1 3

```
   CCCCCCCCCCCC  LLL                  IIIIIIIII      UUU         UUU  TTTTTTTTTTTTTTT   LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII      UUU         UUU  TTTTTTTTTTTTTTT   LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII      UUU         UUU  TTTTTTTTTTTTTTT   LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
CCC              LLL                     III         UUU         UUU        TTT         LLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII      UUUUUUUUUUUUUUU        TTT         LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII      UUUUUUUUUUUUUUU        TTT         LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII      UUUUUUUUUUUUUUU        TTT         LLLLLLLLLLLLLLL
```

```
**FILE**ID**SHOWMAIN                                    H  5
```

```
  SSSSSSSS  HH      HH    000000    WW      WW  MM      MM    AAAAAA      IIIIII    NN      NN
  SSSSSSSS  HH      HH    000000    WW      WW  MM      MM    AAAAAA      IIIIII    NN      NN
SS          HH      HH  00      00  WW      WW  MMMM  MMMM  AA      AA      II      NN      NN
SS          HH      HH  00      00  WW      WW  MMMM  MMMM  AA      AA      II      NN      NN
SS          HH      HH  00      00  WW      WW  MM  MM  MM  AA      AA      II      NNNN    NN
SS          HH      HH  00      00  WW      WW  MM  MM  MM  AA      AA      II      NNNN    NN
  SSSSSS    HHHHHHHHHH  00      00  WW      WW  MM      MM  AA      AA      II      NN  NN  NN
  SSSSSS    HHHHHHHHHH  00      00  WW      WW  MM      MM  AA      AA      II      NN  NN  NN
        SS  HH      HH  00      00  WW  WW  WW  MM      MM  AAAAAAAAAA      II      NN    NNNN
        SS  HH      HH  00      00  WW  WW  WW  MM      MM  AAAAAAAAAA      II      NN    NNNN
        SS  HH      HH  00      00  WWWW  WWWW  MM      MM  AA      AA      II      NN      NN      ....
        SS  HH      HH  00      00  WWWW  WWWW  MM      MM  AA      AA      II      NN      NN      ....
SSSSSSSS    HH      HH    000000    WW      WW  MM      MM  AA      AA    IIIIII    NN      NN      ....
SSSSSSSS    HH      HH    000000    WW      WW  MM      MM  AA      AA    IIIIII    NN      NN      ....

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
```

```
0 MODULE showmain (IDENT='V04-000',
0        MAIN=show$start,
0        ADDRESSING_MODE(EXTERNAL=GENERAL,
0                NONEXTERNAL=LONG_RELATIVE)
0        ) =
1 BEGIN
1 !
1 !
1 !****************************************************************** *******
1 !*                                                                         *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
1 !*  ALL RIGHTS RESERVED.                                                   *
1 !*                                                                         *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
1 !*  TRANSFERRED.                                                           *
1 !*                                                                         *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
1 !*  CORPORATION.                                                           *
1 !*                                                                         *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
1 !*                                                                         *
1 !*                                                                         *
1 !***************************************************************************
1 !
1 !
1 !++
1 !
1 ! FACILITY:  SHOW utility
1 !
1 ! ABSTRACT:
1 !       This module contains the command processing and dispatch routines.
1 !
1 ! ENVIRONMENT:
1 !       VAX native, user mode.
1 !
1 ! AUTHOR:  Gerry Smith           CREATION DATE:  25-Jun-1982
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-005 AEW0002         Anne E. Warner          10-Jul-1984
1 !               Remove the qualifier MSCP as this module is now called
1 !               from SHOW$DEVICES as SHOW DEVICES/SERVED
1 !
1 !       V03-004 AEW0001         Anne E. Warner          12-Apr-1984
1 !               Add the qualifier MSCP which enables the branch to
1 !               the module SHOW$MSCP which shows information on mass
1 !               served devices.
1 !
1 !       V03-003 MCN0147         Maria del C. Nasr       04-Feb-1984
1 !               Take out reference to external routine FILE_ERROR, since
```

```
;    58          0058  1 !                 it is not used.
;    59          0059  1 !
;    60          0060  1 !         V03-002 GAS0174         Gerry Smith             25-Aug-1983
;    61          0061  1 !                 Split the I/O routines into a different module, SHOWIO.
;    62          0062  1 !                 Also added SHOW BROADCAST.
;    63          0063  1 !
;    64          0064  1 !         V03-001 GAS0154         Gerry Smith              7-Jul-1983
;    65          0065  1 !                 Add SHOW AUDIT.
;    66          0066  1 !
;    67          0067  1 !--
```

```
  1 LIBRARY 'SYS$LIBRARY:STARLET';            ! VAX/VMS common definitions
  1 REQUIRE 'SRC$:SHOWDEF';                   ! SHOW common definitions
  1
  1 !
  1 ! Macro to set up two associated tables.  The first table is a list of
  1 ! descriptor addresses.  These descriptors contain the option names.
  1 ! The second table is a corresponding list of addresses of option routines.
  1 !
  1 ! If a new option is added to SET or SHOW, all that is required in this
  1 ! module is to add one line of code, the option name, e.g. WORKING_SET.
  1 ! Then, the name of the global routine that is dispatched to from this
  1 ! routine will be named SHOW$WORKING_SET.
  1 !
  1 MACRO
  1
  1     option_name [option] = %EXACTSTRING(4, 0, option)%,
  1
  1     option_address [option] = %NAME(%STRING('show$',%STRING(option)))%,
  1
M 1     make_table (name) =
M 1         LITERAL %NAME(%STRING(name,'_table_length')) = %LENGTH - 1;
M 1         EXTERNAL ROUTINE option_address(%REMAINING);
M 1         OWN
M 1             %NAME(%STRING(name,'_option')) : VECTOR[%LENGTH - 1]
M 1             INITIAL (option_name(%REMAINING)),
M 1
M 1             %NAME(%STRING(name,'_routine')) : VECTOR[%LENGTH - 1]
  1             INITIAL (option_address(%REMAINING));%;
  1
```

```
;  100          0196  1 FORWARD ROUTINE
;  101          0197  1     show$start,                             ! Main routine for SHOW
;  102          0198  1     handler;                                ! Condition handler for SHOW
;  103          0199  1
;  104          0200  1 EXTERNAL ROUTINE
;  105          0201  1     open_output : NOVALUE,                  ! Open output file
;  106          0202  1     show$write_line : NOVALUE,              ! Write a line to output file
;  107          0203  1     show$print_line : NOVALUE,              ! Print an already-formatted line
;  108          0204  1     cli$get_value,                          ! Get qualifier value
;  109          0205  1     cli$present;                            ! Test if qualifier present
;  110          0206  1
;  111          0207  1 GLOBAL show$exit_status : $BBLOCK[4]
;  112          0208  1                           INITIAL(1);
;  113          0209  1
;  114          0210  1 !
;  115          0211  1 ! Set up a table of all options, and another table pointing to the address
;  116          0212  1 ! of the routine for each option.
;  117          0213  1 !
;  118          0214  1
;  119        P 0215  1 make_table (show,
;  120        P 0216  1                 accounting,
;  121        P 0217  1                 audit,
;  122        P 0218  1                 broadcast,
;  123        P 0219  1                 devices,
;  124        P 0220  1                 errors,
;  125        P 0221  1                 logical,
;  126        P 0222  1                 magtape,
;  127        P 0223  1                 memory,
;  128        P 0224  1                 network,
;  129        P 0225  1                 printer,
;  130        P 0226  1                 process,
;  131        P 0227  1                 rms_default,
;  132        P 0228  1                 system,
;  133        P 0229  1                 terminal,
;  134        P 0230  1                 users,
;  135          0231  1                 working_set);
;  136          0232  1
;  137          0233  1
;  139          0234  1 ROUTINE show$start =
;  140          0235  2 BEGIN
;  141          0236  2
;  142          0237  2 !---
;  143          0238  2 !
;  144          0239  2 ! This is the main program.  It gathers all the command inputs, and then
;  145          0240  2 ! dispatches to the appropriate routines.
;  146          0241  2 !
;  147          0242  2 !---
;  148          0243  2
;  149          0244  2 LOCAL
;  150          0245  2     status,
;  151          0246  2     option : $BBLOCK[dsc$c_s_bln];
;  152          0247  2
;  153          0248  2 ENABLE handler;                                  ! Enable the condition handler
;  154          0249  2
;  155          0250  2 !
;  156          0251  2 ! Open and connect to the output file.
;  157          0252  2 !
;  158          0253  2 open_output();
;  159          0254  2
;  160          0255  2
;  161          0256  2 !
;  162          0257  2 ! Interrogate the CLI to determine which option one was requested, and
;  163          0258  2 ! dispatch to the appropriate routine.
;  164          0259  2 !
;  165          0260  2 $init_dyndesc(option);
;  166          0261  2
;  167          0262  3 IF NOT (status = cli$get_value(%ASCID 'OPTION', option))
;  168          0263  2 THEN SIGNAL_STOP(.status);
;  169          0264  2
;  170          0265  2 option[dsc$w_length] = MINU (.option[dsc$w_length], 4);
;  171          0266  2
;  172          0267  2 INCR index FROM 0 TO show_table_length - 1 DO
;  173          0268  3     BEGIN
;  174          0269  3     IF CH$EQL(.option[dsc$w_length], .option[dsc$a_pointer],
;  175          0270  3               .option[dsc$w_length], show_option[.index])
;  176          0271  3     THEN
;  177          0272  4         BEGIN
;  178          0273  4         (.show_routine[.index])();
;  179          0274  4         EXITLOOP
;  180          0275  3         END;
;  181          0276  2     END;
;  182          0277  2
;  183          0278  2 RETURN .show$exit_status OR sts$m_inhib_msg;          ! Exit with no message
;  184          0279  1 END;


                                                            .TITLE  SHOWMAIN
                                                            .IDENT  \V04-000\

                                                            .PSECT  $PLIT$,NOWRT,NOEXE,2

                       00  00  4E  4F  49  54  50  4F  00000 P.AAB:  .ASCII  \OPTION\<0><0>                        ;
                                           010E0006  00008 P.AAA:  .LONG   17694726                              ;
                                           00000000'  0000C         .ADDRESS P.AAB                               ;
```

```
                                                                    .PSECT  $OWN$,NOEXE,2

                                    4F  43  43  41  00000 SHOW_OPTION:
                                                                    .ASCII  \ACCO\                               ;
                                    49  44  55  41  00004           .ASCII  \AUDI\                               ;
                                    41  4F  52  42  00008           .ASCII  \BROA\                               ;
                                    49  56  45  44  0000C           .ASCII  \DEVI\                               ;
                                    4F  52  52  45  00010           .ASCII  \ERRO\                               ;
                                    49  47  4F  4C  00014           .ASCII  \LOGI\                               ;
                                    54  47  41  4D  00018           .ASCII  \MAGT\                               ;
                                    4F  4D  45  4D  0001C           .ASCII  \MEMO\                               ;
                                    57  54  45  4E  00020           .ASCII  \NETW\                               ;
                                    4E  49  52  50  00024           .ASCII  \PRIN\                               ;
                                    43  4F  52  50  00028           .ASCII  \PROC\                               ;
                                    5F  53  4D  52  0002C           .ASCII  \RMS_\                               ;
                                    54  53  59  53  00030           .ASCII  \SYST\                               ;
                                    4D  52  45  54  00034           .ASCII  \TERM\                               ;
                                    52  45  53  55  00038           .ASCII  \USER\                               ;
                                    4B  52  4F  57  0003C           .ASCII  \WORK\                               ;
00000000G 00000000G 00000000G 00000000G 00000000G 00000000G 00040 SHOW_ROUTINE:
                                                                    .ADDRESS SHOW$ACCOUNTING, SHOW$AUDIT, -      ;
00000000G 00000000G 00000000G 00000000G 00000000G 00000000G 00058            SHOW$BROADCAST, SHOW$DEVICES, -     ;
                    00000000G 00000000G 00000000G 00000000G 00070            SHOW$ERRORS, SHOW$LOGICAL, SHOW$MAGTAPE, - ;
                                                                             SHOW$MEMORY, SHOW$NETWORK, SHOW$PRINTER, - ;
                                                                             SHOW$PROCESS, SHOW$RMS_DEFAULT, -   ;
                                                                             SHOW$SYSTEM, SHOW$TERMINAL, SHOW$USERS, - ;
                                                                             SHOW$WORKING_SET                    ;

                                                                    .PSECT  $GLOBAL$,NOEXE,2

                                          00000001  00000 SHOW$EXIT_STATUS::
                                                                    .LONG   1                                    ;

                                                                    .EXTRN  OPEN_OUTPUT, SHOW$WRITE_LINE
                                                                    .EXTRN  SHOW$PRINT_LINE
                                                                    .EXTRN  CLI$GET_VALUE, CLI$PRESENT
                                                                    .EXTRN  SHOW$ACCOUNTING
                                                                    .EXTRN  SHOW$AUDIT, SHOW$BROADCAST
                                                                    .EXTRN  SHOW$DEVICES, SHOW$ERRORS
                                                                    .EXTRN  SHOW$LOGICAL, SHOW$MAGTAPE
                                                                    .EXTRN  SHOW$MEMORY, SHOW$NETWORK
                                                                    .EXTRN  SHOW$PRINTER, SHOW$PROCESS
                                                                    .EXTRN  SHOW$RMS_DEFAULT
                                                                    .EXTRN  SHOW$SYSTEM, SHOW$TERMINAL
                                                                    .EXTRN  SHOW$USERS, SHOW$WORKING_SET

                                                                    .PSECT  $CODE$,NOWRT,2

                                                001C  00000 SHOW$START:
                                                                    .WORD   Save R2,R3,R4                        ; 0234
                           5E           08  C2  00002               SUBL2   #8, SP                               ;
                           6D     006A  CF  DE  00005               MOVAL   6$, (FP)                             ; 0235
                00000000G  00           00  FB  0000A               CALLS   #0, OPEN_OUTPUT                      ; 0253
                           6E 020E0300  8F  D0  00011               MOVL    #34471936, OPTION                    ; 0260
                                    04  AE  D4  00018               CLRL    OPTION+4                             ;
                                        5E  DD  0001B               PUSHL   SP                                   ; 0262
                       00000000'        EF  9F  0001D               PUSHAB  P.AAA                                ;
```

```
                  00000000G  00              02  FB 00023          CALLS    #2, CLI$GET_VALUE                    :
                             09              50  E8 0002A          BLBS     STATUS, 1$                           :
                                             50  DD 0002D          PUSHL    STATUS                               : 0263
                  00000000G  00              01  FB 0002F          CALLS    #1, LIB$STOP                         :
                             50              6E  3C 00036 1$:      MOVZWL   OPTION, R0                           : 0265
                             04              50  B1 00039          CMPW     R0, #4                               :
                                             03  1B 0003C          BLEQU    2$                                   :
                             50              04  D0 0003E          MOVL     #4, R0                               :
                             6E              50  B0 00041 2$:      MOVW     R0, OPTION                           :
                                             54  D4 00044          CLRL     INDEX                                : 0269
                                 00000000'EF44  DF 00046 3$:      PUSHAL   SHOW_OPTION[INDEX]                   : 0270
               9E        0R  BE        04    AE  29 0004D          CMPC3    OPTION, @OPTION+4, @(SP)+            :
                                             0D  12 00053          BNEQ     4$                                   :
                             50  00000000'EF44  D0 00055          MOVL     SHOW_ROUTINE[INDEX], R0              : 0273
                             60              00  FB 0005D          CALLS    #0, (R0)                             :
                                             04  11 00060          BRB      5$                                   : 0272
               E0            54              0F  F3 00062 4$:      AOBLEQ   #15, INDEX, 3$                       : 0267
               50 00000000'  EF  10000000    8F  C9 00066 5$:      BISL3    #268435456, SHOW$EXIT_STATUS, R0     : 0278
                                                 04 00072          RET                                           : 0279
                                               0000 00073 6$:      .WORD    Save nothing                         : 0235
                                             7E  D4 00075          CLRL     -(SP)                                :
                                             5E  DD 00077          PUSHL    SP                                   :
                             7E        04    AC  7D 00079          MOVQ     4(AP), -(SP)                         :
                  00000000V  EF              03  FB 0007D          CALLS    #3, HANDLER                          :
                                                 04 00084          RET                                           :

; Routine Size:  133 bytes,    Routine Base:  $CODE$ + 0000
;  186        0280   1 ROUTINE handler (sigargs, mechargs) =
;  187        0281   2 BEGIN
;  188        0282   2
;  189        0283   2 !---
;  190        0284   2 !
;  191        0285   2 ! This routine is a condition handler established by the main
;  192        0286   2 ! routine.  It saves the most severe condition for the exit status.
;  193        0287   2 !
;  194        0288   2 !---
;  195        0289   2
;  196        0290   2 MAP
;  197        0291   2     sigargs : REF $BBLOCK,
;  198        0292   2     mechargs : REF $BBLOCK;
;  199        0293   2 BIND
;  200        0294   2     signame = sigargs[chf$l_sig_name] : $BBLOCK;        ! Name of signal
;  201        0295   2
;  202        0296   2
;  203        0297   2 IF .show$exit_status EQL 1                          ! If no errors yet, use
;  204        0298   2 THEN show$exit_status = .signame;                   ! this one.
;  205        0299   2
;  206        0300   2 IF NOT .signame                                     ! If an error signal
;  207        0301   2 AND .signame[sts$v_severity]                        ! and severity is worse
;  208        0302   2 GTRU .$BBLOCK[show$exit_status, sts$v_severity]     ! than current saved severity
;  209        0303   2 THEN show$exit_status = .signame;                   ! then save it for exit
;  210        0304   2
;  211        0305   2 RETURN ss$_resignal;                                ! Resignal to get message
;  212        0306   1 END;



                                          0004 00000 HANDLER:.WORD   Save R2                                 ; 0280
                           52 00000000'  EF   9E 00002         MOVAB   SHOW$EXIT_STATUS, R2                    ;
                 50       04  AC         04   C1 00009         ADDL3   #4, SIGARGS, R0                         ; 0294
                              01         62   D1 0000E         CMPL    SHOW$EXIT_STATUS, #1                    ; 0297
                                         03   12 00011         BNEQ    1$                                      ;
                              62         60   D0 00013         MOVL    (R0), SHOW$EXIT_STATUS                  ; 0298
                              0F         60   E8 00016 1$:     BLBS    (R0), 2$                                ; 0300
          51      62          03         00   EF 00019         EXTZV   #0, #3, SHOW$EXIT_STATUS, R1            ; 0302
          51      60          03         00   ED 0001E         CMPZV   #0, #3, (R0), R1                        ;
                                         03   1B 00023         BLEQU   2$                                      ;
                              62         60   D0 00025         MOVL    (R0), SHOW$EXIT_STATUS                  ; 0303
                              50   0918  8F   3C 00028 2$:     MOVZWL  #2328, R0                               ; 0305
                                         04   0002D            RET                                             ; 0306

; Routine Size:  46 bytes,    Routine Base:  $CODE$ + 0085
;  214          0307  1 END
;  215          0308  0 ELUDOM



                                                                                              .EXTRN  LIBSSTOP

;                                            PSECT SUMMARY
;
;       Name                             Bytes                         Attributes
;
;  $GLOBAL$                                  4  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                                   128  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                                   16  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                                  179  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                                  Library Statistics
;
;                                                  -------- Symbols --------        Pages        Processing
;       File                                       Total     Loaded   Percent       Mapped       Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1                9776         23         0        581          00:01.0





;                                             COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:SHOWMAIN/OBJ=OBJ$:SHOWMAIN MSRC$:SHOWMAIN/UPDATE=(ENH$:SHOWMAIN)

; Size:          179 code + 148 data bytes
; Run Time:         00:06.6
; Elapsed Time:     00:24.3
; Lines/CPU Min:      2808
; Lexemes/CPU-Min: 24866
; Memory Used:  69 pages
; Compilation Complete
```

0057 AH-BT13A-SE DIGITAL EQUIPMENT CORPORATION
VAX/VMS V4.0 CONFIDENTIAL AND PROPRIETARY

SHOWTERM
LIS

SHOWMISC
LIS

SHOWPROC
LIS

SHOWSYS
LIS

SHOWMAIN
LIS

SHOWMSCP
LIS

SHOWQUE
LIS

SHOWMSG
LIS