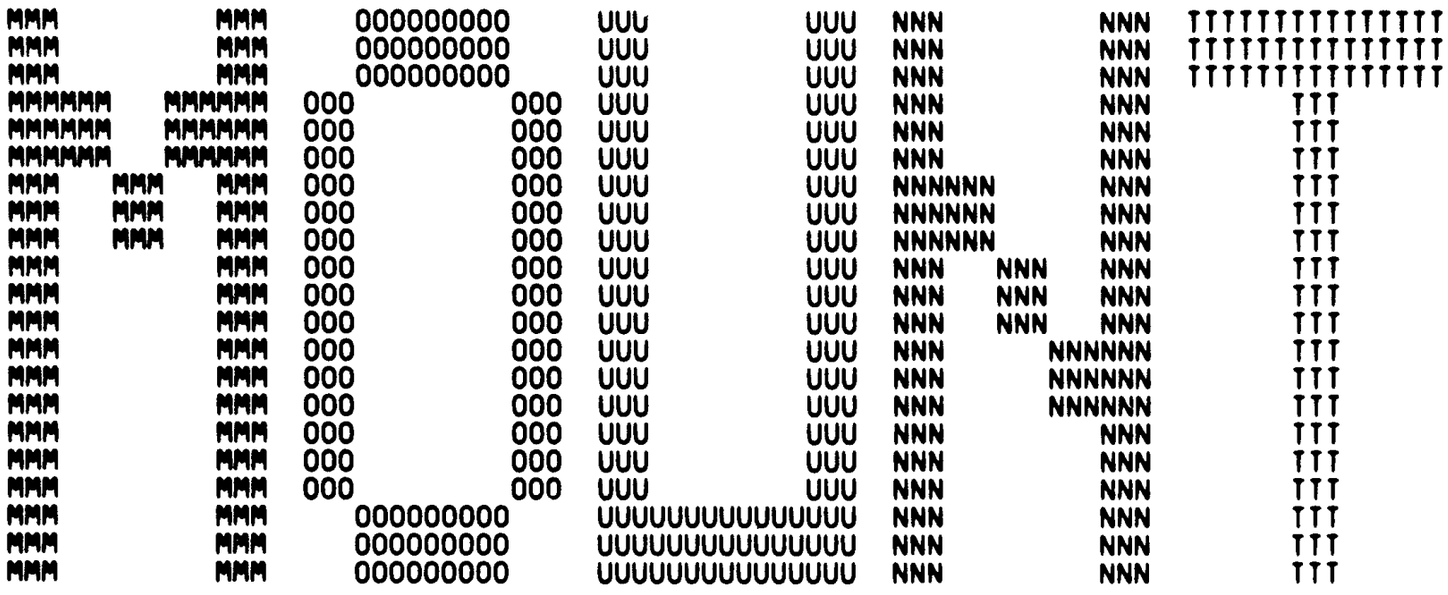
MMM MMM 000000000 UUU UUU NNN NNN TTTTTTTTTTTTTTT
MMM MMM 000000000 UUU UUU NNN NNN TTTTTTTTTTTTTTT
MMM MMM 000000000 UUU UUU NNN NNN TTTTTTTTTTTTTTT
MMMMMMM MMMMMMM 000 000 UUU UUU NNN NNN TTT
MMMMMMM MMMMMMM 000 000 UUU UUU NNN NNN TTT
MMMMMMM MMMMMMM 000 000 UUU UUU NNN NNN TTT
MMM MMM MMM 000 000 UUU UUU NNNNNN NNN TTT
MMM MMM MMM 000 000 UUU UUU NNNNNN NNN TTT
MMM MMM MMM 000 000 UUU UUU NNNNNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNNNNN TTT
MMM MMM 000 000 UUU UUU NNN NNNNNN TTT
MMM MMM 000 000 UUU UUU NNN NNNNNN TTT
MMM MMM 000 000 UUU UUU NNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNN TTT
MMM MMM 000 000 UUU UUU NNN NNN TTT
MMM MMM 000000000 UUUUUUUUUUUUUUUU NNN NNN TTT
MMM MMM 000000000 UUUUUUUUUUUUUUUU NNN NNN TTT
MMM MMM 000000000 UUUUUUUUUUUUUUUU NNN NNN TTT
MMM MMM 000000000 UUUUUUUUUUUUUUUU NNN NNN TTT

```
TTTTTTTTTT  RRRRRRRR    NN      NN  LL            000000      GGGGGGGG
TTTTTTTTTT  RRRRRRRR    NN      NN  LL            000000      GGGGGGGG
    TT      RR      RR  NN      NN  LL          00      00  GG
    TT      RR      RR  NN      NN  LL          00      00  GG
    TT      RR      RR  NNNN    NN  LL          00      00  GG
    TT      RR      RR  NNNN    NN  LL          00      00  GG
    TT      RRRRRRRR    NN  NN  NN  LL          00      00  GG
    TT      RRRRRRRR    NN  NN  NN  LL          00      00  GG
    TT      RR  RR      NN    NNNN  LL          00      00  GG  GGGGGG
    TT      RR  RR      NN    NNNN  LL          00      00  GG  GGGGGG
    TT      RR    RR    NN      NN  LL          00      00  GG      GG      ....
    TT      RR    RR    NN      NN  LL          00      00  GG      GG      ....
    TT      RR      RR  NN      NN  LLLLLLLLLL    000000      GGGGGG      ....
    TT      RR      RR  NN      NN  LLLLLLLLLL    000000      GGGGGG      ....

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
```

```
;     1          0001  0 MODULE TRNLOG (
;     2          0002  0                 LANGUAGE (BLISS32),
;     3          0003  0                 IDENT = 'V04-000'
;     4          0004  0                 ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !*****************************************************************************
;     9          0009  1 !*                                                                           *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                      *
;    13          0013  1 !*                                                                           *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    19          0019  1 !*  TRANSFERRED.                                                              *
;    20          0020  1 !*                                                                           *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    23          0023  1 !*  CORPORATION.                                                              *
;    24          0024  1 !*                                                                           *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*                                                                           *
;    29          0029  1 !*****************************************************************************
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 !
;    33          0033  1 ! FACILITY:  MOUNT Utility Structure Levels 1 & 2
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       This routine performs simple recursive logical name translation.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:
;    40          0040  1 !
;    41          0041  1 !       STARLET operating system, including privileged system services
;    42          0042  1 !       and internal exec routines.
;    43          0043  1 !
;    44          0044  1 !--
;    45          0045  1 !
;    46          0046  1 !
;    47          0047  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  9-Oct-1977  16:10
;    48          0048  1 !
;    49          0049  1 ! MODIFIED BY:
;    50          0050  1 !
;    51          0051  1 !       V03-001 HH0041          Hai Huang                   24-Jul-1984
;    52          0052  1 !               Remove REQUIRE 'LIBD$:[VMSLIB.OBJ]MOUNTMSG.B32'.
;    53          0053  1 !
;    54          0054  1 !       V02-001 ACG0167         Andrew C. Goldstein,    18-Apr-1980  13:39
;    55          0055  1 !               Previous revision history moved to MOUNT.REV
;    56          0056  1 !**
;    57          0057  1
```

```
;    58          0058  1
;    59          0059  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    60          0060  1 REQUIRE 'SRC$:MOUDEF.B32';
```

```
;    62          0592  1 GLOBAL ROUTINE TRAN_LOGNAME (LOG_NAME, RESULT) : NOVALUE =
;    63          0593  1
;    64          0594  1 !++
;    65          0595  1 !
;    66          0596  1 ! FUNCTIONAL DESCRIPTION:
;    67          0597  1 !
;    68          0598  1 !       This routine performs simple recursive logical name translation.
;    69          0599  1 !
;    70          0600  1 !
;    71          0601  1 ! CALLING SEQUENCE:
;    72          0602  1 !       TRAN_LOGNAME (ARG1, ARG2)
;    73          0603  1 !
;    74          0604  1 ! INPUT PARAMETERS:
;    75          0605  1 !       ARG1: descriptor of logical name to translate
;    76          0606  1 !
;    77          0607  1 ! IMPLICIT INPUTS:
;    78          0608  1 !       NONE
;    79          0609  1 !
;    80          0610  1 ! OUTPUT PARAMETERS:
;    81          0611  1 !       ARG2: descriptor of result string buffer
;    82          0612  1 !               (first word receives length of result)
;    83          0613  1 !
;    84          0614  1 ! IMPLICIT OUTPUTS:
;    85          0615  1 !       NONE
;    86          0616  1 !
;    87          0617  1 ! ROUTINE VALUE:
;    88          0618  1 !       NONE
;    89          0619  1 !
;    90          0620  1 ! SIDE EFFECTS:
;    91          0621  1 !       NONE
;    92          0622  1 !
;    93          0623  1 !--
;    94          0624  1
;    95          0625  2 BEGIN
;    96          0626  2
;    97          0627  2 MAP
;    98          0628  2         LOG_NAME        : REF VECTOR,   ! logical name descriptor
;    99          0629  2         RESULT          : REF VECTOR;   ! result string descriptor
;   100          0630  2
;   101          0631  2 LOCAL
;   102          0632  2         NAME_DESC       : VECTOR [2],   ! descriptor of current logical name string
;   103          0633  2         STATUS,                         ! system service status
;   104          0634  2         P;                              ! string search pointer
;   105          0635  2
;   106          0636  2 ! We iterate on logical name translation until the service returns SS$_NOTRAN.
;   107          0637  2 ! Perform device name extraction by using only the part of the logical name to
;   108          0638  2 ! the left of the colon (if any), also checking for node names.
;   109          0639  2 !
;   110          0640  2
;   111          0641  2 NAME_DESC[0] = .LOG_NAME[0];               ! get initial logical name
;   112          0642  2 NAME_DESC[1] = .RESULT[1];
;   113          0643  2 CH$COPY (.LOG_NAME[0], .LOG_NAME[1], 0, .RESULT[0], .RESULT[1]);
;   114          0644  2
;   115          0645  3 IF BEGIN
;   116          0646  3 DECR N FROM 10 TO 1 DO
;   117          0647  4     BEGIN
;   118          0648  4     P = CH$FIND_CH (.NAME_DESC[0], .NAME_DESC[1], ':');
```

```
;  119          0649  4      IF NOT CH$FAIL (.P)
;  120          0650  4      THEN
;  121          0651  5          BEGIN
;  122          0652  5          IF .P - .NAME_DESC[1] LSSU .NAME_DESC[0] - 1
;  123          0653  5          AND .(.P)<0,16> EQL '::'
;  124          0654  5          THEN ERR_EXIT (MOUN$_NONLOCAL);
;  125          0655  5          NAME_DESC[0] = .P - .NAME_DESC[1];
;  126          0656  4          END;
;  127          0657  4
;  128          0658  4      IF CH$RCHAR (.NAME_DESC[1]) EQL '_'
;  129          0659  4      THEN EXITLOOP 0;
;  130          0660  4
;  131        P 0661  4      STATUS = $TRNLOG (LOGNAM = NAME_DESC[0],
;  132        P 0662  4                        RSLLEN = NAME_DESC[0],
;  133          0663  4                        RSLBUF = RESULT[0]);
;  134          0664  4      IF .STATUS EQL SS$_NOTRAN THEN EXITLOOP 0;
;  135          0665  4      IF NOT .STATUS THEN ERR_EXIT (.STATUS);
;  136          0666  4      END
;  137          0667  3 END
;  138          0668  2 THEN ERR_EXIT (MOUN$_NOTRAN);
;  139          0669  2
;  140          0670  2 RESULT[0] = .NAME_DESC[0];
;  141          0671  2
;  142          0672  1 END;                                    . end of routine TRAN_LOGNAME

                                                        .TITLE  TRNLOG
                                                        .IDENT  \V04-000\

                                                        .EXTRN  SYS$TRNLOG

                                                        .PSECT  $CODE$,NOWRT,2

                                    00FC 00000          .ENTRY  TRAN_LOGNAME, Save R2,R3,R4,R5,R6,R7    ; 0592
                          57 00000000G  00   9E 00002   MOVAB   LIB$STOP, R7                           ;
                          5E            04   C2 00009   SUBL2   #4, SP                                ;
                          50      04    AC   D0 0000C   MOVL    LOG_NAME, R0                            ; 0641
                                        60   DD 00010   PUSHL   (R0)                                   ;
                          56      08    AC   D0 00012   MOVL    RESULT, R6                              ; 0642
                    04    AE      04    A6   D0 00016   MOVL    4(R6), NAME_DESC+4                      ;
66                00    04      B0          2C 0001B   MOVC5   (R0), @4(R0), #0, (R6), @4(R6)          ; 0643
                                  04    B6      00021
                          53            0A   D0 00023   MOVL    #10, N                                  ; 0646
              04    BE            6E    3A   3A 00026 1$:  LOCC    #58, NAME_DESC, @NAME_DESC+4            ; 0648
                                        02   12 0002B   BNEQ    2$                                      ;
                                        51   D4 0002D   CLRL    R1                                      ;
                          54            51   D0 0002F 2$:  MOVL    R1, P                                   ;
                                        21   13 00032   BEQL    4$                                      ; 0649
                    52          54      04    AE   C3 00034   SUBL3   NAME_DESC+4, P, R2                  ; 0652
                    50          6E            01   C3 00039   SUBL3   #1, NAME_DESC, R0                   ;
                          50            52   D1 0003D   CMPL    R2, R0                                  ;
                                        10   1E 00040   BGEQU   3$                                      ;
                  3A3A    8F                64   B1 00042   CMPW    (P), #14906                             ; 0653
                                        09   12 00047   BNEQ    3$                                      ;
                      0072803C    8F    DD 00049   PUSHL   #7503932                                    ; 0654
                          67            01   FB 0004F   CALLS   #1, LIB$STOP                            ;
                          6E            52   D0 00052 3$:  MOVL    R2, NAME_DESC                           ; 0655
```

```
                    5F   8F        04     BE   91 00055 4$:      CMPB    @NAME_DESC+4, #95                       ; 0658
                                         33   13 0005A           BEQL    6$                                      ;
                                         7E   7C 0005C           CLRQ    -(SP)                                   ; 0663
                                         7E   D4 0005E           CLRL    -(SP)                                   ;
                                         56   DD 00060           PUSHL   R6                                      ;
                                    10   AE   9F 00062           PUSHAB  NAME_DESC                               ;
                                    14   AE   9F 00065           PUSHAB  NAME_DESC                               ;
                00000000G   00           06   FB 00068           CALLS   #6, SYS$TRNLOG                          ;
                            55           50   D0 0006F           MOVL    R0, STATUS                              ;
                00000629    8F           55   D1 00072           CMPL    STATUS, #1577                           ; 0664
                                         14   13 00079           BEQL    6$                                      ;
                            05           55   E8 0007B           BLBS    STATUS, 5$                              ; 0665
                                         55   DD 0007E           PUSHL   STATUS                                  ;
                            67           01   FB 00080           CALLS   #1, LIB$STOP                            ;
                            A0           53   F5 00083 5$:       SOBGTR  N, 1$                                   ; 0646
                                 00728044 8F   DD 00086           PUSHL   #7503940                                ; 0668
                            67           01   FB 0008C           CALLS   #1, LIB$STOP                            ;
                            66           6E   D0 0008F 6$:       MOVL    NAME_DESC, (R6)                         ; 0670
                                              04 00092           RET                                             ; 0672

; Routine Size:  147 bytes,    Routine Base:  $CODE$ + 0000

;  143          0673  1
;  144          0674  1 END
;  145          0675  0 ELUDOM


                                                                .EXTRN  LIB$STOP

;                                  PSECT SUMMARY

;      Name                        Bytes                       Attributes

;  $CODE$                           147  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]LIB.L32;1           18619         14         0      1000          00:02.0


;                                  COMMAND QUALIFIERS

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

```
; Size:          147 code + 0 data bytes
; Run Time:        00:11.5
; Elapsed Time:    00:33.1
; Lines/CPU Min:     3521
; Lexemes/CPU-Min: 36568
; Memory Used:  104 pages
; Compilation Complete
```

VMOUNT
LIS

MPCLRPFM
LIS

MPAST
LIS

MP

MP
MAP

MP
MDL

TRNLOG
LIS

MPCMOD
LIS

MPMACROS
MAR