H 5

```
FFFFFFFFFFFFFFF         111              111        XXX         XXX
FFFFFFFFFFFFFFF         111              111        XXX         XXX
FFFFFFFFFFFFFFF         111              111        XXX         XXX
FFF                  111111           111111        XXX         XXX
FFF                  111111           111111        XXX         XXX
FFF                  111111           111111        XXX         XXX
FFF                     111              111           XXX   XXX
FFF                     111              111           XXX   XXX
FFF                     111              111           XXX   XXX
FFFFFFFFFFFF            111              111              XXX
FFFFFFFFFFFF            111              111              XXX
FFFFFFFFFFFF            111              111              XXX
FFF                     111              111           XXX   XXX
FFF                     111              111           XXX   XXX
FFF                     111              111           XXX   XXX
FFF                     111              111        XXX         XXX
FFF                     111              111        XXX         XXX
FFF                     111              111        XXX         XXX
FFF                  111111111        111111111     XXX         XXX
FFF                  111111111        111111111     XXX         XXX
FFF                  111111111        111111111     XXX         XXX
```

```
_$25

Symb
----
IOC$
IO_C
IO_C
IO_D
IO_F
IO_S
KILL

KILL
KILL
LB_E
LB_D
LB_F
LB_M
LB_L
LOCA
LOCA
LOCK

LOCK
LOCK
LOCK
LOC_
LOC_
L_CO
L_CO
L_DA
L_DA
MAIN
MAKE
MAKE
MAKE
MAKE
MAKE
MAKE

MAKE
MAKE
MAP_
MAP_

MAP
MARK
MARK
MARK
MARK
MARK
```

```
                                                                  L 16
**FILE**ID**ENTER
```

```
EEEEEEEEEE  NN      NN  TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR
EEEEEEEEEE  NN      NN  TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR
EE          NN      NN      TT      EE          RR      RR
EE          NN      NN      TT      EE          RR      RR
EE          NNNN    NN      TT      EE          RR      RR
EE          NNNN    NN      TT      EE          RR      RR
EEEEEEEE    NN  NN  NN      TT      EEEEEEEE    RRRRRRRR
EEEEEEEE    NN  NN  NN      TT      EEEEEEEE    RRRRRRRR
EE          NN    NNNN      TT      EE          RR  RR
EE          NN    NNNN      TT      EE          RR  RR
EE          NN      NN      TT      EE          RR    RR        ....
EE          NN      NN      TT      EE          RR    RR        ....
EEEEEEEEEE  NN      NN      TT      EEEEEEEEEE  RR      RR      ....
EEEEEEEEEE  NN      NN      TT      EEEEEEEEEE  RR      RR      ....


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

```
0 MODULE ENTER (
0                       LANGUAGE (BLISS32),
0                       IDENT = 'V04-001'
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:  F11ACP Structure Level 2
1 !
1 ! ABSTRACT:
1 !
1 !       This routine creates a new directory entry for the given file.
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:  11-Jan-1978  16:50
1 !
1 ! MODIFIED BY:
1 !
1 !       V04-001 CWH4001         CW Hobbs                  9-Aug-1984
1 !               Fix truncation error.
1 !
1 !       V03-006 CDS0003         Christian D. Saether      3-Aug-1984
1 !               Modify handling of directory index updating.
1 !
1 !       V03-005 LMP0252         L. Mark Pilant,           25-Jun-1984  10:03
```

```
;    58            0058  1 !                  Make sure that the test for removing a directory entry on an
;    59            0059  1 !                  an auto-purge uses the relative position of the entry, no the
;    60            0060  1 !                  address.  This corrects a problem with corrupted directories
;    61            0061  1 !                  for large directories with a version limit of 1.
;    62            0062  1 !
;    63            0063  1 !          V03-004 ACG0419           Andrew C. Goldstein,    19-Apr-1984  15:27
;    64            0064  1 !                  Add NOSAFE switch to MAKE_ENTRY to circumvent CSE bug
;    65            0065  1 !
;    66            0066  1 !          V03-003 ACG0408           Andrew C. Goldstein,    20-Mar-1984  17:39
;    67            0067  1 !                  Make APPLY_RVN and DEFAULT_RVN macros
;    68            0068  1 !
;    69            0069  1 !          V03-002 CDS0002           Christian D. Saether    18-Jan-1984
;    70            0070  1 !                  Modify interface to DEFAULT_RVN and APPLY_RVN.
;    71            0071  1 !
;    72            0072  1 !          V03-001 CDS0001           Christian D. Saether    19-Dec-1983
;    73            0073  1 !                  Use BIND_COMMON macro to reduce number of
;    74            0074  1 !                  external COMMON declarations.
;    75            0075  1 !
;    76            0076  1 !          V02-012 ACG0264           Andrew C. Goldstein,    12-Feb-1982  15:21
;    77            0077  1 !                  Allow negative version numbers (and treat like zero)
;    78            0078  1 !
;    79            0079  1 !          V02-011 ACG0259           Andrew C. Goldstein,    26-Jan-1982  19:17
;    80            0080  1 !                  Fix autopurge when version limit is 1
;    81            0081  1 !
;    82            0082  1 !          V02-010 ACG0238           Andrew C. Goldstein,    10-Dec-1981  14:22
;    83            0083  1 !                  Invalidate RMS directory cache when dir is superseded
;    84            0084  1 !
;    85            0085  1 !          V02-009 ACG0208           Andrew C. Goldstein,    28-Oct-1981  20:44
;    86            0086  1 !                  Add segmented directory record support
;    87            0087  1 !
;    88            0088  1 !          V02-008 ACG34341          Andrew C. Goldstein,    3-Mar-1981  17:06
;    89            0089  1 !                  Fix remove cleanup when first block has been squished
;    90            0090  1 !
;    91            0091  1 !          V02-007 ACG0167           Andrew C. Goldstein,    16-Apr-1980  19:26
;    92            0092  1 !                  Previous revision history moved to F11B.REV
;    93            0093  1 !**
;    94            0094  1
;    95            0095  1
;    96            0096  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    97            0097  1 REQUIRE 'SRC$:FCPDEF.B32';
;    98            1088  1
;    99            1089  1
;   100            1090  1 FORWARD ROUTINE
;   101            1091  1          ENTER           : L_NORM NOVALUE, ! main ENTER routine
;   102            1092  1          MAKE_ENTRY      : L_NORM NOVALUE, ! build new directory entry
;   103            1093  1          RESTORE_DIR     : L_NORM NOVALUE; ! restore directory context
```

```
1 GLOBAL ROUTINE ENTER (ABD, FIB, RESULT_LENGTH, RESULT) : L_NORM NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine enters the given file name in the specified directory.
1 !
1 ! CALLING SEQUENCE:
1 !       ENTER (ARG1, ARG2, ARG3, ARG4)
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1: address of buffer descriptor packet
1 !       ARG2: address of FIB of operation
1 !
1 ! IMPLICIT INPUTS:
1 !       NONE
1 !
1 ! OUTPUT PARAMETERS:
1 !       ARG3: address of longword to receive length of result string
1 !       ARG4: address of result string buffer
1 !
1 ! IMPLICIT OUTPUTS:
1 !       DIR_BUFFER: buffer address of current directory block
1 !       DIR_ENTRY: address of directory record
1 !       DIR_VERSION: address of directory version entry
1 !       DIR_END: end of directory data
1 !       PREV_VERSION: version number of superseded entry
1 !
1 ! ROUTINE VALUE:
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !       directory altered, result string and length written into buffer packet
1 !
1 !--
1
2 BEGIN
2
2 MAP
2       ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH],
2                                       ! descriptor list arg
2       FIB             : REF BBLOCK,   ! FIB argument
2       RESULT          : REF VECTOR [,BYTE]; ! result string arg
2
2 LOCAL
2       NAME_DESC       : BBLOCK [FND_LENGTH], ! file name descriptor block
2       NAME_BUFFER     : VECTOR [FILENAME_LENGTH, BYTE], ! buffer for file name
2       STATUS,                         ! result of directory search
2       P,                              ! string pointer
2       VBN,                            ! current VBN of directory file
2       NAME_LENGTH;                    ! length of file name, rounded even
2
2 BIND_COMMON;
2
2 DIR_CONTEXT_DEF;
2
```

```
2 EXTERNAL ROUTINE
2           PMS_START_SUB   : L_NORM,       ! start subfunction metering
2           PMS_END_SUB     : L_NORM,       ! end subfunction metering
2           DIR_ACCESS      : L_NORM,       ! access the directory
2           PARSE_NAME      : L_NORM,       ! parse file string
2           DIR_SCAN        : L_NORM,       ! search the directory
2           UPDATE_DIRSEQ   : L_NORM        ! update directory sequence count in UCB
2                           ADDRESSING_MODE (LONG_RELATIVE),
2           NEXT_DIR_REC    : L_NORM,       ! advance to next matching record
2           RETURN_DIR      : L_NORM,       ! return data to buffer packet
2           MARK_DIRTY      : L_NORM;       ! mark buffer for rewrite
2
2
2 ! Start metering for this subfunction.
2 !
2
2 PMS_START_SUB (PMS_ENTER);
2
2 ! The file ID to be entered must be non-zero.
2 !
2
2 IF .FIB[FIB$W_FID_NUM] EQL 0
2 THEN ERR_EXIT (SS$_BADPARAM);
2
2 ! Find the name string in the buffer packet. Parse the string into the
2 ! name descriptor block. Mask out the wild card bits, since they are
2 ! not permitted. Check the version number; it must be positive.
2 !
2
2 PARSE_NAME (NAME_DESC, NAME_BUFFER, .ABD[ABD$C_NAME, ABD$W_COUNT],
2             .ABD[ABD$C_NAME, ABD$W_TEXT] + ABD[ABD$C_NAME, ABD$W_TEXT] + 1,
2             .FIB[FIB$W_NMCTL] AND FIB$M_NEWVER);
2 IF .NAME_DESC[FND_WILD]
2 THEN ERR_EXIT (SS$_BADFILENAME);
2 IF .NAME_DESC[FND_VERSION] LSS 0
2 THEN NAME_DESC[FND_VERSION] = 0;
2 NAME_LENGTH = .NAME_DESC[FND_COUNT] + 1 AND NOT 1;
2
2 ! Access the directory.
2 !
2
2 DIR_ACCESS (.FIB, 1);
2
2 ! Search the directory for the indicated name. If the search succeeds,
2 ! we have a duplicate entry. If supersede is enabled, do it; otherwise,
2 ! take an error exit. If the search failed, make a new entry.
2 !
2
2 DIR_RECORD = 0;
2 LAST_ENTRY[0] = 0;
2 STATUS =  DIR_SCAN (NAME_DESC, 0, 0, 0, 0, 0, -1);
2 IF .DIR_VERSION NEQ 0
2 THEN FIB[FIB$W_VERLIMIT] = .VERSION_LIMIT;
2
2 IF .STATUS
2 AND NOT .NAME_DESC[FND_MAX_VER]
2 AND .NAME_DESC[FND_VERSION] NEQ 0
```

```
2 THEN
3     BEGIN
3     IF NOT .FIB[FIB$V_SUPERSEDE]
3     THEN ERR_EXIT (SS$_DUPFILENAME);
3
3     DIR_END = .DIR_VERSION;
3     PREV_VERSION = .DIR_VERSION[DIR$W_VERSION];
3     CH$MOVE (FIB$S_FID, DIR_VERSION[DIR$W_FID], SUPER_FID);
3     APPLY_RVN (SUPER_FID[FID$W_RVN], .CURRENT_RVN);
3     CH$MOVE (FIB$S_FID, FIB[FIB$W_FID], DIR_VERSION[DIR$W_FID]);
3     DEFAULT_RVN (DIR_VERSION[DIR$W_FID_RVN], .CURRENT_RVN);
3     USER_STATUS[0] = SS$_SUPERSEDE;
3     CLEANUP_FLAGS[CLF_SUPERSEDE] = 1;
3
3 ! Determine if the entry being superseded is of the form xxx.DIR;1. If
3 ! so, bump the directory sequence count in the UCB to invalidate RMS caches.
3 !
3
3     IF .DIR_VERSION[DIR$W_VERSION] EQL 1         ! simple tests first
3     THEN
4         BEGIN
4         P = CH$FIND_SUB (.DIR_ENTRY[DIR$B_NAMECOUNT], DIR_ENTRY[DIR$T_NAME],
4                         4, UPLIT BYTE ('.DIR'));
4         IF NOT CH$FAIL (.P)
4         AND .P + 4 EQL DIR_ENTRY[DIR$T_NAME] + .DIR_ENTRY[DIR$B_NAMECOUNT]
4         THEN KERNEL_CALL (UPDATE_DIRSEQ);
3         END;
3     END
3
3 ! The operation is not a supersede. Create the new directory entry.
3 !
3
2 ELSE
2     MAKE_ENTRY (NAME_DESC, .FIB);
2
2 ! Determine whether higher or lower versions exist of the new entry.
2 ! This is deduced from the relative position of the new version in the record.
2 !
2
2 IF .VERSION_COUNT GTRU 0
2 THEN FIB[FIB$V_HIGHVER] = 1;
2
2 IF .DIR_VERSION LSSA .DIR_ENTRY + .DIR_ENTRY[DIR$W_SIZE] + 2 - DIR$C_VERSION
2 OR
3     BEGIN
3     VBN = .DIR_VBN;
3     NEXT_DIR_REC (.DIR_ENTRY, VBN) NEQ 0
3     END
2 THEN FIB[FIB$V_LOWVER] = 1;
2
2 FIB[FIB$W_VERLIMIT] = .VERSION_LIMIT;
2
2 ! Write out the modified directory block and
2 ! return the resultant directory string to the caller.
2 !
2
2 KERNEL_CALL (RETURN_DIR, .RESULT_LENGTH, .RESULT, .ABD);
```

```
;  276         1265  2
;  277         1266  2 MARK_DIRTY (.DIR_BUFFER);
;  278         1267  2
;  279         1268  2 PMS_END_SUB ();
;  280         1269  2
;  281         1270  1 END;                                           ! end of routine ENTER


                                                                               .TITLE  ENTER
                                                                               .IDENT  \V04-001\

                                                                               .PSECT  $CODE$,NOWRT,2

                                    52  49  44  2E  00000 P.AAA:               .ASCII  \.DIR\                                    ;

                                                                               .EXTRN  PMS_START_SUB, PMS_END_SUB
                                                                               .EXTRN  DIR_ACCESS, PARSE_NAME
                                                                               .EXTRN  DIR_SCAN, UPDATE_DIRSEQ
                                                                               .EXTRN  NEXT_DIR_REC, RETURN_DIR
                                                                               .EXTRN  MARK_DIRTY

                                          01FC 00000                           .ENTRY  ENTER, Save R2,R3,R4,R5,R6,R7,R8          ; 1094
                     5E         9C    AE   9E 00002                            MOVAB   -100(SP), SP                              ;
                     56       00DC    CA   9E 00006                            MOVAB   220(BASE), R6                             ; 1145
                     57       01FE    CA   9E 0000B                            MOVAB   510(BASE), R7                             ;
                     58         0C    A6   9E 00010                            MOVAB   12(R6), R8                                ; 1147
                                      07   DD 00014                            PUSHL   #7                                        ; 1167
              0000G  CF               01   FB 00016                            CALLS   #1, PMS_START_SUB                         ;
                     50         08    AC   D0 0001B                            MOVL    FIB, R0                                   ; 1172
                                04    A0   B5 0001F                            TSTW    4(R0)                                     ;
                                      03   12 00022                            BNEQ    1$                                        ;
                                      14   BF 00024                            CHMU    #20                                       ; 1173
                                           04 00026                            RET                                               ;
                     50         08    AC   D0 00027 1$:                        MOVL    FIB, R0                                   ; 1182
                     51         14    A0   3C 0002B                            MOVZWL  20(R0), R1                                ;
        7E           51  FFFFFDFF    8F   CB 0002F                            BICL3   #-513, R1, -(SP)                          ;
                     51         04    AC   D0 00037                            MOVL    ABD, R1                                   ; 1181
                     52         10    A1   9E 0003B                            MOVAB   16(R1), R2                                ;
                     50               62   3C 0003F                            MOVZWL  (R2), R0                                  ;
                                01 A240   9F 00042                            PUSHAB  1(R2)[R0]                                 ;
                     7E         12    A1   3C 00046                            MOVZWL  18(R1), -(SP)                             ; 1180
                                10    AE   9F 0004A                            PUSHAB  NAME_BUFFER                               ;
                                64    AE   9F 0004D                            PUSHAB  NAME_DESC                                 ;
              0000G  CF               05   FB 00050                            CALLS   #5, PARSE_NAME                            ;
                     05         55    AE   E9 00055                            BLBC    NAME_DESC+1, 2$                           ; 1183
                              0818    8F   BF 00059                            CHMU    #2072                                     ; 1184
                                           04 0005D                            RET                                               ;
                                60    AE   B5 0005E 2$:                        TSTW    NAME_DESC+12                              ; 1185
                                      03   18 00061                            BGEQ    3$                                        ;
                                60    AE   B4 00063                            CLRW    NAME_DESC+12                              ; 1186
        50           58         AE    01   C1 00066 3$:                        ADDL3   #1, NAME_DESC+4, R0                       ; 1187
                                50    01   8A 0006B                            BICB2   #1, NAME_LENGTH                           ;
                                      01   DD 0006E                            PUSHL   #1                                        ; 1192
                                08    AC   DD 00070                            PUSHL   FIB                                       ;
              0000G  CF               02   FB 00073                            CALLS   #2, DIR_ACCESS                            ;
                              00D8    CA   D4 00078                            CLRL    216(BASE)                                 ; 1199
                                1C    A6   94 0007C                            CLRB    28(R6)                                    ; 1200
```

```
ENTER                                                         G  1
V04-001                                              16-Sep-1984 00:21:45    VAX-11 Bliss-32 V4.0-742         Page   7
                                                     14-Sep-1984 12:30:19    DISK$VMSMASTER:[F11X.SRC]ENTER.B32;2      (2)

                                 7E          01  CE 0007F       MNEGL   #1, -(SP)                        ; 1201
                                             7E  7C 00082       CLRQ    -(SP)                            ;
                                             7E  7C 00084       CLRQ    -(SP)                            ;
                                             7E  D4 00086       CLRL    -(SP)                            ;
                                       6C    AE  9F 00088       PUSHAB  NAME_DESC                        ;
                          0000G  CF          07  FB 0008B       CALLS   #7, DIR_SCAN                     ;
                                             68  D5 00090       TSTL    (R8)                             ; 1202
                                             09  13 00092       BEQL    4$                               ;
                                 51    08    AC  D0 00094       MOVL    FIB, R1                          ; 1203
                             2C  A1    18    A6  B0 00098       MOVW    24(R6), 44(R1)                   ;
                                 03          50  E8 0009D 4$:   BLBS    STATUS, 6$                       ; 1205
                                           00A5  31 000A0 5$:   BRW     12$                              ;
                  F8         55  AE          01  E0 000A3 6$:   BBS     #1, NAME_DESC+1, 5$              ; 1206
                                       60    AE  B5 000A8       TSTW    NAME_DESC+12                     ; 1207
                                             F3  13 000AB       BEQL    5$                               ;
                                 50    08    AC  D0 000AD       MOVL    FIB, R0                          ; 1210
                  05         15  A0          02  E0 000B1       BBS     #2, 21(R0), 7$                   ;
                                     0868    8F  BF 000B6       CHMU    #2152                            ; 1211
                                                 04 000BA       RET                                      ;
                             10  A6          68  D0 000BB 7$:   MOVL    (R8), 16(R6)                     ; 1213
                           0152  CA    00    B8  32 000BF       CVTWL   @0(R8), 338(BASE)                ; 1214
                                 50          68  D0 000C5       MOVL    (R8), R0                         ; 1215
                  67         02  A0          06  28 000C8       MOVC3   #6, 2(R0), (R7)                  ;
                                       04    A7  95 000CD       TSTB    4(R7)                            ; 1216
                                             05  12 000D0       BNEQ    8$                               ;
                             04  A7    A0    AA  90 000D2       MOVB    -96(BASE), 4(R7)                 ;
                                 01    04    A7  91 000D7 8$:   CMPB    4(R7), #1                        ;
                                             08  12 000DB       BNEQ    9$                               ;
                                       A0    AA  D5 000DD       TSTL    -96(BASE)                        ;
                                             03  12 000E0       BNEQ    9$                               ;
                                       04    A7  94 000E2       CLRB    4(R7)                            ;
                                 51    08    AC  D0 000E5 9$:   MOVL    FIB, R1                          ; 1217
                                 50          68  D0 000E9       MOVL    (R8), R0                         ;
              02  A0         04  A1          06  28 000EC       MOVC3   #6, 4(R1), 2(R0)                 ;
                                 50          68  D0 000F2       MOVL    (R8), R0                         ; 1218
A0    AA      06  A0             08          00  ED 000F5       CMPZV   #0, #8, 6(R0), -96(BASE)         ;
                                             03  12 000FC       BNEQ    10$                              ;
                                       06    A0  94 000FE       CLRB    6(R0)                            ;
                             80  AA  0631    8F  3C 00101 10$:  MOVZWL  #1585, -128(BASE)                ; 1219
                                 6A          20  88 00107       BISB2   #32, (BASE)                      ; 1220
                                 01    00    B8  B1 0010A       CMPW    @0(R8), #1                       ; 1226
                                             43  12 0010E       BNEQ    13$                              ;
                                 50    08    A6  D0 00110       MOVL    8(R6), R0                        ; 1229
                                 51    05    A0  9A 00114       MOVZBL  5(R0), R1                        ;
06    A0          51       FEDF  CF          04  39 00118       MATCHC  #4, P.AAA, R1, 6(R0)             ; 1230
                                             03  13 00120       BEQL    11$                              ;
                                 53          04  D0 00122       MOVL    #4, R3                           ;
                                 53          04  C2 00125 11$:  SUBL2   #4, R3                           ;
                                             29  13 00128       BEQL    13$                              ; 1231
                                 53          04  C0 0012A       ADDL2   #4, R3                           ; 1232
                                 51    08    A6  D0 0012D       MOVL    8(R6), R1                        ;
                                 50    05    A1  9A 00131       MOVZBL  5(R1), R0                        ;
                                 50    06  A140  9E 00135       MOVAB   6(R1)[R0], R0                    ;
                                 50          53  D1 0013A       CMPL    R3, R0                           ;
                                             14  12 0013D       BNEQ    13$                              ;
                      00000000G  EF          00  FB 0013F       CALLS   #0, UPDATE_DIRSEQ                ; 1233
                                             0B  11 00146       BRB     13$                              ; 1205
```

```
                                                                  H  1
ENTER                                                             16-Sep-1984 00:21:45    VAX-11 Bliss-32 V4.0-742                   Page   8
V04-001                                                           14-Sep-1984 12:30:19    DISK$VMSMASTER:[F11X.SRC]ENTER.B32;2        (2)

                                          08    AC    DD 00148 12$:      PUSHL    FIB                                             ; 1241
                                          58    AE    9F 0014B           PUSHAB   NAME_DESC                                       ;
                           0000V   CF             02  FB 0014E           CALLS    #2, MAKE_ENTRY                                  ;
                                          1A    A6    B5 00153 13$:      TSTW     26(R6)                                          ; 1247
                                                09    13 00156           BEQL     14$                                             ;
                                   50     08    AC    D0 00158           MOVL     FIB, R0                                         ; 1248
                           15      A0     80    8F    88 0015C           BISB2    #128, 21(R0)                                    ;
                                   50     08    B6    3C 00161 14$:      MOVZWL   @8(R6), R0                                      ; 1250
                                   50     08    A6    C0 00165           ADDL2    8(R6), R0                                       ;
                                   50           06    C2 00169           SUBL2    #6, R0                                          ;
                                   50           68    D1 0016C           CMPL     (R8), R0                                        ;
                                                11    1F 0016F           BLSSU    15$                                             ;
                                   6E           66    D0 00171           MOVL     (R6), VBN                                       ; 1253
                                                5E    DD 00174           PUSHL    SP                                              ; 1254
                                          08    A6    DD 00176           PUSHL    8(R6)                                           ;
                           0000G   CF           02    FB 00179           CALLS    #2, NEXT_DIR_REC                                ;
                                                50    D5 0017E           TSTL     R0                                              ;
                                                09    13 00180           BEQL     16$                                             ;
                                   50     08    AC    D0 00182 15$:      MOVL     FIB, R0                                         ; 1256
                           15      A0     40    8F    88 00186           BISB2    #64, 21(R0)                                     ;
                                   50     08    AC    D0 0018B 16$:      MOVL     FIB, R0                                         ; 1258
                           2C      A0     18    A6    B0 0018F           MOVW     24(R6), 44(R0)                                  ;
                                          04    AC    DD 00194           PUSHL    ABD                                             ; 1264
                                   7E     0C    AC    7D 00197           MOVQ     RESULT_LENGTH, -(SP)                            ;
                           0000G   CF           03    FB 0019B           CALLS    #3, RETURN_DIR                                  ;
                                          04    A6    DD 001A0           PUSHL    4(R6)                                           ; 1266
                           0000G   CF           01    FB 001A3           CALLS    #1, MARK_DIRTY                                  ;
                           0000G   CF           00    FB 001A8           CALLS    #0, PMS_END_SUB                                 ; 1268
                                                      04 001AD           RET                                                      ; 1270

; Routine Size:  430 bytes,    Routine Base:  $CODE$ + 0004
```

```
1 GLOBAL ROUTINE MAKE_ENTRY (NAME_DESC, FIB) : L_NORM NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine makes a new directory entry, either for a new file
1 !       name or for a new version of an existing file.
1 !
1 !
1 ! CALLING SEQUENCE:
1 !       MAKE_ENTRY (ARG1, ARG2)
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1: address of file name descriptor block
1 !       ARG2: address of user FIB
1 !
1 ! IMPLICIT INPUTS:
1 !       DIR_VBN: VBN of current directory block
1 !       DIR_BUFFER: buffer address of current directory block
1 !       DIR_ENTRY: address of directory record
1 !       DIR_VERSION: address of directory version entry
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       DIR_END: end of directory data
1 !       PREV_VERSION: version number of superseded entry
1 !
1 ! ROUTINE VALUE:
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !       directory buffer altered, directory may be extended
1 !
1 !--
1
2 BEGIN
2
2 !*****
2 SWITCHES NOSAFE;
2 ! To prevent incorrect formation of a value CSE on DIR_VERSION across the
2 ! call to DIR_SCAN in the auto-purge logic. Compiler: T4.0-741.
2 !*****
2
2 MAP
2       NAME_DESC       : REF BBLOCK,   ! file name descriptor block
2       FIB             : REF BBLOCK;   ! user FIB
2
2 LOCAL
2       STATUS,                         ! misc status value
2       EOF,                            ! entering at end of file flag
2       DN              : REF BBLOCK,   ! local pointer to name descriptor
2       NAME_LENGTH,                    ! length of file name, rounded even
2       NEW_SIZE,                       ! size in bytes of new record
2       VERSIONS,                       ! version limit for new directory record
```

```
2               SAVE_VERSION,                   ! saved version number of new entry
2               SAVE_DIR_CONTEXT : BBLOCK [DCX_LENGTH]; ! saved context of enter
2
2       BIND_COMMON;
2
2       DIR_CONTEXT_DEF;
2
2       EXTERNAL ROUTINE
2               DIR_SCAN        : L_NORM,       ! scan directory file
2               READ_BLOCK      : L_NORM,       ! read a disk block
2               REMOVE          : L_NORM,       ! remove a directory entry
2               SHUFFLE_DIR     : L_NORM,       ! extend the directory
2               NEXT_REC        : L_NORM,       ! get next directory record
2               UPDATE_INDX     : NOVALUE;      ! update directory FCB index
2
2
2       ! Set up the position for the insert. If DIR_ENTRY came back zero, either
2       ! we scanned off the end of the directory, or this is a cleanup from a REMOVE
2       ! in which the block DIR_VBN got squished out. We read back the previous
2       ! block, and set up to append the entry, unless it was the first, in which
2       ! case we read the first and enter at its beginning. Scan to the end of the
2       ! records in the block.
2       !
2
2       EOF = 0;
2       DN = .NAME_DESC;
2       NAME_LENGTH = .DN[FND_COUNT] + 1 AND NOT 1;
2       DIR_END = .DIR_ENTRY;
2       IF .DIR_ENTRY EQL 0
2       THEN
3           BEGIN
3           IF .DIR_VBN GTRU 1
3           THEN
4               BEGIN
4               EOF = .EOF + 1;
4               DIR_VBN = .DIR_VBN - 1;
3               END;
3           DIR_ENTRY = DIR_END = DIR_BUFFER = READ_BLOCK (.DIR_VBN+.DIR_FCB[FCB$L_STLBN]-1,
3                                                       1, DIRECTORY_TYPE);
3
2           END;
2
2       UNTIL .DIR_END[DIR$W_SIZE] EQL 65535
2       DO DIR_END = NEXT_REC (.DIR_END);
2       IF .EOF
2       THEN DIR_ENTRY = .DIR_END;
2       DIR_END = .DIR_END + 2;
2
2       ! If there was not a name match, we are constructing a whole new record.
2       ! Compute the record size and see if there is enough space. If not, extend
2       ! the directory. Then shuffle down the rest of the records and build the
2       ! new entry. We update the directory index if this is a new last record
2       ! in the block.
2       !
2
2       IF .DIR_VERSION EQL 0
2       THEN
3           BEGIN
```

```
3       IF .DN[FND_VERSION] EQL 0
3       THEN DN[FND_VERSION] = .DN[FND_VERSION] + 1;
3
3       NEW_SIZE = DIR$C_LENGTH + DIR$C_VERSION + .NAME_LENGTH;
3       IF .NEW_SIZE GTRU .DIR_BUFFER + 512 - .DIR_END
3       THEN SHUFFLE_DIR (1);
3
3       IF .DIR_ENTRY[DIR$W_SIZE] EQL 65535
3       THEN
3           UPDATE_INDX (.DIR_VBN-1, .DN [FND_COUNT], .DN [FND_STRING],
3                        .DIR_FCB);
3
3       CH$MOVE (.DIR_END-.DIR_ENTRY, .DIR_ENTRY, .DIR_ENTRY+.NEW_SIZE);
3
3       DIR_ENTRY[DIR$W_SIZE] = .NEW_SIZE - 2;
3       VERSIONS = .FIB[FIB$W_VERLIMIT];
3       IF .VERSIONS EQL 0
3       THEN VERSIONS = .DIR_FCB[FCB$W_VERSIONS];
3       IF .VERSIONS EQL 0
3       THEN VERSIONS = 32767;
3       DIR_ENTRY[DIR$W_VERLIMIT] = .VERSIONS;
3       VERSION_LIMIT = .VERSIONS;
3
3       DIR_ENTRY[DIR$B_FLAGS] = DIR$C_FID;
3       DIR_ENTRY[DIR$B_NAMECOUNT] = .DN[FND_COUNT];
3       CH$COPY (.DN[FND_COUNT], .DN[FND_STRING],
3                0, .NAME_LENGTH, DIR_ENTRY[DIR$T_NAME]);
3
3       DIR_VERSION = .DIR_ENTRY + .NEW_SIZE - DIR$C_VERSION;
3       END
3
3
3  ! Otherwise we are adding a new version to an existing entry. If the
3  ! version limit is less than maximal, check for version overflow. If
3  ! so, move out the oldest version to be superseded. We do this by
3  ! saving the current directory position and scanning to the lowest
3  ! version of the file. If the scan for lowest version comes back with
3  ! an error, this indicates that we were already at the lowest version;
3  ! if so, give an error.
3  !
3
2  ELSE
3       BEGIN
3       IF  .DN[FND_VERSION] EQL 0
4       OR (.DN[FND_MAX_VER] AND .DN[FND_VERSION] LEQU .DIR_VERSION[DIR$W_VERSION])
3       THEN DN[FND_VERSION] = .DIR_VERSION[DIR$W_VERSION] + 1;
3       IF .DN[FND_VERSION] LSS 0
3       THEN ERR_EXIT (SS$_BADFILEVER);
3
3       IF .VERSION_LIMIT LSSU 32767
3       THEN
4           BEGIN
4           SAVE_VERSION = .DN[FND_VERSION];
4           DN[FND_FLAGS] = 0;
4           DN[FND_VERSION] = 32768;
4           CH$MOVE (DCX_LENGTH, DIR_CONTEXT, SAVE_DIR_CONTEXT);
4           STATUS = DIR_SCAN (.DN, 0, .DIR_VBN-1, .DIR_ENTRY, .DIR_VERSION, .DIR_PRED, -1);
```

```
4          IF .STATUS
4          THEN
5              BEGIN
5              IF .VERSION_COUNT + 1 GEQU .VERSION_LIMIT
5              THEN
6                  BEGIN
6                  PREV_VERSION = .DIR_VERSION[DIR$W_VERSION];
6                  CH$MOVE (.DIR_ENTRY[DIR$B_NAMECOUNT]+1,
6                           DIR_ENTRY[DIR$B_NAMECOUNT], PREV_NAME);
6                  CH$MOVE (FIB$S_FID, DIR_VERSION[DIR$W_FID], SUPER_FID);
6                  APPLY_RVN (SUPER_FID[FID$W_RVN], .CURRENT_RVN);
6                  USER_STATUS[0] = SS$_FILEPURGED;
6                  CLEANUP_FLAGS[CLF_SUPERSEDE] = 1;
6                  REMOVE (.DIR_VBN EQL .SAVE_DIR_CONTEXT[DCX_VBN]
6                          AND .DIR_ENTRY - .DIR_BUFFER EQL
6                              .SAVE_DIR_CONTEXT[DCX_ENTRY] - .SAVE_DIR_CONTEXT[DCX_BUFFER]);
5                  END;
5              END
4          ELSE
4              IF .VERSION_COUNT GEQU .VERSION_LIMIT
4              THEN ERR_EXIT (SS$_TOOMANYVER);
4
4  ! Now reposition to the point of the enter to make the new entry.
4  !
4
4          DN[FND_VERSION] = .SAVE_VERSION;
4          RESTORE_DIR (SAVE_DIR_CONTEXT);
3          END;
3
3  ! Check available space and extend if necessary. If we are out of space
3  ! in the block, the version is going to the front of the record, the
3  ! record is the only one in its block, and the record has a
3  ! predecessor, we backspace to the previous record and try to append
3  ! the entry. This prevents pathological directory fragmentation when
3  ! new versions are inserted in descending order, as might happen during
3  ! a wild card copy. If this gambit fails or is not applicable, we have
3  ! to split the block.
3  !
3
3      IF DIR$C_VERSION GTRU .DIR_BUFFER + 512 - .DIR_END
3      THEN
4          BEGIN
4          IF  .DIR_ENTRY EQL .DIR_BUFFER
4          AND .DIR_VBN GTRU 1
4          AND .DIR_VERSION EQL .DIR_ENTRY + .NAME_LENGTH + DIR$C_LENGTH
4          AND .BBLOCK [NEXT_REC (.DIR_ENTRY), DIR$W_SIZE] EQL 65535
4          AND CH$EQL (.DIR_ENTRY[DIR$B_NAMECOUNT]+1,
4                      DIR_ENTRY[DIR$B_NAMECOUNT],
4                     .DIR_ENTRY[DIR$B_NAMECOUNT]+1,
4                      LAST_ENTRY)
4          THEN
5              BEGIN
5              DIR_VBN = .DIR_VBN - 1;
5              DIR_BUFFER = DIR_END = READ_BLOCK (.DIR_VBN-1+.DIR_FCB[FCB$L_STLBN], 1, DIRECTORY_TYPE);
5              DO
6                  BEGIN
6                  DIR_ENTRY = .DIR_END;
```

```
;   511       1499  6                  DIR_END = NEXT_REC (.DIR_END);
;   512       1500  6                  END
;   513       1501  5              UNTIL .DIR_END[DIR$W_SIZE] EQL 65535;
;   514       1502  5              DIR_VERSION = .DIR_END;
;   515       1503  5              DIR_END = .DIR_END + 2;
;   516       1504  5              IF DIR$C_VERSION GTRU .DIR_BUFFER + 512 - .DIR_END
;   517       1505  5              THEN SHUFFLE_DIR (1);
;   518       1506  5              END
;   519       1507  4          ELSE
;   520       1508  4              SHUFFLE_DIR (1);
;   521       1509  3          END;
;   522       1510  3
;   523       1511  3      DIR_ENTRY[DIR$W_SIZE] = .DIR_ENTRY[DIR$W_SIZE] + DIR$C_VERSION;
;   524       1512  3      CH$MOVE (.DIR_END-.DIR_VERSION, .DIR_VERSION, .DIR_VERSION+DIR$C_VERSION);
;   525       1513  2      END;
;   526       1514  2
;   527       1515  2  ! Now insert the version number and file ID into the version slot.
;   528       1516  2  !
;   529       1517  2
;   530       1518  2  CLEANUP_FLAGS[CLF_REMOVE] = 1;
;   531       1519  2  DIR_VERSION[DIR$W_VERSION] = .DN[FND_VERSION];
;   532       1520  2  CH$MOVE (FIB$S_FID, FIB[FIB$W_FID], DIR_VERSION[DIR$W_FID]);
;   533       1521  2  DEFAULT_RVN (DIR_VERSION[DIR$W_FID_RVN], .CURRENT_RVN);
;   534       1522  2
;   535       1523  1  END;                                ! end of routine MAKE_ENTRY


                                                                        .EXTRN  READ_BLOCK, REMOVE
                                                                        .EXTRN  SHUFFLE_DIR, NEXT_REC
                                                                        .EXTRN  UPDATE_INDX

                                                    0BFC 00000          .ENTRY  MAKE_ENTRY, Save R2,R3,R4,R5,R6,R7,R8,R9,-   ; 1271
                                                                                R11                                          ;
                                5E      90      AE  9E 00002            MOVAB   -112(SP), SP                                 ;
                                      00D0      CA  9F 00006            PUSHAB  208(BASE)                                    ; 1329
                                58    00DC      CA  9E 0000A            MOVAB   220(BASE), R8                                ;
                                      01FE      CA  9F 0000F            PUSHAB  510(BASE)                                    ;
                                        04      A8  9F 00013            PUSHAB  4(R8)                                        ; 1331
                                5B      08      A8  9E 00016            MOVAB   8(R8), R11                                   ;
                                        0C      A8  9F 0001A            PUSHAB  12(R8)                                       ;
                                        10      A8  9F 0001D            PUSHAB  16(R8)                                       ;
                                                52  D4 00020            CLRL    EOF                                          ; 1352
                                59      04      AC  D0 00022            MOVL    NAME_DESC, DN                                ; 1353
        50        04      A9                    01  C1 00026            ADDL3   #1, 4(DN), R0                                ; 1354
        57                50                    01  CB 0002B            BICL3   #1, R0, NAME_LENGTH                          ;
                  00      BE                    6B  D0 0002F            MOVL    (R11), @0(SP)                                ; 1355
                                                6B  D5 00033            TSTL    (R11)                                        ; 1356
                                                29  12 00035            BNEQ    2$                                           ;
                          01                    68  D1 00037            CMPL    (R8), #1                                     ; 1359
                                                04  1B 0003A            BLEQU   1$                                           ;
                                                52  D6 0003C            INCL    EOF                                          ; 1362
                                                68  D7 0003E            DECL    (R8)                                         ; 1363
                                                02  DD 00040  1$:       PUSHL   #2                                           ; 1365
                                                01  DD 00042            PUSHL   #1                                           ;
                                50      18      BE  D0 00044            MOVL    @24(SP), R0                                  ;
        50                68      30      A0    C1 00048            ADDL3   48(R0), (R8), R0                             ;
```

```
                                FF   A0  9F 0004D        PUSHAB  -1(R0)                                   ;
               0000G  CF             03  FB 00050        CALLS   #3, READ_BLOCK                           ;
                  08  BE             50  D0 00055        MOVL    R0, @8(SP)                               ;
                  00  BE             50  D0 00059        MOVL    R0, @0(SP)                               ;
                      6B             50  D0 0005D        MOVL    R0, (R11)                                ;
                      50       00    BE  D0 00060  2$:   MOVL    @0(SP), R0                               ; 1369
                FFFF  8F             60  B1 00064        CMPW    (R0), #65535                             ;
                                     0E  13 00069        BEQL    3$                                       ;
                               00    BE  DD 0006B        PUSHL   @0(SP)                                   ; 1370
               0000G  CF             01  FB 0006E        CALLS   #1, NEXT_REC                             ;
                  00  BE             50  D0 00073        MOVL    R0, @0(SP)                               ;
                                     E7  11 00077        BRB     2$                                       ;
                      04             52  E9 00079  3$:   BLBC    EOF, 4$                                  ; 1371
                      6B       00    BE  D0 0007C        MOVL    @0(SP), (R11)                            ; 1372
                  00  BE             02  C0 00080  4$:   ADDL2   #2, @0(SP)                               ; 1373
                      50       0C    A9  9E 00084        MOVAB   12(DN), R0                               ; 1385
                               04    BE  D5 00088        TSTL    @4(SP)                                   ; 1382
                                     03  13 0008B        BEQL    5$                                       ;
                                 0094    31 0008D        BRW     10$                                      ;
                                     60  B5 00090  5$:   TSTW    (R0)                                     ; 1385
                                     02  12 00092        BNEQ    6$                                       ;
                                     60  B6 00094        INCW    (R0)                                     ; 1386
                      56       0E    A7  9E 00096  6$:   MOVAB   14(R7), NEW_SIZE                         ; 1388
        50        08  BE       00    BE  C3 0009A        SUBL3   @0(SP), @8(SP), R0                       ; 1389
                      50     0200    C0  9E 000A0        MOVAB   512(R0), R0                              ;
                      50             56  D1 000A5        CMPL    NEW_SIZE, R0                             ;
                                     07  1B 000A8        BLEQU   7$                                       ;
                                     01  DD 000AA        PUSHL   #1                                       ; 1390
               0000G  CF             01  FB 000AC        CALLS   #1, SHUFFLE_DIR                          ;
                FFFF  8F       00    BB  B1 000B1  7$:   CMPW    @0(R11), #65535                          ; 1392
                                     10  12 000B7        BNEQ    8$                                       ;
                               10    BE  DD 000B9        PUSHL   @16(SP)                                  ; 1395
                      7E       04    A9  7D 000BC        MOVQ    4(DN), -(SP)                             ; 1394
        7E            68             01  C3 000C0        SUBL3   #1, (R8), -(SP)                          ;
               0000G  CF             04  FB 000C4        CALLS   #4, UPDATE_INDX                          ;
        51        00  BE             6B  C3 000C9  8$:   SUBL3   (R11), @0(SP), R1                        ; 1397
   00 BB46        00  BB             51  28 000CE        MOVC3   R1, @0(R11), @0(R11)[NEW_SIZE]           ;
   00   BB                           56                                                                   ;
                                     02  A3 000D5        SUBW3   #2, NEW_SIZE, @0(R11)                    ; 1399
                      50       08    AC  D0 000DA        MOVL    FIB, R0                                  ; 1400
                      51       2C    A0  3C 000DE        MOVZWL  44(R0), VERSIONS                         ;
                                     0F  12 000E2        BNEQ    9$                                       ; 1401
                      50       10    BE  D0 000E4        MOVL    @16(SP), R0                              ; 1402
                      51       40    A0  3C 000E8        MOVZWL  64(R0), VERSIONS                         ;
                                     05  12 000EC        BNEQ    9$                                       ; 1403
                      51     7FFF    8F  3C 000EE        MOVZWL  #32767, VERSIONS                         ; 1404
                      50             6B  D0 000F3  9$:   MOVL    (R11), R0                                ; 1405
                  02  A0             51  B0 000F6        MOVW    VERSIONS, 2(R0)                          ;
                  18  A8             51  B0 000FA        MOVW    VERSIONS, 24(R8)                         ; 1406
                      50             6B  D0 000FE        MOVL    (R11), R0                                ; 1408
                               04    A0  94 00101        CLRB    4(R0)                                    ;
                      50             6B  D0 00104        MOVL    (R11), R0                                ; 1409
                  05  A0       04    A9  90 00107        MOVB    4(DN), 5(R0)                             ;
                      50             6B  D0 0010C        MOVL    (R11), R0                                ; 1411
 57               00  08  B9       04    A9  2C 0010F    MOVC5   4(DN), @8(DN), #0, NAME_LENGTH, 6(R0)    ;
                               06    A0     00116                                                         ;
        50            6B             56  C1 00118        ADDL3   NEW_SIZE, (R11), R0                      ; 1413
                  04  BE       F8    A0  9E 0011C        MOVAB   -8(R0), @4(SP)                           ;
```

```
                                        01BB 31 00121        BRW     29$                                  ; 1382
                                          60 B5 00124 10$:   TSTW    (R0)                                 ; 1428
                                          0D 13 00126        BEQL    11$                                  ;
         11                69             09 E1 00128        BBC     #9, (DN), 12$                        ; 1429
                           51      04     BE D0 0012C        MOVL    @4(SP), R1                           ;
                           61             60 B1 00130        CMPW    (R0), (R1)                           ;
                                          08 1A 00133        BGTRU   12$                                  ;
                           51      04     BE D0 00135 11$:   MOVL    @4(SP), R1                           ; 1430
              60           61             01 A1 00139        ADDW3   #1, (R1), (R0)                       ;
                                   0C     A9 B5 0013D 12$:   TSTW    12(DN)                               ; 1431
                                          05 18 00140        BGEQ    13$                                  ;
                                 0820     8F BF 00142        CHMU    #2080                                ; 1432
                                             04 00146        RET                                          ;
                     7FFF          8F      18     A8 B1 00147 13$:   CMPW    24(R8), #32767               ; 1434
                                          03 1F 0014D        BLSSU   14$                                  ;
                                        00D7 31 0014F        BRW     23$                                  ;
                           56      0C     A9 32 00152 14$:   CVTWL   12(DN), SAVE_VERSION                 ; 1437
                                          69 B4 00156        CLRW    (DN)                                 ; 1438
                       0C  A9    8000     8F B0 00158        MOVW    #-32768, 12(DN)                      ; 1439
         14   AE           68    0070     8F 28 0015E        MOVC3   #112, (R8), SAVE_DIR_CONTEXT         ; 1440
                           7E             01 CE 00165        MNEGL   #1, -(SP)                            ; 1441
                                   14     A8 DD 00168        PUSHL   20(R8)                               ;
                                   0C     BE DD 0016B        PUSHL   @12(SP)                              ;
                                          6B DD 0016E        PUSHL   (R11)                                ;
              7E           68             01 C3 00170        SUBL3   #1, (R8), -(SP)                      ;
                                          7E D4 00174        CLRL    -(SP)                                ;
                                          59 DD 00176        PUSHL   DN                                   ;
                    0000G  CF             07 FB 00178        CALLS   #7, DIR_SCAN                         ;
                           03             50 E8 0017D        BLBS    STATUS, 15$                          ; 1442
                                        008E 31 00180        BRW     21$                                  ;
                           50      1A     A8 3C 00183 15$:   MOVZWL  26(R8), R0                           ; 1445
                                          50 D6 00187        INCL    R0                                   ;
  50     18   A8           10             00 ED 00189        CMPZV   #0, #16, 24(R8), R0                  ;
                                          7E 1A 0018F        BGTRU   20$                                  ;
                           50      04     BE D0 00191        MOVL    @4(SP), R0                           ; 1448
                     0152  CA             60 32 00195        CVTWL   (R0), 338(BASE)                      ;
                           50             6B D0 0019A        MOVL    (R11), R0                            ; 1449
                           51      05     A0 9A 0019D        MOVZBL  5(R0), R1                            ;
                                          51 D6 001A1        INCL    R1                                   ;
       0156   CA       05  A0             51 28 001A3        MOVC3   R1, 5(R0), 342(BASE)                 ; 1450
                           50      04     BE D0 001AA        MOVL    @4(SP), R0                           ; 1451
         0C   BE       02  A0             06 28 001AE        MOVC3   #6, 2(R0), @12(SP)                   ;
              50       0C  AE             04 C1 001B4        ADDL3   #4, 12(SP), R0                       ; 1452
                                          60 95 001B9        TSTB    (R0)                                 ;
                                          09 12 001BB        BNEQ    16$                                  ;
              50       0C  AE             04 C1 001BD        ADDL3   #4, 12(SP), R0                       ;
                           60      A0     AA 90 001C2        MOVB    -96(BASE), (R0)                      ;
              50       0C  AE             04 C1 001C6 16$:   ADDL3   #4, 12(SP), R0                       ;
                           01             60 91 001CB        CMPB    (R0), #1                             ;
                                          0C 12 001CE        BNEQ    17$                                  ;
                                   A0     AA D5 001D0        TSTL    -96(BASE)                            ;
                                          07 12 001D3        BNEQ    17$                                  ;
              50       0C  AE             04 C1 001D5        ADDL3   #4, 12(SP), R0                       ;
                                          60 94 001DA        CLRB    (R0)                                 ;
                       80  AA    0679     8F 3C 001DC 17$:   MOVZWL  #1657, -128(BASE)                    ; 1453
                           6A             20 88 001E2        BISB2   #32, (BASE)                          ; 1454
                                          51 D4 001E5        CLRL    R1                                   ; 1455
```

```
ENTER                                            C  2
V04-001                                16-Sep-1984 00:21:45    VAX-11 Bliss-32 V4.0-742                      Page  16
                                       14-Sep-1984 12:30:19    DISK$VMSMASTER:[F11X.SRC]ENTER.B32;2               (3)

                 14  AE          68 D1 001E7        CMPL    (R8), SAVE_DIR_CONTEXT                          :
                                 02 12 001EB        BNEQ    18$                                             :
                                 51 D6 001ED        INCL    R1                                              :
        50       6B  08          BE C3 001EF  18$:  SUBL3   @8(SP), (R11), R0                               : 1456
        52       1C  AE  18      AE C3 001F4        SUBL3   SAVE_DIR_CONTEXT+4, SAVE_DIR_CONTEXT+8, R2      : 1457
                                 53 D4 001FA        CLRL    R3                                              :
                     52          50 D1 001FC        CMPL    R0, R2                                          :
                                 02 12 001FF        BNEQ    19$                                             :
                                 53 D6 00201        INCL    R3                                              :
                     54          51 D2 00203  19$:  MCOML   R1, R4                                          : 1456
        7E           53          54 CB 00206        BICL3   R4, R3, -(SP)                                   :
              0000G  CF          01 FB 0020A        CALLS   #1, REMOVE                                      :
                                 0C 11 0020F  20$:  BRB     22$                                             : 1442
                 18  A8  1A      A8 B1 00211  21$:  CMPW    26(R8), 24(R8)                                  : 1461
                                 05 1F 00216        BLSSU   22$                                             :
                       0990      8F BF 00218        CHMU    #2448                                           : 1462
                                    04 0021C        RET                                                     :
                 0C  A9          56 B0 0021D  22$:  MOVW    SAVE_VERSION, 12(DN)                            : 1467
                         14      AE 9F 00221        PUSHAB  SAVE_DIR_CONTEXT                                : 1468
              0000V  CF          01 FB 00224        CALLS   #1, RESTORE_DIR                                 :
        50       08  BE  00      BE C3 00229  23$:  SUBL3   @0(SP), @8(SP), R0                              : 1481
                     50 0200     C0 9E 0022F        MOVAB   512(R0), R0                                     :
                     08          50 D1 00234        CMPL    R0, #8                                          :
                                 03 1F 00237        BLSSU   24$                                             :
                              0091 31 00239        BRW     28$                                             :
                 08  BE          6B D1 0023C  24$:  CMPL    (R11), @8(SP)                                   : 1484
                                 2F 12 00240        BNEQ    25$                                             :
                     01          68 D1 00242        CMPL    (R8), #1                                        : 1485
                                 7F 1B 00245        BLEQU   27$                                             :
        50           6B          57 C1 00247        ADDL3   NAME_LENGTH, (R11), R0                          : 1486
                     50          06 C0 0024B        ADDL2   #6, R0                                          :
                     50  04      BE D1 0024E        CMPL    @4(SP), R0                                      :
                                 72 12 00252        BNEQ    27$                                             :
                                 6B DD 00254        PUSHL   (R11)                                           : 1487
              0000G  CF          01 FB 00256        CALLS   #1, NEXT_REC                                    :
               FFFF  8F          60 B1 0025B        CMPW    (R0), #65535                                    :
                                 64 12 00260        BNEQ    27$                                             :
                     50          6B D0 00262        MOVL    (R11), R0                                       : 1488
                     51  05      A0 9A 00265        MOVZBL  5(R0), R1                                       :
                                 51 D6 00269        INCL    R1                                              :
   1C   A8       05  A0          51 29 0026B        CMPC3   R1, 5(R0), 28(R8)                               : 1489
                                 53 12 00271  25$:  BNEQ    27$                                             :
                                 68 D7 00273        DECL    (R8)                                            : 1494
                                 02 DD 00275        PUSHL   #2                                              : 1495
                                 01 DD 00277        PUSHL   #1                                              :
                     50  18      BE D0 00279        MOVL    @24(SP), R0                                     :
        58           68  30      A0 C1 0027D        ADDL3   48(R0), (R8), R8                                :
                         FF      A8 9F 00282        PUSHAB  -1(R8)                                          :
              0000G  CF          03 FB 00285        CALLS   #3, READ_BLOCK                                  :
                 00  BE          50 D0 0028A        MOVL    R0, @0(SP)                                      :
                 08  BE          50 D0 0028E        MOVL    R0, @8(SP)                                      :
                     6B  00      BE D0 00292  26$:  MOVL    @0(SP), (R11)                                   : 1498
                         00      BE DD 00296        PUSHL   @0(SP)                                          : 1499
              0000G  CF          01 FB 00299        CALLS   #1, NEXT_REC                                    :
                 00  BE          50 D0 0029E        MOVL    R0, @0(SP)                                      :
                     50  00      BE D0 002A2        MOVL    @0(SP), R0                                      : 1501
               FFFF  8F          60 B1 002A6        CMPW    (R0), #65535                                    :
```

```
                                                                       D  2
ENTER                                                            16-Sep-1984 00:21:45    VAX-11 Bliss-32 V4.0-742               Page  17
V04-001                                                          14-Sep-1984 12:30:19    DISK$VMSMASTER:[F11X.SRC]ENTER.B32;2    (3)

                                                     E5  12 002AB            BNEQ    26$                                  ;
                            04    BE        00       BE  D0 002AD            MOVL    @0(SP), @4(SP)                       ; 1502
                            00    BE                 02  C0 002B2            ADDL2   #2, @0(SP)                           ; 1503
                     50     08    BE        00       BE  C3 002B6            SUBL3   @0(SP), @8(SP), R0                   ; 1504
                                  50      0200       C0  9E 002BC            MOVAB   512(R0), R0                          ;
                                  08                 50  D1 002C1            CMPL    R0, #8                               ;
                                                     07  1E 002C4            BGEQU   28$                                  ;
                                                     01  DD 002C6  27$:      PUSHL   #1                                   ; 1508
                         0000G    CF                 01  FB 002C8            CALLS   #1, SHUFFLE_DIR                      ;
                            00    BB                 08  A0 002CD  28$:      ADDW2   #8, @0(R11)                          ; 1511
                                  50        04       BE  D0 002D1            MOVL    @4(SP), R0                           ; 1512
                     51     00    BE                 50  C3 002D5            SUBL3   R0, @0(SP), R1                       ;
               08    A0           60                 51  28 002DA            MOVC3   R1, (R0), 8(R0)                      ;
                            02    AA        40       8F  88 002DF  29$:      BISB2   #64, 2(BASE)                         ; 1518
                                  50        04       BE  D0 002E4            MOVL    @4(SP), R0                           ; 1519
                                  60        0C       A9  B0 002E8            MOVW    12(DN), (R0)                         ;
                                  51        08       AC  D0 002EC            MOVL    FIB, R1                              ; 1520
                                  50        04       BE  D0 002F0            MOVL    @4(SP), R0                           ;
               02    A0     04    A1                 06  28 002F4            MOVC3   #6, 4(R1), 2(R0)                     ;
                                  50        04       BE  D0 002FA            MOVL    @4(SP), R0                           ; 1521
     A0   AA   06    A0           08                 00  ED 002FE            CMPZV   #0, #8, 6(R0), -96(BASE)             ;
                                                     03  12 00305            BNEQ    30$                                  ;
                                            06       A0  94 00307            CLRB    6(R0)                                ;
                                                         04 0030A  30$:      RET                                          ; 1523

; Routine Size:  779 bytes,    Routine Base:  $CODE$ + 01B2
```

```
1 GLOBAL ROUTINE RESTORE_DIR (CONTEXT) : L_NORM NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine repositions a directory to the location specified
1 !       in the context block and copies the context back into the
1 !       main context block.
1 !
1 ! CALLING SEQUENCE:
1 !       RESTORE_DIR (ARG1)
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1: address of context block to use
1 !
1 ! IMPLICIT INPUTS:
1 !       NONE
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       DIR_CONTEXT: receives contents of supplied context
1 !
1 ! ROUTINE VALUE:
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !       directory block read
1 !
1 !--
1
2 BEGIN
2
2 MAP
2       CONTEXT         : REF BBLOCK;   ! directory context arg
2
2 BIND_COMMON;
2
2 DIR_CONTEXT_DEF;
2
2 EXTERNAL ROUTINE
2       READ_BLOCK      : L_NORM;       ! read a disk block
2
2
2 ! We have to reread the block if either the supplied context is the
2 ! main context (indicating a required reposition) or the supplied
2 ! context points to a different block than the main context (indicating
2 ! that we have actually moved).
2 !
2
2 IF .CONTEXT EQL DIR_CONTEXT
2 OR .CONTEXT[DCX_VBN] NEQ .DIR_VBN
2 OR .CONTEXT[DCX_BUFFER] NEQ .DIR_BUFFER
2 THEN
3     BEGIN
```

```
;  594      1581  3          DIR_VBN = .CONTEXT[DCX_VBN];
;  595      1582  3          DIR_ENTRY = .CONTEXT[DCX_ENTRY] - .CONTEXT[DCX_BUFFER];
;  596      1583  3          IF .CONTEXT[DCX_VERSION] NEQ 0
;  597      1584  3          THEN DIR_VERSION = .CONTEXT[DCX_VERSION] - .CONTEXT[DCX_BUFFER];
;  598      1585  3          IF .CONTEXT[DCX_END] NEQ 0
;  599      1586  3          THEN DIR_END = .CONTEXT[DCX_END] - .CONTEXT[DCX_BUFFER];
;  600      1587  3          IF .CONTEXT[DCX_PRED] NEQ 0
;  601      1588  3          THEN DIR_PRED = .CONTEXT[DCX_PRED] - .CONTEXT[DCX_BUFFER];
;  602      1589  3          DIR_BUFFER = READ_BLOCK (.DIR_VBN+.DIR_FCB[FCB$L_STLBN]-1, 1, DIRECTORY_TYPE);
;  603      1590  3          DIR_ENTRY = .DIR_ENTRY + .DIR_BUFFER;
;  604      1591  3          IF .DIR_VERSION NEQ 0
;  605      1592  3          THEN DIR_VERSION = .DIR_VERSION + .DIR_BUFFER;
;  606      1593  3          IF .DIR_END NEQ 0
;  607      1594  3          THEN DIR_END = .DIR_END + .DIR_BUFFER;
;  608      1595  3          IF .DIR_PRED NEQ 0
;  609      1596  3          THEN DIR_PRED = .DIR_PRED + .DIR_BUFFER;
;  610      1597  3          VERSION_LIMIT = .CONTEXT[DCX_VERLIMIT];
;  611      1598  3          VERSION_COUNT = .CONTEXT[DCX_VERCOUNT];
;  612      1599  3          CH$MOVE (FILENAME_LENGTH+1, CONTEXT[DCX_NAME], LAST_ENTRY);
;  613      1600  3          END
;  614      1601  2      ELSE
;  615      1602  2          CH$MOVE (DCX_LENGTH, .CONTEXT, DIR_CONTEXT);
;  616      1603  2
;  617      1604  1      END;                                        ! End of routine RESTORE_DIR


                                              003C 00000        .ENTRY  RESTORE_DIR, Save R2,R3,R4,R5                  ; 1524
                              52    00DC   CA   9E 00002        MOVAB   220(BASE), R2                                  ; 1560
                              53      04   A2   9E 00007        MOVAB   4(R2), R3                                      ; 1562
                              50      04   AC   D0 0000B        MOVL    CONTEXT, R0                                    ; 1576
                              52           50   D1 0000F        CMPL    R0, R2                                         ;
                                           0E   13 00012        BEQL    1$                                             ;
                              62           60   D1 00014        CMPL    (R0), (R2)                                     ; 1577
                                           09   12 00017        BNEQ    1$                                             ;
                              63      04   A0   D1 00019        CMPL    4(R0), (R3)                                    ; 1578
                                           03   12 0001D        BNEQ    1$                                             ;
                                         0095   31 0001F        BRW     8$                                             ;
                              62           60   D0 00022 1$:    MOVL    (R0), (R2)                                     ; 1581
                              50      04   AC   D0 00025        MOVL    CONTEXT, R0                                    ; 1582
          08    A2      08    A0      04   A0   C3 00029        SUBL3   4(R0), 8(R0), 8(R2)                            ;
                              50      04   AC   D0 00030        MOVL    CONTEXT, R0                                    ; 1583
                              0C           A0   D5 00034        TSTL    12(R0)                                         ;
                                           07   13 00037        BEQL    2$                                             ;
          0C    A2      0C    A0      04   A0   C3 00039        SUBL3   4(R0), 12(R0), 12(R2)                          ; 1584
                              50      04   AC   D0 00040 2$:    MOVL    CONTEXT, R0                                    ; 1585
                                      10   A0   D5 00044        TSTL    16(R0)                                         ;
                                           07   13 00047        BEQL    3$                                             ;
          10    A2      10    A0      04   A0   C3 00049        SUBL3   4(R0), 16(R0), 16(R2)                          ; 1586
                              50      04   AC   D0 00050 3$:    MOVL    CONTEXT, R0                                    ; 1587
                                      14   A0   D5 00054        TSTL    20(R0)                                         ;
                                           07   13 00057        BEQL    4$                                             ;
          14    A2      14    A0      04   A0   C3 00059        SUBL3   4(R0), 20(R0), 20(R2)                          ; 1588
                                           02   DD 00060 4$:    PUSHL   #2                                             ; 1589
                                           01   DD 00062        PUSHL   #1                                             ;
```

```
                               50    00D0    CA  D0 00064          MOVL    208(BASE), R0                            ;
                 50            62      30    A0  C1 00069          ADDL3   48(R0), (R2), R0                         ;
                                       FF    A0  9F 0006E          PUSHAB  -1(R0)                                   ;
                         0000G CF            03  FB 00071          CALLS   #3, READ_BLOCK                           ;
                               63            50  D0 00076          MOVL    R0, (R3)                                 ;
                         08    A2            63  C0 00079          ADDL2   (R3), 8(R2)                              ; 1590
                                       0C    A2  D5 0007D          TSTL    12(R2)                                   ; 1591
                                             04  13 00080          BEQL    5$                                       ;
                         0C    A2            63  C0 00082          ADDL2   (R3), 12(R2)                             ; 1592
                                       10    A2  D5 00086 5$:      TSTL    16(R2)                                   ; 1593
                                             04  13 00089          BEQL    6$                                       ;
                         10    A2            63  C0 0008B          ADDL2   (R3), 16(R2)                             ; 1594
                                       14    A2  D5 0008F 6$:      TSTL    20(R2)                                   ; 1595
                                             04  13 00092          BEQL    7$                                       ;
                         14    A2            63  C0 00094          ADDL2   (R3), 20(R2)                             ; 1596
                               50      04    AC  D0 00098 7$:      MOVL    CONTEXT, R0                              ; 1597
                         18    A2      18    A0  B0 0009C          MOVW    24(R0), 24(R2)                           ;
                               50      04    AC  D0 000A1          MOVL    CONTEXT, R0                              ; 1598
                         1A    A2      1A    A0  B0 000A5          MOVW    26(R0), 26(R2)                           ;
                               50      04    AC  D0 000AA          MOVL    CONTEXT, R0                              ; 1599
            1C    A2     1C    A0    0051    8F  28 000AE          MOVC3   #81, 28(R0), 28(R2)                      ;
                                             04  000B6             RET                                              ; 1576
                 62            60    0070    8F  28 000B7 8$:      MOVC3   #112, (R0), (R2)                         ; 1602
                                             04  000BD             RET                                              ; 1604

; Routine Size:  190 bytes,    Routine Base:  $CODE$ + 04BD


;  618          1605  1
;  619          1606  1 END
;  620          1607  0 ELUDOM




;                                 PSECT SUMMARY
;
;       Name                          Bytes                      Attributes
;
;    $CODE$                                1403  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        51          0        1000           00:01.9
;                               COMMAND QUALIFIERS

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

; Size:          1399 code + 4 data bytes
; Run Time:         00:51.8
; Elapsed Time:     01:35.9
; Lines/CPU Min:      1861
; Lexemes/CPU-Min: 47008
; Memory Used:  377 pages
; Compilation Complete
```

**F

0169 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

DEACCS
LIS

DELETE
LIS

DIRSCN
LIS

DISPAT
LIS

CREHDR
LIS

DIRACC
LIS

DELBAD
LIS

CREFCB
LIS

CREWIN
LIS

DISPATCH
LIS

ENTER
LIS

DELFIL
LIS

[illegible]

0170 AH-BT13A-SE DIGITAL EQUIPMENT CORPORATION
VAX/VMS V4.0 CONFIDENTIAL AND PROPRIETARY

EXTIDX
LIS

GTLCAT
LIS

GETLOC
LIS

EXTEND
LIS

EXTHDR
LIS

FILUTL
LIS

GETREQ
LIS

EXTCONTIG
LIS

ERASE
LIS

GETTIM
LIS

FIND
LIS

GETFIB
LIS

INIFC2
LIS

INIFCP
LIS

EXTFCB
LIS

FILESERV
LIS

FILESIZE
LIS

GETPTR
LIS