```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                     2 ***********************************************************************
                                                     3 *
                                                     4 *                TRTE Performance instruction tests
                                                     5 *
                                                     6 ***********************************************************************
                                                     7 *
                                                     8 *  This program ONLY tests the performance of the TRTE instructions.
                                                     9 *
                                                    10 *
                                                    11 *                     ********************
                                                    12 *                     **   IMPORTANT!   **
                                                    13 *                     ********************
                                                    14 *
                                                    15 *        This test uses the Hercules Diagnose X'008' interface
                                                    16 *        to display messages and thus your .tst runtest script
                                                    17 *        MUST contain a "DIAG8CMD ENABLE" statement within it!
                                                    18 *
                                                    19 *
                                                    20 *  NOTE: This test is based on the CLCL-et-al Test but modified to
                                                    21 *        only test the TRTE instruction. -- James Wekel October 2022
                                                    22 *
                                                    23 ***********************************************************************
                                                    24 *
                                                    25 *  Example Hercules Testcase:
                                                    26 *
                                                    27 *
                                                    28 *      *Testcase TRTE-02-performance (Test TRTE instructions)
                                                    29 *
                                                    30 *      mainsize    16
                                                    31 *      numcpu      1
                                                    32 *      sysclear
                                                    33 *      archlvl     z/Arch
                                                    34 *      loadcore    "$(testpath)/TRTE-02-performance.core" 0x0
                                                    35 *      diag8cmd    enable    # (needed for messages to Hercules console)
                                                    36 *      #r           408=ff    # (enable timing tests)
                                                    37 *      runtest     200       # (test duration, depends on host)
                                                    38 *      diag8cmd    disable   # (reset back to default)
                                                    39 *      *Done
                                                    40 *
                                                    41 *
                                                    42 ***********************************************************************
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    44 ***********************************************************************
                                                    45 *
                                                    46 *  Tests:
                                                    47 *
                                                    48 *        All tests are ' TRTE  R2,R4,12 '
                                                    49 *        where the FC table is 128K in length,
                                                    50 *        FC is 2 bytes and an argument length of 2 bytes.
                                                    51 *
                                                    52 *        M3=12 requires page crossover tests for both FC and
                                                    53 *        the argument and has the worst performance compared to
                                                    54 *        M3=0 with the FC table and operand contained within
                                                    55 *        a page. The test should provide a lower bound on
                                                    56 *        performance improvement.
                                                    57 *
                                                    58 *           1. TRTE of 512 bytes
                                                    59 *           2. TRTE of 512 bytes that crosses a page boundary,
                                                    60 *              which results in a CC=3, and a branch back
                                                    61 *              to complete the TRTE instruction.
                                                    62 *           3. TRTE of 2048 bytes
                                                    63 *           4. TRTE of 2048 bytes that crosses a page boundary,
                                                    64 *              which results in a CC=3, and a branch back
                                                    65 *              to complete the TRTE instruction
                                                    66 *
                                                    67 ***********************************************************************


                              00000000  000C3BED    69 TRTE2TST START 0
00000000                      00000000              70          USING TRTE2TST,R0            Low core addressability


00000000                      00000000  000001A0    72          ORG   TRTE2TST+X'1A0'        z/Architecure RESTART PSW
000001A0  00000001 80000000                         73          DC    X'0000000180000000'
000001A8  00000000 00000200                         74          DC    AD(BEGIN)

000001B0                      000001B0  000001D0    76          ORG   TRTE2TST+X'1D0'        z/Architecure PROGRAM CHECK PSW
000001D0  00020001 80000000                         77          DC    X'0002000180000000'
000001D8  00000000 0000DEAD                         78          DC    AD(X'DEAD')


000001E0                      000001E0  00000200    80          ORG   TRTE2TST+X'200'        Start of actual test program...
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    82 ***********************************************************************
                                                    83 *               The actual "TRTE2TST" program itself...
                                                    84 ***********************************************************************
                                                    85 *
                                                    86 *  Architecture Mode: z/Arch
                                                    87 *  Register Usage:
                                                    88 *
                                                    89 *   R0       (work)
                                                    90 *   R1       (work)
                                                    91 *   R2       (work) or MSG subroutine call
                                                    92 *   R3       (work)
                                                    93 *   R4       (work)
                                                    94 *   R5       TRTETEST Base (of current test)
                                                    95 *   R5-R7    (work)
                                                    96 *   R8       (work)
                                                    97 *   R9       Second base register
                                                    98 *   R10-R12  (work)
                                                    99 *   R13      First base register
                                                   100 *   R14      Subroutine call
                                                   101 *   R15      Secondary Subroutine call or work
                                                   102 *
                                                   103 ***********************************************************************

00000200                      00000200             105          USING  BEGIN,R13        FIRST Base Register
00000200                      00001200             106          USING  BEGIN+4096,R9    SECOND Base Register

00000200  05D0                                     108 BEGIN    BALR  R13,0             Initalize FIRST base register
00000202  06D0                                     109          BCTR  R13,0             Initalize FIRST base register
00000204  06D0                                     110          BCTR  R13,0             Initalize FIRST base register

00000206  4190 D800                     00000800   112          LA    R9,2048(,R13)     Initalize SECOND base register
0000020A  4190 9800                     00000800   113          LA    R9,2048(,R9)      Initalize SECOND base register

                                                   115 ***********************************************************************
                                                   116 *        Run the performance test(s)...
                                                   117 ***********************************************************************

0000020E  45E0 D328                     00000528   119          BAL   R14,TEST91       Time TRTE instruction (speed test)


                                                   121 ***********************************************************************
                                                   122 *         Test for normal or unexpected test completion...
                                                   123 ***********************************************************************

00000212  95FF D208                     00000408   125          CLI   TIMEOPT,X'FF'    Was this a timing run?
00000216  4770 DD58                     00000F58   126          BNE   EOJ              No, timing run; just go end normally

0000021A  95FC D200                     00000400   128          CLI   TESTNUM,X'FC'    Did we end on expected test?
0000021E  4770 DD70                     00000F70   129          BNE   FAILTEST         No?! Then FAIL the test!

00000222  9599 D201                     00000401   131          CLI   SUBTEST,X'99'    Did we end on expected SUB-test?
00000226  4770 DD70                     00000F70   132          BNE   FAILTEST         No?! Then FAIL the test!

0000022A  47F0 DD58                     00000F58   134          B     EOJ              Yes, then normal completion!
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   136 ***********************************************************************
                                                   137 *        Fixed test storage locations ...
                                                   138 ***********************************************************************


0000022E                      0000022E  00000400   140          ORG   TRTE2TST+X'400'



00000400                                           142 TESTADDR DS    0D              Where test/subtest numbers will go
00000400  99                                       143 TESTNUM  DC    X'99'           Test number of active test
00000401  99                                       144 SUBTEST  DC    X'99'           Active test sub-test number

00000408                                           146          DS    0D
00000408  00                                       147 TIMEOPT  DC    X'00'           Set to non-zero to run timing tests

00000410                                           149          DS    0D
00000410  00000000 00000000                        150 SAVE1T4  DC    4F'0'
00000420  00000000                                 151 SAVER2   DC    F'0'
00000424  00000000                                 152 SAVER5   DC    F'0'


00000428                      00000428  00000528   154          ORG   *+X'100'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   156 ***********************************************************************
                                                   157 *        TEST91                 Time TRTE instruction  (speed test)
                                                   158 ***********************************************************************

00000528  91FF D208                     00000408   160 TEST91   TM    TIMEOPT,X'FF'        Is timing tests option enabled?
0000052C  078E                                     161          BZR   R14                  No, skip timing tests

0000052E  4150 DE18                     00001018   163          LA    R5,TRTEPERF          Point R5 --> testing control table
00000532                      00000000             164          USING TRTETEST,R5          What each table entry looks like
                                                   165 *
                              00000532  00000001   166 TST91LOP EQU   *
00000532  5050 D224                     00000424   167          ST    R5,SAVER5            Save current pref table base
                                                   168 *
00000536  4360 5000                     00000000   169          IC    R6,TNUM              Set test number
0000053A  4260 D200                     00000400   170          STC   R6,TESTNUM
                                                   171 *
                                                   172 **       Initialize operand data  (move data to testing address)
                                                   173 *
0000053E  58A0 5018                     00000018   174          L     R10,OP1WHERE         Where to move operand-1 data to
00000542  58B0 5008                     00000008   175          L     R11,OP1LEN           Get operand-1 length
00000546  50B0 501C                     0000001C   176          ST    R11,OP1WLEN          and save for later
0000054A  5860 5004                     00000004   177          L     R6,OP1DATA           Where op1 data is right now
0000054E  5870 5008                     00000008   178          L     R7,OP1LEN            How much of it there is
00000552  0EA6                                     179          MVCL  R10,R6
                                                   180 *
00000554  58A0 5014                     00000014   181          L     R10,OP2WHERE         Where to move operand-2 data to
00000558  58B0 5010                     00000010   182          L     R11,OP2LEN           How much of it there is
0000055C  5860 500C                     0000000C   183          L     R6,OP2DATA           Where op2 data is right now
00000560  5870 5010                     00000010   184          L     R7,OP2LEN            How much of it there is
00000564  0EA6                                     185          MVCL  R10,R6


                                                   187 *
                                                   188 **       Next, time the overhead...
                                                   189 *
00000566  5870 DD8C                     00000F8C   190          L     R7,NUMLOOPS
0000056A  B205 DD90                     00000F90   191          STCK  BEGCLOCK
0000056E  9014 D210                     00000410   192          STM   R1,R4,SAVE1T4
00000572  0560                                     193          BALR  R6,0

00000574  9814 5014                     00000014   195          LM    R1,R4,OPSWHERE       Get TRTE operands
00000578  4710 D374                     00000574   196          BC    B'0001',*-4          Not finished
0000057C  9814 5014                     00000014   197          LM    R1,R4,OPSWHERE
00000580  4710 D384                     00000584   198          BC    B'0001',*+4
                                                   199 *        .........ETC.........

                                                   201          PRINT OFF
                                                   396          PRINT ON

00000884  9814 5014                     00000014   398          LM    R1,R4,OPSWHERE
00000888  4710 D68C                     0000088C   399          BC    B'0001',*+4
0000088C  9814 5014                     00000014   400          LM    R1,R4,OPSWHERE
00000890  4710 D694                     00000894   401          BC    B'0001',*+4
                                                   402 *
00000894  0676                                     403          BCTR  R7,R6
00000896  B205 DD98                     00000F98   404          STCK  ENDCLOCK
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

0000089A  45F0 DC08                     00000E08   405              BAL   R15,CALCDUR
0000089E  D207 DDA8 DDA0      00000FA8  00000FA0   406              MVC   OVERHEAD,DURATION
                                                   407 *
                                                   408 **       Now do the actual timing run...
                                                   409 *
000008A4  5870 DD8C                     00000F8C   410              L     R7,NUMLOOPS
000008A8  B205 DD90                     00000F90   411              STCK  BEGCLOCK
000008AC  0560                                     412              BALR  R6,0
                                                   413 *
000008AE  9814 5014                     00000014   414              LM    R1,R4,OPSWHERE       Load TRTE operands
000008B2  B9BF C024                                415              TRTE  R2,R4,12             Do TRTE
000008B6  4710 D6B2                     000008B2   416              BC    B'0001',*-4          Not finished?
000008BA  9814 5014                     00000014   417              LM    R1,R4,OPSWHERE       Load TRTE operands
000008BE  B9BF C024                                418              TRTE  R2,R4,12             Do TRTE
000008C2  4710 D6BE                     000008BE   419              BC    B'0001',*-4          Not finished?
                                                   420 *            .........ETC.........

                                                   422              PRINT OFF
                                                   723              PRINT ON

00000D46  9814 5014                     00000014   725              LM    R1,R4,OPSWHERE
00000D4A  B9BF C024                                726              TRTE  R2,R4,12
00000D4E  4710 DB4A                     00000D4A   727              BC    B'0001',*-4
00000D52  9814 5014                     00000014   728              LM    R1,R4,OPSWHERE
00000D56  B9BF C024                                729              TRTE  R2,R4,12
00000D5A  4710 DB56                     00000D56   730              BC    B'0001',*-4
                                                   731 *
00000D5E  0676                                     732              BCTR  R7,R6
00000D60  B205 DD98                     00000F98   733              STCK  ENDCLOCK
                                                   734 *
00000D64  9814 D210                     00000410   735              LM    R1,R4,SAVE1T4
00000D68  D204 DDE9 DD80      00000FE9  00000F80   736              MVC   PRTLINE+33(5),=CL5'TRTE'
00000D6E  45F0 DB86                     00000D86   737              BAL   R15,RPTSPEED
                                                   738 *
                                                   739 *  more performance tests
                                                   740 *
00000D72  5850 D224                     00000424   741              L     R5,SAVER5          Restore perf table base
00000D76  4150 5034                     00000034   742              LA    R5,TRTENEXT        Go on to next table entry
00000D7A  D503 DD74 5000      00000F74  00000000   743              CLC   =F'0',0(R5)        End of table?
00000D80  4770 D332                     00000532   744              BNE   TST91LOP           No, loop...
00000D84  07FE                                     745              BR    R14                Return to caller or FAILTEST
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   747 ***********************************************************************
                                                   748 *        RPTSPEED                 Report instruction speed
                                                   749 ***********************************************************************

00000D86  50F0 DBF0                     00000DF0   751 RPTSPEED ST    R15,RPTSAVE        Save return address
00000D8A  5050 DBF4                     00000DF4   752          ST    R5,RPTSVR5         Save R5
                                                   753 *
00000D8E  45F0 DC08                     00000E08   754          BAL   R15,CALCDUR        Calculate duration
                                                   755 *
00000D92  4150 DDA8                     00000FA8   756          LA    R5,OVERHEAD        Subtract overhead
00000D96  4160 DDA0                     00000FA0   757          LA    R6,DURATION        From raw timing
00000D9A  4170 DDA0                     00000FA0   758          LA    R7,DURATION        Yielding true instruction timing
00000D9E  45F0 DC5C                     00000E5C   759          BAL   R15,SUBDWORD       Do it
                                                   760 *
00000DA2  98AB DDA0                     00000FA0   761          LM    R10,R11,DURATION   Convert to...
00000DA6  8CA0 000C                     0000000C   762          SRDL  R10,12             ... microseconds
                                                   763 *
00000DAA  4EA0 DDB0                     00000FB0   764          CVD   R10,TICKSAAA       Convert HIGH part to decimal
00000DAE  4EB0 DDB8                     00000FB8   765          CVD   R11,TICKSBBB       Convert LOW  part to decimal
                                                   766 *
00000DB2  F877 DDC0 DDB0      00000FC0  00000FB0   767          ZAP   TICKSTOT,TICKSAAA            Calculate...
00000DB8  FC75 DDC0 DD85      00000FC0  00000F85   768          MP    TICKSTOT,=P'4294967296'      ...decimal...
00000DBE  FA77 DDC0 DDB8      00000FC0  00000FB8   769          AP    TICKSTOT,TICKSBBB            ...microseconds
                                                   770 *
00000DC4  D20B DDF3 DE0C      00000FF3  0000100C   771          MVC   PRTLINE+43(L'EDIT),EDIT          (edit into...
00000DCA  DE0B DDF3 DDC3      00000FF3  00000FC3   772          ED    PRTLINE+43(L'EDIT),TICKSTOT+3     ...print line)


                                                   774 *
                                                   775 *        Use Hercules Diagnose for Message to console
                                                   776 *
00000DD0  9002 DBF8                     00000DF8   777          STM   R0,R2,RPTDWSAV     Save regs used by MSG
00000DD4  4100 0044                     00000044   778          LA    R0,PRTLNG          Message length
00000DD8  4110 DDC8                     00000FC8   779          LA    R1,PRTLINE         Message address
00000DDC  4520 DC90                     00000E90   780          BAL   R2,MSG             Call Hercules console MSG display
00000DE0  9802 DBF8                     00000DF8   781          LM    R0,R2,RPTDWSAV     Restore regs

00000DE4  5850 DBF4                     00000DF4   783          L     R5,RPTSVR5         Restore R5
00000DE8  58F0 DBF0                     00000DF0   784          L     R15,RPTSAVE        Restore return address
00000DEC  07FF                                     785          BR    R15                Return to caller

00000DF0  00000000                                 787 RPTSAVE  DC    F'0'               R15 save area
00000DF4  00000000                                 788 RPTSVR5  DC    F'0'               R5 save area

00000DF8  00000000 00000000                        790 RPTDWSAV DC    2D'0'              R0-R2 save area for MSG call
```

```
  LOC       OBJECT CODE       ADDR1     ADDR2    STMT

                                                   792 ***********************************************************************
                                                   793 *        CALCDUR                  Calculate DURATION
                                                   794 ***********************************************************************

00000E08  50F0 DC4C                     00000E4C   796 CALCDUR  ST    R15,CALCRET        Save return address
00000E0C  9057 DC50                     00000E50   797          STM   R5,R7,CALCWORK     Save work registers
                                                   798 *
00000E10  9867 DD90                     00000F90   799          LM    R6,R7,BEGCLOCK     Remove CPU number from clock value
00000E14  8C60 0006                     00000006   800          SRDL  R6,6                            "
00000E18  8D60 0006                     00000006   801          SLDL  R6,6                            "
00000E1C  9067 DD90                     00000F90   802          STM   R6,R7,BEGCLOCK                  "
                                                   803 *
00000E20  9867 DD98                     00000F98   804          LM    R6,R7,ENDCLOCK     Remove CPU number from clock value
00000E24  8C60 0006                     00000006   805          SRDL  R6,6                            "
00000E28  8D60 0006                     00000006   806          SLDL  R6,6                            "
00000E2C  9067 DD98                     00000F98   807          STM   R6,R7,ENDCLOCK                  "
                                                   808 *
00000E30  4150 DD90                     00000F90   809          LA    R5,BEGCLOCK        Starting time
00000E34  4160 DD98                     00000F98   810          LA    R6,ENDCLOCK        Ending time
00000E38  4170 DDA0                     00000FA0   811          LA    R7,DURATION        Difference
00000E3C  45F0 DC5C                     00000E5C   812          BAL   R15,SUBDWORD       Calculate duration
                                                   813 *
00000E40  9857 DC50                     00000E50   814          LM    R5,R7,CALCWORK     Restore work registers
00000E44  58F0 DC4C                     00000E4C   815          L     R15,CALCRET        Restore return address
00000E48  07FF                                     816          BR    R15                Return to caller

00000E4C  00000000                                 818 CALCRET  DC    F'0'               R15 save area
00000E50  00000000 00000000                        819 CALCWORK DC    3F'0'              R5-R7 save area



                                                   821 ***********************************************************************
                                                   822 *        SUBDWORD                 Subtract two doublewords
                                                   823 *        R5 --> subtrahend, R6 --> minuend, R7 --> result
                                                   824 ***********************************************************************

00000E5C  9014 DC80                     00000E80   826 SUBDWORD STM   R1,R4,SUBDWSAV     Save registers
                                                   827 *
00000E60  9812 5000                     00000000   828          LM    R1,R2,0(R5)        Subtrahend  (value to subtract)
00000E64  9834 6000                     00000000   829          LM    R3,R4,0(R6)        Minuend     (what to subtract FROM)
00000E68  1F42                                     830          SLR   R4,R2              Subtract LOW part
00000E6A  47B0 DC72                     00000E72   831          BNM   *+4+4              (branch if no borrow)
00000E6E  5F30 DD78                     00000F78   832          SL    R3,=F'1'           (otherwise do borrow)
00000E72  1F31                                     833          SLR   R3,R1              Subtract HIGH part
00000E74  9034 7000                     00000000   834          STM   R3,R4,0(R7)        Store results
                                                   835 *
00000E78  9814 DC80                     00000E80   836          LM    R1,R4,SUBDWSAV     Restore registers
00000E7C  07FF                                     837          BR    R15                Return to caller

00000E80  00000000 00000000                        839 SUBDWSAV DC    2D'0'              R1-R4 save area
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   841 ***********************************************************************
                                                   842 *        Issue HERCULES MESSAGE pointed to by R1, length in R0
                                                   843 *              R2 = return address
                                                   844 ***********************************************************************

00000E90  4900 DD7C                     00000F7C   846 MSG      CH    R0,=H'0'               Do we even HAVE a message?
00000E94  07D2                                     847          BNHR  R2                     No, ignore

00000E96  9002 DCC8                     00000EC8   849          STM   R0,R2,MSGSAVE          Save registers

00000E9A  4900 DD7E                     00000F7E   851          CH    R0,=AL2(L'MSGMSG)      Message length within limits?
00000E9E  47D0 DCA6                     00000EA6   852          BNH   MSGOK                  Yes, continue
00000EA2  4100 005F                     0000005F   853          LA    R0,L'MSGMSG            No, set to maximum

00000EA6  1820                                     855 MSGOK    LR    R2,R0                  Copy length to work register
00000EA8  0620                                     856          BCTR  R2,0                   Minus-1 for execute
00000EAA  4420 DCD4                     00000ED4   857          EX    R2,MSGMVC              Copy message to O/P buffer

00000EAE  4120 200A                     0000000A   859          LA    R2,1+L'MSGCMD(,R2)     Calculate true command length
00000EB2  4110 DCDA                     00000EDA   860          LA    R1,MSGCMD              Point to true command

00000EB6  83120008                                 862          DC    X'83',X'12',X'0008'    Issue Hercules Diagnose X'008'
00000EBA  4780 DCC0                     00000EC0   863          BZ    MSGRET                 Return if successful
00000EBE  0000                                     864          DC    H'0'                   CRASH for debugging purposes

00000EC0  9802 DCC8                     00000EC8   866 MSGRET   LM    R0,R2,MSGSAVE          Restore registers
00000EC4  07F2                                     867          BR    R2                     Return to caller







00000EC8  00000000 00000000                        869 MSGSAVE  DC    3F'0'                  Registers save area
00000ED4  D200 DCE3 1000      00000EE3  00000000   870 MSGMVC   MVC   MSGMSG(0),0(R1)        Executed instruction


00000EDA  D4E2C7D5 D6C8405C                        872 MSGCMD   DC    C'MSGNOH * '           *** HERCULES MESSAGE COMMAND ***
00000EE3  40404040 40404040                        873 MSGMSG   DC    CL95' '                The message text to be displayed
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT
                                                   875 ***********************************************************************
                                                   876 *        Normal completion or Abnormal termination PSWs
                                                   877 ***********************************************************************


00000F48  00020001 80000000                        879 EOJPSW   DC    0D'0',X'0002000180000000',AD(0)

00000F58  B2B2 DD48                     00000F48   881 EOJ      LPSWE EOJPSW              Normal completion


00000F60  00020001 80000000                        883 FAILPSW  DC    0D'0',X'0002000180000000',AD(X'BAD')

00000F70  B2B2 DD60                     00000F60   885 FAILTEST LPSWE FAILPSW             Abnormal termination
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   887 ***********************************************************************
                                                   888 *        Working Storage
                                                   889 ***********************************************************************

00000F74                                           891          LTORG ,                Literals pool
00000F74  00000000                                 892                =F'0'
00000F78  00000001                                 893                =F'1'
00000F7C  0000                                     894                =H'0'
00000F7E  005F                                     895                =AL2(L'MSGMSG)
00000F80  E3D9E3C5 40                              896                =CL5'TRTE'
00000F85  04294967 296C                            897                =P'4294967296'


                                    00000400  00000001   899 K        EQU   1024             One KB
                                    00001000  00000001   900 PAGE     EQU   (4*K)            Size of one page
                                    00010000  00000001   901 K64      EQU   (64*K)           64 KB
                                    00100000  00000001   902 MB       EQU   (K*K)             1 MB

00000F8C  00002710                                 904 NUMLOOPS DC    F'10000'         10,000 * 100 = 1,000,000

00000F90  BBBBBBBB BBBBBBBB                        906 BEGCLOCK DC    0D'0',8X'BB'     Begin
00000F98  EEEEEEEE EEEEEEEE                        907 ENDCLOCK DC    0D'0',8X'EE'     End
00000FA0  DDDDDDDD DDDDDDDD                        908 DURATION DC    0D'0',8X'DD'     Diff
00000FA8  FFFFFFFF FFFFFFFF                        909 OVERHEAD DC    0D'0',8X'FF'     Overhead

00000FB0  00000000 0000000C                        911 TICKSAAA DC    PL8'0'           Clock ticks high part
00000FB8  00000000 0000000C                        912 TICKSBBB DC    PL8'0'           Clock ticks low part
00000FC0  00000000 0000000C                        913 TICKSTOT DC    PL8'0'           Total clock ticks

00000FC8  40404040 40404040                        915 PRTLINE  DC    C'         1,000,000 iterations of XXXXX'
00000FEE  40A39696 9240F9F9                        916          DC    C' took 999,999,999 microseconds'
                                    00000044  00000001   917 PRTLNG   EQU   *-PRTLINE
0000100C  40202020 6B202020                        918 EDIT     DC    X'402020206B2020206B202120'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   920 ***********************************************************************
                                                   921 *        TRTETEST DSECT
                                                   922 ***********************************************************************


                                                   924 TRTETEST DSECT ,
00000000  00                                       925 TNUM     DC    X'00'          TRTE table Number
00000001  00                                       926          DC    X'00'
00000002  00                                       927          DC    X'00'
00000003  00                                       928 M3       DC    X'00'          M3 byte stored into TRTE instruction


00000004  00000000                                 930 OP1DATA  DC    A(0)           Pointer to Operand-1 data
00000008  00000000                                 931 OP1LEN   DC    F'0'           How much data is there - 1
0000000C  00000000                                 932 OP2DATA  DC    A(0)           Pointer to FC table data
00000010  00000000                                 933 OP2LEN   DC    F'0'           How much data is there - FC Table

                              00000014  00000001   935 OPSWHERE EQU   *
00000014  00000000                                 936 OP2WHERE DC    A(0)           Where FC Table  data should be placed
00000018  00000000                                 937 OP1WHERE DC    A(0)           Where Operand-1 data should be placed
0000001C  00000000                                 938 OP1WLEN  DC    F'0'           How much data is there - 1
00000020  00000000                                 939          DC    A(0)           pollute - found FC

00000024  00000000                                 941 FAILMASK DC    A(0)           Failure Branch on Condition mask

                                                   943 *                             Ending register values
00000028  00000000                                 944 ENDREGS  DC    A(0)              Operand 1 address
0000002C  00000000                                 945          DC    A(0)              Operand 1 length
00000030  00000000                                 946          DC    A(0)              Function Code

                              00000034  00000001   948 TRTENEXT EQU   *              Start of next table entry...


                              AABBCCDD  00000001   950 REG2PATT EQU   X'AABBCCDD'    Polluted Register pattern
                              000000DD  00000001   951 REG2LOW  EQU         X'DD'    (last byte above)
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                   953 ***********************************************************************
                                                   954 *        TRTE Performace Test data...
                                                   955 ***********************************************************************

                              00000000  000C3BED   957 TRTE2TST CSECT ,
00001018                                           958 TRTEPERF DC    0A(0)                      Start of table



                                                   960 ***********************************************************************
                                                   961 *        tests with   M3: A=1,F=1,L=0, reserved=0    (12)
                                                   962 *                     FC Table : SIZE: 131,072 (2 BYTE ARGUMENT)
                                                   963 *                                Function Code is 2 bytes
                                                   964 *
                                                   965 *              Note: Op1 length must be a multiple of 2
                                                   966 ***********************************************************************


00001018                                           968 F12T8    DS    0F
00001018  F8                                       969          DC    X'F8'                       Test Num
00001019  0000                                     970          DC    X'00',X'00'
0000101B  C0                                       971          DC    X'C0'                       M3: A=1,F=1,L=0,--=0
0000101C  000013F0 00000200                        972          DC    A(TRTOP1F1),A(512)          Source - Op 1 & length
00001024  000A39EE 00020000                        973          DC    A(TRTOPCF1),A(2*K64)        Source - FC Table & length
                                                   974 *                                          Target -
0000102C  00710000 00910000                        975          DC    A(7*MB+(1*K64)),A(9*MB+(1*K64)),A(0)  FC, Op1, Op1L
00001038  AABBCCDD                                 976          DC    A(REG2PATT)
0000103C  0000000B                                 977          DC    A(11) CC1
00001040  009101FE 00000002                        978          DC    A(9*MB+(1*K64)+510),A(2),XL4'F1'



0000104C                                           980 F12T8A   DS    0F
0000104C  F9                                       981          DC    X'F9'                       Test Num
0000104D  0000                                     982          DC    X'00',X'00'
0000104F  C0                                       983          DC    X'C0'                       M3: A=1,F=1,L=0,--=0
00001050  000013F0 00000200                        984          DC    A(TRTOP1F1),A(512)          Source - Op 1 & length
00001058  000A39EE 00020000                        985          DC    A(TRTOPCF1),A(2*K64)        Source - FC Table & length
                                                   986 *                                          Target - FC, Op1, Op1L
00001060  0072FF81 0092FF81                        987          DC    A(7*MB+(3*K64)-127),A(9*MB+(3*K64)-127),A(0)
0000106C  AABBCCDD                                 988          DC    A(REG2PATT)
00001070  0000000A                                 989          DC    A(10) CC1 or CC3
00001074  0093017F 00000002                        990          DC    A(9*MB+(3*K64)-127+510),A(2),XL4'F1'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00001080                                           992 F12T11   DS    0F
00001080  FB                                       993          DC    X'FB'                       Test Num
00001081  0000                                     994          DC    X'00',X'00'
00001083  C0                                       995          DC    X'C0'                       M3: A=1,F=1,L=0,--=0
00001084  000025F0 00000800                        996          DC    A(TRTO1LF0),A(2048)         Source - Op 1 & length
0000108C  000837F0 00020000                        997          DC    A(TRTOPCF0),A(2*K64)        Source - FC Table & length
                                                   998 *                                          Target -
00001094  00760000 00960000                        999          DC    A(7*MB+(6*K64)),A(9*MB+(6*K64)),A(0) FC, Op1, Op1L
000010A0  AABBCCDD                                1000          DC    A(REG2PATT)
000010A4  0000000B                                1001          DC    A(11) CC1
000010A8  009607FE 00000002                       1002          DC    A(9*MB+(6*K64)+2048-2),A(2),XL4'F0'




000010B4                                          1004 F12T11A  DS    0F
000010B4  FC                                      1005          DC    X'FC'                       Test Num
000010B5  0000                                    1006          DC    X'00',X'00'
000010B7  C0                                      1007          DC    X'C0'                       M3: A=1,F=1,L=0,--=0
000010B8  000025F0 00000800                       1008          DC    A(TRTO1LF0),A(2048)         Source - Op 1 & length
000010C0  000837F0 00020000                       1009          DC    A(TRTOPCF0),A(2*K64)        Source - FC Table & length
                                                  1010 *                                          Target - FC, Op1, Op1L
000010C8  0078FE1F 0098FE1F                       1011          DC    A(7*MB+(9*K64)-481),A(9*MB+(9*K64)-481),A(0)
000010D4  AABBCCDD                                1012          DC    A(REG2PATT)
000010D8  0000000A                                1013          DC    A(10) CC1 or CC3
000010DC  0099061D 00000002                       1014          DC    A(9*MB+(9*K64)-481+2048-2),A(2),XL4'F0'




000010E8  00000000                                1016          DC    A(0)        end of table
000010EC  00000000                                1017          DC    A(0)        end of table
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                  1019 ***********************************************************************
                                                  1020 *        TRTE op1 scan data...
                                                  1021 ***********************************************************************

000010F0  78125634 78125634                       1023 TRTOP10  DC    64XL4'78125634'                               (CC0)

000011F0  78125634 78125634                       1025 TRTOP111 DC    04XL4'78125634',X'00110000',59XL4'78125634'   (CC1)

000012F0  78125634 78125634                       1027 TRTOP1F0 DC    63XL4'78125634',X'000000F0'                   (CC1)

000013F0  78125634 78125634                       1029 TRTOP1F1 DC    127XL4'78125634',X'000000F1'                  (CC1)

000015F0  98765432 98765432                       1031 TRTO1L0  DC    512XL4'98765432'                              (CC0)

00001DF0  98765432 98765432                       1033 TRTO1L11 DC    256XL4'98765432',X'00110000',255XL4'98765432' (CC1)

000025F0  98765432 98765432                       1035 TRTO1LF0 DC    511XL4'98765432',X'000000F0'                  (CC1)
```

```
  LOC         OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1037 ***********************************************************************
                                                  1038 *        Function Code (FC) Tables (GR1)
                                                  1039 ***********************************************************************

00002DF0  00000000 00000000                       1041 TRTOP20  DC    256X'00'                           no stop
00002EF0                      00002EF0  00022EF0  1042          ORG   *+2*K64

00022EF0  00000000 00000000                       1044 TRTOP211 DC    17X'00',X'11',238X'00'             stop on X'11'

00022FF0  00000000 00000000                       1046 TRTOP2F0 DC    240X'00',X'F0',15X'00'             stop on X'F0'

000230F0  00000000 00000000                       1048 TRTOP411 DC    34X'00',X'0011',476X'00'           stop on X'11'

000232F0  00000000 00000000                       1050 TRTOP4F0 DC    480X'00',X'00F0',30X'00'           stop on X'F0'

000234F0  00000000 00000000                       1052 TRTOP811 DC    17X'00',X'11',238X'00'             stop on X'11'
000235F0                      000235F0  000435F0  1053          ORG   *+2*K64

000435F0  00000000 00000000                       1055 TRTOP8F0 DC    240X'00',X'F0',15X'00'             stop on X'F0'
000436F0                      000436F0  000636F0  1056          ORG   *+2*K64

000636F0  00000000 00000000                       1058 TRTOP8F1 DC    240X'00',X'00',X'F1',14X'00'       stop on X'F1'
000637F0                      000637F0  000837F0  1059          ORG   *+2*K64

000837F0  00000000 00000000                       1061 TRTOPCF0 DC    480X'00',X'00F0',28X'00'           stop on X'F0'
000839EE                      000839EE  000A39EE  1062          ORG   *+2*K64

000A39EE  00000000 00000000                       1064 TRTOPCF1 DC    480X'00',X'0000',X'00F1',28X'00'   stop on X'F1'
000A3BEE                      000A3BEE  000C3BEE  1065          ORG   *+2*K64
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                  1067 ***********************************************************************
                                                  1068 *        Register equates
                                                  1069 ***********************************************************************

                              00000000  00000001  1071 R0       EQU   0
                              00000001  00000001  1072 R1       EQU   1
                              00000002  00000001  1073 R2       EQU   2
                              00000003  00000001  1074 R3       EQU   3
                              00000004  00000001  1075 R4       EQU   4
                              00000005  00000001  1076 R5       EQU   5
                              00000006  00000001  1077 R6       EQU   6
                              00000007  00000001  1078 R7       EQU   7
                              00000008  00000001  1079 R8       EQU   8
                              00000009  00000001  1080 R9       EQU   9
                              0000000A  00000001  1081 R10      EQU   10
                              0000000B  00000001  1082 R11      EQU   11
                              0000000C  00000001  1083 R12      EQU   12
                              0000000D  00000001  1084 R13      EQU   13
                              0000000E  00000001  1085 R14      EQU   14
                              0000000F  00000001  1086 R15      EQU   15


                                                  1088          END
```

```
     SYMBOL        TYPE   VALUE      LENGTH    DEFN  REFERENCES

BEGCLOCK            D    00000F90           8   906   191   411   799   802   809
BEGIN               I    00000200           2   108    74   105   106
CALCDUR             I    00000E08           4   796   405   754
CALCRET             F    00000E4C           4   818   796   815
CALCWORK            F    00000E50           4   819   797   814
DURATION            D    00000FA0           8   908   406   757   758   761   811
EDIT                X    0000100C          12   918   771   772
ENDCLOCK            D    00000F98           8   907   404   733   804   807   810
ENDREGS             A    00000028           4   944
EOJ                 I    00000F58           4   881   126   134
EOJPSW              D    00000F48           8   879   881
F12T11              F    00001080           4   992
F12T11A             F    000010B4           4  1004
F12T8               F    00001018           4   968
F12T8A              F    0000104C           4   980
FAILMASK            A    00000024           4   941
FAILPSW             D    00000F60           8   883   885
FAILTEST            I    00000F70           4   885   129   132
IMAGE               1    00000000      801774     0
K                   U    00000400           1   899   900   901   902
K64                 U    00010000           1   901  1042  1053  1056  1059  1062  1065   973   975   978   985   987   990   997
                                                      999  1002  1009  1011  1014
M3                  X    00000003           1   928
MB                  U    00100000           1   902   975   978   987   990   999  1002  1011  1014
MSG                 I    00000E90           4   846   780
MSGCMD              C    00000EDA           9   872   859   860
MSGMSG              C    00000EE3          95   873   853   870   851
MSGMVC              I    00000ED4           6   870   857
MSGOK               I    00000EA6           2   855   852
MSGRET              I    00000EC0           4   866   863
MSGSAVE             F    00000EC8           4   869   849   866
NUMLOOPS            F    00000F8C           4   904   190   410
OP1DATA             A    00000004           4   930   177
OP1LEN              F    00000008           4   931   175   178
OP1WHERE            A    00000018           4   937   174
OP1WLEN             F    0000001C           4   938   176
OP2DATA             A    0000000C           4   932   183
OP2LEN              F    00000010           4   933   182   184
OP2WHERE            A    00000014           4   936   181
OPSWHERE            U    00000014           1   935   195   197   203   205   207   209   211   213   215   217   219   221   223
                                                      225   227   229   231   233   235   237   239   241   243   245   247   249
                                                      251   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   398   400   414   417   425   428
                                                      431   434   437   440   443   446   449   452   456   459   462   465   468
                                                      471   474   477   480   483   487   490   493   496   499   502   505   508
                                                      511   514   518   521   524   527   530   533   536   539   542   545   549
                                                      552   555   558   561   564   567   570   573   576   580   583   586   589
                                                      592   595   598   601   604   607   611   614   617   620   623   626   629
                                                      632   635   638   642   645   648   651   654   657   660   663   666   669
                                                      673   676   679   682   685   688   691   694   697   700   704   707   710
                                                      713   716   719   725   728
OVERHEAD            D    00000FA8           8   909   406   756
```

```
     SYMBOL        TYPE   VALUE      LENGTH    DEFN  REFERENCES

PAGE                U    00001000           1   900
PRTLINE             C    00000FC8          38   915   917   736   771   772   779
PRTLNG              U    00000044           1   917   778
R0                  U    00000000           1  1071    70   777   778   781   846   849   851   853   855   866
R1                  U    00000001           1  1072   192   195   197   203   205   207   209   211   213   215   217   219   221
                                                   223   225   227   229   231   233   235   237   239   241   243   245   247
                                                   249   251   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   398   400   414   417   425
                                                   428   431   434   437   440   443   446   449   452   456   459   462   465
                                                   468   471   474   477   480   483   487   490   493   496   499   502   505
                                                   508   511   514   518   521   524   527   530   533   536   539   542   545
                                                   549   552   555   558   561   564   567   570   573   576   580   583   586
                                                   589   592   595   598   601   604   607   611   614   617   620   623   626
                                                   629   632   635   638   642   645   648   651   654   657   660   663   666
                                                   669   673   676   679   682   685   688   691   694   697   700   704   707
                                                   710   713   716   719   725   728   735   779   826   828   833   836   860
                                                   870
R10                 U    0000000A           1  1081   174   179   181   185   761   762   764
R11                 U    0000000B           1  1082   175   176   182   761   765
R12                 U    0000000C           1  1083
R13                 U    0000000D           1  1084   105   108   109   110   112
R14                 U    0000000E           1  1085   119   161   745
R15                 U    0000000F           1  1086   405   737   751   754   759   784   785   796   812   815   816   837
R2                  U    00000002           1  1073   415   418   426   429   432   435   438   441   444   447   450   453   457
                                                   460   463   466   469   472   475   478   481   484   488   491   494   497
                                                   500   503   506   509   512   515   519   522   525   528   531   534   537
                                                   540   543   546   550   553   556   559   562   565   568   571   574   577
                                                   581   584   587   590   593   596   599   602   605   608   612   615   618
                                                   621   624   627   630   633   636   639   643   646   649   652   655   658
                                                   661   664   667   670   674   677   680   683   686   689   692   695   698
                                                   701   705   708   711   714   717   720   726   729   777   780   781   828
                                                   830   847   849   855   856   857   859   866   867
R3                  U    00000003           1  1074   829   832   833   834
R4                  U    00000004           1  1075   192   195   197   203   205   207   209   211   213   215   217   219   221
                                                   223   225   227   229   231   233   235   237   239   241   243   245   247
                                                   249   251   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   398   400   414   415   417
                                                   418   425   426   428   429   431   432   434   435   437   438   440   441
                                                   443   444   446   447   449   450   452   453   456   457   459   460   462
                                                   463   465   466   468   469   471   472   474   475   477   478   480   481
                                                   483   484   487   488   490   491   493   494   496   497   499   500   502
                                                   503   505   506   508   509   511   512   514   515   518   519   521   522
                                                   524   525   527   528   530   531   533   534   536   537   539   540   542
                                                   543   545   546   549   550   552   553   555   556   558   559   561   562
                                                   564   565   567   568   570   571   573   574   576   577   580   581   583
                                                   584   586   587   589   590   592   593   595   596   598   599   601   602
                                                   604   605   607   608   611   612   614   615   617   618   620   621   623
                                                   624   626   627   629   630   632   633   635   636   638   639   642   643
     SYMBOL        TYPE   VALUE      LENGTH    DEFN  REFERENCES

                                                      645   646   648   649   651   652   654   655   657   658   660   661   663
                                                      664   666   667   669   670   673   674   676   677   679   680   682   683
                                                      685   686   688   689   691   692   694   695   697   698   700   701   704
                                                      705   707   708   710   711   713   714   716   717   719   720   725   726
                                                      728   729   735   826   829   830   834   836
R5                  U    00000005           1  1076   163   164   167   741   742   743   752   756   783   797   809   814   828
R6                  U    00000006           1  1077   169   170   177   179   183   185   193   403   412   732   757   799   800
                                                      801   802   804   805   806   807   810   829
R7                  U    00000007           1  1078   178   184   190   403   410   732   758   797   799   802   804   807   811
                                                      814   834
R8                  U    00000008           1  1079
R9                  U    00000009           1  1080   106   112   113
REG2LOW             U    000000DD           1   951
REG2PATT            U    AABBCCDD           1   950   976   988  1000  1012
RPTDWSAV            D    00000DF8           8   790   777   781
RPTSAVE             F    00000DF0           4   787   751   784
RPTSPEED            I    00000D86           4   751   737
RPTSVR5             F    00000DF4           4   788   752   783
SAVE1T4             F    00000410           4   150   192   735
SAVER2              F    00000420           4   151
SAVER5              F    00000424           4   152   167   741
SUBDWORD            I    00000E5C           4   826   759   812
SUBDWSAV            D    00000E80           8   839   826   836
SUBTEST             X    00000401           1   144   131
TEST91              I    00000528           4   160   119
TESTADDR            D    00000400           8   142
TESTNUM             X    00000400           1   143   128   170
TICKSAAA            P    00000FB0           8   911   764   767
TICKSBBB            P    00000FB8           8   912   765   769
TICKSTOT            P    00000FC0           8   913   767   768   769   772
TIMEOPT             X    00000408           1   147   125   160
TNUM                X    00000000           1   925   169
TRTE2TST            J    00000000      801774    69    72    76    80   140    70
TRTENEXT            U    00000034           1   948   742
TRTEPERF            A    00001018           4   958   163
TRTETEST            4    00000000          52   924   164
TRTO1L0             X    000015F0           4  1031
TRTO1L11            X    00001DF0           4  1033
TRTO1LF0            X    000025F0           4  1035   996  1008
TRTOP10             X    000010F0           4  1023
TRTOP111            X    000011F0           4  1025
TRTOP1F0            X    000012F0           4  1027
TRTOP1F1            X    000013F0           4  1029   972   984
TRTOP20             X    00002DF0           1  1041
TRTOP211            X    00022EF0           1  1044
TRTOP2F0            X    00022FF0           1  1046
TRTOP411            X    000230F0           1  1048
TRTOP4F0            X    000232F0           1  1050
TRTOP811            X    000234F0           1  1052
TRTOP8F0            X    000435F0           1  1055
TRTOP8F1            X    000636F0           1  1058
TRTOPCF0            X    000837F0           1  1061   997  1009
TRTOPCF1            X    000A39EE           1  1064   973   985
TST91LOP            U    00000532           1   166   744
=AL2(L'MSGMSG)      R    00000F7E           2   895   851
=CL5'TRTE'          C    00000F80           5   896   736
```

```
     SYMBOL        TYPE   VALUE      LENGTH    DEFN  REFERENCES

=F'0'               F    00000F74           4   892   743
=F'1'               F    00000F78           4   893   832
=H'0'               H    00000F7C           2   894   846
=P'4294967296'      P    00000F85           6   897   768
```

```
MACRO  DEFN  REFERENCES

No defined macros
```

```
   DESC     SYMBOL    SIZE       POS          ADDR

Entry: 0

Image      IMAGE     801774  00000-C3BED  00000-C3BED
  Region             801774  00000-C3BED  00000-C3BED
    CSECT  TRTE2TST  801774  00000-C3BED  00000-C3BED
   STMT                                                        FILE NAME

1     c:\Users\Fish\Documents\Visual Studio 2008\Projects\MyProjects\ASMA-0\TRTE-02-performance\TRTE-02-performance.asm

** NO ERRORS FOUND **
```