```
 LOC   OBJECT CODE   ADDR1   ADDR2   STMT

                                       2 ********************************************************************************
                                       3 *
                                       4 *                              ShiftLeft
                                       5 *
                                       6 ********************************************************************************
                                       7 *
                                       8 *  This program tests the algebraic "Shift Left" instructions
                                       9 *
                                      10 *                       SLA, SLDA, SLAK, SLAG
                                      11 *
                                      12 *  to ensure proper results and setting of Condition Code.
                                      13 *
                                      14 *  The original implementation of these instructions in Hercules was
                                      15 *  determined to be relatively inefficient, so efforts were made to
                                      16 *  try and speed them up. This test verifies that the instructions
                                      17 *  still produce correct results.
                                      18 *
                                      19 ********************************************************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT
                                                     21 ********************************************************************************
                                                     22 *                                LOW CORE
                                                     23 ********************************************************************************

                                00000000  0000053B    25 SHIFTEST START 0

00000000                        00000000              27          USING *,R0               Use absolute addressing


00000000                        00000000  000001A0    29          ORG   SHIFTEST+X'1A0'    z/Arch Restart new PSW

000001A0  00000001                                    31          DC    XL4'00000001'
000001A4  80000000                                    32          DC    XL4'80000000'
000001A8  00000000                                    33          DC    XL4'00000000'
000001AC  000001E0                                    34          DC    A(BEGIN)


000001B0                        000001B0  000001D0    36          ORG   SHIFTEST+X'1D0'    z/Arch Program new PSW

000001D0  00020001                                    38          DC    XL4'00020001'
000001D4  80000000                                    39          DC    XL4'80000000'
000001D8  00000000                                    40          DC    XL4'00000000'
000001DC  0000DEAD                                    41          DC    A(X'DEAD')
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    43 ********************************************************************************
                                                    44 *                          Main Program
                                                    45 ********************************************************************************

000001E0                                            47 BEGIN    DS    0H

000001E0  45E0 0220                       00000220  49          BAL   R14,SLA           Test Shift Left Single
000001E4  45E0 0256                       00000256  50          BAL   R14,SLDA          Test Shift Left Double
000001E8  45E0 0296                       00000296  51          BAL   R14,SLAK          Test Shift Left Single Distinct
000001EC  45E0 02DE                       000002DE  52          BAL   R14,SLAG          Test Shift Left Single Long

000001F0  B2B2 0200                       00000200  54          LPSWE GOODPSW           Success! All tests passed!



000001F4  4BD0 034C                       0000034C  56 FAILTEST SH    R13,=H'4'         Backup to actual failure location
000001F8  B2B2 0210                       00000210  57          LPSWE FAILPSW           Abnormal termination disabled wait




00000200                                            59 GOODPSW  DC    0D'0'             Test SUCCESS disabled wait PSW
00000200  00020001                                  60          DC    XL4'00020001'
00000204  80000000                                  61          DC    XL4'80000000'
00000208  00000000 00000000                         62          DC    AD(0)


00000210                                            64 FAILPSW  DC    0D'0'               Test FAILURE disabled wait PSW
00000210  00020001                                  65          DC    XL4'00020001'
00000214  80000000                                  66          DC    XL4'80000000'
00000218  00000000 00000BAD                         67          DC    AD(X'BAD')
```

```
  LOC         OBJECT CODE       ADDR1     ADDR2     STMT

                                                       69 ************************************************************************
                                                       70 *  8B   SLA   - Shift Left Single                              [RS-a]
                                                       71 ************************************************************************
                                                       72 *
                                                       73 *    SHIFT LEFT SINGLE (SLA)
                                                       74 *
                                                       75 *    The SHIFT LEFT SINGLE instruction is similar to
                                                       76 *    SHIFT LEFT DOUBLE, except that it shifts only the
                                                       77 *    31 numeric bits of a single register. Therefore, this
                                                       78 *    instruction performs an algebraic left shift of a 32-bit
                                                       79 *    signed binary integer.
                                                       80 *
                                                       81 *    For example, if the contents of register 2 are:
                                                       82 *
                                                       83 *    00 7F 0A 72 = 00000000 01111111 00001010 01110010
                                                       84 *
                                                       85 *    The instruction:
                                                       86 *
                                                       87 *    Machine Format
                                                       88 *
                                                       89 *      0           1           2           3           4
                                                       90 *      +-----+-----+-----+-----+-----+-----+-----+-----+
                                                       91 *      |     8B    |  2  | /// |  0  |      008        |    RS-a
                                                       92 *      +-----+-----+-----+-----+-----+-----+-----+-----+
                                                       93 *
                                                       94 *    Assembler Format
                                                       95 *
                                                       96 *      Op Code  R1,D2(B2)
                                                       97 *      ------------------
                                                       98 *        SLA     2,8(0)
                                                       99 *
                                                      100 *    results in register 2 being shifted left eight bit
                                                      101 *    positions so that its new contents are:
                                                      102 *
                                                      103 *    7F 0A 72 00 =
                                                      104 *
                                                      105 *      01111111 00001010 01110010 00000000
                                                      106 *
                                                      107 *    Condition code 2 is set to indicate that the result is
                                                      108 *    greater than zero.
                                                      109 *
                                                      110 *    If a left shift of nine places had been specified, a
                                                      111 *    significant bit would have been shifted out of bit
                                                      112 *    position 1. Condition code 3 would have been set to
                                                      113 *    indicate this overflow and, if the fixed-point-overflow
                                                      114 *    mask bit in the PSW were one, a fixed-point overflow
                                                      115 *    interruption would have occurred.
                                                      116 *
                                                      117 ************************************************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000220                     00000000             119          USING TTAB32,R1

00000220  5810 0340                    00000340   121 SLA      L     R1,=A(TST32TAB)      R1 --> test table

00000224  9825 1000                    00000000   123 SLA1     LM    R2,R5,0(R1)          Load parameters
00000228  4344 033A                    0000033A   124          IC    R4,BCMASKS(R4)       Get BC instruction mask

0000022C  8B20 3000                    00000000   126          SLA   R2,0(R3)            Do the shift
00000230  4440 0252                    00000252   127          EX    R4,SLACC             Expected CC?
00000234  45D0 01F4                    000001F4   128          BAL   R13,FAILTEST         Unexpected CC! FAIL!

00000238  1525                                    130 SLA2     CLR   R2,R5               Expected results?
0000023A  4780 0242                    00000242   131          BE    SLA3                 Yes, continue
0000023E  45D0 01F4                    000001F4   132          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

00000242  4110 1010                    00000010   134 SLA3     LA    R1,TT32NEXT          Next test table entry
00000246  D503 0344 1000      00000344 00000000   135          CLC   =CL4'END!',0(R1)     End of test table?
0000024C  4770 0224                    00000224   136          BNE   SLA1                 No, loop...
00000250  07FE                                     137          BR    R14                  Yes, return to caller

00000252  4700 0238                    00000238   139 SLACC    BC    0,SLA2               Expected condition code?

00000256                                           141          DROP  R1
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    143 ************************************************************************
                                                    144 *  8F   SLDA  - Shift Left Double                            [RS-a]
                                                    145 ************************************************************************
                                                    146 *
                                                    147 *    SHIFT LEFT DOUBLE (SLDA)
                                                    148 *
                                                    149 *    The SHIFT LEFT DOUBLE instruction shifts the 63
                                                    150 *    numeric bits of an even-odd register pair to the left,
                                                    151 *    leaving the sign bit unchanged. Thus, the instruction
                                                    152 *    performs an algebraic left shift of a 64-bit signed
                                                    153 *    binary integer.
                                                    154 *
                                                    155 *    For example, if the contents of registers 2 and 3 are:
                                                    156 *
                                                    157 *    00 7F 0A 72   FE DC BA 98 =
                                                    158 *
                                                    159 *      00000000 01111111 00001010 01110010
                                                    160 *      11111110 11011100 10111010 10011000
                                                    161 *
                                                    162 *
                                                    163 *    The instruction:
                                                    164 *
                                                    165 *    Machine Format
                                                    166 *
                                                    167 *      0           1           2           3           4
                                                    168 *      +-----+-----+-----+-----+-----+-----+-----+-----+
                                                    169 *      |     8F    |  2  | /// |  0  |      01F        |    RS-a
                                                    170 *      +-----+-----+-----+-----+-----+-----+-----+-----+
                                                    171 *
                                                    172 *    Assembler Format
                                                    173 *
                                                    174 *      Op Code  R1,D2(B2)
                                                    175 *      ------------------
                                                    176 *        SLDA    2,31(0)
                                                    177 *
                                                    178 *    results in registers 2 and 3 both being left-shifted 31
                                                    179 *    bit positions, so that their new contents are:
                                                    180 *
                                                    181 *    7F 6E 5D 4C   00 00 00 00 =
                                                    182 *
                                                    183 *      01111111 01101110 01011101 01001100
                                                    184 *      00000000 00000000 00000000 00000000
                                                    185 *
                                                    186 *    Because significant bits are shifted out of bit position
                                                    187 *    1 of register 2, overflow is indicated by setting
                                                    188 *    condition code 3, and, if the fixed-point-overflow mask bit
                                                    189 *    in the PSW is one, a fixed-point-overflow program
                                                    190 *    interruption occurs.
                                                    191 *
                                                    192 ************************************************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

00000256                     00000000             194          USING TTAB64,R1

00000256  5810 0348                      00000348   196 SLDA     L     R1,=A(TST64TAB)      R1 --> test table

0000025A  9827 1000                      00000000   198 SLDA1    LM    R2,R7,0(R1)          Load parameters
0000025E  4355 033A                      0000033A   199          IC    R5,BCMASKS(R5)       Get BC instruction mask

00000262  8F20 4000                      00000000   201          SLDA  R2,0(R4)             Do the shift
00000266  4450 0292                      00000292   202          EX    R5,SLDACC            Expected CC?
0000026A  45D0 01F4                      000001F4   203          BAL   R13,FAILTEST         Unexpected CC! FAIL!

0000026E  1526                                      205 SLDA2    CLR   R2,R6               Expected results?
00000270  4780 0278                      00000278   206          BE    SLDA3                Yes, continue
00000274  45D0 01F4                      000001F4   207          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

00000278  1537                                      209 SLDA3    CLR   R3,R7               Expected results?
0000027A  4780 0282                      00000282   210          BE    SLDA4                Yes, continue
0000027E  45D0 01F4                      000001F4   211          BAL   R13,FAILTEST         No! Unexpected results! FAIL!
                                                    212
00000282  4110 1018                      00000018   213 SLDA4    LA    R1,TT64NEXT          Next test table entry
00000286  D503 0344 1000      00000344  00000000   214          CLC   =CL4'END!',0(R1)     End of test table?
0000028C  4770 025A                      0000025A   215          BNE   SLDA1                No, loop...
00000290  07FE                                      216          BR    R14                  Yes, return to caller

00000292  4700 026E                      0000026E   218 SLDACC   BC    0,SLDA2              Expected condition code?

00000296                                            220          DROP  R1
```

```
  LOC         OBJECT CODE       ADDR1     ADDR2     STMT

                                                    222 ********************************************************************************
                                                    223 *  EBDD SLAK  - Shift Left Single Distinct                    [RSY-a]
                                                    224 ********************************************************************************
                                                    225 *
                                                    226 *    SHIFT LEFT SINGLE DISTINCT (SLAK)
                                                    227 *
                                                    228 *
                                                    229 *      Op Code  R1,R3,D2(B2)
                                                    230 *      ------------------
                                                    231 *        SLAK    2,3,8(0)
                                                    232 *
                                                    233 *
                                                    234 *    This instruction is basically identical to SLA except that
                                                    235 *    the value TO BE shifted is held in R3 and remains unchanged,
                                                    236 *    with the results of the 31-bit shift being placed into R1.
                                                    237 *
                                                    238 ********************************************************************************


00000296                        00000000            240          USING TTAB32,R1

00000296  5810 0340                      00000340   242 SLAK     L     R1,=A(TST32TAB)      R1 --> test table

0000029A  9825 1000                      00000000   244 SLAK1    LM    R2,R5,0(R1)          Load parameters
0000029E  1862                                      245          LR    R6,R2                Load beginning value

000002A0  1F22                                      246          SLR   R2,R2                Clear target register
000002A2  4344 033A                      0000033A   247          IC    R4,BCMASKS(R4)       Get BC instruction mask

000002A6  EB26 3000 00DD                 00000000   249          SLAK  R2,R6,0(R3)          Do the shift
000002AC  4440 02DA                      000002DA   250          EX    R4,SLAKCC            Expected CC?
000002B0  45D0 01F4                      000001F4   251          BAL   R13,FAILTEST         NOT CC2! FAIL!

000002B4  1525                                      253 SLAK2    CLR   R2,R5                Expected results?
000002B6  4780 02BE                      000002BE   254          BE    SLAK3                Yes, continue
000002BA  45D0 01F4                      000001F4   255          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

000002BE  5560 1000                      00000000   257 SLAK3    CL    R6,BEGVAL32          Input register unchanged?
000002C2  4780 02CA                      000002CA   258          BE    SLAK4                Yes, continue
000002C6  45D0 01F4                      000001F4   259          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

000002CA  4110 1010                      00000010   261 SLAK4    LA    R1,TT32NEXT          Next test table entry
000002CE  D503 0344 1000       00000344 00000000   262          CLC   =CL4'END!',0(R1)     End of test table?
000002D4  4770 029A                      0000029A   263          BNE   SLAK1                No, loop...
000002D8  07FE                                      264          BR    R14                  Yes, return to caller

000002DA  4700 02B4                      000002B4   266 SLAKCC   BC    0,SLAK2              Expected condition code?

000002DE                                            268          DROP  R1
```

```
  LOC         OBJECT CODE       ADDR1     ADDR2     STMT

                                                      270 ********************************************************************************
                                                      271 *  EB0B SLAG  - Shift Left Single Long                          [RSY-a]
                                                      272 ********************************************************************************
                                                      273 *
                                                      274 *    SHIFT LEFT SINGLE LONG (SLAG)
                                                      275 *
                                                      276 *
                                                      277 *    Assembler Format
                                                      278 *
                                                      279 *      Op Code  R1,R3,D2(B2)
                                                      280 *      ------------------
                                                      281 *        SLAG    2,3,31(0)
                                                      282 *
                                                      283 *
                                                      284 *    This instruction is identical to SLAK except that the shift is a
                                                      285 *    63-bit shift instead of a 31-bit shift.
                                                      286 *
                                                      287 ********************************************************************************


000002DE                        00000000              289          USING TTAB64,R1

000002DE  5810 0348                    00000348       291 SLAG     L     R1,=A(TST64TAB)      R1 --> test table

000002E2  B90B 0022                                   293 SLAG1    SLGR  R2,R2                Clear target register
000002E6  E330 1000 0004               00000000       294          LG    R3,BEGVAL64          Load beginning value
000002EC  5840 1008                    00000008       295          L     R4,SHIFT64           Get shift amount
000002F0  5850 100C                    0000000C       296          L     R5,CC64              Get expected CC
000002F4  4355 033A                    0000033A       297          IC    R5,BCMASKS(R5)       Get BC instruction mask
000002F8  E360 1010 0004               00000010       298          LG    R6,ENDVAL64          Load expected ending value

000002FE  EB23 4000 000B               00000000       300          SLAG  R2,R3,0(R4)          Do the shift
00000304  4450 0336                    00000336       301          EX    R5,SLAGCC            Expected CC?
00000308  45D0 01F4                    000001F4       302          BAL   R13,FAILTEST         Unexpected CC! FAIL!

0000030C  B921 0026                                   304 SLAG2    CLGR  R2,R6                Expected results?
00000310  4780 0318                    00000318       305          BE    SLAG3                Yes, continue
00000314  45D0 01F4                    000001F4       306          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

00000318  E330 1000 0021               00000000       308 SLAG3    CLG   R3,BEGVAL64          Input register unchanged?
0000031E  4780 0326                    00000326       309          BE    SLAG4                Yes, continue
00000322  45D0 01F4                    000001F4       310          BAL   R13,FAILTEST         No! Unexpected results! FAIL!

00000326  4110 1018                    00000018       312 SLAG4    LA    R1,TT64NEXT          Next test table entry
0000032A  D503 0344 1000       00000344 00000000       313          CLC   =CL4'END!',0(R1)     End of test table?
00000330  4770 02E2                    000002E2       314          BNE   SLAG1                No, loop...
00000334  07FE                                        315          BR    R14                  Yes, return to caller

00000336  4700 030C                    0000030C       317 SLAGCC   BC    0,SLAG2              Expected condition code?

0000033A                                              319          DROP  R1
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT
                                                    321 ********************************************************************************
                                                    322 *                          Working Storage
                                                    323 ********************************************************************************

0000033A  80402010                                  325 BCMASKS  DC    X'80',X'40',X'20',X'10'    CC 0, 1, 2, 3

00000340                                            327          LTORG ,        Literals Pool
00000340  00000350                                  328                =A(TST32TAB)
00000344  C5D5C45A                                  329                =CL4'END!'
00000348  00000418                                  330                =A(TST64TAB)
0000034C  0004                                      331                =H'4'

00000350                                            333 TST32TAB DC    0D'0'
                                                    334 ********************************************
                                                    335 *        old way slowest possible positive
                                                    336 *                                 shift CC
00000350  00000001 0000001E                         337          DC    A(X'00000001'),A(30),A(2)
0000035C  40000000                                  338          DC    A(X'40000000')
                                                    339 *
                                                    340 ********************************************
                                                    341 *        old way slowest possible negative
                                                    342 *                                 shift CC
00000360  FFFFFFFF 0000001F                         343          DC    A(X'FFFFFFFF'),A(31),A(1)
0000036C  80000000                                  344          DC    A(X'80000000')
                                                    345 *
                                                    346 ********************************************
                                                    347 *        positive, 0 bits
                                                    348 *                                 shift CC
00000370  00000123 00000000                         349          DC    A(X'00000123'),A(0),A(2)
0000037C  00000123                                  350          DC    A(X'00000123')
                                                    351 *
                                                    352 ********************************************
                                                    353 *        negative, 0 bits
                                                    354 *                                 shift CC
00000380  80000123 00000000                         355          DC    A(X'80000123'),A(0),A(1)
0000038C  80000123                                  356          DC    A(X'80000123')
                                                    357 *
                                                    358 ********************************************
                                                    359 *        max positive, 1 bit
                                                    360 *                                 shift CC
00000390  7FFFFFFF 00000001                         361          DC    A(X'7FFFFFFF'),A(1),A(3)
0000039C  7FFFFFFE                                  362          DC    A(X'7FFFFFFE')
                                                    363 *
                                                    364 ********************************************
                                                    365 *        max negative, 1 bit
                                                    366 *                                 shift CC
000003A0  80000000 00000001                         367          DC    A(X'80000000'),A(1),A(3)
000003AC  80000000                                  368          DC    A(X'80000000')
                                                    369 *
                                                    370 ********************************************
                                                    371 *        positive, 1 bit
                                                    372 *                                 shift CC
000003B0  22222222 00000001                         373          DC    A(X'22222222'),A(1),A(2)
000003BC  44444444                                  374          DC    A(X'44444444')
                                                    375 *
                                                    376 ********************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    377 *        negative, 1 bit
                                                    378 *                                shift CC
000003C0  CAAAAAAA 00000001                         379          DC    A(X'CAAAAAAA'),A(1),A(1)
000003CC  95555554                                  380          DC    A(X'95555554')
                                                    381 *
                                                    382 ******************************************
                                                    383 *        positive, 1 bit, OVERFLOW
                                                    384 *                                shift CC
000003D0  77777777 00000001                         385          DC    A(X'77777777'),A(1),A(3)
000003DC  6EEEEEEE                                  386          DC    A(X'6EEEEEEE')
                                                    387 *
                                                    388 ******************************************
                                                    389 *        negative, 1 bit, OVERFLOW
                                                    390 *                                shift CC
000003E0  88888888 00000001                         391          DC    A(X'88888888'),A(1),A(3)
000003EC  91111110                                  392          DC    A(X'91111110')
                                                    393 *
                                                    394 ******************************************
                                                    395 *        (original POPS test 1)
                                                    396 *                                shift CC
000003F0  007F0A72 00000008                         397          DC    A(X'007F0A72'),A(8),A(2)
000003FC  7F0A7200                                  398          DC    A(X'7F0A7200')
                                                    399 *
                                                    400 ******************************************
                                                    401 *        (original POPS test 2)
                                                    402 *                                shift CC
00000400  007F0A72 00000009                         403          DC    A(X'007F0A72'),A(9),A(3)
0000040C  7E14E400                                  404          DC    A(X'7E14E400')
                                                    405 *
                                                    406 ******************************************
00000410  C5D5C45A                                  407          DC    CL4'END!'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT
00000418                                            409 TST64TAB DC    0D'0'
                                                    410 ******************************************
                                                    411 *        old way slowest possible positive
                                                    412 *                                        shift CC
00000418  00000000 00000001                         413          DC    A(X'00000000'),A(X'00000001'),A(62),A(2)
00000428  40000000 00000000                         414          DC    A(X'40000000'),A(X'00000000')
                                                    415 *
                                                    416 ******************************************
                                                    417 *        old way slowest possible negative
                                                    418 *                                        shift CC
00000430  FFFFFFFF FFFFFFFF                         419          DC    A(X'FFFFFFFF'),A(X'FFFFFFFF'),A(63),A(1)
00000440  80000000 00000000                         420          DC    A(X'80000000'),A(X'00000000')
                                                    421 *
                                                    422 ******************************************
                                                    423 *        positive, 0 bits
                                                    424 *                                        shift CC
00000448  00000000 00000123                         425          DC    A(X'00000000'),A(X'00000123'),A(0),A(2)
00000458  00000000 00000123                         426          DC    A(X'00000000'),A(X'00000123')
                                                    427 *
                                                    428 ******************************************
                                                    429 *        negative, 0 bits
                                                    430 *                                        shift CC
00000460  80000000 00000123                         431          DC    A(X'80000000'),A(X'00000123'),A(0),A(1)
00000470  80000000 00000123                         432          DC    A(X'80000000'),A(X'00000123')
                                                    433 *
                                                    434 ******************************************
                                                    435 *        max positive, 1 bit
                                                    436 *                                        shift CC
00000478  7FFFFFFF FFFFFFFF                         437          DC    A(X'7FFFFFFF'),A(X'FFFFFFFF'),A(1),A(3)
00000488  7FFFFFFF FFFFFFFE                         438          DC    A(X'7FFFFFFF'),A(X'FFFFFFFE')
                                                    439 *
                                                    440 ******************************************
                                                    441 *        max negative, 1 bit
                                                    442 *                                        shift CC
00000490  80000000 00000000                         443          DC    A(X'80000000'),A(X'00000000'),A(1),A(3)
000004A0  80000000 00000000                         444          DC    A(X'80000000'),A(X'00000000')
                                                    445 *
                                                    446 ******************************************
                                                    447 *        positive, 1 bit
                                                    448 *                                        shift CC
000004A8  22222222 22222222                         449          DC    A(X'22222222'),A(X'22222222'),A(1),A(2)
000004B8  44444444 44444444                         450          DC    A(X'44444444'),A(X'44444444')
                                                    451 *
                                                    452 ******************************************
                                                    453 *        negative, 1 bit
                                                    454 *                                        shift CC
000004C0  CAAAAAAA AAAAAAAA                         455          DC    A(X'CAAAAAAA'),A(X'AAAAAAAA'),A(1),A(1)
000004D0  95555555 55555554                         456          DC    A(X'95555555'),A(X'55555554')
                                                    457 *
                                                    458 ******************************************
                                                    459 *        positive, 1 bit, OVERFLOW
                                                    460 *                                        shift CC
000004D8  77777777 77777777                         461          DC    A(X'77777777'),A(X'77777777'),A(1),A(3)
000004E8  6EEEEEEE EEEEEEEE                         462          DC    A(X'6EEEEEEE'),A(X'EEEEEEEE')
                                                    463 *
                                                    464 ******************************************
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2    STMT

                                                    465 *        negative, 1 bit, OVERFLOW
                                                    466 *                                                shift CC
000004F0  88888888 88888888                         467          DC    A(X'88888888'),A(X'88888888'),A(1),A(3)
00000500  91111111 11111110                         468          DC    A(X'91111111'),A(X'11111110')
                                                    469 *
                                                    470 ********************************************
                                                    471 *        (original POPS test 1)
                                                    472 *                                                shift CC
00000508  007F0A72 FEDCBA98                         473          DC    A(X'007F0A72'),A(X'FEDCBA98'),A(31),A(3)
00000518  7F6E5D4C 00000000                         474          DC    A(X'7F6E5D4C'),A(X'00000000')
                                                    475 *
                                                    476 ********************************************
                                                    477 *        (original POPS test 2)
                                                    478 *                                                shift CC
00000520  007F0A72 FEDCBA98                         479          DC    A(X'007F0A72'),A(X'FEDCBA98'),A(8),A(2)
00000530  7F0A72FE DCBA9800                         480          DC    A(X'7F0A72FE'),A(X'DCBA9800')
                                                    481 *
                                                    482 ********************************************
00000538  C5D5C45A                                  483          DC    CL4'END!'
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

                                                    485 ********************************************************************************
                                                    486 *                          Test tables DSECTs
                                                    487 ********************************************************************************


                                                    489 TTAB32   DSECT
00000000  00000000                                  490 BEGVAL32 DS    A            Starting value
00000004  00000000                                  491 SHIFT32  DS    A            shift amount (#of bits to shift)
00000008  00000000                                  492 CC32     DS    A            Expected condition code
0000000C  00000000                                  493 ENDVAL32 DS    A            Expected ending value
                          00000010  00000001        494 TT32NEXT EQU   *


                                                    496 TTAB64   DSECT
00000000  00000000                                  497 BEGVAL64 DS    A            Starting value (hi 32)
00000004  00000000                                  498          DS    A            Starting value (lo 32)
00000008  00000000                                  499 SHIFT64  DS    A            shift amount (#of bits to shift)
0000000C  00000000                                  500 CC64     DS    A            Expected condition code
00000010  00000000                                  501 ENDVAL64 DS    A            Expected ending value (hi 32)
00000014  00000000                                  502          DS    A            Expected ending value (lo 32)
                          00000018  00000001        503 TT64NEXT EQU   *


                                                    505 ********************************************************************************
                                                    506 *                          Register Equates
                                                    507 ********************************************************************************

                          00000000  00000001        509 R0       EQU   0
                          00000001  00000001        510 R1       EQU   1
                          00000002  00000001        511 R2       EQU   2
                          00000003  00000001        512 R3       EQU   3
                          00000004  00000001        513 R4       EQU   4
                          00000005  00000001        514 R5       EQU   5
                          00000006  00000001        515 R6       EQU   6
                          00000007  00000001        516 R7       EQU   7
                          00000008  00000001        517 R8       EQU   8
                          00000009  00000001        518 R9       EQU   9
                          0000000A  00000001        519 R10      EQU   10
                          0000000B  00000001        520 R11      EQU   11
                          0000000C  00000001        521 R12      EQU   12
                          0000000D  00000001        522 R13      EQU   13
                          0000000E  00000001        523 R14      EQU   14
                          0000000F  00000001        524 R15      EQU   15


                                                    526          END
```

SYMBOL	TYPE	VALUE	LENGTH	DEFN	REFERENCES
BCMASKS	X	00033A	1	325	124 199 247 297
BEGIN	H	0001E0	2	47	34
BEGVAL32	A	000000	4	490	257
BEGVAL64	A	000000	4	497	294 308
CC32	A	000008	4	492	
CC64	A	00000C	4	500	296
ENDVAL32	A	00000C	4	493	
ENDVAL64	A	000010	4	501	298
FAILPSW	D	000210	8	64	57
FAILTEST	I	0001F4	4	56	128 132 203 207 211 251 255 259 302 306 310
GOODPSW	D	000200	8	59	54
IMAGE	1	000000	1340	0	
R0	U	000000	1	509	27
R1	U	000001	1	510	119 121 123 134 135 141 194 196 198 213 214 220 240 242 244 261 262 268 289 291 312 313 319
R10	U	00000A	1	519	
R11	U	00000B	1	520	
R12	U	00000C	1	521	
R13	U	00000D	1	522	56 128 132 203 207 211 251 255 259 302 306 310
R14	U	00000E	1	523	49 50 51 52 137 216 264 315
R15	U	00000F	1	524	
R2	U	000002	1	511	123 126 130 198 201 205 244 245 246 249 253 293 300 304
R3	U	000003	1	512	126 209 249 294 300 308
R4	U	000004	1	513	124 127 201 247 250 295 300
R5	U	000005	1	514	123 130 199 202 244 253 296 297 301
R6	U	000006	1	515	205 245 249 257 298 304
R7	U	000007	1	516	198 209
R8	U	000008	1	517	
R9	U	000009	1	518	
SHIFT32	A	000004	4	491	
SHIFT64	A	000008	4	499	295
SHIFTEST	J	000000	1340	25	29 36
SLA	I	000220	4	121	49
SLA1	I	000224	4	123	136
SLA2	I	000238	2	130	139
SLA3	I	000242	4	134	131
SLACC	I	000252	4	139	127
SLAG	I	0002DE	4	291	52
SLAG1	I	0002E2	4	293	314
SLAG2	I	00030C	4	304	317
SLAG3	I	000318	6	308	305
SLAG4	I	000326	4	312	309
SLAGCC	I	000336	4	317	301
SLAK	I	000296	4	242	51
SLAK1	I	00029A	4	244	263
SLAK2	I	0002B4	2	253	266
SLAK3	I	0002BE	4	257	254
SLAK4	I	0002CA	4	261	258
SLAKCC	I	0002DA	4	266	250
SLDA	I	000256	4	196	50
SLDA1	I	00025A	4	198	215
SLDA2	I	00026E	2	205	218
SLDA3	I	000278	2	209	206
SLDA4	I	000282	4	213	210
SLDACC	I	000292	4	218	202
TST32TAB	D	000350	8	333	121

SYMBOL	TYPE	VALUE	LENGTH	DEFN	REFERENCES
TST64TAB	D	000418	8	409	196
TT32NEXT	U	000010	1	494	134 261
TT64NEXT	U	000018	1	503	213 312
TTAB32	4	000000	16	489	119 240
TTAB64	4	000000	24	496	194 289
=A(TST32TAB)	A	000340	4	328	121 242
=A(TST64TAB)	A	000348	4	330	196 291
=CL4'END!'	C	000344	4	329	135 214 262 313
=H'4'	H	00034C	2	331	56

MACRO DEFN REFERENCES

No defined macros

```
   DESC     SYMBOL    SIZE    POS      ADDR

Entry: 0

Image      IMAGE     1340  000-53B  000-53B
  Region             1340  000-53B  000-53B
    CSECT  SHIFTEST  1340  000-53B  000-53B
```

STMT FILE NAME

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

** NO ERRORS FOUND **