```
IIIIIIIII     NNN         NNN     IIIIIIIII     TTTTTTTTTTTTTTT
IIIIIIIII     NNN         NNN     IIIIIIIII     TTTTTTTTTTTTTTT
IIIIIIIII     NNN         NNN     IIIIIIIII     TTTTTTTTTTTTTTT
   III        NNN         NNN        III              TTT
   III        NNN         NNN        III              TTT
   III        NNN         NNN        III              TTT
   III        NNNNNN      NNN        III              TTT
   III        NNNNNN      NNN        III              TTT
   III        NNNNNN      NNN        III              TTT
   III        NNN   NNN   NNN        III              TTT
   III        NNN   NNN   NNN        III              TTT
   III        NNN   NNN   NNN        III              TTT
   III        NNN      NNNNNN        III              TTT
   III        NNN      NNNNNN        III              TTT
   III        NNN      NNNNNN        III              TTT
   III        NNN         NNN        III              TTT
   III        NNN         NNN        III              TTT
   III        NNN         NNN        III              TTT
IIIIIIIII     NNN         NNN     IIIIIIIII           TTT
IIIIIIIII     NNN         NNN     IIIIIIIII           TTT
IIIIIIIII     NNN         NNN     IIIIIIIII           TTT
```

```
                                                                      N  2
**FILE**ID**RDHOME
```

```
RRRRRRRR    DDDDDDDD    HH      HH    OOOOOO    MM      MM  EEEEEEEEEE
RRRRRRRR    DDDDDDDD    HH      HH    OOOOOO    MM      MM  EEEEEEEEEE
RR      RR  DD      DD  HH      HH  OO      OO  MMMM  MMMM  EE
RR      RR  DD      DD  HH      HH  OO      OO  MMMM  MMMM  EE
RR      RR  DD      DD  HH      HH  OO      OO  MM  MM  MM  EE
RR      RR  DD      DD  HH      HH  OO      OO  MM  MM  MM  EE
RRRRRRRR    DD      DD  HHHHHHHHHH  OO      OO  MM      MM  EEEEEEEE
RRRRRRRR    DD      DD  HHHHHHHHHH  OO      OO  MM      MM  EEEEEEEE
RR  RR      DD      DD  HH      HH  OO      OO  MM      MM  EE
RR  RR      DD      DD  HH      HH  OO      OO  MM      MM  EE
RR    RR    DD      DD  HH      HH  OO      OO  MM      MM  EE          ....
RR    RR    DD      DD  HH      HH  OO      OO  MM      MM  EE          ....
RR      RR  DDDDDDDD    HH      HH    OOOOOO    MM      MM  EEEEEEEEEE  ....
RR      RR  DDDDDDDD    HH      HH    OOOOOO    MM      MM  EEEEEEEEEE  ....


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 RDHOME (
0                 LANGUAGE (BLISS32),
0                 IDENT = 'V04-000'
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 ! FACILITY:  MOUNT Utility Structure Levels 1 & 2
1 !
1 ! ABSTRACT:
1 !        This routine reads the home block, if any, of the volume being mounted.
1 !
1 ! ENVIRONMENT:
1 !
1 !       STARLET operating system, including privileged system services
1 !       and internal exec routines.
1 !
1 !--
1
1 !
1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  13-Oct-1977  21:29
1 !
1 ! MODIFIED BY:
1 !
1 !
1 !       V02-004 STJ0002         Steven T. Jeffreys,     29-Aug-1980
1 !               Liberal re-write to facilitate operator assisted mount.
1 !               As a result, the code indentation is screwed up.
1 !
1 !       V02-003 ACG0167         Andrew C. Goldstein,    18-Apr-1980  13:39
1 !               Previous revision history moved to MOUNT.REV
1 !**
```

```
;    58          0058  1
;    59          0059  1
;    60          0060  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    61          0061  1 REQUIRE 'SRC$:INIDEF.B32';
;    62          0352  1
;    63          0353  1
;    64          0354  1 FORWARD ROUTINE
;    65          0355  1         READ_HOMEBLOCK,                  ! read the home block
;    66          0356  1         SET_VALID,                       ! set volume valid bit
;    67          0357  1         CLEAR_VALID;                     ! clear volume valid bit
```

```
EX

Mo
--
LI
IN
IN
IN
ST
ST
CL
DC
RP
PA
CH
CO
DI
UP
HP
VA
CV
JB
SY
SY
RM
SY
```

```
1 GLOBAL ROUTINE READ_HOMEBLOCK (VOLUME_LABEL, FULL_SEARCH) =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine reads the home block, if any, of the volume being mounted.
1 !       Various statuses are returned to be interpreted in the context of the
1 !       operation; errors that are clear losers are signalled.
1 !
1 !
1 ! CALLING SEQUENCE:
1 !       READ_HOMEBLOCK (ARG1, ARG2)
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1: address of volume label string descriptor
1 !       ARG2: 1 if search whole disk for home block
1 !             0 if limited search
1 !
1 ! IMPLICIT INPUTS:
1 !       CHANNEL: channel number assigned to device being mounted
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       HOME_BLOCK: buffer contains home block if found
1 !       HOMEBLOCK_LBN: LBN of home block read
1 !
1 ! ROUTINE VALUE:
1 !       1 if valid and correct home block found
1 !       SS$_NOHOMEBLK if home block not found
1 !       SS$_INCVOLLABEL if home block found but wrong volume name
1 !       SS$_DEVOFFLINE
1 !       SS$_MEDOFL
1 !       SS$_FILESTRUCT
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2
2 MAP
2       VOLUME_LABEL    : REF VECTOR;   ! volume label string descriptor
2
2 LOCAL
2       STATUS,                         ! system service status
2       IO_STATUS       : VECTOR[2],    ! I/O status block
2       DELTA,                          ! home block search delta
2       BLOCKFACT,                      ! device blocking factor
2       LBN;                            ! current LBN being tried
2
2 EXTERNAL
2       CHANNEL,                        ! channel number for I/O
2       DEVICE_CHAR     : BBLOCK,       ! disk device characteristics
```

```
:   126          0415  2          DEVCHAR_DESC     : VECTOR,          ! device characteristics descriptor
:   127          0416  2          HOME_BLOCK       : BBLOCK,          ! home block buffer
:   128          0417  2          HOMEBLOCK_LBN,                      ! LBN of home block read
:   129          0418  2          CTL$GL_PHD       : REF BBLOCK ADDRESSING_MODE (ABSOLUTE);
:   130          0419  2                                              ! pointer to process header
:   131          0420  2
:   132          0421  2  EXTERNAL ROUTINE
:   133          0422  2          READ_BLOCK,                         ! read a disk block
:   134          0423  2          CHECK_HOMEBLK1,                     ! verify structure level 1 home block
:   135          0424  2          CHECK_HOMEBLK2;                     ! verify structure level 2 home block
:   136          0425  2
:   137          0426  2
:   138          0427  2
:   139          0428  2  ! Issue a pack acknowlege so we can talk to the disk.
:   140          0429  2  ! Inhibit error logging to avoid saturating the error log
:   141          0430  2  ! with media/device offline errors.  These errors cause
:   142          0431  2  ! us to return and try an operator assisted mount.
:   143          0432  2  !
:   144          0433  2
:   145        P 0434  2  STATUS = $QIOW (CHAN = .CHANNEL,
:   146        P 0435  2                  FUNC = (IO$_PACKACK OR IO$M_INHERLOG),
:   147          0436  2                  IOSB = IO_STATUS[0]);
:   148          0437  2  IF .STATUS THEN STATUS = .(IO_STATUS[0])<0,16>;
:   149          0438  2  IF NOT .STATUS
:   150          0439  2  THEN
:   151          0440  2      !
:   152          0441  2      ! Allow ILLIOFUNC errors, as devices that have
:   153          0442  2      ! no PACKACK function will return this status.
:   154          0443  2      !
:   155          0444  2      IF .STATUS NEQ SS$_ILLIOFUNC
:   156          0445  2      THEN
:   157          0446  2          ERR_EXIT (.STATUS);
:   158          0447  2
:   159          0448  2  ! Set the volume valid bit to indicate presence of a volume.
:   160          0449  2  !
:   161          0450  2
:   162          0451  2  KERNEL_CALL (SET_VALID);
:   163          0452  2
:   164          0453  2  ! Re-read the device characteristics. (For some devices, packack causes the
:   165          0454  2  ! correct characteristics to be determined and recorded in the I/O data base.)
:   166          0455  2  !
:   167          0456  2
:   168          0457  2  $GETCHN (CHAN = .CHANNEL, PRIBUF = DEVCHAR_DESC);
:   169          0458  2
:   170          0459  2  ! Compute the home block search delta from the volume geometry in the
:   171          0460  2  ! device table. This is done according to the following rules, where volume
:   172          0461  2  ! geometry is expressed in the order sectors, tracks, cylinders:
:   173          0462  2  !
:   174          0463  2  !       n x 1 x 1:      1
:   175          0464  2  !       1 x n x 1:      1
:   176          0465  2  !       1 x 1 x n:      1
:   177          0466  2  !
:   178          0467  2  !       n x m x 1:      n+1
:   179          0468  2  !       n x 1 x m:      n+1
:   180          0469  2  !       1 x n x m:      n+1
:   181          0470  2  !
:   182          0471  2  !       s x t x c:      (t+1)*s+1
```

```
;   183        0472  2 !
;   184        0473  2
;   185        0474  3 BLOCKFACT = (.DEVICE_CHAR[DIB$B_SECTORS]
;   186        0475  3             * .DEVICE_CHAR[DIB$B_TRACKS]
;   187        0476  3             * .DEVICE_CHAR[DIB$W_CYLINDERS])
;   188        0477  2             / .DEVICE_CHAR[DIB$L_MAXBLOCK];
;   189        0478  2
;   190        0479  2 DELTA = 1;
;   191        0480  2 IF   .DEVICE_CHAR[DIB$W_CYLINDERS] GTR 1
;   192        0481  2 AND .DEVICE_CHAR[DIB$B_TRACKS] GTR 1
;   193        0482  2 THEN DELTA = .DELTA + .DEVICE_CHAR[DIB$B_TRACKS];
;   194        0483  2
;   195        0484  2 IF   .DEVICE_CHAR[DIB$B_SECTORS] GTR 1
;   196        0485  3 AND (.DEVICE_CHAR[DIB$Q_CYLINDERS] GTR 1
;   197        0486  3 OR .DEVICE_CHAR[DIB$B_TRACKS] GTR 1)
;   198        0487  2 THEN DELTA = (.DELTA * .DEVICE_CHAR[DIB$B_SECTORS] + .BLO:KFACT) / .BLOCKFACT;
;   199        0488  2
;   200        0489  2 IF .DELTA EQL 0
;   201        0490  2 OR .DELTA GTRU .DEVICE_CHAP[DIB$L_MAXBLOCK] / 10
;   202        0491  2 THEN DELTA = 1;
;   203        0492  2
;   204        0493  2 ! Search for the home block. If the device is being mounted foreign, we
;   205        0494  2 ! limit the home block search to 10 hits to save time. If the device is
;   206        0495  2 ! being mounted Files-11 then we try all the way just in case the home block
;   207        0496  2 ! is in a weird location. Note the potential protection hole: Disks with the
;   208        0497  2 ! home block far into the disk are not protected from being mounted foreign
;   209        0498  2 ! by non-privileged users. C'est la vie.
;   210        0499  2 !
;   211        0500  2
;   212        0501  2 LBN = 1;
;   213        0502  2
;   214        0503  3 IF (
;   215        0504  3     DECRU J FROM (IF .FULL_SEARCH THEN -1 ELSE 10) TO 1 DO
;   216        0505  4     BEGIN
;   217        0506  4         STATUS = READ_BLOCK (.LBN, HOME_BLOCK);
;   218        0507  4         IF .STATUS
;   219        0508  4         THEN
;   220        0509  5             BEGIN
;   221        0510  5             IF .HOME_BLOCK[HM2$B_STRUCLEV] EQL 1
;   222        0511  5             THEN STATUS = CHECK_ROMEBLK1 (HOME_BLOCK, .LBN, .VOLUME_LABEL)
;   223        0512  5             ELSE IF .HOME_BLOCK[HM2$B_STRUCLEV] EQL 2
;   224        0513  5             THEN STATUS = CHECK_HOMEB[K2 (HOME_BLOCK, .LBN, .VOLUME_LABEL)
;   225        0514  5             ELSE STATUS = 0;
;   226        0515  5             IF .STATUS
;   227        0516  5             OR .STATUS EQL SS$_INCVOLLABEL
;   228        0517  5             THEN EXITLOOP 0;
;   229        0518  5             END
;   230        0519  4         ELSE
;   231        0520  5             BEGIN
;   232        0521  5             IF .STATUS EQL SS$_ILLBLKNUM
;   233        0522  5             THEN EXITLOOP -1;
;   234        0523  5             IF   .STATUS NEQ SS$_PARITY
;   235        0524  5             AND .STATUS NEQ SS$_FORMAT
;   236        0525  5             AND .STATUS NEQ SS$_DATACHECK
;   237        0526  5             THEN ERR_EXIT (.STATUS);
;   238        0527  4             END;
;   239        0528  4
;   240          0529  4           IF NOT .FULL_SEARCH
;   241          0530  4           AND .BBLOCK [CTL$GL_PHD[PHD$Q_PRIVMSK], PRV$V_VOLPRO]
;   242          0531  4           THEN RETURN (SS$_NOHOMEBLK);
;   243          0532  4
;   244          0533  4           LBN = .LBN + .DELTA;
;   245          0534  4           END
;   246          0535  3        )
;   247          0536  2  THEN
;   248          0537  3       BEGIN
;   249          0538  3       READ_BLOCK (1, HOME_BLOCK);
;   250          0539  3       RETURN (SS$_NOHOMEBLK);
;   251          0540  2       END;
;   252          0541  2
;   253          0542  2  HOMEBLOCK_LBN = .LBN;
;   254          0543  2
;   255          0544  2  RETURN .STATUS;
;   256          0545  2
;   257          0546  1  END;                                    ! end of routine READ_HOMEBLK


                                                                   .TITLE  RDHOME
                                                                   .IDENT  \V04-000\

                                                                   .EXTRN  CHANNEL, DEVICE_CHAR
                                                                   .EXTRN  DEVCHAR_DESC, HOME_BLOCK
                                                                   .EXTRN  HOMEBLOCK_LBN, CTL$GL_PHD
                                                                   .EXTRN  READ_BLOCK, CHECK_HOMEBLK1
                                                                   .EXTRN  CHECK_HOMEBLK2, SYS$QIOW
                                                                   .EXTRN  SYS$CMKRNL, SYS$GETCHN

                                                                   .PSECT  $CODE$,NOWRT,2

                                         03FC 00000                .ENTRY  READ_HOMEBLOCK, Save R2,R3,R4,R5,R6,R7,R8,- ; 0358
                                                                           R9                                          ;
                       59 00000000G   00   9E 00002                MOVAB   LIB$STOP, R9                                ;
                       58     0000G   CF   9E 00009                MOVAB   HOME_BLOCK, R8                              ;
                       57     0000G   CF   9E 0000E                MOVAB   DEVICE_CHAR+10, R7                          ;
                       5E             08   C2 00013                SUBL2   #8, SP                                      ;
                                      7E   7C 00016                CLRQ    -(SP)                                       ; 0436
                                      7E   7C 00018                CLRQ    -(SP)                                       ;
                                      7E   7C 0001A                CLRQ    -(SP)                                       ;
                                      7E   7C 0001C                CLRQ    -(SP)                                       ;
                                20    AE   9F 0001E                PUSHAB  IO_STATUS                                   ;
                       7E     0808    8F   3C 00021                MOVZWL  #2056, -(SP)                                ;
                              0000G   CF   DD 00026                PUSHL   CHANNEL                                     ;
                                      7E   D4 0002A                CLRL    -(SP)                                       ;
              00000000G  00           0C   FB 0002C                CALLS   #12, SYS$QIOW                               ;
                         53           50   D0 00033                MOVL    R0, STATUS                                  ;
                         06           53   E9 00036                BLBC    STATUS, 1$                                  ; 0437
                         53           6E   3C 00039                MOVZWL  IO_STATUS, STATUS                           ;
                         0E           53   E8 0003C                BLBS    STATUS, 2$                                  ; 0438
              000000F4   8F           53   D1 0003F  1$:           CMPL    STATUS, #244                                ; 0444
                                      05   13 00046                BEQL    2$                                          ;
                                      53   DD 00048                PUSHL   STATUS                                      ; 0446
                         69           01   FB 0004A                CALLS   #1, LIB$STOP                                ;
                                      7E   D4 0004D  2$:           CLRL    -(SP)                                       ; 0451
                                      5E   DD 0004F                PUSHL   SP                                          ;
```

```
                           0000V  CF   9F 00051        PUSHAB  SET_VALID                               ;
      00000000G  9F               03   FB 00055        CALLS   #3, @#SYS$CMKRNL                        ;
                                  7E   7C 0005C        CLRQ    -(SP)                                   ; 0457
                           0000G  CF   9F 0005E        PUSHAB  DEVCHAR_DESC                            ;
                                  7E   D4 00062        CLRL    -(SP)                                   ;
                           0000G  CF   DD 00064        PUSHL   CHANNEL                                 ;
      00000000G  00               05   FB 00068        CALLS   #5, SYS$GETCHN                          ;
                 50          FE   A7   9A 0006F        MOVZBL  DEVICE_CHAR+8, R0                       ; 0474
                 51          FF   A7   9A 00073        MOVZBL  DEVICE_CHAR+9, R1                       ; 0475
52               50               51   C5 00077        MULL3   R1, R0, R2                              ;
                 54               67   3C 0007B        MOVZWL  DEVICE_CHAR+10, R4                      ; 0476
                 52               54   C4 0007E        MULL2   R4, R2                                  ;
54               52          66   A7   C7 00081        DIVL3   DEVICE_CHAR+112, R2, BLOCKFACT          ; 0477
                 52               01   D0 00086        MOVL    #1, DELTA                               ; 0479
                                  55   D4 00089        CLRL    R5                                      ; 0480
                 01               67   B1 0008B        CMPW    DEVICE_CHAR+10, #1                      ;
                                  0A   1B 0008E        BLEQU   3$                                      ;
                                  55   D6 00090        INCL    R5                                      ;
                 01               51   91 00092        CMPB    R1, #1                                  ; 0481
                                  03   1B 00095        BLEQU   3$                                      ;
                 52               51   C0 00097        ADDL2   R1, DELTA                               ; 0482
                 01               50   91 0009A 3$:    CMPB    R0, #1                                  ; 0484
                                  12   1B 0009D        BLEQU   5$                                      ;
                 05               55   E8 0009F        BLBS    R5, 4$                                  ; 0485
                 01               51   91 000A2        CMPB    R1, #1                                  ; 0486
                                  0A   1B 000A5        BLEQU   5$                                      ;
                 50               52   C4 000A7 4$:    MULL2   DELTA, R0                               ; 0487
                 50               54   C0 000AA        ADDL2   BLOCKFACT, R0                           ;
52               50               54   C7 000AD        DIVL3   BLOCKFACT, R0, DELTA                    ;
                                  52   D5 000B1 5$:    TSTL    DELTA                                   ; 0489
                                  0A   13 000B3        BEQL    6$                                      ;
50          66   A7               0A   C7 000B5        DIVL3   #10, DEVICE_CHAR+112, R0                ; 0490
                 50               52   D1 000BA        CMPL    DELTA, R0                               ;
                                  03   1B 000BD        BLEQU   7$                                      ;
                 52               01   D0 000BF 6$:    MOVL    #1, DELTA                               ; 0491
                 54               01   D0 000C2 7$:    MOVL    #1, LBN                                 ; 0501
                 05          08   AC   E9 000C5        BLBC    FULL_SEARCH, 8$                         ; 0504
                 55               01   CE 000C9        MNEGL   #1, R5                                  ;
                                  03   11 000CC        BRB     9$                                      ;
                 55               0A   D0 000CE 8$:    MOVL    #10, R5                                 ;
                 56          08   AC   D2 000D1 9$:    MCOML   FULL_SEARCH, R6                         ; 0529
                                0086   31 000D5        BRW     18$                                     ; 0533
                           0110   8F   BB 000D8 10$:   PUSHR   #^M<R4,R8>                              ; 0506
          0000G  CF               02   FB 000DC        CALLS   #2, READ_BLOCK                          ;
                 53               50   D0 000E1        MOVL    R0, STATUS                              ;
                 3B               53   E9 000E4        BLBC    STATUS, 15$                             ; 0507
                 01          0D   A8   91 000E7        CMPB    HOME_BLOCK+13, #1                       ; 0510
                                  0E   12 000EB        BNEQ    11$                                     ;
                             04   AC   DD 000ED        PUSHL   VOLUME_LABEL                            ; 0511
                                  54   DD 000F0        PUSHL   LBN                                     ;
                                  58   DD 000F2        PUSHL   R8                                      ;
          0000G  CF               03   FB 000F4        CALLS   #3, CHECK_HOMEBLK1                      ;
                                  12   11 000F9        BRB     12$                                     ;
                 02          0D   A8   91 000FB 11$:   CMPB    HOME_BLOCK+13, #2                       ; 0512
                                  11   12 000FF        BNEQ    13$                                     ;
                             04   AC   DD 00101        PUSHL   VOLUME_LABEL                            ; 0513
                                  54   DD 00104        PUSHL   LBN                                     ;
```

```
                                                    58  DD 00106          PUSHL   R8                                        ;
                          0000G   CF                03  FB 00108          CALLS   #3, CHECK_HOMEBLK2                        ;
                                  53                50  D0 0010D  12$:    MOVL    R0, STATUS                                ;
                                                    02  11 00110          BRB     14$                                       ;
                                                    53  D4 00112  13$:    CLRL    STATUS                                    ; 0514
                                  5D                53  E8 00114  14$:    BLBS    STATUS, 21$                               ; 0515
                      0000010C    8F                53  D1 00117          CMPL    STATUS, #268                              ; 0516
                                                    28  12 0011E          BNEQ    16$                                       ;
                                                    52  11 00120          BRB     21$                                       ; 0517
                      000000DC    8F                53  D1 00122  15$:    CMPL    STATUS, #220                              ; 0521
                                                    3A  13 00129          BEQL    19$                                       ;
                      000001F4    8F                53  D1 0012B          CMPL    STATUS, #500                              ; 0523
                                                    17  13 00132          BEQL    16$                                       ;
                      000000BC    8F                53  D1 00134          CMPL    STATUS, #188                              ; 0524
                                                    0E  13 0013B          BEQL    16$                                       ;
                      0000005C    8F                53  D1 0013D          CMPL    STATUS, #92                               ; 0525
                                                    05  13 00144          BEQL    16$                                       ;
                                                    53  DD 00146          PUSHL   STATUS                                    ; 0526
                                  69                01  FB 00148          CALLS   #1, LIB$STOP                              ;
                                  0B                56  E9 0014B  16$:    BLBC    R6, 17$                                   ; 0529
                                  50  00000000G     9F  D0 0014E          MOVL    @#CTL$GL_PHD, R0                          ; 0530
                 15               60                15  E0 00155          BBS     #21, (R0), 20$                            ;
                                  54                52  C0 00159  17$:    ADDL2   DELTA, LBN                                ; 0533
                                                    55  D7 0015C          DECL    J                                         ; 0504
                                                    55  D5 0015E  18$:    TSTL    J                                         ;
                                                    03  13 00160          BEQL    19$                                       ;
                                                  FF73  31 00162          BRW     10$                                       ;
                                                    58  DD 00165  19$:    PUSHL   R8                                        ; 0538
                                                    01  DD 00167          PUSHL   #1                                        ;
                          0000G   CF                02  FB 00169          CALLS   #2, READ_BLOCK                            ;
                                  50       08E0     8F  3C 0016E  20$:    MOVZWL  #2272, R0                                 ; 0539
                                                        04 00173          RET                                               ;
                          0000G   CF                54  D0 00174  21$:    MOVL    LBN, HOMEBLOCK_LBN                        ; 0542
                                  50                53  D0 00179          MOVL    STATUS, R0                                ; 0544
                                                        04 0017C          RET                                               ; 0546

; Routine Size:  381 bytes,    Routine Base:  $CODE$ + 0000
```

```
1 GLOBAL ROUTINE SET_VALID =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine sets the volume valid bit in the UCB, causing
1 !       PACKACK to be reissued in case of a power failure or volume
1 !       status transition.
1 !
1 !
1 ! CALLING SEQUENCE:
1 !       SET_VALID ()
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !       CHANNEL: channel number assigned to device
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       1
1 !
1 ! SIDE EFFECTS:
1 !       valid bit set in UCB
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2       UCB             : REF BBLOCK;   ! pointer to UCB
2
2 EXTERNAL
2       CHANNEL;                        ! channel assigned to device
2
2 EXTERNAL ROUTINE
2       GET_CHANNELUCB;                 ! get UCB of channel
2
2
2 ! Get the UCB address from the channel and set the bit.
2 !
2
2 UCB = GET_CHANNELUCB (.CHANNEL);
2 UCB[UCB$V_VALID] = 1;
2
2 RETURN 1;
2
1 END;                                  ! end of routine SET_VALID
```

```
                                                          .EXTRN  GET_CHANNELUCB

                                         0000 00000       .ENTRY  SET_VALID, Save nothing                 ; 0547
                            0000G  CF      DD 00002       PUSHL   CHANNEL                                 ; 0596
                0000G  CF          01      FB 00006       CALLS   #1, GET_CHANNELUCB                      ;
                  65   A0          08      88 0000B       BISB2   #8, 101(UCB)                            ; 0597
                       50          01      D0 0000F       MOVL    #1, R0                                  ; 0599
                                           04 00012       RET                                             ; 0601

; Routine Size:  19 bytes,    Routine Base:  $CODE$ + 017D
```

```
_$

Sy
--
DC
DC
DC
DC
DC
DO
DO
DS
DS
EO
EO
EX
IN
IN
IN
IN
IN
IN
IN
IN
IN
IN
IN
IN
IN
JO
JO
LG
LG
LG
LG
LI
LI
LI
LI
LI
LI
LI
LI
LI
LI

LI
LI
MA
MA
NO
NO
OW
OW
PA
PA
```

```
1 GLOBAL ROUTINE CLEAR_VALID =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine clears the volume valid bit in the UCB.
1 !
1 !
1 ! CALLING SEQUENCE:
1 !       CLEAR_VALID ()
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !       CHANNEL: channel number assigned to device
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       1
1 !
1 ! SIDE EFFECTS:
1 !       valid bit clear in UCB
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2       UCB             : REF BBLOCK;   ! pointer to UCB
2
2 EXTERNAL
2       CHANNEL;                        ! channel assigned to device
2
2 EXTERNAL ROUTINE
2       GET_CHANNELUCB;                 ! get UCB of channel
2
2
2 ! Get the UCB address from the channel and clear t ie bit.
2 !
2
2 UCB = GET_CHANNELUCB (.CHANNEL);
2 UCB[UCB$V_VALID] = 0;
2
2 RETURN 1;
2
1 END;                                  ! end of routine CLEAR_VALID
```

```
                                           0000 00000         .ENTRY  CLEAR_VALID, Save nothing                   ; 0602
                                  0000G  CF   DD 00002         PUSHL   CHANNEL                                     ; 0649
                      0000G  CF          01   FB 00006         CALLS   #1, GET_CHANNELUCB                          ;
                         65  A0          08   8A 0000B         BICB2   #8, 101(UCB)                                ; 0650
                             50          01   D0 0000F         MOVL    #1, R0                                      ; 0652
                                              04 00012         RET                                                 ; 0654

; Routine Size:  19 bytes,    Routine Base:  $CODE$ + 0190


;  368          0655  1
;  369          0656  1 END
;  370          0657  0 ELUDOM



                                                              .EXTRN  LIB$STOP

;                               PSECT SUMMARY
;
;     Name                     Bytes                     Attributes
;
;  $CODE$                          419  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        24         0        1000           00:01.9



;                                COMMAND QUALIFIERS

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

; Size:          419 code + 0 data bytes
; Run Time:         00:13.1
; Elapsed Time:     00:39.9
; Lines/CPU Min:     3004
; Lexemes/CPU-Min: 25417
; Memory Used:  147 pages
; Compilation Complete
```

0188 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY