```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                     2 ***********************************************************************
                                                     3 *
                                                     4 *Testcase IEEE CONVERT TO FIXED 32
                                                     5 *  Test case capability includes ieee exceptions trappable and
                                                     6 *  otherwise.  Test result, FPC flags, DXC, and condition code are
                                                     7 *  saved for all tests.
                                                     8 *
                                                     9 *
                                                    10 *                      ********************
                                                    11 *                      **   IMPORTANT!   **
                                                    12 *                      ********************
                                                    13 *
                                                    14 *        This test uses the Hercules Diagnose X'008' interface
                                                    15 *        to display messages and thus your .tst runtest script
                                                    16 *        MUST contain a "DIAG8CMD ENABLE" statement within it!
                                                    17 *
                                                    18 *
                                                    19 ***********************************************************************

                                                    21 ***********************************************************************
                                                    22 *
                                                    23 *                       bfp-006-cvttofix.asm
                                                    24 *
                                                    25 *        This assembly-language source file is part of the
                                                    26 *        Hercules Binary Floating Point Validation Package
                                                    27 *                        by Stephen R. Orso
                                                    28 *
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    57 * OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
                                                    58 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
                                                    59 * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
                                                    60 *
                                                    61 *********************************************************************

                                                    63 *********************************************************************
                                                    64 *
                                                    65 * Tests the following three conversion instructions
                                                    66 *   CONVERT TO FIXED (short BFP to int-32, RRE)
                                                    67 *   CONVERT TO FIXED (long BFP to int-32, RRE)
                                                    68 *   CONVERT TO FIXED (extended BFP to int-32, RRE)
                                                    69 *   CONVERT TO FIXED (short BFP to int-32, RRF-e)
                                                    70 *   CONVERT TO FIXED (long BFP to int-32, RRF-e)
                                                    71 *   CONVERT TO FIXED (extended BFP to int-32, RRF-e)
                                                    72 *
                                                    73 * Test data is compiled into this program.  The test script that runs
                                                    74 * this program can provide alternative test data through Hercules R
                                                    75 * commands.
                                                    76 *
                                                    77 * Test Case Order
                                                    78 * 1) Short BFP to Int-32
                                                    79 * 2) Short BFP to Int-32 with all rounding modes
                                                    80 * 3) Long BFP Int-32
                                                    81 * 3) Long BFP Int-32 with all rounding modes
                                                    82 * 4) Extended BFP to Int-32
                                                    83 * 4) Extended BFP to Int-32 with all rounding modes
                                                    84 *
                                                    85 * Provided test data is:
                                                    86 *       1, 2, 4, -2, QNaN, SNaN, 2 147 483 648, -2 147 483 648.
                                                    87 *   The last two values will trigger inexact exceptions when converted
                                                    88 *   To int-32.  Underflow does not get raised during Convert to Fixed.
                                                    89 * Provided test data for rounding tests:
                                                    90 *      -9.5, -5.5, -2.5, -1.5, -0.5, +0.5, +1.5, +2.5, +5.5, +9.5
                                                    91 *   This data is taken from Table 9-11 on page 9-16 of SA22-7832-10.
                                                    92 *   While the table illustrates LOAD FP INTEGER, the same results
                                                    93 *   should be generated when creating an int-32 or int-64 integer.
                                                    94 *
                                                    95 * Note that three input test data sets are provided, one each for
                                                    96 *   short, long, and extended precision BFP.  All are converted to
                                                    97 *   int-32.
                                                    98 *
                                                    99 * Also tests the following floating point support instructions
                                                   100 *   LOAD  (Short)
                                                   101 *   LOAD  (Long)
                                                   102 *   LOAD FPC
                                                   103 *   SET BFP ROUNDING MODE 2-BIT
                                                   104 *   SET BFP ROUNDING MODE 3-BIT
                                                   105 *   STORE (Short)
                                                   106 *   STORE (Long)
                                                   107 *   STORE FPC
                                                   108 *
                                                   109 *********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   111 *
                                                   112 *  Note: for compatibility with the z/CMS test rig, do not change
                                                   113 *  or use R11, R14, or R15.  Everything else is fair game.
                                                   114 *
                              00000000  00009D4B   115 BFPCVTTF START 0
                              00000000  00000001   116 R0       EQU   0
                              00000001  00000001   117 R1       EQU   1
                              00000002  00000001   118 R2       EQU   2
                              00000003  00000001   119 R3       EQU   3
                              00000004  00000001   120 R4       EQU   4
                              00000005  00000001   121 R5       EQU   5
                              00000006  00000001   122 R6       EQU   6
                              00000007  00000001   123 R7       EQU   7
                              00000008  00000001   124 R8       EQU   8
                              00000009  00000001   125 R9       EQU   9
                              0000000A  00000001   126 R10      EQU   10
                              0000000B  00000001   127 R11      EQU   11
                              0000000C  00000001   128 R12      EQU   12
                              0000000D  00000001   129 R13      EQU   13
                              0000000E  00000001   130 R14      EQU   14
                              0000000F  00000001   131 R15      EQU   15
                                                   132 *
                                                   133 * Floating Point Register equates to keep the cross reference clean
                                                   134 *
                              00000000  00000001   135 FPR0     EQU   0
                              00000001  00000001   136 FPR1     EQU   1
                              00000002  00000001   137 FPR2     EQU   2
                              00000003  00000001   138 FPR3     EQU   3
                              00000004  00000001   139 FPR4     EQU   4
                              00000005  00000001   140 FPR5     EQU   5
                              00000006  00000001   141 FPR6     EQU   6
                              00000007  00000001   142 FPR7     EQU   7
                              00000008  00000001   143 FPR8     EQU   8
                              00000009  00000001   144 FPR9     EQU   9
                              0000000A  00000001   145 FPR10    EQU   10
                              0000000B  00000001   146 FPR11    EQU   11
                              0000000C  00000001   147 FPR12    EQU   12
                              0000000D  00000001   148 FPR13    EQU   13
                              0000000E  00000001   149 FPR14    EQU   14
                              0000000F  00000001   150 FPR15    EQU   15
                                                   151 *
00000000                      00000000             152          USING *,R15
00000000                      00009980             153          USING HELPERS,R12
                                                   154 *
                                                   155 * Above works on real iron (R15=0 after sysclear)
                                                   156 * and in z/CMS (R15 points to start of load module)
                                                   157 *


                                                   159 *********************************************************************
                                                   160 *
                                                   161 * Low core definitions, Restart PSW, and Program Check Routine.
                                                   162 *
                                                   163 *********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000000                    00000000  0000008E   165          ORG   BFPCVTTF+X'8E'      Program check interrution code
0000008E  0000                                   166 PCINTCD  DS    H
                                                 167 *
                            00000150  00000000   168 PCOLDPSW EQU   BFPCVTTF+X'150'     z/Arch Program check old PSW
                                                 169 *
00000090                    00000090  000001A0   170          ORG   BFPCVTTF+X'1A0'     z/Arch Restart PSW
000001A0  00000001 80000000                      171          DC    X'0000000180000000',AD(START)
                                                 172 *
000001B0                    000001B0  000001D0   173          ORG   BFPCVTTF+X'1D0'     z/Arch Program check NEW PSW
000001D0  00000000 00000000                      174          DC    X'0000000000000000',AD(PROGCHK)
                                                 175 *
                                                 176 * Program check routine.  If Data Exception, continue execution at
                                                 177 * the instruction following the program check.  Otherwise, hard wait.
                                                 178 * No need to collect data.  All interesting DXC stuff is captured
                                                 179 * in the FPCR.
                                                 180 *
000001E0                    000001E0  00000200   181          ORG   BFPCVTTF+X'200'
00000200                                         182 PROGCHK  DS    0H              Program check occured...
00000200  9507 F08F                   0000008F   183          CLI   PCINTCD+1,X'07'  Data Exception?
00000204  A774 0004                   0000020C   184          JNE   PCNOTDTA        ..no, hardwait (not sure if R15 is ok)
00000208  B2B2 F150                   00000150   185          LPSWE PCOLDPSW        ..yes, resume program execution

0000020C  900F F23C                   0000023C   187 PCNOTDTA STM   R0,R15,SAVEREGS  Save registers
00000210  58C0 F27C                   0000027C   188          L     R12,AHELPERS     Get address of helper subroutines
00000214  4DD0 C000                   00009980   189          BAS   R13,PGMCK        Report this unexpected program check
00000218  980F F23C                   0000023C   190          LM    R0,R15,SAVEREGS  Restore registers

0000021C  12EE                                   192          LTR   R14,R14         Return address provided?
0000021E  077E                                   193          BNZR  R14             Yes, return to z/CMS test rig.
00000220  B2B2 F228                   00000228   194          LPSWE PROGPSW         Not data exception, enter disabled wait
00000228  00020000 00000000                      195 PROGPSW  DC    0D'0',X'0002000000000000',XL6'00',X'DEAD' Abnormal end
00000238  B2B2 F2E0                   000002E0   196 FAIL     LPSWE FAILPSW         Not data exception, enter disabled wait
0000023C  00000000 00000000                      197 SAVEREGS DC    16F'0'          Registers save area
0000027C  00009980                               198 AHELPERS DC    A(HELPERS)      Address of helper subroutines
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   200 ********************************************************************
                                                   201 *
                                                   202 *  Main program.  Enable Advanced Floating Point, process test cases.
                                                   203 *
                                                   204 ********************************************************************


00000280  B600 F2F0                     000002F0   206 START    STCTL R0,R0,CTLR0    Store CR0 to enable AFP
00000284  9604 F2F1                     000002F1   207          OI    CTLR0+1,X'04'  Turn on AFP bit
00000288  B700 F2F0                     000002F0   208          LCTL  R0,R0,CTLR0    Reload updated CR0
                                                   209 *
                                                   210 * Short BFP Input testing
                                                   211 *
0000028C  41A0 F2FC                     000002FC   212          LA    R10,SHORTS     Point to short BFP test inputs
00000290  4DD0 F35C                     0000035C   213          BAS   R13,CFEBR      Convert values to fixed from short BFP
00000294  41A0 F32C                     0000032C   214          LA    R10,RMSHORTS   Point to inputs for rounding mode tests
00000298  4DD0 F3BA                     000003BA   215          BAS   R13,CFEBRA     Convert using all rounding mode options
                                                   216 *
                                                   217 * Short BFP Input testing
                                                   218 *
0000029C  41A0 F30C                     0000030C   219          LA    R10,LONGS      Point to long BFP test inputs
000002A0  4DD0 F504                     00000504   220          BAS   R13,CFDBR      Convert values to fixed from long BFP
000002A4  41A0 F33C                     0000033C   221          LA    R10,RMLONGS    Point to inputs for rounding mode tests
000002A8  4DD0 F562                     00000562   222          BAS   R13,CFDBRA     Convert using all rounding mode options
                                                   223 *
                                                   224 * Short BFP Input testing
                                                   225 *
000002AC  41A0 F31C                     0000031C   226          LA    R10,EXTDS      Point to extended BFP test inputs
000002B0  4DD0 F6AC                     000006AC   227          BAS   R13,CFXBR      Convert values to fixed from extended
000002B4  41A0 F34C                     0000034C   228          LA    R10,RMEXTDS    Point to inputs for rounding mode tests
000002B8  4DD0 F70E                     0000070E   229          BAS   R13,CFXBRA     Convert using all rounding mode options
                                                   230 *
                                                   231 ********************************************************************
                                                   232 *                    Verify test results...
                                                   233 ********************************************************************
                                                   234 *
000002BC  58C0 F27C                     0000027C   235          L     R12,AHELPERS     Get address of helper subroutines
000002C0  4DD0 C0A0                     00009A20   236          BAS   R13,VERISUB      Go verify results
000002C4  12EE                                     237          LTR   R14,R14          Was return address provided?
000002C6  077E                                     238          BNZR  R14              Yes, return to z/CMS test rig.
000002C8  B2B2 F2D0                     000002D0   239          LPSWE GOODPSW          Load SUCCESS PSW
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000002D0                                           241          DS    0D            Ensure correct alignment for PSW
000002D0  00020000 00000000                        242 GOODPSW  DC    X'0002000000000000',AD(0)  Normal end - disabled wait
000002E0  00020000 00000000                        243 FAILPSW  DC    X'0002000000000000',XL6'00',X'0BAD' Abnormal end
                                                   244 *
000002F0  00000000                                 245 CTLR0    DS    F
000002F4  00000000                                 246 FPCREGNT DC    X'00000000'  FPCR, trap all IEEE exceptions, zero flags
000002F8  F8000000                                 247 FPCREGTR DC    X'F8000000'  FPCR, trap no IEEE exceptions, zero flags
                                                   248 *
                                                   249 * Input values parameter list, four fullwords:
                                                   250 *      1) Count,
                                                   251 *      2) Address of inputs,
                                                   252 *      3) Address to place results, and
                                                   253 *      4) Address to place DXC/Flags/cc values.
                                                   254 *
000002FC                                           255 SHORTS   DS    0F           Inputs for short BFP testing
000002FC  00000009                                 256          DC    A(SBFPCT/4)
00000300  0000085C                                 257          DC    A(SBFPIN)
00000304  00001000                                 258          DC    A(SINTOUT)
00000308  00001100                                 259          DC    A(SINTFLGS)
                                                   260 *
0000030C                                           261 LONGS    DS    0F           Inputs for long BFP testing
0000030C  0000000A                                 262          DC    A(LBFPCT/8)
00000310  000008B8                                 263          DC    A(LBFPIN)
00000314  00002000                                 264          DC    A(LINTOUT)
00000318  00002100                                 265          DC    A(LINTFLGS)
                                                   266 *
0000031C                                           267 EXTDS    DS    0F           Inputs for Extended BFP testing
0000031C  0000000A                                 268          DC    A(XBFPCT/16)
00000320  00000980                                 269          DC    A(XBFPIN)
00000324  00003000                                 270          DC    A(XINTOUT)
00000328  00003100                                 271          DC    A(XINTFLGS)
                                                   272 *
0000032C                                           273 RMSHORTS DS    0F           Inputs for long BFP rounding mode tests
0000032C  0000000E                                 274          DC    A(SBFPRMCT/4)
00000330  00000880                                 275          DC    A(SBFPINRM)  Short BFP rounding mode test inputs
00000334  00001200                                 276          DC    A(SINTRMO)   Space for rounding mode test results
00000338  00001600                                 277          DC    A(SINTRMOF)  Space for rounding mode test flags
                                                   278 *
0000033C                                           279 RMLONGS  DS    0F           Inputs for long BFP rounding mode tests
0000033C  0000000F                                 280          DC    A(LBFPRMCT/8)
00000340  00000908                                 281          DC    A(LBFPINRM)  Long BFP rounding mode test inputs
00000344  00002200                                 282          DC    A(LINTRMO)   Space for rounding mode tests results
00000348  00002600                                 283          DC    A(LINTRMOF)  Space for rounding mode test flags
                                                   284 *
0000034C                                           285 RMEXTDS  DS    0F           Inputs for ext'd BFP rounding mode tests
0000034C  0000000F                                 286          DC    A(XBFPRMCT/16)
00000350  00000A20                                 287          DC    A(XBFPINRM)  Extended BFP rounding mode test inputs
00000354  00003200                                 288          DC    A(XINTRMO)   Space for rounding mode results
00000358  00003600                                 289          DC    A(XINTRMOF)  Space for rounding mode test flags
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   291 ********************************************************************
                                                   292 *
                                                   293 * Convert short BFP to integer-32 format.  A pair of results is
                                                   294 * generated for each input: one with all exceptions non-trappable, and
                                                   295 * the second with all exceptions trappable.   The FPCR and condition
                                                   296 * code is stored for each result.
                                                   297 *
                                                   298 ********************************************************************


0000035C  9823 A000                  00000000    300 CFEBR    LM    R2,R3,0(R10)  Get count and address of test input values
00000360  9878 A008                  00000008    301          LM    R7,R8,8(R10)  Get address of result area and flag area.
00000364  1222                                   302          LTR   R2,R2         Any test cases?
00000366  078D                                   303          BZR   R13           ..No, return to caller
00000368  0DC0                                   304          BASR  R12,0         Set top of loop
                                                 305 *
0000036A  7880 3000                  00000000    306          LE    FPR8,0(,R3)   Get short BFP test value
0000036E  B29D F2F4                  000002F4    307          LFPC  FPCREGNT      Set exceptions non-trappable
00000372  B398 0018                              308          CFEBR R1,0,FPR8     Cvt float in FPR8 to Int in GPR1
00000376  5010 7000                  00000000    309          ST    R1,0(,R7)     Store int-32 result
0000037A  B29C 8000                  00000000    310          STFPC 0(R8)         Store resulting FPC flags and DXC
0000037E  B222 0010                              311          IPM   R1            Get condition code and program mask
00000382  8810 001C                  0000001C    312          SRL   R1,28         Isolate CC in low order byte
00000386  4210 8003                  00000003    313          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                                 314 *
0000038A  B29D F2F8                  000002F8    315          LFPC  FPCREGTR      Set exceptions trappable
0000038E  1711                                   316          XR    R1,R1         Clear any residual result in R1
00000390  0410                                   317          SPM   R1            Clear out any residual nz condition code
00000392  B398 0018                              318          CFEBR R1,0,FPR8     Cvt float in FPR8 to Int in GPR1
00000396  5010 7004                  00000004    319          ST    R1,4(,R7)     Store short BFP result
0000039A  B29C 8004                  00000004    320          STFPC 4(R8)         Store resulting FPC flags and DXC
0000039E  B222 0010                              321          IPM   R1            Get condition code and program mask
000003A2  8810 001C                  0000001C    322          SRL   R1,28         Isolate CC in low order byte
000003A6  4210 8007                  00000007    323          STC   R1,7(,R8)     Save CC as low byte of FPCR
                                                 324 *
000003AA  4130 3004                  00000004    325          LA    R3,4(,R3)     Point to next input value
000003AE  4170 7008                  00000008    326          LA    R7,8(,R7)     Point to next int-32 converted value pair
000003B2  4180 8008                  00000008    327          LA    R8,8(,R8)     Point to next FPCR/CC result area
000003B6  062C                                   328          BCTR  R2,R12        Convert next input value.
000003B8  07FD                                   329          BR    R13           All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   331 ***********************************************************************
                                                   332 *
                                                   333 * Convert short BFP to int-32 using each possible rounding mode.
                                                   334 * Ten test results are generated for each input.  A 48-byte test
                                                   335 * result section is used to keep results sets aligned on a quad-double
                                                   336 * word.
                                                   337 *
                                                   338 * The first four tests use rounding modes specified in the FPC with
                                                   339 * the IEEE Inexact exception supressed.  SRNM (2-bit) is used  for the
                                                   340 * first two FPCR-controlled tests and SRNMB (3-bit) is used for the
                                                   341 * last two to get full coverage of that instruction pair.
                                                   342 *
                                                   343 * The next six results use instruction-specified rounding modes.
                                                   344 *
                                                   345 * The default rounding mode (0 for RNTE) is not tested in this section;
                                                   346 * prior tests used the default rounding mode.  RNTE is tested
                                                   347 * explicitly as a rounding mode in this section.
                                                   348 *
                                                   349 ***********************************************************************

000003BA  9823 A000                 00000000   351 CFEBRA   LM    R2,R3,0(R10)  Get count and address of test input values
000003BE  9878 A008                 00000008   352          LM    R7,R8,8(R10)  Get address of result area and flag area.
000003C2  1222                                 353          LTR   R2,R2         Any test cases?
000003C4  078D                                 354          BZR   R13           ..No, return to caller
000003C6  0DC0                                 355          BASR  R12,0         Set top of loop
                                               356 *
000003C8  7880 3000                 00000000   357          LE    FPR8,0(,R3)    Get short BFP test value
                                               358 *
                                               359 * Test cases using rounding mode specified in the FPCR
                                               360 *
000003CC  B29D F2F4                 000002F4   361          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000003D0  B299 0001                 00000001   362          SRNM  1             SET FPC to RZ, towards zero.
000003D4  B398 0418                            363          CFEBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding, inexact masked
000003D8  5010 7000                 00000000   364          ST    R1,0*4(,R7)   Store integer-32 result
000003DC  B29C 8000                 00000000   365          STFPC 0(R8)         Store resulting FPC flags and DXC
000003E0  B222 0010                            366          IPM   R1            Get condition code and program mask
000003E4  8810 001C                 0000001C   367          SRL   R1,28         Isolate CC in low order byte
000003E8  4210 8003                 00000003   368          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                               369 *
000003EC  B29D F2F4                 000002F4   370          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000003F0  B299 0002                 00000002   371          SRNM  2             SET FPC to RP, to +infinity
000003F4  B398 0418                            372          CFEBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
000003F8  5010 7004                 00000004   373          ST    R1,1*4(,R7)   Store integer-32 result
000003FC  B29C 8004                 00000004   374          STFPC 1*4(R8)       Store resulting FPC flags and DXC
00000400  B222 0010                            375          IPM   R1            Get condition code and program mask
00000404  8810 001C                 0000001C   376          SRL   R1,28         Isolate CC in low order byte
00000408  4210 8007                 00000007   377          STC   R1,(1*4)+3(,R8)    Save CC as low byte of FPCR
                                               378 *
0000040C  B29D F2F4                 000002F4   379          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000410  B2B8 0003                 00000003   380          SRNMB 3             SET FPC to RM, to -infinity
00000414  B398 0418                            381          CFEBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000418  5010 7008                 00000008   382          ST    R1,2*4(,R7)   Store integer-32 result
0000041C  B29C 8008                 00000008   383          STFPC 2*4(R8)       Store resulting FPC flags and DXC
00000420  B222 0010                            384          IPM   R1            Get condition code and program mask
00000424  8810 001C                 0000001C   385          SRL   R1,28         Isolate CC in low order byte
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000428  4210 800B                     0000000B   386          STC   R1,(2*4)+3(,R8)    Save CC as low byte of FPCR
                                                   387 *
0000042C  B29D F2F4                     000002F4   388          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000430  B2B8 0007                     00000007   389          SRNMB 7             RFS, Prepare for Shorter Precision
00000434  B398 0418                                390          CFEBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000438  5010 700C                     0000000C   391          ST    R1,3*4(,R7)   Store integer-32 result
0000043C  B29C 800C                     0000000C   392          STFPC 3*4(R8)       Store resulting FPC flags and DXC
00000440  B222 0010                                393          IPM   R1            Get condition code and program mask
00000444  8810 001C                     0000001C   394          SRL   R1,28         Isolate CC in low order byte
00000448  4210 800F                     0000000F   395          STC   R1,(3*4)+3(,R8)    Save CC as low byte of FPCR
                                                   396 *
                                                   397 * Test cases using rounding mode specified in the instruction M3 field
                                                   398 *
0000044C  B29D F2F4                     000002F4   399          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000450  B398 1018                                400          CFEBRA R1,1,FPR8,B'0000'  RNTA, to nearest, ties away
00000454  5010 7010                     00000010   401          ST    R1,4*4(,R7)   Store integer-32 result
00000458  B29C 8010                     00000010   402          STFPC 4*4(R8)       Store resulting FPC flags and DXC
0000045C  B222 0010                                403          IPM   R1            Get condition code and program mask
00000460  8810 001C                     0000001C   404          SRL   R1,28         Isolate CC in low order byte
00000464  4210 8013                     00000013   405          STC   R1,(4*4)+3(,R8)    Save CC as low byte of FPCR
                                                   406 *
00000468  B29D F2F4                     000002F4   407          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
0000046C  B398 3018                                408          CFEBRA R1,3,FPR8,B'0000'  RFS, prepare for shorter precision
00000470  5010 7014                     00000014   409          ST    R1,5*4(,R7)   Store integer-32 result
00000474  B29C 8014                     00000014   410          STFPC 5*4(R8)       Store resulting FPC flags and DXC
00000478  B222 0010                                411          IPM   R1            Get condition code and program mask
0000047C  8810 001C                     0000001C   412          SRL   R1,28         Isolate CC in low order byte
00000480  4210 8017                     00000017   413          STC   R1,(5*4)+3(,R8)    Save CC as low byte of FPCR
                                                   414 *
00000484  B29D F2F4                     000002F4   415          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000488  B398 4018                                416          CFEBRA R1,4,FPR8,B'0000'  RNTE, to nearest, ties to even
0000048C  5010 7018                     00000018   417          ST    R1,6*4(,R7)   Store integer-32 result
00000490  B29C 8018                     00000018   418          STFPC 6*4(R8)       Store resulting FPC flags and DXC
00000494  B222 0010                                419          IPM   R1            Get condition code and program mask
00000498  8810 001C                     0000001C   420          SRL   R1,28         Isolate CC in low order byte
0000049C  4210 801B                     0000001B   421          STC   R1,(6*4)+3(,R8)    Save CC as low byte of FPCR
                                                   422 *
000004A0  B29D F2F4                     000002F4   423          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000004A4  B398 5018                                424          CFEBRA R1,5,FPR8,B'0000'  RZ, toward zero
000004A8  5010 701C                     0000001C   425          ST    R1,7*4(,R7)   Store integer-32 result
000004AC  B29C 801C                     0000001C   426          STFPC 7*4(R8)       Store resulting FPC flags and DXC
000004B0  B222 0010                                427          IPM   R1            Get condition code and program mask
000004B4  8810 001C                     0000001C   428          SRL   R1,28         Isolate CC in low order byte
000004B8  4210 801F                     0000001F   429          STC   R1,(7*4)+3(,R8)    Save CC as low byte of FPCR
                                                   430 *
000004BC  B29D F2F4                     000002F4   431          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000004C0  B398 6018                                432          CFEBRA R1,6,FPR8,B'0000'  RP, to +inf
000004C4  5010 7020                     00000020   433          ST    R1,8*4(,R7)   Store integer-32 result
000004C8  B29C 8020                     00000020   434          STFPC 8*4(R8)       Store resulting FPC flags and DXC
000004CC  B222 0010                                435          IPM   R1            Get condition code and program mask
000004D0  8810 001C                     0000001C   436          SRL   R1,28         Isolate CC in low order byte
000004D4  4210 8023                     00000023   437          STC   R1,(8*4)+3(,R8)    Save CC as low byte of FPCR
                                                   438 *
000004D8  B29D F2F4                     000002F4   439          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000004DC  B398 7018                                440          CFEBRA R1,7,FPR8,B'0000'  RM, to -inf
000004E0  5010 7024                     00000024   441          ST    R1,9*4(,R7)   Store integer-32 result
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000004E4  B29C 8024                    00000024   442          STFPC 9*4(R8)       Store resulting FPC flags and DXC
000004E8  B222 0010                               443          IPM   R1            Get condition code and program mask
000004EC  8810 001C                    0000001C   444          SRL   R1,28         Isolate CC in low order byte
000004F0  4210 8027                    00000027   445          STC   R1,(9*4)+3(,R8)    Save CC as low byte of FPCR
                                                  446 *
000004F4  4130 3004                    00000004   447          LA    R3,4(,R3)     Point to next input value
000004F8  4170 7030                    00000030   448          LA    R7,12*4(,R7)  Point to next int-32 converted value set
000004FC  4180 8030                    00000030   449          LA    R8,12*4(,R8)  Point to next FPCR/CC result area
00000500  062C                                    450          BCTR  R2,R12        Convert next input value.
00000502  07FD                                    451          BR    R13           All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   453 ********************************************************************
                                                   454 *
                                                   455 * Convert long BFP inputs to integer-32.  A pair of results is
                                                   456 * generated for each input: one with all exceptions non-trappable, and
                                                   457 * the second with all exceptions trappable.   The FPCR and condition
                                                   458 * code is stored for each result.
                                                   459 *
                                                   460 ********************************************************************


00000504  9823 A000                    00000000    462 CFDBR    LM    R2,R3,0(R10)  Get count and address of test input values
00000508  9878 A008                    00000008    463          LM    R7,R8,8(R10)  Get address of result area and flag area.
0000050C  1222                                     464          LTR   R2,R2         Any test cases?
0000050E  078D                                     465          BZR   R13           ..No, return to caller
00000510  0DC0                                     466          BASR  R12,0         Set top of loop
                                                   467 *
00000512  6880 3000                    00000000    468          LD    FPR8,0(,R3)   Get long BFP test value
00000516  B29D F2F4                    000002F4    469          LFPC  FPCREGNT      Set exceptions non-trappable
0000051A  B399 0018                                470          CFDBR R1,0,FPR8     Cvt float in FPR8 to Int in GPR1
0000051E  5010 7000                    00000000    471          ST    R1,0(,R7)     Store long BFP result
00000522  B29C 8000                    00000000    472          STFPC 0(R8)         Store resulting FPC flags and DXC
00000526  B222 0010                                473          IPM   R1            Get condition code and program mask
0000052A  8810 001C                    0000001C    474          SRL   R1,28         Isolate CC in low order byte
0000052E  4210 8003                    00000003    475          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                                   476 *
00000532  B29D F2F8                    000002F8    477          LFPC  FPCREGTR      Set exceptions trappable
00000536  1711                                     478          XR    R1,R1         Clear any residual result in R1
00000538  0410                                     479          SPM   R1            Clear out any residual nz condition code
0000053A  B399 0018                                480          CFDBR R1,0,FPR8     Cvt float in FPR8 to Int in GPR1
0000053E  5010 7004                    00000004    481          ST    R1,4(,R7)     Store int-32 result
00000542  B29C 8004                    00000004    482          STFPC 4(R8)         Store resulting FPC flags and DXC
00000546  B222 0010                                483          IPM   R1            Get condition code and program mask
0000054A  8810 001C                    0000001C    484          SRL   R1,28         Isolate CC in low order byte
0000054E  4210 8007                    00000007    485          STC   R1,7(,R8)     Save CC as low byte of FPCR
                                                   486 *
00000552  4130 3008                    00000008    487          LA    R3,8(,R3)     Point to next input value
00000556  4170 7008                    00000008    488          LA    R7,8(,R7)     Point to next int-32 converted value pair
0000055A  4180 8008                    00000008    489          LA    R8,8(,R8)     Point to next FPCR/CC result area
0000055E  062C                                     490          BCTR  R2,R12        Convert next input value.
00000560  07FD                                     491          BR    R13           All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   493 ******************************************************************
                                                   494 *
                                                   495 * Convert long BFP to int-32 using each possible rounding mode.
                                                   496 * Ten test results are generated for each input.  A 48-byte test result
                                                   497 * section is used to keep results sets aligned on a quad-double word.
                                                   498 *
                                                   499 * The first four tests use rounding modes specified in the FPC with the
                                                   500 * IEEE Inexact exception supressed.  SRNM (2-bit) is used  for the first
                                                   501 * two FPCR-controlled tests and SRNMB (3-bit) is used for the last two
                                                   502 * to get full coverage of that instruction pair.
                                                   503 *
                                                   504 * The next six results use instruction-specified rounding modes.
                                                   505 *
                                                   506 * The default rounding mode (0 for RNTE) is not tested in this section;
                                                   507 * prior tests used the default rounding mode.  RNTE is tested explicitly
                                                   508 * as a rounding mode in this section.
                                                   509 *
                                                   510 ******************************************************************


00000562  9823 A000                  00000000   512 CFDBRA   LM    R2,R3,0(R10)  Get count and address of test input values
00000566  9878 A008                  00000008   513          LM    R7,R8,8(R10)  Get address of result area and flag area.
0000056A  1222                                  514          LTR   R2,R2         Any test cases?
0000056C  078D                                  515          BZR   R13           ..No, return to caller
0000056E  0DC0                                  516          BASR  R12,0         Set top of loop
                                                517 *
00000570  6880 3000                  00000000   518          LD    FPR8,0(,R3)    Get long BFP test value
                                                519 *
                                                520 * Test cases using rounding mode specified in the FPCR
                                                521 *
00000574  B29D F2F4                  000002F4   522          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000578  B299 0001                  00000001   523          SRNM  1             SET FPC to RZ, towards zero.
0000057C  B399 0418                             524          CFDBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000580  5010 7000                  00000000   525          ST    R1,0*4(,R7)   Store integer-32 result
00000584  B29C 8000                  00000000   526          STFPC 0(R8)         Store resulting FPC flags and DXC
00000588  B222 0010                             527          IPM   R1            Get condition code and program mask
0000058C  8810 001C                  0000001C   528          SRL   R1,28         Isolate CC in low order byte
00000590  4210 8003                  00000003   529          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                                530 *
00000594  B29D F2F4                  000002F4   531          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000598  B299 0002                  00000002   532          SRNM  2             SET FPC to RP, to +infinity
0000059C  B399 0418                             533          CFDBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
000005A0  5010 7004                  00000004   534          ST    R1,1*4(,R7)   Store integer-32 result
000005A4  B29C 8004                  00000004   535          STFPC 1*4(R8)       Store resulting FPC flags and DXC
000005A8  B222 0010                             536          IPM   R1            Get condition code and program mask
000005AC  8810 001C                  0000001C   537          SRL   R1,28         Isolate CC in low order byte
000005B0  4210 8007                  00000007   538          STC   R1,(1*4)+3(,R8)    Save CC as low byte of FPCR
                                                539 *
000005B4  B29D F2F4                  000002F4   540          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000005B8  B2B8 0003                  00000003   541          SRNMB 3             SET FPC to RM, to -infinity
000005BC  B399 0418                             542          CFDBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
000005C0  5010 7008                  00000008   543          ST    R1,2*4(,R7)   Store integer-32 result
000005C4  B29C 8008                  00000008   544          STFPC 2*4(R8)       Store resulting FPC flags and DXC
000005C8  B222 0010                             545          IPM   R1            Get condition code and program mask
000005CC  8810 001C                  0000001C   546          SRL   R1,28         Isolate CC in low order byte
000005D0  4210 800B                  0000000B   547          STC   R1,(2*4)+3(,R8)    Save CC as low byte of FPCR
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   548 *
000005D4  B29D F2F4                      000002F4   549          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000005D8  B2B8 0007                      00000007   550          SRNMB 7             RFS, Prepare for Shorter Precision
000005DC  B399 0418                                 551          CFDBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
000005E0  5010 700C                      0000000C   552          ST    R1,3*4(,R7)   Store integer-32 result
000005E4  B29C 800C                      0000000C   553          STFPC 3*4(R8)       Store resulting FPC flags and DXC
000005E8  B222 0010                                 554          IPM   R1            Get condition code and program mask
000005EC  8810 001C                      0000001C   555          SRL   R1,28         Isolate CC in low order byte
000005F0  4210 800F                      0000000F   556          STC   R1,(3*4)+3(,R8)    Save CC as low byte of FPCR
                                                   557 *
                                                   558 * Test cases using rounding mode specified in the instruction M3 field
                                                   559 *
000005F4  B29D F2F4                      000002F4   560          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000005F8  B399 1018                                 561          CFDBRA R1,1,FPR8,B'0000'  RNTA, to nearest, ties away
000005FC  5010 7010                      00000010   562          ST    R1,4*4(,R7)   Store integer-32 result
00000600  B29C 8010                      00000010   563          STFPC 4*4(R8)       Store resulting FPC flags and DXC
00000604  B222 0010                                 564          IPM   R1            Get condition code and program mask
00000608  8810 001C                      0000001C   565          SRL   R1,28         Isolate CC in low order byte
0000060C  4210 8013                      00000013   566          STC   R1,(4*4)+3(,R8)    Save CC as low byte of FPCR
                                                   567 *
00000610  B29D F2F4                      000002F4   568          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000614  B399 3018                                 569          CFDBRA R1,3,FPR8,B'0000'  RFS, prepare for shorter precision
00000618  5010 7014                      00000014   570          ST    R1,5*4(,R7)   Store integer-32 result
0000061C  B29C 8014                      00000014   571          STFPC 5*4(R8)       Store resulting FPC flags and DXC
00000620  B222 0010                                 572          IPM   R1            Get condition code and program mask
00000624  8810 001C                      0000001C   573          SRL   R1,28         Isolate CC in low order byte
00000628  4210 8017                      00000017   574          STC   R1,(5*4)+3(,R8)    Save CC as low byte of FPCR
                                                   575 *
0000062C  B29D F2F4                      000002F4   576          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000630  B399 4018                                 577          CFDBRA R1,4,FPR8,B'0000'  RNTE, to nearest, ties to even
00000634  5010 7018                      00000018   578          ST    R1,6*4(,R7)   Store integer-32 result
00000638  B29C 8018                      00000018   579          STFPC 6*4(R8)       Store resulting FPC flags and DXC
0000063C  B222 0010                                 580          IPM   R1            Get condition code and program mask
00000640  8810 001C                      0000001C   581          SRL   R1,28         Isolate CC in low order byte
00000644  4210 801B                      0000001B   582          STC   R1,(6*4)+3(,R8)    Save CC as low byte of FPCR
                                                   583 *
00000648  B29D F2F4                      000002F4   584          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
0000064C  B399 5018                                 585          CFDBRA R1,5,FPR8,B'0000'  RZ, toward zero
00000650  5010 701C                      0000001C   586          ST    R1,7*4(,R7)   Store integer-32 result
00000654  B29C 801C                      0000001C   587          STFPC 7*4(R8)       Store resulting FPC flags and DXC
00000658  B222 0010                                 588          IPM   R1            Get condition code and program mask
0000065C  8810 001C                      0000001C   589          SRL   R1,28         Isolate CC in low order byte
00000660  4210 801F                      0000001F   590          STC   R1,(7*4)+3(,R8)    Save CC as low byte of FPCR
                                                   591 *
00000664  B29D F2F4                      000002F4   592          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000668  B399 6018                                 593          CFDBRA R1,6,FPR8,B'0000'  RP, to +inf
0000066C  5010 7020                      00000020   594          ST    R1,8*4(,R7)   Store integer-32 result
00000670  B29C 8020                      00000020   595          STFPC 8*4(R8)       Store resulting FPC flags and DXC
00000674  B222 0010                                 596          IPM   R1            Get condition code and program mask
00000678  8810 001C                      0000001C   597          SRL   R1,28         Isolate CC in low order byte
0000067C  4210 8023                      00000023   598          STC   R1,(8*4)+3(,R8)    Save CC as low byte of FPCR
                                                   599 *
00000680  B29D F2F4                      000002F4   600          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000684  B399 7018                                 601          CFDBRA R1,7,FPR8,B'0000'  RM, to -inf
00000688  5010 7024                      00000024   602          ST    R1,9*4(,R7)   Store integer-32 result
0000068C  B29C 8024                      00000024   603          STFPC 9*4(R8)       Store resulting FPC flags and DXC
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000690  B222 0010                                604          IPM   R1              Get condition code and program mask
00000694  8810 001C                      0000001C  605          SRL   R1,28           Isolate CC in low order byte
00000698  4210 8027                      00000027  606          STC   R1,(9*4)+3(,R8)     Save CC as low byte of FPCR
                                                   607 *
0000069C  4130 3008                      00000008  608          LA    R3,8(,R3)       Point to next input values
000006A0  4170 7030                      00000030  609          LA    R7,12*4(,R7)  Point to next int-32 converted value set
000006A4  4180 8030                      00000030  610          LA    R8,12*4(,R8)  Point to next FPCR/CC result area
000006A8  062C                                     611          BCTR  R2,R12          Convert next input value.
000006AA  07FD                                     612          BR    R13             All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   614 ********************************************************************
                                                   615 *
                                                   616 * Convert extended BFP to integer-32.  A pair of results is generated
                                                   617 * for each input: one with all exceptions non-trappable, and the
                                                   618 * second with all exceptions trappable.   The FPCR and condition code
                                                   619 * are stored for each result.
                                                   620 *
                                                   621 ********************************************************************


000006AC  9823 A000                    00000000    623 CFXBR    LM    R2,R3,0(R10)  Get count and address of test input values
000006B0  9878 A008                    00000008    624          LM    R7,R8,8(R10)  Get address of result area and flag area.
000006B4  1222                                     625          LTR   R2,R2         Any test cases?
000006B6  078D                                     626          BZR   R13           ..No, return to caller
000006B8  0DC0                                     627          BASR  R12,0         Set top of loop
                                                   628 *
000006BA  6880 3000                    00000000    629          LD    FPR8,0(,R3)   Get extended BFP test value part 1
000006BE  68A0 3008                    00000008    630          LD    FPR10,8(,R3)  Get extended BFP test value part 1
000006C2  B29D F2F4                    000002F4    631          LFPC  FPCREGNT      Set exceptions non-trappable
000006C6  B39A 0018                                632          CFXBR R1,0,FPR8     Cvt float in FPR8-FPR10 to Int-32 in GPR1
000006CA  5010 7000                    00000000    633          ST    R1,0(,R7)     Store integer-32 result
000006CE  B29C 8000                    00000000    634          STFPC 0(R8)         Store resulting FPC flags and DXC
000006D2  B222 0010                                635          IPM   R1            Get condition code and program mask
000006D6  8810 001C                    0000001C    636          SRL   R1,28         Isolate CC in low order byte
000006DA  4210 8003                    00000003    637          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                                   638 *
000006DE  B29D F2F8                    000002F8    639          LFPC  FPCREGTR      Set exceptions trappable
000006E2  1711                                     640          XR    R1,R1         Clear any residual result in R1
000006E4  0410                                     641          SPM   R1            Clear out any residual nz condition code
000006E6  B39A 0018                                642          CFXBR R1,0,FPR8     Cvt float in FPR8-FPR10 to Int-32 in GPR1
000006EA  5010 7004                    00000004    643          ST    R1,4(,R7)     Store integer-32 result
000006EE  B29C 8004                    00000004    644          STFPC 4(R8)         Store resulting FPC flags and DXC
000006F2  B222 0010                                645          IPM   R1            Get condition code and program mask
000006F6  8810 001C                    0000001C    646          SRL   R1,28         Isolate CC in low order byte
000006FA  4210 8007                    00000007    647          STC   R1,7(,R8)     Save CC as low byte of FPCR
                                                   648 *
000006FE  4130 3010                    00000010    649          LA    R3,16(,R3)    Point to next extended BFP input value
00000702  4170 7008                    00000008    650          LA    R7,8(,R7)     Point to next int-32 converted value pair
00000706  4180 8008                    00000008    651          LA    R8,8(,R8)     Point to next FPCR/CC result area
0000070A  062C                                     652          BCTR  R2,R12        Convert next input value.
0000070C  07FD                                     653          BR    R13           All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   655 ******************************************************************
                                                   656 *
                                                   657 * Convert extended BFP to int-32 using each possible rounding mode.
                                                   658 * Ten test results are generated for each input.  A 48-byte test result
                                                   659 * section is used to keep results sets aligned on a quad-double word.
                                                   660 *
                                                   661 * The first four tests use rounding modes specified in the FPC with the
                                                   662 * IEEE Inexact exception supressed.  SRNM (2-bit) is used  for the
                                                   663 * first two FPCR-controlled tests and SRNMB (3-bit) is used for the
                                                   664 * last two To get full coverage of that instruction pair.
                                                   665 *
                                                   666 * The next six results use instruction-specified rounding modes.
                                                   667 *
                                                   668 * The default rounding mode (0 for RNTE) is not tested in this section;
                                                   669 * prior tests used the default rounding mode.  RNTE is tested
                                                   670 * explicitly as a rounding mode in this section.
                                                   671 *
                                                   672 ******************************************************************


0000070E  9823 A000                  00000000    674 CFXBRA   LM    R2,R3,0(R10)  Get count and address of test input values
00000712  9878 A008                  00000008    675          LM    R7,R8,8(R10)  Get address of result area and flag area.
00000716  1222                                   676          LTR   R2,R2         Any test cases?
00000718  078D                                   677          BZR   R13           ..No, return to caller
0000071A  0DC0                                   678          BASR  R12,0         Set top of loop
                                                 679 *
0000071C  6880 3000                  00000000    680          LD    FPR8,0(,R3)   Get extended BFP test value part 1
00000720  6820 3008                  00000008    681          LD    R2,8(,R3)     Get extended BFP test value part 2
                                                 682 *
                                                 683 * Test cases using rounding mode specified in the FPCR
                                                 684 *
00000724  B29D F2F4                  000002F4    685          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000728  B2B8 0001                  00000001    686          SRNMB 1             SET FPC to RZ, towards zero.
0000072C  B39A 0418                              687          CFXBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000730  5010 7000                  00000000    688          ST    R1,0*4(,R7)   Store integer-32 result
00000734  B29C 8000                  00000000    689          STFPC 0(R8)         Store resulting FPC flags and DXC
00000738  B222 0010                              690          IPM   R1            Get condition code and program mask
0000073C  8810 001C                  0000001C    691          SRL   R1,28         Isolate CC in low order byte
00000740  4210 8003                  00000003    692          STC   R1,3(,R8)     Save CC as low byte of FPCR
                                                 693 *
00000744  B29D F2F4                  000002F4    694          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000748  B2B8 0002                  00000002    695          SRNMB 2             SET FPC to RP, to +infinity
0000074C  B39A 0418                              696          CFXBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000750  5010 7004                  00000004    697          ST    R1,1*4(,R7)   Store integer-32 result
00000754  B29C 8004                  00000004    698          STFPC 1*4(R8)       Store resulting FPC flags and DXC
00000758  B222 0010                              699          IPM   R1            Get condition code and program mask
0000075C  8810 001C                  0000001C    700          SRL   R1,28         Isolate CC in low order byte
00000760  4210 8007                  00000007    701          STC   R1,(1*4)+3(,R8)    Save CC as low byte of FPCR
                                                 702 *
00000764  B29D F2F4                  000002F4    703          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000768  B2B8 0003                  00000003    704          SRNMB 3             SET FPC to RM, to -infinity
0000076C  B39A 0418                              705          CFXBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000770  5010 7008                  00000008    706          ST    R1,2*4(,R7)   Store integer-32 result
00000774  B29C 8008                  00000008    707          STFPC 2*4(R8)       Store resulting FPC flags and DXC
00000778  B222 0010                              708          IPM   R1            Get condition code and program mask
0000077C  8810 001C                  0000001C    709          SRL   R1,28         Isolate CC in low order byte
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000780  4210 800B                     0000000B   710          STC   R1,(2*4)+3(,R8)    Save CC as low byte of FPCR
                                                   711 *
00000784  B29D F2F4                     000002F4   712          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000788  B2B8 0007                     00000007   713          SRNMB 7             RFS, Prepare for Shorter Precision
0000078C  B39A 0418                                714          CFXBRA R1,0,FPR8,B'0100'  FPC ctl'd rounding inexact masked
00000790  5010 700C                     0000000C   715          ST    R1,3*4(,R7)   Store integer-32 result
00000794  B29C 800C                     0000000C   716          STFPC 3*4(R8)       Store resulting FPC flags and DXC
00000798  B222 0010                                717          IPM   R1            Get condition code and program mask
0000079C  8810 001C                     0000001C   718          SRL   R1,28         Isolate CC in low order byte
000007A0  4210 800F                     0000000F   719          STC   R1,(3*4)+3(,R8)    Save CC as low byte of FPCR
                                                   720 *
                                                   721 * Test cases using rounding mode specified in the instruction M3 field
                                                   722 *
000007A4  B29D F2F4                     000002F4   723          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000007A8  B39A 1018                                724          CFXBRA R1,1,FPR8,B'0000'  RNTA, to nearest, ties away
000007AC  5010 7010                     00000010   725          ST    R1,4*4(,R7)   Store integer-32 result
000007B0  B29C 8010                     00000010   726          STFPC 4*4(R8)       Store resulting FPC flags and DXC
000007B4  B222 0010                                727          IPM   R1            Get condition code and program mask
000007B8  8810 001C                     0000001C   728          SRL   R1,28         Isolate CC in low order byte
000007BC  4210 8013                     00000013   729          STC   R1,(4*4)+3(,R8)    Save CC as low byte of FPCR
                                                   730 *
000007C0  B29D F2F4                     000002F4   731          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000007C4  B39A 3018                                732          CFXBRA R1,3,FPR8,B'0000'  RFS, prepare for shorter precision
000007C8  5010 7014                     00000014   733          ST    R1,5*4(,R7)   Store integer-32 result
000007CC  B29C 8014                     00000014   734          STFPC 5*4(R8)       Store resulting FPC flags and DXC
000007D0  B222 0010                                735          IPM   R1            Get condition code and program mask
000007D4  8810 001C                     0000001C   736          SRL   R1,28         Isolate CC in low order byte
000007D8  4210 8017                     00000017   737          STC   R1,(5*4)+3(,R8)    Save CC as low byte of FPCR
                                                   738 *
000007DC  B29D F2F4                     000002F4   739          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000007E0  B39A 4018                                740          CFXBRA R1,4,FPR8,B'0000'  RNTE, to nearest, ties to even
000007E4  5010 7018                     00000018   741          ST    R1,6*4(,R7)   Store integer-32 result
000007E8  B29C 8018                     00000018   742          STFPC 6*4(R8)       Store resulting FPC flags and DXC
000007EC  B222 0010                                743          IPM   R1            Get condition code and program mask
000007F0  8810 001C                     0000001C   744          SRL   R1,28         Isolate CC in low order byte
000007F4  4210 801B                     0000001B   745          STC   R1,(6*4)+3(,R8)    Save CC as low byte of FPCR
                                                   746 *
000007F8  B29D F2F4                     000002F4   747          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
000007FC  B39A 5018                                748          CFXBRA R1,5,FPR8,B'0000'  RZ, toward zero
00000800  5010 701C                     0000001C   749          ST    R1,7*4(,R7)   Store integer-32 result
00000804  B29C 801C                     0000001C   750          STFPC 7*4(R8)       Store resulting FPC flags and DXC
00000808  B222 0010                                751          IPM   R1            Get condition code and program mask
0000080C  8810 001C                     0000001C   752          SRL   R1,28         Isolate CC in low order byte
00000810  4210 801F                     0000001F   753          STC   R1,(7*4)+3(,R8)    Save CC as low byte of FPCR
                                                   754 *
00000814  B29D F2F4                     000002F4   755          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000818  B39A 6018                                756          CFXBRA R1,6,FPR8,B'0000'  RP, to +inf
0000081C  5010 7020                     00000020   757          ST    R1,8*4(,R7)   Store integer-32 result
00000820  B29C 8020                     00000020   758          STFPC 8*4(R8)       Store resulting FPC flags and DXC
00000824  B222 0010                                759          IPM   R1            Get condition code and program mask
00000828  8810 001C                     0000001C   760          SRL   R1,28         Isolate CC in low order byte
0000082C  4210 8023                     00000023   761          STC   R1,(8*4)+3(,R8)    Save CC as low byte of FPCR
                                                   762 *
00000830  B29D F2F4                     000002F4   763          LFPC  FPCREGNT      Set exceptions non-trappable, clear flags
00000834  B39A 7018                                764          CFXBRA R1,7,FPR8,B'0000'  RM, to -inf
00000838  5010 7024                     00000024   765          ST    R1,9*4(,R7)   Store integer-32 result
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

0000083C  B29C 8024                     00000024   766          STFPC 9*4(R8)       Store resulting FPC flags and DXC
00000840  B222 0010                                767          IPM   R1            Get condition code and program mask
00000844  8810 001C                     0000001C   768          SRL   R1,28         Isolate CC in low order byte
00000848  4210 8027                     00000027   769          STC   R1,(9*4)+3(,R8)    Save CC as low byte of FPCR
                                                   770 *
0000084C  4130 3010                     00000010   771          LA    R3,16(,R3)    Point to next input value
00000850  4170 7030                     00000030   772          LA    R7,12*4(,R7)  Point to next int-32 converted value set
00000854  4180 8030                     00000030   773          LA    R8,12*4(,R8)  Point to next FPCR/CC result area
00000858  062C                                     774          BCTR  R2,R12        Convert next input value.
0000085A  07FD                                     775          BR    R13           All converted; return.
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   777 ********************************************************************
                                                   778 *
                                                   779 * Floating point inputs for Convert To Fixed testing.  The same test
                                                   780 * values in the appropriate input format are used for short, long,
                                                   781 * and extended format tests.  The last four values should generate
                                                   782 * exceptions.
                                                   783 *
                                                   784 ********************************************************************


                                                   786 *
                                                   787 * Inputs for basic tests of short BFP to int-32
                                                   788 *
0000085C                                           789 SBFPIN   DS    0F                Inputs for short BFP testing
0000085C  3F800000                                 790          DC    X'3F800000'   +1.0
00000860  40000000                                 791          DC    X'40000000'   +2.0
00000864  40800000                                 792          DC    X'40800000'   +4.0
00000868  C0000000                                 793          DC    X'C0000000'   -2.0
0000086C  7F810000                                 794          DC    X'7F810000'   SNaN
00000870  7FC10000                                 795          DC    X'7FC10000'   QNaN
                                                   796 * The following two will overflow int-32 regardless of rounding mode
00000874  4F000000                                 797          DC    X'4F000000'   +max int-32 + 1.  (2,147,483,647 + 1)
00000878  CF000001                                 798          DC    X'CF000001'   -max int-32 - 2.  (-2,147,483,647 - 2)
0000087C  4EFFFFFF                                 799          DC    X'4EFFFFFF'   Largest short bfp that fits in int-32
                                                   800 *                            ..2,147,483,520 = 0x7FFFFF80
                                                   801 *
                                   00000024  00000001  802 SBFPCT   EQU   *-SBFPIN         Count of short BFP in list * 4
                                                   803 *
                                                   804 * Inputs for exhaustive rounding mode tests of short BFP to int-32
                                                   805 *
00000880                                           806 SBFPINRM DS    0F
00000880  C1180000                                 807          DC    X'C1180000'         -9.5
00000884  C0B00000                                 808          DC    X'C0B00000'         -5.5
00000888  C0200000                                 809          DC    X'C0200000'         -2.5
0000088C  BFC00000                                 810          DC    X'BFC00000'         -1.5
00000890  BF000000                                 811          DC    X'BF000000'         -0.5
00000894  3F000000                                 812          DC    X'3F000000'         +0.5
00000898  3FC00000                                 813          DC    X'3FC00000'         +1.5
0000089C  40200000                                 814          DC    X'40200000'         +2.5
000008A0  40B00000                                 815          DC    X'40B00000'         +5.5
000008A4  41180000                                 816          DC    X'41180000'         +9.5
000008A8  3F400000                                 817          DC    X'3F400000'         +0.75
000008AC  3E800000                                 818          DC    X'3E800000'         +0.25
000008B0  BF400000                                 819          DC    X'BF400000'         -0.75
000008B4  BE800000                                 820          DC    X'BE800000'         -0.25
                                                   821 *
                                                   822 * There is no short BFP represtation for values between 2,147,483,520
                                                   823 * and 2,147,483,648, making it difficult to come up with a test case
                                                   824 * that overflows for only some of the rounding modes available.
                                                   825 *
                                   00000038  00000001  826 SBFPRMCT EQU   *-SBFPINRM   Count of short BFP in list * 4
                                                   827 *
                                                   828 * Inputs for basic tests of long BFP to int-32
                                                   829 *
000008B8                                           830 LBFPIN   DS    0F                Inputs for long BFP testing
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000008B8  3FF00000 00000000                        831          DC    X'3FF0000000000000'    +1.0
000008C0  40000000 00000000                        832          DC    X'4000000000000000'    +2.0
000008C8  40100000 00000000                        833          DC    X'4010000000000000'    +4.0
000008D0  C0000000 00000000                        834          DC    X'C000000000000000'    -2.0
000008D8  7FF01000 00000000                        835          DC    X'7FF0100000000000'    SNaN
000008E0  7FF81000 00000000                        836          DC    X'7FF8100000000000'    QNaN
000008E8  41E00000 00000000                        837          DC    X'41E0000000000000'   +max int-32 + 1 (+2147483647 + 1)
000008F0  C1E00000 00200000                        838          DC    X'C1E0000000200000'   -max int-32 - 2 (-2147483647 - 2)
000008F8  41DFFFFF FFC00000                        839          DC    X'41DFFFFFFFC00000'   Largest long bfp that fits in
                                                   840 *                           ..int-32: 2,147,483,647 = 0x7FFFFFFF
00000900  41DFFFFF FFE00000                        841          DC    X'41DFFFFFFFE00000'   2,147,483,647.5 - overflows on
                                                   842 *                                    RNTE; test of traps
                               00000050  00000001  843 LBFPCT   EQU   *-LBFPIN     Count of long BFP in list * 8
                                                   844 *
                                                   845 * Inputs for exhaustive rounding mode tests of long BFP to int-32
                                                   846 *
00000908                                           847 LBFPINRM DS    0F
00000908  C0230000 00000000                        848          DC    X'C023000000000000'        -9.5
00000910  C0160000 00000000                        849          DC    X'C016000000000000'        -5.5
00000918  C0040000 00000000                        850          DC    X'C004000000000000'        -2.5
00000920  BFF80000 00000000                        851          DC    X'BFF8000000000000'        -1.5
00000928  BFE00000 00000000                        852          DC    X'BFE0000000000000'        -0.5
00000930  3FE00000 00000000                        853          DC    X'3FE0000000000000'        +0.5
00000938  3FF80000 00000000                        854          DC    X'3FF8000000000000'        +1.5
00000940  40040000 00000000                        855          DC    X'4004000000000000'        +2.5
00000948  40160000 00000000                        856          DC    X'4016000000000000'        +5.5
00000950  40230000 00000000                        857          DC    X'4023000000000000'        +9.5
00000958  3FE80000 00000000                        858          DC    X'3FE8000000000000'        +0.75
00000960  3FD00000 00000000                        859          DC    X'3FD0000000000000'        +0.25
00000968  BFE80000 00000000                        860          DC    X'BFE8000000000000'        -0.75
00000970  BFD00000 00000000                        861          DC    X'BFD0000000000000'        -0.25
00000978  41DFFFFF FFE00000                        862          DC    X'41DFFFFFFFE00000'   2,147,483,647.5 - overflows on
                                                   863 *                                    some but not all rounding modes
                               00000078  00000001  864 LBFPRMCT EQU   *-LBFPINRM   Count of long BFP in list * 8
                                                   865 *
                                                   866 * Inputs for basic tests of extended BFP to int-32
                                                   867 *
00000980                                           868 XBFPIN   DS    0D                Inputs for extended BFP testing
00000980  3FFF0000 00000000                        869          DC    X'3FFF0000000000000000000000000000'    +1.0
00000990  40000000 00000000                        870          DC    X'40000000000000000000000000000000'    +2.0
000009A0  40010000 00000000                        871          DC    X'40010000000000000000000000000000'    +4.0
000009B0  C0000000 00000000                        872          DC    X'C0000000000000000000000000000000'    -2.0
000009C0  7FFF0100 00000000                        873          DC    X'7FFF0100000000000000000000000000'    SNaN
000009D0  7FFF8100 00000000                        874          DC    X'7FFF8100000000000000000000000000'    QNaN
000009E0  401E0000 00000000                        875          DC    X'401E0000000000000000000000000000'   +max int-32 + 1
000009F0  C01E0000 00020000                        876          DC    X'C01E0000000200000000000000000000'   -max int-32 - 2
00000A00  401DFFFF FFFC0000                        877          DC    X'401DFFFFFFFC00000000000000000000'   Largest long bfp
                                                   878 *                 that fits in int-32: 2,147,483,647 = 0x7FFFFFFF
00000A10  401DFFFF FFFE0000                        879          DC    X'401DFFFFFFFE00000000000000000000'   2,147,483,647.5
                                                   880 *                     - overflows on RNTE; test of traps
                               000000A0  00000001  881 XBFPCT   EQU   *-XBFPIN     Count of extended BFP in list * 16
                                                   882 *
                                                   883 * Inputs for exhaustive rounding mode tests of long BFP to int-32
                                                   884 *
00000A20                                           885 XBFPINRM DS    0D
00000A20  C0023000 00000000                        886          DC    X'C0023000000000000000000000000000'         -9.5
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000A30  C0016000 00000000                        887          DC    X'C0016000000000000000000000000000'          -5.5
00000A40  C0004000 00000000                        888          DC    X'C0004000000000000000000000000000'          -2.5
00000A50  BFFF8000 00000000                        889          DC    X'BFFF8000000000000000000000000000'          -1.5
00000A60  BFFE0000 00000000                        890          DC    X'BFFE0000000000000000000000000000'          -0.5
00000A70  3FFE0000 00000000                        891          DC    X'3FFE0000000000000000000000000000'          +0.5
00000A80  3FFF8000 00000000                        892          DC    X'3FFF8000000000000000000000000000'          +1.5
00000A90  40004000 00000000                        893          DC    X'40004000000000000000000000000000'          +2.5
00000AA0  40016000 00000000                        894          DC    X'40016000000000000000000000000000'          +5.5
00000AB0  40023000 00000000                        895          DC    X'40023000000000000000000000000000'          +9.5
00000AC0  3FFE8000 00000000                        896          DC    X'3FFE8000000000000000000000000000'          +0.75
00000AD0  3FFD0000 00000000                        897          DC    X'3FFD0000000000000000000000000000'          +0.25
00000AE0  BFFE8000 00000000                        898          DC    X'BFFE8000000000000000000000000000'          -0.75
00000AF0  BFFD0000 00000000                        899          DC    X'BFFD0000000000000000000000000000'          -0.25
00000B00  401DFFFF FFFE0000                        900          DC    X'401DFFFFFFFE00000000000000000000'   2,147,483,647.5
                                                   901 *                      - overflows on some but not all rounding modes
                              000000F0  00000001   902 XBFPRMCT EQU   *-XBFPINRM   Count of extended BFP in list * 16
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   904 ***********************************************************************
                                                   905 *                 ACTUAL results saved here
                                                   906 ***********************************************************************
                                                   907 *
                                                   908 *               Locations for ACTUAL results
                                                   909 *
                                                   910 *
                              00001000  00000000   911 SINTOUT  EQU   BFPCVTTF+X'1000'    uint-32 values from short BFP
                                                   912 *                                 ..9 pairs used, room for 16
                              00001100  00000000   913 SINTFLGS EQU   BFPCVTTF+X'1100'    FPCR flags and DXC from short BFP
                                                   914 *                                 ..9 pairs used, room for 16
                              00001200  00000000   915 SINTRMO  EQU   BFPCVTTF+X'1200'    Short rounding mode test results
                                                   916 *                                 ..14 sets used, room for 20
                              00001600  00000000   917 SINTRMOF EQU   BFPCVTTF+X'1600'    Short rounding mode FPCR contents
                                                   918 *                                 ..14 sets used, room for 20
                                                   919 *
                              00002000  00000000   920 LINTOUT  EQU   BFPCVTTF+X'2000'    uint-32 values from long BFP
                                                   921 *                                 ..10 pairs used, room for 16
                              00002100  00000000   922 LINTFLGS EQU   BFPCVTTF+X'2100'    FPCR flags and DXC from long BFP
                                                   923 *                                 ..10 pairs used, room for 16
                              00002200  00000000   924 LINTRMO  EQU   BFPCVTTF+X'2200'    Long rounding mode test results
                                                   925 *                                 ..14 sets used, room for 20
                              00002600  00000000   926 LINTRMOF EQU   BFPCVTTF+X'2600'    Long rounding mode FPCR contents
                                                   927 *                                 ..14 sets used, room for 20
                                                   928 *
                              00003000  00000000   929 XINTOUT  EQU   BFPCVTTF+X'3000'    uint-32 values from extended BFP
                                                   930 *                                 ..10 pairs used, room for 16
                              00003100  00000000   931 XINTFLGS EQU   BFPCVTTF+X'3100'    FPCR flags and DXC from extended BFP
                                                   932 *                                 ..10 pairs used, room for 16
                              00003200  00000000   933 XINTRMO  EQU   BFPCVTTF+X'3200'    Extended rounding mode test results
                                                   934 *                                 ..14 sets used, room for 20
                              00003600  00000000   935 XINTRMOF EQU   BFPCVTTF+X'3600'    Extended rounding mode FPCR contents
                                                   936 *                                 ..14 sets used, room for 20
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   938 ********************************************************************
                                                   939 *                         EXPECTED results
                                                   940 ********************************************************************
                                                   941 *
00000B10                     00000B10  00005000    942          ORG   BFPCVTTF+X'5000'   (past end of actual results)
                                                   943 *
                             00005000  00000001    944 SINTOUT_GOOD EQU *
00005000  C3C6C5C2 D9409985                        945  DC CL48'CFEBR result pairs 1-2'
00005030  00000001 00000001                        946  DC XL16'00000001000000010000000200000002'
00005040  C3C6C5C2 D9409985                        947  DC CL48'CFEBR result pairs 3-4'
00005070  00000004 00000004                        948  DC XL16'0000000400000004FFFFFFFEFFFFFFFE'
00005080  C3C6C5C2 D9409985                        949  DC CL48'CFEBR result pairs 5-6'
000050B0  80000000 00000000                        950  DC XL16'80000000000000008000000000000000'
000050C0  C3C6C5C2 D9409985                        951  DC CL48'CFEBR result pairs 7-8'
000050F0  7FFFFFFF 00000000                        952  DC XL16'7FFFFFFF000000008000000000000000'
00005100  C3C6C5C2 D9409985                        953  DC CL48'CFEBR result pair 9'
00005130  7FFFFF80 7FFFFF80                        954  DC XL16'7FFFFF807FFFFF800000000000000000'
                             00000005  00000001    955 SINTOUT_NUM EQU (*-SINTOUT_GOOD)/64
                                                   956 *
                                                   957 *
                             00005140  00000001    958 SINTFLGS_GOOD EQU *
00005140  C3C6C5C2 D940C6D7                        959  DC CL48'CFEBR FPCR pairs 1-2'
00005170  00000002 F8000002                        960  DC XL16'00000002F800000200000002F8000002'
00005180  C3C6C5C2 D940C6D7                        961  DC CL48'CFEBR FPCR pairs 3-4'
000051B0  00000002 F8000002                        962  DC XL16'00000002F800000200000001F8000001'
000051C0  C3C6C5C2 D940C6D7                        963  DC CL48'CFEBR FPCR pairs 5-6'
000051F0  00880003 F8008000                        964  DC XL16'00880003F800800000880003F8008000'
00005200  C3C6C5C2 D940C6D7                        965  DC CL48'CFEBR FPCR pairs 7-8'
00005230  00880003 F8008000                        966  DC XL16'00880003F800800000880003F8008000'
00005240  C3C6C5C2 D940C6D7                        967  DC CL48'CFEBR FPCR pair 9'
00005270  00000002 F8000002                        968  DC XL16'00000002F80000020000000000000000'
                             00000005  00000001    969 SINTFLGS_NUM EQU (*-SINTFLGS_GOOD)/64
                                                   970 *
                                                   971 *
                             00005280  00000001    972 SINTRMO_GOOD EQU *
00005280  C3C6C5C2 D9C14060                        973  DC CL48'CFEBRA -9.5 FPCR modes 1-3, 7'
000052B0  FFFFFFF7 FFFFFFF7                        974  DC XL16'FFFFFFF7FFFFFFF7FFFFFFF6FFFFFFF7'
000052C0  C3C6C5C2 D9C14060                        975  DC CL48'CFEBRA -9.5 M3 modes 1, 3-5'
000052F0  FFFFFFF6 FFFFFFF7                        976  DC XL16'FFFFFFF6FFFFFFF7FFFFFFF6FFFFFFF7'
00005300  C3C6C5C2 D9C14060                        977  DC CL48'CFEBRA -9.5 M3 modes 6, 7'
00005330  FFFFFFF7 FFFFFFF6                        978  DC XL16'FFFFFFF7FFFFFFF60000000000000000'
00005340  C3C6C5C2 D9C14060                        979  DC CL48'CFEBRA -5.5 FPCR modes 1-3, 7'
00005370  FFFFFFFB FFFFFFFB                        980  DC XL16'FFFFFFFBFFFFFFFBFFFFFFFAFFFFFFFB'
00005380  C3C6C5C2 D9C14060                        981  DC CL48'CFEBRA -5.5 M3 modes 1, 3-5'
000053B0  FFFFFFFA FFFFFFFB                        982  DC XL16'FFFFFFFAFFFFFFFBFFFFFFFAFFFFFFFB'
000053C0  C3C6C5C2 D9C14060                        983  DC CL48'CFEBRA -5.5 M3 modes 6, 7'
000053F0  FFFFFFFB FFFFFFFA                        984  DC XL16'FFFFFFFBFFFFFFFA0000000000000000'
00005400  C3C6C5C2 D9C14060                        985  DC CL48'CFEBRA -2.5 FPCR modes 1-3, 7'
00005430  FFFFFFFE FFFFFFFE                        986  DC XL16'FFFFFFFEFFFFFFFEFFFFFFFDFFFFFFFD'
00005440  C3C6C5C2 D9C14060                        987  DC CL48'CFEBRA -2.5 M3 modes 1, 3-5'
00005470  FFFFFFFD FFFFFFFD                        988  DC XL16'FFFFFFFDFFFFFFFDFFFFFFFEFFFFFFFE'
00005480  C3C6C5C2 D9C14060                        989  DC CL48'CFEBRA -2.5 M3 modes 6, 7'
000054B0  FFFFFFFE FFFFFFFD                        990  DC XL16'FFFFFFFEFFFFFFFD0000000000000000'
000054C0  C3C6C5C2 D9C14060                        991  DC CL48'CFEBRA -1.5 FPCR modes 1-3, 7'
000054F0  FFFFFFFF FFFFFFFF                        992  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFF'
00005500  C3C6C5C2 D9C14060                        993  DC CL48'CFEBRA -1.5 M3 modes 1, 3-5'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00005530  FFFFFFFE FFFFFFFF                        994  DC XL16'FFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFF'
00005540  C3C6C5C2 D9C14060                        995  DC CL48'CFEBRA -1.5 M3 modes 6, 7'
00005570  FFFFFFFF FFFFFFFE                        996  DC XL16'FFFFFFFFFFFFFFFE0000000000000000'
00005580  C3C6C5C2 D9C14060                        997  DC CL48'CFEBRA -0.5 FPCR modes 1-3, 7'
000055B0  00000000 00000000                        998  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
000055C0  C3C6C5C2 D9C14060                        999  DC CL48'CFEBRA -0.5 M3 modes 1, 3-5'
000055F0  FFFFFFFF FFFFFFFF                       1000  DC XL16'FFFFFFFFFFFFFFFF0000000000000000'
00005600  C3C6C5C2 D9C14060                       1001  DC CL48'CFEBRA -0.5 M3 modes 6, 7'
00005630  00000000 FFFFFFFF                       1002  DC XL16'00000000FFFFFFFF0000000000000000'
00005640  C3C6C5C2 D9C1404E                       1003  DC CL48'CFEBRA +0.5 FPCR modes 1-3, 7'
00005670  00000000 00000001                       1004  DC XL16'00000000000000010000000000000001'
00005680  C3C6C5C2 D9C1404E                       1005  DC CL48'CFEBRA +0.5 M3 modes 1, 3-5'
000056B0  00000001 00000001                       1006  DC XL16'00000001000000010000000000000000'
000056C0  C3C6C5C2 D9C1404E                       1007  DC CL48'CFEBRA +0.5 M3 modes 6, 7'
000056F0  00000001 00000000                       1008  DC XL16'00000001000000000000000000000000'
00005700  C3C6C5C2 D9C1404E                       1009  DC CL48'CFEBRA +1.5 FPCR modes 1-3, 7'
00005730  00000001 00000002                       1010  DC XL16'00000001000000020000000100000001'
00005740  C3C6C5C2 D9C1404E                       1011  DC CL48'CFEBRA +1.5 M3 modes 1, 3-5'
00005770  00000002 00000001                       1012  DC XL16'00000002000000010000000200000001'
00005780  C3C6C5C2 D9C1404E                       1013  DC CL48'CFEBRA +1.5 M3 modes 6, 7'
000057B0  00000002 00000001                       1014  DC XL16'00000002000000010000000000000000'
000057C0  C3C6C5C2 D9C1404E                       1015  DC CL48'CFEBRA +2.5 FPCR modes 1-3, 7'
000057F0  00000002 00000003                       1016  DC XL16'00000002000000030000000200000003'
00005800  C3C6C5C2 D9C1404E                       1017  DC CL48'CFEBRA +2.5 M3 modes 1, 3-5'
00005830  00000003 00000003                       1018  DC XL16'00000003000000030000000200000002'
00005840  C3C6C5C2 D9C1404E                       1019  DC CL48'CFEBRA +2.5 M3 modes 6, 7'
00005870  00000003 00000002                       1020  DC XL16'00000003000000020000000000000000'
00005880  C3C6C5C2 D9C1404E                       1021  DC CL48'CFEBRA +5.5 FPCR modes 1-3, 7'
000058B0  00000005 00000006                       1022  DC XL16'00000005000000060000000500000005'
000058C0  C3C6C5C2 D9C1404E                       1023  DC CL48'CFEBRA +5.5 M3 modes 1, 3-5'
000058F0  00000006 00000005                       1024  DC XL16'00000006000000050000000600000005'
00005900  C3C6C5C2 D9C1404E                       1025  DC CL48'CFEBRA +5.5 M3 modes 6, 7'
00005930  00000006 00000005                       1026  DC XL16'00000006000000050000000000000000'
00005940  C3C6C5C2 D9C1404E                       1027  DC CL48'CFEBRA +9.5 FPCR modes 1-3, 7'
00005970  00000009 0000000A                       1028  DC XL16'000000090000000A0000000900000009'
00005980  C3C6C5C2 D9C1404E                       1029  DC CL48'CFEBRA +9.5 M3 modes 1, 3-5'
000059B0  0000000A 00000009                       1030  DC XL16'0000000A000000090000000A00000009'
000059C0  C3C6C5C2 D9C1404E                       1031  DC CL48'CFEBRA +9.5 M3 modes 6, 7'
000059F0  0000000A 00000009                       1032  DC XL16'0000000A000000090000000000000000'
00005A00  C3C6C5C2 D9C1404E                       1033  DC CL48'CFEBRA +0.75 FPCR modes 1-3, 7'
00005A30  00000000 00000001                       1034  DC XL16'00000000000000010000000000000001'
00005A40  C3C6C5C2 D9C1404E                       1035  DC CL48'CFEBRA +0.75 M3 modes 1, 3-5'
00005A70  00000001 00000001                       1036  DC XL16'00000001000000010000000100000000'
00005A80  C3C6C5C2 D9C1404E                       1037  DC CL48'CFEBRA +0.75 M3 modes 6, 7'
00005AB0  00000001 00000000                       1038  DC XL16'00000001000000000000000000000000'
00005AC0  C3C6C5C2 D9C1404E                       1039  DC CL48'CFEBRA +0.25 FPCR modes 1-3, 7'
00005AF0  00000000 00000001                       1040  DC XL16'00000000000000010000000000000001'
00005B00  C3C6C5C2 D9C1404E                       1041  DC CL48'CFEBRA +0.25 M3 modes 1, 3-5'
00005B30  00000000 00000001                       1042  DC XL16'00000000000000010000000000000000'
00005B40  C3C6C5C2 D9C1404E                       1043  DC CL48'CFEBRA +0.25 M3 modes 6, 7'
00005B70  00000001 00000000                       1044  DC XL16'00000001000000000000000000000000'
00005B80  C3C6C5C2 D9C14060                       1045  DC CL48'CFEBRA -0.75 FPCR modes 1-3, 7'
00005BB0  00000000 00000000                       1046  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00005BC0  C3C6C5C2 D9C14060                       1047  DC CL48'CFEBRA -0.75 M3 modes 1, 3-5'
00005BF0  FFFFFFFF FFFFFFFF                       1048  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFF00000000'
00005C00  C3C6C5C2 D9C14060                       1049  DC CL48'CFEBRA -0.75 M3 modes 6, 7'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00005C30  00000000 FFFFFFFF                       1050  DC XL16'00000000FFFFFFFF0000000000000000'
00005C40  C3C6C5C2 D9C14060                       1051  DC CL48'CFEBRA -0.25 FPCR modes 1-3, 7'
00005C70  00000000 00000000                       1052  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00005C80  C3C6C5C2 D9C14060                       1053  DC CL48'CFEBRA -0.25 M3 modes 1, 3-5'
00005CB0  00000000 FFFFFFFF                       1054  DC XL16'00000000FFFFFFFF0000000000000000'
00005CC0  C3C6C5C2 D9C14060                       1055  DC CL48'CFEBRA -0.25 M3 modes 6, 7'
00005CF0  00000000 FFFFFFFF                       1056  DC XL16'00000000FFFFFFFF0000000000000000'
                               0000002A  00000001 1057 SINTRMO_NUM EQU (*-SINTRMO_GOOD)/64
                                                  1058 *
                                                  1059 *
                               00005D00  00000001 1060 SINTRMOF_GOOD EQU *
00005D00  C3C6C5C2 D9C14060                       1061  DC CL48'CFEBRA -9.5 FPCR modes 1-3, 7 FPCR'
00005D30  00000001 00000001                       1062  DC XL16'00000001000000010000000100000001'
00005D40  C3C6C5C2 D9C14060                       1063  DC CL48'CFEBRA -9.5 M3 modes 1, 3-5 FPCR'
00005D70  00080001 00080001                       1064  DC XL16'00080001000800010008000100080001'
00005D80  C3C6C5C2 D9C14060                       1065  DC CL48'CFEBRA -9.5 M3 modes 6, 7 FPCR'
00005DB0  00080001 00080001                       1066  DC XL16'00080001000800010000000000000000'
00005DC0  C3C6C5C2 D9C14060                       1067  DC CL48'CFEBRA -5.5 FPCR modes 1-3, 7 FPCR'
00005DF0  00000001 00000001                       1068  DC XL16'00000001000000010000000100000001'
00005E00  C3C6C5C2 D9C14060                       1069  DC CL48'CFEBRA -5.5 M3 modes 1, 3-5 FPCR'
00005E30  00080001 00080001                       1070  DC XL16'00080001000800010008000100080001'
00005E40  C3C6C5C2 D9C14060                       1071  DC CL48'CFEBRA -5.5 M3 modes 6, 7 FPCR'
00005E70  00080001 00080001                       1072  DC XL16'00080001000800010000000000000000'
00005E80  C3C6C5C2 D9C14060                       1073  DC CL48'CFEBRA -2.5 FPCR modes 1-3, 7 FPCR'
00005EB0  00000001 00000001                       1074  DC XL16'00000001000000010000000100000001'
00005EC0  C3C6C5C2 D9C14060                       1075  DC CL48'CFEBRA -2.5 M3 modes 1, 3-5 FPCR'
00005EF0  00080001 00080001                       1076  DC XL16'00080001000800010008000100080001'
00005F00  C3C6C5C2 D9C14060                       1077  DC CL48'CFEBRA -2.5 M3 modes 6, 7 FPCR'
00005F30  00080001 00080001                       1078  DC XL16'00080001000800010000000000000000'
00005F40  C3C6C5C2 D9C14060                       1079  DC CL48'CFEBRA -1.5 FPCR modes 1-3, 7 FPCR'
00005F70  00000001 00000001                       1080  DC XL16'00000001000000010000000100000001'
00005F80  C3C6C5C2 D9C14060                       1081  DC CL48'CFEBRA -1.5 M3 modes 1, 3-5 FPCR'
00005FB0  00080001 00080001                       1082  DC XL16'00080001000800010008000100080001'
00005FC0  C3C6C5C2 D9C14060                       1083  DC CL48'CFEBRA -1.5 M3 modes 6, 7 FPCR'
00005FF0  00080001 00080001                       1084  DC XL16'00080001000800010000000000000000'
00006000  C3C6C5C2 D9C14060                       1085  DC CL48'CFEBRA -0.5 FPCR modes 1-3, 7 FPCR'
00006030  00000001 00000001                       1086  DC XL16'00000001000000010000000100000001'
00006040  C3C6C5C2 D9C14060                       1087  DC CL48'CFEBRA -0.5 M3 modes 1, 3-5 FPCR'
00006070  00080001 00080001                       1088  DC XL16'00080001000800010008000100080001'
00006080  C3C6C5C2 D9C14060                       1089  DC CL48'CFEBRA -0.5 M3 modes 6, 7 FPCR'
000060B0  00080001 00080001                       1090  DC XL16'00080001000800010000000000000000'
000060C0  C3C6C5C2 D9C1404E                       1091  DC CL48'CFEBRA +0.5 FPCR modes 1-3, 7 FPCR'
000060F0  00000002 00000002                       1092  DC XL16'00000002000000020000000200000002'
00006100  C3C6C5C2 D9C1404E                       1093  DC CL48'CFEBRA +0.5 M3 modes 1, 3-5 FPCR'
00006130  00080002 00080002                       1094  DC XL16'00080002000800020008000200080002'
00006140  C3C6C5C2 D9C1404E                       1095  DC CL48'CFEBRA +0.5 M3 modes 6, 7 FPCR'
00006170  00080002 00080002                       1096  DC XL16'00080002000800020000000000000000'
00006180  C3C6C5C2 D9C1404E                       1097  DC CL48'CFEBRA +1.5 FPCR modes 1-3, 7 FPCR'
000061B0  00000002 00000002                       1098  DC XL16'00000002000000020000000200000002'
000061C0  C3C6C5C2 D9C1404E                       1099  DC CL48'CFEBRA +1.5 M3 modes 1, 3-5 FPCR'
000061F0  00080002 00080002                       1100  DC XL16'00080002000800020008000200080002'
00006200  C3C6C5C2 D9C1404E                       1101  DC CL48'CFEBRA +1.5 M3 modes 6, 7 FPCR'
00006230  00080002 00080002                       1102  DC XL16'00080002000800020000000000000000'
00006240  C3C6C5C2 D9C1404E                       1103  DC CL48'CFEBRA +2.5 FPCR modes 1-3, 7 FPCR'
00006270  00000002 00000002                       1104  DC XL16'00000002000000020000000200000002'
00006280  C3C6C5C2 D9C1404E                       1105  DC CL48'CFEBRA +2.5 M3 modes 1, 3-5 FPCR'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000062B0  00080002 00080002                       1106  DC XL16'00080002000800020008000200080002'
000062C0  C3C6C5C2 D9C1404E                       1107  DC CL48'CFEBRA +2.5 M3 modes 6, 7 FPCR'
000062F0  00080002 00080002                       1108  DC XL16'00080002000800020000000000000000'
00006300  C3C6C5C2 D9C1404E                       1109  DC CL48'CFEBRA +5.5 FPCR modes 1-3, 7 FPCR'
00006330  00000002 00000002                       1110  DC XL16'00000002000000020000000200000002'
00006340  C3C6C5C2 D9C1404E                       1111  DC CL48'CFEBRA +5.5 M3 modes 1, 3-5 FPCR'
00006370  00080002 00080002                       1112  DC XL16'00080002000800020008000200080002'
00006380  C3C6C5C2 D9C1404E                       1113  DC CL48'CFEBRA +5.5 M3 modes 6, 7 FPCR'
000063B0  00080002 00080002                       1114  DC XL16'00080002000800020000000000000000'
000063C0  C3C6C5C2 D9C1404E                       1115  DC CL48'CFEBRA +9.5 FPCR modes 1-3, 7 FPCR'
000063F0  00000002 00000002                       1116  DC XL16'00000002000000020000000200000002'
00006400  C3C6C5C2 D9C1404E                       1117  DC CL48'CFEBRA +9.5 M3 modes 1, 3-5 FPCR'
00006430  00080002 00080002                       1118  DC XL16'00080002000800020008000200080002'
00006440  C3C6C5C2 D9C1404E                       1119  DC CL48'CFEBRA +9.5 M3 modes 6, 7 FPCR'
00006470  00080002 00080002                       1120  DC XL16'00080002000800020000000000000000'
00006480  C3C6C5C2 D9C1404E                       1121  DC CL48'CFEBRA +0.75 FPCR modes 1-3, 7 FPCR'
000064B0  00000002 00000002                       1122  DC XL16'00000002000000020000000200000002'
000064C0  C3C6C5C2 D9C1404E                       1123  DC CL48'CFEBRA +0.75 M3 modes 1, 3-5 FPCR'
000064F0  00080002 00080002                       1124  DC XL16'00080002000800020008000200080002'
00006500  C3C6C5C2 D9C1404E                       1125  DC CL48'CFEBRA +0.75 M3 modes 6, 7 FPCR'
00006530  00080002 00080002                       1126  DC XL16'00080002000800020000000000000000'
00006540  C3C6C5C2 D9C1404E                       1127  DC CL48'CFEBRA +0.25 FPCR modes 1-3, 7 FPCR'
00006570  00000002 00000002                       1128  DC XL16'00000002000000020000000200000002'
00006580  C3C6C5C2 D9C1404E                       1129  DC CL48'CFEBRA +0.25 M3 modes 1, 3-5 FPCR'
000065B0  00080002 00080002                       1130  DC XL16'00080002000800020008000200080002'
000065C0  C3C6C5C2 D9C1404E                       1131  DC CL48'CFEBRA +0.25 M3 modes 6, 7 FPCR'
000065F0  00080002 00080002                       1132  DC XL16'00080002000800020000000000000000'
00006600  C3C6C5C2 D9C14060                       1133  DC CL48'CFEBRA -0.75 FPCR modes 1-3, 7 FPCR'
00006630  00000001 00000001                       1134  DC XL16'00000001000000010000000100000001'
00006640  C3C6C5C2 D9C14060                       1135  DC CL48'CFEBRA -0.75 M3 modes 1, 3-5 FPCR'
00006670  00080001 00080001                       1136  DC XL16'00080001000800010008000100080001'
00006680  C3C6C5C2 D9C14060                       1137  DC CL48'CFEBRA -0.75 M3 modes 6, 7 FPCR'
000066B0  00080001 00080001                       1138  DC XL16'00080001000800010000000000000000'
000066C0  C3C6C5C2 D9C14060                       1139  DC CL48'CFEBRA -0.25 FPCR modes 1-3, 7 FPCR'
000066F0  00000001 00000001                       1140  DC XL16'00000001000000010000000100000001'
00006700  C3C6C5C2 D9C14060                       1141  DC CL48'CFEBRA -0.25 M3 modes 1, 3-5 FPCR'
00006730  00080001 00080001                       1142  DC XL16'00080001000800010008000100080001'
00006740  C3C6C5C2 D9C14060                       1143  DC CL48'CFEBRA -0.25 M3 modes 6, 7 FPCR'
00006770  00080001 00080001                       1144  DC XL16'00080001000800010000000000000000'
                                 0000002A  00000001  1145 SINTRMOF_NUM EQU (*-SINTRMOF_GOOD)/64
                                                  1146 *
                                                  1147 *
                                 00006780  00000001  1148 LINTOUT_GOOD EQU *
00006780  C3C6C4C2 D9409985                       1149  DC CL48'CFDBR result pairs 1-2'
000067B0  00000001 00000001                       1150  DC XL16'00000001000000010000000200000002'
000067C0  C3C6C4C2 D9409985                       1151  DC CL48'CFDBR result pairs 3-4'
000067F0  00000004 00000004                       1152  DC XL16'0000000400000004FFFFFFFEFFFFFFFE'
00006800  C3C6C4C2 D9409985                       1153  DC CL48'CFDBR result pairs 5-6'
00006830  80000000 00000000                       1154  DC XL16'80000000000000008000000000000000'
00006840  C3C6C4C2 D9409985                       1155  DC CL48'CFDBR result pairs 7-8'
00006870  7FFFFFFF 00000000                       1156  DC XL16'7FFFFFFF000000008000000000000000'
00006880  C3C6C4C2 D9409985                       1157  DC CL48'CFDBR result pairs 9-10'
000068B0  7FFFFFFF 7FFFFFFF                       1158  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF00000000'
                                 00000005  00000001  1159 LINTOUT_NUM EQU (*-LINTOUT_GOOD)/64
                                                  1160 *
                                                  1161 *
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                              000068C0  00000001  1162 LINTFLGS_GOOD EQU *
000068C0  C3C6C4C2 D940C6D7                       1163  DC CL48'CFDBR FPCR pairs 1-2'
000068F0  00000002 F8000002                       1164  DC XL16'00000002F800000200000002F8000002'
00006900  C3C6C4C2 D940C6D7                       1165  DC CL48'CFDBR FPCR pairs 3-4'
00006930  00000002 F8000002                       1166  DC XL16'00000002F800000200000001F8000001'
00006940  C3C6C4C2 D940C6D7                       1167  DC CL48'CFDBR FPCR pairs 5-6'
00006970  00880003 F8008000                       1168  DC XL16'00880003F800800000880003F8008000'
00006980  C3C6C4C2 D940C6D7                       1169  DC CL48'CFDBR FPCR pairs 7-8'
000069B0  00880003 F8008000                       1170  DC XL16'00880003F800800000880003F8008000'
000069C0  C3C6C4C2 D940C6D7                       1171  DC CL48'CFDBR FPCR pairs 9-10'
000069F0  00000002 F8000002                       1172  DC XL16'00000002F800000200880003F8008000'
                              00000005  00000001  1173 LINTFLGS_NUM EQU (*-LINTFLGS_GOOD)/64
                                                  1174 *
                                                  1175 *
                              00006A00  00000001  1176 LINTRMO_GOOD EQU *
00006A00  C3C6C4C2 D9C14060                       1177  DC CL48'CFDBRA -9.5 FPCR modes 1-3, 7'
00006A30  FFFFFFF7 FFFFFFF7                       1178  DC XL16'FFFFFFF7FFFFFFF7FFFFFFF6FFFFFFF7'
00006A40  C3C6C4C2 D9C14060                       1179  DC CL48'CFDBRA -9.5 M3 modes 1, 3-5'
00006A70  FFFFFFF6 FFFFFFF7                       1180  DC XL16'FFFFFFF6FFFFFFF7FFFFFFF6FFFFFFF7'
00006A80  C3C6C4C2 D9C14060                       1181  DC CL48'CFDBRA -9.5 M3 modes 6, 7'
00006AB0  FFFFFFF7 FFFFFFF6                       1182  DC XL16'FFFFFFF7FFFFFFF60000000000000000'
00006AC0  C3C6C4C2 D9C14060                       1183  DC CL48'CFDBRA -5.5 FPCR modes 1-3, 7'
00006AF0  FFFFFFFB FFFFFFFB                       1184  DC XL16'FFFFFFFBFFFFFFFBFFFFFFFAFFFFFFFB'
00006B00  C3C6C4C2 D9C14060                       1185  DC CL48'CFDBRA -5.5 M3 modes 1, 3-5'
00006B30  FFFFFFFA FFFFFFFB                       1186  DC XL16'FFFFFFFAFFFFFFFBFFFFFFFAFFFFFFFB'
00006B40  C3C6C4C2 D9C14060                       1187  DC CL48'CFDBRA -5.5 M3 modes 6, 7'
00006B70  FFFFFFFB FFFFFFFA                       1188  DC XL16'FFFFFFFBFFFFFFFA0000000000000000'
00006B80  C3C6C4C2 D9C14060                       1189  DC CL48'CFDBRA -2.5 FPCR modes 1-3, 7'
00006BB0  FFFFFFFE FFFFFFFE                       1190  DC XL16'FFFFFFFEFFFFFFFEFFFFFFFDFFFFFFFD'
00006BC0  C3C6C4C2 D9C14060                       1191  DC CL48'CFDBRA -2.5 M3 modes 1, 3-5'
00006BF0  FFFFFFFD FFFFFFFD                       1192  DC XL16'FFFFFFFDFFFFFFFDFFFFFFFEFFFFFFFE'
00006C00  C3C6C4C2 D9C14060                       1193  DC CL48'CFDBRA -2.5 M3 modes 6, 7'
00006C30  FFFFFFFE FFFFFFFD                       1194  DC XL16'FFFFFFFEFFFFFFFD0000000000000000'
00006C40  C3C6C4C2 D9C14060                       1195  DC CL48'CFDBRA -1.5 FPCR modes 1-3, 7'
00006C70  FFFFFFFF FFFFFFFF                       1196  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFF'
00006C80  C3C6C4C2 D9C14060                       1197  DC CL48'CFDBRA -1.5 M3 modes 1, 3-5'
00006CB0  FFFFFFFE FFFFFFFF                       1198  DC XL16'FFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFF'
00006CC0  C3C6C4C2 D9C14060                       1199  DC CL48'CFDBRA -1.5 M3 modes 6, 7'
00006CF0  FFFFFFFF FFFFFFFE                       1200  DC XL16'FFFFFFFFFFFFFFFE0000000000000000'
00006D00  C3C6C4C2 D9C14060                       1201  DC CL48'CFDBRA -0.5 FPCR modes 1-3, 7'
00006D30  00000000 00000000                       1202  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00006D40  C3C6C4C2 D9C14060                       1203  DC CL48'CFDBRA -0.5 M3 modes 1, 3-5'
00006D70  FFFFFFFF FFFFFFFF                       1204  DC XL16'FFFFFFFFFFFFFFFF0000000000000000'
00006D80  C3C6C4C2 D9C14060                       1205  DC CL48'CFDBRA -0.5 M3 modes 6, 7'
00006DB0  00000000 FFFFFFFF                       1206  DC XL16'00000000FFFFFFFF0000000000000000'
00006DC0  C3C6C4C2 D9C1404E                       1207  DC CL48'CFDBRA +0.5 FPCR modes 1-3, 7'
00006DF0  00000000 00000001                       1208  DC XL16'00000000000000010000000000000001'
00006E00  C3C6C4C2 D9C1404E                       1209  DC CL48'CFDBRA +0.5 M3 modes 1, 3-5'
00006E30  00000001 00000001                       1210  DC XL16'00000001000000010000000000000000'
00006E40  C3C6C4C2 D9C1404E                       1211  DC CL48'CFDBRA +0.5 M3 modes 6, 7'
00006E70  00000001 00000000                       1212  DC XL16'00000001000000000000000000000000'
00006E80  C3C6C4C2 D9C1404E                       1213  DC CL48'CFDBRA +1.5 FPCR modes 1-3, 7'
00006EB0  00000001 00000002                       1214  DC XL16'00000001000000020000000100000001'
00006EC0  C3C6C4C2 D9C1404E                       1215  DC CL48'CFDBRA +1.5 M3 modes 1, 3-5'
00006EF0  00000002 00000001                       1216  DC XL16'00000002000000010000000200000001'
00006F00  C3C6C4C2 D9C1404E                       1217  DC CL48'CFDBRA +1.5 M3 modes 6, 7'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00006F30  00000002 00000001                       1218  DC XL16'00000002000000010000000000000000'
00006F40  C3C6C4C2 D9C1404E                       1219  DC CL48'CFDBRA +2.5 FPCR modes 1-3, 7'
00006F70  00000002 00000003                       1220  DC XL16'00000002000000030000000200000003'
00006F80  C3C6C4C2 D9C1404E                       1221  DC CL48'CFDBRA +2.5 M3 modes 1, 3-5'
00006FB0  00000003 00000003                       1222  DC XL16'00000003000000030000000200000002'
00006FC0  C3C6C4C2 D9C1404E                       1223  DC CL48'CFDBRA +2.5 M3 modes 6, 7'
00006FF0  00000003 00000002                       1224  DC XL16'00000003000000020000000000000000'
00007000  C3C6C4C2 D9C1404E                       1225  DC CL48'CFDBRA +5.5 FPCR modes 1-3, 7'
00007030  00000005 00000006                       1226  DC XL16'00000005000000060000000500000005'
00007040  C3C6C4C2 D9C1404E                       1227  DC CL48'CFDBRA +5.5 M3 modes 1, 3-5'
00007070  00000006 00000005                       1228  DC XL16'00000006000000050000000600000005'
00007080  C3C6C4C2 D9C1404E                       1229  DC CL48'CFDBRA +5.5 M3 modes 6, 7'
000070B0  00000006 00000005                       1230  DC XL16'00000006000000050000000000000000'
000070C0  C3C6C4C2 D9C1404E                       1231  DC CL48'CFDBRA +9.5 FPCR modes 1-3, 7'
000070F0  00000009 0000000A                       1232  DC XL16'000000090000000A0000000900000009'
00007100  C3C6C4C2 D9C1404E                       1233  DC CL48'CFDBRA +9.5 M3 modes 1, 3-5'
00007130  0000000A 00000009                       1234  DC XL16'0000000A000000090000000A00000009'
00007140  C3C6C4C2 D9C1404E                       1235  DC CL48'CFDBRA +9.5 M3 modes 6, 7'
00007170  0000000A 00000009                       1236  DC XL16'0000000A000000090000000000000000'
00007180  C3C6C4C2 D9C1404E                       1237  DC CL48'CFDBRA +0.75 FPCR modes 1-3, 7'
000071B0  00000000 00000001                       1238  DC XL16'00000000000000010000000000000001'
000071C0  C3C6C4C2 D9C1404E                       1239  DC CL48'CFDBRA +0.75 M3 modes 1, 3-5'
000071F0  00000001 00000001                       1240  DC XL16'00000001000000010000000100000000'
00007200  C3C6C4C2 D9C1404E                       1241  DC CL48'CFDBRA +0.75 M3 modes 6, 7'
00007230  00000001 00000000                       1242  DC XL16'00000001000000000000000000000000'
00007240  C3C6C4C2 D9C1404E                       1243  DC CL48'CFDBRA +0.25 FPCR modes 1-3, 7'
00007270  00000000 00000001                       1244  DC XL16'00000000000000010000000000000001'
00007280  C3C6C4C2 D9C1404E                       1245  DC CL48'CFDBRA +0.25 M3 modes 1, 3-5'
000072B0  00000000 00000001                       1246  DC XL16'00000000000000010000000000000000'
000072C0  C3C6C4C2 D9C1404E                       1247  DC CL48'CFDBRA +0.25 M3 modes 6, 7'
000072F0  00000001 00000000                       1248  DC XL16'00000001000000000000000000000000'
00007300  C3C6C4C2 D9C14060                       1249  DC CL48'CFDBRA -0.75 FPCR modes 1-3, 7'
00007330  00000000 00000000                       1250  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00007340  C3C6C4C2 D9C14060                       1251  DC CL48'CFDBRA -0.75 M3 modes 1, 3-5'
00007370  FFFFFFFF FFFFFFFF                       1252  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFF00000000'
00007380  C3C6C4C2 D9C14060                       1253  DC CL48'CFDBRA -0.75 M3 modes 6, 7'
000073B0  00000000 FFFFFFFF                       1254  DC XL16'00000000FFFFFFFF0000000000000000'
000073C0  C3C6C4C2 D9C14060                       1255  DC CL48'CFDBRA -0.25 FPCR modes 1-3, 7'
000073F0  00000000 00000000                       1256  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00007400  C3C6C4C2 D9C14060                       1257  DC CL48'CFDBRA -0.25 M3 modes 1, 3-5'
00007430  00000000 FFFFFFFF                       1258  DC XL16'00000000FFFFFFFF0000000000000000'
00007440  C3C6C4C2 D9C14060                       1259  DC CL48'CFDBRA -0.25 M3 modes 6, 7'
00007470  00000000 FFFFFFFF                       1260  DC XL16'00000000FFFFFFFF0000000000000000'
00007480  C3C6C4C2 D9C14094                       1261  DC CL48'CFDBRA max+0.5 FPCR modes 1-3, 7'
000074B0  7FFFFFFF 7FFFFFFF                       1262  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF7FFFFFFF'
000074C0  C3C6C4C2 D9C14094                       1263  DC CL48'CFDBRA max+0.5 M3 modes 1, 3-5'
000074F0  7FFFFFFF 7FFFFFFF                       1264  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF7FFFFFFF'
00007500  C3C6C4C2 D9C14094                       1265  DC CL48'CFDBRA max+0.5 M3 modes 6, 7'
00007530  7FFFFFFF 7FFFFFFF                       1266  DC XL16'7FFFFFFF7FFFFFFF0000000000000000'
                                0000002D  00000001  1267 LINTRMO_NUM EQU (*-LINTRMO_GOOD)/64
                                                  1268 *
                                                  1269 *
                                00007540  00000001  1270 LINTRMOF_GOOD EQU *
00007540  C3C6C4C2 D9C14060                       1271  DC CL48'CFDBRA -9.5 FPCR modes 1-3, 7 FPCR'
00007570  00000001 00000001                       1272  DC XL16'00000001000000010000000100000001'
00007580  C3C6C4C2 D9C14060                       1273  DC CL48'CFDBRA -9.5 M3 modes 1, 3-5 FPCR'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000075B0  00080001 00080001                       1274  DC XL16'00080001000800010008000100080001'
000075C0  C3C6C4C2 D9C14060                       1275  DC CL48'CFDBRA -9.5 M3 modes 6, 7 FPCR'
000075F0  00080001 00080001                       1276  DC XL16'00080001000800010000000000000000'
00007600  C3C6C4C2 D9C14060                       1277  DC CL48'CFDBRA -5.5 FPCR modes 1-3, 7 FPCR'
00007630  00000001 00000001                       1278  DC XL16'00000001000000010000000100000001'
00007640  C3C6C4C2 D9C14060                       1279  DC CL48'CFDBRA -5.5 M3 modes 1, 3-5 FPCR'
00007670  00080001 00080001                       1280  DC XL16'00080001000800010008000100080001'
00007680  C3C6C4C2 D9C14060                       1281  DC CL48'CFDBRA -5.5 M3 modes 6, 7 FPCR'
000076B0  00080001 00080001                       1282  DC XL16'00080001000800010000000000000000'
000076C0  C3C6C4C2 D9C14060                       1283  DC CL48'CFDBRA -2.5 FPCR modes 1-3, 7 FPCR'
000076F0  00000001 00000001                       1284  DC XL16'00000001000000010000000100000001'
00007700  C3C6C4C2 D9C14060                       1285  DC CL48'CFDBRA -2.5 M3 modes 1, 3-5 FPCR'
00007730  00080001 00080001                       1286  DC XL16'00080001000800010008000100080001'
00007740  C3C6C4C2 D9C14060                       1287  DC CL48'CFDBRA -2.5 M3 modes 6, 7 FPCR'
00007770  00080001 00080001                       1288  DC XL16'00080001000800010000000000000000'
00007780  C3C6C4C2 D9C14060                       1289  DC CL48'CFDBRA -1.5 FPCR modes 1-3, 7 FPCR'
000077B0  00000001 00000001                       1290  DC XL16'00000001000000010000000100000001'
000077C0  C3C6C4C2 D9C14060                       1291  DC CL48'CFDBRA -1.5 M3 modes 1, 3-5 FPCR'
000077F0  00080001 00080001                       1292  DC XL16'00080001000800010008000100080001'
00007800  C3C6C4C2 D9C14060                       1293  DC CL48'CFDBRA -1.5 M3 modes 6, 7 FPCR'
00007830  00080001 00080001                       1294  DC XL16'00080001000800010000000000000000'
00007840  C3C6C4C2 D9C14060                       1295  DC CL48'CFDBRA -0.5 FPCR modes 1-3, 7 FPCR'
00007870  00000001 00000001                       1296  DC XL16'00000001000000010000000100000001'
00007880  C3C6C4C2 D9C14060                       1297  DC CL48'CFDBRA -0.5 M3 modes 1, 3-5 FPCR'
000078B0  00080001 00080001                       1298  DC XL16'00080001000800010008000100080001'
000078C0  C3C6C4C2 D9C14060                       1299  DC CL48'CFDBRA -0.5 M3 modes 6, 7 FPCR'
000078F0  00080001 00080001                       1300  DC XL16'00080001000800010000000000000000'
00007900  C3C6C4C2 D9C1404E                       1301  DC CL48'CFDBRA +0.5 FPCR modes 1-3, 7 FPCR'
00007930  00000002 00000002                       1302  DC XL16'00000002000000020000000200000002'
00007940  C3C6C4C2 D9C1404E                       1303  DC CL48'CFDBRA +0.5 M3 modes 1, 3-5 FPCR'
00007970  00080002 00080002                       1304  DC XL16'00080002000800020008000200080002'
00007980  C3C6C4C2 D9C1404E                       1305  DC CL48'CFDBRA +0.5 M3 modes 6, 7 FPCR'
000079B0  00080002 00080002                       1306  DC XL16'00080002000800020000000000000000'
000079C0  C3C6C4C2 D9C1404E                       1307  DC CL48'CFDBRA +1.5 FPCR modes 1-3, 7 FPCR'
000079F0  00000002 00000002                       1308  DC XL16'00000002000000020000000200000002'
00007A00  C3C6C4C2 D9C1404E                       1309  DC CL48'CFDBRA +1.5 M3 modes 1, 3-5 FPCR'
00007A30  00080002 00080002                       1310  DC XL16'00080002000800020008000200080002'
00007A40  C3C6C4C2 D9C1404E                       1311  DC CL48'CFDBRA +1.5 M3 modes 6, 7 FPCR'
00007A70  00080002 00080002                       1312  DC XL16'00080002000800020000000000000000'
00007A80  C3C6C4C2 D9C1404E                       1313  DC CL48'CFDBRA +2.5 FPCR modes 1-3, 7 FPCR'
00007AB0  00000002 00000002                       1314  DC XL16'00000002000000020000000200000002'
00007AC0  C3C6C4C2 D9C1404E                       1315  DC CL48'CFDBRA +2.5 M3 modes 1, 3-5 FPCR'
00007AF0  00080002 00080002                       1316  DC XL16'00080002000800020008000200080002'
00007B00  C3C6C4C2 D9C1404E                       1317  DC CL48'CFDBRA +2.5 M3 modes 6, 7 FPCR'
00007B30  00080002 00080002                       1318  DC XL16'00080002000800020000000000000000'
00007B40  C3C6C4C2 D9C1404E                       1319  DC CL48'CFDBRA +5.5 FPCR modes 1-3, 7 FPCR'
00007B70  00000002 00000002                       1320  DC XL16'00000002000000020000000200000002'
00007B80  C3C6C4C2 D9C1404E                       1321  DC CL48'CFDBRA +5.5 M3 modes 1, 3-5 FPCR'
00007BB0  00080002 00080002                       1322  DC XL16'00080002000800020008000200080002'
00007BC0  C3C6C4C2 D9C1404E                       1323  DC CL48'CFDBRA +5.5 M3 modes 6, 7 FPCR'
00007BF0  00080002 00080002                       1324  DC XL16'00080002000800020000000000000000'
00007C00  C3C6C4C2 D9C1404E                       1325  DC CL48'CFDBRA +9.5 FPCR modes 1-3, 7 FPCR'
00007C30  00000002 00000002                       1326  DC XL16'00000002000000020000000200000002'
00007C40  C3C6C4C2 D9C1404E                       1327  DC CL48'CFDBRA +9.5 M3 modes 1, 3-5 FPCR'
00007C70  00080002 00080002                       1328  DC XL16'00080002000800020008000200080002'
00007C80  C3C6C4C2 D9C1404E                       1329  DC CL48'CFDBRA +9.5 M3 modes 6, 7 FPCR'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00007CB0  00080002 00080002                       1330  DC XL16'00080002000800020000000000000000'
00007CC0  C3C6C4C2 D9C1404E                       1331  DC CL48'CFDBRA +0.75 FPCR modes 1-3, 7 FPCR'
00007CF0  00000002 00000002                       1332  DC XL16'00000002000000020000000200000002'
00007D00  C3C6C4C2 D9C1404E                       1333  DC CL48'CFDBRA +0.75 M3 modes 1, 3-5 FPCR'
00007D30  00080002 00080002                       1334  DC XL16'00080002000800020008000200080002'
00007D40  C3C6C4C2 D9C1404E                       1335  DC CL48'CFDBRA +0.75 M3 modes 6, 7 FPCR'
00007D70  00080002 00080002                       1336  DC XL16'00080002000800020000000000000000'
00007D80  C3C6C4C2 D9C1404E                       1337  DC CL48'CFDBRA +0.25 FPCR modes 1-3, 7 FPCR'
00007DB0  00000002 00000002                       1338  DC XL16'00000002000000020000000200000002'
00007DC0  C3C6C4C2 D9C1404E                       1339  DC CL48'CFDBRA +0.25 M3 modes 1, 3-5 FPCR'
00007DF0  00080002 00080002                       1340  DC XL16'00080002000800020008000200080002'
00007E00  C3C6C4C2 D9C1404E                       1341  DC CL48'CFDBRA +0.25 M3 modes 6, 7 FPCR'
00007E30  00080002 00080002                       1342  DC XL16'00080002000800020000000000000000'
00007E40  C3C6C4C2 D9C14060                       1343  DC CL48'CFDBRA -0.75 FPCR modes 1-3, 7 FPCR'
00007E70  00000001 00000001                       1344  DC XL16'00000001000000010000000100000001'
00007E80  C3C6C4C2 D9C14060                       1345  DC CL48'CFDBRA -0.75 M3 modes 1, 3-5 FPCR'
00007EB0  00080001 00080001                       1346  DC XL16'00080001000800010008000100080001'
00007EC0  C3C6C4C2 D9C14060                       1347  DC CL48'CFDBRA -0.75 M3 modes 6, 7 FPCR'
00007EF0  00080001 00080001                       1348  DC XL16'00080001000800010000000000000000'
00007F00  C3C6C4C2 D9C14060                       1349  DC CL48'CFDBRA -0.25 FPCR modes 1-3, 7 FPCR'
00007F30  00000001 00000001                       1350  DC XL16'00000001000000010000000100000001'
00007F40  C3C6C4C2 D9C14060                       1351  DC CL48'CFDBRA -0.25 M3 modes 1, 3-5 FPCR'
00007F70  00080001 00080001                       1352  DC XL16'00080001000800010008000100080001'
00007F80  C3C6C4C2 D9C14060                       1353  DC CL48'CFDBRA -0.25 M3 modes 6, 7 FPCR'
00007FB0  00080001 00080001                       1354  DC XL16'00080001000800010000000000000000'
00007FC0  C3C6C4C2 D9C14094                       1355  DC CL48'CFDBRA max+0.5 FPCR modes 1-3, 7 FPCR'
00007FF0  00000002 00800003                       1356  DC XL16'00000002008000030000000200000002'
00008000  C3C6C4C2 D9C14094                       1357  DC CL48'CFDBRA max+0.5 M3 modes 1, 3-5 FPCR'
00008030  00880003 00080002                       1358  DC XL16'00880003000800020088000300080002'
00008040  C3C6C4C2 D9C14094                       1359  DC CL48'CFDBRA max+0.5 M3 modes 6, 7 FPCR'
00008070  00880003 00080002                       1360  DC XL16'00880003000800020000000000000000'
                               0000002D  00000001  1361 LINTRMOF_NUM EQU (*-LINTRMOF_GOOD)/64
                                                  1362 *
                                                  1363 *
                               00008080  00000001  1364 XINTOUT_GOOD EQU *
00008080  C3C6E7C2 D9409985                       1365  DC CL48'CFXBR result pairs 1-2'
000080B0  00000001 00000001                       1366  DC XL16'00000001000000010000000200000002'
000080C0  C3C6E7C2 D9409985                       1367  DC CL48'CFXBR result pairs 3-4'
000080F0  00000004 00000004                       1368  DC XL16'0000000400000004FFFFFFFEFFFFFFFE'
00008100  C3C6E7C2 D9409985                       1369  DC CL48'CFXBR result pairs 5-6'
00008130  80000000 00000000                       1370  DC XL16'80000000000000008000000000000000'
00008140  C3C6E7C2 D9409985                       1371  DC CL48'CFXBR result pairs 7-8'
00008170  7FFFFFFF 00000000                       1372  DC XL16'7FFFFFFF000000008000000000000000'
00008180  C3C6E7C2 D9409985                       1373  DC CL48'CFXBR result pairs 9-10'
000081B0  7FFFFFFF 7FFFFFFF                       1374  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF00000000'
                               00000005  00000001  1375 XINTOUT_NUM EQU (*-XINTOUT_GOOD)/64
                                                  1376 *
                                                  1377 *
                               000081C0  00000001  1378 XINTFLGS_GOOD EQU *
000081C0  C3C6E7C2 D940C6D7                       1379  DC CL48'CFXBR FPCR pairs 1-2'
000081F0  00000002 F8000002                       1380  DC XL16'00000002F800000200000002F8000002'
00008200  C3C6E7C2 D940C6D7                       1381  DC CL48'CFXBR FPCR pairs 3-4'
00008230  00000002 F8000002                       1382  DC XL16'00000002F800000200000001F8000001'
00008240  C3C6E7C2 D940C6D7                       1383  DC CL48'CFXBR FPCR pairs 5-6'
00008270  00880003 F8008000                       1384  DC XL16'00880003F800800000880003F8008000'
00008280  C3C6E7C2 D940C6D7                       1385  DC CL48'CFXBR FPCR pairs 7-8'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000082B0  00880003 F8008000                       1386  DC XL16'00880003F800800000880003F8008000'
000082C0  C3C6E7C2 D940C6D7                       1387  DC CL48'CFXBR FPCR pairs 9-10'
000082F0  00000002 F8000002                       1388  DC XL16'00000002F800000200880003F8008000'
                             00000005  00000001  1389 XINTFLGS_NUM EQU (*-XINTFLGS_GOOD)/64
                                                  1390 *
                                                  1391 *
                             00008300  00000001  1392 XINTRMO_GOOD EQU *
00008300  C3C6E7C2 D9C14060                       1393  DC CL48'CFXBRA -9.5 FPCR modes 1-3, 7'
00008330  FFFFFFF7 FFFFFFF7                       1394  DC XL16'FFFFFFF7FFFFFFF7FFFFFFF6FFFFFFF7'
00008340  C3C6E7C2 D9C14060                       1395  DC CL48'CFXBRA -9.5 M3 modes 1, 3-5'
00008370  FFFFFFF6 FFFFFFF7                       1396  DC XL16'FFFFFFF6FFFFFFF7FFFFFFF6FFFFFFF7'
00008380  C3C6E7C2 D9C14060                       1397  DC CL48'CFXBRA -9.5 M3 modes 6, 7'
000083B0  FFFFFFF7 FFFFFFF6                       1398  DC XL16'FFFFFFF7FFFFFFF60000000000000000'
000083C0  C3C6E7C2 D9C14060                       1399  DC CL48'CFXBRA -5.5 FPCR modes 1-3, 7'
000083F0  FFFFFFFB FFFFFFFB                       1400  DC XL16'FFFFFFFBFFFFFFFBFFFFFFFAFFFFFFFB'
00008400  C3C6E7C2 D9C14060                       1401  DC CL48'CFXBRA -5.5 M3 modes 1, 3-5'
00008430  FFFFFFFA FFFFFFFB                       1402  DC XL16'FFFFFFFAFFFFFFFBFFFFFFFAFFFFFFFB'
00008440  C3C6E7C2 D9C14060                       1403  DC CL48'CFXBRA -5.5 M3 modes 6, 7'
00008470  FFFFFFFB FFFFFFFA                       1404  DC XL16'FFFFFFFBFFFFFFFA0000000000000000'
00008480  C3C6E7C2 D9C14060                       1405  DC CL48'CFXBRA -2.5 FPCR modes 1-3, 7'
000084B0  FFFFFFFE FFFFFFFE                       1406  DC XL16'FFFFFFFEFFFFFFFEFFFFFFFDFFFFFFFD'
000084C0  C3C6E7C2 D9C14060                       1407  DC CL48'CFXBRA -2.5 M3 modes 1, 3-5'
000084F0  FFFFFFFD FFFFFFFD                       1408  DC XL16'FFFFFFFDFFFFFFFDFFFFFFFEFFFFFFFE'
00008500  C3C6E7C2 D9C14060                       1409  DC CL48'CFXBRA -2.5 M3 modes 6, 7'
00008530  FFFFFFFE FFFFFFFD                       1410  DC XL16'FFFFFFFEFFFFFFFD0000000000000000'
00008540  C3C6E7C2 D9C14060                       1411  DC CL48'CFXBRA -1.5 FPCR modes 1-3, 7'
00008570  FFFFFFFF FFFFFFFF                       1412  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFF'
00008580  C3C6E7C2 D9C14060                       1413  DC CL48'CFXBRA -1.5 M3 modes 1, 3-5'
000085B0  FFFFFFFE FFFFFFFF                       1414  DC XL16'FFFFFFFEFFFFFFFFFFFFFFFEFFFFFFFF'
000085C0  C3C6E7C2 D9C14060                       1415  DC CL48'CFXBRA -1.5 M3 modes 6, 7'
000085F0  FFFFFFFF FFFFFFFE                       1416  DC XL16'FFFFFFFFFFFFFFFE0000000000000000'
00008600  C3C6E7C2 D9C14060                       1417  DC CL48'CFXBRA -0.5 FPCR modes 1-3, 7'
00008630  00000000 00000000                       1418  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00008640  C3C6E7C2 D9C14060                       1419  DC CL48'CFXBRA -0.5 M3 modes 1, 3-5'
00008670  FFFFFFFF FFFFFFFF                       1420  DC XL16'FFFFFFFFFFFFFFFF0000000000000000'
00008680  C3C6E7C2 D9C14060                       1421  DC CL48'CFXBRA -0.5 M3 modes 6, 7'
000086B0  00000000 FFFFFFFF                       1422  DC XL16'00000000FFFFFFFF0000000000000000'
000086C0  C3C6E7C2 D9C1404E                       1423  DC CL48'CFXBRA +0.5 FPCR modes 1-3, 7'
000086F0  00000000 00000001                       1424  DC XL16'00000000000000010000000000000001'
00008700  C3C6E7C2 D9C1404E                       1425  DC CL48'CFXBRA +0.5 M3 modes 1, 3-5'
00008730  00000001 00000001                       1426  DC XL16'00000001000000010000000000000000'
00008740  C3C6E7C2 D9C1404E                       1427  DC CL48'CFXBRA +0.5 M3 modes 6, 7'
00008770  00000001 00000000                       1428  DC XL16'00000001000000000000000000000000'
00008780  C3C6E7C2 D9C1404E                       1429  DC CL48'CFXBRA +1.5 FPCR modes 1-3, 7'
000087B0  00000001 00000002                       1430  DC XL16'00000001000000020000000100000001'
000087C0  C3C6E7C2 D9C1404E                       1431  DC CL48'CFXBRA +1.5 M3 modes 1, 3-5'
000087F0  00000002 00000001                       1432  DC XL16'00000002000000010000000200000001'
00008800  C3C6E7C2 D9C1404E                       1433  DC CL48'CFXBRA +1.5 M3 modes 6, 7'
00008830  00000002 00000001                       1434  DC XL16'00000002000000010000000000000000'
00008840  C3C6E7C2 D9C1404E                       1435  DC CL48'CFXBRA +2.5 FPCR modes 1-3, 7'
00008870  00000002 00000003                       1436  DC XL16'00000002000000030000000200000003'
00008880  C3C6E7C2 D9C1404E                       1437  DC CL48'CFXBRA +2.5 M3 modes 1, 3-5'
000088B0  00000003 00000003                       1438  DC XL16'00000003000000030000000200000002'
000088C0  C3C6E7C2 D9C1404E                       1439  DC CL48'CFXBRA +2.5 M3 modes 6, 7'
000088F0  00000003 00000002                       1440  DC XL16'00000003000000020000000000000000'
00008900  C3C6E7C2 D9C1404E                       1441  DC CL48'CFXBRA +5.5 FPCR modes 1-3, 7'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00008930  00000005 00000006                       1442  DC XL16'00000005000000060000000500000005'
00008940  C3C6E7C2 D9C1404E                       1443  DC CL48'CFXBRA +5.5 M3 modes 1, 3-5'
00008970  00000006 00000005                       1444  DC XL16'00000006000000050000000600000005'
00008980  C3C6E7C2 D9C1404E                       1445  DC CL48'CFXBRA +5.5 M3 modes 6, 7'
000089B0  00000006 00000005                       1446  DC XL16'00000006000000050000000000000000'
000089C0  C3C6E7C2 D9C1404E                       1447  DC CL48'CFXBRA +9.5 FPCR modes 1-3, 7'
000089F0  00000009 0000000A                       1448  DC XL16'000000090000000A0000000900000009'
00008A00  C3C6E7C2 D9C1404E                       1449  DC CL48'CFXBRA +9.5 M3 modes 1, 3-5'
00008A30  0000000A 00000009                       1450  DC XL16'0000000A000000090000000A00000009'
00008A40  C3C6E7C2 D9C1404E                       1451  DC CL48'CFXBRA +9.5 M3 modes 6, 7'
00008A70  0000000A 00000009                       1452  DC XL16'0000000A000000090000000000000000'
00008A80  C3C6E7C2 D9C1404E                       1453  DC CL48'CFXBRA +0.75 FPCR modes 1-3, 7'
00008AB0  00000000 00000001                       1454  DC XL16'00000000000000010000000000000001'
00008AC0  C3C6E7C2 D9C1404E                       1455  DC CL48'CFXBRA +0.75 M3 modes 1, 3-5'
00008AF0  00000001 00000001                       1456  DC XL16'00000001000000010000000100000000'
00008B00  C3C6E7C2 D9C1404E                       1457  DC CL48'CFXBRA +0.75 M3 modes 6, 7'
00008B30  00000001 00000000                       1458  DC XL16'00000001000000000000000000000000'
00008B40  C3C6E7C2 D9C1404E                       1459  DC CL48'CFXBRA +0.25 FPCR modes 1-3, 7'
00008B70  00000000 00000001                       1460  DC XL16'00000000000000010000000000000001'
00008B80  C3C6E7C2 D9C1404E                       1461  DC CL48'CFXBRA +0.25 M3 modes 1, 3-5'
00008BB0  00000000 00000001                       1462  DC XL16'00000000000000010000000000000000'
00008BC0  C3C6E7C2 D9C1404E                       1463  DC CL48'CFXBRA +0.25 M3 modes 6, 7'
00008BF0  00000001 00000000                       1464  DC XL16'00000001000000000000000000000000'
00008C00  C3C6E7C2 D9C14060                       1465  DC CL48'CFXBRA -0.75 FPCR modes 1-3, 7'
00008C30  00000000 00000000                       1466  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00008C40  C3C6E7C2 D9C14060                       1467  DC CL48'CFXBRA -0.75 M3 modes 1, 3-5'
00008C70  FFFFFFFF FFFFFFFF                       1468  DC XL16'FFFFFFFFFFFFFFFFFFFFFFFF00000000'
00008C80  C3C6E7C2 D9C14060                       1469  DC CL48'CFXBRA -0.75 M3 modes 6, 7'
00008CB0  00000000 FFFFFFFF                       1470  DC XL16'00000000FFFFFFFF0000000000000000'
00008CC0  C3C6E7C2 D9C14060                       1471  DC CL48'CFXBRA -0.25 FPCR modes 1-3, 7'
00008CF0  00000000 00000000                       1472  DC XL16'0000000000000000FFFFFFFFFFFFFFFF'
00008D00  C3C6E7C2 D9C14060                       1473  DC CL48'CFXBRA -0.25 M3 modes 1, 3-5'
00008D30  00000000 FFFFFFFF                       1474  DC XL16'00000000FFFFFFFF0000000000000000'
00008D40  C3C6E7C2 D9C14060                       1475  DC CL48'CFXBRA -0.25 M3 modes 6, 7'
00008D70  00000000 FFFFFFFF                       1476  DC XL16'00000000FFFFFFFF0000000000000000'
00008D80  C3C6E7C2 D9C14094                       1477  DC CL48'CFXBRA max+0.5 FPCR modes 1-3, 7'
00008DB0  7FFFFFFF 7FFFFFFF                       1478  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF7FFFFFFF'
00008DC0  C3C6E7C2 D9C14094                       1479  DC CL48'CFXBRA max+0.5 M3 modes 1, 3-5'
00008DF0  7FFFFFFF 7FFFFFFF                       1480  DC XL16'7FFFFFFF7FFFFFFF7FFFFFFF7FFFFFFF'
00008E00  C3C6E7C2 D9C14094                       1481  DC CL48'CFXBRA max+0.5 M3 modes 6, 7'
00008E30  7FFFFFFF 7FFFFFFF                       1482  DC XL16'7FFFFFFF7FFFFFFF0000000000000000'
                                 0000002D  00000001  1483 XINTRMO_NUM EQU (*-XINTRMO_GOOD)/64
                                                     1484 *
                                                     1485 *
                                 00008E40  00000001  1486 XINTRMOF_GOOD EQU *
00008E40  C3C6E7C2 D9C14060                       1487  DC CL48'CFXBRA -9.5 FPCR modes 1-3, 7 FPCR'
00008E70  00000001 00000001                       1488  DC XL16'00000001000000010000000100000001'
00008E80  C3C6E7C2 D9C14060                       1489  DC CL48'CFXBRA -9.5 M3 modes 1, 3-5 FPCR'
00008EB0  00080001 00080001                       1490  DC XL16'00080001000800010008000100080001'
00008EC0  C3C6E7C2 D9C14060                       1491  DC CL48'CFXBRA -9.5 M3 modes 6, 7 FPCR'
00008EF0  00080001 00080001                       1492  DC XL16'00080001000800010000000000000000'
00008F00  C3C6E7C2 D9C14060                       1493  DC CL48'CFXBRA -5.5 FPCR modes 1-3, 7 FPCR'
00008F30  00000001 00000001                       1494  DC XL16'00000001000000010000000100000001'
00008F40  C3C6E7C2 D9C14060                       1495  DC CL48'CFXBRA -5.5 M3 modes 1, 3-5 FPCR'
00008F70  00080001 00080001                       1496  DC XL16'00080001000800010008000100080001'
00008F80  C3C6E7C2 D9C14060                       1497  DC CL48'CFXBRA -5.5 M3 modes 6, 7 FPCR'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00008FB0  00080001 00080001                       1498  DC XL16'00080001000800010000000000000000'
00008FC0  C3C6E7C2 D9C14060                       1499  DC CL48'CFXBRA -2.5 FPCR modes 1-3, 7 FPCR'
00008FF0  00000001 00000001                       1500  DC XL16'00000001000000010000000100000001'
00009000  C3C6E7C2 D9C14060                       1501  DC CL48'CFXBRA -2.5 M3 modes 1, 3-5 FPCR'
00009030  00080001 00080001                       1502  DC XL16'00080001000800010008000100080001'
00009040  C3C6E7C2 D9C14060                       1503  DC CL48'CFXBRA -2.5 M3 modes 6, 7 FPCR'
00009070  00080001 00080001                       1504  DC XL16'00080001000800010000000000000000'
00009080  C3C6E7C2 D9C14060                       1505  DC CL48'CFXBRA -1.5 FPCR modes 1-3, 7 FPCR'
000090B0  00000001 00000001                       1506  DC XL16'00000001000000010000000100000001'
000090C0  C3C6E7C2 D9C14060                       1507  DC CL48'CFXBRA -1.5 M3 modes 1, 3-5 FPCR'
000090F0  00080001 00080001                       1508  DC XL16'00080001000800010008000100080001'
00009100  C3C6E7C2 D9C14060                       1509  DC CL48'CFXBRA -1.5 M3 modes 6, 7 FPCR'
00009130  00080001 00080001                       1510  DC XL16'00080001000800010000000000000000'
00009140  C3C6E7C2 D9C14060                       1511  DC CL48'CFXBRA -0.5 FPCR modes 1-3, 7 FPCR'
00009170  00000001 00000001                       1512  DC XL16'00000001000000010000000100000001'
00009180  C3C6E7C2 D9C14060                       1513  DC CL48'CFXBRA -0.5 M3 modes 1, 3-5 FPCR'
000091B0  00080001 00080001                       1514  DC XL16'00080001000800010008000100080001'
000091C0  C3C6E7C2 D9C14060                       1515  DC CL48'CFXBRA -0.5 M3 modes 6, 7 FPCR'
000091F0  00080001 00080001                       1516  DC XL16'00080001000800010000000000000000'
00009200  C3C6E7C2 D9C1404E                       1517  DC CL48'CFXBRA +0.5 FPCR modes 1-3, 7 FPCR'
00009230  00000002 00000002                       1518  DC XL16'00000002000000020000000200000002'
00009240  C3C6E7C2 D9C1404E                       1519  DC CL48'CFXBRA +0.5 M3 modes 1, 3-5 FPCR'
00009270  00080002 00080002                       1520  DC XL16'00080002000800020008000200080002'
00009280  C3C6E7C2 D9C1404E                       1521  DC CL48'CFXBRA +0.5 M3 modes 6, 7 FPCR'
000092B0  00080002 00080002                       1522  DC XL16'00080002000800020000000000000000'
000092C0  C3C6E7C2 D9C1404E                       1523  DC CL48'CFXBRA +1.5 FPCR modes 1-3, 7 FPCR'
000092F0  00000002 00000002                       1524  DC XL16'00000002000000020000000200000002'
00009300  C3C6E7C2 D9C1404E                       1525  DC CL48'CFXBRA +1.5 M3 modes 1, 3-5 FPCR'
00009330  00080002 00080002                       1526  DC XL16'00080002000800020008000200080002'
00009340  C3C6E7C2 D9C1404E                       1527  DC CL48'CFXBRA +1.5 M3 modes 6, 7 FPCR'
00009370  00080002 00080002                       1528  DC XL16'00080002000800020000000000000000'
00009380  C3C6E7C2 D9C1404E                       1529  DC CL48'CFXBRA +2.5 FPCR modes 1-3, 7 FPCR'
000093B0  00000002 00000002                       1530  DC XL16'00000002000000020000000200000002'
000093C0  C3C6E7C2 D9C1404E                       1531  DC CL48'CFXBRA +2.5 M3 modes 1, 3-5 FPCR'
000093F0  00080002 00080002                       1532  DC XL16'00080002000800020008000200080002'
00009400  C3C6E7C2 D9C1404E                       1533  DC CL48'CFXBRA +2.5 M3 modes 6, 7 FPCR'
00009430  00080002 00080002                       1534  DC XL16'00080002000800020000000000000000'
00009440  C3C6E7C2 D9C1404E                       1535  DC CL48'CFXBRA +5.5 FPCR modes 1-3, 7 FPCR'
00009470  00000002 00000002                       1536  DC XL16'00000002000000020000000200000002'
00009480  C3C6E7C2 D9C1404E                       1537  DC CL48'CFXBRA +5.5 M3 modes 1, 3-5 FPCR'
000094B0  00080002 00080002                       1538  DC XL16'00080002000800020008000200080002'
000094C0  C3C6E7C2 D9C1404E                       1539  DC CL48'CFXBRA +5.5 M3 modes 6, 7 FPCR'
000094F0  00080002 00080002                       1540  DC XL16'00080002000800020000000000000000'
00009500  C3C6E7C2 D9C1404E                       1541  DC CL48'CFXBRA +9.5 FPCR modes 1-3, 7 FPCR'
00009530  00000002 00000002                       1542  DC XL16'00000002000000020000000200000002'
00009540  C3C6E7C2 D9C1404E                       1543  DC CL48'CFXBRA +9.5 M3 modes 1, 3-5 FPCR'
00009570  00080002 00080002                       1544  DC XL16'00080002000800020008000200080002'
00009580  C3C6E7C2 D9C1404E                       1545  DC CL48'CFXBRA +9.5 M3 modes 6, 7 FPCR'
000095B0  00080002 00080002                       1546  DC XL16'00080002000800020000000000000000'
000095C0  C3C6E7C2 D9C1404E                       1547  DC CL48'CFXBRA +0.75 FPCR modes 1-3, 7 FPCR'
000095F0  00000002 00000002                       1548  DC XL16'00000002000000020000000200000002'
00009600  C3C6E7C2 D9C1404E                       1549  DC CL48'CFXBRA +0.75 M3 modes 1, 3-5 FPCR'
00009630  00080002 00080002                       1550  DC XL16'00080002000800020008000200080002'
00009640  C3C6E7C2 D9C1404E                       1551  DC CL48'CFXBRA +0.75 M3 modes 6, 7 FPCR'
00009670  00080002 00080002                       1552  DC XL16'00080002000800020000000000000000'
00009680  C3C6E7C2 D9C1404E                       1553  DC CL48'CFXBRA +0.25 FPCR modes 1-3, 7 FPCR'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

000096B0  00000002 00000002                       1554  DC XL16'00000002000000020000000200000002'
000096C0  C3C6E7C2 D9C1404E                       1555  DC CL48'CFXBRA +0.25 M3 modes 1, 3-5 FPCR'
000096F0  00080002 00080002                       1556  DC XL16'00080002000800020008000200080002'
00009700  C3C6E7C2 D9C1404E                       1557  DC CL48'CFXBRA +0.25 M3 modes 6, 7 FPCR'
00009730  00080002 00080002                       1558  DC XL16'00080002000800020000000000000000'
00009740  C3C6E7C2 D9C14060                       1559  DC CL48'CFXBRA -0.75 FPCR modes 1-3, 7 FPCR'
00009770  00000001 00000001                       1560  DC XL16'00000001000000010000000100000001'
00009780  C3C6E7C2 D9C14060                       1561  DC CL48'CFXBRA -0.75 M3 modes 1, 3-5 FPCR'
000097B0  00080001 00080001                       1562  DC XL16'00080001000800010008000100080001'
000097C0  C3C6E7C2 D9C14060                       1563  DC CL48'CFXBRA -0.75 M3 modes 6, 7 FPCR'
000097F0  00080001 00080001                       1564  DC XL16'00080001000800010000000000000000'
00009800  C3C6E7C2 D9C14060                       1565  DC CL48'CFXBRA -0.25 FPCR modes 1-3, 7 FPCR'
00009830  00000001 00000001                       1566  DC XL16'00000001000000010000000100000001'
00009840  C3C6E7C2 D9C14060                       1567  DC CL48'CFXBRA -0.25 M3 modes 1, 3-5 FPCR'
00009870  00080001 00080001                       1568  DC XL16'00080001000800010008000100080001'
00009880  C3C6E7C2 D9C14060                       1569  DC CL48'CFXBRA -0.25 M3 modes 6, 7 FPCR'
000098B0  00080001 00080001                       1570  DC XL16'00080001000800010000000000000000'
000098C0  C3C6E7C2 D9C14094                       1571  DC CL48'CFXBRA max+0.5 FPCR modes 1-3, 7 FPCR'
000098F0  00000002 00800003                       1572  DC XL16'00000002008000030000000200000002'
00009900  C3C6E7C2 D9C14094                       1573  DC CL48'CFXBRA max+0.5 M3 modes 1, 3-5 FPCR'
00009930  00880003 00080002                       1574  DC XL16'00880003000800020088000300080002'
00009940  C3C6E7C2 D9C14094                       1575  DC CL48'CFXBRA max+0.5 M3 modes 6, 7 FPCR'
00009970  00880003 00080002                       1576  DC XL16'00880003000800020000000000000000'
                             0000002D  00000001  1577 XINTRMOF_NUM EQU (*-XINTRMOF_GOOD)/64
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00009980                                          1579 HELPERS  DS    0H       (R12 base of helper subroutines)

                                                  1581 ***********************************************************************
                                                  1582 *               REPORT UNEXPECTED PROGRAM CHECK
                                                  1583 ***********************************************************************

00009980                                          1585 PGMCK    DS    0H
00009980  F342 C072 F08E      000099F2  0000008E  1586          UNPK  PROGCODE(L'PROGCODE+1),PCINTCD(L'PCINTCD+1)
00009986  926B C076                     000099F6  1587          MVI   PGMCOMMA,C','
0000998A  DC03 C072 C178      000099F2  00009AF8  1588          TR    PROGCODE,HEXTRTAB

00009990  F384 C07C F150      000099FC  00000150  1590          UNPK  PGMPSW+(0*9)(9),PCOLDPSW+(0*4)(5)
00009996  9240 C084                     00009A04  1591          MVI   PGMPSW+(0*9)+8,C' '
0000999A  DC07 C07C C178      000099FC  00009AF8  1592          TR    PGMPSW+(0*9)(8),HEXTRTAB

000099A0  F384 C085 F154      00009A05  00000154  1594          UNPK  PGMPSW+(1*9)(9),PCOLDPSW+(1*4)(5)
000099A6  9240 C08D                     00009A0D  1595          MVI   PGMPSW+(1*9)+8,C' '
000099AA  DC07 C085 C178      00009A05  00009AF8  1596          TR    PGMPSW+(1*9)(8),HEXTRTAB

000099B0  F384 C08E F158      00009A0E  00000158  1598          UNPK  PGMPSW+(2*9)(9),PCOLDPSW+(2*4)(5)
000099B6  9240 C096                     00009A16  1599          MVI   PGMPSW+(2*9)+8,C' '
000099BA  DC07 C08E C178      00009A0E  00009AF8  1600          TR    PGMPSW+(2*9)(8),HEXTRTAB

000099C0  F384 C097 F15C      00009A17  0000015C  1602          UNPK  PGMPSW+(3*9)(9),PCOLDPSW+(3*4)(5)
000099C6  9240 C09F                     00009A1F  1603          MVI   PGMPSW+(3*9)+8,C' '
000099CA  DC07 C097 C178      00009A17  00009AF8  1604          TR    PGMPSW+(3*9)(8),HEXTRTAB

000099D0  4100 0042                     00000042  1606          LA    R0,L'PROGMSG     R0 <== length of message
000099D4  4110 C05E                     000099DE  1607          LA    R1,PROGMSG       R1 --> the message text itself
000099D8  4520 C27A                     00009BFA  1608          BAL   R2,MSG           Go display this message
                                                  1609
000099DC  07FD                                    1610          BR    R13              Return to caller



000099DE                                          1612 PROGMSG  DS   0CL66
000099DE  D7D9D6C7 D9C1D440                       1613          DC    CL20'PROGRAM CHECK! CODE '
000099F2  88888888                                1614 PROGCODE DC    CL4'hhhh'
000099F6  6B                                      1615 PGMCOMMA DC    CL1','
000099F7  40D7E2E6 40                             1616          DC    CL5' PSW '
000099FC  88888888 88888888                       1617 PGMPSW   DC    CL36'hhhhhhhh hhhhhhhh hhhhhhhh hhhhhhhh '
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1619 ********************************************************************
                                                  1620 *                    VERIFICATION ROUTINE
                                                  1621 ********************************************************************

00009A20                                          1623 VERISUB  DS    0H
                                                  1624 *
                                                  1625 **       Loop through the VERIFY TABLE...
                                                  1626 *

00009A20  4110 C32C                    00009CAC  1628          LA    R1,VERIFTAB      R1 --> Verify table
00009A24  4120 000C                    0000000C  1629          LA    R2,VERIFLEN      R2 <== Number of entries
00009A28  0D30                                   1630          BASR  R3,0             Set top of loop

00009A2A  9846 1000                    00000000  1632          LM    R4,R6,0(R1)      Load verify table values
00009A2E  4D70 C0C2                    00009A42  1633          BAS   R7,VERIFY        Verify results
00009A32  4110 100C                    0000000C  1634          LA    R1,12(,R1)       Next verify table entry
00009A36  0623                                   1635          BCTR  R2,R3            Loop through verify table

00009A38  9500 C278                    00009BF8  1637          CLI   FAILFLAG,X'00'   Did all tests verify okay?
00009A3C  078D                                   1638          BER   R13              Yes, return to caller
00009A3E  47F0 F238                    00000238  1639          B     FAIL             No, load FAILURE disabled wait PSW




                                                  1641 *
                                                  1642 **       Loop through the ACTUAL / EXPECTED results...
                                                  1643 *

00009A42  0D80                                   1645 VERIFY   BASR  R8,0             Set top of loop

00009A44  D50F 4000 5030      00000000  00000030  1647          CLC   0(16,R4),48(R5)  Actual results == Expected results?
00009A4A  4770 C0DA                    00009A5A  1648          BNE   VERIFAIL         No, show failure
00009A4E  4140 4010                    00000010  1649 VERINEXT LA    R4,16(,R4)       Next actual result
00009A52  4150 5040                    00000040  1650          LA    R5,64(,R5)       Next expected result
00009A56  0668                                   1651          BCTR  R6,R8            Loop through results

00009A58  07F7                                   1653          BR    R7               Return to caller
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1655 ***********************************************************************
                                                  1656 *                         Report the failure...
                                                  1657 ***********************************************************************

00009A5A  9005 C250                     00009BD0  1659 VERIFAIL STM   R0,R5,SAVER0R5   Save registers
00009A5E  92FF C278                     00009BF8  1660          MVI   FAILFLAG,X'FF'   Remember verification failure
                                                  1661 *
                                                  1662 **       First, show them the description...
                                                  1663 *
00009A62  D22F C1E0 5000      00009B60  00000000  1664          MVC   FAILDESC,0(R5)   Save results/test description
00009A68  4100 0044                     00000044  1665          LA    R0,L'FAILMSG1    R0 <== length of message
00009A6C  4110 C1CC                     00009B4C  1666          LA    R1,FAILMSG1      R1 --> the message text itself
00009A70  4520 C27A                     00009BFA  1667          BAL   R2,MSG           Go display this message
                                                  1668 *
                                                  1669 **       Save address of actual and expected results
                                                  1670 *
00009A74  5040 C24C                     00009BCC  1671          ST    R4,AACTUAL       Save A(actual results)
00009A78  4150 5030                     00000030  1672          LA    R5,48(,R5)       R5 ==> expected results
00009A7C  5050 C248                     00009BC8  1673          ST    R5,AEXPECT       Save A(expected results)
                                                  1674 *
                                                  1675 **       Format and show them the EXPECTED ("Want") results...
                                                  1676 *
00009A80  D205 C210 C3C0      00009B90  00009D40  1677          MVC   WANTGOT,=CL6'Want: '
00009A86  F384 C216 C248      00009B96  00009BC8  1678          UNPK  FAILADR(L'FAILADR+1),AEXPECT(L'AEXPECT+1)
00009A8C  9240 C21E                     00009B9E  1679          MVI   BLANKEQ,C' '
00009A90  DC07 C216 C178      00009B96  00009AF8  1680          TR    FAILADR,HEXTRTAB

00009A96  F384 C221 5000      00009BA1  00000000  1682          UNPK  FAILVALS+(0*9)(9),(0*4)(5,R5)
00009A9C  9240 C229                     00009BA9  1683          MVI   FAILVALS+(0*9)+8,C' '
00009AA0  DC07 C221 C178      00009BA1  00009AF8  1684          TR    FAILVALS+(0*9)(8),HEXTRTAB

00009AA6  F384 C22A 5004      00009BAA  00000004  1686          UNPK  FAILVALS+(1*9)(9),(1*4)(5,R5)
00009AAC  9240 C232                     00009BB2  1687          MVI   FAILVALS+(1*9)+8,C' '
00009AB0  DC07 C22A C178      00009BAA  00009AF8  1688          TR    FAILVALS+(1*9)(8),HEXTRTAB

00009AB6  F384 C233 5008      00009BB3  00000008  1690          UNPK  FAILVALS+(2*9)(9),(2*4)(5,R5)
00009ABC  9240 C23B                     00009BBB  1691          MVI   FAILVALS+(2*9)+8,C' '
00009AC0  DC07 C233 C178      00009BB3  00009AF8  1692          TR    FAILVALS+(2*9)(8),HEXTRTAB

00009AC6  F384 C23C 500C      00009BBC  0000000C  1694          UNPK  FAILVALS+(3*9)(9),(3*4)(5,R5)
00009ACC  9240 C244                     00009BC4  1695          MVI   FAILVALS+(3*9)+8,C' '
00009AD0  DC07 C23C C178      00009BBC  00009AF8  1696          TR    FAILVALS+(3*9)(8),HEXTRTAB

00009AD6  4100 0035                     00000035  1698          LA    R0,L'FAILMSG2    R0 <== length of message
00009ADA  4110 C210                     00009B90  1699          LA    R1,FAILMSG2      R1 --> the message text itself
00009ADE  4520 C27A                     00009BFA  1700          BAL   R2,MSG           Go display this message
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1702 *
                                                  1703 **       Format and show them the ACTUAL ("Got") results...
                                                  1704 *
00009AE2  D205 C210 C3C6      00009B90  00009D46  1705          MVC   WANTGOT,=CL6'Got:  '
00009AE8  F384 C216 C24C      00009B96  00009BCC  1706          UNPK  FAILADR(L'FAILADR+1),AACTUAL(L'AACTUAL+1)
00009AEE  9240 C21E                     00009B9E  1707          MVI   BLANKEQ,C' '
00009AF2  DC07 C216 C178      00009B96  00009AF8  1708          TR    FAILADR,HEXTRTAB

00009AF8  F384 C221 4000      00009BA1  00000000  1710          UNPK  FAILVALS+(0*9)(9),(0*4)(5,R4)
00009AFE  9240 C229                     00009BA9  1711          MVI   FAILVALS+(0*9)+8,C' '
00009B02  DC07 C221 C178      00009BA1  00009AF8  1712          TR    FAILVALS+(0*9)(8),HEXTRTAB

00009B08  F384 C22A 4004      00009BAA  00000004  1714          UNPK  FAILVALS+(1*9)(9),(1*4)(5,R4)
00009B0E  9240 C232                     00009BB2  1715          MVI   FAILVALS+(1*9)+8,C' '
00009B12  DC07 C22A C178      00009BAA  00009AF8  1716          TR    FAILVALS+(1*9)(8),HEXTRTAB

00009B18  F384 C233 4008      00009BB3  00000008  1718          UNPK  FAILVALS+(2*9)(9),(2*4)(5,R4)
00009B1E  9240 C23B                     00009BBB  1719          MVI   FAILVALS+(2*9)+8,C' '
00009B22  DC07 C233 C178      00009BB3  00009AF8  1720          TR    FAILVALS+(2*9)(8),HEXTRTAB

00009B28  F384 C23C 400C      00009BBC  0000000C  1722          UNPK  FAILVALS+(3*9)(9),(3*4)(5,R4)
00009B2E  9240 C244                     00009BC4  1723          MVI   FAILVALS+(3*9)+8,C' '
00009B32  DC07 C23C C178      00009BBC  00009AF8  1724          TR    FAILVALS+(3*9)(8),HEXTRTAB

00009B38  4100 0035                     00000035  1726          LA    R0,L'FAILMSG2    R0 <== length of message
00009B3C  4110 C210                     00009B90  1727          LA    R1,FAILMSG2      R1 --> the message text itself
00009B40  4520 C27A                     00009BFA  1728          BAL   R2,MSG           Go display this message

00009B44  9805 C250                     00009BD0  1730          LM    R0,R5,SAVER0R5   Restore registers
00009B48  47F0 C0CE                     00009A4E  1731          B     VERINEXT         Continue with verification...


00009B4C                                          1733 FAILMSG1 DS   0CL68
00009B4C  C3D6D4D7 C1D9C9E2                       1734          DC    CL20'COMPARISON FAILURE! '
00009B60  4D8485A2 83998997                       1735 FAILDESC DC    CL48'(description)'


00009B90                                          1737 FAILMSG2 DS   0CL53
00009B90  40404040 4040                           1738 WANTGOT  DC    CL6' '           'Want: ' -or- 'Got:  '
00009B96  C1C1C1C1 C1C1C1C1                       1739 FAILADR  DC    CL8'AAAAAAAA'
00009B9E  407E40                                  1740 BLANKEQ  DC    CL3' = '
00009BA1  88888888 88888888                       1741 FAILVALS DC    CL36'hhhhhhhh hhhhhhhh hhhhhhhh hhhhhhhh '


00009BC8  00000000                                1743 AEXPECT  DC    F'0'             ==> Expected ("Want") results
00009BCC  00000000                                1744 AACTUAL  DC    F'0'             ==> Actual ("Got") results
00009BD0  00000000 00000000                       1745 SAVER0R5 DC    6F'0'            Registers R0 - R5 save area
00009BE8  F0F1F2F3 F4F5F6F7                       1746 CHARHEX  DC    CL16'0123456789ABCDEF'
                              00009AF8  00000010  1747 HEXTRTAB EQU   CHARHEX-X'F0'    Hexadecimal translation table
00009BF8  00                                      1748 FAILFLAG DC    X'00'            FF = Fail, 00 = Success
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1750 ***********************************************************************
                                                  1751 *        Issue HERCULES MESSAGE pointed to by R1, length in R0
                                                  1752 ***********************************************************************

00009BFA  4900 C3BC                     00009D3C  1754 MSG      CH    R0,=H'0'                Do we even HAVE a message?
00009BFE  07D2                                    1755          BNHR  R2                      No, ignore

00009C00  9002 C2B0                     00009C30  1757          STM   R0,R2,MSGSAVE           Save registers

00009C04  4900 C3BE                     00009D3E  1759          CH    R0,=AL2(L'MSGMSG)       Message length within limits?
00009C08  47D0 C290                     00009C10  1760          BNH   MSGOK                   Yes, continue
00009C0C  4100 005F                     0000005F  1761          LA    R0,L'MSGMSG             No, set to maximum

00009C10  1820                                    1763 MSGOK    LR    R2,R0                   Copy length to work register
00009C12  0620                                    1764          BCTR  R2,0                    Minus-1 for execute
00009C14  4420 C2BC                     00009C3C  1765          EX    R2,MSGMVC               Copy message to O/P buffer

00009C18  4120 200A                     0000000A  1767          LA    R2,1+L'MSGCMD(,R2)      Calculate true command length
00009C1C  4110 C2C2                     00009C42  1768          LA    R1,MSGCMD               Point to true command

00009C20  83120008                                1770          DC    X'83',X'12',X'0008'     Issue Hercules Diagnose X'008'
00009C24  4780 C2AA                     00009C2A  1771          BZ    MSGRET                  Return if successful
00009C28  0000                                    1772          DC    H'0'                    CRASH for debugging purposes

00009C2A  9802 C2B0                     00009C30  1774 MSGRET   LM    R0,R2,MSGSAVE           Restore registers
00009C2E  07F2                                    1775          BR    R2                      Return to caller




00009C30  00000000 00000000                       1777 MSGSAVE  DC    3F'0'                   Registers save area
00009C3C  D200 C2CB 1000      00009C4B  00000000  1778 MSGMVC   MVC   MSGMSG(0),0(R1)         Executed instruction


00009C42  D4E2C7D5 D6C8405C                       1780 MSGCMD   DC    C'MSGNOH * '            *** HERCULES MESSAGE COMMAND ***
00009C4B  40404040 40404040                       1781 MSGMSG   DC    CL95' '                 The message text to be displayed
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1783 ***********************************************************************
                                                  1784 *                         VERIFY TABLE
                                                  1785 ***********************************************************************
                                                  1786 *
                                                  1787 *        A(actual results), A(expected results), A(#of results)
                                                  1788 *
                                                  1789 ***********************************************************************

00009CAC                                          1791 VERIFTAB DC    0F'0'
00009CAC  00001000                                1792          DC    A(SINTOUT)
00009CB0  00005000                                1793          DC    A(SINTOUT_GOOD)
00009CB4  00000005                                1794          DC    A(SINTOUT_NUM)
                                                  1795 *
00009CB8  00001100                                1796          DC    A(SINTFLGS)
00009CBC  00005140                                1797          DC    A(SINTFLGS_GOOD)
00009CC0  00000005                                1798          DC    A(SINTFLGS_NUM)
                                                  1799 *
00009CC4  00001200                                1800          DC    A(SINTRMO)
00009CC8  00005280                                1801          DC    A(SINTRMO_GOOD)
00009CCC  0000002A                                1802          DC    A(SINTRMO_NUM)
                                                  1803 *
00009CD0  00001600                                1804          DC    A(SINTRMOF)
00009CD4  00005D00                                1805          DC    A(SINTRMOF_GOOD)
00009CD8  0000002A                                1806          DC    A(SINTRMOF_NUM)
                                                  1807 *
00009CDC  00002000                                1808          DC    A(LINTOUT)
00009CE0  00006780                                1809          DC    A(LINTOUT_GOOD)
00009CE4  00000005                                1810          DC    A(LINTOUT_NUM)
                                                  1811 *
00009CE8  00002100                                1812          DC    A(LINTFLGS)
00009CEC  000068C0                                1813          DC    A(LINTFLGS_GOOD)
00009CF0  00000005                                1814          DC    A(LINTFLGS_NUM)
                                                  1815 *
00009CF4  00002200                                1816          DC    A(LINTRMO)
00009CF8  00006A00                                1817          DC    A(LINTRMO_GOOD)
00009CFC  0000002D                                1818          DC    A(LINTRMO_NUM)
                                                  1819 *
00009D00  00002600                                1820          DC    A(LINTRMOF)
00009D04  00007540                                1821          DC    A(LINTRMOF_GOOD)
00009D08  0000002D                                1822          DC    A(LINTRMOF_NUM)
                                                  1823 *
00009D0C  00003000                                1824          DC    A(XINTOUT)
00009D10  00008080                                1825          DC    A(XINTOUT_GOOD)
00009D14  00000005                                1826          DC    A(XINTOUT_NUM)
                                                  1827 *
00009D18  00003100                                1828          DC    A(XINTFLGS)
00009D1C  000081C0                                1829          DC    A(XINTFLGS_GOOD)
00009D20  00000005                                1830          DC    A(XINTFLGS_NUM)
                                                  1831 *
00009D24  00003200                                1832          DC    A(XINTRMO)
00009D28  00008300                                1833          DC    A(XINTRMO_GOOD)
00009D2C  0000002D                                1834          DC    A(XINTRMO_NUM)
                                                  1835 *
00009D30  00003600                                1836          DC    A(XINTRMOF)
00009D34  00008E40                                1837          DC    A(XINTRMOF_GOOD)
00009D38  0000002D                                1838          DC    A(XINTRMOF_NUM)
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1839 *
                              0000000C  00000001  1840 VERIFLEN EQU   (*-VERIFTAB)/12    #of entries in verify table
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00009D3C                                          1842          END
00009D3C  0000                                    1843                =H'0'
00009D3E  005F                                    1844                =AL2(L'MSGMSG)
00009D40  E68195A3 7A40                           1845                =CL6'Want: '
00009D46  C796A37A 4040                           1846                =CL6'Got:  '
```

```
     SYMBOL        TYPE  VALUE   LENGTH  DEFN  REFERENCES

AACTUAL             F    009BCC       4  1744  1671  1706
AEXPECT             F    009BC8       4  1743  1673  1678
AHELPERS            A    00027C       4   198   188   235
BFPCVTTF            J    000000   40268   115   165   168   170   173   181   911   913   915   917   920   922   924   926   929
                                                931   933   935   942
BLANKEQ             C    009B9E       3  1740  1679  1707
CFDBR               I    000504       4   462   220
CFDBRA              I    000562       4   512   222
CFEBR               I    00035C       4   300   213
CFEBRA              I    0003BA       4   351   215
CFXBR               I    0006AC       4   623   227
CFXBRA              I    00070E       4   674   229
CHARHEX             C    009BE8      16  1746  1747
CTLR0               F    0002F0       4   245   206   207   208
EXTDS               F    00031C       4   267   226
FAIL                I    000238       4   196  1639
FAILADR             C    009B96       8  1739  1678  1680  1706  1708
FAILDESC            C    009B60      48  1735  1664
FAILFLAG            X    009BF8       1  1748  1637  1660
FAILMSG1            C    009B4C      68  1733  1665  1666
FAILMSG2            C    009B90      53  1737  1698  1699  1726  1727
FAILPSW             X    0002E0       8   243   196
FAILVALS            C    009BA1      36  1741  1682  1683  1684  1686  1687  1688  1690  1691  1692  1694  1695  1696  1710  1711
                                               1712  1714  1715  1716  1718  1719  1720  1722  1723  1724
FPCREGNT            X    0002F4       4   246   307   361   370   379   388   399   407   415   423   431   439   469   522   531
                                                540   549   560   568   576   584   592   600   631   685   694   703   712   723
                                                731   739   747   755   763
FPCREGTR            X    0002F8       4   247   315   477   639
FPR0                U    000000       1   135
FPR1                U    000001       1   136
FPR10               U    00000A       1   145   630
FPR11               U    00000B       1   146
FPR12               U    00000C       1   147
FPR13               U    00000D       1   148
FPR14               U    00000E       1   149
FPR15               U    00000F       1   150
FPR2                U    000002       1   137
FPR3                U    000003       1   138
FPR4                U    000004       1   139
FPR5                U    000005       1   140
FPR6                U    000006       1   141
FPR7                U    000007       1   142
FPR8                U    000008       1   143   306   308   318   357   363   372   381   390   400   408   416   424   432   440
                                                468   470   480   518   524   533   542   551   561   569   577   585   593   601
                                                629   632   642   680   687   696   705   714   724   732   740   748   756   764
FPR9                U    000009       1   144
GOODPSW             X    0002D0       8   242   239
HELPERS             H    009980       2  1579   153   198
HEXTRTAB            U    009AF8      16  1747  1588  1592  1596  1600  1604  1680  1684  1688  1692  1696  1708  1712  1716  1720
                                               1724
IMAGE               1    000000   40268     0
LBFPCT              U    000050       1   843   262
LBFPIN              F    0008B8       4   830   843   263
LBFPINRM            F    000908       4   847   864   281
LBFPRMCT            U    000078       1   864   280
LINTFLGS            U    002100       0   922   265  1812
```

```
     SYMBOL        TYPE  VALUE   LENGTH  DEFN  REFERENCES

LINTFLGS_GOOD       U    0068C0       1  1162  1173  1813
LINTFLGS_NUM        U    000005       1  1173  1814
LINTOUT             U    002000       0   920   264  1808
LINTOUT_GOOD        U    006780       1  1148  1159  1809
LINTOUT_NUM         U    000005       1  1159  1810
LINTRMO             U    002200       0   924   282  1816
LINTRMOF            U    002600       0   926   283  1820
LINTRMOF_GOOD       U    007540       1  1270  1361  1821
LINTRMOF_NUM        U    00002D       1  1361  1822
LINTRMO_GOOD        U    006A00       1  1176  1267  1817
LINTRMO_NUM         U    00002D       1  1267  1818
LONGS               F    00030C       4   261   219
MSG                 I    009BFA       4  1754  1608  1667  1700  1728
MSGCMD              C    009C42       9  1780  1767  1768
MSGMSG              C    009C4B      95  1781  1761  1778  1759
MSGMVC              I    009C3C       6  1778  1765
MSGOK               I    009C10       2  1763  1760
MSGRET              I    009C2A       4  1774  1771
MSGSAVE             F    009C30       4  1777  1757  1774
PCINTCD             H    00008E       2   166   183  1586
PCNOTDTA            I    00020C       4   187   184
PCOLDPSW            U    000150       0   168   185  1590  1594  1598  1602
PGMCK               H    009980       2  1585   189
PGMCOMMA            C    0099F6       1  1615  1587
PGMPSW              C    0099FC      36  1617  1590  1591  1592  1594  1595  1596  1598  1599  1600  1602  1603  1604
PROGCHK             H    000200       2   182   174
PROGCODE            C    0099F2       4  1614  1586  1588
PROGMSG             C    0099DE      66  1612  1606  1607
PROGPSW             D    000228       8   195   194
R0                  U    000000       1   116   187   190   206   208  1606  1659  1665  1698  1726  1730  1754  1757  1759  1761
                                                1763  1774
R1                  U    000001       1   117   308   309   311   312   313   316   317   318   319   321   322   323   363   364
                                                 366   367   368   372   373   375   376   377   381   382   384   385   386   390
                                                 391   393   394   395   400   401   403   404   405   408   409   411   412   413
                                                 416   417   419   420   421   424   425   427   428   429   432   433   435   436
                                                 437   440   441   443   444   445   470   471   473   474   475   478   479   480
                                                 481   483   484   485   524   525   527   528   529   533   534   536   537   538
                                                 542   543   545   546   547   551   552   554   555   556   561   562   564   565
                                                 566   569   570   572   573   574   577   578   580   581   582   585   586   588
                                                 589   590   593   594   596   597   598   601   602   604   605   606   632   633
                                                 635   636   637   640   641   642   643   645   646   647   687   688   690   691
                                                 692   696   697   699   700   701   705   706   708   709   710   714   715   717
                                                 718   719   724   725   727   728   729   732   733   735   736   737   740   741
                                                 743   744   745   748   749   751   752   753   756   757   759   760   761   764
                                                 765   767   768   769  1607  1628  1632  1634  1666  1699  1727  1768  1778
R10                 U    00000A       1   126   212   214   219   221   226   228   300   301   351   352   462   463   512   513
                                                 623   624   674   675
R11                 U    00000B       1   127
R12                 U    00000C       1   128   153   188   235   304   328   355   450   466   490   516   611   627   652   678
                                                 774
R13                 U    00000D       1   129   189   213   215   220   222   227   229   236   303   329   354   451   465   491
                                                 515   612   626   653   677   775  1610  1638
R14                 U    00000E       1   130   192   193   237   238
R15                 U    00000F       1   131   152   187   190
R2                  U    000002       1   118   300   302   328   351   353   450   462   464   490   512   514   611   623   625
                                                 652   674   676   681   774  1608  1629  1635  1667  1700  1728  1755  1757  1763
     SYMBOL        TYPE  VALUE   LENGTH  DEFN  REFERENCES

                                               1764  1765  1767  1774  1775
R3                 U     000003        1   119  300   306   325   351   357   447   462   468   487   512   518   608   623   629
                                                630   649   674   680   681   771  1630  1635
R4                 U     000004        1   120 1632  1647  1649  1671  1710  1714  1718  1722
R5                 U     000005        1   121 1647  1650  1659  1664  1672  1673  1682  1686  1690  1694  1730
R6                 U     000006        1   122 1632  1651
R7                 U     000007        1   123  301   309   319   326   352   364   373   382   391   401   409   417   425   433
                                                441   448   463   471   481   488   513   525   534   543   552   562   570   578
                                                586   594   602   609   624   633   643   650   675   688   697   706   715   725
                                                733   741   749   757   765   772  1633  1653
R8                 U     000008        1   124  301   310   313   320   323   327   352   365   368   374   377   383   386   392
                                                395   402   405   410   413   418   421   426   429   434   437   442   445   449
                                                463   472   475   482   485   489   513   526   529   535   538   544   547   553
                                                556   563   566   571   574   579   582   587   590   595   598   603   606   610
                                                624   634   637   644   647   651   675   689   692   698   701   707   710   716
                                                719   726   729   734   737   742   745   750   753   758   761   766   769   773
                                               1645  1651
R9                 U     000009        1   125
RMEXTDS            F     00034C        4   285  228
RMLONGS            F     00033C        4   279  221
RMSHORTS           F     00032C        4   273  214
SAVER0R5           F     009BD0        4  1745 1659  1730
SAVEREGS           F     00023C        4   197  187   190
SBFPCT             U     000024        1   802  256
SBFPIN             F     00085C        4   789  802   257
SBFPINRM           F     000880        4   806  826   275
SBFPRMCT           U     000038        1   826  274
SHORTS             F     0002FC        4   255  212
SINTFLGS           U     001100        0   913  259  1796
SINTFLGS_GOOD      U     005140        1   958  969  1797
SINTFLGS_NUM       U     000005        1   969 1798
SINTOUT            U     001000        0   911  258  1792
SINTOUT_GOOD       U     005000        1   944  955  1793
SINTOUT_NUM        U     000005        1   955 1794
SINTRMO            U     001200        0   915  276  1800
SINTRMOF           U     001600        0   917  277  1804
SINTRMOF_GOOD      U     005D00        1  1060 1145  1805
SINTRMOF_NUM       U     00002A        1  1145 1806
SINTRMO_GOOD       U     005280        1   972 1057  1801
SINTRMO_NUM        U     00002A        1  1057 1802
START              I     000280        4   206  171
VERIFAIL           I     009A5A        4  1659 1648
VERIFLEN           U     00000C        1  1840 1629
VERIFTAB           F     009CAC        4  1791 1840  1628
VERIFY             I     009A42        2  1645 1633
VERINEXT           I     009A4E        4  1649 1731
VERISUB            H     009A20        2  1623  236
WANTGOT            C     009B90        6  1738 1677  1705
XBFPCT             U     0000A0        1   881  268
XBFPIN             D     000980        8   868  881   269
XBFPINRM           D     000A20        8   885  902   287
XBFPRMCT           U     0000F0        1   902  286
XINTFLGS           U     003100        0   931  271  1828
XINTFLGS_GOOD      U     0081C0        1  1378 1389  1829
XINTFLGS_NUM       U     000005        1  1389 1830
XINTOUT            U     003000        0   929  270  1824
```

```
     SYMBOL        TYPE  VALUE   LENGTH  DEFN  REFERENCES

XINTOUT_GOOD         U    008080       1  1364  1375  1825
XINTOUT_NUM          U    000005       1  1375  1826
XINTRMO              U    003200       0   933   288  1832
XINTRMOF             U    003600       0   935   289  1836
XINTRMOF_GOOD        U    008E40       1  1486  1577  1837
XINTRMOF_NUM         U    00002D       1  1577  1838
XINTRMO_GOOD         U    008300       1  1392  1483  1833
XINTRMO_NUM          U    00002D       1  1483  1834
=AL2(L'MSGMSG)       R    009D3E       2  1844  1759
=CL6'Got:  '         C    009D46       6  1846  1705
=CL6'Want: '         C    009D40       6  1845  1677
=H'0'                H    009D3C       2  1843  1754
```

```
MACRO  DEFN  REFERENCES

No defined macros
```

```
   DESC     SYMBOL    SIZE     POS        ADDR

Entry: 0

Image      IMAGE     40268  0000-9D4B  0000-9D4B
  Region             40268  0000-9D4B  0000-9D4B
    CSECT  BFPCVTTF  40268  0000-9D4B  0000-9D4B
   STMT                                                       FILE NAME

1     c:\Users\Fish\Documents\Visual Studio 2008\Projects\MyProjects\ASMA-0\bfp-006-cvttofix\bfp-006-cvttofix.asm

** NO ERRORS FOUND **
```