```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                     2 *Testcase mvsos001:   MVCOS
                                                     3 * Created and placed into the public domain
                                                     4 * 27 JAN 2021 by Bob Polmanter.

                              00000000  00000001     6 R0       EQU   0          General Purpose Registers
                              00000001  00000001     7 R1       EQU   1
                              00000002  00000001     8 R2       EQU   2
                              00000003  00000001     9 R3       EQU   3
                              00000004  00000001    10 R4       EQU   4
                              00000005  00000001    11 R5       EQU   5
                              00000006  00000001    12 R6       EQU   6
                              00000007  00000001    13 R7       EQU   7
                              00000008  00000001    14 R8       EQU   8
                              00000009  00000001    15 R9       EQU   9
                              0000000A  00000001    16 R10      EQU   10
                              0000000B  00000001    17 R11      EQU   11
                              0000000C  00000001    18 R12      EQU   12
                              0000000D  00000001    19 R13      EQU   13
                              0000000E  00000001    20 R14      EQU   14
                              0000000F  00000001    21 R15      EQU   15
                              00000000  00000001    22 AR0      EQU   0          Access Registers
                              00000001  00000001    23 AR1      EQU   1
                              00000002  00000001    24 AR2      EQU   2
                              00000003  00000001    25 AR3      EQU   3
                              00000004  00000001    26 AR4      EQU   4
                              00000005  00000001    27 AR5      EQU   5
                              00000006  00000001    28 AR6      EQU   6
                              00000007  00000001    29 AR7      EQU   7
                              00000008  00000001    30 AR8      EQU   8
                              00000009  00000001    31 AR9      EQU   9
                              0000000A  00000001    32 AR10     EQU   10
                              0000000B  00000001    33 AR11     EQU   11
                              0000000C  00000001    34 AR12     EQU   12
                              0000000D  00000001    35 AR13     EQU   13
                              0000000E  00000001    36 AR14     EQU   14
                              0000000F  00000001    37 AR15     EQU   15
                              00000000  00000001    38 CR0      EQU   0          Control Registers
                              00000001  00000001    39 CR1      EQU   1
                              00000002  00000001    40 CR2      EQU   2
                              00000003  00000001    41 CR3      EQU   3
                              00000004  00000001    42 CR4      EQU   4
                              00000005  00000001    43 CR5      EQU   5
                              00000006  00000001    44 CR6      EQU   6
                              00000007  00000001    45 CR7      EQU   7
                              00000008  00000001    46 CR8      EQU   8
                              00000009  00000001    47 CR9      EQU   9
                              0000000A  00000001    48 CR10     EQU   10
                              0000000B  00000001    49 CR11     EQU   11
                              0000000C  00000001    50 CR12     EQU   12
                              0000000D  00000001    51 CR13     EQU   13
                              0000000E  00000001    52 CR14     EQU   14
                              0000000F  00000001    53 CR15     EQU   15
```

```
                                                    55 ***********************************************************************
                                                    56 *
                                                    57 *    These tests and this programming were validated on a z114
                                                    58 *    using a z/VM 6.4 virtual machine on 27 January 2021.
                                                    59 *
                                                    60 ***********************************************************************
                                                    61 *
                                                    62 * Tests performed with MVCOS:
                                                    63 *
                                                    64 * 1.  Execute the MVCOS instruction iteratively, each time trying a
                                                    65 *     different combination of machine state, address space control
                                                    66 *     mode, MVCOS operand 1 control modes, MVCOS operand 2 control
                                                    67 *     modes, and key enablement in both operand 1 and then operand 2.
                                                    68 *     These individual tests are nested in a series of loops, so that
                                                    69 *     each state or mode is tested with each other combination of
                                                    70 *     states or modes in an exhaustive way.  A visual description of
                                                    71 *     the nested loops and their related tests is shown below.
                                                    72 *
                                                    73 *     After execution of each MVCOS instruction, the results of the
                                                    74 *     actual data moved are checked to determine if the data fetched
                                                    75 *     indeed came from the specified address space, and if the data
                                                    76 *     stored was indeed placed into the specified address space, as
                                                    77 *     determined by the settings in R0 for operand 2 and operand 1,
                                                    78 *     respectively.
                                                    79 *
                                                    80 * Upon success this is the number of tests performed:
                                                    81 *   Successfully completed MVCOS executions:             384
                                                    82 *   Expected protection check events:                  1,152
                                                    83 *   Expected special operation exception events:          92
                                                    84 *   TOTAL TESTS:                                       1,328
                                                    85 *
                                                    86 * Test Success:   Disabled Wait PSW X'000'
                                                    87 * Test Failure:   Disabled Wait PSW X'BAD'
                                                    88 *
                                                    89 * Unexpected program check:  Disabled Wait PSW X'00F'
                                                    90 *
                                                    91 *
                                                    92 * The expected protection checks arise from enabling key controlled
                                                    93 * protection in register 0 as specified by the instruction. Keys
                                                    94 * have been deliberately set to allow some accesses and to fail some
                                                    95 * access attempts. Because either MVCOS operands (or both) could be
                                                    96 * using a failing key, there is more of these failures, whereas
                                                    97 * success requires both operands to have the right key setting.
                                                    98 *
                                                    99 * The expected special operation exceptions arise when the Home
                                                   100 * address space control mode is selected in Register 0 -AND- the
                                                   101 * PSW is in the problem state. This applies to operand 1 OAC only.
                                                   102 * This tests that MVCOS is honoring that specification as documented
                                                   103 * in the Principles.
                                                   104 *
                                                   105 ***********************************************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   107 ***********************************************************************
                                                   108 *
                                                   109 *                           TEST METHOD
                                                   110 *
                                                   111 * 1. Setup.
                                                   112 *
                                                   113 * Three address spaces are created: Primary, Secondary, and Home.
                                                   114 * Literals are placed into a page frame belonging to each one of the
                                                   115 * address spaces identifying them by PRI, SEC, and HOM. Those pages
                                                   116 * are the targets of MVCOS operand 1.  A second set of
                                                   117 * literals is placed into another page frame belonging to each
                                                   118 * one of the address spaces identifying them also as FROMPRI, FROMSEC,
                                                   119 * and FROMHOM. These pages are fetched by MVCOS operand 2.
                                                   120 *
                                                   121 * The literals in the target page frames of each address space are
                                                   122 * located at virtual 00010FF0 in each space.
                                                   123 *
                                                   124 * The literals in the 'from' page frames of each address space are
                                                   125 * located at virtual 00012FF8 in each space.
                                                   126 *
                                                   127 * The locations are set to cause MVCOS to move data across page
                                                   128 * boundaries.
                                                   129 *
                                                   130 * The nested loops are entered to set the register 0 MVCOS controls.
                                                   131 *
                                                   132 *
                                                   133 * 2. Executing MVCOS.
                                                   134 *
                                                   135 * A FROM literal is moved to the target page by MVCOS. The address
                                                   136 * space fetched from and the address space target are of course
                                                   137 * determined by the MVCOS controls in Register 0.
                                                   138 *
                                                   139 *
                                                   140 * 3. Validation.
                                                   141 *
                                                   142 * After the MVCOS, the register 0 controls are extracted and used
                                                   143 * to determine programatically which address space literal was
                                                   144 * requested to be moved to which target, and the results are compared
                                                   145 * to determine if those literals are actually where they are supposed
                                                   146 * to be.
                                                   147 *
                                                   148 * After successful validation, the original placement of the literals
                                                   149 * is restored, and the next loop iteration advances to the next test.
                                                   150 *
                                                   151 ***********************************************************************
                                                   153 ***********************************************************************
                                                   154 *
                                                   155 *                    DEBUGGING THIS TEST PROGRAM
                                                   156 *
                                                   157 * If any MVCOS test fails (data from the specified address space is
                                                   158 * not identified as expected), the machine will be halted to preserve
                                                   159 * results and a disabled wait PSW of X'BAD' will be loaded. Use
                                                   160 * register 0 and the address space control value in byte PSWASC to
                                                   161 * determine what should have been moved to where.  View the literals
                                                   162 * at virtual location X'00010FF0' to determine which space you are
                                                   163 * viewing and what FROM literal was actually  moved to that space.
                                                   164 * Compare that to the R0 controls and PSWASC mode to validate whether
                                                   165 * MVCOS moved correctly.  You can also use the memory map listed
                                                   166 * below to view the real pages by address to inspect the literals in
                                                   167 * each address space.
                                                   168 *
                                                   169 * If an unexpected program check occurs, the PSW will be loaded with
                                                   170 * a disabled wait code X'00F'. The machine is halted immediately
                                                   171 * upon occurance and no registers are altered; hence their values
                                                   172 * reflect the state of the failure.
                                                   173 *
                                                   174 *
                                                   175 ***********************************************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   177 ***********************************************************************
                                                   178 *
                                                   179 *                    Memory Map - REAL STORAGE
                                                   180 *
                                                   181 *         Hex
                                                   182 * RAddr   Len     Description
                                                   183 * -----   ----    -----------------------------------------------------
                                                   184 *     0   2000  - Absolute page 0 and 1
                                                   185 *  2000   2000  - Program code
                                                   186 *  4000   1000  - Segment table, primary space
                                                   187 *  5000   1000  - Segment table, secondary space
                                                   188 *  6000   1000  - Segment table, home space
                                                   189 *  7000    800  - Page tables, primary space
                                                   190 *  7800    800  - Page tables, secondary space
                                                   191 *  8000    800  - Page tables, home space
                                                   192 *  9000   1000  - Primary ASTE, DUCT, DU-AL, ALE blocks
                                                   193 *  A000   1000  - Home ASTE block
                                                   194 *  B000   5000  - unused; available
                                                   195 * 10000  10000  - Pages backing Home virtual space at vaddr 10000
                                                   196 * 20000  10000  - Pages backing Primary virtual space at vaddr 10000
                                                   197 * 30000  10000  - Pages backing Secondary virtual space at vaddr 10000
                                                   198 *
                                                   199 *
                                                   200 *                    Memory Map - VIRTUAL STORAGE
                                                   201 *
                                                   202 * VAddr   Len   Raddr Key   Description
                                                   203 * -----   ----   ---- ---   ---------------------------------------
                                                   204 * 00000  10000  00000  00 - Common V=R storage (all address spaces)
                                                   205 * 10000  10000  10000  00 - Home space storage
                                                   206 * 10000  10000  20000  40 - Primary space storage
                                                   207 * 10000  10000  30000  80 - Secondary space storage
                                                   208 *
                                                   209 *
                                                   210 ***********************************************************************
                                                   211 *                    IN EACH ADDRESS SPACE:
                                                   212 *
                                                   213 * VADDR 10FF0 length 32:  Literal identifying the space target
                                                   214 *                          (e.g., CL16'PRI-PG1',CL16'PRI-PG2'
                                                   215 *
                                                   216 * VADDR 12FF8 length 16:  Literal identifying the space source
                                                   217 *                          (e.g., CL16'FROMPRI1FROMPRI2'
                                                   218 *
                                                   219 * After a successful MVCOS, the storage at location 10FF0 would look
                                                   220 * like this example with a move from secondary to primary:
                                                   221 *
                                                   222 * VADDR 10FF0   CL32'PRI-PG1FROMSEC1FROMSEC2         '
                                                   223 *
                                                   224 * thus showing that the target area is still named PRI,and the data
                                                   225 * came from two pages in the secondary space.
                                                   226 *
                                                   227 *
                                                   228 ***********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   230 ***********************************************************************
                                                   231 *
                                                   232 *             VISUAL DESCRIPTION OF NESTED LOOP TESTS
                                                   233 *
                                                   234 * The sequence of loops nested below allows each combination to be
                                                   235 * tested one at a time.  OAC1 and OAC2 are the Operand Access
                                                   236 * Control 1 and 2, respectively, in Register 0 that control MVCOS.
                                                   237 *
                                                   238 * Loop  iterations  Description
                                                   239 *    1       2      Supervisor state, then problem state
                                                   240 *    2       4      Cycle through each PSW ASC mode P,AR,S,H
                                                   241 *    3       2      OAC1 A validity bit off, then on
                                                   242 *    4       4      When OAC1 A=1, cycle through each ASC mode in OAC1
                                                   243 *    5       3      When OAC1 A=1 & OAC1 is AR, cycle ALETs 0,1,2 oper1
                                                   244 *    6       2      OAC2 A validity bit off, then on
                                                   245 *    7       4      When OAC2 A=1, cycle through each ASC mode in OAC2
                                                   246 *    8       3      When OAC2 A=1 & OAC2 is AR, cycle ALETs 0,1,2 oper2
                                                   247 *    9       2      OAC1 K validity bit off, then on
                                                   248 *   10       2      OAC2 K validity bit off, then on
                                                   249 *
                                                   250 *                   Execute MVCOS
                                                   251 *
                                                   252 *                   Check results; PSW=X'BAD' if failed
                                                   253 *                   Next loop 10
                                                   254 *                   Next loop 9
                                                   255 *                   Next loop 8
                                                   256 *                   Next loop 7
                                                   257 *                   Next loop 6
                                                   258 *                   Next loop 5
                                                   259 *                   Next loop 4
                                                   260 *                   Next loop 3
                                                   261 *                   Next loop 2
                                                   262 *                   Next loop 1
                                                   263 *                   Terminate with success, PSW=X'000'
                                                   264 *
                                                   265 * Note on loop 4 & 7: when either OAC A validity bit is 0, then the
                                                   266 * PSW ASC mode is used by MVCOS and the iteration count is 1.
                                                   267 *
                                                   268 * Note on loop 5 & 8: when either OAC A validity bit is 0 -AND- the
                                                   269 * PSW ASC mode is P,S,or H, then the PSW ASC mode is used by MVCOS and
                                                   270 * the iteration count is 1.  However, if the PSW ASC mode is AR, then
                                                   271 * the iteration count remains 3 for these loops so the ALETs can be
                                                   272 * cycled through each test for each operand 1 or operand 2, in turn.
                                                   273 *
                                                   274 ***********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   276 ***********************************************************************
                                                   277 *                     Low Core / Prefix Area
                                                   278 ***********************************************************************
                                                   279 *
                              00000000  0000A03F   280 MVCOS001 START 0
                              00000000  00000001   281 STRTLABL EQU   *
00000000                      00000000             282          USING STRTLABL,0


                                                   284 *             Selected z/Arch low core layout
                                                   285 *
00000000                      00000000  00000088   286          ORG   STRTLABL+X'88'      interrupt code area EC mode
00000088  00000000                                 287 SVCINTC  DC    X'00000000'         SVC interrupt code area
0000008C  00000000                                 288 PGMINTC  DC    X'00000000'         Prog check interrupt code area
                                                   289 *
00000090                      00000090  00000140   290          ORG   STRTLABL+X'140'
00000140  00000000 00000000                        291 SVCOPSW  DS    XL16                SVC old PSW
00000150  00000000 00000000                        292 PGMOPSW  DS    XL16                Program check old PSW
                                                   293 *
00000160                      00000160  000001A0   294          ORG   STRTLABL+X'1A0'     New PSWs
000001A0  00000000 80000000                        295 RESTART  DC    X'00000000',X'80000000',A(0),A(START)    DAT OFF
000001B0  00000000 00000000                        296 EXTNPSW  DC    XL16'00'
000001C0  04004000 80000000                        297 SVCNPSW  DC    X'04004000',X'80000000',A(0),A(SVCFLIH)  DAT ON, AR MODE
000001D0  04004000 80000000                        298 PGMNPSW  DC    X'04004000',X'80000000',A(0),A(PGMFLIH)  DAT ON, AR MODE


                                                   300 *                     Test Counters
                                                   301 *
000001E0                      000001E0  00000200   302          ORG   STRTLABL+X'200'     Test counters
00000200  0000000C                                 303 MVCOSOK  DC    PL4'0'               # of successful tests
00000204  0000000C                                 304 PIC04    DC    PL4'0'               # of Pchecks 04
00000208  0000000C                                 305 PIC13    DC    PL4'0'               # of Pchecks 13
0000020C  00000000                                 306          DC    F'0'
                                                   307 *
  LOC        OBJECT CODE        ADDR1     ADDR2    STMT

                                                   309 ***********************************************************************
                                                   310 *                       Main program
                                                   311 ***********************************************************************
                                                   312 *
00000210                      00000210  00002000   313          ORG   STRTLABL+X'2000'
00002000  0DF0                                     314 START    BASR  R15,0
00002002  06F0                                     315          BCTR  R15,0
00002004  06F0                                     316          BCTR  R15,0
00002006                      00002000             317          USING START,R15
                                                   318 *
00002006  1B22                                     319          SR    R2,R2                   STATUS REG SET TO 0
00002008  4130 0001                     00000001   320          LA    R3,1                    R3=1 MEANS SET Z/ARCH MODE
                                                   321 *                                      R3=0 MEANS SET ESA/390 MODE
0000200C  1B44                                     322          SR    R4,R4                   CPU Addr = 0
0000200E  AE24 0012                     00000012   323          SIGP  R2,R4,X'12'             X'12' = SET ARCHITECTURE
                                                   324 *
00002012  4100 0008                     00000008   325          LA    R0,X'08'                Set KEY=0 fetch prot enabled
00002016  4120 0040                     00000040   326          LA    R2,64                   # of real pages to set
0000201A  1B11                                     327          SR    R1,R1                   Starting addr
                                                   328 *
0000201C  B22B 0001                                329 SET000   SSKE  R0,R1                   Set the key
00002020  A71A 1000                                330          AHI   R1,4096                 Bump to next page
00002024  4620 F01C                     0000201C   331          BCT   R2,SET000
                                                   332 *
00002028  9A0F F4F8                     000024F8   333          LAM   AR0,AR15,AREGS          Clear all ARs
0000202C  EB0F F478 002F                00002478   334          LCTLG CR0,CR15,CREGS          Load all the CRs
00002032  8000 F3B4                     000023B4   335          SSM   =X'04'                  Turn on DAT
                                                   336 *
00002036  5850 F544                     00002544   337          L     R5,VADDRTO              Get vaddr in 1st virtual page
0000203A  5860 F548                     00002548   338          L     R6,VADDRFRM             Copy
                                                   339 *
0000203E  D21F 5000 F3B8      00000000  000023B8   340          MVC   0(32,R5),PRIPG1         Set literal identifier in pages
00002044  D20F 6000 F418      00000000  00002418   341          MVC   0(16,R6),FROMPRI        Set literal identifier in pages
                                                   342 *
0000204A  B219 0100                     00000100   343          SAC   SECMODE                 Secondary mode
0000204E  D21F 5000 F3D8      00000000  000023D8   344          MVC   0(32,R5),SECPG1         Set literal identifier in pages
00002054  D20F 6000 F428      00000000  00002428   345          MVC   0(16,R6),FROMSEC        Set literal identifier in pages
                                                   346 *
0000205A  B219 0300                     00000300   347          SAC   HOMEMODE                Home space mode
0000205E  D21F 5000 F3F8      00000000  000023F8   348          MVC   0(32,R5),HOMPG1         Set literal identifier in pages
00002064  D20F 6000 F438      00000000  00002438   349          MVC   0(16,R6),FROMHOM        Set literal identifier in pages
                                                   350 *
0000206A  B219 0200                     00000200   351          SAC   ARMODE                  Enter AR mode
0000206E  5800 F344                     00002344   352          L     R0,=X'10031003'         Initialize MVCOS controls; on
                                                   353 *                                       first pass below this will be
                                                   354 *                                       set to X'00000000' for 1st test
00002072  920C F561                     00002561   355          MVI   PSWASC,X'0C'            Initialize PSW ASC ctl byte; on
                                                   356 *                                       first pass below this will be
                                                   357 *                                       set to X'00' (PRI) for 1st test
00002076  9201 F560                     00002560   358          MVI   PSWSTATE,X'01'          Initialize PSW state control; on
                                                   359 *                                       first pass below this will be
                                                   360 *                                       set to X'00',SUPRV for 1st test
  LOC       OBJECT CODE       ADDR1     ADDR2    STMT

                                                   362 ***********************************************************************
0000207A  41E0 0002                     00000002   363          LA    R14,2                   # of PSW state tests
                              0000207E  00000001   364 STATE000 EQU   *
0000207E  9701 F560                     00002560   365          XI    PSWSTATE,X'01'          Flip current PSW state ctl byte
00002082  4310 F560                     00002560   366          IC    R1,PSWSTATE             Get new state ctl byte
00002086  4410 F08E                     0000208E   367          EX    R1,SVCSTATE             Flip to next PSW state
0000208A  47F0 F090                     00002090   368          B     SKIP001                 Continue test prep
                                                   369 *
0000208E  0A00                                     370 SVCSTATE SVC   0                       Executed instruction
                              00002090  00000001   371 SKIP001  EQU   *



                                                   373 ***********************************************************************
00002090  41D0 0004                     00000004   374          LA    R13,4                   # of PSW ASC tests
                              00002094  00000001   375 ASC010   EQU   *
00002094  4310 F561                     00002561   376          IC    R1,PSWASC               Get last mode used
00002098  4110 1004                     00000004   377          LA    R1,X'04'(,R1)           Increment bit 5 to next ASC mode
0000209C  5410 F348                     00002348   378          N     R1,=X'0000000C'         Keep only bits 4 and 5
000020A0  4210 F561                     00002561   379          STC   R1,PSWASC               Set new mode to use



                                                   381 ***********************************************************************
000020A4  41C0 0002                     00000002   382          LA    R12,2                   # From A validity tests
                              000020A8  00000001   383 TOA000   EQU   *
000020A8  1810                                     384          LR    R1,R0                   Copy control bits
000020AA  5410 F34C                     0000234C   385          N     R1,=A(OAC1A)            Keep only bits we want
000020AE  5710 F34C                     0000234C   386          X     R1,=A(OAC1A)            Flip these bits
000020B2  5400 F350                     00002350   387          N     R0,=A(X'FFFFFFFF'-OAC1A) Force these bits off
000020B6  1601                                     388          OR    R0,R1                   Set ctl based on flip results
000020B8  5400 F354                     00002354   389          N     R0,=A(X'FFFFFFFF'-(ASCHOM*65536)) Force bits off in OAC1
000020BC  41B0 0001                     00000001   390          LA    R11,1                   Assume 1 test if A=0 (using PSW)
000020C0  A700 0001                                391          TMLH  R0,ASCA                 Was A set on or off?
000020C4  4780 F0FE                     000020FE   392          BZ    TOAS200                 A is off, use PSW ASC
                                                   393 *
                                                   394 *                                      A=1: rotate thru OAC1AS modes
000020C8  5600 F358                     00002358   395          O     R0,=A(ASCHOM*65536)     Force on; will wrap to 00 next
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   397 ***********************************************************************
000020CC  41B0 0004                     00000004   398          LA    R11,4                    4 test to rotate thru OAC1 ASCs
                              000020D0  00000001   399 TOAS000  EQU   *
000020D0  A700 0001                                400          TMLH  R0,ASCA                  Was A set on or off?
000020D4  4780 F0FE                     000020FE   401          BZ    TOAS200                  A is off, use PSW ASC
                                                   402 *                                       A is on, do OAC1 AS changes
000020D8  1810                                     403          LR    R1,R0                    Copy control bits
000020DA  5A10 F35C                     0000235C   404          A     R1,=X'00400000'          Increment bit to next ASC mode
000020DE  5410 F358                     00002358   405          N     R1,=A(ASCHOM*65536)      Keep only the bits we want
000020E2  5400 F354                     00002354   406          N     R0,=A(X'FFFFFFFF'-(ASCHOM*65536)) Force bits off in OAC1
000020E6  1601                                     407          OR    R0,R1                    Set ctl based on flip results
                                                   408 *
000020E8  41A0 0001                     00000001   409          LA    R10,1                    1 test required if ASC is P,S,H
000020EC  1810                                     410          LR    R1,R0                    Copy control bits
000020EE  5410 F360                     00002360   411          N     R1,=A(OAC1A+ASCHOM*65536) Keep only these bits
000020F2  5510 F364                     00002364   412          CL    R1,=A(OAC1A+ASCAR*65536) Using MVCOS control AR mode?
000020F6  4770 F128                     00002128   413          BNE   TOAS290                  No. Use 1 test in R10 for P,S,H
000020FA  47F0 F10A                     0000210A   414          B     TOAS210                  Yes. 3 tests in R10 for AR
                                                   415 *
                              000020FE  00000001   416 TOAS200  EQU   *
000020FE  41A0 0001                     00000001   417          LA    R10,1                    1 test required if PSW is P,S,H
00002102  9504 F561                     00002561   418          CLI   PSWASC,X'04'             Using ASC=AR ?
00002106  4770 F128                     00002128   419          BNE   TOAS290                  No. only 1 test per ASC mode
                                                   420 *
                              0000210A  00000001   421 TOAS210  EQU   *



                                                   423 ***********************************************************************
0000210A  41A0 0003                     00000003   424          LA    R10,3                    3 tests required for ASC=AR
                              0000210E  00000001   425 TOAS220  EQU   *
0000210E  5810 F540                     00002540   426          L     R1,TALET                 Get from ALET
00002112  B24E 0051                                427          SAR   AR5,R1                   Set in from AR
00002116  4110 1001                     00000001   428          LA    R1,1(,R1)                Bump ALET
0000211A  5910 F368                     00002368   429          C     R1,=F'3'                 Exceeded max of 2?
0000211E  4740 F124                     00002124   430          BL    TOAS230                  No
00002122  1B11                                     431          SR    R1,R1                    Restart back at ALET 0
                                                   432 *
                              00002124  00000001   433 TOAS230  EQU   *
00002124  5010 F540                     00002540   434          ST    R1,TALET                 Save updated ALET
                                                   435 *
                              00002128  00000001   436 TOAS290  EQU   *
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   438 ***********************************************************************
00002128  4190 0002                     00000002   439          LA    R9,2                    # From A validity tests
                              0000212C  00000001   440 FRMA000  EQU   *
0000212C  1810                                     441          LR    R1,R0                   Copy control bits
0000212E  5410 F36C                     0000236C   442          N     R1,=A(OAC2A)            Keep only bits we want
00002132  5710 F36C                     0000236C   443          X     R1,=A(OAC2A)            Flip these bits
00002136  5400 F370                     00002370   444          N     R0,=A(X'FFFFFFFF'-OAC2A) Force these bits off
0000213A  1601                                     445          OR    R0,R1                   Set ctl based on flip results
0000213C  5400 F374                     00002374   446          N     R0,=A(X'FFFFFFFF'-ASCHOM) Force these bits off in OAC2
00002140  4180 0001                     00000001   447          LA    R8,1                    Assume 1 test if A=0 (using PSW)
00002144  A701 0001                                448          TMLL  R0,ASCA                 Was A set on or off?
00002148  4780 F182                     00002182   449          BZ    FRMAS200                A is off, use PSW ASC
                                                   450 *
                                                   451 *                                      A=1: rotate thru OAC2AS modes
0000214C  5600 F378                     00002378   452          O     R0,=A(ASCHOM)           Force on; will wrap to 00 next


                                                   454 ***********************************************************************
00002150  4180 0004                     00000004   455          LA    R8,4                    4 test to rotate thru OAC2 ASCs
                              00002154  00000001   456 FRMAS000 EQU   *
00002154  A701 0001                                457          TMLL  R0,ASCA                 Was A set on or off?
00002158  4780 F182                     00002182   458          BZ    FRMAS200                A is off, use PSW ASC
                                                   459 *                                      A is on, do OAC2 AS changes
0000215C  1810                                     460          LR    R1,R0                   Copy control bits
0000215E  4110 1040                     00000040   461          LA    R1,B'01000000'(,R1)     Increment bit 1 to next ASC mode
00002162  5410 F378                     00002378   462          N     R1,=A(ASCHOM)           Keep only the bits we want
00002166  5400 F374                     00002374   463          N     R0,=A(X'FFFFFFFF'-ASCHOM) Force these bits off in OAC2
0000216A  1601                                     464          OR    R0,R1                   Set ctl based on flip results
                                                   465 *
0000216C  4170 0001                     00000001   466          LA    R7,1                    1 test required if ASC is P,S,H
00002170  1810                                     467          LR    R1,R0                   Copy control bits
00002172  5410 F37C                     0000237C   468          N     R1,=A(OAC2A+ASCHOM)     Keep only these bits
00002176  5510 F380                     00002380   469          CL    R1,=A(OAC2A+ASCAR)      Using MVCOS control AR mode?
0000217A  4770 F1AC                     000021AC   470          BNE   FRMAS290                No. Use 1 test in R7 for P,S,H
0000217E  47F0 F18E                     0000218E   471          B     FRMAS210                Yes. 3 tests in R7 for AR
                                                   472 *
                              00002182  00000001   473 FRMAS200 EQU   *
00002182  4170 0001                     00000001   474          LA    R7,1                    1 test required if PSW is P,S,H
00002186  9504 F561                     00002561   475          CLI   PSWASC,X'04'            Using ASC=AR ?
0000218A  4770 F1AC                     000021AC   476          BNE   FRMAS290                No. only 1 test per ASC mode
                                                   477 *
                              0000218E  00000001   478 FRMAS210 EQU   *
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   480 ***********************************************************************
0000218E  4170 0003                     00000003   481          LA    R7,3                    3 tests required for ASC=AR
                              00002192  00000001   482 FRMAS220 EQU   *
00002192  5810 F53C                     0000253C   483          L     R1,FALET                Get from ALET
00002196  B24E 0061                                484          SAR   AR6,R1                  Set in from AR
0000219A  4110 1001                     00000001   485          LA    R1,1(,R1)               Bump ALET
0000219E  5910 F368                     00002368   486          C     R1,=F'3'                Exceeded max of 2?
000021A2  4740 F1A8                     000021A8   487          BL    FRMAS230                No
000021A6  1B11                                     488          SR    R1,R1                   Restart back at ALET 0
                                                   489 *
                              000021A8  00000001   490 FRMAS230 EQU   *
000021A8  5010 F53C                     0000253C   491          ST    R1,FALET                Save updated ALET
                                                   492 *
                              000021AC  00000001   493 FRMAS290 EQU   *



                                                   495 ***********************************************************************
000021AC  4140 0002                     00000002   496          LA    R4,2                    # To Key tests
                              000021B0  00000001   497 TKEY000  EQU   *                       To Key
000021B0  1810                                     498          LR    R1,R0                   Copy control bits
000021B2  5410 F384                     00002384   499          N     R1,=A(OAC1KEY+OAC1K)    Keep only bits we want
000021B6  5710 F384                     00002384   500          X     R1,=A(OAC1KEY+OAC1K)    Flip these bits
000021BA  5400 F388                     00002388   501          N     R0,=A(X'FFFFFFFF'-OAC1KEY-OAC1K) Force these bits off
000021BE  1601                                     502          OR    R0,R1                   Set ctl based on flip results



                                                   504 ***********************************************************************
000021C0  4130 0002                     00000002   505          LA    R3,2                    # From Key tests
                              000021C4  00000001   506 FKEY000  EQU   *                       From Key
000021C4  1810                                     507          LR    R1,R0                   Copy control bits
000021C6  5410 F38C                     0000238C   508          N     R1,=A(OAC2KEY+OAC2K)    Keep only bits we want
000021CA  5710 F38C                     0000238C   509          X     R1,=A(OAC2KEY+OAC2K)    Flip these bits
000021CE  5400 F390                     00002390   510          N     R0,=A(X'FFFFFFFF'-OAC2KEY-OAC2K) Force these bits off
000021D2  1601                                     511          OR    R0,R1                   Set ctl based on flip results
                                                   512 *
000021D4  1B22                                     513          SR    R2,R2                   Clear for ICM
000021D6  4320 F561                     00002561   514          IC    R2,PSWASC               Get ASC we need to test
000021DA  9101 008B                     0000008B   515          TM    SVCINTC+3,X'01'         Are we in problem state?
000021DE  4780 F1EA                     000021EA   516          BZ    BEGIN000                No.  Do every test
000021E2  BD21 F3B5                     000023B5   517          CLM   R2,1,=X'0C'             Entering HOME ASC mode?
000021E6  4780 F26A                     0000226A   518          BE    NEXTTEST                Y, not permitted in prob state
```

```
  LOC        OBJECT CODE        ADDR1     ADDR2     STMT

                                                   520 ***********************************************************************
                                                   521 *
                                                   522 *     Now do the actual 'MVCOS' and check the results afterwards
                                                   523 *
                                                   524 ***********************************************************************
                                                   525 *
                              000021EA  00000001   526 BEGIN000 EQU   *
000021EA  5812 F550                     00002550   527          L     R1,SACIDX(R2)           Get corresponding SAC bits
000021EE  B219 1000                     00000000   528          SAC   0(R1)                   Put machine in mode we need
                                                   529 *
000021F2  4110 0010                     00000010   530          LA    R1,16                   Length to move
000021F6  C810 5008 6000      00000008  00000000   531          MVCOS 8(R5),0(R6),R1
                              000021FC  00000001   532 MVCOS    EQU   *                       Addr after the instruction
                                                   533 *
000021FC  B219 0200                     00000200   534          SAC   ARMODE                  Resume AR mode
00002200  1B11                                     535          SR    R1,R1                   Clear for IC
00002202  4310 F561                     00002561   536          IC    R1,PSWASC               Get PSW ASC mode bits
00002206  A700 0001                                537          TMLH  R0,ASCA                 Is AS setting in OAC1 valid?
0000220A  4780 F218                     00002218   538          BZ    CHK010                  No, use the PSW ASC in R1
0000220E  1810                                     539          LR    R1,R0                   Copy current setting
00002210  5410 F394                     00002394   540          N     R1,=X'00C00000'         Keep OAC1 AS bits
00002214  8810 0014                     00000014   541          SRL   R1,20                   Make AS value a 4-byte index
                                                   542 *
                              00002218  00000001   543 CHK010   EQU   *
00002218  4401 F298                     00002298   544          EX    R0,SETAR(R1)            Set AR1 to access the right AS
0000221C  5810 F544                     00002544   545          L     R1,VADDRTO              -> literal target area
                                                   546 *
00002220  1B22                                     547          SR    R2,R2                   Clear for IC
00002222  4320 F561                     00002561   548          IC    R2,PSWASC               Get PSW ASC mode bits
00002226  A701 0001                                549          TMLL  R0,ASCA                 Is AS setting in OAC2 valid?
0000222A  4780 F238                     00002238   550          BZ    CHK020                  No, use the PSW ASC in R2
                                                   551 *
0000222E  1820                                     552          LR    R2,R0                   Copy current setting
00002230  5420 F398                     00002398   553          N     R2,=X'000000C0'         Keep OAC2 AS bits
00002234  8820 0004                     00000004   554          SRL   R2,4                    Make AS value a 4-byte index
                                                   555 *
                              00002238  00000001   556 CHK020   EQU   *
00002238  4402 F2A8                     000022A8   557          EX    R0,GETALET(R2)          Get the MVCOS operand 2 ALET
0000223C  8920 0004                     00000004   558          SLL   R2,4                    Multiply by 16 to make index
00002240  4122 F418                     00002418   559          LA    R2,FROMPRI(R2)          -> space identifier literal
                                                   560 *
00002244  D50F 1008 2000      00000008  00000000   561          CLC   8(16,R1),0(R2)          Check if MVCOS worked
0000224A  4780 F252                     00002252   562          BE    CHK100                  TEST SUCCESS
0000224E  B2B2 F448                     00002448   563          LPSWE TESTFAIL                Stop machine if test failed
                                                   564 *
                              00002252  00000001   565 CHK100   EQU   *
00002252  FA30 0200 F3B6      00000200  000023B6   566          AP    MVCOSOK,=P'1'           Increment # successful tests
00002258  B24F 0021                                567          EAR   R2,AR1                  Get the ALET we loaded into AR1
0000225C  8920 0005                     00000005   568          SLL   R2,5                    Multiply by 32 to make index
00002260  4122 F3B8                     000023B8   569          LA    R2,PRIPG1(R2)           -> space identifier literal
00002264  D21F 1000 2000      00000000  00000000   570          MVC   0(32,R1),0(R2)          Restore original id literal
```

```
  LOC        OBJECT CODE        ADDR1     ADDR2    STMT

                                                   572 ************************************************************************
                                                   573 *                    Loop through all tests
                                                   574 ************************************************************************
                              0000226A  00000001   575 NEXTTEST EQU   *
0000226A  4630 F1C4                     000021C4   576          BCT   R3,FKEY000              Vary OAC2 K bit
0000226E  4640 F1B0                     000021B0   577          BCT   R4,TKEY000              Vary OAC1 K bit
00002272  4670 F192                     00002192   578          BCT   R7,FRMAS220             Vary from ALETs when OAC2 ASC=AR
00002276  4680 F154                     00002154   579          BCT   R8,FRMAS000             Cycle through OAC2 ASC modes
0000227A  4690 F12C                     0000212C   580          BCT   R9,FRMA000              Vary OAC2 A bit
0000227E  46A0 F10E                     0000210E   581          BCT   R10,TOAS220             Vary To ALETs when OAC1 ASC=AR
00002282  46B0 F0D0                     000020D0   582          BCT   R11,TOAS000             Cycle through OAC1 ASC modes
00002286  46C0 F0A8                     000020A8   583          BCT   R12,TOA000              Vary OAC1 A bit
0000228A  46D0 F094                     00002094   584          BCT   R13,ASC010              Switch to next PSW ASC mode
0000228E  46E0 F07E                     0000207E   585          BCT   R14,STATE000            Switch to next PSW state

                                                   587 ************************************************************************
                                                   588 *                         TEST SUCCESS?
                                                   589 ************************************************************************
                                                   590 *
00002292  0A00                                     591          SVC   0                       Back to supervisor state
00002294  B2B2 F458                     00002458   592          LPSWE GOODPSW                 Stop on success

                                                   594 ************************************************************************
                                                   595 *        SETAR and GETALET are blocks of EXecuted instructions
                                                   596 ************************************************************************
                                                   597 *
00002298  9A11 F39C                     0000239C   598 SETAR    LAM   AR1,AR1,=F'0'   AS=00   Primary, set ALET=0
0000229C  B24D 0015                                599          CPYA  AR1,AR5            01   AR, set AR1 to MVCOS Operand 1
000022A0  9A11 F3A0                     000023A0   600          LAM   AR1,AR1,=F'1'      10   Secondary, set ALET=1
000022A4  9A11 F3A4                     000023A4   601          LAM   AR1,AR1,=F'2'      11   Home set ALET=2

000022A8  4120 0000                     00000000   603 GETALET  LA    R2,0            AS=00   Primary, set ALET=0
000022AC  B24F 0026                                604          EAR   R2,AR6             01   AR, set R2 to MVCOS Operand 2 AR
000022B0  4120 0001                     00000001   605          LA    R2,1               10   Secondary, set ALET=1
000022B4  4120 0002                     00000002   606          LA    R2,2               11   Home set ALET=2

                                                   608 ************************************************************************
                                                   609 *                     HERE FOR SVCs
                                                   610 *
                                                   611 *         SVC  0  -  Set supervisor state in PSW
                                                   612 *         SVC  1  -  Set problem state in PSW
                                                   613 *
                                                   614 ************************************************************************
000022B8                                           615          DC    0D'0'
                              000022B8  00000001   616 SVCFLIH  EQU   *                       SVC Interruption Routine
000022B8  D200 0141 008B      00000141  0000008B   617          MVC   SVCOPSW+1(1),SVCINTC+3  Set state based on SVC num
000022BE  B2B2 0140                     00000140   618          LPSWE SVCOPSW                 Resume execution
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   620 ***********************************************************************
                                                   621 *                 HERE FOR PROGRAM CHECKS
                                                   622 ***********************************************************************
                                                   623 *
000022C8                                           624          DC    0D'0'
                              000022C8  00000001   625 PGMFLIH  EQU   *                       Program check interruptions
000022C8  9513 008F                     0000008F   626          CLI   PGMINTC+3,X'13'         Was this a special op exception?
000022CC  4780 F2FA                     000022FA   627          BE    PGM13                   Yes, use microscope
000022D0  9504 008F                     0000008F   628          CLI   PGMINTC+3,X'04'         Was this a protection exception?
000022D4  4770 F33E                     0000233E   629          BNE   PGMSTOP                 No, stop immediate
                                                   630 *
                              000022D8  00000001   631 PGM04    EQU   *                       Examine PIC 4
000022D8  5000 F538                     00002538   632          ST    R0,WORK                 Save current control bits
000022DC  D403 F538 F3A8      00002538  000023A8   633          NC    WORK,=A(OAC1K+OAC2K)    Either key validity bit = 1?
000022E2  4780 F33E                     0000233E   634          BZ    PGMSTOP                 N, PIC 04 from something else
000022E6  D503 015C F3AC      0000015C  000023AC   635          CLC   PGMOPSW+12(4),=A(MVCOS) Was it the MVCOS that failed?
000022EC  4770 F33E                     0000233E   636          BNE   PGMSTOP                 Nope, halt the machine
000022F0  FA30 0204 F3B6      00000204  000023B6   637          AP    PIC04,=P'1'             Increment # successful tests
000022F6  47F0 F330                     00002330   638          B     PGMEXIT                 Exit FLIH
                                                   639 *
                              000022FA  00000001   640 PGM13    EQU   *
000022FA  9101 008B                     0000008B   641          TM    SVCINTC+3,X'01'         Were we in problem state?
000022FE  4780 F33E                     0000233E   642          BZ    PGMSTOP                 No,error! PIC 13 shouldnt happen
00002302  D503 015C F3AC      0000015C  000023AC   643          CLC   PGMOPSW+12(4),=A(MVCOS) Was it the MVCOS that failed?
00002308  4770 F33E                     0000233E   644          BNE   PGMSTOP                 Nope, halt the machine
0000230C  5000 F538                     00002538   645          ST    R0,WORK                 Save current control bits
00002310  D403 F538 F3B0      00002538  000023B0   646          NC    WORK,=A(OAC1A+OAC2A)    Either Access validity bit = 1?
00002316  4780 F33E                     0000233E   647          BZ    PGMSTOP                 N, PIC 13 from something else
0000231A  A700 00C0                                648          TMLH  R0,ASCHOM               Operand 1 ASC is HOME?
0000231E  4710 F32A                     0000232A   649          BO    PGM13CT                 Yes, PIC13 is ok in prob state
00002322  A701 00C0                                650          TMLL  R0,ASCHOM               Operand 2 ASC is HOME?
00002326  4780 F33E                     0000233E   651          BZ    PGMSTOP                 No. Something wrong
                                                   652 *
                              0000232A  00000001   653 PGM13CT  EQU   *
0000232A  FA30 0208 F3B6      00000208  000023B6   654          AP    PIC13,=P'1'             Increment # successful tests
                                                   655 *
                              00002330  00000001   656 PGMEXIT  EQU   *                       Exit from FLIH everything OK
00002330  9101 008B                     0000008B   657          TM    SVCINTC+3,X'01'         Were we in problem state?
00002334  4780 F26A                     0000226A   658          BZ    NEXTTEST                No, proceed to next test
00002338  0A01                                     659          SVC   1                       Return to problem state
0000233A  47F0 F26A                     0000226A   660          B     NEXTTEST                And return to next test



                                                   662 ***********************************************************************
                                                   663 *                   UNEXPECTED PROGRAM CHECK
                                                   664 ***********************************************************************
                              0000233E  00000001   665 PGMSTOP  EQU   *                       Halt if something wrong
0000233E  B2B2 F468                     00002468   666          LPSWE HALT                    Here for unexpected prog checks
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   668 ***********************************************************************
                                                   669 *                        WORKING STORAGE
                                                   670 ***********************************************************************
                                                   671 *
00002344                                           672          LTORG
00002344  10031003                                 673                =X'10031003'
00002348  0000000C                                 674                =X'0000000C'
0000234C  00010000                                 675                =A(OAC1A)
00002350  FFFEFFFF                                 676                =A(X'FFFFFFFF'-OAC1A)
00002354  FF3FFFFF                                 677                =A(X'FFFFFFFF'-(ASCHOM*65536))
00002358  00C00000                                 678                =A(ASCHOM*65536)
0000235C  00400000                                 679                =X'00400000'
00002360  00C10000                                 680                =A(OAC1A+ASCHOM*65536)
00002364  00410000                                 681                =A(OAC1A+ASCAR*65536)
00002368  00000003                                 682                =F'3'
0000236C  00000001                                 683                =A(OAC2A)
00002370  FFFFFFFE                                 684                =A(X'FFFFFFFF'-OAC2A)
00002374  FFFFFF3F                                 685                =A(X'FFFFFFFF'-ASCHOM)
00002378  000000C0                                 686                =A(ASCHOM)
0000237C  000000C1                                 687                =A(OAC2A+ASCHOM)
00002380  00000041                                 688                =A(OAC2A+ASCAR)
00002384  10020000                                 689                =A(OAC1KEY+OAC1K)
00002388  EFFDFFFF                                 690                =A(X'FFFFFFFF'-OAC1KEY-OAC1K)
0000238C  00001002                                 691                =A(OAC2KEY+OAC2K)
00002390  FFFFEFFD                                 692                =A(X'FFFFFFFF'-OAC2KEY-OAC2K)
00002394  00C00000                                 693                =X'00C00000'
00002398  000000C0                                 694                =X'000000C0'
0000239C  00000000                                 695                =F'0'
000023A0  00000001                                 696                =F'1'
000023A4  00000002                                 697                =F'2'
000023A8  00020002                                 698                =A(OAC1K+OAC2K)
000023AC  000021FC                                 699                =A(MVCOS)
000023B0  00010001                                 700                =A(OAC1A+OAC2A)
000023B4  04                                       701                =X'04'
000023B5  0C                                       702                =X'0C'
000023B6  1C                                       703                =P'1'


000023B8                                           705          DC    0D'0'
000023B8  D7D9C960 D7C7F140                        706 PRIPG1   DC    CL16'PRI-PG1'            Eyecatcher
000023C8  D7D9C960 D7C7F240                        707 PRIPG2   DC    CL16'PRI-PG2'            Eyecatcher
000023D8  E2C5C360 D7C7F140                        708 SECPG1   DC    CL16'SEC-PG1'            Eyecatcher
000023E8  E2C5C360 D7C7F240                        709 SECPG2   DC    CL16'SEC-PG2'            Eyecatcher
000023F8  C8D6D460 D7C7F140                        710 HOMPG1   DC    CL16'HOM-PG1'            Eyecatcher
00002408  C8D6D460 D7C7F240                        711 HOMPG2   DC    CL16'HOM-PG2'            Eyecatcher
                                                   712 *
00002418  C6D9D6D4 D7D9C9F1                        713 FROMPRI  DC    CL16'FROMPRI1FROMPRI2'   Eyecatcher
00002428  C6D9D6D4 E2C5C3F1                        714 FROMSEC  DC    CL16'FROMSEC1FROMSEC2'   Eyecatcher
00002438  C6D9D6D4 C8D6D4F1                        715 FROMHOM  DC    CL16'FROMHOM1FROMHOM2'   Eyecatcher
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00002448  04024000 80000000                        717 TESTFAIL DC    X'0402400080000000',XL4'00',X'00000BAD'  TEST FAILED
00002458  04024000 80000000                        718 GOODPSW  DC    X'0402400080000000',XL4'00',X'00000000'  Test Success
00002468  04024000 80000000                        719 HALT     DC    X'04024000',X'80000000',XL4'00',X'0000000F' Abnormal end


                                                   721 *        Control registers
                                                   722 *
00002478                                           723 CREGS    DC    0D'0'
00002478  00000000 04000000                        724 CREG0    DC    X'00000000',X'04000000' Secondary space control bit 37
00002480  00000000 00004000                        725 CREG1    DC    X'00000000',A(SEGPRI)   Primary ASCE
00002488  00000000 00009100                        726 CREG2    DC    X'00000000',A(DUCT)     Dispatchable Unit Ctl Table
00002490  00000000 C0000001                        727 CREG3    DC    X'00000000',X'C0000001' PKM=C000, Secondary ASN=1
00002498  00000000 00000000                        728 CREG4    DC    X'00000000',X'00000000' Primary ASN=0
000024A0  00000000 00009000                        729 CREG5    DC    X'00000000',A(PASTEO)   Primary ASTE Origin
000024A8  00000000 00000000                        730 CREG6    DC    X'00000000',X'00000000'
000024B0  00000000 00005000                        731 CREG7    DC    X'00000000',A(SEGSEC)   Secondary ASCE
000024B8  00000000 00000000                        732 CREG8    DC    X'00000000',X'00000000'
000024C0  00000000 00000000                        733 CREG9    DC    X'00000000',X'00000000'
000024C8  00000000 00000000                        734 CREG10   DC    X'00000000',X'00000000'
000024D0  00000000 00000000                        735 CREG11   DC    X'00000000',X'00000000'
000024D8  00000000 00000000                        736 CREG12   DC    X'00000000',X'00000000'
000024E0  00000000 00006000                        737 CREG13   DC    X'00000000',A(SEGHOM)   Home ASCE
000024E8  00000000 00000000                        738 CREG14   DC    X'00000000',X'00000000'
000024F0  00000000 00000000                        739 CREG15   DC    X'00000000',X'00000000'


000024F8  00000000 00000000                        741 AREGS    DC    16F'0'                  Init for Access Registers
00002538  00000000                                 742 WORK     DC    F'0'                    Work area
0000253C  00000000                                 743 FALET    DC    F'0'                    FROM ALET
00002540  00000000                                 744 TALET    DC    F'0'                    TO ALET
                                                   745 *
00002544  00010FF0                                 746 VADDRTO  DC    X'00010FF0'             Virtual addr within all 3 addr
                                                   747 *                                       space where the identifying
                                                   748 *                                       space literal is placed
                                                   749 *
00002548  00012FF8                                 750 VADDRFRM DC    X'00012FF8'             Virtual addr within all 3 addr
                                                   751 *                                       space where the 'from'
                                                   752 *                                       identifying literal is placed


0000254C                                           754 CONTROL  DC    0F'0'                   R0 MVCOS Control bits
0000254C  0000                                     755 OAC1     DC    H'0'                     1st OAC
                              10000000  00000001   756 OAC1KEY  EQU   X'10000000'               1st key
                              00020000  00000001   757 OAC1K    EQU   X'00020000'               1st key validity bit
                              00010000  00000001   758 OAC1A    EQU   X'00010000'               1st ASC validity bit
0000254E  0000                                     759 OAC2     DC    H'0'                     2nd OAC
                              00001000  00000001   760 OAC2KEY  EQU   X'00001000'               2nd key
                              00000002  00000001   761 OAC2K    EQU   X'00000002'               2nd key validity bit
                              00000001  00000001   762 OAC2A    EQU   X'00000001'               2nd ASC validity bit
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   764 *        Bits in OAC1,OAC2 control bytes
                                                   765 *
                              00000100  00000001   766 KEY      EQU   X'0100' .... ...1 .... ....  Key 1 set
                              00000000  00000001   767 ASCPRI   EQU   X'0000' .... .... 00.. ....  Primary space
                              00000040  00000001   768 ASCAR    EQU   X'0040' .... .... 01.. ....  AR
                              00000080  00000001   769 ASCSEC   EQU   X'0080' .... .... 10.. ....  Secondary space
                              000000C0  00000001   770 ASCHOM   EQU   X'00C0' .... .... 11.. ....  Home space
                              00000002  00000001   771 ASCK     EQU   X'0002' .... .... .... ..1.  Specified key valid
                              00000001  00000001   772 ASCA     EQU   X'0001' .... .... .... ...1  Specified AS valid


00002550  00000000                                 774 SACIDX   DC    X'00000000'        Pri   SAC bits to set PSW 16-17=00
00002554  00000200                                 775          DC    X'00000200'        AR                              01
00002558  00000100                                 776          DC    X'00000100'        Sec                             10
0000255C  00000300                                 777          DC    X'00000300'        Home                            11
                                                   778 *
00002560  00                                       779 PSWSTATE DC    X'00'                    Tracks suprv/prob state in PSW
00002561  00                                       780 PSWASC   DC    X'00'                    Tracks ASC Mode setting of PSW
                                                   781 *                                        in bits 4-5 of this byte
                                                   782 *                                        during the MVCOS execution


00002562                      00002562  00004000   784          ORG   STRTLABL+X'4000'
00004000  00000000 00007000                        785 SEGPRI   DC    X'00000000',A(PAGPRI)
                                                   786 *
00004008                      00004008  00005000   787          ORG   STRTLABL+X'5000'
00005000  00000000 00007800                        788 SEGSEC   DC    X'00000000',A(PAGSEC)
                                                   789 *
00005008                      00005008  00006000   790          ORG   STRTLABL+X'6000'
00006000  00000000 00008000                        791 SEGHOM   DC    X'00000000',A(PAGHOM)
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00006008                      00006008  00007000   793          ORG   STRTLABL+X'7000'
                              00007000  00000001   794 PAGPRI   EQU   *                        Primary Space Page Tables
                                                   795 *
                                                   796 * Addresses 0-FFFF common to all addresses spaces, VIRT=REAL
00007000  00000000 00000000                        797          DC    X'00000000',X'00000000'  R=00000  V=00000
00007008  00000000 00001000                        798          DC    X'00000000',X'00001000'  R=01000  V=01000
00007010  00000000 00002000                        799          DC    X'00000000',X'00002000'  R=02000  V=02000
00007018  00000000 00003000                        800          DC    X'00000000',X'00003000'  R=03000  V=03000
00007020  00000000 00004000                        801          DC    X'00000000',X'00004000'  R=04000  V=04000
00007028  00000000 00005000                        802          DC    X'00000000',X'00005000'  R=05000  V=05000
00007030  00000000 00006000                        803          DC    X'00000000',X'00006000'  R=06000  V=06000
00007038  00000000 00007000                        804          DC    X'00000000',X'00007000'  R=07000  V=07000
00007040  00000000 00008000                        805          DC    X'00000000',X'00008000'  R=08000  V=08000
00007048  00000000 00009000                        806          DC    X'00000000',X'00009000'  R=09000  V=09000
00007050  00000000 0000A000                        807          DC    X'00000000',X'0000A000'  R=0A000  V=0A000
00007058  00000000 0000B000                        808          DC    X'00000000',X'0000B000'  R=0B000  V=0B000
00007060  00000000 0000C000                        809          DC    X'00000000',X'0000C000'  R=0C000  V=0C000
00007068  00000000 0000D000                        810          DC    X'00000000',X'0000D000'  R=0D000  V=0D000
00007070  00000000 0000E000                        811          DC    X'00000000',X'0000E000'  R=0E000  V=0E000
00007078  00000000 0000F000                        812          DC    X'00000000',X'0000F000'  R=0F000  V=0F000


                                                   814 * Begin primary space only storage V-addrs 10000-1FFFF
00007080  00000000 00020000                        815          DC    X'00000000',X'00020000'  R=20000  V=10000
00007088  00000000 00021000                        816          DC    X'00000000',X'00021000'  R=21000  V=11000
00007090  00000000 00022000                        817          DC    X'00000000',X'00022000'  R=22000  V=12000
00007098  00000000 00023000                        818          DC    X'00000000',X'00023000'  R=23000  V=13000
000070A0  00000000 00024000                        819          DC    X'00000000',X'00024000'  R=24000  V=14000
000070A8  00000000 00025000                        820          DC    X'00000000',X'00025000'  R=25000  V=15000
000070B0  00000000 00026000                        821          DC    X'00000000',X'00026000'  R=26000  V=16000
000070B8  00000000 00027000                        822          DC    X'00000000',X'00027000'  R=27000  V=17000
000070C0  00000000 00028000                        823          DC    X'00000000',X'00028000'  R=28000  V=18000
000070C8  00000000 00029000                        824          DC    X'00000000',X'00029000'  R=29000  V=19000
000070D0  00000000 0002A000                        825          DC    X'00000000',X'0002A000'  R=2A000  V=1A000
000070D8  00000000 0002B000                        826          DC    X'00000000',X'0002B000'  R=2B000  V=1B000
000070E0  00000000 0002C000                        827          DC    X'00000000',X'0002C000'  R=2C000  V=1C000
000070E8  00000000 0002D000                        828          DC    X'00000000',X'0002D000'  R=2D000  V=1D000
000070F0  00000000 0002E000                        829          DC    X'00000000',X'0002E000'  R=2E000  V=1E000
000070F8  00000000 0002F000                        830          DC    X'00000000',X'0002F000'  R=2F000  V=1F000
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00007100                     00007100  00007800   832          ORG   STRTLABL+X'7800'
                             00007800  00000001   833 PAGSEC   EQU   *                         Secondary Space Page Tables
                                                  834 *
                                                  835 * Addresses 0-FFFF common to all addresses spaces, VIRT=REAL
00007800  00000000 00000000                       836          DC    X'00000000',X'00000000'  R=00000  V=00000
00007808  00000000 00001000                       837          DC    X'00000000',X'00001000'  R=01000  V=01000
00007810  00000000 00002000                       838          DC    X'00000000',X'00002000'  R=02000  V=02000
00007818  00000000 00003000                       839          DC    X'00000000',X'00003000'  R=03000  V=03000
00007820  00000000 00004000                       840          DC    X'00000000',X'00004000'  R=04000  V=04000
00007828  00000000 00005000                       841          DC    X'00000000',X'00005000'  R=05000  V=05000
00007830  00000000 00006000                       842          DC    X'00000000',X'00006000'  R=06000  V=06000
00007838  00000000 00007000                       843          DC    X'00000000',X'00007000'  R=07000  V=07000
00007840  00000000 00008000                       844          DC    X'00000000',X'00008000'  R=08000  V=08000
00007848  00000000 00009000                       845          DC    X'00000000',X'00009000'  R=09000  V=09000
00007850  00000000 0000A000                       846          DC    X'00000000',X'0000A000'  R=0A000  V=0A000
00007858  00000000 0000B000                       847          DC    X'00000000',X'0000B000'  R=0B000  V=0B000
00007860  00000000 0000C000                       848          DC    X'00000000',X'0000C000'  R=0C000  V=0C000
00007868  00000000 0000D000                       849          DC    X'00000000',X'0000D000'  R=0D000  V=0D000
00007870  00000000 0000E000                       850          DC    X'00000000',X'0000E000'  R=0E000  V=0E000
00007878  00000000 0000F000                       851          DC    X'00000000',X'0000F000'  R=0F000  V=0F000


                                                  853 * Begin secondary space only storage V-addrs 10000-1FFFF
00007880  00000000 00030000                       854          DC    X'00000000',X'00030000'  R=30000  V=10000
00007888  00000000 00031000                       855          DC    X'00000000',X'00031000'  R=31000  V=11000
00007890  00000000 00032000                       856          DC    X'00000000',X'00032000'  R=32000  V=12000
00007898  00000000 00033000                       857          DC    X'00000000',X'00033000'  R=33000  V=13000
000078A0  00000000 00034000                       858          DC    X'00000000',X'00034000'  R=34000  V=14000
000078A8  00000000 00035000                       859          DC    X'00000000',X'00035000'  R=35000  V=15000
000078B0  00000000 00036000                       860          DC    X'00000000',X'00036000'  R=36000  V=16000
000078B8  00000000 00037000                       861          DC    X'00000000',X'00037000'  R=37000  V=17000
000078C0  00000000 00038000                       862          DC    X'00000000',X'00038000'  R=38000  V=18000
000078C8  00000000 00039000                       863          DC    X'00000000',X'00039000'  R=39000  V=19000
000078D0  00000000 0003A000                       864          DC    X'00000000',X'0003A000'  R=3A000  V=1A000
000078D8  00000000 0003B000                       865          DC    X'00000000',X'0003B000'  R=3B000  V=1B000
000078E0  00000000 0003C000                       866          DC    X'00000000',X'0003C000'  R=3C000  V=1C000
000078E8  00000000 0003D000                       867          DC    X'00000000',X'0003D000'  R=3D000  V=1D000
000078F0  00000000 0003E000                       868          DC    X'00000000',X'0003E000'  R=3E000  V=1E000
000078F8  00000000 0003F000                       869          DC    X'00000000',X'0003F000'  R=3F000  V=1F000
```

```
  LOC          OBJECT CODE        ADDR1     ADDR2    STMT

00007900                      00007900  00008000   871          ORG   STRTLABL+X'8000'
                              00008000  00000001   872 PAGHOM   EQU   *                        Home Space Page Tables
                                                   873 *
                                                   874 * Addresses 0-FFFF common to all addresses spaces, VIRT=REAL
00008000  00000000 00000000                        875          DC    X'00000000',X'00000000'  R=00000  V=00000
00008008  00000000 00001000                        876          DC    X'00000000',X'00001000'  R=01000  V=01000
00008010  00000000 00002000                        877          DC    X'00000000',X'00002000'  R=02000  V=02000
00008018  00000000 00003000                        878          DC    X'00000000',X'00003000'  R=03000  V=03000
00008020  00000000 00004000                        879          DC    X'00000000',X'00004000'  R=04000  V=04000
00008028  00000000 00005000                        880          DC    X'00000000',X'00005000'  R=05000  V=05000
00008030  00000000 00006000                        881          DC    X'00000000',X'00006000'  R=06000  V=06000
00008038  00000000 00007000                        882          DC    X'00000000',X'00007000'  R=07000  V=07000
00008040  00000000 00008000                        883          DC    X'00000000',X'00008000'  R=08000  V=08000
00008048  00000000 00009000                        884          DC    X'00000000',X'00009000'  R=09000  V=09000
00008050  00000000 0000A000                        885          DC    X'00000000',X'0000A000'  R=0A000  V=0A000
00008058  00000000 0000B000                        886          DC    X'00000000',X'0000B000'  R=0B000  V=0B000
00008060  00000000 0000C000                        887          DC    X'00000000',X'0000C000'  R=0C000  V=0C000
00008068  00000000 0000D000                        888          DC    X'00000000',X'0000D000'  R=0D000  V=0D000
00008070  00000000 0000E000                        889          DC    X'00000000',X'0000E000'  R=0E000  V=0E000
00008078  00000000 0000F000                        890          DC    X'00000000',X'0000F000'  R=0F000  V=0F000


                                                   892 * Begin home space only storage V-addrs 10000-1FFFF (V=R)
00008080  00000000 00010000                        893          DC    X'00000000',X'00010000'  R=10000  V=10000
00008088  00000000 00011000                        894          DC    X'00000000',X'00011000'  R=11000  V=11000
00008090  00000000 00012000                        895          DC    X'00000000',X'00012000'  R=12000  V=12000
00008098  00000000 00013000                        896          DC    X'00000000',X'00013000'  R=13000  V=13000
000080A0  00000000 00014000                        897          DC    X'00000000',X'00014000'  R=14000  V=14000
000080A8  00000000 00015000                        898          DC    X'00000000',X'00015000'  R=15000  V=15000
000080B0  00000000 00016000                        899          DC    X'00000000',X'00016000'  R=16000  V=16000
000080B8  00000000 00017000                        900          DC    X'00000000',X'00017000'  R=17000  V=17000
000080C0  00000000 00018000                        901          DC    X'00000000',X'00018000'  R=18000  V=18000
000080C8  00000000 00019000                        902          DC    X'00000000',X'00019000'  R=19000  V=19000
000080D0  00000000 0001A000                        903          DC    X'00000000',X'0001A000'  R=1A000  V=1A000
000080D8  00000000 0001B000                        904          DC    X'00000000',X'0001B000'  R=1B000  V=1B000
000080E0  00000000 0001C000                        905          DC    X'00000000',X'0001C000'  R=1C000  V=1C000
000080E8  00000000 0001D000                        906          DC    X'00000000',X'0001D000'  R=1D000  V=1D000
000080F0  00000000 0001E000                        907          DC    X'00000000',X'0001E000'  R=1E000  V=1E000
000080F8  00000000 0001F000                        908          DC    X'00000000',X'0001F000'  R=1F000  V=1F000
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00008100                      00008100  00009000   910          ORG   STRTLABL+X'9000'
00009000                                           911 PASTEO   DS    0XL64                    Primary ASN Second Table Entry
00009000  00000000                                 912          DC    A(0)           +0        ATO
00009004  00000000                                 913          DC    A(0)            4        AX,ATL
00009008  00000000 00004000                        914          DC    A(0),A(SEGPRI)  8        Primary ASCE (same as CREG1)
00009010  00000000                                 915          DC    A(0)           16        ALD
00009014  00000000                                 916          DC    A(0)           20        ASTESN
00009018  00000000                                 917          DC    A(0)           24        LTD
0000901C  00000000                                 918          DC    A(0)           28        Ctl prog use
00009020  00000000                                 919          DC    A(0)           32        Ctl prog use
00009024  00000000                                 920          DC    A(0)           36        Ctl prog use
00009028  00000000                                 921          DC    A(0)           40        unassigned
0000902C  00000000                                 922          DC    A(0)           44        ASTEIN
00009030  00000000                                 923          DC    A(0)           48        unassigned
00009034  00000000                                 924          DC    A(0)           58        unassigned
00009038  00000000                                 925          DC    A(0)           56        unassigned
0000903C  00000000                                 926          DC    A(0)           60        unassigned



                                                   928 *
                                                   929 *       Dispatchable Unit Control Table (DUCT)
                                                   930 *
                                                   931 *         This DUCT is used by the primary space programming
                                                   932 *         when in Access Register mode in order to use the DU-AL.
                                                   933 *
00009040                      00009040  00009100   934          ORG   STRTLABL+X'9100'
00009100                                           935 DUCT     DS    0XL64                    Dispatchable Unit Control Tbl
00009100  00000000                                 936          DC    A(0)     +0              BASTEO
00009104  00000000                                 937          DC    A(0)      4              SSASTEO
00009108  00000000                                 938          DC    A(0)      8              unassigned
0000910C  00000000                                 939          DC    A(0)     12              SSASTESN
00009110  00009200                                 940 DUALD    DC    A(DUAL)  16              DU-AL origin
00009114  00000000                                 941          DC    A(0)     20              PSW key masks
00009118  00000000                                 942          DC    A(0)     24              unassigned
0000911C  00000000                                 943          DC    A(0)     28              unassigned
00009120  00000000                                 944          DC    A(0)     32              Return addr high
00009124  00000000                                 945          DC    A(0)     36              Return addr low
00009128  00000000                                 946          DC    A(0)     40              unassigned
0000912C  00000000                                 947          DC    A(0)     44              TRCB
00009130  00000000                                 948          DC    A(0)     48              unassigned
00009134  00000000                                 949          DC    A(0)     52              unassigned
00009138  00000000                                 950          DC    A(0)     56              unassigned
0000913C  00000000                                 951          DC    A(0)     60              unassigned
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   953 *
                                                   954 *        Dispatchable Unit - Access List (DU-AL)
                                                   955 *
                                                   956 *         8 access list entries, only entry 2 is valid (AR ALET = 2)
                                                   957 *
00009140                      00009140  00009200   958          ORG   STRTLABL+X'9200'
                              00009200  00000001   959 DUAL     EQU   *                        DU Access List
00009200  80000000 00000000                        960 ALE0     DC    X'80',15X'00'            ALE 0 invalid
00009210  80000000 00000000                        961 ALE1     DC    X'80',15X'00'            ALE 1 invalid
                                                   962 *
00009220                                           963 ALE2     DS    0XL16                    ALE 2 -> HOME space
00009220  00000000                                 964          DC    A(0)                     I,FO,P,ALESN,ALEAX all 0
00009224  00000000                                 965          DC    A(0)                     unassigned
00009228  0000A000                                 966          DC    A(HASTEO)                Home space ASTE Origin
0000922C  00000000                                 967          DC    A(0)                     ASTESN seq # set to 0
                                                   968 *
00009230  80000000 00000000                        969 ALE3     DC    X'80',15X'00'            ALE 3 invalid
00009240  80000000 00000000                        970 ALE4     DC    X'80',15X'00'            ALE 4 invalid
00009250  80000000 00000000                        971 ALE5     DC    X'80',15X'00'            ALE 5 invalid
00009260  80000000 00000000                        972 ALE6     DC    X'80',15X'00'            ALE 6 invalid
00009270  80000000 00000000                        973 ALE7     DC    X'80',15X'00'            ALE 7 invalid



                                                   975 *        The HASTE is needed for ALET 2's ALE entry above
                                                   976 *
00009280                      00009280  0000A000   977          ORG   STRTLABL+X'A000'
0000A000                                           978 HASTEO   DS    0XL64                    Home ASN Second Table Entry
0000A000  00000000                                 979          DC    A(0)           +0        ATO
0000A004  00000000                                 980          DC    A(0)            4        AX,ATL
0000A008  00000000 00006000                        981          DC    A(0),A(SEGHOM)  8        Home ASCE (same as CREG13)
0000A010  00000000                                 982          DC    A(0)           16        ALD
0000A014  00000000                                 983          DC    A(0)           20        ASTESN
0000A018  00000000                                 984          DC    A(0)           24        LTD
0000A01C  00000000                                 985          DC    A(0)           28        Ctl prog use
0000A020  00000000                                 986          DC    A(0)           32        Ctl prog use
0000A024  00000000                                 987          DC    A(0)           36        Ctl prog use
0000A028  00000000                                 988          DC    A(0)           40        unassigned
0000A02C  00000000                                 989          DC    A(0)           44        ASTEIN
0000A030  00000000                                 990          DC    A(0)           48        unassigned
0000A034  00000000                                 991          DC    A(0)           58        unassigned
0000A038  00000000                                 992          DC    A(0)           56        unassigned
0000A03C  00000000                                 993          DC    A(0)           60        unassigned



                              00000100  00000001   995 SECMODE  EQU   256      Secondary mode
                              00000300  00000001   996 HOMEMODE EQU   768      Home space mode
                              00000200  00000001   997 ARMODE   EQU   512      Enter AR mode


                                        00002000   999          END START
```

```
     SYMBOL        TYPE   VALUE      LENGTH   DEFN  REFERENCES

ALE0                X    00009200          1   960
ALE1                X    00009210          1   961
ALE2                X    00009220         16   963
ALE3                X    00009230          1   969
ALE4                X    00009240          1   970
ALE5                X    00009250          1   971
ALE6                X    00009260          1   972
ALE7                X    00009270          1   973
AR0                 U    00000000          1    22  333
AR1                 U    00000001          1    23  567  598  599  600  601
AR10                U    0000000A          1    32
AR11                U    0000000B          1    33
AR12                U    0000000C          1    34
AR13                U    0000000D          1    35
AR14                U    0000000E          1    36
AR15                U    0000000F          1    37  333
AR2                 U    00000002          1    24
AR3                 U    00000003          1    25
AR4                 U    00000004          1    26
AR5                 U    00000005          1    27  427  599
AR6                 U    00000006          1    28  484  604
AR7                 U    00000007          1    29
AR8                 U    00000008          1    30
AR9                 U    00000009          1    31
AREGS               F    000024F8          4   741  333
ARMODE              U    00000200          1   997  351  534
ASC010              U    00002094          1   375  584
ASCA                U    00000001          1   772  391  400  448  457  537  549
ASCAR               U    00000040          1   768  412  469
ASCHOM              U    000000C0          1   770  648  650  389  395  411  446  452  468
ASCK                U    00000002          1   771
ASCPRI              U    00000000          1   767
ASCSEC              U    00000080          1   769
BEGIN000            U    000021EA          1   526  516
CHK010              U    00002218          1   543  538
CHK020              U    00002238          1   556  550
CHK100              U    00002252          1   565  562
CONTROL             F    0000254C          4   754
CR0                 U    00000000          1    38  334
CR1                 U    00000001          1    39
CR10                U    0000000A          1    48
CR11                U    0000000B          1    49
CR12                U    0000000C          1    50
CR13                U    0000000D          1    51
CR14                U    0000000E          1    52
CR15                U    0000000F          1    53  334
CR2                 U    00000002          1    40
CR3                 U    00000003          1    41
CR4                 U    00000004          1    42
CR5                 U    00000005          1    43
CR6                 U    00000006          1    44
CR7                 U    00000007          1    45
```

```
     SYMBOL        TYPE   VALUE      LENGTH   DEFN  REFERENCES

CR8                 U    00000008          1    46
CR9                 U    00000009          1    47
CREG0               X    00002478          4   724
CREG1               X    00002480          4   725
CREG10              X    000024C8          4   734
CREG11              X    000024D0          4   735
CREG12              X    000024D8          4   736
CREG13              X    000024E0          4   737
CREG14              X    000024E8          4   738
CREG15              X    000024F0          4   739
CREG2               X    00002488          4   726
CREG3               X    00002490          4   727
CREG4               X    00002498          4   728
CREG5               X    000024A0          4   729
CREG6               X    000024A8          4   730
CREG7               X    000024B0          4   731
CREG8               X    000024B8          4   732
CREG9               X    000024C0          4   733
CREGS               D    00002478          8   723  334
DUAL                U    00009200          1   959  940
DUALD               A    00009110          4   940
DUCT                X    00009100         64   935  726
EXTNPSW             X    000001B0         16   296
FALET               F    0000253C          4   743  483  491
FKEY000             U    000021C4          1   506  576
FRMA000             U    0000212C          1   440  580
FRMAS000            U    00002154          1   456  579
FRMAS200            U    00002182          1   473  449  458
FRMAS210            U    0000218E          1   478  471
FRMAS220            U    00002192          1   482  578
FRMAS230            U    000021A8          1   490  487
FRMAS290            U    000021AC          1   493  470  476
FROMHOM             C    00002438         16   715  349
FROMPRI             C    00002418         16   713  341  559
FROMSEC             C    00002428         16   714  345
GETALET             I    000022A8          4   603  557
GOODPSW             X    00002458          8   718  592
HALT                X    00002468          4   719  666
HASTEO              X    0000A000         64   978  966
HOMEMODE            U    00000300          1   996  347
HOMPG1              C    000023F8         16   710  348
HOMPG2              C    00002408         16   711
IMAGE               1    00000000      41024     0
KEY                 U    00000100          1   766
MVCOS               U    000021FC          1   532  635
MVCOS001            J    00000000      41024   280
MVCOSOK             P    00000200          4   303  566
NEXTTEST            U    0000226A          1   575  518  658  660
OAC1                H    0000254C          2   755
OAC1A               U    00010000          1   758  385  387  411  412  646
OAC1K               U    00020000          1   757  499  501  633
OAC1KEY             U    10000000          1   756  499  501
```

```
     SYMBOL        TYPE   VALUE      LENGTH   DEFN  REFERENCES

OAC2                H    0000254E          2   759
OAC2A               U    00000001          1   762  442  444  468  469  646
OAC2K               U    00000002          1   761  508  510  633
OAC2KEY             U    00001000          1   760  508  510
PAGHOM              U    00008000          1   872  791
PAGPRI              U    00007000          1   794  785
PAGSEC              U    00007800          1   833  788
PASTEO              X    00009000         64   911  729
PGM04               U    000022D8          1   631
PGM13               U    000022FA          1   640  627
PGM13CT             U    0000232A          1   653  649
PGMEXIT             U    00002330          1   656  638
PGMFLIH             U    000022C8          1   625  298
PGMINTC             X    0000008C          4   288  626  628
PGMNPSW             X    000001D0          4   298
PGMOPSW             X    00000150         16   292  635  643
PGMSTOP             U    0000233E          1   665  629  634  636  642  644  647  651
PIC04               P    00000204          4   304  637
PIC13               P    00000208          4   305  654
PRIPG1              C    000023B8         16   706  340  569
PRIPG2              C    000023C8         16   707
PSWASC              X    00002561          1   780  355  376  379  418  475  514  536  548
PSWSTATE            X    00002560          1   779  358  365  366
R0                  U    00000000          1     6  325  329  352  384  387  388  389  391  395  400  403  406  407  410  441  444
                                                    445  446  448  452  457  460  463  464  467  498  501  502  507  510  511  537
                                                    539  544  549  552  557  632  645  648  650
R1                  U    00000001          1     7  327  329  330  366  367  376  377  378  379  384  385  386  388  403  404  405
                                                    407  410  411  412  426  427  428  429  431  434  441  442  443  445  460  461
                                                    462  464  467  468  469  483  484  485  486  488  491  498  499  500  502  507
                                                    508  509  511  527  528  530  531  535  536  539  540  541  544  545  561  570
R10                 U    0000000A          1    16  409  417  424  581
R11                 U    0000000B          1    17  390  398  582
R12                 U    0000000C          1    18  382  583
R13                 U    0000000D          1    19  374  584
R14                 U    0000000E          1    20  363  585
R15                 U    0000000F          1    21  314  315  316  317
R2                  U    00000002          1     8  319  323  326  331  513  514  517  527  547  548  552  553  554  557  558  559
                                                    561  567  568  569  570  603  604  605  606
R3                  U    00000003          1     9  320  505  576
R4                  U    00000004          1    10  322  323  496  577
R5                  U    00000005          1    11  337  340  344  348  531
R6                  U    00000006          1    12  338  341  345  349  531
R7                  U    00000007          1    13  466  474  481  578
R8                  U    00000008          1    14  447  455  579
R9                  U    00000009          1    15  439  580
RESTART             X    000001A0          4   295
SACIDX              X    00002550          4   774  527
SECMODE             U    00000100          1   995  343
SECPG1              C    000023D8         16   708  344
SECPG2              C    000023E8         16   709
SEGHOM              X    00006000          4   791  737  981
SEGPRI              X    00004000          4   785  725  914
```

```
     SYMBOL        TYPE   VALUE      LENGTH   DEFN  REFERENCES

SEGSEC              X    00005000          4   788  731
SET000              I    0000201C          4   329  331
SETAR               I    00002298          4   598  544
SKIP001             U    00002090          1   371  368
START               I    00002000          2   314  295  317  999
STATE000            U    0000207E          1   364  585
STRTLABL            U    00000000          1   281  286  290  294  302  313  784  787  790  793  832  871  910  934  958  977  282
SVCFLIH             U    000022B8          1   616  297
SVCINTC             X    00000088          4   287  515  617  641  657
SVCNPSW             X    000001C0          4   297
SVCOPSW             X    00000140         16   291  617  618
SVCSTATE            I    0000208E          2   370  367
TALET               F    00002540          4   744  426  434
TESTFAIL            X    00002448          8   717  563
TKEY000             U    000021B0          1   497  577
TOA000              U    000020A8          1   383  583
TOAS000             U    000020D0          1   399  582
TOAS200             U    000020FE          1   416  392  401
TOAS210             U    0000210A          1   421  414
TOAS220             U    0000210E          1   425  581
TOAS230             U    00002124          1   433  430
TOAS290             U    00002128          1   436  413  419
VADDRFRM            X    00002548          4   750  338
VADDRTO             X    00002544          4   746  337  545
WORK                F    00002538          4   742  632  633  645  646
=A(ASCHOM)          A    00002378          4   686  452  462
=A(ASCHOM*65536)    A    00002358          4   678  395  405
=A(MVCOS)           A    000023AC          4   699  635  643
=A(OAC1A)           A    0000234C          4   675  385  386
=A(OAC1A+ASCAR*65536)
                    A    00002364          4   681  412
=A(OAC1A+ASCHOM*65536)
                    A    00002360          4   680  411
=A(OAC1A+OAC2A)     A    000023B0          4   700  646
=A(OAC1K+OAC2K)     A    000023A8          4   698  633
=A(OAC1KEY+OAC1K)
                    A    00002384          4   689  499  500
=A(OAC2A)           A    0000236C          4   683  442  443
=A(OAC2A+ASCAR)     A    00002380          4   688  469
=A(OAC2A+ASCHOM)    A    0000237C          4   687  468
=A(OAC2KEY+OAC2K)
                    A    0000238C          4   691  508  509
=A(X'FFFFFFFF'-(ASCHOM*65536))
                    A    00002354          4   677  389  406
=A(X'FFFFFFFF'-ASCHOM)
                    A    00002374          4   685  446  463
=A(X'FFFFFFFF'-OAC1A)
                    A    00002350          4   676  387
=A(X'FFFFFFFF'-OAC1KEY-OAC1K)
                    A    00002388          4   690  501
=A(X'FFFFFFFF'-OAC2A)
                    A    00002370          4   684  444
     SYMBOL        TYPE   VALUE      LENGTH   DEFN  REFERENCES

=A(X'FFFFFFFF'-OAC2KEY-OAC2K)
                    A    00002390          4   692  510
=F'0'               F    0000239C          4   695  598
=F'1'               F    000023A0          4   696  600
=F'2'               F    000023A4          4   697  601
=F'3'               F    00002368          4   682  429  486
=P'1'               P    000023B6          1   703  566  637  654
=X'0000000C'        X    00002348          4   674  378
=X'000000C0'        X    00002398          4   694  553
=X'00400000'        X    0000235C          4   679  404
=X'00C00000'        X    00002394          4   693  540
=X'04'              X    000023B4          1   701  335
=X'0C'              X    000023B5          1   702  517
=X'10031003'        X    00002344          4   673  352
```

```
MACRO   DEFN  REFERENCES

No defined macros
```

```
   DESC     SYMBOL    SIZE     POS        ADDR

Entry: 2000

Image      IMAGE     41024  0000-A03F  0000-A03F
  Region             41024  0000-A03F  0000-A03F
    CSECT  MVCOS001  41024  0000-A03F  0000-A03F
   STMT                                        FILE NAME

1     c:\Users\Fish\Documents\Visual Studio 2008\Projects\MyProjects\ASMA-0\MVCOS\MVCOS.asm

** NO ERRORS FOUND **
```