```
 LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     2 ************************************************************************
                                                     3 *
                                                     4 *              TRTR instruction tests
                                                     5 *
                                                     6 *        NOTE: This test is based the CLCL-et-al Test
                                                     7 *              modified to only test the Performance
                                                     8 *              of the TRTR instruction.
                                                     9 *
                                                    10 *              The MSG routine is from the Hercules Binary
                                                    11 *              Floating Point Validation Package by Stephen R. Orso
                                                    12
                                                    13 *                        ********************
                                                    14 *                        **   IMPORTANT!   **
                                                    15 *                        ********************
                                                    16 *
                                                    17 *              This test uses the Hercules Diagnose X'008' interface
                                                    18 *              to display messages and thus your .tst runtest script
                                                    19 *              MUST contain a "DIAG8CMD ENABLE" statement within it!
                                                    20 *
                                                    21 *        James Wekel November 2022
                                                    22 ************************************************************************


                                                    24 ************************************************************************
                                                    25 *
                                                    26 *            TRTR Performance instruction tests
                                                    27 *
                                                    28 ************************************************************************
                                                    29 *
                                                    30 *  This program ONLY tests the performance of the TRTR
                                                    31 *  instructions.
                                                    32 *
                                                    33 *     Tests:
                                                    34 *
                                                    35 *           All tests are ' TRTR  0(255,R3),0(R5) '
                                                    36 *           where operand-1 is 256 bytes.
                                                    37 *
                                                    38 *           1. TRTR with CC=0 - no crossed pages
                                                    39 *           2. TRTR with CC=1 - both operand-1 and
                                                    40 *                  function code table cross page boundaries.
                                                    41 *           3. TRTR with CC=2 - both operand-1 and
                                                    42 *                  function code table cross page boundaries.
                                                    43 *
                                                    44 ************************************************************************
```

```
 LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     46 ********************************************************************************
                                                     47 *
                                                     48 *  Example Hercules Testcase:
                                                     49 *
                                                     50 *
                                                     51 *        *Testcase TRTR-02-performance (Test TRTR instructions)
                                                     52 *        mainsize   16
                                                     53 *        numcpu     1
                                                     54 *        sysclear
                                                     55 *        archlvl    z/Arch
                                                     56 *
                                                     57 *        loadcore   "$(testpath)/TRTR-02-performance.core" 0x0
                                                     58 *
                                                     59 *        diag8cmd   enable  # (needed for messages to Hercules console)
                                                     60 *        #r         408=ff  # (enable timing tests)
                                                     61 *        runtest    300     # (test duration, depends on host)
                                                     62 *        diag8cmd   disable # (reset back to default)
                                                     63 *
                                                     64 *        *Done
                                                     65 *
                                                     66 *
                                                     67 ********************************************************************************
                                                     69 ********************************************************************************
                                                     70 *                Low Core Definitions
                                                     71 ********************************************************************************
                                                     72 *
                          00000000  000014AF         73 TRTR2TST START 0
00000000                  00000000                   74          USING TRTR2TST,R0           Low core addressability


00000000                  00000000  000001A0         76          ORG   TRTR2TST+X'1A0'       z/Architecure RESTART PSW
000001A0  00000001 80000000                          77          DC    X'0000000180000000'
000001A8  00000000 00000200                          78          DC    AD(BEGIN)


000001B0                  000001B0  000001D0         80          ORG   TRTR2TST+X'1D0'       z/Architecure PROGRAM CHECK PSW
000001D0  00020001 80000000                          81          DC    X'0002000180000000'
000001D8  00000000 0000DEAD                          82          DC    AD(X'DEAD')


000001E0                  000001E0  00000200         84          ORG   TRTR2TST+X'200'       Start of actual test program...
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT
                                                      86 ********************************************************************
                                                      87 *               The actual "TRTR2TST" program itself...
                                                      88 ********************************************************************
                                                      89 *
                                                      90 *  Architecture Mode: 370
                                                      91 *  Register Usage:
                                                      92 *
                                                      93 *   R0       (work)
                                                      94 *   R1       (work)
                                                      95 *   R2       (work) or MSG subroutine call
                                                      96 *   R3       (work)
                                                      97 *   R4       (work)
                                                      98 *   R5       TRTRTEST Base (of current test)
                                                      99 *   R5-R7    (work)
                                                     100 *   R8       First base register
                                                     101 *   R9       Second base register
                                                     102 *   R10-R12  (work)
                                                     103 *   R13      Testing control table - base current entry
                                                     104 *   R14      Subroutine call
                                                     105 *   R15      Secondary Subroutine call or work
                                                     106 *
                                                     107 ********************************************************************
00000200                          00000200           109          USING  BEGIN,R8        FIRST Base Register
00000200                          00001200           110          USING  BEGIN+4096,R9    SECOND Base Register
00000200  0580                                       112 BEGIN    BALR   R8,0             Initalize FIRST base register
00000202  0680                                       113          BCTR   R8,0             Initalize FIRST base register
00000204  0680                                       114          BCTR   R8,0             Initalize FIRST base register

00000206  4190 8800                      00000800    116          LA     R9,2048(,R8)     Initalize SECOND base register
0000020A  4190 9800                      00000800    117          LA     R9,2048(,R9)     Initalize SECOND base register

                                                     119 *
                                                     120 **       Run the performance tests...
                                                     121 *
0000020E  45E0 8328                      00000528    122          BAL    R14,TEST91       Time TRTR   instruction  (speed test)
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    124 ********************************************************************
                                                    125 *         Test for normal or unexpected test completion...
                                                    126 ********************************************************************

00000212  95FF 8208                       00000408  128          CLI   TIMEOPT,X'FF'    Was this a timing run?
00000216  4770 8B30                       00000D30  129          BNE   EOJ              No, timing run; just go end normally

0000021A  9525 8200                       00000400  131          CLI   TESTNUM,X'25'    Did we end on expected test?
0000021E  4770 8B48                       00000D48  132          BNE   FAILTEST         No?! Then FAIL the test!

00000222  9599 8201                       00000401  134          CLI   SUBTEST,X'99'    Did we end on expected SUB-test?
00000226  4770 8B48                       00000D48  135          BNE   FAILTEST         No?! Then FAIL the test!

0000022A  47F0 8B30                       00000D30  137          B     EOJ              Yes, then normal completion!




                                                    139 ********************************************************************
                                                    140 *        Fixed test storage locations ...
                                                    141 ********************************************************************


0000022E                        0000022E  00000400  143          ORG   BEGIN+X'200'
                                                    144
00000400                                            145 TESTADDR DS    0D                Where test/subtest numbers will go
00000400  99                                        146 TESTNUM  DC    X'99'      Test number of active test
00000401  99                                        147 SUBTEST  DC    X'99'      Active test sub-test number


00000408                                            149          DS    0D
00000408  00                                        150 TIMEOPT  DC    X'00'      Set to non-zero to run timing tests


00000410                                            152          DS    0D
00000410  00000000 00000000                         153 SAVE3T5  DC    4F'0'
00000420  00000000                                  154 SAVER2   DC    F'0'
00000424  00000000                                  155 SAVER13   DC    F'0'


00000428                        00000428  00000528  157          ORG   *+X'100'
```

```
 LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   159 ***************************************************************************
                                                   160 *     Define come helpful macros to ensure our counts are correct
                                                   161 ***************************************************************************



                                                   163          MACRO
                                                   164          OVERONLY &NUM              &NUM = number of sets
                                                   165          LCLA  &CTR
                                                   166 &CTR     SETA  &NUM
                                                   167 .LOOP    ANOP
                                                   168 .*
                                                   169 *
                                                   170          LM    R3,R5,OPSPERF        Get TRTR operands
                                                   171 .*
                                                   172 &CTR     SETA  &CTR-1
                                                   173          AIF   (&CTR GT 0).LOOP
                                                   174          MEND



                                                   176          MACRO
                                                   177          DOINSTR &NUM               &NUM = number of sets
                                                   178          LCLA  &CTR
                                                   179 &CTR     SETA  &NUM
                                                   180 .LOOP    ANOP
                                                   181 .*
                                                   182 *
                                                   183          LM    R3,R5,OPSPERF        Load TRTR operands
                                                   184          TRTR  0(255,R3),0(R5)      Do TRTR
                                                   185 .*
                                                   186 &CTR     SETA  &CTR-1
                                                   187          AIF   (&CTR GT 0).LOOP
                                                   188          MEND
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     190 ********************************************************************
                                                     191 *        TEST91                  Time TRTR instruction  (speed test)
                                                     192 ********************************************************************

00000528  91FF 8208                       00000408   194 TEST91   TM    TIMEOPT,X'FF'    Is timing tests option enabled?
0000052C  078E                                       195          BZR   R14              No, skip timing tests

0000052E  41D0 8BF0                       00000DF0   197          LA    R13,TRTRPERF         Point R13 --> testing control table
00000532                   00000000                  198          USING TRTRTEST,R13         What each table entry looks like
                                                     199 *
                           00000532       00000001   200 TST91LOP EQU   *
00000532  50D0 8224                       00000424   201          ST    R13,SAVER13          save current pref table base
                                                     202 *
00000536  4360 D000                       00000000   203          IC    R6,TNUM            Set test number
0000053A  4260 8200                       00000400   204          STC   R6,TESTNUM
                                                     205 *
                                                     206 **       Initialize operand data  (move data to testing address)
                                                     207 *
0000053E  58A0 D01C                       0000001C   208          L     R10,OP1WHERE         Where to move operand-1 data to
00000542  58B0 D008                       00000008   209          L     R11,OP1LEN           operand-1 length
00000546  50B0 D020                       00000020   210          ST    R11,OP1WLEN            and save for later
0000054A  5860 D004                       00000004   211          L     R6,OP1DATA           Where op1 data is right now
0000054E  5870 D008                       00000008   212          L     R7,OP1LEN            How much of it there is
00000552  0EA6                                       213          MVCL  R10,R6
                                                     214 *
00000554  58A0 D024                       00000024   215          L     R10,OP2WHERE         Where to move operand-2 data to
00000558  58B0 D010                       00000010   216          L     R11,OP2LEN           How much of it there is
0000055C  5860 D00C                       0000000C   217          L     R6,OP2DATA           Where op2 data is right now
00000560  5870 D010                       00000010   218          L     R7,OP2LEN            How much of it there is
00000564  0EA6                                       219          MVCL  R10,R6


00000566  9815 D014                       00000014   221          LM    R1,R5,OPSWHERE     get TRTR input; set OP addr to end
0000056A  1A34                                       222          AR    R3,R4              add OP length -1
0000056C  0630                                       223          BCTR  R3,0
0000056E  9035 8940                       00000B40   224          STM   R3,R5,OPSPERF      save for preformance test
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT
                                                    226 ************************************************************************
                                                    227 *        Next, time the overhead...
                                                    228 ************************************************************************

00000572  5870 8B64                    00000D64    230          L     R7,NUMLOOPS
00000576  B205 8B68                    00000D68    231          STCK  BEGCLOCK
0000057A  9035 8210                    00000410    232          STM   R3,R5,SAVE3T5
0000057E  0560                                     233          BALR  R6,0
                                                    234 *                                    100 sets of overhead
                                                    235          OVERONLY 2                  (first 2)
                                                    236+*
00000580  9835 8940                    00000B40    237+         LM    R3,R5,OPSPERF         Get TRTR operands
                                                    238+*
00000584  9835 8940                    00000B40    239+         LM    R3,R5,OPSPERF         Get TRTR operands

                                                    241 *        .........ETC.........

                                                    243          PRINT OFF
                                                    437          PRINT ON

                                                    439          OVERONLY 2                  (last 2)
                                                    440+*
00000708  9835 8940                    00000B40    441+         LM    R3,R5,OPSPERF         Get TRTR operands
                                                    442+*
0000070C  9835 8940                    00000B40    443+         LM    R3,R5,OPSPERF         Get TRTR operands
                                                    444 *
00000710  0676                                     445          BCTR  R7,R6
00000712  B205 8B70                    00000D70    446          STCK  ENDCLOCK
00000716  45F0 89E0                    00000BE0    447          BAL   R15,CALCDUR
0000071A  D207 8B80 8B78      00000D80 00000D78    448          MVC   OVERHEAD,DURATION
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    450 ************************************************************************
                                                    451 *        Now do the actual timing run...
                                                    452 ************************************************************************

00000720  5870 8B64                      00000D64   454          L     R7,NUMLOOPS
00000724  B205 8B68                      00000D68   455          STCK  BEGCLOCK
00000728  0560                                      456          BALR  R6,0
                                                    457 *                                    100 sets of instructions
                                                    458          DOINSTR 2                   (first 2)
                                                    459+*
0000072A  9835 8940                      00000B40   460+         LM    R3,R5,OPSPERF         Load TRTR operands
0000072E  D0FE 3000 5000      00000000   00000000   461+         TRTR  0(255,R3),0(R5)       Do TRTR
                                                    462+*
00000734  9835 8940                      00000B40   463+         LM    R3,R5,OPSPERF         Load TRTR operands
00000738  D0FE 3000 5000      00000000   00000000   464+         TRTR  0(255,R3),0(R5)       Do TRTR

                                                    466 *        .........ETC.........

                                                    468          PRINT OFF
                                                    758          PRINT ON

                                                    760          DOINSTR 2                   (last 2)
                                                    761+*
00000AFE  9835 8940                      00000B40   762+         LM    R3,R5,OPSPERF         Load TRTR operands
00000B02  D0FE 3000 5000      00000000   00000000   763+         TRTR  0(255,R3),0(R5)       Do TRTR
                                                    764+*
00000B08  9835 8940                      00000B40   765+         LM    R3,R5,OPSPERF         Load TRTR operands
00000B0C  D0FE 3000 5000      00000000   00000000   766+         TRTR  0(255,R3),0(R5)       Do TRTR


00000B12  0676                                      768          BCTR  R7,R6
00000B14  B205 8B70                      00000D70   769          STCK  ENDCLOCK


00000B18  9835 8210                      00000410   771          LM    R3,R5,SAVE3T5
00000B1C  D204 8BC1 8B58      00000DC1   00000D58   772          MVC   PRTLINE+33(5),=CL5'TRTR'
00000B22  45F0 8960                      00000B60   773          BAL   R15,RPTSPEED
                                                    774 *
                                                    775 *  more performance tests?
                                                    776 *
00000B26  58D0 8224                      00000424   777          L     R13,SAVER13             restore perf table base
00000B2A  41D0 D034                      00000034   778          LA    R13,TRTRNEXT           Go on to next table entry
00000B2E  D503 8B4C D000      00000D4C   00000000   779          CLC   =F'0',0(R13)           End of table?
00000B34  4770 8332                      00000532   780          BNE   TST91LOP              No, loop...
00000B38  07FE                                      781          BR    R14                   Return to caller or FAILTEST


00000B40  00000000 00000000                         783 OPSPERF  DS    4D                    Performance test R3-R5
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    785 ********************************************************************************
                                                    786 *        RPTSPEED                  Report instruction speed
                                                    787 ********************************************************************************

00000B60  50F0 89C8                      00000BC8   789 RPTSPEED ST    R15,RPTSAVE        Save return address
00000B64  5050 89CC                      00000BCC   790          ST    R5,RPTSVR5         Save R5
                                                    791 *
00000B68  45F0 89E0                      00000BE0   792          BAL   R15,CALCDUR        Calculate duration
                                                    793 *
00000B6C  4150 8B80                      00000D80   794          LA    R5,OVERHEAD        Subtract overhead
00000B70  4160 8B78                      00000D78   795          LA    R6,DURATION        From raw timing
00000B74  4170 8B78                      00000D78   796          LA    R7,DURATION        Yielding true instruction timing
00000B78  45F0 8A34                      00000C34   797          BAL   R15,SUBDWORD       Do it
                                                    798 *
00000B7C  98AB 8B78                      00000D78   799          LM    R10,R11,DURATION   Convert to...
00000B80  8CA0 000C                      0000000C   800          SRDL  R10,12             ... microseconds
                                                    801 *
00000B84  4EA0 8B88                      00000D88   802          CVD   R10,TICKSAAA       convert HIGH part to decimal
00000B88  4EB0 8B90                      00000D90   803          CVD   R11,TICKSBBB       convert LOW  part to decimal
                                                    804 *
00000B8C  F877 8B98 8B88      00000D98   00000D88   805          ZAP   TICKSTOT,TICKSAAA           Calculate...
00000B92  FC75 8B98 8B5D      00000D98   00000D5D   806          MP    TICKSTOT,=P'4294967296'     ...decimal...
00000B98  FA77 8B98 8B90      00000D98   00000D90   807          AP    TICKSTOT,TICKSBBB           ...microseconds
                                                    808 *
00000B9E  D20B 8BCB 8BE4      00000DCB   00000DE4   809          MVC   PRTLINE+43(L'EDIT),EDIT         (edit into...
00000BA4  DE0B 8BCB 8B9B      00000DCB   00000D9B   810          ED    PRTLINE+43(L'EDIT),TICKSTOT+3    ...print line)
```

```
                                                    812 *
                                                    813 *        Use Hercules Diagnose for Message to console
                                                    814 *
00000BAA  9002 89D0                      00000BD0   815          STM   R0,R2,RPTDWSAV       save regs used by MSG
00000BAE  4100 0044                      00000044   816          LA    R0,PRTLNG            message length
00000BB2  4110 8BA0                      00000DA0   817          LA    R1,PRTLINE           messagfe address
00000BB6  4520 8A68                      00000C68   818          BAL   R2,MSG               call Hercules console MSG display
00000BBA  9802 89D0                      00000BD0   819          LM    R0,R2,RPTDWSAV       restore regs


00000BBE  5850 89CC                      00000BCC   821          L     R5,RPTSVR5         Restore R5
00000BC2  58F0 89C8                      00000BC8   822          L     R15,RPTSAVE        Restore return address
00000BC6  07FF                                      823          BR    R15                Return to caller

00000BC8  00000000                                  825 RPTSAVE  DC    F'0'               R15 save area
00000BCC  00000000                                  826 RPTSVR5  DC    F'0'               R5 save area

00000BD0  00000000 00000000                         828 RPTDWSAV DC    2D'0'              R0-R2 save area for MSG call
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    830 ********************************************************************************
                                                    831 *        CALCDUR                  Calculate DURATION
                                                    832 ********************************************************************************

00000BE0  50F0 8A24                      00000C24   834 CALCDUR  ST    R15,CALCRET        Save return address
00000BE4  9057 8A28                      00000C28   835          STM   R5,R7,CALCWORK     Save work registers
                                                    836 *
00000BE8  9867 8B68                      00000D68   837          LM    R6,R7,BEGCLOCK     Remove CPU number from clock value
00000BEC  8C60 0006                      00000006   838          SRDL  R6,6                       "
00000BF0  8D60 0006                      00000006   839          SLDL  R6,6                       "
00000BF4  9067 8B68                      00000D68   840          STM   R6,R7,BEGCLOCK             "
                                                    841 *
00000BF8  9867 8B70                      00000D70   842          LM    R6,R7,ENDCLOCK     Remove CPU number from clock value
00000BFC  8C60 0006                      00000006   843          SRDL  R6,6                       "
00000C00  8D60 0006                      00000006   844          SLDL  R6,6                       "
00000C04  9067 8B70                      00000D70   845          STM   R6,R7,ENDCLOCK             "
                                                    846 *
00000C08  4150 8B68                      00000D68   847          LA    R5,BEGCLOCK        Starting time
00000C0C  4160 8B70                      00000D70   848          LA    R6,ENDCLOCK        Ending time
00000C10  4170 8B78                      00000D78   849          LA    R7,DURATION        Difference
00000C14  45F0 8A34                      00000C34   850          BAL   R15,SUBDWORD       Calculate duration
                                                    851 *
00000C18  9857 8A28                      00000C28   852          LM    R5,R7,CALCWORK     Restore work registers
00000C1C  58F0 8A24                      00000C24   853          L     R15,CALCRET        Restore return address
00000C20  07FF                                      854          BR    R15                Return to caller

00000C24  00000000                                  856 CALCRET  DC    F'0'               R15 save area
00000C28  00000000 00000000                         857 CALCWORK DC    3F'0'              R5-R7 save area



                                                    859 ********************************************************************************
                                                    860 *        SUBDWORD                 Subtract two doublewords
                                                    861 *        R5 --> subtrahend, R6 --> minuend, R7 --> result
                                                    862 ********************************************************************************

00000C34  9014 8A58                      00000C58   864 SUBDWORD STM   R1,R4,SUBDWSAV   Save registers
                                                    865 *
00000C38  9812 5000                      00000000   866          LM    R1,R2,0(R5)        Subtrahend  (value to subtract)
00000C3C  9834 6000                      00000000   867          LM    R3,R4,0(R6)        Minuend     (what to subtract FROM)
00000C40  1F42                                      868          SLR   R4,R2              Subtract LOW part
00000C42  47B0 8A4A                      00000C4A   869          BNM   *+4+4                (branch if no borrow)
00000C46  5F30 8B50                      00000D50   870          SL    R3,=F'1'             (otherwise do borrow)
00000C4A  1F31                                      871          SLR   R3,R1              Subtract HIGH part
00000C4C  9034 7000                      00000000   872          STM   R3,R4,0(R7)        Store results
                                                    873 *
00000C50  9814 8A58                      00000C58   874          LM    R1,R4,SUBDWSAV     Restore registers
00000C54  07FF                                      875          BR    R15                Return to caller

00000C58  00000000 00000000                         877 SUBDWSAV DC    2D'0'              R1-R4 save area
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    879 ********************************************************************
                                                    880 *        Issue HERCULES MESSAGE pointed to by R1, length in R0
                                                    881 *              R2 = return address
                                                    882 ********************************************************************

00000C68  4900 8B54                    00000D54    884 MSG      CH    R0,=H'0'                Do we even HAVE a message?
00000C6C  07D2                                     885          BNHR  R2                      No, ignore

00000C6E  9002 8AA0                    00000CA0    887          STM   R0,R2,MSGSAVE           Save registers

00000C72  4900 8B56                    00000D56    889          CH    R0,=AL2(L'MSGMSG)       Message length within limits?
00000C76  47D0 8A7E                    00000C7E    890          BNH   MSGOK                   Yes, continue
00000C7A  4100 005F                    0000005F    891          LA    R0,L'MSGMSG             No, set to maximum

00000C7E  1820                                     893 MSGOK    LR    R2,R0                   Copy length to work register
00000C80  0620                                     894          BCTR  R2,0                    Minus-1 for execute
00000C82  4420 8AAC                    00000CAC    895          EX    R2,MSGMVC               Copy message to O/P buffer

00000C86  4120 200A                    0000000A    897          LA    R2,1+L'MSGCMD(,R2)      Calculate true command length
00000C8A  4110 8AB2                    00000CB2    898          LA    R1,MSGCMD               Point to true command

00000C8E  83120008                                 900          DC    X'83',X'12',X'0008'     Issue Hercules Diagnose X'008'
00000C92  4780 8A98                    00000C98    901          BZ    MSGRET                  Return if successful
00000C96  0000                                     902          DC    H'0'                    CRASH for debugging purposes

00000C98  9802 8AA0                    00000CA0    904 MSGRET   LM    R0,R2,MSGSAVE           Restore registers
00000C9C  07F2                                     905          BR    R2                      Return to caller



00000CA0  00000000 00000000                        907 MSGSAVE  DC    3F'0'                   Registers save area
00000CAC  D200 8ABB 1000       00000CBB  00000000  908 MSGMVC   MVC   MSGMSG(0),0(R1)         Executed instruction


00000CB2  D4E2C7D5 D6C8405C                        910 MSGCMD   DC    C'MSGNOH * '            *** HERCULES MESSAGE COMMAND ***
00000CBB  40404040 40404040                        911 MSGMSG   DC    CL95' '                 The message text to be displayed
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT
                                                    913 ************************************************************************
                                                    914 *        Normal completion or Abnormal termination PSWs
                                                    915 ************************************************************************


00000D20  00020001 80000000                         917 EOJPSW   DC    0D'0',X'0002000180000000',AD(0)
00000D30  B2B2 8B20                 00000D20        919 EOJ      LPSWE EOJPSW               Normal completion


00000D38  00020001 80000000                         921 FAILPSW  DC    0D'0',X'0002000180000000',AD(X'BAD')
00000D48  B2B2 8B38                 00000D38        923 FAILTEST LPSWE FAILPSW              Abnormal termination
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                     925 ********************************************************************************
                                                     926 *        Working Storage
                                                     927 ********************************************************************************



00000D4C                                             929          LTORG ,                 Literals pool
00000D4C  00000000                                   930                =F'0'
00000D50  00000001                                   931                =F'1'
00000D54  0000                                       932                =H'0'
00000D56  005F                                       933                =AL2(L'MSGMSG)
00000D58  E3D9E3D9 40                                934                =CL5'TRTR'
00000D5D  04294967 296C                              935                =P'4294967296'

                          00000400  00000001        937 K        EQU   1024            One KB
                          00001000  00000001        938 PAGE     EQU   (4*K)           Size of one page
                          00004000  00000001        939 K16      EQU   (16*K)          16 KB
                          00008000  00000001        940 K32      EQU   (32*K)          32 KB
                          00010000  00000001        941 K64      EQU   (64*K)          64 KB
                          00100000  00000001        942 MB       EQU   (K*K)            1 MB

00000D64  00002710                                   944 NUMLOOPS DC    F'10000'        10,000 * 100 = 1,000,000

00000D68  BBBBBBBB BBBBBBBB                          946 BEGCLOCK DC    0D'0',8X'BB'    Begin
00000D70  EEEEEEEE EEEEEEEE                          947 ENDCLOCK DC    0D'0',8X'EE'    End
00000D78  DDDDDDDD DDDDDDDD                          948 DURATION DC    0D'0',8X'DD'    Diff
00000D80  FFFFFFFF FFFFFFFF                          949 OVERHEAD DC    0D'0',8X'FF'    Overhead

00000D88  00000000 0000000C                          951 TICKSAAA DC    PL8'0'          Clock ticks high part
00000D90  00000000 0000000C                          952 TICKSBBB DC    PL8'0'          Clock ticks low part
00000D98  00000000 0000000C                          953 TICKSTOT DC    PL8'0'          Total clock ticks

00000DA0  40404040 40404040                          955 PRTLINE  DC    C'         1,000,000 iterations of XXXXX'
00000DC6  40A39696 9240F9F9                          956          DC    C' took 999,999,999 microseconds'
                          00000044  00000001        957 PRTLNG   EQU   *-PRTLINE
00000DE4  40202020 6B202020                          958 EDIT     DC    X'402020206B2020206B202120'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    961 ********************************************************************************
                                                    962 *        TRTRTEST DSECT
                                                    963 ********************************************************************************

                                                    965 TRTRTEST DSECT ,
00000000  00                                        966 TNUM     DC    X'00'          TRTR table Number
00000001  00                                        967          DC    X'00'
00000002  00                                        968          DC    X'00'
00000003  00                                        969          DC    X'00'

00000004  00000000                                  971 OP1DATA  DC    A(0)           Pointer to Operand-1 data
00000008  00000000                                  972 OP1LEN   DC    F'0'           How much data is there - 1
0000000C  00000000                                  973 OP2DATA  DC    A(0)           Pointer to FC table data
00000010  00000000                                  974 OP2LEN   DC    F'0'           How much data is there - FC Table

                          00000014  00000001        976 OPSWHERE EQU   *
00000014  00000000                                  977 GR1PATT  DC    A(0)           GR1 - Polluted Register pattern
00000018  00000000                                  978 GR2PATT  DC    A(0)           GR2 - Polluted Register pattern
0000001C  00000000                                  979 OP1WHERE DC    A(0)           Where Operand-1 data should be placed
00000020  00000000                                  980 OP1WLEN  DC    F'0'           How much data is there - 1
00000024  00000000                                  981 OP2WHERE DC    A(0)           Where FC Table  data should be placed

00000028  00000000                                  983 FAILMASK DC    A(0)           Failure Branch on Condition mask

                                                    985 *                             Ending register values
0000002C  00000000                                  986 ENDREGS  DC    A(0)              GR1 - FC address
00000030  00000000                                  987          DC    A(0)              GR2 - Function Code

                          00000034  00000001        989 TRTRNEXT EQU   *              Start of next table entry...

                          AABBCCDD  00000001        991 REG2PATT EQU   X'AABBCCDD'     Polluted Register pattern
                          000000DD  00000001        992 REG2LOW  EQU         X'DD'     (last byte above)
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                               00000000  000014AF   994 TRTR2TST CSECT ,
                                                    995 ********************************************************************
                                                    996 *        TRTR Performace Test data...
                                                    997 ********************************************************************
00000DF0                                            998 TRTRPERF DC    0A(0)      start of table




                                                   1000 ********************************************************************
                                                   1001 *        tests with op-1 length 256
                                                   1002 ********************************************************************


00000DF0                                           1004 CC0T4    DS    0F
00000DF0  04                                       1005          DC    X'04'                        Test Num
00000DF1  0000                                     1006          DC    X'00',X'00'
00000DF3  00                                       1007          DC    X'00'
                                                   1008 *
00000DF4  00000E94 00000100                        1009          DC    A(TRTOP10),A(256)            Source - Op 1 & length
00000DFC  00001194 00000100                        1010          DC    A(TRTOP20),A(256)            Source - FC Table & length
                                                   1011 *                                           Target -
00000E04  AABBCCDD AABBCCDD                        1012          DC    A(REG2PATT),A(REG2PATT)                GR1, GR2
00000E0C  0010C000 00000000                        1013          DC    A(1*MB+(3*K16)),A(0),A(2*MB+(3*K16))  Op1, Op1L, FCT
                                                   1014 *
00000E18  00000007                                 1015          DC    A(7)                          not CC0
00000E1C  AABBCCDD AABBCCDD                        1016          DC    A(REG2PATT),A(REG2PATT)       FC address, Code




00000E24                                           1018 CC1T5    DS    0F
00000E24  15                                       1019          DC    X'15'                        Test Num
00000E25  0000                                     1020          DC    X'00',X'00'
00000E27  00                                       1021          DC    X'00'
                                                   1022 *
00000E28  00000F94 00000100                        1023          DC    A(TRTOP1F0),A(256)           Source - Op 1 & length
00000E30  000012A0 00000100                        1024          DC    A(TRTOP2F0),A(256)           Source - FC Table & length
                                                   1025 *                                           Target -
00000E38  AABBCCDD AABBCCDD                        1026          DC    A(REG2PATT),A(REG2PATT)                GR1, GR2
00000E40  0030FFF3 00000000                        1027          DC    A(3*MB+(4*K16)-13),A(0),A(4*MB+(4*K16)-29)  Op1,.., FCT
                                                   1028 *
00000E4C  0000000B                                 1029          DC    A(11)                         not CC1
00000E50  0030FFF4 AABBCCF0                        1030          DC    A(3*MB+(4*K16)-13+1),XL4'AABBCCF0' FC address, Code
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

00000E58                                          1032 CC2T5    DS    0F
00000E58  25                                      1033          DC    X'25'                        Test Num
00000E59  0000                                    1034          DC    X'00',X'00'
00000E5B  00                                      1035          DC    X'00'
                                                  1036 *
00000E5C  00001094 00000100                       1037          DC    A(TRTOP1F1),A(256)           Source - Op 1 & length
00000E64  000013A8 00000100                       1038          DC    A(TRTOP8F1),A(256)           Source - FC Table & length
                                                  1039 *                                           Target -
00000E6C  AABBCCDD AABBCCDD                       1040          DC    A(REG2PATT),A(REG2PATT)                GR1, GR2
00000E74  0050FFDF 00000000                       1041          DC    A(5*MB+(4*K16)-33),A(0),A(6*MB+(4*K16)-41)  Op1,.., FCT
                                                  1042 *
00000E80  0000000D                                1043          DC    A(13)                         not CC2
00000E84  0050FFDF AABBCCF1                       1044          DC    A(5*MB+(4*K16)-33),XL4'AABBCCF1' FC address, Code


                                                  1046
00000E8C  00000000                                1047          DC    A(0)        end of table
00000E90  00000000                                1048          DC    A(0)        end of table
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   1050 ********************************************************************************
                                                   1051 *        TRTR op1 scan data...
                                                   1052 ********************************************************************************


00000E94  78125634 78125634                        1054 TRTOP10  DC    64XL4'78125634'    (CC0)

00000F94  00F00000 78125634                        1056 TRTOP1F0 DC    X'00F00000',63XL4'78125634'    (CC1)

00001094  F1000000 78125634                        1058 TRTOP1F1 DC    X'F1000000',63XL4'78125634'    (CC2)


                                                   1060 ********************************************************************************
                                                   1061 *        Function Code (FC) Tables
                                                   1062 ********************************************************************************


00001194  00000000 00000000                        1064 TRTOP20  DC    256X'00'            no stop
00001298  00000000 00000000                        1065          DS    D

000012A0  00000000 00000000                        1067 TRTOP2F0 DC    240X'00',X'F0',15X'00'     stop on X'F0'
000013A0  00000000 00000000                        1068          DS    D

000013A8  00000000 00000000                        1070 TRTOP8F1 DC    240X'00',X'00',X'F1',14X'00'     stop on X'F1'
000014A8  00000000 00000000                        1071          DS    D


                                                   1073 ********************************************************************************
                                                   1074 *        Register equates
                                                   1075 ********************************************************************************


                  00000000  00000001  1077 R0       EQU   0
                  00000001  00000001  1078 R1       EQU   1
                  00000002  00000001  1079 R2       EQU   2
                  00000003  00000001  1080 R3       EQU   3
                  00000004  00000001  1081 R4       EQU   4
                  00000005  00000001  1082 R5       EQU   5
                  00000006  00000001  1083 R6       EQU   6
                  00000007  00000001  1084 R7       EQU   7
                  00000008  00000001  1085 R8       EQU   8
                  00000009  00000001  1086 R9       EQU   9
                  0000000A  00000001  1087 R10      EQU   10
                  0000000B  00000001  1088 R11      EQU   11
                  0000000C  00000001  1089 R12      EQU   12
                  0000000D  00000001  1090 R13      EQU   13
                  0000000E  00000001  1091 R14      EQU   14
                  0000000F  00000001  1092 R15      EQU   15
```

```
 LOC        OBJECT CODE       ADDR1     ADDR2     STMT
                                                  1094          END
```

| SYMBOL | TYPE | VALUE | LENGTH | DEFN | REFERENCES |
|---|---|---|---|---|---|
| BEGCLOCK | D | 00000D68 | 8 | 946 | 231 455 837 840 847 |
| BEGIN | I | 00000200 | 2 | 112 | 143 78 109 110 |
| CALCDUR | I | 00000BE0 | 4 | 834 | 447 792 |
| CALCRET | F | 00000C24 | 4 | 856 | 834 853 |
| CALCWORK | F | 00000C28 | 4 | 857 | 835 852 |
| CC0T4 | F | 00000DF0 | 4 | 1004 | |
| CC1T5 | F | 00000E24 | 4 | 1018 | |
| CC2T5 | F | 00000E58 | 4 | 1032 | |
| DURATION | D | 00000D78 | 8 | 948 | 448 795 796 799 849 |
| EDIT | X | 00000DE4 | 12 | 958 | 809 810 |
| ENDCLOCK | D | 00000D70 | 8 | 947 | 446 769 842 845 848 |
| ENDREGS | A | 0000002C | 4 | 986 | |
| EOJ | I | 00000D30 | 4 | 919 | 129 137 |
| EOJPSW | D | 00000D20 | 8 | 917 | 919 |
| FAILMASK | A | 00000028 | 4 | 983 | |
| FAILPSW | D | 00000D38 | 8 | 921 | 923 |
| FAILTEST | I | 00000D48 | 4 | 923 | 132 135 |
| GR1PATT | A | 00000014 | 4 | 977 | |
| GR2PATT | A | 00000018 | 4 | 978 | |
| IMAGE | 1 | 00000000 | 5296 | 0 | |
| K | U | 00000400 | 1 | 937 | 938 939 940 941 942 |
| K16 | U | 00004000 | 1 | 939 | 1013 1027 1030 1041 1044 |
| K32 | U | 00008000 | 1 | 940 | |
| K64 | U | 00010000 | 1 | 941 | |
| MB | U | 00100000 | 1 | 942 | 1013 1027 1030 1041 1044 |
| MSG | I | 00000C68 | 4 | 884 | 818 |
| MSGCMD | C | 00000CB2 | 9 | 910 | 897 898 |
| MSGMSG | C | 00000CBB | 95 | 911 | 891 908 889 |
| MSGMVC | I | 00000CAC | 6 | 908 | 895 |
| MSGOK | I | 00000C7E | 2 | 893 | 890 |
| MSGRET | I | 00000C98 | 4 | 904 | 901 |
| MSGSAVE | F | 00000CA0 | 4 | 907 | 887 904 |
| NUMLOOPS | F | 00000D64 | 4 | 944 | 230 454 |
| OP1DATA | A | 00000004 | 4 | 971 | 211 |
| OP1LEN | F | 00000008 | 4 | 972 | 209 212 |
| OP1WHERE | A | 0000001C | 4 | 979 | 208 |
| OP1WLEN | F | 00000020 | 4 | 980 | 210 |
| OP2DATA | A | 0000000C | 4 | 973 | 217 |
| OP2LEN | F | 00000010 | 4 | 974 | 216 218 |
| OP2WHERE | A | 00000024 | 4 | 981 | 215 |
| OPSPERF | D | 00000B40 | 8 | 783 | 224 237 239 246 248 250 252 254 256 258 260 262 264 |
| | | | | | 266 268 270 272 274 276 278 280 282 284 286 288 290 |
| | | | | | 292 294 296 298 300 302 304 306 308 310 312 314 316 |
| | | | | | 318 320 322 324 326 328 330 332 334 336 338 340 342 |
| | | | | | 344 346 348 350 352 354 356 358 360 362 364 366 368 |
| | | | | | 370 372 374 376 378 380 382 384 386 388 390 392 394 |
| | | | | | 396 398 400 402 404 406 408 410 412 414 416 418 420 |
| | | | | | 422 424 426 428 430 432 434 436 441 443 460 463 471 |
| | | | | | 474 477 480 483 486 489 492 495 498 501 504 507 510 |
| | | | | | 513 516 519 522 525 528 531 534 537 540 543 546 549 |
| | | | | | 552 555 558 561 564 567 570 573 576 579 582 585 588 |
| | | | | | 591 594 597 600 603 606 609 612 615 618 621 624 627 |
| | | | | | 630 633 636 639 642 645 648 651 654 657 660 663 666 |
| | | | | | 669 672 675 678 681 684 687 690 693 696 699 702 705 |
| | | | | | 708 711 714 717 720 723 726 729 732 735 738 741 744 |
| | | | | | 747 750 753 756 762 765 |

```
     SYMBOL         TYPE    VALUE      LENGTH    DEFN   REFERENCES

OPSWHERE             U     00000014          1    976   221
OVERHEAD             D     00000D80          8    949   448   794
PAGE                 U     00001000          1    938
PRTLINE              C     00000DA0         38    955   957   772   809   810   817
PRTLNG               U     00000044          1    957   816
R0                   U     00000000          1   1077    74   815   816   819   884   887   889   891   893   904
R1                   U     00000001          1   1078   221   817   864   866   871   874   898   908
R10                  U     0000000A          1   1087   208   213   215   219   799   800   802
R11                  U     0000000B          1   1088   209   210   216   799   803
R12                  U     0000000C          1   1089
R13                  U     0000000D          1   1090   197   198   201   777   778   779
R14                  U     0000000E          1   1091   122   195   781
R15                  U     0000000F          1   1092   447   773   789   792   797   822   823   834   850   853   854   875
R2                   U     00000002          1   1079   815   818   819   866   868   885   887   893   894   895   897   904   905
R3                   U     00000003          1   1080   222   223   224   232   237   239   246   248   250   252   254   256   258
                                                        260   262   264   266   268   270   272   274   276   278   280   282   284
                                                        286   288   290   292   294   296   298   300   302   304   306   308   310
                                                        312   314   316   318   320   322   324   326   328   330   332   334   336
                                                        338   340   342   344   346   348   350   352   354   356   358   360   362
                                                        364   366   368   370   372   374   376   378   380   382   384   386   388
                                                        390   392   394   396   398   400   402   404   406   408   410   412   414
                                                        416   418   420   422   424   426   428   430   432   434   436   441   443
                                                        460   461   463   464   471   472   474   475   477   478   480   481   483
                                                        484   486   487   489   490   492   493   495   496   498   499   501   502
                                                        504   505   507   508   510   511   513   514   516   517   519   520   522
                                                        523   525   526   528   529   531   532   534   535   537   538   540   541
                                                        543   544   546   547   549   550   552   553   555   556   558   559   561
                                                        562   564   565   567   568   570   571   573   574   576   577   579   580
                                                        582   583   585   586   588   589   591   592   594   595   597   598   600
                                                        601   603   604   606   607   609   610   612   613   615   616   618   619
                                                        621   622   624   625   627   628   630   631   633   634   636   637   639
                                                        640   642   643   645   646   648   649   651   652   654   655   657   658
                                                        660   661   663   664   666   667   669   670   672   673   675   676   678
                                                        679   681   682   684   685   687   688   690   691   693   694   696   697
                                                        699   700   702   703   705   706   708   709   711   712   714   715   717
                                                        718   720   721   723   724   726   727   729   730   732   733   735   736
                                                        738   739   741   742   744   745   747   748   750   751   753   754   756
                                                        757   762   763   765   766   771   867   870   871   872
R4                   U     00000004          1   1081   222   864   867   868   872   874
R5                   U     00000005          1   1082   221   224   232   237   239   246   248   250   252   254   256   258   260
                                                        262   264   266   268   270   272   274   276   278   280   282   284   286
                                                        288   290   292   294   296   298   300   302   304   306   308   310   312
                                                        314   316   318   320   322   324   326   328   330   332   334   336   338
                                                        340   342   344   346   348   350   352   354   356   358   360   362   364
                                                        366   368   370   372   374   376   378   380   382   384   386   388   390
                                                        392   394   396   398   400   402   404   406   408   410   412   414   416
                                                        418   420   422   424   426   428   430   432   434   436   441   443   460
                                                        461   463   464   471   472   474   475   477   478   480   481   483   484
                                                        486   487   489   490   492   493   495   496   498   499   501   502   504
                                                        505   507   508   510   511   513   514   516   517   519   520   522   523
                                                        525   526   528   529   531   532   534   535   537   538   540   541   543
                                                        544   546   547   549   550   552   553   555   556   558   559   561   562
                                                        564   565   567   568   570   571   573   574   576   577   579   580   582
                                                        583   585   586   588   589   591   592   594   595   597   598   600   601
                                                        603   604   606   607   609   610   612   613   615   616   618   619   621
                                                        622   624   625   627   628   630   631   633   634   636   637   639   640
```

| SYMBOL | TYPE | VALUE | LENGTH | DEFN | REFERENCES |
|---|---|---|---|---|---|
| | | | | | 642 643 645 646 648 649 651 652 654 655 657 658 660 |
| | | | | | 661 663 664 666 667 669 670 672 673 675 676 678 679 |
| | | | | | 681 682 684 685 687 688 690 691 693 694 696 697 699 |
| | | | | | 700 702 703 705 706 708 709 711 712 714 715 717 718 |
| | | | | | 720 721 723 724 726 727 729 730 732 733 735 736 738 |
| | | | | | 739 741 742 744 745 747 748 750 751 753 754 756 757 |
| | | | | | 762 763 765 766 771 790 794 821 835 847 852 866 |
| R6 | U | 00000006 | 1 | 1083 | 203 204 211 213 217 219 233 445 456 768 795 837 838 |
| | | | | | 839 840 842 843 844 845 848 867 |
| R7 | U | 00000007 | 1 | 1084 | 212 218 230 445 454 768 796 835 837 840 842 845 849 |
| | | | | | 852 872 |
| R8 | U | 00000008 | 1 | 1085 | 109 112 113 114 116 |
| R9 | U | 00000009 | 1 | 1086 | 110 116 117 |
| REG2LOW | U | 000000DD | 1 | 992 | |
| REG2PATT | U | AABBCCDD | 1 | 991 | 1012 1016 1026 1040 |
| RPTDWSAV | D | 00000BD0 | 8 | 828 | 815 819 |
| RPTSAVE | F | 00000BC8 | 4 | 825 | 789 822 |
| RPTSPEED | I | 00000B60 | 4 | 789 | 773 |
| RPTSVR5 | F | 00000BCC | 4 | 826 | 790 821 |
| SAVE3T5 | F | 00000410 | 4 | 153 | 232 771 |
| SAVER13 | F | 00000424 | 4 | 155 | 201 777 |
| SAVER2 | F | 00000420 | 4 | 154 | |
| SUBDWORD | I | 00000C34 | 4 | 864 | 797 850 |
| SUBDWSAV | D | 00000C58 | 8 | 877 | 864 874 |
| SUBTEST | X | 00000401 | 1 | 147 | 134 |
| TEST91 | I | 00000528 | 4 | 194 | 122 |
| TESTADDR | D | 00000400 | 8 | 145 | |
| TESTNUM | X | 00000400 | 1 | 146 | 131 204 |
| TICKSAAA | P | 00000D88 | 8 | 951 | 802 805 |
| TICKSBBB | P | 00000D90 | 8 | 952 | 803 807 |
| TICKSTOT | P | 00000D98 | 8 | 953 | 805 806 807 810 |
| TIMEOPT | X | 00000408 | 1 | 150 | 128 194 |
| TNUM | X | 00000000 | 1 | 966 | 203 |
| TRTOP10 | X | 00000E94 | 4 | 1054 | 1009 |
| TRTOP1F0 | X | 00000F94 | 4 | 1056 | 1023 |
| TRTOP1F1 | X | 00001094 | 4 | 1058 | 1037 |
| TRTOP20 | X | 00001194 | 1 | 1064 | 1010 |
| TRTOP2F0 | X | 000012A0 | 1 | 1067 | 1024 |
| TRTOP8F1 | X | 000013A8 | 1 | 1070 | 1038 |
| TRTR2TST | J | 00000000 | 5296 | 73 | 76 80 84 74 |
| TRTRNEXT | U | 00000034 | 1 | 989 | 778 |
| TRTRPERF | A | 00000DF0 | 4 | 998 | 197 |
| TRTRTEST | 4 | 00000000 | 52 | 965 | 198 |
| TST91LOP | U | 00000532 | 1 | 200 | 780 |
| =AL2(L'MSGMSG) | R | 00000D56 | 2 | 933 | 889 |
| =CL5'TRTR' | C | 00000D58 | 5 | 934 | 772 |
| =F'0' | F | 00000D4C | 4 | 930 | 779 |
| =F'1' | F | 00000D50 | 4 | 931 | 870 |
| =H'0' | H | 00000D54 | 2 | 932 | 884 |
| =P'4294967296' | P | 00000D5D | 6 | 935 | 806 |

```
 MACRO    DEFN  REFERENCES

DOINSTR    177    458    469    760
OVERONLY   164    235    244    439
```

```
   DESC      SYMBOL    SIZE      POS         ADDR

Entry: 0

Image       IMAGE     5296  0000-14AF  0000-14AF
  Region              5296  0000-14AF  0000-14AF
    CSECT   TRTR2TST  5296  0000-14AF  0000-14AF
```

```
   STMT                  FILE NAME
1     /devstor/dev/tests/TRTR-02-performance.asm

** NO ERRORS FOUND **
```