000B34
0001	0001	0C	000B38
0001	0001	0C	000B3C
0001	0001	0C	000B40
0001	0001	0C	000B44
0001	0001	0C	000B48
0001	0001	0C	000B4C
0001	0001	0C	000B50
0001	0001	0C	000B54
0001	0001	0C	000B58
0001	0001	0C	000B5C
0001	0001	0C	000B60
0001	0001	0C	000B64
0001	0001	0C	000B68
0001	0001	0C	000B6C
0001	0001	0C	000B70
0001	0001	0C	000B74
0001	0001	0C	000B78
0001	0001	0C	000B7C
0001	0001	0C	000B80
0001	0001	0C	000B84
0001	0001	0C	000B88
0001	0001	0C	000B8C
0001	0001	0C	000B90
0001	0001	0C	000B94
0001	0001	0C	000B98
0001	0001	0C	000B9C
0001	0001	0C	000BA0
0001	0001	0C	000BA4
0001	0001	0C	000BA8
0001	0001	0C	000BAC
0001	0001	0C	000BB0
0001	0001	0C	000BB4
0001	0001	0C	000BB8
0001	0001	0C	000BBC
0001	0001	0C	000BC0
0001	0001	0C	000BC4
0001	0001	0C	000BC8
0001	0001	0C	000BCC
0001	0001	0C	000BD0
0001	0001	0C	000BD4
0001	0001	0C	000BD8
0001	0001	0C	000BDC
0001	0001	0C	000BE0

ASM H V 05 22.10 05/07/81

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000BE4
0001	0001	0C	000BE8
0001	0001	0C	000BEC
0001	0001	0C	000BF0
0001	0001	0C	000BF4
0001	0001	0C	000BF8
0001	0001	0C	000BFC
0001	0001	0C	000C00
0001	0001	0C	000C04
0001	0001	0C	000C08
0001	0001	0C	000C0C
0001	0001	0C	000C10
0001	0001	0C	000C14
0001	0001	0C	000C18
0001	0001	0C	000C1C
0001	0001	0C	000C20
0001	0001	0C	000C24
0001	0001	0C	000C28
0001	0001	0C	000C2C
0001	0001	0C	000C30
0001	0001	0C	000C34
0001	0001	0C	000C38
0001	0001	0C	000C3C
0001	0001	0C	000C40
0001	0001	0C	000C44
0001	0001	0C	000C48
0001	0001	0C	000C4C
0001	0001	0C	000C50
0001	0001	0C	000C54
0001	0001	0C	000C58
0001	0001	0C	000C5C
0001	0001	0C	000C60
0001	0001	0C	000C64
0001	0001	0C	000C68
0001	0001	0C	000C6C
0001	0001	0C	000F54
0001	0001	0C	001004
0001	0001	0C	001008
0001	0001	0C	00100C
0001	0001	0C	001010
0001	0001	0C	001014
0001	0001	0C	001018
0001	0001	0C	00101C
0001	0001	0C	001020
0001	0001	0C	001024
0001	0001	0C	001028
0001	0001	0C	00102C
0001	0001	0C	001030
0001	0001	0C	001034
0001	0001	0C	001038
0001	0001	0C	00103C
0001	0001	0C	001040
0001	0001	0C	001044
0001	0001	0C	001048
0001	0001	0C	00104C

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	001050
0001	0001	0C	001054
0001	0001	0C	001058
0001	0001	0C	00105C
0001	0001	0C	001060
0001	0001	0C	001064
0001	0001	0C	001068
0001	0001	0C	00106C
0001	0001	0C	001070
0001	0001	0C	001074
0001	0001	0C	001078
0001	0001	0C	00107C
0001	0001	0C	001080
0001	0001	0C	001084
0001	0001	0C	001088
0001	0001	0C	00108C
0001	0001	0C	001090
0001	0001	0C	001094
0001	0001	0C	001098
0001	0001	0C	00109C
0001	0001	0C	0010A0
0001	0001	0C	001114
0001	0001	0C	001168
0001	0001	0C	001CF8
0001	0001	0C	001D9C
0001	0001	0C	001EBC
0001	0001	0C	001F88
0001	0001	0C	001FE8
0001	0001	0C	002000
0001	0001	0C	002004
0001	0001	0C	002038
0001	0001	0C	002068
0001	0001	08	002071
0001	0001	0C	002140
0001	0001	0C	002158
0001	0001	0C	00215C
0001	0001	0C	00219C
0001	0001	0C	002208
0001	0001	08	002289
0001	0001	0C	0022D0
0001	0001	0C	0022E8
0001	0001	0C	0022EC
0001	0001	0C	002330
0001	0001	0C	002348
0001	0001	0C	002354
0001	0001	08	002369

ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES												
ACARRY	00010	001364	2737	2843	2856											
ACOMA	00002	0007A8	1062	1048	1049	4198	4199									
ACOMP	00002	000F30	2157	2111	2112	2120	2121									
ADDA	00004	00125B	2647	2644												
ADDB	00004	0012D0	2680	2637												
ADDC	00004	00124C	2643	2639	2685											
ADDE	00004	0012DC	2683	2641												
ADDF	00004	00122C	2634	2629	2645											
ADDFQ	00001	00006000	4851	4859	4863	4870	4874									
ADDG	00006	00127E	2657	2649												
ADDH	00004	001294	2663	2651	2667											
ADDL	00004	00128C	2661	2653												
ADDM	00004	00126A	2652	2662												
ADDN	00004	0012A8	2668	2666												
ADDPQ	00004	0012E8	2703	2630	2634	3055	3096									
ADDPQX	00004	0012F4	2706	3063	3104											
ADDPS	00006	0012C0	2675	2673												
ADDQS	00006	0012C8	2677	2671												
ADDT	00004	0012AE	2670	2674												
ADDZ	00004	001318	2718	2714												
ADD1A	00006	001194	2491	2535												
ADD2A	00004	0011BE	2505	2503												
ADD3	00004	00120E	2624	2621												
ADD3A	00004	0011BA	2504	2507												
ADD4	00004	001212	2626	2623												
ADD4A	00004	0011AA	2498	2489	2534	3866										
ADD5	00004	00121E	2629	2619	3531											
ADD5A	00006	00118E	2490	3867												
ADD6A	00004	00118A	2489	2518												
AL	00004	00052C	0805	0815												
ALAMES	00004	000100	0386	0366												
ALAREG	00004	000084	0311	0300	0308											
ALARM	00004	000060	0300	0386	0541	4175										
ALARSW	00004	0000C0	0367	0355	0368											
AM11	00004	0011CA	2516	1540												
AP	00002	00055B	0816	0806												
APEND	00004	00123E	2639	2626	2635	2682	2683									
APSIGN	00001	0013A4	2745	2491	2620	2657	2675	2818	2879	3045	3082	3128	3129	3132	3136	3488	3872
AQ	00006	00055E	0818	0808												
AQEND	00004	001230	2635	2505	2627	2680	3877									
AQSIGN	00001	0013A5	2746	2490	2532	2533	2620	2677	3046	3484	3512	3639	3640	3871	3872	
AR	00001	000193	0428	2984												
ARCHK	00004	00157C	2983	2052												
ATAB1	00010	001324	2729	2834												
ATAB2	00010	001374	2739	2681	2684	2844	2857	3101	3316	3927						
ATAB3	00010	001334	2731	2664	2705	3062										
AU	00004	00053C	0809	0817												
A21	00004	001186	2488	1556												
A9CD	00004	0040BC	4976	5003	5013											
BB42	00004	001108	2359	1206	1589	2365										
BB42A	00004	001118	2364	2359												
BB42B	00004	001124	2367	2360												
BCXM64	00004	001DBE	3926	1623	3901											
BCX63	00004	001DB6	3913	1622												
BD43	00004	000F32	2166	1590												

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
BIA      00004   000FB2  2224   2255
BIC      00004   000FB4  2237   2232
BID      00004   000FC4  2241   2234  2238
BI46     00004   000F7A  2222   1593
BLXM66   00004   001E12  3972   1625
BLX1     00004   001DEB  3950   3974
BLX2     00002   001E0C  3959   3994
BLX65    00004   001DE4  3949   1624
BLX651   00004   001DF2  3953   3955
BNF44    00004   000F5A  2185   1591
BNG55    00002 00000A00  2375   1608
BNH      ****UNDEFINED****       1329  1902  4084
BNIS1    00004   000FBC  2239   2222  2226  2253
BNIS2    00004   000FC8  2242   2223  2254
BNI47    00004   000FD0  2253   1594
BNR45    00004   000F6A  2197   1592
BOM      00001 00000005  0121   2034  2037  2042  2045  2050  2058  2943  2946  4520
BP       ****UNDEFINED****       4838
BRCC     00004   00000C  0224   4942  5023
BSE      00002   001CEE  3806   3790  3794  3798  3802
BSX67    00004   001E1E  3985   1626
BSX67A   00004   001E26  3988   3990
BS1      00004   001CB2  3791   3788
BS2      00004   001CC2  3795   3792
BS3      00004   001CD2  3799   3796
BS4      00004   001CE2  3803   3800
BS60     00004   001C9A  3785   1619
BTAM10   00004   001C78  3753   1539
BTA20    00004   001C10  3711   1555
BTA20A   00004   001C14  3712   3755
BTA20B   00002   001C2E  3723   3722  3735  3741
BTA20C   00006   001C36  3726   3737
BTA20D   00004   001C3C  3727   3720  3740
BTA20E   00002   001C48  3731   3721  3732  3739
BTA20F   00002   001C5A  3736   3725
BTA20G   00002   001C60  3738   3730
BTA20H   00004   001C66  3740   3734
BTA20J   00002   001C6E  3742   3727
BTFL07   00004   001948  3410   1530
BTM17    00004   0010FC  2340   1546
BT27     00004   0010CC  2319   1562
BT27A    00004   0010D0  2320   2342
BT27B    00002   0010E2  2325   2329  3420
BUFFP    00008   0009D8  1475   1157  1159  1381  1382  1383  2432  2433  2434  3259  3260  3261  3262  3352  3353  3354
                                3355  3370  3371  3373  3374  3376  3377  3379  3380  3490  3491  3492  3493
BUFFQ    00008   0009E0  1476   1388  1389  1390  1402  1403  1404  1445  1446  2227  2228  2229
BUFF2    00002   0004F0  0736   0714  0715  0728  0728  0729  4776  4777  4777  4778
BXA      00004   001D62  3870   3844  3900  3913  3926
BXB      00004   001DA0  3884   3870  3876
BXC      00004   001D82  3877   3873  3875  3887
BXD      00004   001D20  3852   3933  3973
BXE      00006   001D04  3846   3949  3985
BXF      00002   001D5C  3868   2498
BXG      00002   001D44  3861   3845  3862  3932
BXM62    00004   001DAE  3900   1621
```

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
BX61     00004   001CFC  3844  1620  3914
BZM      00001 0000000C  0123  1056  3068  3109
B49      00004   000F3E  2169  1596  2188  2200  2239  2243
CAPACT   00004   000018  0229  4934
CCWINI   ****UNDEFINED****     4976
CCWIST   ****UNDEFINED****     4978
CCWLD    ****UNDEFINED****     4982
CCWLDR   ****UNDEFINED****     4984
CCWRD    ****UNDEFINED****     4947  4964  4965
CCWRIT   ****UNDEFINED****     4967  4969  4986  4989  4994  5004  5012
CF33     00004   001144  2411  1574
CHKSTP   00001 0000019D  0447  0740
CHK360   00001   0001C7  0441  0743
CM14     00004   000E60  2089  1543
CNTR     00001 00000005  0098  0378  1048  1050  1055  1149  1149  1157  1205  1268  1295  1296  1296  1297  1384  1388
                               1391  1827  1852  2090  2170  2171  2224  2320  2322  2341  2364  2367  2430  2517  2545
                               2776  3169  3207  3411  3413  3712  3714  3753  3786  3806  3853  3879  3884  3886  3928
                               3929  3959  3972  4198  4451  4458  4639  4839  4937
CODERR   00002   000A00  1505  1523  1527  1533  1534  1535  1536  1537  1538  1549  1550  1551  1552  1553  1554  1565
                               1566  1567  1568  1569  1570  1581  1582  1583  1584  1585  1586  1587  1597  1598  1599
                               1600  1601  1602  1603  1604  1605  1606  1607  1609  1610  1611  1612  1613  1614  1615
                               1616  1617  1618  1627  1628  1629  1630  1631  1632  1633  1634  1639  1640  1641  1642
                               1643  1644  1645  1646  1647  1648  1649  1650  1651  1652  1653  1654  1655  1656  1657
                               1658  1659  1660  1661  1662  1663  1664  1665  1666  1667  1668  1669  1670  1671  1672
                               1673  1674  1675  1676  1910  2375  4718  4719  4723  4724  4725  4726  4727
CODINA   00004   0023EE  4784  4769
CODINV   00004   0004E8  0733  0716
COMDB    00002   000588  0825  0797  0818
COMP     00004   000E8C  2109  2063  3318
COMPA    00004   000E9C  2112  2110
COMPB    00004   000F10  2148  2113
COMPB1   00004   000F20  2152  2122
COMPC    00004   000EC0  2121  2119
COMPD    00004   000ECC  2124  2153  2155
COMPE    00004   000F18  2150  2114
COMPE1   00004   000F28  2154  2123
COMPF    00002   000EAC  2116  2127  2149  2151
COMP1    00004   000EE4  2131  2125  2135
COMP2    00004   000EFA  2138  2129  2132
CONVP    00004   000906  1381  1053  1828  1853  2089  2169  2340  2395  2411  2516  2544  2775  3168  3206  3754  3785
                               3852
CONVPQ   00004   000902  1379  1831  1856  1890  1929  1969  2022  2319  2452  2488  2531  2761  3153  3183  3248  3308
                               3396  3410  3428  3450  3560  3636  3711  3763
CONVQ    00006   00091E  1388  2166  2185  2197  3849
CONVTB   00001 00000080  0124  0713  0799  0800  0800  0804  4766
COREST   00004   000014  0226  4933  4945
CORES1   00001   003002  4892  0226  0737
CUTZ     00004   0014D4  2898  2706
CVQ      00004   0012EC  2704  2622  2624  2648  2704  3048  3095
C24      00004   000DC2  2022  1559  2091
C24A     00004   000DE2  2030  2025
C24ARC   00002   000E5C  2066  2034  2037  2072
C24AR1   00004   000E60  2068  2042  2045
C24AR2   00004   000E74  2073  2069
C24B     00004   000E4C  2062  2027  2031
```

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES													
C24C	00004	000DEE	2033	2029													
C24D	00004	000E2E	2054	2048	2059												
C24E	00002	000DFE	2039	2050													
C24F	00004	000E44	2060	2055													
C24I	00004	000E26	2052	2064	2075	2504	3156	3383	3878								
DEC20K	00004	001F50	4169	4158	4387												
DEC4	00002	0009D0	1466	1401	3950												
DISIO	00004	0007E2	1108	0630	1100	1104											
DMS0001B	00001	000044	0281	0276	0277												
DMS0001C	00001	000048	0282	0278													
DMS0001E	00002	00004C	0283	0279													
DN19	00004	00171A	3168	1548													
DNCD	00004	00209C	4382	1028													
DNCD1	00004	0020AE	4387	4389													
DNCD2	00004	0020BA	4391	4393													
DNPT	00002	00000A00	4726	1027													
DNTAB	00004	00076C	1025	0992													
DNTY	00002	001F2A	4157	1026	4160												
DN35	00004	0006DA	0989	1576													
DPROG	00004	00079E	1055	1051													
DSKINT	00004	004000	4899	5024	5025												
DSKPR	****UNDEFINED****			5004	5011	5012											
DUMCAR	00004	002010	4300	4280	4303	4306	4348	4384									
DUMPRI	00004	002120	4467	4447	4560	4593											
DUMP1R	00002	0021A0	4504	4467	4472												
DUMP2R	00004	002134	4473	4505													
D29	00004	001700	3153	1564													
D29A	00004	001704	3154	3170													
D29B	00002	001710	3157	3155													
END1	****UNDEFINED****			5017													
ENTRY0	00004	000007F6	1464	2053	2058	2061	2065	2071	2074	2506	2765	3159	3279	3283	3286	3359	3362	3365
				3585	3621	3624	3881											
ENTRY1	00004	0007F6	1149	0373	0540	1464	1833	1862	1893	1946	1986	2168	2187	2199	2240	2242	2397	2413
				2457	3196	3430	3462	3470	3768	4203	4207	4651	4663					
ENTRY2	00004	0007FA	1153	2173														
ENTRY3	00002	0007B0	1094	0538	4320	4527												
EOR1	00002	002426	4831	4816														
EOR2	00002	002418	4825	4819														
EOR3	00002	00244E	4844	4832														
EOR4	00004	002434	4836	4845														
EOR5	00004	00243E	4839	4833														
EOR6A	00002	00244A	4842	4840														
EPH	00004	001AA0	3536	3322	3357	3367	3494	3507	3578	3663								
EQH	00004	001AA4	3537	3507	3664	3666	3667											
ERROR	00004	0004F8	0739	0955	0971	0978	0979	0980	0990	1001	1003	1004	1006	1010	1015	1019	1020	1025
ERRORP	00004	0009F0	1501	0960	1453	1932	1972	4841										
ERXOP	00004	001CF4	3808	3804														
EXCHK	00004	001598	2990	3282	3361	3382	3584											
EXCKPR	00004	002334	4688	4661														
EXCPRN	00004	0021BC	4512	4484														
EXCRNC	00004	0002F0	0616	0593														
EXCRN1	00004	000324	0631	0627														
EXCRN2	00004	00031C	0629	0608														
EXCRN3	00004	000308	0622	0620														
EXCRN4	00004	000314	0627	0621														

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
EXCR3    ****UNDEFINED****        0623  4315  4318  4517
EXCWNC   00004   00203C  4314     4297
EXP      00001   00018B  0426     2991
EXPAND   00006   0004D4  0728     0717
EXPOW    00004   001B9A  3657     3278  3358  3580
EXP1AB   00004   001BC8  3670     3659
EXP2AB   00004   001BF0  3681     3662
EXP3     00002   001BDA  3675     3679
EXP4     00002   001BFE  3685     3688
FADDA    00004   001B0E  3592     3565  3567
FADDB    00004   001AE0  3577     3609
FADDC    00004   001AD8  3574     3572
FADDU    00004   001B16  3595     3615
FADD01   00004   001AA8  3560     1524
FADD1    00006   001AC6  3570     3573
FADD3    00002   001B50  3611     3596
FADD4    00004   001B3E  3606     3602  3610
FADD5    00002   001AB0  3562     3642
FADD6    00006   001B26  3600     3605
FADD9    00004   001AFA  3584     3588
FDIA     00004   0018E4  3366     3320
FDIB     00002   00183C  3323     3368
FDIC     00004 00000570  2766     2877  3059  3070  3102  3111  3254  3313  3324  3325  3340  3349  3369  3927  3928  3929
                                   3930  3931
FDIE     00002   001852  3328     3332
FDIV09   00004   001804  3308     1532
FDIV6    00002   00188C  3344     3348
FDIV7    00004   0018D8  3363     3351
FIXADD   00004   0011F8  2619     2495  3487
FIXDA    00004   0015CC  3055     3061
FIXDIA   00002   0015C6  3052     3081
FIXDIE   00004   0015F2  3065     3089
FIXDIV   00004   0015A4  3043     3154  3335
FIXD1    00004   0015E6  3062     3060
FIXD10   00004   0016A2  3114     3109  3112
FIXD11   00004   00168A  3108     3113
FIXD12   00004   0016BE  3122     3119
FIXD13   00004   0015BC  3048     3121
FIXD15   00004   0016F2  3136     3130
FIXD16   00004   0016EC  3134     3137
FIXD17   00004   0016FA  3138     3133
FIXD18   00004   001654  3092     3087
FIXD19   00004   00163E  3086     3091
FIXD2    00002   00163A  3084     3065
FIXD4    00004   001610  3073     3068  3071
FIXD5    00004   0015F8  3067     3072
FIXD6    00006   001630  3082     3080
FIXD8    00004   001672  3101     3098
FIXD9    00004   001660  3096     3100
FIXOFL   00004   0018F2  3369     3337
FLCOSW   00004   0023A8  4764     4075  4125  4165  4283  4351  4449  4562  4774
FLEXSW   00004   0004B2  0717     0508  0581  0731  0734  4017  4227
FMUL03   00004   00176E  3248     1526
FMUL1    00006   0017B8  3269     3265
FMUL2    00004   0017BE  3270     3274
```

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
FMUL3    00004  0017F8  3284  3277
FMUL4    00004  0017F0  3282  3280
FSLA     00002  0019A2  3452  3454
FSLB     00006  0019AC  3455  3460
FSLC     00002  0019C4  3461  3457
FSLD     00004  0019E0  3469  3466
FSLE     00004  0019CE  3464  3468
FSL05    00004  00199C  3450  1528
FSRA     00002  00197C  3432  3436
FSRB     00002  00198E  3437  3441
FSRC     00006  001980  3434  3521
FSR08    00004  00196A  3428  1531
FSUB02   00004  001B80  3636  1525
GETEOM   00004  0023FE  4815  4822  4830
GETEOR   00004  0023F6  4811  4078  4468
HLT      00001  0001A5  0432  2434  2436
HPEZ     00001  000003  0193  1240  2028  2032  2051  2060  2115  2136  2142  2145  2148  2150  2152  2154  2279  2280
                              2281  2949  2953  2955  3043  3276  3281  3319  3350  3583  3620  3874
H48      00002  001150  2430  1595
INDAD    00002  00093C  1397  1454  3801  3805
INDERR   00010  000FF0  2266  2268  2273  2276  2278  2284  2285  2286  2288  2289  2290  2291  2292  2294  2295  2296
                              2297
INDEX    00004  00095C  1426  1385  1392  1405  3857
INDIC    00004  001548  2942  2502  2764  3158  3257  3339  3587  3619  3869
INDICA   00004  001004  2268  2236
INDICM   00010  000FDC  2264  2231
INDICX   00010  0010A4  2308  2233
INDX1    00004  000972  1432  1430
INDX2    00004  00097E  1435  1433
INDX3    00002  00098A  1438  1436  3856  3858
INDX4    00004  00099C  1443  1441  3855  3859
INDX5    00004  0009C2  1453  1427  1439  1451
INDX6    00004  0009B6  1450  1448
INDX8    00004  0009C6  1454  3848  3850
IND1     00002  001550  2944  2948
IND2     00004  001568  2951  2943  2946
IND3     00004  001576  2955  2952
INITP    00004  0009E8  1477  2023  2138  2492  2499  2647  2820  2878  2889  3044  3122  3189  3194  3258  3317  3496
                              3525  3568  3577  3586  3592  3617
INITQ    00004  0009EC  1478  2873  3127  3372
INTREQ   00001  000121  0408  0376  4533  4698
INVCOD   00001 0000019D 0444  1507
INVP     00001 0000019D 0445  1502
INVQ     00001 0000019D 0446  1504
INVXOP   00001  0001B1  0435  3809
IOBUFF   00008  0003A8  0670  0514  0598  0648  0703  4092  4329  4507  4508  4508  4604  4762
IOKPR    ****UNDEFINED****     4949  4969
IR2      00004  001134  2372  2321  2367  2368  3412  3713
IXBAND   00001  000006  0209  1426  1440  2298  2299  2300  3789  3793  3797  3846  3847  3851  4941
KAUTO    00004  000880  1292  0535  1267
KCHECK   00004  00083C  1220  1204
KCLEAR   00006  0008BA  1325  1329
KEYBIT   00001  000000  0161  0372  0532  0534  0706  1112  1153  1196  1202  1207  1220  1222  1236  1238  1266  1292
                              1294  1318  1320  1331  1337  1340  4885  4896  4939  4940
KINSER   00004  000870  1266  1237
```

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
KMESS    00002 001F90 4208  4204  4206
KPRCCW   00008 002368 4704  4652  4656
KPRGO    00002 0022CC 4654  4872
KPRINT   00002 002290 4637  1005
KPRIR    00004 00234C 4695  4690
KPRPSW   00008 0022E0 4659  4683  4686
KPRUE    00004 002358 4699  4689
KPR1     00004 0022C6 4652  4646
KRESET   00004 00084C 1236  1221  1245
KRESIO   00004 000864 1242  1223
KSAVE    00004 000828 1202  1090  1195  1343
KSTART   00004 00089E 1318  1293
KSTAR1   00004 0008D6 1333  1319
KSTAR2   00004 0008EA 1340  1334
KTAB     00009 001F92 4209  4200
KTABA    00004 00070C 1001  0921
KTYPE    00004 001F64 4195  1002
K34      00004 000670 0918  1575
LASCAR   00001 00014A 0414  0643
LASTCD   00001 000005 0205  0644  1239  2277
LDM18    00004 001762 3206  1547
LD28     00004 001726 3183  1563
LD28A    00004 00172A 3184  3208
LD28B    00004 001742 3190  3193
LIMITX   00004 0004F4 0737  0708
MAPORG   00001 00000006 0099  0726  1268  1269  1297  1298  1321  1322  1322  1323  1443  1453  1866  1869  1903  1905
                              2431  2816  2817  2817  2823  2830  2876  3045  3117  3131  3184  3185  3185  3195  3255
                              3275  3314  3315  3315  3321  3327  3333  3341  3366  3724  3729  4157  4304  4386  4780
                              4828  4834  4933  4935  4937
MASK     00004 000510 0797  0509  0521  0582  0595  0818  4018  4076  4080  4126  4164  4228  4281  4284  4349  4385
                              4448  4464  4561  4594
MASKRG   00004 000570 0824  0801  0819  1052  1054  2766
MAXVAL   00004 0009CC 1465  1449  1450  1452
MA70     00004 001C84 3763  1635
MCLEAR   00004 0014D8 2899  1330  2628  2827  3367
MESSAG   00004 000088 0351  0385  0387  0641  0742  1110
MESSAV   00004 000108 0389  0354  0370  0374  0379
MESSA2   00004 00008C 0353  4205
MESSIO   00002 0000AA 0360  0351  0357  0359  0369  0381  4195
MESSOK   00004 0000CC 0370  0364
MESSWI   00004 0000D0 0371  0223  0353  0380
MESTIR   00004 0000DC 0374  0636  4324  4532  4696
MESTOP   00004 0000F4 0380  0224  0739  1501  1503  1506  2174  2362  2435  2988  3808  3882  4692
MF71     00004 00116C 2452  1636
MLDT     00006 00056A 0822  0811  0813
MM13     00004 0013B8 2775  1542
MQTAB    00008 0014F4 2910  2829
MTAB1    00008 001504 2912  2708  2838  2849  2851  2863  3322
MTAB2    00010 001534 2918  2841  2855
MULT     00004 0013C4 2816  2762  3253
MULTA    00004 001468 2866  2860
MULTB    00002 001442 2854  2864
MULTC    00004 001476 2871  2867
MULTD    00002 0013F8 2832  2869
MULTE    00004 0014B6 2889  2872  2874
```

ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES														
MULTF	00002	0013E6	2826	2891														
MULT1	00004	0014C0	2892	2881														
MULT2	00004	0014C8	2894	2883														
MULT3	00004	0014AC	2885	2893														
MULT4	00004	0014B0	2886	2895														
MVBLAN	00004	0021A6	4506	4470														
M23	00004	0013A6	2761	1558														
M23A	00004	0013AA	2762	2777														
NOBRAD	00001	00019D	0430	0444	0445	0446	0447	2175	2363	3883								
NOP	00001	00000000	0120	0304	0365	0367	0705	0717	2064	2239	2242	2359	2360	2496	2498	2629	2635	2639
				2704	3870	4094	4099	4300	4467	4764	4942							
NOP41	00004	00114C	2413	1588														
NOQEND	00004	000E54	2064	2062	2128													
NRMKPR	00004	002328	4683	4660	4702													
NRMPRN	00004	00217E	4494	4483														
NRMPR4	00004	0021BA	4497	4528														
NRMPR5	00004	0021F4	4527	4525														
NRMRNC	00004	0002E0	0610	0592	0632													
NRMUNC	00004	00202C	4309	4296	4322													
NYCHEK	00001	000002	0183	0520	0605	0624	0733	1097	1101	1105	1242	2274	2275	2287	2293	2304	2305	2306
				2307	4098	4319	4529	4784										
OFL	00004	001B02	3586	3582														
OFLOW	00001	000004	0199	1241	2282	2283	2983	2990										
OFLSWI	00004	001584	2985	2992														
OPTBL	00004	000A08	1523	1161														
OTHE	00006	004128	5004	5000														
OTHE1	00004	004150	5014	5006														
OTHE2	00004	00409A	4968	5019														
OUTIN	00006	000780	1048	0918	0930	0941	0954	0970	0989									
PACK1	00006	0023D4	4776	4764														
PACK2	00004	0023B0	4766	4779														
PACK3	00004	0023C0	4770	4785														
PBB	00002	0000213E	4868	4869														
PBB1	00002	000002FA	4869	4870														
PBB2	00002	000062FA	4870	4871														
PDLOK	00001	000154	0416	4326														
PNBB	00002	000022CE	4872	4873														
PNBB1	00002	0000048A	4873	4874														
PNBB2	00002	0000648A	4874	4875														
PRA	00004	002224	4557	1024														
PRD	00004	002258	4590	1029														
PRICHK	00001	0001C0	0439	1107														
PRLOCK	00004	002176	4491	4485	4498													
PRN	00004	0020CC	4444	1014														
PRNA	00004	0020E0	4449	4595														
PRNB	00004	0020E4	4451	4563														
PRNCSW	00008	002148	4481	4494	4524													
PRNDEL	00004	0020F8	4456	4452														
PRNERR	00001	002144	4480	4516	4519	4521												
PRNGO	00002	00213C	4477	4510	4868													
PRNIR	00004	002204	4532	4514														
PRNPSW	00008	002150	4482	4499	4503													
PRNSAV	00004	00220C	4534	4495	4500	4512												
PRNUE	00004	0021F0	4526	4513														
PRN1	00004	002100	4458	4453														

ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES
PRN2	00004	002114	4463	4455 4461
PRN3	00004	002118	4464	4457
PROSWI	00001	000001	0173	1095 2269 2270 2271 2272 2985
PR1	00004	0008FC	1345	1205 2364
PR1CCW	00008	002288	4604	4454 4456 4460 4462 4473 4479 4487 4509
PXA2PR	00004	0021FC	4529	4518 4520
PXA4PR	00004	0021DC	4521	4530
PXA5PR	00004	0021E8	4524	4522
Q10	00001	002370	4706	4639 4640 4645
Q11	00001	002371	4707	4643
RACD	00004	001FA2	4225	1018
RAPT	00002	00000A00	4718	1017
RATAB	00004	000744	1015	0957
RATY	00004	001E44	4015	1016 4854 4858 4862 4869 4873 4882 4944
RA1	00004	0006B6	0958	0974
RA37	00004	0006A6	0954	1578
RCDCCW	00008	000350	0648	0587
RCDI1	00004	000004F8	0980	0931
RCLEAR	00004	0014EC	2901	2842
RDRNOF	00001	00013E	0412	0638
RDTAB	00006	0006FC	0998	0932 0956
READY	00001	0001E2	0451	0225
REDCHK	00001	0001DB	0448	0629 1099
REG1	****UNDEFINED****			4968 5015
REG15	00004	0014D0	2897	2821 2886 3846 3851 3854 3860 3863 3884
REG2	****UNDEFINED****			5014
REQUST	****UNDEFINED****			4926
RESTAR	00006	000806	1157	1339
RET	00002	00091C	1387	1379 1394
RFLAG	00004	0014F0	2902	2836 2848
RNCD	00004	000272	0579	1009 1300
RNCDIR	00004	00032C	0635	0618
RNCDUE	00004	000338	0640	0599 0617
RNCD1A	00004	000282	0583	0645 4229
RNCD2	00004	000266	0540	0603 0606 0607
RNCERR	00001	000294	0588	0619 0622
RNCPSW	00008	0002A0	0591	0611 0614
RNCSAV	00004	00034C	0646	0616 0631 0635 0640 4314 4321 4323 4688 4691 4695 4699
RNCSWI	00004	0002CC	0603	0583 0628
RNLOCK	00004	0002B8	0596	0594 0610
RNPT	00002	00000A00	4719	1008
RNTAB	00004	000720	1006	0933
RNTY	00004	0001F4	0506	1007 1270
RNTYAL	00004	00026A	0541	0517 0519
RNTYGO	00002	000208	0511	0516 0536 0542 4019
RNTYOK	00004	000228	0521	0515
RN36	00004	000682	0930	1577
RP	00001	00000008	0104	0708 0718 0719 0719 0720 0726 0729 0730 0730 0958 0961 1269 1298 1386 1832
				1857 1859 1864 1865 1866 1873 1891 1897 1897 1898 1905 1930 1933 1935 1944
				1947 1970 1973 1974 1975 1978 1980 1981 1982 1983 1987 2023 2024 2033 2039
				2041 2049 2054 2056 2057 2066 2068 2111 2116 2120 2126 2131 2133 2134 2138
				2139 2170 2322 2323 2324 2325 2327 2396 2412 2453 2454 2456 2491 2492 2493
				2631 2640 2642 2647 2650 2657 2669 2675 2677 2684 2763 2818 2819 2820 2833
				2866 2868 2877 2878 2879 2880 2889 2951 3044 3052 3066 3084 3093 3107 3118
				3120 3120 3123 3124 3157 3186 3188 3191 3249 3251 3255 3256 3258 3259 3273

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

```
                                    3275  3284  3285  3311  3312  3313  3324  3326  3327  3328  3330  3331  3333  3334  3338
                                    3340  3343  3344  3346  3349  3352  3363  3364  3369  3370  3380  3381  3397  3398  3401
                                    3413  3414  3415  3416  3419  3431  3432  3434  3437  3439  3440  3455  3459  3459  3461
                                    3463  3464  3464  3465  3467  3485  3486  3488  3489  3490  3496  3497  3506  3511  3512
                                    3513  3515  3519  3525  3532  3562  3577  3592  3601  3608  3611  3618  3622  3623  3667
                                    3668  3670  3671  3672  3681  3682  3683  3714  3723  3724  3726  3736  3765  3767  3806
                                    3863  3865  3868  3871  3959  4159  4161  4302  4305  4388  4390  4765  4770  4770  4771
                                    4776  4780  4811  4827  4831  4844
RQ       00001 00000009  0105       1393  1827  1832  1852  1857  1859  1860  1868  1869  1871  1891  1892  1900  1900  1901
                                    1903  1935  1944  1945  1947  1951  1953  1974  1976  1981  1984  2026  2030  2036  2040
                                    2044  2047  2067  2070  2073  2090  2110  2117  2119  2124  2167  2186  2198  2224  2225
                                    2227  2324  2326  2327  2328  2341  2453  2455  2490  2494  2517  2532  2545  2776  2820
                                    2828  2871  2873  2878  2882  2891  2892  3044  3046  3047  3053  3082  3085  3094  3127
                                    3128  3169  3188  3189  3191  3192  3194  3195  3207  3250  3252  3261  3309  3310  3313
                                    3324  3354  3369  3372  3373  3399  3400  3401  3417  3418  3419  3431  3433  3434  3435
                                    3437  3451  3456  3461  3465  3469  3482  3483  3484  3492  3496  3506  3511  3514  3515
                                    3516  3519  3520  3522  3523  3524  3525  3529  3563  3637  3638  3639  3726  3728  3729
                                    3733  3738  3753  3766  3767  3786  3787  3791  3795  3799  3803  3931  3950  3956  3957
                                    3972  3988  3989  3991  3992  3993
R1       00001 00000001  0094       0269  0270  0272  0286  0287  0300  0305  0306  0307  0308  0309  0354  0356  0357  0358
                                    0358  0359  0370  0386  0538  0541  0607  0616  0627  0631  0635  0640  1096  1106  1113
                                    1158  1159  1160  1160  1161  1161  1162  2630  2634  2720  2822  2835  2836  2839  2840
                                    2841  2841  2842  2843  2844  2844  2845  2847  2848  2849  2849  2852  2854  2855  2855
                                    2856  2857  2857  2858  2862  2863  2863  3055  3063  3078  3079  3096  3104  3117  3118
                                    4175  4314  4320  4321  4323  4466  4471  4495  4500  4504  4506  4512  4527  4688  4691
                                    4695  4699  4812  4823  4931  4963  4968  4970  5015  5016  5018  5018
R11      ****UNDEFINED****          0271  0271  0272  0278
R12      ****UNDEFINED****          0270  0276  0281
R2       00001 00000002  0095       2823  2826  2890  3051  3077  3077  3079  3105  3316  3317  3487  3530  3530  3564  4813
                                    4826  4827  4835  4932  5014  5016
SATT     00002   000008  0222       4925
SAVUCB   ****UNDEFINED****          0375
SEE1     00004 000004F8  0978       0919
SF32     00004   001138  2395       1573
SHIFT    00002   0019E8  3482       3561
SHIFTA   00002   001A26  3499       3502
SHIFTB   00002   001A8A  3527       3505
SHIFTC   00002   001A72  3519       3510
SHIFTE   00002   001A5C  3513       3517
SHIFTF   00004   001B62  3617       3518  3528  3581  3598  3612
SHIFTS   00002   0019F2  3485       3641
SIMB1    00001 00000003  0096       1089  4916  4917  4917  4918  4918
SIMB2    00001 00000004  0097       1089  4886  4918
SIZE     00001 00000007  0101       0720  1328  1871  1873  1898  1901  3736  3738  4771  4818  4934  4935
SKIP1    00010 0000237F  4713       4666
SKIP1A   00004   00231A  4676       4673
SKIP12   00010 0000238C  4715       4676
SKIP9    00010 00002387  4714       4671
SKIP9A   00004   00230A  4671       4668
SM12     00004   0011EC  2544       1541
SPSKBT   00004   0020EC  4453       4463  4649  4650
SPSKIP   00002   0022F8  4665       4489  4662
START    00004   000818  1162       1156  1299  1338  1341
STC      00002   000F58  2176       0958  0959  1930  1931  1936  1937  1938  1941  1949  1970  1971  2171  2172  2718  2719
                                    3879  3880
```

SYMBOL LEN VALUE DEFN REFERENCES ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES
SW	00001	000900	1347	1163 1333 1342
SWBB42	00004	00110C	2360	2330 2369 3743
S22	00004	0011D6	2531	1557
S22A	00006	0011DA	2532	2546
TABIMM	00010	00237A	4711	4642 4713 4714 4715
TABLE1	00013	00058A	0834	0506 0579 4073 4162 4278 4382 4444 4590
TABLE2	00008	0005D5	0850	4015 4225
TABLE3	00003	000623	0860	4123 4346 4557
TABQ10	00008	002372	4709	4641
TAPE43	00001	000007	0215	1244 2301 2302 2303 4490 4497 4523 4526 4669 4674 4679 4700
TBDNCD	00003	0020C8	4395	4383
TBDNTY	00003	001F54	4171	4163
TBPRA	00003	002240	4566	4558
TBPRD	00008	002270	4598	4591
TBPRN	00003	002210	4536	4445
TBRACD	00003	001FB6	4230	4226
TBRATY	00005	001E58	4021	4016
TBRNCD	00003	000358	0650	0580
TBRNTY	00003	0005AE	0840	0507
TBWACD	00003	002098	4353	4347
TBWATY	00003	001F0E	4130	4124
TBWNCD	00003	002C78	4330	4279
TBWNTY	00003	001EE6	4103	4074
TDM15	00004	000C70	1827	1544
TD25	00004	000C7C	1831	1560 1829
TESTBR	00004	000F44	2171	2331 2366 2370 3744 3807 3960
TFA	00004	000CA0	1858	1861 3402
TFL06	00004	001932	3396	1529
TFM16	00004	000CBA	1852	1545
TF26	00004	000C96	1856	1561 1854
TNFA	00004	000D9A	1979	1985 1988
TNFB	00004	000DBA	1987	1977
TNF73	00004	000D72	1969	1638
TNSA	00004	000D3E	1943	1940 1948 1952 1954
TNSB	00004	000D6A	1953	1939 1950
TNSC	00004	000D5A	1949	1942
TNSWA	00002	000CCA	1873	1867
TNSWB	00002	000CC0	1868	1874
TNSWR	00002	000CB6	1864	1858 1943 1979 3190
TNS72	00004	000D06	1929	1637
TRNM30	00002	00000A00	1910	1571
TRWR	00004	000CEA	1897	1894
TR31	00004	000CD0	1890	1572 1895
TR31A	00004	000CF4	1900	1906
TR31B	00002	000D00	1905	1899
TYPCSW	****UNDEFINED****			0524
TYPEND	00004	00024A	0531	0526
TYPERR	****UNDEFINED****			0518 4096
TYPIOSV	00004	00005C	0288	0269 0286
TYPSWI	00004	00025E	0536	0522 0529 0533 4100 4301 4492
UNTCHK	00001	00016B	0420	4693
VALAAB	00004	00049E	0712	0705
VALIN	00004	00047A	0702	0531 0602
VALOUT	00004	0023A2	4762	4089 4286 4474
VALOU1	00002	0023E8	4780	4772

ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES													
VALOU2	00004	0023CA	4773	4783													
VAL1	00004	000484	0705	0510	0702	0723											
VAL2	00004	0004C4	0722	0711	0727												
VAWR	00002	0004CE	0726	0721													
VVV1	00004	00407A	4957	4951													
VVV2	00002	004060	4950	4956													
VV4	00006	00407E	4958	4957													
WAA	00002	00001FE6	4857	4858													
WAA1	00002	000001A2	4858	4859													
WAA2	00002	000061A2	4859	4860													
WACD	00004	00207C	4346	1023													
WAPT	00002	00000A00	4724	1022													
WATAB	00004	000758	1020	0973													
WATY	00004	001EF6	4123	1021													
WA39	00004	0006C6	0970	1580													
WCDCCW	00008	002070	4329	4291													
WDI1	00004	000004F8	0979	0942													
WHCC	****UNDEFINED****			4973	5008												
WHDEV	****UNDEFINED****			4971	4977	4983											
WHDL	****UNDEFINED****			4964													
WHEK	****UNDEFINED****			4990													
WHESTC	****UNDEFINED****			4979	4985												
WHHH	****UNDEFINED****			4972	4996	5001	5007										
WHRO	****UNDEFINED****			4974	4975	5002	5009	5010									
WHRW	****UNDEFINED****			4980	4991												
WHSKEW	****UNDEFINED****			4981	4992												
WHVDSK	****UNDEFINED****			4970													
WNAA	00004	00000072	4862	4863													
WNAA1	00004	00006072	4863	4864													
WNCD	00004	001FBA	4278	1013													
WNCDIR	00004	002060	4323	4316													
WNCD1	00004	001FD2	4284	4307	4352	4394											
WNCD2	00002	001FE4	4289	4857													
WNCERR	00001	001FEC	4292	4317													
WNCPSW	00008	001FF8	4295	4310	4313												
WNLOCK	00004	00200C	4299	4298	4309												
WNPT	00002	00000A00	4723	1012													
WNTAB	00004	000730	1010	0944													
WNTY	00004	001E7A	4073	1011													
WNTYAL	00004	001F5C	4175	4095	4097												
WNTYGO	00004	001E9B	4081	4099													
WNTYG2	00002	001EE0	4101	4084													
WNTYG3	00004	001EAE	4088	4102													
WNTY1	00004	001E8E	4078	4128	4167												
WNTY3	00004	001EB6	4091	4088	4176	4862											
WN38	00004	000694	0941	1579													
WRICHK	00001	0001B9	0437	1103													
WRISWI	00004	001ED8	4099	4081	4087	4093											
WRITAB	00005	0006EC	0995	0920	0943	0972	0991										
WR1	00001	0000000A	0106	0507	0580	0703	0710	0712	0722	0722	0816	1052	1054	1323	1324	1325	1325	1327
				1327	1384	1391	1398	1400	1401	1402	1429	1432	1435	1933	1934	1937	2109	2118
				2118	2229	2231	2233	2235	2235	2236	2320	2321	2430	2431	2432	2454	2456	2493
				2499	2632	2643	2645	2658	2665	2672	2707	2719	2763	2826	2832	2869	2876	2885
				2894	2942	2944	2945	2947	3052	3056	3058	3059	3064	3084	3086	3090	3093	3099
				3102	3103	3105	3106	3123	3126	3134	3138	3157	3256	3323	3325	3334	3338	3340

ASM H V 05 22.10 05/07/81

SYMBOL	LEN	VALUE	DEFN	REFERENCES
				3342 3411 3412 3451 3452 3453 3455 3456 3458 3458 3489 3562 3586 3593 3599
				3617 3618 3657 3658 3660 3660 3661 3672 3673 3674 3675 3676 3677 3678 3683
				3684 3685 3686 3687 3712 3713 3853 3868 3951 3952 3955 3987 3990 4016 4074
				4124 4163 4226 4279 4304 4305 4347 4383 4445 4558 4591 4637 4640 4641 4641
				4642 4642 4644 4762 4767 4768 4773 4773 4925 4926 4936 4944 4946 4952 4953
				4953 4958 4970 4971 4987 4988 4997 4999
WR2	00001	0000000B	0107	0506 0579 0704 0712 0713 0713 0714 0718 0805 0807 0809 0810 0812 0812 0814
				0814 0816 0822 1324 1326 1326 1328 1330 1428 1431 1431 1434 1434 1437 1437
				1438 1438 1442 1442 1443 1443 1445 2230 2231 2232 2233 2234 2494 2633 2636
				2638 2681 2703 2832 2837 2845 2846 2850 2858 2861 2862 2875 3053 3057 3061
				3066 3067 3069 3070 3072 3074 3076 3085 3088 3091 3094 3097 3100 3264 3266
				3266 3268 3269 3270 3270 3272 3321 3323 3326 3329 3330 3355 3356 3366 3491
				3494 3495 3503 3503 3508 3508 3509 3520 3523 3527 3529 3563 3568 3569 3570
				3570 3571 3573 3574 3593 3595 3600 3601 3603 3603 3611 3614 3614 3715 3731
				3764 3765 3766 3864 3864 3865 3952 3953 3954 3956 3957 3958 3991 3992 4015
				4073 4123 4162 4225 4278 4346 4382 4444 4486 4486 4487 4488 4488 4557 4590
				4638 4643 4644 4644 4647 4648 4649 4652 4667 4672 4677 4763 4765 4766 4766
				4767 4778 4948 4950 4955 4986 4988 4989 4990 4993 4993 4994 4995 4995 4996
				4998 4998 4999 5001 5002 5011
WR3	00001	0000000C	0108	0801 0804 0819 0822 1446 1447 2236 2237 2241 2628 2650 2652 2654 2656 2661
				2662 2664 2665 2667 2668 2669 2670 2672 2674 2707 2708 2708 2709 2710 2713
				2716 2717 2718 2827 2828 2829 2829 2830 2830 2835 2847 3049 3074 3075 3076
				3107 3108 3110 3111 3113 3114 3116 3254 3269 3271 3271 3272 3273 3371 3375
				3376 3493 3494 3495 3497 3499 3501 3575 3576 3579 3594 3597 3597 3613 3613
				3722 3731 3885 3886 4077 4127 4166 4282 4350 4446 4559 4592 4665 4665 4666
				4667 4671 4672 4676 4677 4770 4811 4817 4825 4828 4836 4945 4946 4947 4950
				4955 4957 4965 4966 4966 4967
WR4	00001	0000000D	0109	0523 0525 0530 0601 0723 0802 0809 0813 0814 0920 0920 0921 0921 0922 0932
				0932 0933 0933 0934 0943 0943 0944 0944 0945 0956 0956 0957 0957 0962 0972
				0972 0973 0973 0991 0991 0992 0992 0993 1049 1383 1386 1390 1393 1400 1404
				1447 1449 1450 1452 1453 2654 2655 2655 2656 2703 2705 2705 2706 2709 2712
				2713 2716 2833 2834 2834 2835 2847 3050 3062 3101 3114 3115 3116 3122 3124
				3125 3260 3263 3267 3341 3342 3343 3345 3346 3347 3353 3356 3357 3374 3375
				3498 3500 3500 3509 3527 3578 3579 3599 3600 3604 3604 3606 3606 3607 3608
				3658 3661 3663 4079 4086 4088 4101 4199 4200 4285 4386 4387 4388 4392 4469
				4469 4471 4473 4504 4506 4509 4774 4811 4815 4817 4817 4818 4823 4825 4826
				4828 4831 4834 4835 4837 4842 4842 4844 4949 4952 4954 4954 4958
WR5	00001	0000000E	0110	0524 0525 0527 0527 0530 0801 0803 0810 0811 0819 0918 0930 0941 0954 0970
				0989 1052 1054 1056 1057 1330 2503 2628 2651 2663 2663 2710 2711 2715 2840
				2843 2854 2856 2859 2859 2984 2987 2991 3054 3065 3073 3073 3075 3092 3092
				3115 3262 3263 3566 3566 3876 4079 4083 4085 4101 4157 4158 4159 4161 4197
				4200 4202 4202 4204 4302 4390 4391 4391 4392 4812 4822 4976 4977 4978 4979
				4982 4983 4984 4985
WR6	00001	0000000F	0111	2827 2837 2838 2838 2839 2850 2851 2851 2852 4938

ASM H V 05 22.10 05/07/81

```
THE FOLLOWING STATEMENTS WERE FLAGGED
   00270 00271 00272 00276 00278 00281 00375 00518 00524 00623 01329 01902 04084 04096 04315 04318 04517 04838
   04854 04926 04947 04949 04964 04965 04967 04968 04969 04970 04971 04972 04973 04974 04975 04976 04977 04978
   04979 04980 04981 04982 04983 04984 04985 04986 04989 04990 04991 04992 04994 04996 05001 05002 05004 05007
   05008 05009 05010 05011 05012 05014 05015 05017
     62 STATEMENTS FLAGGED IN THIS ASSEMBLY       8 WAS HIGHEST SEVERITY CODE

   OVERRIDING PARAMETERS-  ALIGN,DECK,LIST,ESD,RLD,XREF(SHORT),FLAG(000),NORENT,NOOBJECT,NOTEST,LINECOUNT(55)
   OPTIONS FOR THIS ASSEMBLY
    DECK, NOOBJECT, LIST, XREF(SHORT), NORENT, NOTEST, NOBATCH, ALIGN, ESD, RLD, LINECOUNT(55), FLAG(0), SYSPARM()
     STANDARD DD NAMES- SYSLIN   SYSLIB   SYSIN    SYSPRINT SYSPUNCH SYSUT1
   OVERRIDING DD NAMES- 00000000 CMSLIB   ASSEMBLE LISTING  TEXT     SYSUT1

    8195 CARDS FROM SYSIN     132 CARDS FROM SYSLIB
    6112 LINES OUTPUT         208 CARDS OUTPUT
```