```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT
                                                      2 *
                                                      3 *        Jurgen Winkelmann's MSA-5 'PRNO' instruction test
                                                      4 *
                                                      5 *          This module tests the PRNO instruction
                                                      6 *          in a standalone environment.
                                                      7 *
                                                      8 *        Operation -
                                                      9 *
                                                     10 *          PRNOTEST exercises PRNO QUERY, DRNG, and TRNG functions
                                                     11 *          and does plausibility checks on the results.
                                                     12 *
                                                     13 *          - If all tests pass, PRNOTEST enters a disabled wait state
                                                     14 *            with a PSW address of X'0000000000000000' (all zeros).
                                                     15 *
                                                     16 *          - If a test fails, the test sequence is aborted
                                                     17 *            and a disabled wait state X'000000000000DEAD' is entered.
                                                     18 *
                                                     19 * Fish 2025-09-02:
                                                     20 *                  1.  Do each test for each addressing mode too
                                                     21 *                  2.  Added GitHub #765 test ==> Addresssing Exception
                                                     22 *
                                                     23          PRINT OFF  (register equates)
                                                     40          PRINT ON
                          00000000  000007FF       41 PRNOTEST CSECT
00000000                  00000000                  42          USING *,0
00000000                  00000000  000001A0       43          ORG   PRNOTEST+X'1A0'                    # z/Arch restart PSW
000001A0  00000000 00000000                       44 RSTRTNEW DC    X'00000000000000000000000000000200' # z/Arch restart PSW  (24-bit mode)
                                                     45 **       DC    X'00000000800000000000000000000200' # z/Arch restart PSW  (31-bit mode)
                                                     46 **       DC    X'00000001800000000000000000000200' # z/Arch restart PSW  (64-bit mode)
000001B0                  000001B0  000001D0       47          ORG   PRNOTEST+X'1D0'                    # z/Arch pgm new PSW
000001D0  00020000 00000000                       48 PROGNEW  DC    X'00020000000000000000000000000DEAD' # z/Arch pgm new PSW  (24-bit mode)
                                                     49 **       DC    X'00020000800000000000000000000DEAD' # z/Arch pgm new PSW  (31-bit mode)
                                                     50 **       DC    X'00020001800000000000000000000DEAD' # z/Arch pgm new PSW  (64-bit mode)
000001E0                  000001E0  00000200       51          ORG   PRNOTEST+X'200'
                                                     52 ***
                                                     53 ***      QUERY
                                                     54 ***
00000200  C001 0000 0000                          55          LGFI  R0,0             R0->function code 0
00000206  D2EF 0480 0600        00000480  00000600  56          MVC   PB(240),PBNULL   clear parameter block
0000020C  4110 0480                       00000480  57          LA    R1,PB            R1->parameter block
00000210  B93C 0024                                 58          PRNO  R2,R4            perform random number operation
00000214  D50F 06F0 0480        000006F0  00000480  59          CLC   ERQUERY(16),PB   compare with expected result
0000021A  4780 0220                       00000220  60          BE    *+6              result OK
0000021E  0000                                      61          DC    H'0'             disabled wait DEAD if result invalid
                                                     62 ***
                                                     63 ***      DRNG: FIPS known answer test
                                                     64 ***
00000220  C001 0000 0083                          65          LGFI  R0,131           R0->function code 3 with modifier: seed
00000226  D2EF 0480 0600        00000480  00000600  66          MVC   PB(240),PBNULL   clear parameter block
0000022C  4110 0480                       00000480  67          LA    R1,PB            R1->parameter block
00000230  4120 0800                       00000800  68          LA    R2,FO            R2->first  operand address
00000234  C031 0000 0000                            69          LGFI  R3,0             R3->first  operand length
0000023A  4140 0570                       00000570  70          LA    R4,SO            R2->second operand address
0000023E  C051 0000 0040                            71          LGFI  R5,64            R3->second operand length
00000244  D23F 0570 0708        00000570  00000708  72          MVC   SO(64),ENTROPY   provide predefined entropy
0000024A  B93C 0024                                 73          PRNO  R2,R4            perform random number seed operation
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

0000024E  C001 0000 0003                             74          LGFI  R0,3            R0->function code 3: generate
00000254  4110 0480                  00000480        75          LA    R1,PB           R1->parameter block
00000258  4120 0800                  00000800        76          LA    R2,FO           R2->first  operand address
0000025C  C031 0000 0040                             77          LGFI  R3,64           R3->first  operand length
00000262  4140 0570                  00000570        78          LA    R4,SO           R2->second operand address
00000266  C051 0000 0000                             79          LGFI  R5,0            R3->second operand length
0000026C  B93C 0024                                  80          PRNO  R2,R4           perform random number generate operation
00000270  D53F 0748 0800      00000748  00000800     81          CLC   ERFIPS(64),FO   compare with expected result
00000276  4780 027C                  0000027C        82          BE    *+6             result OK
0000027A  0000                                       83          DC    H'0'            disabled wait DEAD if result invalid
                                                     84 ***
                                                     85 ***       DRNG: Reseed and generate
                                                     86 ***
0000027C  C001 0000 0083                             87          LGFI  R0,131          R0->function code 3 with modifier: reseed
00000282  4110 0480                  00000480        88          LA    R1,PB           R1->parameter block
00000286  4120 0800                  00000800        89          LA    R2,FO           R2->first  operand address
0000028A  C031 0000 0000                             90          LGFI  R3,0            R3->first  operand length
00000290  4140 0570                  00000570        91          LA    R4,SO           R2->second operand address
00000294  C051 0000 0040                             92          LGFI  R5,64           R3->second operand length
0000029A  D23F 0570 0491      00000570  00000491     93          MVC   SO(64),PB+17    steal seed material (ignored by Hercules)
000002A0  B93C 0024                                  94          PRNO  R2,R4           perform random number reseed operation
000002A4  C001 0000 0003                             95          LGFI  R0,3            R0->function code 3: generate
000002AA  4110 0480                  00000480        96          LA    R1,PB           R1->parameter block
000002AE  4120 0800                  00000800        97          LA    R2,FO           R2->first  operand address
000002B2  C031 0001 0000                             98          LGFI  R3,65536        R3->first  operand length
000002B8  4140 0570                  00000570        99          LA    R4,SO           R2->second operand address
000002BC  C051 0000 0000                            100          LGFI  R5,0            R3->second operand length
000002C2  B93C 0024                                 101          PRNO  R2,R4           perform random number generate operation
000002C6  D53F 0800 0600      00000800  00000600    102          CLC   FO(64),PBNULL   first 64 bytes zero ..
000002CC  4770 02D2                  000002D2        103          BNE   *+6             .. is not plausible
000002D0  0000                                      104          DC    H'0'            disabled wait DEAD if first 64 bytes zero
000002D2  C031 0001 07C1                            105          LGFI  R3,FODISP+65536-63 last 64 bytes ..
000002D8  D53F 3000 0600      00000000  00000600    106          CLC   0(64,R3),PBNULL .. zero ..
000002DE  4770 02E4                  000002E4        107          BNE   *+6             .. is not plausible
000002E2  0000                                      108          DC    H'0'            disabled wait DEAD if last 64 bytes zero
                                                    109 ***
                                                    110 ***       TRNG Query
                                                    111 ***
000002E4  C001 0000 0070                            112          LGFI  R0,112          R0->function code 112
000002EA  D2EF 0480 0600      00000480  00000600    113          MVC   PB(240),PBNULL  clear parameter block
000002F0  4110 0480                  00000480        114          LA    R1,PB           R1->parameter block
000002F4  B93C 0024                                 115          PRNO  R2,R4           perform random number operation
000002F8  D507 0700 0480      00000700  00000480    116          CLC   TRQUERY(8),PB   compare with expected result
000002FE  4780 0304                  00000304        117          BE    *+6             result OK
00000302  0000                                      118          DC    H'0'            disabled wait DEAD if result invalid
                                                    119 ***
                                                    120 ***       TRNG
                                                    121 ***
00000304  C001 0000 0072                            122          LGFI  R0,114          R0->function code 114: TRNG
0000030A  4120 0800                  00000800       123          LA    R2,FO           R2->first  operand address
0000030E  C031 0000 0040                            124          LGFI  R3,64           R3->first  operand length
00000314  4140 0570                  00000570       125          LA    R4,SO           R4->second operand address
00000318  C051 0000 0040                            126          LGFI  R5,64           R5->second operand length
0000031E  9500 0411                  00000411       127          CLI   GH765,0         normal test?
00000322  4780 0332                  00000332       128          BE    PRNO            yes, continue
00000326  E350 03E0 0004             000003E0       129          LG    R5,=X'0505050500000040'      #765 salva-rczero
```

```
  LOC         OBJECT CODE       ADDR1     ADDR2    STMT

0000032C  D203 01DC 03E8      000001DC  000003E8   130          MVC   PROGNEW+12(4),=AL4(0)        Program Intgerrupt handler
                                                   131 *                                           (Addr Exception expected!)
00000332  B93C 0024                                132 PRNO     PRNO  R2,R4           perform random number generate operation
00000336  9500 0411                      00000411  133          CLI   GH765,0         Normal run?
0000033A  4780 0360                      00000360  134          BE    CONT            Yes, continue
0000033E  9540 0410                      00000410  135          CLI   AM,64           64-bit addressing mode?
00000342  4770 0360                      00000360  136          BNE   CONT            No, continue
00000346  B2B2 0350                      00000350  137 FAIL765  LPSWE PSW765          WTF?! We should have Program Checked!!
00000350  00020001 80000000                        138 PSW765   DC    0D'0',X'0002000180000000000000000000765' GH765 FAILURE!
00000360  D53F 0800 0600      00000800  00000600  139 CONT     CLC   FO(64),PBNULL  first operand zero ..
00000366  4770 036C                      0000036C  140          BNE   *+6             .. is not plausible
0000036A  0000                                     141          DC    H'0'            disabled wait DEAD if first operand zero
0000036C  D53F 0570 0600      00000570  00000600  142          CLC   SO(64),PBNULL  seconf operand zero ..
00000372  4770 0378                      00000378  143          BNE   *+6             .. is not plausible
00000376  0000                                     144          DC    H'0'            disabled wait DEAD if second operand zero
00000378  9518 0410                      00000410  145          CLI   AM,24           AM24 done?
0000037C  4780 0392                      00000392  146          BE    AM31            Yes, Then test AM31
00000380  951F 0410                      00000410  147          CLI   AM,31           AM31 done?
00000384  4780 03A6                      000003A6  148          BE    AM64            Yes, Then test AM64
00000388  9540 0410                      00000410  149          CLI   AM,64           AM64 done?
0000038C  4780 03BA                      000003BA  150          BE    TRY765          Yes, Do GitHub #765 test if we haven't yet
00000390  0000                                     151          DC    H'0'            WTF?!
00000392  D201 01A3 03EC      000001A3  000003EC  152 AM31     MVC   RSTRTNEW+3(2),=XL2'0080'   Switch to AM31
00000398  D201 01D3 03EC      000001D3  000003EC  153          MVC   PROGNEW+3(2),=XL2'0080'    Switch to AM31
0000039E  921F 0410                      00000410  154          MVI   AM,31                     Switch to AM31
000003A2  B2B2 01A0                      000001A0  155          LPSWE RSTRTNEW                  Do all in AM31
000003A6  D201 01A3 03EE      000001A3  000003EE  156 AM64     MVC   RSTRTNEW+3(2),=XL2'0180'   Switch to AM64
000003AC  D201 01D3 03EE      000001D3  000003EE  157          MVC   PROGNEW+3(2),=XL2'0180'    Switch to AM64
000003B2  9240 0410                      00000410  158          MVI   AM,64                     Switch to AM64
000003B6  B2B2 01A0                      000001A0  159          LPSWE RSTRTNEW                  Do all in AM64
000003BA  9500 0411                      00000411  160 TRY765   CLI   GH765,0                    First time here?
000003BE  4770 03DA                      000003DA  161          BNE   BAD765                    No?! WTF?!
000003C2  92FF 0411                      00000411  162          MVI   GH765,X'FF'               Remember what we're doing
000003C6  9218 0410                      00000410  163          MVI   AM,24                     Switch back to AM24 again
000003CA  D201 01A3 03F0      000001A3  000003F0  164          MVC   RSTRTNEW+3(2),=XL2'0000'   Switch back to AM24 again
000003D0  D201 01D3 03F0      000001D3  000003F0  165          MVC   PROGNEW+3(2),=XL2'0000'    Switch back to AM24 again
000003D6  B2B2 01A0                      000001A0  166          LPSWE RSTRTNEW                  Do all AM tests all over again
000003DA  4DE0 0346                      00000346  167 BAD765   BAS   R14,FAIL765               GH #765 FAILURE
000003E0                                           168          LTORG ,
000003E0  05050505 00000040                        169                =X'0505050500000040'
000003E8  00000000                                 170                =AL4(0)
000003EC  0080                                     171                =XL2'0080'
000003EE  0180                                     172                =XL2'0180'
000003F0  0000                                     173                =XL2'0000'
000003F2                      000003F2  00000400   174          ORG   PRNOTEST+X'400'
00000400  00020001 80000000                        175 OKAYPSW  DC    X'0002000180000000000000000000000000' SUCCESSFUL wait PSW
00000410  18                                       176 AM       DC    AL1(24)                 current addressing mode
00000411  00                                       177 GH765    DC    X'00'                   FF = try GH #765 test, 00 = don't
00000412                      00000412  00000480   178          ORG   PRNOTEST+X'480'
00000480  00000000 00000000                        179 PB       DS    XL240           current parameter block
00000570  00000000 00000000                        180 SO       DS    XL64            second operand
000005B0                      000005B0  00000600   181          ORG   PRNOTEST+X'600'
00000600  00000000 00000000                        182 PBNULL   DC    240X'00'        empty parameter block
000006F0  90000000 00000000                        183 ERQUERY  DC    X'900000000000000000000000000A000' expected query reslt
00000700  000000C0 00000020                        184 TRQUERY  DC    X'000000C000000020'  expected Hercules TRNG query result
00000708  3295117F 02371270                        185 ENTROPY  DC    X'3295117F02371270'  predefined entropy for
```

```
  LOC        OBJECT CODE       ADDR1     ADDR2     STMT

00000710  105A3783 CFE0BF5A                          186          DC    X'105A3783CFE0BF5A'  FIPS known answer test
00000718  C1408E6C EAC5AEEB                          187          DC    X'C1408E6CEAC5AEEB'
00000720  D6D814BC 827DC04D                          188          DC    X'D6D814BC827DC04D'
00000728  21A1E480 E3D5F2E3                          189          DC    X'21A1E480E3D5F2E3'
00000730  78319ADE 9BDDDA4C                          190          DC    X'78319ADE9BDDDA4C'
00000738  2E93B74E 348D5EE3                          191          DC    X'2E93B74E348D5EE3'
00000740  2ED46A1A E62566E0                          192          DC    X'2ED46A1AE62566E0'
00000748  F1DFE833 0811ECD1                          193 ERFIPS   DC    X'F1DFE8330811ECD1'  expected generate result for
00000750  0AEB6872 8FAC57B0                          194          DC    X'0AEB68728FAC57B0'  FIPS known answer test
00000758  5DC8B411 6DFCC066                          195          DC    X'5DC8B4116DFCC066'
00000760  C4FBB654 B317FB0E                          196          DC    X'C4FBB654B317FB0E'
00000768  01126574 8F7929B0                          197          DC    X'011265748F7929B0'
00000770  18036662 5DE0665B                          198          DC    X'180366625DE0665B'
00000778  116C878B 0F05BAD8                          199          DC    X'116C878B0F05BAD8'
00000780  31941625 8824DFDC                          200          DC    X'319416258824DFDC'
                             00000800  00000001    201 FODISP   EQU   X'800'          where 'FO' ("first operand") should be
00000788                     00000788  00000800    202          ORG   PRNOTEST+FODISP
00000800                                           203 FO       DS    0X             first operand
                                                   204          END
```

SYMBOL	TYPE	VALUE	LENGTH	DEFN	REFERENCES
AM	R	000410	1	176	135 145 147 149 154 158 163
AM31	I	000392	6	152	146
AM64	I	0003A6	6	156	148
BAD765	I	0003DA	4	167	161
CONT	I	000360	6	139	134 136
ENTROPY	X	000708	8	185	72
ERFIPS	X	000748	8	193	81
ERQUERY	X	0006F0	16	183	59
FAIL765	I	000346	4	137	167
FO	X	000800	1	203	68 76 81 89 97 102 123 139
FODISP	U	000800	1	201	202 105
GH765	X	000411	1	177	127 133 160 162
IMAGE	1	000000	2048	0	
OKAYPSW	X	000400	16	175	
PB	X	000480	240	179	56 57 59 66 67 75 88 93 96 113 114 116
PBNULL	X	000600	1	182	56 66 102 106 113 139 142
PRNO	I	000332	4	132	128
PRNOTEST	J	000000	2048	41	43 47 51 174 178 181 202
PROGNEW	X	0001D0	16	48	130 153 157 165
PSW765	D	000350	8	138	137
R0	U	000000	1	24	55 65 74 87 95 112 122
R1	U	000001	1	25	57 67 75 88 96 114
R10	U	00000A	1	34	
R11	U	00000B	1	35	
R12	U	00000C	1	36	
R13	U	00000D	1	37	
R14	U	00000E	1	38	167
R15	U	00000F	1	39	
R2	U	000002	1	26	58 68 73 76 80 89 94 97 101 115 123 132
R3	U	000003	1	27	69 77 90 98 105 106 124
R4	U	000004	1	28	58 70 73 78 80 91 94 99 101 115 125 132
R5	U	000005	1	29	71 79 92 100 126 129
R6	U	000006	1	30	
R7	U	000007	1	31	
R8	U	000008	1	32	
R9	U	000009	1	33	
RSTRTNEW	X	0001A0	16	44	152 155 156 159 164 166
SO	X	000570	64	180	70 72 78 91 93 99 125 142
TRQUERY	X	000700	8	184	116
TRY765	I	0003BA	4	160	150
=AL4(0)	R	0003E8	4	170	130
=X'0505050500000040'	X	0003E0	8	169	129
=XL2'0000'	X	0003F0	2	173	164 165
=XL2'0080'	X	0003EC	2	171	152 153
=XL2'0180'	X	0003EE	2	172	156 157

MACRO DEFN REFERENCES

No defined macros

```
   DESC     SYMBOL    SIZE    POS      ADDR

Entry: not defined

Image      IMAGE     2048  000-7FF  000-7FF
  Region             2048  000-7FF  000-7FF
    CSECT  PRNOTEST  2048  000-7FF  000-7FF
```

```
   STMT                                        FILE NAME
1     C:\Users\Fish\Documents\Visual Studio 2008\Projects\MyProjects\ASMA-0\PRNO\PRNO.asm

** NO ERRORS FOUND **
```