M 11

```
DDDDDDDDDDDD        CCCCCCCCCCCC  LLL
DDDDDDDDDDDD        CCCCCCCCCCCC  LLL
DDDDDDDDDDDD        CCCCCCCCCCCC  LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDD         DDD  CCC              LLL
DDDDDDDDDDDD        CCCCCCCCCCCC  LLLLLLLLLLLLLLL
DDDDDDDDDDDD        CCCCCCCCCCCC  LLLLLLLLLLLLLLL
DDDDDDDDDDDD        CCCCCCCCCCCC  LLLLLLLLLLLLLLL
```

```
**FILE**ID**CLINT                                                        M 5




  CCCCCCCC  LL            IIIIII    NN      NN  TTTTTTTTTT
  CCCCCCCC  LL            IIIIII    NN      NN  TTTTTTTTTT
CC          LL              II      NN      NN      TT
CC          LL              II      NN      NN      TT
CC          LL              II      NNNN    NN      TT
CC          LL              II      NNNN    NN      TT
CC          LL              II      NN  NN  NN      TT
CC          LL              II      NN  NN  NN      TT
CC          LL              II      NN    NNNN      TT
CC          LL              II      NN    NNNN      TT
CC          LL              II      NN      NN      TT          ....
CC          LL              II      NN      NN      TT          ....
  CCCCCCCC  LLLLLLLLLL    IIIIII    NN      NN      TT          ....
  CCCCCCCC  LLLLLLLLLL    IIIIII    NN      NN      TT          ....


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 cli$interface (IDENT='V04-000',
0         ADDRESSING_MODE(NONEXTERNAL=LONG_RELATIVE,
0                         EXTERNAL=GENERAL)) =
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 ! FACILITY:  Command language interface routines
1 !
1 ! ABSTRACT:
1 !
1 !       These routines are used to enable a newly activated
1 !       image to obtain the command parameters and qualifiers
1 !       from the command language interpreter.
1 !
1 ! ENVIRONMENT:
1 !
1 !       VAX/VMS operating system. unprivileged user mode,
1 !
1 ! AUTHOR:  Peter George, October 1981
1 !
1 ! Modified by:
1 !
1 !       V03-011 PCG0013         Peter George            17-Feb-1983
1 !               Include INTDEF instead of CLIDEF.
1 !
1 !       V03-010 PCG0012         Peter George            16-Feb-1983
1 !               Fix typo.
1 !
1 !       V03-009 PCG0011         Peter George            08-Feb-1983
1 !               No need to resignal STR$COPY_DX errors.
1 !
1 !       V03-008 PCG0010         Peter George            14-Dec-1982
```

```
;    58        0058  1 !                Allocate local descriptor for command string.
;    59        0059  1 !
;    60        0060  1 !        V03-007 PCG0009         Peter George            29-Nov-1982
;    61        0061  1 !                Add argument to str$analyze_sdesc call.
;    62        0062  1 !
;    63        0063  1 !        V03-006 PCG0008         Peter George            22-Nov-1982
;    64        0064  1 !                Add CLI$NEXT_QUAL.  Fix optional length assignment
;    65        0065  1 !                in CLI$GET_VALUE.  Analyze input string descriptors.
;    66        0066  1 !
;    67        0067  1 !        V03-005 PCG0007         Peter George            18-Oct-1982
;    68        0068  1 !                Add prompt argument to CLI$DCL_PARSE.
;    69        0069  1 !
;    70        0070  1 !        V03-004 PCG0006         Peter George            12-Oct-1982
;    71        0071  1 !                Remove declaration of CLI$END_PARSE.
;    72        0072  1 !
;    73        0073  1 !        V03-003 PCG0005         Peter George            10-Oct-1982
;    74        0074  1 !                Look in SYS$LIBRARY for UTILDEF.
;    75        0075  1 !
;    76        0076  1 !        V03-002 PCG0004         Peter George            30-Sep-1982
;    77        0077  1 !                Use INT data structure.
;    78        0078  1 !                Add prompt and continuation routines to CLI$DCL_PARSE.
;    79        0079  1 !                Add value address to CLI$GET_VALUE.
;    80        0080  1 !                Add user argument to CLI$DISPATCH.
;    81        0081  1 !                Delete CLI$END_PARSE.
;    82        0082  1 !
;    83        0083  1 !        V03-001 PCG0003         Peter George            18-Jun-1982
;    84        0084  1 !                Add CLI$_LOCPRES and CLI$_LOCNEG statuses.
;    85        0085  1 !                Signal errors from STR$COPY_DX.
;    86        0086  1 !
;    87        0087  1 !--
;    88        0088  1
;    89        0089  1 !
;    90        0090  1 ! Include files
;    91        0091  1 !
;    92        0092  1 LIBRARY 'SYS$LIBRARY:STARLET';
;    93        0093  1
;    94        0094  1 REQUIRE 'SYS$LIBRARY:UTILDEF';          ! Common VMS BLISS definitions
;    95        0270  1
;    96        0271  1 REQUIRE 'LIB$:INTDEF';                 ! Interface definitions
;     98        0297   1 !
;     99        0298   1 ! Table of contents
;    100        0299   1 !
;    101        0300   1 FORWARD ROUTINE
;    102        0301   1     cli$present,                           ! Determine if entity present
;    103        0302   1     cli$get_value,                         ! Get value of entity
;    104        0303   1     cli$dcl_parse,                         ! Parse a command line
;    105        0304   1     cli$dispatch,                          ! Dispatch to user processing routine
;    106        0305   1     cli$next_qual;                         ! Get next qualifier
;    107        0306   1
;    108        0307   1 EXTERNAL ROUTINE
;    109        0308   1     sys$cli,                               ! CLI callback routine
;    110        0309   1     lib$get_vm,                            ! Allocate virtual memory
;    111        0310   1     lib$free_vm,                           ! Deallocate virtual memory
;    112        0311   1     str$analyze_sdesc,                     ! Analyze string descriptor
;    113        0312   1     str$copy_dx;                           ! Copy to any class string
;    114        0313   1
;    115        0314   1 !
;    116        0315   1 ! External message definitions
;    117        0316   1 !
;    118        0317   1 EXTERNAL LITERAL
;    119        0318   1     cli$_nocomd,                           ! Prompt was ctrl/z-ed
;    120        0319   1     cli$_concat,                           ! Value is concatenated
;    121        0320   1     cli$_comma,                            ! Value is comma separated
;    122        0321   1     cli$_present,                          ! Entity is explicitly globally present
;    123        0322   1     cli$_negated,                          ! Entity is explicitly not globally present
;    124        0323   1     cli$_locpres,                          ! Entity is explicitly locally present
;    125        0324   1     cli$_locneg,                           ! Entity is explicitly not locally present
;    126        0325   1     cli$_defaulted,                        ! Entity is implicitly present
;    127        0326   1     cli$_absent;                           ! Entity is implicitly not present
```

```
:  129          0327  1 GLOBAL ROUTINE cli$present (name) =
:  130          0328  1
:  131          0329  1 !---
:  132          0330  1 !
:  133          0331  1 !       Determine if an entity is present on the command line.
:  134          0332  1 !
:  135          0333  1 ! Inputs:
:  136          0334  1 !
:  137          0335  1 !       name = Address of entity name descriptor
:  138          0336  1 !
:  139          0337  1 ! Outputs:
:  140          0338  1 !
:  141          0339  1 !       routine value = True if present, else false.
:  142          0340  1 !---
:  143          0341  1
:  144          0342  2 BEGIN
:  145          0343  2
:  146          0344  2 LOCAL
:  147          0345  2     req_desc :  BBLOCK [cli$c_reqdesc], ! Request descriptor block
:  148          0346  2     rpw :       BBLOCK [cli$c_workarea],! Result parse work area
:  149          0347  2     req_flags : BITVECTOR [32];         ! Request flags array
:  150          0348  2
:  151          0349  2 CH$FILL(0, cli$c_reqdesc, req_desc);            ! Zero request descriptor block
:  152          0350  2 req_desc [int_b_type] = cli$k_present;          ! Set request type
:  153          0351  2 req_desc [int_l_getvm] = lib$get_vm;            ! Set address of get vm routine
:  154          0352  2 req_desc [int_l_freevm] = lib$free_vm;          ! Set address of free vm routine
:  155          0353  2 !
:  156          0354  2 ! Set entity name
:  157          0355  2 !
:  158          0356  2 str$analyze_sdesc (.name, req_desc [int_w_entlen], req_desc [int_l_entaddr]);
:  159          0357  2 RETURN (SYS$CLI (req_desc, rpw, req_flags));    ! Call callback utility
:  160          0358  1 END;

                                                                        .TITLE  CLI$INTERFACE
                                                                        .IDENT  \V04-000\

                                                                        .EXTRN  SYS$CLI, LIB$GET_VM
                                                                        .EXTRN  LIB$FREE_VM, STR$ANALYZE_SDESC
                                                                        .EXTRN  STR$COPY_DX, CLI$_NOCOMD
                                                                        .EXTRN  CLI$_CONCAT, CLI$_COMMA
                                                                        .EXTRN  CLI$_PRESENT, CLI$_NEGATED
                                                                        .EXTRN  CLI$_LOCPRES, CLI$_LOCNEG
                                                                        .EXTRN  CLI$_DEFAULTED, CLI$_ABSENT

                                                                        .PSECT  $CODE$,NOWRT,2

                                                     003C 00000         .ENTRY  CLI$PRESENT, Save R2,R3,R4,R5            ; 0327
                                   5E      FF60   CE   9E 00002         MOVAB   -160(SP), SP                             ;
         1C           00           6E             00   2C 00007         MOVC5   #0, (SP), #0, #28, REQ_DESC              ; 0349
                                                  E4   AD 0000C
                            E4     AD        50   8F   90 0000E         MOVB    #80, REQ_DESC                            ; 0350
                            F4     AD 00000000G   00   9E 00013         MOVAB   LIB$GET_VM, REQ_DESC+16                  ; 0351
                            F8     AD 00000000G   00   9E 0001B         MOVAB   LIB$FREE_VM, REQ_DESC+20                 ; 0352
                                             F0   AD   9F 00023         PUSHAB  REQ_DESC+12                              ; 0356
                                             EC   AD   9F 00026         PUSHAB  REQ_DESC+8                               ;
                                             04   AC   DD 00029         PUSHL   NAME                                     ;
```

```
                         00000000G  00            03  FB 0002C         CALLS    #3, STR$ANALYZE_SDESC                      ;
                                                  5E  DD 00033         PUSHL    SP                                         ; 0357
                                             08   AE  9F 00035         PUSHAB   RPW                                        ;
                                             E4   AD  9F 00038         PUSHAB   REQ_DESC                                   ;
                         00000000G  00            03  FB 0003B         CALLS    #3, SYS$CLI                                ;
                                                      04 00042         RET                                                 ; 0358

; Routine Size:  67 bytes,    Routine Base:  $CODE$ + 0000
```

```
1 GLOBAL ROUTINE cli$get_value (name, retdesc, retlength) =
1
1 !---
1 !
1 !       This routine is called to obtain the next value
1 !       associated with a named entity on the command line.
1 !
1 ! Inputs:
1 !
1 !       name = Address of entity name descriptor
1 !       retdesc = Address of a return buffer descriptor
1 !       retlength = Address of a longword to return the value
1 !                   length in
1 !
1 ! Outputs:
1 !
1 !       True = A value was returned (there may be more).
1 !       False = There is no more values associated with entity.
1 !
1 !       All other errors are signaled.
1 !---
1
2 BEGIN
2
2 MAP
2     retdesc : REF BBLOCK;
2
2 BUILTIN
2     NULLPARAMETER;
2
2 LOCAL
2     req_desc :  BBLOCK [cli$c_reqdesc], ! Request descriptor block
2     rpw :       BBLOCK [cli$c_workarea],! Result parse work area
2     req_flags : BITVECTOR [32],         ! Request flags array
2     status;
2
2 CH$FILL(0, cli$c_reqdesc, req_desc);           ! Zero request descriptor block
2 req_desc [int_b_type] = cli$k_getvalue;        ! Set request type
2 req_desc [int_l_getvm] = lib$get_vm;           ! Set address of get vm routine
2 req_desc [int_l_freevm] = lib$free_vm;         ! Set address of free vm routine
2
2 !
2 ! Set entity name
2 !
2 str$analyze_sdesc (.name, req_desc [int_w_entlen], req_desc [int_l_entaddr]);
2 status = sys$cli (req_desc, rpw, req_flags);    ! Call DCL utility
2
2 !
2 ! If return length was requested, then find it now.
2 !
2 IF NOT NULLPARAMETER (3)                       ! If return length requested
2    THEN (.retlength) <0,16> =                  ! then original estimate
2               .req_desc [int_w_retlen];        !
2
2 str$copy_dx (.retdesc, req_desc[int_w_retlen]); ! Copy result into return desc
2
2 IF NOT NULLPARAMETER (3)                       ! If return length requested
```

```
;  219          0416  3     THEN BEGIN                                    ! Then return correct value
;  220          0417  3          LOCAL temp : BBLOCK [dsc$c_s_bln];       !
;  221          0418  3          str$analyze_sdesc (.retdesc,             ! Get latest estimate
;  222          0419  3                  temp [dsc$w_length], temp [dsc$a_pointer]);
;  223          0420  3          IF .temp LSSU .retdesc [dsc$w_length]    ! Return the smaller estimate
;  224          0421  3             THEN (.retlength) <0,16> = .temp;     !
;  225          0422  2          END;
;  226          0423  2
;  227          0424  2 RETURN .status;
;  228          0425  1 END;



                                                        007C 00000         .ENTRY  CLI$GET_VALUE, Save R2,R3,R4,R5,R6            ; 0359
                                        56 00000000G  00  9E 00002         MOVAB   STR$ANALYZE_SDESC, R6                         ;
                                        5E     FF58   CE  9E 00009         MOVAB   -168(SP), SP                                  ;
           1C            00             6E            00  2C 0000E         MOVC5   #0, (SP), #0, #28, REQ_DESC                   ; 0395
                                                 E4   AD     00013                                                               ;
                                   E4   AD       51   8F  90 00015         MOVB    #81, REQ_DESC                                 ; 0396
                                   F4   AD 00000000G  00  9E 0001A         MOVAB   LIB$GET_VM, REQ_DESC+16                       ; 0397
                                   F8   AD 00000000G  00  9E 00022         MOVAB   LIB$FREE_VM, REQ_DESC+20                      ; 0398
                                                 F0   AD  9F 0002A         PUSHAB  REQ_DESC+12                                   ; 0403
                                                 EC   AD  9F 0002D         PUSHAB  REQ_DESC+8                                    ;
                                                 04   AC  DD 00030         PUSHL   NAME                                          ;
                                        66            03  FB 00033         CALLS   #3, STR$ANALYZE_SDESC                         ;
                                                      5E  DD 00036         PUSHL   SP                                            ; 0404
                                                 10   AE  9F 00038         PUSHAB  RPW                                           ;
                                                 E4   AD  9F 0003B         PUSHAB  REQ_DESC                                      ;
                            00000000G   00            03  FB 0003E         CALLS   #3, SYS$CLI                                   ;
                                        52            50  D0 00045         MOVL    R0, STATUS                                    ;
                                        03            6C  91 00048         CMPB    (AP), #3                                      ; 0409
                                                      0A  1F 0004B         BLSSU   1$                                            ;
                                                 0C   AC  D5 0004D         TSTL    12(AP)                                        ;
                                                      05  13 00050         BEQL    1$                                            ;
                                   0C   BC       EC   AD  B0 00052         MOVW    REQ_DESC+8, @RETLENGTH                        ; 0411
                                                 EC   AD  9F 00057 1$:     PUSHAB  REQ_DESC+8                                    ; 0413
                                                 08   AC  DD 0005A         PUSHL   RETDESC                                       ;
                            00000000G   00            02  FB 0005D         CALLS   #2, STR$COPY_DX                               ;
                                        03            6C  91 00064         CMPB    (AP), #3                                      ; 0415
                                                      1F  1F 00067         BLSSU   2$                                            ;
                                                 0C   AC  D5 00069         TSTL    12(AP)                                        ;
                                                      1A  13 0006C         BEQL    2$                                            ;
                                                 08   AE  9F 0006E         PUSHAB  TEMP+4                                        ; 0419
                                                 08   AE  9F 00071         PUSHAB  TEMP                                          ;
                                                 08   AC  DD 00074         PUSHL   RETDESC                                       ;
                                        66            03  FB 00077         CALLS   #3, STR$ANALYZE_SDESC                         ;
      04   AE       08   BC             10            00  ED 0007A         CMPZV   #0, #16, @RETDESC, TEMP                       ; 0420
                                                      05  1B 00081         BLEQU   2$                                            ;
                                   0C   BC       04   AE  B0 00083         MOVW    TEMP, @RETLENGTH                              ; 0421
                                        50            52  D0 00088 2$:     MOVL    STATUS, R0                                    ; 0424
                                                          04 0008B         RET                                                   ; 0425

; Routine Size:  140 bytes,    Routine Base:  $CODE$ + 0043
```

```
1 GLOBAL ROUTINE cli$dcl_parse (command, tables, promptrtn, continrtn, prompt) =
1
1 !---
1 !
1 !       This routine can be called to parse a command line.
1 !
1 ! Inputs:
1 !
1 !       command = address of descriptor of command string
1 !       tables =  address of command tables
1 !       promptrtn = address of user missing parameter prompt routine
1 !       continrtn = address of user line continuation routine
1 !       prompt = address of user prompt string descriptor
1 !
1 ! Outputs:
1 !
1 !       The command is parsed and the command work area is initialized.
1 !
1 !---
1
2 BEGIN
2
2 BUILTIN
2     NULLPARAMETER;
2
2 LITERAL
2     elements = 4;
2
2 LOCAL
2     command_desc : BBLOCK [dsc$c_s_bln],          ! Local command descriptor
2     rtnlist : BBLOCK [4*(elements+1)],            ! Routine list
2     req_desc :  BBLOCK [cli$c_reqdesc],           ! Request descriptor block
2     rpw :       BBLOCK [cli$c_workarea],          ! Result parse work area
2     req_flags : BITVECTOR [32];                   ! Request flags array
2
2 CH$FILL(0, cli$c_reqdesc, req_desc);             ! Zero request descriptor block
2 req_desc [int_b_type] = cli$k_dclparse;          ! Set request type
2 req_desc [int_l_getvm] = lib$get_vm;             ! Set address of get vm routine
2 req_desc [int_l_freevm] = lib$free_vm;           ! Set address of free vm routine
2
2 IF NOT NULLPARAMETER(1)                          ! Set command line descriptor
3    THEN BEGIN
3         str$analyze_sdesc (.command, command_desc [dsc$w_length],
3                           command_desc [dsc$a_pointer]);
3         req_desc [int_l_entaddr] = command_desc;
2         END;
2
2 req_desc [int_l_tables] = .tables;               ! Set address of command tables
2
2 IF NOT NULLPARAMETER(3) OR NOT NULLPARAMETER(4) ! If prompt or continue routine
3    THEN BEGIN
3         CH$FILL(0, 4*(elements+1), rtnlist);    ! Zero the list
3         rtnlist [int_l_listlen] = elements;     ! Set number of elements in list
3         req_desc [int_l_list] = rtnlist;        ! And connect it
2         END;
2
2 IF NOT NULLPARAMETER(3)
```

```
;  287        0483  2      THEN rtnlist [int_l_promptrtn] = .promptrtn; ! Set address of prompt routine
;  288        0484  2   IF NOT NULLPARAMETER(4)
;  289        0485  2      THEN rtnlist [int_l_continrtn] = .continrtn; ! Set address of continue routine
;  290        0486  2   IF NOT NULLPARAMETER(5)                      ! Set prompt descriptor
;  291        0487  2      THEN str$analyze_sdesc (.prompt, rtnlist [int_w_pmptlen],
;  292        0488  2                                  rtnlist [int_l_pmptaddr]);
;  293        0489  2
;  294        0490  2   RETURN (sys$cli (req_desc, rpw, req_flags));     ! Call DCL utility
;  295        0491  1   END;



                                                 007C 00000          .ENTRY  CLI$DCL_PARSE, Save R2,R3,R4,R5,R6             ; 0426
                         56 00000000G       00   9E 00002          MOVAB   STR$ANALYZE_SDESC, R6                           ;
                         5E       FF44      CE   9E 00009          MOVAB   -188(SP), SP                                    ;
      1C          00     6E                 00   2C 0000E          MOVC5   #0, (SP), #0, #28, REQ_DESC                     ; 0461
                                    C8      AD      00013                                                                 ;
                 C8      AD         53      8F   90 00015          MOVB    #83, REQ_DESC                                   ; 0462
                 D8      AD 00000000G       00   9E 0001A          MOVAB   LIB$GET_VM, REQ_DESC+16                         ; 0463
                 DC      AD 00000000G       00   9E 00022          MOVAB   LIB$FREE_VM, REQ_DESC+20                        ; 0464
                                            6C   95 0002A          TSTB    (AP)                                            ; 0466
                                            16   13 0002C          BEQL    1$                                              ;
                                    04      AC   D5 0002E          TSTL    4(AP)                                           ;
                                            11   13 00031          BEQL    1$                                              ;
                                    FC      AD   9F 00033          PUSHAB  COMMAND_DESC+4                                  ; 0469
                                    F8      AD   9F 00036          PUSHAB  COMMAND_DESC                                    ; 0468
                                    04      AC   DD 00039          PUSHL   COMMAND                                         ;
                         66                 03   FB 0003C          CALLS   #3, STR$ANALYZE_SDESC                           ;
                 D4      AD         F8      AD   9E 0003F          MOVAB   COMMAND_DESC, REQ_DESC+12                       ; 0470
                 CC      AD         08      AC   D0 00044 1$:      MOVL    TABLES, REQ_DESC+4                              ; 0473
                         03                 6C   91 00049          CMPB    (AP), #3                                        ; 0475
                                            05   1F 0004C          BLSSU   2$                                              ;
                                    0C      AC   D5 0004E          TSTL    12(AP)                                          ;
                                            0A   12 00051          BNEQ    3$                                              ;
                         04                 6C   91 00053 2$:      CMPB    (AP), #4                                        ;
                                            15   1F 00056          BLSSU   4$                                              ;
                                    10      AC   D5 00058          TSTL    16(AP)                                          ;
                                            10   13 0005B          BEQL    4$                                              ;
      14          00     6E                 00   2C 0005D 3$:      MOVC5   #0, (SP), #0, #20, RTNLIST                      ; 0477
                                    E4      AD      00062                                                                 ;
                 E4      AD                 04   D0 00064          MOVL    #4, RTNLIST                                     ; 0478
                 E0      AD         E4      AD   9E 00068          MOVAB   RTNLIST, REQ_DESC+24                            ; 0479
                         03                 6C   91 0006D 4$:      CMPB    (AP), #3                                        ; 0482
                                            0A   1F 00070          BLSSU   5$                                              ;
                                    0C      AC   D5 00072          TSTL    12(AP)                                          ;
                                            05   13 00075          BEQL    5$                                              ;
                 E8      AD         0C      AC   D0 00077          MOVL    PROMPTRTN, RTNLIST+4                            ; 0483
                         04                 6C   91 0007C 5$:      CMPB    (AP), #4                                        ; 0484
                                            0A   1F 0007F          BLSSU   6$                                              ;
                                    10      AC   D5 00081          TSTL    16(AP)                                          ;
                                            05   13 00084          BEQL    6$                                              ;
                 EC      AD         10      AC   D0 00086          MOVL    CONTINRTN, RTNLIST+8                            ; 0485
                         05                 6C   91 0008B 6$:      CMPB    (AP), #5                                        ; 0486
                                            11   1F 0008E          BLSSU   7$                                              ;
                                    14      AC   D5 00090          TSTL    20(AP)                                          ;
```

```
                                          0C  13 00093          BEQL    7$                                              ;
                                     F4   AD  9F 00095          PUSHAB  RTNLIST+16                                      ; 0488
                                     F0   AD  9F 00098          PUSHAB  RTNLIST+12                                      ; 0487
                                     14   AC  DD 0009B          PUSHL   PROMPT                                          ;
                            66            03  FB 0009E          CALLS   #3, STR$ANALYZE_SDESC                           ;
                                          5E  DD 000A1 7$:      PUSHL   SP                                              ; 0490
                                     08   AE  9F 000A3          PUSHAB  RPW                                             ;
                                     C8   AD  9F 000A6          PUSHAB  REQ_DESC                                        ;
                   00000000G  00            03  FB 000A9          CALLS   #3, SYS$CLI                                     ;
                                              04 000B0          RET                                                     ; 0491

; Routine Size:  177 bytes,    Routine Base:  $CODE$ + 00CF
```

```
0492  1 GLOBAL ROUTINE cli$dispatch (argument) =
0493  1
0494  1 !---
0495  1 !
0496  1 !       This routine can be called to dispatch to any verb processing
0497  1 !       routines if the command has the ROUTINE attribute.
0498  1 !
0499  1 ! Inputs:
0500  1 !
0501  1 !       argument = address of user supplied argument to his own routine
0502  1 !
0503  1 ! Outputs:
0504  1 !
0505  1 !       The verb routine is called (if any).
0506  1 !
0507  1 !       The status passed back from the routine is returned in R0.
0508  1 !       If no routine is specified, success is returned.
0509  1 !---
0510  1
0511  2 BEGIN
0512  2
0513  2 BUILTIN
0514  2     NULLPARAMETER;
0515  2
0516  2 LOCAL
0517  2     req_desc :  BBLOCK [cli$c_reqdesc], ! Request descriptor block
0518  2     rpw :       BBLOCK [cli$c_workarea],! Result parse work area
0519  2     req_flags : BITVECTOR [32];         ! Request flags array
0520  2
0521  2 CH$FILL(0, cli$c_reqdesc, req_desc);             ! Zero request descriptor block
0522  2 req_desc [int_b_type] = cli$k_dispatch;          ! Set request type
0523  2 req_desc [int_l_getvm] = lib$get_vm;             ! Set address of get vm routine
0524  2 req_desc [int_l_freevm] = lib$free_vm;           ! Set address of free vm routine
0525  2 IF NOT NULLPARAMETER (1)
0526  2     THEN req_desc [int_l_entaddr] = .argument;   ! Set address of user argument
0527  2 RETURN (SYS$CLI (req_des., rpw, req_flags));     ! Call callback utility
0528  1 END;
```

```
                                        003C 00000          .ENTRY   CLI$DISPATCH, Save R2,R3,R4,R5                ; 0492
                       5E      FF60   CE  9E 00002          MOVAB    -160(SP), SP                                 ;
      1C        00     6E             00  2C 00007          MOVC5    #0, (SP), #0, #28, REQ_DESC                  ; 0521
                                E4    AD     0000C                                                                ;
                  E4   AD       54    8F  90 0000E          MOVB     #84, REQ_DESC                                ; 0522
                  F4   AD 00000000G   00  9E 00013          MOVAB    LIB$GET_VM, REQ_DESC+16                      ; 0523
                  F8   AD 00000000G   00  9E 0001B          MOVAB    LIB$FREE_VM, REQ_DESC+20                     ; 0524
                                      6C  95 00023          TSTB     (AP)                                         ; 0525
                                      0A  13 00025          BEQL     1$                                           ;
                                04    AC  D5 00027          TSTL     4(AP)                                        ;
                                      05  13 0002A          BEQL     1$                                           ;
                  F0   AD       04    AC  D0 0002C          MOVL     ARGUMENT, REQ_DESC+12                        ; 0526
                                      5E  DD 00031 1$:      PUSHL    SP                                           ; 0527
                                08    AE  9F 00033          PUSHAB   RPW                                          ;
                                E4    AD  9F 00036          PUSHAB   REQ_DESC                                     ;
```

```
                      00000000G  00             03  FB 00039          CALLS   #3, SYS$CLI                              ;
                                                    04 00040          RET                                              ; 0528

; Routine Size:  65 bytes,    Routine Base:  $CODE$ + 0180
;  335          0529  1 GLOBAL ROUTINE cli$next_qual (name) =
;  336          0530  1
;  337          0531  1 !---
;  338          0532  1 !
;  339          0533  1 !      Move to the next command qualifier on the line.
;  340          0534  1 !
;  341          0535  1 ! Inputs:
;  342          0536  1 !
;  343          0537  1 !      name = Address of entity name descriptor
;  344          0538  1 !
;  345          0539  1 ! Outputs:
;  346          0540  1 !
;  347          0541  1 !      routine value = True if present, else false.
;  348          0542  1 !---
;  349          0543  1
;  350          0544  2 BEGIN
;  351          0545  2
;  352          0546  2 MAP
;  353          0547  2     name : REF BBLOCK;
;  354          0548  2
;  355          0549  2 LOCAL
;  356          0550  2     req_desc :  BBLOCK [cli$c_reqdesc], ! Request descriptor block
;  357          0551  2     rpw :       BBLOCK [cli$c_workarea],! Result parse work area
;  358          0552  2     req_flags : BITVECTOR [32];         ! Request flags array
;  359          0553  2
;  360          0554  2 CH$FILL(0, cli$c_reqdesc, req_desc);            ! Zero request descriptor block
;  361          0555  2 req_desc [int_b_type] = cli$k_nextqual;         ! Set request type
;  362          0556  2 req_desc [int_l_getvm] = lib$get_vm;            ! Set address of get vm routine
;  363          0557  2 req_desc [int_l_freevm] = lib$free_vm;          ! Set address of free vm routine
;  364          0558  2 !
;  365          0559  2 ! Set entity name
;  366          0560  2 !
;  367          0561  2 str$analyze_sdesc (.name, req_desc [int_w_entlen], req_desc [int_l_entaddr]);
;  368          0562  2 RETURN (sys$cli (req_desc, rpw, req_flags));    ! Call DCL utility
;  369          0563  1 END;



                                            003C 00000          .ENTRY  CLI$NEXT_QUAL, Save R2,R3,R4,R5           ; 0529
                           5E      FF60  CE  9E 00002          MOVAB   -160(SP), SP                              ;
        1C          00     6E            00  2C 00007          MOVC5   #0, (SP), #0, #28, REQ_DESC               ; 0554
                                     E4  AD     0000C                                                            ;
                      E4   AD        55  8F  90 0000E          MOVB    #85, REQ_DESC                             ; 0555
                      F4   AD 00000000G  00  9E 00013          MOVAB   LIB$GET_VM, REQ_DESC+16                   ; 0556
                      F8   AD 00000000G  00  9E 0001B          MOVAB   LIB$FREE_VM, REQ_DESC+20                  ; 0557
                                     F0  AD  9F 00023          PUSHAB  REQ_DESC+12                               ; 0561
                                     EC  AD  9F 00026          PUSHAB  REQ_DESC+8                                ;
                                     04  AC  DD 00029          PUSHL   NAME                                      ;
                   00000000G  00         03  FB 0002C          CALLS   #3, STR$ANALYZE_SDESC                     ;
                                         5E  DD 00033          PUSHL   SP                                        ; 0562
                                     08  AE  9F 00035          PUSHAB  RPW                                       ;
                                     E4  AD  9F 00038          PUSHAB  REQ_DESC                                  ;
                   00000000G  00         03  FB 0003B          CALLS   #3, SYS$CLI                               ;
                                             04 00042          RET                                               ; 0563
```

```
; Routine Size:  67 bytes,    Routine Base:  $CODE$ + 01C1
;  371          0564  1 END
;  372          0565  0 ELUDOM



;                                       PSECT SUMMARY
;
;       Name                        Bytes                        Attributes
;
;  $CODE$                               516  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        15         0        581        00:01.0



;                                       COMMAND QUALIFIERS

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

; Size:          516 code + 0 data bytes
; Run Time:         00:10.2
; Elapsed Time:     00:40.1
; Lines/CPU Min:     3317
; Lexemes/CPU-Min: 20213
; Memory Used:  93 pages
; Compilation Complete
```

0069 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY