B 9

```
MMM         MMM     IIIIIIIII     RRRRRRRRRRRR     RRRRRRRRRRRR        000000000     RRRRRRRRRRRR
MMM         MMM     IIIIIIIII     RRRRRRRRRRRR     RRRRRRRRRRRR        000000000     RRRRRRRRRRRR
MMM         MMM     IIIIIIIII     RRRRRRRRRRRR     RRRRRRRRRRRR        000000000     RRRRRRRRRRRR
MMMMMM   MMMMMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMMMMM   MMMMMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMMMMM   MMMMMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMM   MMM   MMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMM   MMM   MMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMM   MMM   MMM        III        RRR         RRR  RRR         RRR  000         000  RRR         RRR
MMM         MMM        III        RRRRRRRRRRRR     RRRRRRRRRRRR     000         000  RRRRRRRRRRRR
MMM         MMM        III        RRRRRRRRRRRR     RRRRRRRRRRRR     000         000  RRRRRRRRRRRR
MMM         MMM        III        RRRRRRRRRRRR     RRRRRRRRRRRR     000         000  RRRRRRRRRRRR
MMM         MMM        III        RRR   RRR        RRR   RRR        000         000  RRR   RRR
MMM         MMM        III        RRR   RRR        RRR   RRR        000         000  RRR   RRR
MMM         MMM        III        RRR   RRR        RRR   RRR        000         000  RRR   RRR
MMM         MMM        III        RRR      RRR     RRR      RRR     000         000  RRR      RRR
MMM         MMM        III        RRR      RRR     RRR      RRR     000         000  RRR      RRR
MMM         MMM        III        RRR      RRR     RRR      RRR     000         000  RRR      RRR
MMM         MMM     IIIIIIIII     RRR         RRR  RRR         RRR     000000000     RRR         RRR
MMM         MMM     IIIIIIIII     RRR         RRR  RRR         RRR     000000000     RRR         RRR
MMM         MMM     IIIIIIIII     RRR         RRR  RRR         RRR     000000000     RRR         RRR
```

MI
VA

Pa
Sy
Ps
Cr
As

Th
80
Th
29
8

Ma
--
_$

86

Th

MA

```
**FILE**ID**MIRROR                                                    F 9
```

```
MM      MM    IIIIII    RRRRRRRR    RRRRRRRR      000000    RRRRRRRR
MM      MM    IIIIII    RRRRRRRR    RRRRRRRR      000000    RRRRRRRR
MMMM  MMMM      II      RR      RR  RR      RR  00      00  RR      RR
MMMM  MMMM      II      RR      RR  RR      RR  00      00  RR      RR
MM  MM  MM      II      RR      RR  RR      RR  00      00  RR      RR
MM  MM  MM      II      RR      RR  RR      RR  00      00  RR      RR
MM      MM      II      RRRRRRRR    RRRRRRRR    00      00  RRRRRRRR
MM      MM      II      RRRRRRRR    RRRRRRRR    00      00  RRRRRRRR
MM      MM      II      RR  RR      RR  RR      00      00  RR  RR
MM      MM      II      RR  RR      RR  RR      00      00  RR  RR
MM      MM      II      RR    RR    RR    RR    00      00  RR    RR        ....
MM      MM      II      RR    RR    RR    RR    00      00  RR    RR        ....
MM      MM    IIIIII    RR      RR  RR      RR    000000    RR      RR      ....
MM      MM    IIIIII    RR      RR  RR      RR    000000    RR      RR      ....


LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
Table of contents
```

```
0000            .TITLE  MIRROR -        DECNET-VAX LOOPBACK MIRROR
0000            .IDENT  'V04-000'
0000
0000    ;
0000    ;****************************************************************************
0000    ;*                                                                          *
0000    ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000    ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000    ;*  ALL RIGHTS RESERVED.                                                    *
0000    ;*                                                                          *
0000    ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000    ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000    ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000    ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000    ;*  TRANSFERRED.                                                            *
0000    ;*                                                                          *
0000    ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000    ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    ;*  CORPORATION.                                                            *
0000    ;*                                                                          *
0000    ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000    ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    ;*                                                                          *
0000    ;*                                                                          *
0000    ;****************************************************************************
0000    ;
0000    ;++
0000    ; FACILITY:     NETWORK MANAGEMENT LISTENER
0000    ;
0000    ; ABSTRACT:     This single module is the LOOPBACK MIRROR. It is the
0000    ;               only network management function specified for the
0000    ;               application layer of the Digital Network Architecture.
0000    ;               Its purpose is to test logical links either between nodes
0000    ;               or within a single node.
0000    ;
0000    ;
0000    ; ENVIRONMENT:
0000    ;       MODE = USER
0000    ;
0000    ; AUTHOR:       KARL MALIK, CREATION DATE: 15-APR-1980
0000    ;
0000    ; MODIFIED BY:
0000    ;
0000    ;       V03-001 MSH0001         Maryann S. Hinden               19-NOV-1981
0000    ;               Add global references to library calls.
0000    ;
0000    ;--
0000
```

```
                                      0000     51          .SBTTL  DECLARATIONS
                                      0000     52 ;
                                      0000     53 ; INCLUDE FILES:
                                      0000     54 ;
                                      0000     55 ;         NONE
                                      0000     56 ;
                                      0000     57 ; MACROS:
                                      0000     58 ;
                                      0000     59 ;         NONE
                                      0000     60
                                      0000     61 ;
                                      0000     62 ; EQUATED SYMBOLS:
                                      0000     63 ;
                                      0000     64 ;
                                      0000     65 ; OWN STORAGE:
                                      0000     66 ;
                                  00000000     67          .PSECT  MIRROR$DATA     SHR,NOEXE,RD,WRT,BYTE
                                      0000     68
45 4E 24 53 59 53 00000008'010E0000'  0000     69 LOGNAME:.ASCID  /SYS$NET/              ; Logical name & descriptor
                                  54  000E
52 4F 52 52 49 4D 00000017'010E0000'  000F     70 LOGNAM2:.ASCID  /MIRROR$SIZE/          ; Logical name & descriptor
                      45 5A 49 53 24  001D
                                      0022     71 LOG2DESC:
                            00000064  0022     72          .LONG   100
                            0000002A' 0026     73          .LONG   LOG2BUF
                            0000008E  002A     74 LOG2BUF:.BLKB   100
   3A 54 45 4E 5F 00000096'010E0000'  008E     75 DEVDESC:.ASCID  /_NET:/                ; Pseudo-device & descriptor
                                      009B     76 DEV_CHAN:
                            0000009D  009B     77          .BLKW   1                     ; Word to receive device channel #
                            000000A5  009D     78 IOSB:    .BLKQ   1                     ; I/O status block
                                      00A5     79
                            0000003F  00A5     80 NCBDESC:.LONG   63                     ; NCB descriptor
                            000000AD' 00A9     81          .LONG   NCB
                            000000EC  00AD     82 NCB:     .BLKB   63                     ; Network connect block
                                      00EC     83
                            000000F0  00EC     84 BASEADR:.BLKL   1                      ; Adr of buffer if MIRROR$SIZE used
                            000010F1  00F0     85 BUFFER:.BLKB    4096+1                 ; Buffer
                            00001000  10F1     86 MAX_DATA=<.-BUFFER>-1                  ; Max size of data receivable (not
                                      10F1     87                                        ; Counting function code)
                                      10F1     88 DATA_SIZE:                             ; Current size of buffer
                            00001001  10F1     89          .LONG   MAX_DATA+1            ;
                                      10F5     90 RECV_SIZE:
                            000010F9  10F5     91          .BLKL   1                     ; # of bytes received
                                      10F9     92
```

```
            10F9  ;++
            10F9  ; FUNCTIONAL DESCRIPTION:
            10F9  ;
            10F9  ;
            10F9  ;       When the LOOPBACK MIRROR accepts a connect, it returns its
            10F9  ;       maximum data size minus 1 in the accept data. This is the amount
            10F9  ;       of data it can handle, not counting the function code.
            10F9  ;
            10F9  ;       (It is possible to specify the maximum data size by defining
            10F9  ;       the logical name 'MIRROR$SIZE' in SYS$SYSTEM:MIRROR.COM. If
            10F9  ;       'MIRROR$SIZE' is not defined ( or is equal to zero ) then the
            10F9  ;       default value of 4096 is used.
            10F9  ;
            10F9  ;       When a Logical Loopback message is received, it is changed
            10F9  ;       into the appropriate response message and returned to the user.
            10F9  ;       The Loopback Mirror continues to repeat all traffic offered.
            10F9  ;       The initiator of the link disconnects it.
            10F9  ;
            10F9  ; CALLING SEQUENCE:
            10F9  ;
            10F9  ;       CONNECTED TO BY NETWORK MANAGEMENT LISTENER
            10F9  ;
            10F9  ; INPUT PARAMETERS:
            10F9  ;
            10F9  ;       NONE
            10F9  ;
            10F9  ; IMPLICIT INPUTS:
            10F9  ;
            10F9  ;       LOGICAL NAME 'SYS$NET'
            10F9  ;       LOGICAL NAME 'MIRROR$SIZE' (OPTIONAL)
            10F9  ;
            10F9  ; OUTPUT PARAMETERS:
            10F9  ;
            10F9  ;       NONE
            10F9  ;
            10F9  ; IMPLICIT OUTPUTS:
            10F9  ;
            10F9  ;       NONE
            10F9  ;
            10F9  ; COMPLETION CODES:
            10F9  ;
            10F9  ;       SUCCESS -> SENDS 1 AS THE FIRST BYTE OF THE LOGICAL LOOPBACK
            10F9  ;                  RESPONSE MESSAGE.
            10F9  ;
            10F9  ;       FAILURE -> SENDS -1 (REMAINDER OF LOGICAL LOOPBACK MESSAGE IS
            10F9  ;                  DISCARDED).
            10F9  ;
            10F9  ;
            10F9  ; SIDE EFFECTS:
            10F9  ;
            10F9  ;       NONE
            10F9  ;
            10F9  ;--
            10F9          .SBTTL  MAIN
        00000000          .PSECT  MIRROR$CODE     NOSHR,EXE,RD,NOWRT,BYTE
            0000
      0000  0000          .ENTRY  MIRROR,^M<>             ; Entry point from exec
```

```
                            0002          .ENABL  LSB                     ; Enable local symbol block
                            0002  ;
                            0002  ; ASSIGN A CHANNEL TO _NET: .
                            0002  ;
                            0002          $ASSIGN_S -
                            0002                  DEVNAM=W^DEVDESC,-      ; Adr of pseudo device descriptor
                            0002                  CHAN=W^DEV_CHAN         ; Adr of word to store channel #
                            0013
                03 50   E8  0013          BLBS    R0,10$                  ; Branch on success
                 0186   31  0016          BRW     EXIT                    ; Branch (failure)
                            0019  ;
                            0019  ; TRANSLATE THE LOGICAL NAME SYS$NET TO OBTAIN THE CONNECTION INFORMATION
                            0019  ; CONTAINED IN THE NCB.
                            0019  ;
                            0019  10$:    $TRNLOG_S -                     ; Translate SYS$NET
                            0019                  LOGNAM=W^LOGNAME,-      ; Descriptor for SYS$NET
                            0019                  RSLLEN=W^NCBDESC,-      ; Substitute new length
                            0019                  RSLBUF=W^NCBDESC        ; Address of modified NCB descriptor
                03 50   E8  0032          BLBS    R0,30$                  ; Branch on success
                 0167   31  0035  20$:    BRW     EXIT                    ; Branch (failure)
         50   0000'8F   B1  0038  30$:    CMPW    #SS$_NOTRAN,R0          ; Did we translate?
                   F6   13  003D          BEQL    20$                     ; No translation for SYS$NET, just quit
                            003F
                            003F  ;
                            003F  ; TRANSLATE THE LOGICAL NAME MIRROR$SIZE AND CONVERT IT TO A HEX VALUE. IF
                            003F  ; NO TRANSLATION (OR VALUE IS TOO BIG OR SMALL) THEN USE 4096 DEFAULT.
                            003F  ;
                            003F          $TRNLOG_S -                     ; Translate MIRROR$SIZE
                            003F                  LOGNAM=W^LOGNAM2,-      ; Descriptor for MIRROR$SIZE
                            003F                  RSLLEN=W^LOG2DESC,-
                            003F                  RSLBUF=W^LOG2DESC
                03 50   E8  0058          BLBS    R0,40$                  ; Branch on success
                 0141   31  005B          BRW     EXIT                    ; Branch (failure)
         55   00F0'CF   DE  005E  40$:    MOVAL   W^BUFFER,R5             ; Set up to use default buffer
         50   0000'8F   B1  0063          CMPW    #SS$_NOTRAN,R0          ; Did we translate?
                   38   13  0068          BEQL    W^DEFAULT               ; No translation, branch and use default
                            006A
              10F1'CF   DF  006A          PUSHAL  W^DATA_SIZE             ; Adr to receive hex output
              0022'CF   7F  006E          PUSHAQ  W^LOG2DESC              ; Adr of ascii descriptor
     00000000'GF   02   FB  0072          CALLS   #2,G^FOR$CNV_IN_I       ; Convert ascii to hex
                03 50   E8  0079          BLBS    R0,50$                  ; Branch on success
                 0120   31  007C          BRW     EXIT                    ; Branch (failure)
                            007F
                            007F  ;
                            007F  ; MAKE SURE THAT THE VALUE IS WITHIN ACCEPTABLE LIMITS.
                            007F  ;
              10F1'CF   D5  007F  50$:    TSTL    W^DATA_SIZE             ; Is it 0?
                   1D   13  0083          BEQL    W^DEFAULT               ; Branch if entered value is 0
              10F1'CF   D6  0085          INCL    W^DATA_SIZE             ; Leave room for function code
              00EC'CF   DF  0089          PUSHAL  W^BASEADR               ; Adr to put starting adr of buffer
              10F1'CF   DF  008D          PUSHAL  W^DATA_SIZE             ; # of bytes to allocate
     00000000'GF   02   FB  0091          CALLS   #2,G^LIB$GET_VM         ; Allocate the new buffer
                07 50   E9  0098          BLBC    R0,W^DEFAULT            ; Use the default if LIB$GET_VM failed
         55   00EC'CF   D0  009B          MOVL    W^BASEADR,R5            ; Set up to use new buffer
                   07   11  00A0          BRB     55$                     ; OK, entered value is acceptable
                            00A2
                            00A2  DEFAULT:
```

```
         10F1'CF   1000 8F   3C 00A2            MOVZWL  #MAX_DATA,W^DATA_SIZE    ; Use 4096 default value
                   10F1'CF   D6 00A9    55$:    INCL    W^DATA_SIZE              ; Leave room for function code
                                00AD
                                00AD    ;
                                00AD    ; LOCATE THE OPTIONAL DATA IN THE NCB (IF PRESENT) AND REPLACE IT
                                00AD    ; WITH THE MAXIMUM DATA SIZE.
                                00AD    ;
   00AD'CF   00A5'CF      2F    3A 00AD    60$:    LOCC    #^A\/\,W^NCBDESC,W^NCB   ; Locate the slash
                          06    12 00B5            BNEQ    70$                      ; Branch if successful
                   50    00'    3C 00B7            MOVZWL  S^#SS$_ABORT,R0          ; Set up error code (LOCC zeroed R0)
                        00E2    31 00BA            BRW     EXIT                     ; Branch (failure)
                          81    B5 00BD    70$:    TSTW    (R1)+                    ; Skip over word
                          81    95 00BF            TSTB    (R1)+                    ; Skip one more byte
                   81     02    90 00C1            MOVB    #2,(R1)+                 ; Alter count field
              61   10F1'CF      B0 00C4            MOVW    W^DATA_SIZE,(R1)         ; Substitute maximum data size
                          81    B7 00C9            DECW    (R1)+                    ; Leave room for function code
                   61     22    90 00CB            MOVB    #^A/"/,(R1)              ; Add new termination character
                                   00CE    ;
                                   00CE    ; ACCEPT THE CONNECTION REQUEST FROM NETWORK MANAGEMENT LISTENER.
                                   00CE    ;
                                   00CE            $QIOW_S -                        ; Issue connect confirm request
                                   00CE                    EFN=#1,-                 ; Use local event flag #1
                                   00CE                    CHAN=W^DEV_CHAN,-        ; Use the assigned channel
                                   00CE                    FUNC=S^#IO$_ACCESS,-     ; Accept the link
                                   00CE                    IOSB=W^IOSB,-            ; Address of I/O status block
                                   00CE                    P2=#NCBDESC              ; Address of NCB descriptor
                       03 50    E8 00F1            BLBS    R0,80$                   ; Branch on success
                        00A8    31 00F4            BRW     EXIT                     ; Branch (failure)
              50   009D'CF      3C 00F7    80$:    MOVZWL  W^IOSB,R0                ; Get I/O completion status
                       03 50    E8 00FC            BLBS    R0,W^LOOP                ; Branch on success
                        009D    31 00FF            BRW     EXIT                     ; Branch (failure)
                                   0102    ;
                                   0102    ; RECEIVE A MESSAGE FROM NML.
                                   0102    ;
                                   0102    LOOP:   $QIOW_S -                        ; Issue receive request
                                   0102                    EFN=#1,-                 ; Use local event flag #1
                                   0102                    CHAN=W^DEV_CHAN,-        ; Use assigned channel
                                   0102                    FUNC=S^#IO$_READVBLK,-   ; Read virtual block
                                   0102                    IOSB=W^IOSB,-            ; Address of I/O status block
                                   0102                    P1=(R5),-                ; Address of input buffer
                                   0102                    P2=W^DATA_SIZE           ; Length of input buffer
     009D'CF   0000'8F      B1 0123            CMPW    #SS$_DATAOVERUN,W^IOSB   ; Too much data?
                          2A    13 012A            BEQL    90$                      ; Branch if too much data
     009D'CF   0000'8F      B1 012C            CMPW    #SS$_LINKABORT,W^IOSB    ; Did partner issue disconnect abort?
                          67    13 0133            BEQL    W^DISC                   ; Branch if disconnect abort
     009D'CF   0000'8F      B1 0135            CMPW    #SS$_LINKDISCON,W^IOSB   ; Did partner issue sync disconnect?
                          5E    13 013C            BEQL    W^DISC                   ; Branch if sync disconnect
     009D'CF   0000'8F      B1 013E            CMPW    #SS$_ABORT,W^IOSB        ; Was the link aborted (old code)?
                          55    13 0145            BEQL    W^DISC                   ; Branch if link aborted
                       55 50    E9 0147            BLBC    R0,W^EXIT                ; Branch on failure
              50   009D'CF      3C 014A            MOVZWL  W^IOSB,R0                ; Get I/O completion status
                       4D 50    E9 014F            BLBC    R0,W^EXIT                ; Branch on failure
                                   0152    ;
                                   0152    ; MAKE SURE THAT THE FUNCTION CODE IS ZERO.
                                   0152    ;
                          65    95 0152            TSTB    (R5)                     ; Is function code correct?
                          0B    13 0154            BEQL    100$                     ; Branch if zero (valid)
         65   FF 8F    90  0156    90$:     MOVB    #-1,(R5)                  ; Set up failure return code
     009F'CF     01    9B  015A             MOVZBW  #1,W^IOSB+2               ; Set up new length
                 03    11  015F             BRB     REFLECT                   ; Branch and continue
         65      01    90  0161    100$:    MOVB    #1,(R5)                   ; Set up success return code
                           0164    ;
                           0164    ; REFLECT THE MESSAGE BACK TO NML.
                           0164    ;
                           0164    REFLECT:
10F5'CF     009F'CF    B0  0164             MOVW    W^IOSB+2,W^RECV_SIZE      ; Get length of message to reflect
                           016B             $QIOW_S -                         ; Issue transmit request
                           016B                     EFN=#1,-                  ; Use local event flag #1
                           016B                     CHAN=W^DEV_CHAN,-         ; Use assigned channel
                           016B                     FUNC=#IO$_WRITEVBLK,-     ; Write virtual block
                           016B                     IOSB=W^IOSB,-             ; Address of I/O status block
                           016B                     P1=(R5),-                 ; Address of output buffer
                           016B                     P2=W^RECV_SIZE            ; Length of output buffer
              0E 50    E9  018E             BLBC    R0,EXIT                   ; Branch on failure
         50 009D'CF    3C  0191             MOVZWL  W^IOSB,R0                 ; Get I/O completion status
              06 50    E9  0196             BLBC    R0,EXIT                   ; Branch on failure
               FF66    31  0199             BRW     LOOP                      ; Reissue receive request
                           019C    ;
                           019C    ; EXIT WITH STATUS (IN R0).
                           019C    ;
         50     00'    3C  019C    DISC:    MOVZWL  S^#SS$_NORMAL,R0          ; Set up success status code
                           019F    EXIT:    $EXIT_S R0                        ; Exit with status to be displayed
                           01A8                                               ; on exit
                           01A8             .DSABL  LSB                       ; Disable local symbol block
                           01A8             .END    MIRROR                    ; Image transfer address
```

```
$$T1            = 00000001
BASEADR           000000EC R     01
BUFFER            000000F0 R     01
DATA_SIZE         000010F1 R     01
DEFAULT           000000A2 R     02
DEVDESC           0000008E R     01
DEV_CHAN          0000009B R     01
DISC              0000019C R     02
EXIT              0000019F R     02
FOR$CNV_IN_I      ********   X   02
IO$_ACCESS        ********   X   02
IO$_READVBLK      ********   X   02
IO$_WRITEVBLK     ********   X   02
IOSB              0000009D R     01
LIB$GET_VM        ********   X   02
LOG2BUF           0000002A R     01
LOG2DESC          00000022 R     01
LOGNAM2           0000000F R     01
LOGNAME           00000000 R     01
LOOP              00000102 R     02
MAX_DATA        = 00001000
MIRROR            00000000 RG    02
NCB               000000AD R     01
NCBDESC           000000A5 R     01
RECV_SIZE         000010F5 R     01
REFLECT           00000164 R     02
SS$_ABORT         ********   X   02
SS$_DATAOVERUN    ********   X   02
SS$_LINKABORT     ********   X   02
SS$_LINKDISCON    ********   X   02
SS$_NORMAL        ********   X   02
SS$_NOTRAN        ********   X   02
SYS$ASSIGN        ********  GX   02
SYS$EXIT          ********  GX   02
SYS$QIOW          ********  GX   02
SYS$TRNLOG        ********  GX   02

                                     +----------------+
                                     ! Psect synopsis !
                                     +----------------+

PSECT name                  Allocation          PSECT No.  Attributes
----------                  ----------          ---------  ----------
. ABS .                     00000000  (    0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD   NOWRT NOVEC BYTE
MIRROR$DATA                 000010F9  ( 4345.)  01 (  1.)  NOPIC   USR   CON   REL   LCL   SHR NOEXE   RD     WRT NOVEC BYTE
MIRROR$CODE                 000001A8  (  424.)  02 (  2.)  NOPIC   USR   CON   REL   LCL NOSHR   EXE   RD   NOWRT NOVEC BYTE

                                   +------------------------+
                                   ! Performance indicators !
                                   +------------------------+

Phase                    Page faults    CPU Time       Elapsed Time
-----                    -----------    --------       ------------
Initialization                47        00:00:00.11    00:00:01.65
Command processing           143        00:00:00.57    00:00:03.60
Pass 1                       129        00:00:01.68    00:00:07.88
Symbol table sort              0        00:00:00.02    00:00:00.04
```

```
Pass 2                               68      00:00:00.76     00:00:03.61
Symbol table output                   4      00:00:00.04     00:00:00.18
Psect synopsis output                 2      00:00:00.03     00:00:00.08
Cross-reference output                0      00:00:00.00     00:00:00.00
Assembler run totals                395      00:00:03.23     00:00:17.05

The working set limit was 1050 pages.
8044 bytes (16 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 36 non-local and 11 local symbols.
292 source lines were read in Pass 1, producing 17 object records in Pass 2.
8 pages of virtual memory were used to define 8 macros.

                                              +----------------------------+
                                              ! Macro library statistics !
                                              +----------------------------+

Macro library name                               Macros defined
------------------                               --------------
_$255$DUA28:[SYSLIB]STARLET.MLB;2                          8

86 GETS were required to define 8 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:MIRROR/OBJ=OBJ$:MIRROR MSRC$:MIRROR/UPDATE=(ENH$:MIRROR)
```

0235 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

MIRROR

MMGADJWSL
LIS

MIRROR
MAP

MMGTST

MMGADJWSL
MAP

MMGCRTDEL
MAP

MIRROR
LIS

MMGEXPCNT
MAP

MMGNSPWSL
MAP

MMGXQUOTA
MAP

MMGCRDLSP
MAP

MMGCRTFIL
MAP

MMGLKWULW
MAP

MMGSETPRT
MAP