```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                      2 ***********************************************************************
                                                      3 *
                                                      4 *              CU14 instruction tests
                                                      5 *
                                                      6 *        NOTE: This test is based the CLCL-et-al Test
                                                      7 *              modified to only test the Performance
                                                      8 *              of the CU14 instruction. The default is NOT to
                                                      9 *              run the performance test. CU14-02-performance.tst
                                                     10 *              must be modified to enable the test.
                                                     11
                                                     12 *              The MSG routine is from the Hercules Binary
                                                     13 *              Floating Point Validation Package by Stephen R. Orso
                                                     14
                                                     15 *                         ********************
                                                     16 *                         **   IMPORTANT!   **
                                                     17 *                         ********************
                                                     18 *
                                                     19 *              This test uses the Hercules Diagnose X'008' interface
                                                     20 *              to display messages and thus your .tst runtest script
                                                     21 *              MUST contain a "DIAG8CMD ENABLE" statement within it!
                                                     22 *
                                                     23 *        James Wekel February 2024
                                                     24 ***********************************************************************

                                                     26 ***********************************************************************
                                                     27 *
                                                     28 *            CU14 Performance instruction tests
                                                     29 *
                                                     30 ***********************************************************************
                                                     31 *
                                                     32 *  This program ONLY tests the performance of the CU14
                                                     33 *  instructions.
                                                     34 *
                                                     35 *     Tests:
                                                     36 *
                                                     37 *           All tests are 'CU14  R0,R2'
                                                     38 *
                                                     39 *           1. CU14 with CC=0 - no crossed pages
                                                     40 *                               source: 61 bytes (28 UTF8 Chars)
                                                     41 *
                                                     42 *           2. CU14 with CC=0 - source cross page
                                                     43 *                               source: 61 bytes (28 UTF8 Chars)
                                                     44 *
                                                     45 *           3. CU14 with CC=0 - target cross page
                                                     46 *                               source: 61 bytes (28 UTF8 Chars)
                                                     47 *
                                                     48 *           4. CU14 with CC=0 - both arguments crossed pages
                                                     49 *                               source: 61 bytes (28 UTF8 Chars)
                                                     50 *
                                                     51 *           5. CU14 with CC=3 - both arguments crossed pages
                                                     52 *                               source: 13,738 bytes only 4095+
                                                     53 *                               processed
                                                     54 *
                                                     55 ***********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     57 ***********************************************************************
                                                     58 *
                                                     59 *  Example Hercules Testcase:
                                                     60 *
                                                     61 *
                                                     62 *        *Testcase CU14-02-performance (Test CU14 instructions)
                                                     63 *        mainsize   16
                                                     64 *        numcpu     1
                                                     65 *        sysclear
                                                     66 *        archlvl    z/Arch
                                                     67 *
                                                     68 *        loadcore   "$(testpath)/CU14-02-performance.core" 0x0
                                                     69 *
                                                     70 *        diag8cmd   enable  # (needed for messages to Hercules console)
                                                     71 *        #r         408=ff  # (enable timing tests)
                                                     72 *        runtest    300     # (test duration, depends on host)
                                                     73 *        diag8cmd   disable # (reset back to default)
                                                     74 *
                                                     75 *        *Done
                                                     76 *
                                                     77 *
                                                     78 ***********************************************************************



                                                     80 ***********************************************************************
                                                     81 *              Low Core Definitions
                                                     82 ***********************************************************************
                                                     83 *
                              00000000  00000D17     84 CU142TST START 0
00000000                      00000000               85          USING CU142TST,R0            Low core addressability


00000000                      00000000  000001A0     87          ORG   CU142TST+X'1A0'        z/Architecure RESTART PSW
000001A0  00000001 80000000                          88          DC    X'0000000180000000'
000001A8  00000000 00000200                          89          DC    AD(BEGIN)


000001B0                      000001B0  000001D0     91          ORG   CU142TST+X'1D0'        z/Architecure PROGRAM CHECK PSW
000001D0  00020001 80000000                          92          DC    X'0002000180000000'
000001D8  00000000 0000DEAD                          93          DC    AD(X'DEAD')


000001E0                      000001E0  00000200     95          ORG   CU142TST+X'200'        Start of actual test program...
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     97 ***********************************************************************
                                                     98 *               The actual "CU142TST" program itself...
                                                     99 ***********************************************************************
                                                    100 *
                                                    101 *  Architecture Mode: 370
                                                    102 *  Register Usage:
                                                    103 *
                                                    104 *   R0       (work)
                                                    105 *   R1       (work)
                                                    106 *   R2       (work) or MSG subroutine call
                                                    107 *   R3       (work)
                                                    108 *   R4       (work)
                                                    109 *   R5-R7    (work)
                                                    110 *   R8       First base register
                                                    111 *   R9       Second base register
                                                    112 *   R10-R12  (work)
                                                    113 *   R13      CU14TEST Base (of current test)
                                                    114 *   R14      Subroutine call
                                                    115 *   R15      Secondary Subroutine call or work
                                                    116 *
                                                    117 ***********************************************************************

00000200                      00000200              119          USING  BEGIN,R8        FIRST Base Register
00000200                      00001200              120          USING  BEGIN+4096,R9    SECOND Base Register

00000200  0580                                      122 BEGIN    BALR  R8,0             Initalize FIRST base register
00000202  0680                                      123          BCTR  R8,0             Initalize FIRST base register
00000204  0680                                      124          BCTR  R8,0             Initalize FIRST base register

00000206  4190 8800                     00000800    126          LA    R9,2048(,R8)     Initalize SECOND base register
0000020A  4190 9800                     00000800    127          LA    R9,2048(,R9)     Initalize SECOND base register

                                                    129 *
                                                    130 **       Run the performance tests...
                                                    131 *
0000020E  45E0 8328                     00000528    132          BAL   R14,TEST91       Time CU14   instruction  (speed test)
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    134 ***********************************************************************
                                                    135 *         Test for normal or unexpected test completion...
                                                    136 ***********************************************************************

00000212  95FF 8208                     00000408    138          CLI   TIMEOPT,X'FF'    Was this a timing run?
00000216  4770 8A58                     00000C58    139          BNE   EOJ              No, timing run; just go end normally

0000021A  9505 8200                     00000400    141          CLI   TESTNUM,X'05'    Did we end on expected test?
0000021E  4770 8A70                     00000C70    142          BNE   FAILTEST         No?! Then FAIL the test!

00000222  9599 8201                     00000401    144          CLI   SUBTEST,X'99'    Did we end on expected SUB-test?
00000226  4770 8A70                     00000C70    145          BNE   FAILTEST         No?! Then FAIL the test!

0000022A  47F0 8A58                     00000C58    147          B     EOJ              Yes, then normal completion!



                                                    149 ***********************************************************************
                                                    150 *        Fixed test storage locations ...
                                                    151 ***********************************************************************


0000022E                      0000022E  00000400    153          ORG   BEGIN+X'200'
                                                    154
00000400                                            155 TESTADDR DS    0D                Where test/subtest numbers will go
00000400  99                                        156 TESTNUM  DC    X'99'      Test number of active test
00000401  99                                        157 SUBTEST  DC    X'99'      Active test sub-test number


00000408                                            159          DS    0D
00000408  00                                        160 TIMEOPT  DC    X'00'      Set to non-zero to run timing tests


00000410                                            162          DS    0D
00000410  00000000 00000000                         163 SAVE3T5  DC    4F'0'
00000420  00000000                                  164 SAVER2   DC    F'0'
00000424  00000000                                  165 SAVER13  DC    F'0'


00000428                      00000428  00000528    167          ORG   *+X'100'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    169 ***********************************************************************
                                                    170 *     Define come helpful macros to ensure our counts are correct
                                                    171 ***********************************************************************


                                                    173          MACRO
                                                    174          OVERONLY &NUM               &NUM = number of sets
                                                    175          LCLA  &CTR
                                                    176 &CTR     SETA  &NUM
                                                    177 .LOOP    ANOP
                                                    178 .*
                                                    179 *
                                                    180          LM    R0,R3,OPSPERF         Get CU14 operands
                                                    181 .*
                                                    182 &CTR     SETA  &CTR-1
                                                    183          AIF   (&CTR GT 0).LOOP
                                                    184          MEND


                                                    186          MACRO
                                                    187          DOINSTR &NUM                &NUM = number of sets
                                                    188          LCLA  &CTR
                                                    189 &CTR     SETA  &NUM
                                                    190 .LOOP    ANOP
                                                    191 .*
                                                    192 *
                                                    193          LM    R0,R3,OPSPERF         Load CU14 operands
                                                    194          CU14  R0,R2                 Do CU14
                                                    195 .*
                                                    196 &CTR     SETA  &CTR-1
                                                    197          AIF   (&CTR GT 0).LOOP
                                                    198          MEND
  LOC        OBJECT CODE        ADDR1     ADDR2     STMT

                                                    200 ***********************************************************************
                                                    201 *        TEST91                 Time CU14 instruction  (speed test)
                                                    202 ***********************************************************************

00000528  91FF 8208                     00000408    204 TEST91   TM    TIMEOPT,X'FF'    Is timing tests option enabled?
0000052C  078E                                      205          BZR   R14              No, skip timing tests

0000052E  41D0 8B18                     00000D18    207          LA    R13,CU14CTL          Point R13 --> testing control table
00000532                      00000000              208          USING CU14PERF,R13         What each table entry looks like
                                                    209 *
                              00000532  00000001    210 TST91LOP EQU   *
00000532  50D0 8224                     00000424    211          ST    R13,SAVER13          save current pref table base
                                                    212 *
00000536  4360 D000                     00000000    213          IC    R6,TNUM            Set test number
0000053A  4260 8200                     00000400    214          STC   R6,TESTNUM
                                                    215 *
                                                    216 **       Initialize operand data  (move source to testing address)
                                                    217 *
                                                    218 *                                   Source
0000053E  5800 D018                     00000018    219          L     R0,OP2WHERE         Where to move operand-2 data to
00000542  5810 D010                     00000010    220          L     R1,OP2LEN             How much of it there is
00000546  5820 D00C                     0000000C    221          L     R2,OP2DATA          Where op2 data is right now
0000054A  5830 D010                     00000010    222          L     R3,OP2LEN             How much of it there is
0000054E  0E02                                      223          MVCL  R0,R2
                                                    224
                                                    225 *                                  set up CU14 args
00000550  5800 D014                     00000014    226          L     R0,OP1WHERE         Where to move operand-1 data to
00000554  5810 D008                     00000008    227          L     R1,OP1LEN           operand-1 length
00000558  5820 D018                     00000018    228          L     R2,OP2WHERE         Where to move operand-2 data to
0000055C  5830 D010                     00000010    229          L     R3,OP2LEN           operand-2 length
00000560  9003 8868                     00000A68    230          STM   R0,R3,OPSPERF      save for each CU14 execution
                                                    231 *
  LOC        OBJECT CODE       ADDR1     ADDR2      STMT

                                                    233 ***********************************************************************
                                                    234 *        Next, time the overhead...
                                                    235 ***********************************************************************

00000564  5870 8A8C                     00000C8C    237          L     R7,NUMLOOPS
00000568  B205 8A90                     00000C90    238          STCK  BEGCLOCK
0000056C  9035 8210                     00000410    239          STM   R3,R5,SAVE3T5
00000570  0560                                      240          BALR  R6,0
                                                    241 *                                  100 sets of overhead
                                                    242          OVERONLY 2                (first 2)
                                                    243+*
00000572  9803 8868                     00000A68    244+         LM    R0,R3,OPSPERF        Get CU14 operands
                                                    245+*
00000576  9803 8868                     00000A68    246+         LM    R0,R3,OPSPERF        Get CU14 operands

                                                    248 *        .........ETC.........

                                                    250          PRINT OFF
                                                    444          PRINT ON

                                                    446          OVERONLY 2                (last 2)
                                                    447+*
000006FA  9803 8868                     00000A68    448+         LM    R0,R3,OPSPERF        Get CU14 operands
                                                    449+*
000006FE  9803 8868                     00000A68    450+         LM    R0,R3,OPSPERF        Get CU14 operands
                                                    451 *
00000702  0676                                      452          BCTR  R7,R6
00000704  B205 8A98                     00000C98    453          STCK  ENDCLOCK
00000708  45F0 8908                     00000B08    454          BAL   R15,CALCDUR
0000070C  D207 8AA8 8AA0      00000CA8  00000CA0    455          MVC   OVERHEAD,DURATION
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    457 ***********************************************************************
                                                    458 *        Now do the actual timing run...
                                                    459 ***********************************************************************

00000712  5870 8A8C                     00000C8C    461          L     R7,NUMLOOPS
00000716  B205 8A90                     00000C90    462          STCK  BEGCLOCK
0000071A  0560                                      463          BALR  R6,0
                                                    464 *                                  100 sets of instructions
                                                    465          DOINSTR 2                 (first 2)
                                                    466+*
0000071C  9803 8868                     00000A68    467+         LM    R0,R3,OPSPERF        Load CU14 operands
00000720  B9B0 0002                                 468+         CU14  R0,R2                Do CU14
                                                    469+*
00000724  9803 8868                     00000A68    470+         LM    R0,R3,OPSPERF        Load CU14 operands
00000728  B9B0 0002                                 471+         CU14  R0,R2                Do CU14

                                                    473 *        .........ETC.........

                                                    475          PRINT OFF
                                                    765          PRINT ON

                                                    767          DOINSTR 2                 (last 2)
                                                    768+*
00000A2C  9803 8868                     00000A68    769+         LM    R0,R3,OPSPERF        Load CU14 operands
00000A30  B9B0 0002                                 770+         CU14  R0,R2                Do CU14
                                                    771+*
00000A34  9803 8868                     00000A68    772+         LM    R0,R3,OPSPERF        Load CU14 operands
00000A38  B9B0 0002                                 773+         CU14  R0,R2                Do CU14


00000A3C  0676                                      775          BCTR  R7,R6
00000A3E  B205 8A98                     00000C98    776          STCK  ENDCLOCK


00000A42  9835 8210                     00000410    778          LM    R3,R5,SAVE3T5
00000A46  D204 8AE9 8A80      00000CE9  00000C80    779          MVC   PRTLINE+33(5),=CL5'CU14'
00000A4C  45F0 8888                     00000A88    780          BAL   R15,RPTSPEED
                                                    781 *
                                                    782 *  more performance tests?
                                                    783 *
00000A50  58D0 8224                     00000424    784          L     R13,SAVER13          restore perf table base
00000A54  41D0 D028                     00000028    785          LA    R13,CU14NEXT        Go on to next table entry
00000A58  D503 8A74 D000      00000C74  00000000    786          CLC   =F'0',0(R13)        End of table?
00000A5E  4770 8332                     00000532    787          BNE   TST91LOP           No, loop...
00000A62  07FE                                      788          BR    R14                Return to caller or FAILTEST


00000A68  00000000 00000000                         790 OPSPERF  DS    4D                 Performance test R0-R3
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    792 ***********************************************************************
                                                    793 *        RPTSPEED                 Report instruction speed
                                                    794 ***********************************************************************

00000A88  50F0 88F0                     00000AF0    796 RPTSPEED ST    R15,RPTSAVE        Save return address
00000A8C  5050 88F4                     00000AF4    797          ST    R5,RPTSVR5         Save R5
                                                    798 *
00000A90  45F0 8908                     00000B08    799          BAL   R15,CALCDUR        Calculate duration
                                                    800 *
00000A94  4150 8AA8                     00000CA8    801          LA    R5,OVERHEAD        Subtract overhead
00000A98  4160 8AA0                     00000CA0    802          LA    R6,DURATION        From raw timing
00000A9C  4170 8AA0                     00000CA0    803          LA    R7,DURATION        Yielding true instruction timing
00000AA0  45F0 895C                     00000B5C    804          BAL   R15,SUBDWORD       Do it
                                                    805 *
00000AA4  98AB 8AA0                     00000CA0    806          LM    R10,R11,DURATION   Convert to...
00000AA8  8CA0 000C                     0000000C    807          SRDL  R10,12             ... microseconds
                                                    808 *
00000AAC  4EA0 8AB0                     00000CB0    809          CVD   R10,TICKSAAA       convert HIGH part to decimal
00000AB0  4EB0 8AB8                     00000CB8    810          CVD   R11,TICKSBBB       convert LOW  part to decimal
                                                    811 *
00000AB4  F877 8AC0 8AB0      00000CC0  00000CB0    812          ZAP   TICKSTOT,TICKSAAA            Calculate...
00000ABA  FC75 8AC0 8A85      00000CC0  00000C85    813          MP    TICKSTOT,=P'4294967296'      ...decimal...
00000AC0  FA77 8AC0 8AB8      00000CC0  00000CB8    814          AP    TICKSTOT,TICKSBBB            ...microseconds
                                                    815 *
00000AC6  D20B 8AF3 8B0C      00000CF3  00000D0C    816          MVC   PRTLINE+43(L'EDIT),EDIT          (edit into...
00000ACC  DE0B 8AF3 8AC3      00000CF3  00000CC3    817          ED    PRTLINE+43(L'EDIT),TICKSTOT+3     ...print line)



                                                    819 *
                                                    820 *        Use Hercules Diagnose for Message to console
                                                    821 *
00000AD2  9002 88F8                     00000AF8    822          STM   R0,R2,RPTDWSAV       save regs used by MSG
00000AD6  4100 0044                     00000044    823          LA    R0,PRTLNG            message length
00000ADA  4110 8AC8                     00000CC8    824          LA    R1,PRTLINE           messagfe address
00000ADE  4520 8990                     00000B90    825          BAL   R2,MSG               call Hercules console MSG display
00000AE2  9802 88F8                     00000AF8    826          LM    R0,R2,RPTDWSAV       restore regs

00000AE6  5850 88F4                     00000AF4    828          L     R5,RPTSVR5         Restore R5
00000AEA  58F0 88F0                     00000AF0    829          L     R15,RPTSAVE        Restore return address
00000AEE  07FF                                      830          BR    R15                Return to caller

00000AF0  00000000                                  832 RPTSAVE  DC    F'0'               R15 save area
00000AF4  00000000                                  833 RPTSVR5  DC    F'0'               R5 save area

00000AF8  00000000 00000000                         835 RPTDWSAV DC    2D'0'              R0-R2 save area for MSG call
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    837 ***********************************************************************
                                                    838 *        CALCDUR                  Calculate DURATION
                                                    839 ***********************************************************************

00000B08  50F0 894C                     00000B4C    841 CALCDUR  ST    R15,CALCRET        Save return address
00000B0C  9057 8950                     00000B50    842          STM   R5,R7,CALCWORK     Save work registers
                                                    843 *
00000B10  9867 8A90                     00000C90    844          LM    R6,R7,BEGCLOCK     Remove CPU number from clock value
00000B14  8C60 0006                     00000006    845          SRDL  R6,6                            "
00000B18  8D60 0006                     00000006    846          SLDL  R6,6                            "
00000B1C  9067 8A90                     00000C90    847          STM   R6,R7,BEGCLOCK                  "
                                                    848 *
00000B20  9867 8A98                     00000C98    849          LM    R6,R7,ENDCLOCK     Remove CPU number from clock value
00000B24  8C60 0006                     00000006    850          SRDL  R6,6                            "
00000B28  8D60 0006                     00000006    851          SLDL  R6,6                            "
00000B2C  9067 8A98                     00000C98    852          STM   R6,R7,ENDCLOCK                  "
                                                    853 *
00000B30  4150 8A90                     00000C90    854          LA    R5,BEGCLOCK        Starting time
00000B34  4160 8A98                     00000C98    855          LA    R6,ENDCLOCK        Ending time
00000B38  4170 8AA0                     00000CA0    856          LA    R7,DURATION        Difference
00000B3C  45F0 895C                     00000B5C    857          BAL   R15,SUBDWORD       Calculate duration
                                                    858 *
00000B40  9857 8950                     00000B50    859          LM    R5,R7,CALCWORK     Restore work registers
00000B44  58F0 894C                     00000B4C    860          L     R15,CALCRET        Restore return address
00000B48  07FF                                      861          BR    R15                Return to caller

00000B4C  00000000                                  863 CALCRET  DC    F'0'               R15 save area
00000B50  00000000 00000000                         864 CALCWORK DC    3F'0'              R5-R7 save area




                                                    866 ***********************************************************************
                                                    867 *        SUBDWORD                 Subtract two doublewords
                                                    868 *        R5 --> subtrahend, R6 --> minuend, R7 --> result
                                                    869 ***********************************************************************

00000B5C  9014 8980                     00000B80    871 SUBDWORD STM   R1,R4,SUBDWSAV   Save registers
                                                    872 *
00000B60  9812 5000                     00000000    873          LM    R1,R2,0(R5)        Subtrahend  (value to subtract)
00000B64  9834 6000                     00000000    874          LM    R3,R4,0(R6)        Minuend     (what to subtract FROM)
00000B68  1F42                                      875          SLR   R4,R2              Subtract LOW part
00000B6A  47B0 8972                     00000B72    876          BNM   *+4+4                (branch if no borrow)
00000B6E  5F30 8A78                     00000C78    877          SL    R3,=F'1'             (otherwise do borrow)
00000B72  1F31                                      878          SLR   R3,R1              Subtract HIGH part
00000B74  9034 7000                     00000000    879          STM   R3,R4,0(R7)        Store results
                                                    880 *
00000B78  9814 8980                     00000B80    881          LM    R1,R4,SUBDWSAV     Restore registers
00000B7C  07FF                                      882          BR    R15                Return to caller

00000B80  00000000 00000000                         884 SUBDWSAV DC    2D'0'              R1-R4 save area
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    886 ***********************************************************************
                                                    887 *        Issue HERCULES MESSAGE pointed to by R1, length in R0
                                                    888 *              R2 = return address
                                                    889 ***********************************************************************

00000B90  4900 8A7C                     00000C7C    891 MSG      CH    R0,=H'0'               Do we even HAVE a message?
00000B94  07D2                                      892          BNHR  R2                     No, ignore

00000B96  9002 89C8                     00000BC8    894          STM   R0,R2,MSGSAVE          Save registers

00000B9A  4900 8A7E                     00000C7E    896          CH    R0,=AL2(L'MSGMSG)      Message length within limits?
00000B9E  47D0 89A6                     00000BA6    897          BNH   MSGOK                  Yes, continue
00000BA2  4100 005F                     0000005F    898          LA    R0,L'MSGMSG            No, set to maximum

00000BA6  1820                                      900 MSGOK    LR    R2,R0                  Copy length to work register
00000BA8  0620                                      901          BCTR  R2,0                   Minus-1 for execute
00000BAA  4420 89D4                     00000BD4    902          EX    R2,MSGMVC              Copy message to O/P buffer

00000BAE  4120 200A                     0000000A    904          LA    R2,1+L'MSGCMD(,R2)     Calculate true command length
00000BB2  4110 89DA                     00000BDA    905          LA    R1,MSGCMD              Point to true command

00000BB6  83120008                                  907          DC    X'83',X'12',X'0008'    Issue Hercules Diagnose X'008'
00000BBA  4780 89C0                     00000BC0    908          BZ    MSGRET                 Return if successful
00000BBE  0000                                      909          DC    H'0'                   CRASH for debugging purposes

00000BC0  9802 89C8                     00000BC8    911 MSGRET   LM    R0,R2,MSGSAVE          Restore registers
00000BC4  07F2                                      912          BR    R2                     Return to caller





00000BC8  00000000 00000000                         914 MSGSAVE  DC    3F'0'                  Registers save area
00000BD4  D200 89E3 1000      00000BE3  00000000    915 MSGMVC   MVC   MSGMSG(0),0(R1)        Executed instruction


00000BDA  D4E2C7D5 D6C8405C                         917 MSGCMD   DC    C'MSGNOH * '           *** HERCULES MESSAGE COMMAND ***
00000BE3  40404040 40404040                         918 MSGMSG   DC    CL95' '                The message text to be displayed
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    920 ***********************************************************************
                                                    921 *        Normal completion or Abnormal termination PSWs
                                                    922 ***********************************************************************



00000C48  00020001 80000000                         924 EOJPSW   DC    0D'0',X'0002000180000000',AD(0)

00000C58  B2B2 8A48                     00000C48    926 EOJ      LPSWE EOJPSW               Normal completion



00000C60  00020001 80000000                         928 FAILPSW  DC    0D'0',X'0002000180000000',AD(X'BAD')

00000C70  B2B2 8A60                     00000C60    930 FAILTEST LPSWE FAILPSW              Abnormal termination
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    932 ***********************************************************************
                                                    933 *        Working Storage
                                                    934 ***********************************************************************

00000C74                                            936          LTORG ,                Literals pool
00000C74  00000000                                  937                =F'0'
00000C78  00000001                                  938                =F'1'
00000C7C  0000                                      939                =H'0'
00000C7E  005F                                      940                =AL2(L'MSGMSG)
00000C80  C3E4F1F4 40                               941                =CL5'CU14'
00000C85  04294967 296C                             942                =P'4294967296'

                              00000400  00000001    944 K        EQU   1024             One KB
                              00001000  00000001    945 PAGE     EQU   (4*K)            Size of one page
                              00004000  00000001    946 K16      EQU   (16*K)           16 KB
                              00008000  00000001    947 K32      EQU   (32*K)           32 KB
                              00010000  00000001    948 K64      EQU   (64*K)           64 KB
                              00100000  00000001    949 MB       EQU   (K*K)             1 MB

00000C8C  00002710                                  951 NUMLOOPS DC    F'10000'         10,000 * 100 = 1,000,000

00000C90  BBBBBBBB BBBBBBBB                         953 BEGCLOCK DC    0D'0',8X'BB'     Begin
00000C98  EEEEEEEE EEEEEEEE                         954 ENDCLOCK DC    0D'0',8X'EE'     End
00000CA0  DDDDDDDD DDDDDDDD                         955 DURATION DC    0D'0',8X'DD'     Diff
00000CA8  FFFFFFFF FFFFFFFF                         956 OVERHEAD DC    0D'0',8X'FF'     Overhead

00000CB0  00000000 0000000C                         958 TICKSAAA DC    PL8'0'           Clock ticks high part
00000CB8  00000000 0000000C                         959 TICKSBBB DC    PL8'0'           Clock ticks low part
00000CC0  00000000 0000000C                         960 TICKSTOT DC    PL8'0'           Total clock ticks

00000CC8  40404040 40404040                         962 PRTLINE  DC    C'         1,000,000 iterations of XXXXX'
00000CEE  40A39696 9240F9F9                         963          DC    C' took 999,999,999 microseconds'
                              00000044  00000001    964 PRTLNG   EQU   *-PRTLINE
00000D0C  40202020 6B202020                         965 EDIT     DC    X'402020206B2020206B202120'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT


                                                    969 ***********************************************************************
                                                    970 *        CU14TEST DSECT
                                                    971 ***********************************************************************

                                                    973 CU14PERF DSECT ,
00000000  00                                        974 TNUM     DC    X'00'          CU14 test number
00000001  00                                        975          DC    X'00'
00000002  00                                        976          DC    X'00'
00000003  00                                        977 M3       DC    X'00'          M3 byte stored into CU14 instruction


                                                    979
00000004  00000000                                  980 OP1DATA  DC    A(0)           Pointer to Operand 1       - result
00000008  00000000                                  981 OP1LEN   DC    F'0'             length                   - result
0000000C  00000000                                  982 OP2DATA  DC    A(0)           Pointer to Operand-2 data  - source
00000010  00000000                                  983 OP2LEN   DC    F'0'             length                   - source

                              00000014  00000001    985 OPSWHERE EQU   *
00000014  00000000                                  986 OP1WHERE DC    A(0)           result - Where should be placed
00000018  00000000                                  987 OP2WHERE DC    A(0)           source - Where should be placed

0000001C  00000000                                  989 FAILMASK DC    A(0)           Failure Branch on Condition mask

                                                    991 *                             Ending register values
00000020  00000000                                  992 ENDLN1   DC    A(0)              target length
00000024  00000000                                  993 ENDLN2   DC    A(0)              source length
                                                    994

                              00000028  00000001    996 CU14NEXT EQU   *              Start of next table entry...
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                              00000D18  00004452    999 CU14TST CSECT  ,

                                                   1001 ***********************************************************************
                                                   1002 *        CU14 Testing Control tables   (ref: CU14TEST DSECT)
                                                   1003 ***********************************************************************
                                                   1004          PRINT DATA
00000D18                                           1005 CU14CTL  DC    0A(0)    start of table
                                                   1006 ***********************************************************************
                                                   1007 *        tests with CC=0 M3=0
                                                   1008 ***********************************************************************

00000D18                                           1010 CC0T1    DS    0F
00000D18  01                                       1011          DC    X'01'                       Test Num
00000D19  0000                                     1012          DC    X'00',X'00'
00000D1B  00                                       1013          DC    X'00'                       M3
                                                   1014 *
00000D1C  00000E38 00000070                        1015          DC    A(UTF32A),A(UTF32AED-UTF32A)   target - Op1 & length
00000D24  00000DF0 0000003D                        1016          DC    A(UTF8A),A(UTF8AEND-UTF8A)     Source - Op2 & length
                                                   1017
00000D2C  00400000                                 1018          DC    A(4*MB+(0*K16))                target
00000D30  00200000                                 1019          DC    A(2*MB+(0*K16))                source
                                                   1020 *
00000D34  00000007                                 1021          DC    A(7)                           FailCC - not CC0
00000D38  00000000                                 1022          DC    A(0)                           Result - target len
00000D3C  00000000                                 1023          DC    A(0)                           Result - source len

00000D40                                           1025 CC0T2    DS    0F
00000D40  02                                       1026          DC    X'02'                       Test Num
00000D41  0000                                     1027          DC    X'00',X'00'
00000D43  00                                       1028          DC    X'00'                       M3
                                                   1029 *
00000D44  00000E38 00000070                        1030          DC    A(UTF32A),A(UTF32AED-UTF32A)   target - Op1 & length
00000D4C  00000DF0 0000003D                        1031          DC    A(UTF8A),A(UTF8AEND-UTF8A)     Source - Op2 & length
                                                   1032
00000D54  0010C000                                 1033          DC    A(1*MB+(3*K16))               target
00000D58  00213FE9                                 1034          DC    A(2*MB+(5*K16)-23)            source
                                                   1035 *
00000D5C  00000007                                 1036          DC    A(7)                           FailCC - not CC0
00000D60  00000000                                 1037          DC    A(0)                           Result - target len
00000D64  00000000                                 1038          DC    A(0)                           Result - source len
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

00000D68                                           1040 CC0T3    DS    0F
00000D68  03                                       1041          DC    X'03'                       Test Num
00000D69  0000                                     1042          DC    X'00',X'00'
00000D6B  00                                       1043          DC    X'00'                       M3
                                                   1044 *
00000D6C  00000E38 00000070                        1045          DC    A(UTF32A),A(UTF32AED-UTF32A)   target - Op1 & length
00000D74  00000DF0 0000003D                        1046          DC    A(UTF8A),A(UTF8AEND-UTF8A)     Source - Op2 & length
                                                   1047
00000D7C  0011BFE9                                 1048          DC    A(1*MB+(7*K16)-23)             target
00000D80  00224000                                 1049          DC    A(2*MB+(9*K16))                source
                                                   1050 *
00000D84  00000007                                 1051          DC    A(7)                           FailCC - not CC0
00000D88  00000000                                 1052          DC    A(0)                           Result - target len
00000D8C  00000000                                 1053          DC    A(0)                           Result - source len


00000D90                                           1055 CC0T4    DS    0F
00000D90  04                                       1056          DC    X'04'                       Test Num
00000D91  0000                                     1057          DC    X'00',X'00'
00000D93  00                                       1058          DC    X'00'                       M3
                                                   1059 *
00000D94  00000E38 00000070                        1060          DC    A(UTF32A),A(UTF32AED-UTF32A)   target - Op1 & length
00000D9C  00000DF0 0000003D                        1061          DC    A(UTF8A),A(UTF8AEND-UTF8A)     Source - Op2 & length
                                                   1062
00000DA4  0012BFE9                                 1063          DC    A(1*MB+(11*K16)-23)            target
00000DA8  00233FE9                                 1064          DC    A(2*MB+(13*K16)-23)            source
                                                   1065 *
00000DAC  00000007                                 1066          DC    A(7)                           FailCC - not CC0
00000DB0  00000000                                 1067          DC    A(0)                           Result - target len
00000DB4  00000000                                 1068          DC    A(0)                           Result - source len


00000DB8                                           1070 CC0T5    DS    0F
00000DB8  05                                       1071          DC    X'05'                       Test Num
00000DB9  0000                                     1072          DC    X'00',X'00'
00000DBB  00                                       1073          DC    X'00'                       M3
                                                   1074 *
00000DBC  00000E38 000FC000                        1075          DC    A(UTF32A),A(1*MB-16*K)   target - Op1 & length
00000DC4  00000EA8 000035AB                        1076          DC    A(UTF8B),A(UTF8BEND-UTF8B)     Source - Op2 & length
                                                   1077
00000DCC  005FFF01                                 1078          DC    A(6*MB+(0*K16)-(255))         target
00000DD0  002FFF01                                 1079          DC    A(3*MB+(0*K16)-(255))         source
                                                   1080 *
00000DD4  00000007                                 1081          DC    A(7)                           FailCC - not CC0
00000DD8  00000000                                 1082          DC    A(0)                           Result - target len
00000DDC  00000000                                 1083          DC    A(0)                           Result - source len


00000DE0  00000000                                 1085          DC    A(0)     end of table
00000DE4  00000000                                 1086          DC    A(0)     end of table
00000DE8  00000000                                 1087          DC    A(0)     end of table
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                   1089 ***********************************************************************
                                                   1090 *        CU14  UTF-8 tests
                                                   1091 ***********************************************************************

00000DEC  0000003D                                 1093 UTF8ALN  DC    A(UTF8AEND-UTF8A)
00000DF0                                           1094 UTF8A    DS    0H
00000DF0  00                                       1095          DC    XL1'00'  first UTF-8 1 Byte character
00000DF1  31                                       1096          DC    XL1'31'  1
00000DF2  39                                       1097          DC    XL1'39'  9
00000DF3  40                                       1098          DC    XL1'40'  @
00000DF4  41                                       1099          DC    XL1'41'  A
00000DF5  42                                       1100          DC    XL1'42'  B
00000DF6  7F                                       1101          DC    XL1'7F'  last UTF-8 1 Byte character

00000DF7  C280                                     1103          DC    XL2'C280'  first UTF-8 2 Byte character
00000DF9  C380                                     1104          DC    XL2'C380'  c3 80 LATIN CAPITAL LETTER A WITH GRAVE
00000DFB  C3B8                                     1105          DC    XL2'C3B8'  c3 b8 LATIN SMALL LETTER O WITH STROKE
00000DFD  D09C                                     1106          DC    XL2'D09C'  D0 9C Ðœ Cyrillic Capital Letter Em
00000DFF  DFBF                                     1107          DC    XL2'DFBF'  last UTF-8 2 Byte character DF BF ß¿

00000E01  43                                       1109          DC    XL1'43'  C

00000E02  E0A080                                   1111          DC    XL3'E0A080'  first UTF-8 3 Byte character
                                                   1112 *                            E0 A0 80 à € Samaritan Letter Alaf
00000E05  E0A18D                                   1113          DC    XL3'E0A18D'  E0 A1 8D à¡• Mandaic Letter An
00000E08  EA9FBD                                   1114          DC    XL3'EA9FBD'  EA 9F BD êŸ½ Latin Epigraphic Inverted M
00000E0B  EFBF87                                   1115          DC    XL3'EFbf87'  EF BF 87 ï¿‡ Halfwidth Hangul Letter E
00000E0E  EFBFBF                                   1116          DC    XL3'EFBFBF'  last UTF-8 3 Byte character EF BF BF

00000E11  44                                       1118          DC    XL1'44'  D

00000E12  F0908080                                 1120          DC    XL4'F0908080'  first UTF-8 4 Byte character
                                                   1121 *                            F0 90 80 80 ð•€€ Linear B Syllable B008 A
00000E16  F0908487                                 1122          DC    XL4'F0908487' F0 90 84 87 ð•„‡ Aegean Number One
00000E1A  F09294B5                                 1123          DC    XL4'F09294B5' F0 92 94 B5 Cuneiform Sign She Plus Sar
00000E1E  F09082B8                                 1124          DC    XL4'F09082B8'  F0 90 82 B8 ð•‚¸ Linear B Ideogram B177
00000E22  F096AB83                                 1125          DC    XL4'F096AB83'  F0 96 A8 83 ð–¨ƒ Bamum Letter Phase-f Ka
00000E26  F0989A9F                                 1126          DC    XL4'F0989A9F'  last UTF-8 4 Byte character

00000E2A  45                                       1128          DC    XL1'45'  E
00000E2B  4E                                       1129          DC    XL1'4E'  N
00000E2C  44                                       1130          DC    XL1'44'  D
00000E2D                                           1131 UTF8AEND DS    0X
                                                   1132
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                   1134 ***********************************************************************
                                                   1135 *        CU14  UTF-32 Result
                                                   1136 ***********************************************************************
00000E2D  E4C6E3F3 F27A                            1137          DC    C'UFT32:'
00000E34  00000070                                 1138 UTF32ALN DC    A(UTF32AED-UTF32A)
00000E38                                           1139 UTF32A   DC    0X
00000E38  00000000                                 1140          DC    X'00000000'
00000E3C  00000031                                 1141          DC    X'00000031'
00000E40  00000039                                 1142          DC    X'00000039'
00000E44  00000040                                 1143          DC    X'00000040'
00000E48  00000041                                 1144          DC    X'00000041'
00000E4C  00000042                                 1145          DC    X'00000042'
00000E50  0000007F                                 1146          DC    X'0000007F'
00000E54  00000080                                 1147          DC    X'00000080'
00000E58  000000C0                                 1148          DC    X'000000C0'
00000E5C  000000F8                                 1149          DC    X'000000F8'
00000E60  0000041C                                 1150          DC    X'0000041C'
00000E64  000007FF                                 1151          DC    X'000007FF'
00000E68  00000043                                 1152          DC    X'00000043'
00000E6C  00000800                                 1153          DC    X'00000800'
00000E70  0000084D                                 1154          DC    X'0000084D'
00000E74  0000A7FD                                 1155          DC    X'0000A7FD'
00000E78  0000FFC7                                 1156          DC    X'0000FFC7'
00000E7C  0000FFFF                                 1157          DC    X'0000FFFF'
00000E80  00000044                                 1158          DC    X'00000044'
00000E84  00010000                                 1159          DC    X'00010000'
00000E88  00010107                                 1160          DC    X'00010107'
00000E8C  00012535                                 1161          DC    X'00012535'
00000E90  000100B8                                 1162          DC    X'000100B8'
00000E94  00016AC3                                 1163          DC    X'00016AC3'
00000E98  0001869F                                 1164          DC    X'0001869F'
00000E9C  00000045                                 1165          DC    X'00000045'
00000EA0  0000004E                                 1166          DC    X'0000004E'
00000EA4  00000044                                 1167          DC    X'00000044'
00000EA8                                           1168 UTF32AED DS    0X
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                   1170 ***********************************************************************
                                                   1171 *        UTF-8 LONG LONG LONG String  (CC=3 result)
                                                   1172 ***********************************************************************

00000EA8                                           1174 UTF8B    DS    0F
00000EA8  0A                                       1175          DC    x'0a'
00000EA9  3C                                       1176          DC    x'3c'
00000EAA  21                                       1177          DC    x'21'
00000EAB  44                                       1178          DC    x'44'
00000EAC  4F                                       1179          DC    x'4f'
00000EAD  43                                       1180          DC    x'43'
00000EAE  54                                       1181          DC    x'54'
00000EAF  59                                       1182          DC    x'59'
00000EB0  50                                       1183          DC    x'50'
                                                   1184 *        .........ETC.........

                                                   1186          PRINT OFF
                                                  14906          PRINT ON
                                                  14907 *        .........ETC.........
                                                  14908
00004447  79                                      14909          DC    x'79'
00004448  3E                                      14910          DC    x'3e'
00004449  0A                                      14911          DC    x'0a'
0000444A  3C                                      14912          DC    x'3c'
0000444B  2F                                      14913          DC    x'2f'
0000444C  68                                      14914          DC    x'68'
0000444D  74                                      14915          DC    x'74'
0000444E  6D                                      14916          DC    x'6d'
0000444F  6C                                      14917          DC    x'6c'
00004450  3E                                      14918          DC    x'3e'
00004451  0A                                      14919          DC    x'0a'
00004452  0A                                      14920          DC    x'0a'
00004453                                          14921 UTF8BEND DS    0C
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                  14923 ***********************************************************************
                                                  14924 *        Register equates
                                                  14925 ***********************************************************************


                              00000000  00000001  14927 R0       EQU   0
                              00000001  00000001  14928 R1       EQU   1
                              00000002  00000001  14929 R2       EQU   2
                              00000003  00000001  14930 R3       EQU   3
                              00000004  00000001  14931 R4       EQU   4
                              00000005  00000001  14932 R5       EQU   5
                              00000006  00000001  14933 R6       EQU   6
                              00000007  00000001  14934 R7       EQU   7
                              00000008  00000001  14935 R8       EQU   8
                              00000009  00000001  14936 R9       EQU   9
                              0000000A  00000001  14937 R10      EQU   10
                              0000000B  00000001  14938 R11      EQU   11
                              0000000C  00000001  14939 R12      EQU   12
                              0000000D  00000001  14940 R13      EQU   13
                              0000000E  00000001  14941 R14      EQU   14
                              0000000F  00000001  14942 R15      EQU   15








                                                  14944          END
```

```
     SYMBOL        TYPE  VALUE    LENGTH   DEFN  REFERENCES

BEGCLOCK            D    000C90        8    953    238    462    844    847    854
BEGIN               I    000200        2    122    153     89    119    120
CALCDUR             I    000B08        4    841    454    799
CALCRET             F    000B4C        4    863    841    860
CALCWORK            F    000B50        4    864    842    859
CC0T1               F    000D18        4   1010
CC0T2               F    000D40        4   1025
CC0T3               F    000D68        4   1040
CC0T4               F    000D90        4   1055
CC0T5               F    000DB8        4   1070
CU142TST            J    000000     3352     84     87     91     95     85
CU14CTL             A    000D18        4   1005    207
CU14NEXT            U    000028        1    996    785
CU14PERF            4    000000       40    973    208
CU14TST             J    000D18    14139    999
DURATION            D    000CA0        8    955    455    802    803    806    856
EDIT                X    000D0C       12    965    816    817
ENDCLOCK            D    000C98        8    954    453    776    849    852    855
ENDLN1              A    000020        4    992
ENDLN2              A    000024        4    993
EOJ                 I    000C58        4    926    139    147
EOJPSW              D    000C48        8    924    926
FAILMASK            A    00001C        4    989
FAILPSW             D    000C60        8    928    930
FAILTEST            I    000C70        4    930    142    145
IMAGE               1    000000    17491      0
K                   U    000400        1    944    945    946    947    948    949   1075
K16                 U    004000        1    946   1018   1019   1033   1034   1048   1049   1063   1064   1078   1079
K32                 U    008000        1    947
K64                 U    010000        1    948
M3                  X    000003        1    977
MB                  U    100000        1    949   1018   1019   1033   1034   1048   1049   1063   1064   1075   1078   1079
MSG                 I    000B90        4    891    825
MSGCMD              C    000BDA        9    917    904    905
MSGMSG              C    000BE3       95    918    898    915    896
MSGMVC              I    000BD4        6    915    902
MSGOK               I    000BA6        2    900    897
MSGRET              I    000BC0        4    911    908
MSGSAVE             F    000BC8        4    914    894    911
NUMLOOPS            F    000C8C        4    951    237    461
OP1DATA             A    000004        4    980
OP1LEN              F    000008        4    981    227
OP1WHERE            A    000014        4    986    226
OP2DATA             A    00000C        4    982    221
OP2LEN              F    000010        4    983    220    222    229
OP2WHERE            A    000018        4    987    219    228
OPSPERF             D    000A68        8    790    230    244    246    253    255    257    259    261    263    265    267
                                                   269    271    273    275    277    279    281    283    285    287    289
                                                   291    293    295    297    299    301    303    305    307    309    311
                                                   313    315    317    319    321    323    325    327    329    331    333
                                                   335    337    339    341    343    345    347    349    351    353    355
                                                   357    359    361    363    365    367    369    371    373    375    377
                                                   379    381    383    385    387    389    391    393    395    397    399
                                                   401    403    405    407    409    411    413    415    417    419    421
                                                   423    425    427    429    431    433    435    437    439    441    443
                                                   448    450    467    470    478    481    484    487    490    493    496
     SYMBOL        TYPE  VALUE    LENGTH   DEFN  REFERENCES

                                                   499    502    505    508    511    514    517    520    523    526    529
                                                   532    535    538    541    544    547    550    553    556    559    562
                                                   565    568    571    574    577    580    583    586    589    592    595
                                                   598    601    604    607    610    613    616    619    622    625    628
                                                   631    634    637    640    643    646    649    652    655    658    661
                                                   664    667    670    673    676    679    682    685    688    691    694
                                                   697    700    703    706    709    712    715    718    721    724    727
                                                   730    733    736    739    742    745    748    751    754    757    760
                                                   763    769    772
OPSWHERE            U    000014        1    985
OVERHEAD            D    000CA8        8    956    455    801
PAGE                U    001000        1    945
PRTLINE             C    000CC8       38    962    964    779    816    817    824
PRTLNG              U    000044        1    964    823
R0                  U    000000        1  14927     85    219    223    226    230    244    246    253    255    257    259
                                                   261    263    265    267    269    271    273    275    277    279    281
                                                   283    285    287    289    291    293    295    297    299    301    303
                                                   305    307    309    311    313    315    317    319    321    323    325
                                                   327    329    331    333    335    337    339    341    343    345    347
                                                   349    351    353    355    357    359    361    363    365    367    369
                                                   371    373    375    377    379    381    383    385    387    389    391
                                                   393    395    397    399    401    403    405    407    409    411    413
                                                   415    417    419    421    423    425    427    429    431    433    435
                                                   437    439    441    443    448    450    467    468    470    471    478
                                                   479    481    482    484    485    487    488    490    491    493    494
                                                   496    497    499    500    502    503    505    506    508    509    511
                                                   512    514    515    517    518    520    521    523    524    526    527
                                                   529    530    532    533    535    536    538    539    541    542    544
                                                   545    547    548    550    551    553    554    556    557    559    560
                                                   562    563    565    566    568    569    571    572    574    575    577
                                                   578    580    581    583    584    586    587    589    590    592    593
                                                   595    596    598    599    601    602    604    605    607    608    610
                                                   611    613    614    616    617    619    620    622    623    625    626
                                                   628    629    631    632    634    635    637    638    640    641    643
                                                   644    646    647    649    650    652    653    655    656    658    659
                                                   661    662    664    665    667    668    670    671    673    674    676
                                                   677    679    680    682    683    685    686    688    689    691    692
                                                   694    695    697    698    700    701    703    704    706    707    709
                                                   710    712    713    715    716    718    719    721    722    724    725
                                                   727    728    730    731    733    734    736    737    739    740    742
                                                   743    745    746    748    749    751    752    754    755    757    758
                                                   760    761    763    764    769    770    772    773    822    823    826
                                                   891    894    896    898    900    911
R1                  U    000001        1  14928    220    227    824    871    873    878    881    905    915
R10                 U    00000A        1  14937    806    807    809
R11                 U    00000B        1  14938    806    810
R12                 U    00000C        1  14939
R13                 U    00000D        1  14940    207    208    211    784    785    786
R14                 U    00000E        1  14941    132    205    788
R15                 U    00000F        1  14942    454    780    796    799    804    829    830    841    857    860    861
                                                   882
R2                  U    000002        1  14929    221    223    228    468    471    479    482    485    488    491    494
                                                   497    500    503    506    509    512    515    518    521    524    527
                                                   530    533    536    539    542    545    548    551    554    557    560
                                                   563    566    569    572    575    578    581    584    587    590    593
                                                   596    599    602    605    608    611    614    617    620    623    626
     SYMBOL         TYPE  VALUE    LENGTH   DEFN  REFERENCES

                                                   629    632    635    638    641    644    647    650    653    656    659
                                                   662    665    668    671    674    677    680    683    686    689    692
                                                   695    698    701    704    707    710    713    716    719    722    725
                                                   728    731    734    737    740    743    746    749    752    755    758
                                                   761    764    770    773    822    825    826    873    875    892    894
                                                   900    901    902    904    911    912
R3                   U    000003        1  14930  222    229    230    239    244    246    253    255    257    259    261
                                                   263    265    267    269    271    273    275    277    279    281    283
                                                   285    287    289    291    293    295    297    299    301    303    305
                                                   307    309    311    313    315    317    319    321    323    325    327
                                                   329    331    333    335    337    339    341    343    345    347    349
                                                   351    353    355    357    359    361    363    365    367    369    371
                                                   373    375    377    379    381    383    385    387    389    391    393
                                                   395    397    399    401    403    405    407    409    411    413    415
                                                   417    419    421    423    425    427    429    431    433    435    437
                                                   439    441    443    448    450    467    470    478    481    484    487
                                                   490    493    496    499    502    505    508    511    514    517    520
                                                   523    526    529    532    535    538    541    544    547    550    553
                                                   556    559    562    565    568    571    574    577    580    583    586
                                                   589    592    595    598    601    604    607    610    613    616    619
                                                   622    625    628    631    634    637    640    643    646    649    652
                                                   655    658    661    664    667    670    673    676    679    682    685
                                                   688    691    694    697    700    703    706    709    712    715    718
                                                   721    724    727    730    733    736    739    742    745    748    751
                                                   754    757    760    763    769    772    778    874    877    878    879
R4                   U    000004        1  14931  871    874    875    879    881
R5                   U    000005        1  14932  239    778    797    801    828    842    854    859    873
R6                   U    000006        1  14933  213    214    240    452    463    775    802    844    845    846    847
                                                   849    850    851    852    855    874
R7                   U    000007        1  14934  237    452    461    775    803    842    844    847    849    852    856
                                                   859    879
R8                   U    000008        1  14935  119    122    123    124    126
R9                   U    000009        1  14936  120    126    127
RPTDWSAV             D    000AF8        8    835  822    826
RPTSAVE              F    000AF0        4    832  796    829
RPTSPEED             I    000A88        4    796  780
RPTSVR5              F    000AF4        4    833  797    828
SAVE3T5              F    000410        4    163  239    778
SAVER13              F    000424        4    165  211    784
SAVER2               F    000420        4    164
SUBDWORD             I    000B5C        4    871  804    857
SUBDWSAV             D    000B80        8    884  871    881
SUBTEST              X    000401        1    157  144
TEST91               I    000528        4    204  132
TESTADDR             D    000400        8    155
TESTNUM              X    000400        1    156  141    214
TICKSAAA             P    000CB0        8    958  809    812
TICKSBBB             P    000CB8        8    959  810    814
TICKSTOT             P    000CC0        8    960  812    813    814    817
TIMEOPT              X    000408        1    160  138    204
TNUM                 X    000000        1    974  213
TST91LOP             U    000532        1    210  787
UTF32A               X    000E38        1   1139 1015   1030   1045   1060   1075   1138
UTF32AED             X    000EA8        1   1168 1015   1030   1045   1060   1138
UTF32ALN             A    000E34        4   1138
UTF8A                H    000DF0        2   1094 1016   1031   1046   1061   1093
```

```
     SYMBOL        TYPE  VALUE    LENGTH   DEFN  REFERENCES

UTF8AEND            X    000E2D        1   1131   1016   1031   1046   1061   1093
UTF8ALN             A    000DEC        4   1093
UTF8B               F    000EA8        4   1174   1076
UTF8BEND            C    004453        1  14921   1076
=AL2(L'MSGMSG)      R    000C7E        2    940    896
=CL5'CU14'          C    000C80        5    941    779
=F'0'               F    000C74        4    937    786
=F'1'               F    000C78        4    938    877
=H'0'               H    000C7C        2    939    891
=P'4294967296'      P    000C85        6    942    813
```

```
 MACRO     DEFN  REFERENCES

DOINSTR     187     465     476     767
OVERONLY    174     242     251     446
```

```
   DESC      SYMBOL    SIZE     POS        ADDR

Entry: 0

Image      IMAGE     17491  0000-4452  0000-4452
  Region             17491  0000-4452  0000-4452
    CSECT  CU142TST   3352  0000-0D17  0000-0D17
    CSECT  CU14TST   14139  0D18-4452  0D18-4452
   STMT                      FILE NAME

1     /devstor/dev/tests/CU14-02-performance.asm

** NO ERRORS FOUND **
```