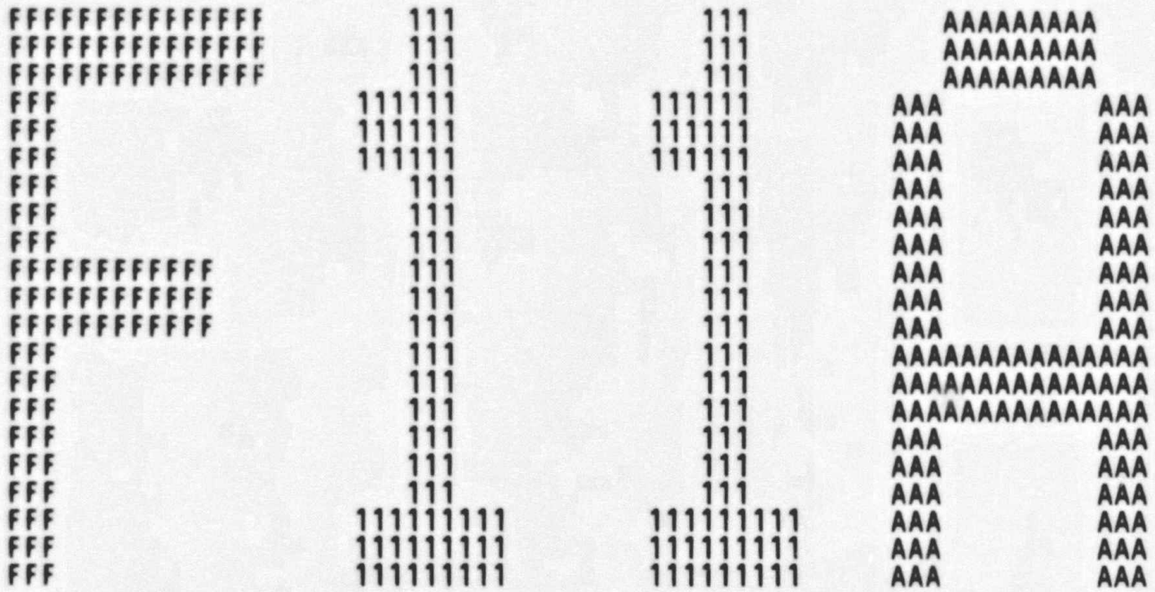
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
AAAAAAAAA
AAAAAAAAA
AAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA

FILEID**FIND

```
FFFFFFFFFF    IIIIII    NN      NN  DDDDDDDD
FFFFFFFFFF    IIIIII    NN      NN  DDDDDDDD
FF              II      NN      NN  DD      DD
FF              II      NN      NN  DD      DD
FF              II      NNNN    NN  DD      DD
FF              II      NNNN    NN  DD      DD
FFFFFFFF        II      NN  NN  NN  DD      DD
FFFFFFFF        II      NN  NN  NN  DD      DD
FF              II      NN    NNNN  DD      DD
FF              II      NN    NNNN  DD      DD
FF              II      NN      NN  DD      DD      ....
FF              II      NN      NN  DD      DD      ....
FF            IIIIII    NN      NN  DDDDDDDD        ....
FF            IIIIII    NN      NN  DDDDDDDD        ....

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

```
;      1     0001  0 MODULE FIND (
;      2     0002  0                 LANGUAGE (BLISS32),
;      3     0003  0                 IDENT = 'V04-000'
;      4     0004  0                 ) =
;      5     0005  1 BEGIN
;      6     0006  1
;      7     0007  1 !
;      8     0008  1 !*****************************************************************************
;      9     0009  1 !*                                                                           *
;     10     0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     11     0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;     12     0012  1 !*  ALL RIGHTS RESERVED.                                                      *
;     13     0013  1 !*                                                                           *
;     14     0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;     15     0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;     16     0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;     17     0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;     18     0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;     19     0019  1 !*  TRANSFERRED.                                                              *
;     20     0020  1 !*                                                                           *
;     21     0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;     22     0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;     23     0023  1 !*  CORPORATION.                                                              *
;     24     0024  1 !*                                                                           *
;     25     0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;     26     0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;     27     0027  1 !*                                                                           *
;     28     0028  1 !*                                                                           *
;     29     0029  1 !*****************************************************************************
;     30     0030  1
;     31     0031  1 !++
;     32     0032  1 !
;     33     0033  1 ! FACILITY:  F11ACP Structure Level 1
;     34     0034  1 !
;     35     0035  1 ! ABSTRACT:
;     36     0036  1 !
;     37     0037  1 !       This routine performs a find operation on the indicated directory.
;     38     0038  1 !
;     39     0039  1 ! ENVIRONMENT:
;     40     0040  1 !
;     41     0041  1 !       STARLET operating system, including privileged system services
;     42     0042  1 !       and internal exec routines.
;     43     0043  1 !
;     44     0044  1 !--
;     45     0045  1 !
;     46     0046  1 !
;     47     0047  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  2-Jan-1977  22:11
;     48     0048  1 !
;     49     0049  1 ! MODIFIED BY:
;     50     0050  1 !
;     51     0051  1 !       V03-001 LMP0241         L. Mark Pilant,         26-Apr-1984  10:45
;     52     0052  1 !               Include the FIB in the MAKE_NAMEBLOCK routine call.
;     53     0053  1 !
;     54     0054  1 !       V02-004 ACG0238         Andrew C. Goldstein,    6-Jan-1982  13:11
;     55     0055  1 !               Make UPDATE_DIRSEQ global
;     56     0056  1 !
;     57     0057  1 !       V02-003 ACG0167         Andrew C. Goldstein,    7-May-1980  18:50
```

```
;    58          0058  1 !                       Previous revision history moved to F11A.REV
;    59          0059  1 !**
;    60          0060  1
;    61          0061  1
;    62          0062  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    63          0063  1 REQUIRE 'SRC$:FCPDEF.B32';
;    64          0378  1
;    65          0379  1
;    66          0380  1 FORWARD ROUTINE
;    67          0381  1         FIND                : NOVALUE,      ! find directory entry
;    68          0382  1         UPDATE_DIRSEQ;                      ! bump UCB directory sequence count
```

```
;    70          0383  1 GLOBAL ROUTINE FIND (ABD, FIB, REMOVE) : NOVALUE =
;    71          0384  1
;    72          0385  1 !++
;    73          0386  1 !
;    74          0387  1 ! FUNCTIONAL DESCRIPTION:
;    75          0388  1 !
;    76          0389  1 !       This routine performs a FIND operation on the indicated directory.
;    77          0390  1 !
;    78          0391  1 ! CALLING SEQUENCE:
;    79          0392  1 !       FIND (ARG1, ARG2, ARG3)
;    80          0393  1 !
;    81          0394  1 ! INPUT PARAMETERS:
;    82          0395  1 !       ARG1: address of descriptor list in buffer packet
;    83          0396  1 !       ARG2: address of FIB
;    84          0397  1 !       ARG3: 0 to just do a find
;    85          0398  1 !             1 to remove the found entry
;    86          0399  1 !
;    87          0400  1 ! IMPLICIT INPUTS:
;    88          0401  1 !       NONE
;    89          0402  1 !
;    90          0403  1 ! OUTPUT PARAMETERS:
;    91          0404  1 !       ARG2: file ID and context returned in FIB
;    92          0405  1 !
;    93          0406  1 ! IMPLICIT OUTPUTS:
;    94          0407  1 !       DIR_RECORD: record number of found directory entry
;    95          0408  1 !
;    96          0409  1 ! ROUTINE VALUE:
;    97          0410  1 !       NONE
;    98          0411  1 !
;    99          0412  1 ! SIDE EFFECTS:
;   100          0413  1 !       Directory LRU may be altered
;   101          0414  1 !       directory blocks read
;   102          0415  1 !
;   103          0416  1 !--
;   104          0417  1
;   105          0418  2 BEGIN
;   106          0419  2
;   107          0420  2 MAP
;   108          0421  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH],
;   109          0422  2                                         ! descriptor list arg
;   110          0423  2         FIB             : REF BBLOCK;   ! FIB argument
;   111          0424  2
;   112          0425  2 LOCAL
;   113          0426  2         NAMEBLOCK       : BBLOCK [NMB$C_LENGTH],
;   114          0427  2                                         ! RAD-50 name block
;   115          0428  2         OLD_NAMEBLOCK   : BBLOCK [NMB$C_LENGTH],
;   116          0429  2                                         ! another RAD-50 name block
;   117          0430  2         COMPLEX,                        ! complex operation flag
;   118          0431  2         SAVED_FLAGS,                    ! original flag bits for operation
;   119          0432  2         SAVED_CONTEXT,                  ! record number of first occurrence
;   120          0433  2         STRING          : VECTOR [20, BYTE],
;   121          0434  2                                         ! space for result string
;   122          0435  2         COUNT,                          ! count of name string
;   123          0436  2         STRINGP,                        ! address of name string
;   124          0437  2         RECADDR         : REF BBLOCK;   ! pointer to found directory entry
;   125          0438  2
;   126          0439  2 EXTERNAL
```

```
;  127     0440  2          CLEANUP_FLAGS   : BITVECTOR,    ! cleanup action flags
;  128     0441  2          DIR_RECORD,                     ! record number of directory entry
;  129     0442  2          SUPER_FID       : BBLOCK;       ! file ID of superseded/deleted file
;  130     0443  2
;  131     0444  2      EXTERNAL ROUTINE
;  132     0445  2          PMS_START_SUB,                  ! start subfunction metering
;  133     0446  2          PMS_END_SUB,                    ! end subfunction metering
;  134     0447  2          COPY_NAME,                      ! copy file name to result string
;  135     0448  2          DIR_ACCESS,                     ! access the directory
;  136     0449  2          MAKE_NAMEBLOCK,                 ! construct name block
;  137     0450  2          DIR_SCAN,                       ! search the directory
;  138     0451  2          MAKE_STRING,                    ! convert directory entry to string
;  139     0452  2          RETURN_DIR,                     ! return data to buffer packet
;  140     0453  2          DIRPUT;                         ! write directory record
;  141     0454  2
;  142     0455  2
;  143     0456  2      ! Start metering for this subfunction.
;  144     0457  2      !
;  145     0458  2
;  146     0459  2      PMS_START_SUB (PMS_FIND);
;  147     0460  2
;  148     0461  2      ! If this is an operation on a spooled device, noop it and return a file ID
;  149     0462  2      ! of -2, -2 with success.
;  150     0463  2      !
;  151     0464  2
;  152     0465  2      IF .CLEANUP_FLAGS[CLF_SPOOLFILE]
;  153     0466  2      THEN
;  154     0467  3          BEGIN
;  155     0468  3          FIB[FIB$W_FID_NUM] = -2;
;  156     0469  3          FIB[FIB$W_FID_SEQ] = -2;
;  157     0470  3          FIB[FIB$W_FID_RVN] = 0;
;  158     0471  3          KERNEL_CALL (COPY_NAME, .ABD);
;  159     0472  3          RETURN;
;  160     0473  3          END;
;  161     0474  2
;  162     0475  2      ! Find the name string in the buffer packet. Construct the RAD-50 name block
;  163     0476  2      ! from it and the other data in the FIB.
;  164     0477  2      !
;  165     0478  2
;  166     0479  2      COUNT = .ABD[ABD$C_NAME, ABD$W_COUNT];
;  167     0480  2      STRINGP = .ABD[ABD$C_NAME, ABD$W_TEXT] + ABD[ABD$C_NAME, ABD$W_TEXT] + 1;
;  168     0481  2
;  169     0482  2      MAKE_NAMEBLOCK (.FIB, .COUNT, .STRINGP, NAMEBLOCK);
;  170     0483  2      CH$MOVE (FIB$S_FID, FIB[FIB$W_FID], NAMEBLOCK[NMB$W_FID]);
;  171     0484  2      NAMEBLOCK[NMB$W_FLAGS] = .NAMEBLOCK[NMB$W_FLAGS] OR .FIB[FIB$W_NMCTL];
;  172     0485  2      NAMEBLOCK[NMB$W_CONTEXT] = .FIB[FIB$L_WCC];
;  173     0486  2
;  174     0487  2      ! Determine if this is a complex operation (wild card name, default version).
;  175     0488  2      ! If so, set up the required parameters.
;  176     0489  2      !
;  177     0490  2
;  178     0491  2      COMPLEX = 0;
;  179     0492  3      IF (.NAMEBLOCK[NMB$V_ALLNAM] OR .NAMEBLOCK[NMB$V_ALLTYP])
;  180     0493  2      AND NOT .NAMEBLOCK[NMB$V_ALLVER]
;  181     0494  2      AND .NAMEBLOCK[NMB$W_VERSION] EQL 0
;  182     0495  2      THEN
;  183     0496  3          BEGIN
```

```
;  184     0497  3      COMPLEX = 1;
;  185     0498  3      NAMEBLOCK[NMB$V_ALLVER] = 1;
;  186     0499  3      SAVED_FLAGS = .NAMEBLOCK[NMB$W_FLAGS];
;  187     0500  2      END;
;  188     0501  2
;  189     0502  2  ! Access the directory. We need write access if this is a REMOVE.
;  190     0503  2  !
;  191     0504  2
;  192     0505  2  DIR_ACCESS (.FIB, .REMOVE);
;  193     0506  2
;  194     0507  2  ! If this is a wild card operation (i.e., if the wild card context is nonzero),
;  195     0508  2  ! position to the indicated record. This is done with the positional context
;  196     0509  2  ! and the supplied resultant name string, if any.
;  197     0510  2  !
;  198     0511  2
;  199     0512  2  IF .FIB[FIB$L_WCC] NEQ 0
;  200     0513  2  THEN
;  201     0514  3      BEGIN
;  202     0515  3      COUNT = MINU (.ABD[ABD$C_RES, ABD$W_COUNT],
;  203     0516  3                    IF .ABD[ABD$C_RESL, ABD$W_COUNT] GEQU 2
;  204     0517  4                    THEN .(ABD[ABD$C_RESL, ABD$W_TEXT] +
;  205     0518  3                           .ABD[ABD$C_RESL, ABD$W_TEXT] + 1)<0,16>
;  206     0519  3                    ELSE 0
;  207     0520  3                   );
;  208     0521  3      IF .COUNT NEQ 0
;  209     0522  3      THEN
;  210     0523  4          BEGIN
;  211     0524  4          STRINGP = .ABD[ABD$C_RES, ABD$W_TEXT] + ABD[ABD$C_RES, ABD$W_TEXT] + 1;
;  212     0525  4          MAKE_NAMEBLOCK (0, .COUNT, .STRINGP, OLD_NAMEBLOCK);
;  213     0526  4          IF .OLD_NAMEBLOCK[NMB$V_WILD]
;  214     0527  4          THEN ERR_EXIT (SS$_BADFILENAME);
;  215     0528  4          OLD_NAMEBLOCK[NMB$W_CONTEXT] = .FIB[FIB$L_WCC] - 1;
;  216     0529  4          IF .COMPLEX THEN OLD_NAMEBLOCK[NMB$V_ALLVER] = 1;
;  217     0530  4
;  218     0531  4          IF DIR_SCAN (OLD_NAMEBLOCK, 0) EQL 0
;  219     0532  4          THEN
;  220     0533  5              BEGIN
;  221     0534  5              OLD_NAMEBLOCK[NMB$W_CONTEXT] = 0;
;  222     0535  5              IF DIR_SCAN (OLD_NAMEBLOCK, 0) NEQ 0
;  223     0536  5              THEN NAMEBLOCK[NMB$W_CONTEXT] = .DIR_RECORD;
;  224     0537  5              END
;  225     0538  4          ELSE
;  226     0539  4              NAMEBLOCK[NMB$W_CONTEXT] = .DIR_RECORD;
;  227     0540  3          END;
;  228     0541  2      END;
;  229     0542  2
;  230     0543  2  ! Now search the directory. If this is a complex operation, we may
;  231     0544  2  ! loop several times before finding a suitable entry.
;  232     0545  2  !
;  233     0546  2
;  234     0547  2  WHILE 1 DO
;  235     0548  3      BEGIN
;  236     0549  3      RECADDR = DIR_SCAN (NAMEBLOCK, 0);
;  237     0550  3      IF .RECADDR EQL 0                       ! if not found
;  238     0551  3      THEN
;  239     0552  3          IF .FIB[FIB$L_WCC] EQL 0
;  240     0553  4          THEN ERR_EXIT (SS$_NOSUCHFILE)
```

```
;  241     0554  3          ELSE
;  242     0555  4              BEGIN
;  243     0556  4              FIB[FIB$L_WCC] = 0;
;  244     0557  4              ERR_EXIT (SS$_NOMOREFILES);
;  245     0558  3              END;
;  246     0559  3      SAVED_CONTEXT = .DIR_RECORD;          ! save record number found
;  247     0560  3      IF NOT .COMPLEX THEN EXITLOOP;        ! for simple lookups that's all
;  248     0561  3
;  249     0562  3  ! This is a complex operation. We have now found some occurrence of the desired
;  250     0563  3  ! file. Now search from the beginning of the directory for the first occurring
;  251     0564  3  ! version of this entry. If it precedes the one found, then we have already
;  252     0565  3  ! found it on a previous call, so try from the start for something new. If
;  253     0566  3  ! it follows the one found, we are in trouble.
;  254     0567  3  !
;  255     0568  3
;  256     0569  3      CH$MOVE (NMB$C_DIRENTRY-2, .RECADDR, NAMEBLOCK);
;  257     0570  3      NAMEBLOCK[NMB$W_FLAGS] = NMB$M_ALLVER;
;  258     0571  3      NAMEBLOCK[NMB$W_CONTEXT] = 0;
;  259     0572  3
;  260     0573  3      RECADDR = DIR_SCAN (NAMEBLOCK, 0);
;  261     0574  3      IF .RECADDR EQL 0 OR .DIR_RECORD GTRU .SAVED_CONTEXT
;  262     0575  3      THEN ERR_EXIT (SS$_BADIRECTORY);
;  263     0576  3
;  264     0577  3  ! If the entry found on the second scan is the same as the first, then this is
;  265     0578  3  ! a first occurrence. Now scan the remainder of the directory for the latest
;  266     0579  3  ! version of this occurrence. We will return the latest version entry to
;  267     0580  3  ! the caller, but the context of the first occurrence found.
;  268     0581  3  !
;  269     0582  3
;  270     0583  3      IF .DIR_RECORD EQL .SAVED_CONTEXT
;  271     0584  3      THEN
;  272     0585  4          BEGIN
;  273     0586  4          NAMEBLOCK[NMB$W_FLAGS] = 0;
;  274     0587  4          NAMEBLOCK[NMB$W_CONTEXT] = .SAVED_CONTEXT - 1;
;  275     0588  4          RECADDR = DIR_SCAN (NAMEBLOCK, 0);
;  276     0589  4          IF .RECADDR EQL 0
;  277     0590  4          THEN BUG_CHECK (DIRENTRY, FATAL, 'ACP failed to find same directory entry');
;  278     0591  4          EXITLOOP;
;  279     0592  3          END;
;  280     0593  3
;  281     0594  3  ! This was a duplicate. Restore the original flags and continue the basic
;  282     0595  3  ! scan from where we left off.
;  283     0596  3  !
;  284     0597  3
;  285     0598  3      NAMEBLOCK[NMB$W_FLAGS] = .SAVED_FLAGS;
;  286     0599  3      NAMEBLOCK[NMB$W_CONTEXT] = .SAVED_CONTEXT;
;  287     0600  2      END;                                  ! end of directory scan loop
;  288     0601  2
;  289     0602  2  ! Now convert the resulting directory entry into the ASCII resultant string.
;  290     0603  2  ! Return this and the record number to the buffer packet.
;  291     0604  2  !
;  292     0605  2
;  293     0606  2  CH$MOVE (FIB$S_FID, RECADDR[NMB$W_FID], FIB[FIB$W_FID]);
;  294     0607  2  FIB[FIB$L_WCC] =
;  295     0608  3      (
;  296     0609  3      IF .FIB[FIB$V_WILD]
;  297     0610  3      THEN .SAVED_CONTEXT
```

```
3       ELSE 0
2       );
2   COUNT = MAKE_STRING (.RECADDR, STRING);
2   KERNEL_CALL (RETURN_DIR, .COUNT, STRING, .ABD);
2
2   ! If a remove operation is requested, do so by simply zeroing the file number
2   ! word of the directory entry and writing it back. Note that we save the
2   ! removed FID for replacement in case of subsequent error.
2   ! If the entry being removed is of the form xxx.DIR;1, bump the sequence count
2   ! in the UCB to invalidate RMS caches.
2   !
2
2   IF .REMOVE
2   THEN
3       BEGIN
3       CH$MOVE (FIB$S_FID, RECADDR[NMB$W_FID],SUPER_FID);
3       RECADDR[NMB$W_FID_NUM] = 0;
3       DIRPUT (.RECADDR);
3       CLEANUP_FLAGS[CLF_REENTER] = 1;
3
3       IF   .RECADDR[NMB$W_TYPE] EQL %RAD50_11 'DIR'
3       AND .RECADDR[NMB$W_VERSION] EQL 1
3       THEN KERNEL_CALL (UPDATE_DIRSEQ);
2       END;
2
2   ! Stop metering of this subfunction
2   !
2
2   PMS_END_SUB ();
2
1   END;                                          ! end of routine FIND


                                                  .TITLE   FIND
                                                  .IDENT   \V04-000\

                                                  .EXTRN   CLEANUP_FLAGS, DIR_RECORD
                                                  .EXTRN   SUPER_FID, PMS_START_SUB
                                                  .EXTRN   PMS_END_SUB, COPY_NAME
                                                  .EXTRN   DIR_ACCESS, MAKE_NAMEBLOCK
                                                  .EXTRN   DIR_SCAN, MAKE_STRING
                                                  .EXTRN   RETURN_DIR, DIRPUT
                                                  .EXTRN   SYS$CMKRNL, BUG$_DIRENTRY

                                                  .PSECT   $CODE$,NOWRT,2

                           0FFC 00000             .ENTRY   FIND, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11   ; 0383
          5E       9C   AE   9E 00002             MOVAB    -100(SP), SP                                  ;
                        06   DD 00006             PUSHL    #6                                            ; 0459
    0000G CF            01   FB 00008             CALLS    #1, PMS_START_SUB                             ;
                 0000G  CF   95 0000D             TSTB     CLEANUP_FLAGS                                 ; 0465
                        22   18 00011             BGEQ     1$                                            ;
          50       08   AC   D0 00013             MOVL     FIB, R0                                       ; 0468
     04   A0 FFFEFFFE   8F   D0 00017             MOVL     #-65538, 4(R0)                                ;
                   08   A0   B4 0001F             CLRW     8(R0)                                         ; 0470
                   04   AC   DD 00022             PUSHL    ABD                                           ; 0471
                        01   DD 00025             PUSHL    #1                                            ;
```

```
                                  5E   DD 00027         PUSHL   SP                                   ;
                         0000G    CF   9F 00029         PUSHAB  COPY_NAME                            ;
          00000000G  9F           04   FB 0002D         CALLS   #4, @#SYS$CMKRNL                     ;
                                       04 00034         RET                                          ; 0467
                  56         04   AC   7D 00035  1$:    MOVQ    ABD, R6                              ; 0479
                  59         12   A6   3C 00039         MOVZWL  18(R6), COUNT                        ;
                  51         10   A6   9E 0003D         MOVAB   16(R6), R1                           ; 0480
                  50              61   3C 00041         MOVZWL  (R1), R0                             ;
                  58      01 A140      9E 00044         MOVAB   1(R1)[R0], STRINGP                   ;
                             3C   AE   9F 00049         PUSHAB  NAMEBLOCK                            ; 0482
                                  58   DD 0004C         PUSHL   STRINGP                              ;
                         0280     8F   BB 0004E         PUSHR   #^M<R7,R9>                           ;
              0000G  CF           04   FB 00052         CALLS   #4, MAKE_NAMEBLOCK                   ;
3C    AE         04  A7           06   28 00057         MOVC3   #6, 4(R7), NAMEBLOCK                 ; 0483
                 4C  AE      14   A7   A8 0005D         BISW2   20(R7), NAMEBLOCK+16                 ; 0484
                 62  AE      10   A7   B0 00062         MOVW    16(R7), NAMEBLOCK+38                 ; 0485
                                  5A   D4 00067         CLRL    COMPLEX                              ; 0491
      05         4C  AE           05   E0 00069         BBS     #5, NAMEBLOCK+16, 2$                 ; 0492
      15         4C  AE           04   E1 0006E         BBC     #4, NAMEBLOCK+16, 3$                 ;
      10         4C  AE           03   E0 00073  2$:    BBS     #3, NAMEBLOCK+16, 3$                 ; 0493
                             4A   AE   B5 00078         TSTW    NAMEBLOCK+14                         ; 0494
                                  0B   12 0007B         BNEQ    3$                                   ;
                     5A           01   D0 0007D         MOVL    #1, COMPLEX                          ; 0497
                 4C  AE           08   88 00080         BISB2   #8, NAMEBLOCK+16                     ; 0498
                     5B      4C   AE   3C 00084         MOVZWL  NAMEBLOCK+16, SAVED_FLAGS            ; 0499
                             0C   AC   DD 00088  3$:    PUSHL   REMOVE                               ; 0505
                                  57   DD 0008B         PUSHL   R7                                   ;
              0000G  CF           02   FB 0008D         CALLS   #2, DIR_ACCESS                       ;
                             10   A7   D5 00092         TSTL    16(R7)                               ; 0512
                                  7F   13 00095         BEQL    10$                                  ;
                     02      1A   A6   B1 00097         CMPW    26(R6), #2                           ; 0516
                                  0D   1F 0009B         BLSSU   4$                                   ;
                     50      18   A6   3C 0009D         MOVZWL  24(R6), R0                           ; 0518
                          19 A046      9F 000A1         PUSHAB  25(R0)[R6]                           ; 0517
                     50           9E   3C 000A5         MOVZWL  @(SP)+, R0                           ;
                                  02   11 000A8         BRB     5$                                   ;
                                  50   D4 000AA  4$:    CLRL    R0                                   ; 0516
                     51      22   A6   3C 000AC  5$:    MOVZWL  34(R6), R1                           ;
                     50           51   D1 000B0         CMPL    R1, R0                               ;
                                  03   1B 000B3         BLEQU   6$                                   ;
                     51           50   D0 000B5         MOVL    R0, R1                               ;
                     59           51   D0 000B8  6$:    MOVL    R1, COUNT                            ; 0515
                                  59   13 000BB         BEQL    10$                                  ; 0521
                     51      20   A6   9E 000BD         MOVAB   32(R6), R1                           ; 0524
                     50           61   3C 000C1         MOVZWL  (R1), R0                             ;
                     58   01 A140      9E 000C4         MOVAB   1(R1)[R0], STRINGP                   ;
                             14   AE   9F 000C9         PUSHAB  OLD_NAMEBLOCK                        ; 0525
                                  58   DD 000CC         PUSHL   STRINGP                              ;
                                  59   DD 000CE         PUSHL   COUNT                                ;
                                  7E   D4 000D0         CLRL    -(SP)                                ;
              0000G  CF           04   FB 000D2         CALLS   #4, MAKE_NAMEBLOCK                   ;
                     05      25   AE   E9 000D7         BLBC    OLD_NAMEBLOCK+17, 7$                 ; 0526
                           0818   8F   BF 000DB         CHMU    #2072                                ; 0527
                                       04 000DF         RET                                          ;
                     50      08   AC   D0 000E0  7$:    MOVL    FIB, R0                              ; 0528
3A    AE         10  A0           01   A3 000E4         SUBW3   #1, 16(R0), OLD_NAMEBLOCK+38         ;
                     04           5A   E9 000EA         BLBC    COMPLEX, 8$                          ; 0529
```

```
            24    AE            08   88 000ED       BISB2   #8, OLD_NAMEBLOCK+16                           ;
                                7E   D4 000F1 8$:   CLRL    -(SP)                                           ; 0531
                          18    AE   9F 000F3       PUSHAB  OLD_NAMEBLOCK                                   ;
                 0000G    CF    02   FB 000F6       CALLS   #2, DIR_SCAN                                    ;
                                50   D5 000FB       TSTL    R0                                              ;
                                11   12 000FD       BNEQ    9$                                              ;
                          3A    AE   B4 000FF       CLRW    OLD_NAMEBLOCK+38                                ; 0534
                                7E   D4 00102       CLRL    -(SP)                                           ; 0535
                          18    AE   9F 00104       PUSHAB  OLD_NAMEBLOCK                                   ;
                 0000G    CF    02   FB 00107       CALLS   #2, DIR_SCAN                                    ;
                                50   D5 0010C       TSTL    R0                                              ;
                                06   13 0010E       BEQL    10$                                             ;
            62    AE    0000G   CF   B0 00110 9$:   MOVW    DIR_RECORD, NAMEBLOCK+38                        ; 0539
                                7E   D4 00116 10$:  CLRL    -(SP)                                           ; 0549
                          40    AE   9F 00118       PUSHAB  NAMEBLOCK                                       ;
                 0000G    CF    02   FB 0011B       CALLS   #2, DIR_SCAN                                    ;
                          58    50   D0 00120       MOVL    R0, RECADDR                                     ;
                                16   12 00123       BNEQ    12$                                             ; 0550
                  50      08    AC   D0 00125       MOVL    FIB, R0                                         ; 0552
                          10    A0   D5 00129       TSTL    16(R0)                                          ;
                                05   12 0012C       BNEQ    11$                                             ;
                        0910    8F   BF 0012E       CHMU    #2320                                           ; 0553
                                     04 00132       RET                                                     ; 0555
                          10    A0   D4 00133 11$:  CLRL    16(R0)                                          ; 0556
                        0930    8F   BF 00136       CHMU    #2352                                           ; 0557
                                     04 0013A       RET                                                     ; 0555
                  56    0000G   CF   D0 0013B 12$:  MOVL    DIR_RECORD, SAVED_CONTEXT                       ; 0559
                  50            5A   E9 00140       BLBC    COMPLEX, 16$                                    ; 0560
      3C    AE    68            0E   28 00143       MOVC3   #14, (RECADDR), NAMEBLOCK                       ; 0569
            4C    AE            08   B0 00148       MOVW    #8, NAMEBLOCK+16                                ; 0570
                          62    AE   B4 0014C       CLRW    NAMEBLOCK+38                                    ; 0571
                                7E   D4 0014F       CLRL    -(SP)                                           ; 0573
                          40    AE   9F 00151       PUSHAB  NAMEBLOCK                                       ;
                 0000G    CF    02   FB 00154       CALLS   #2, DIR_SCAN                                    ;
                          58    50   D0 00159       MOVL    R0, RECADDR                                     ;
                                07   13 0015C       BEQL    13$                                             ; 0574
                  56    0000G   CF   D1 0015E       CMPL    DIR_RECORD, SAVED_CONTEXT                       ;
                                05   1B 00163       BLEQU   14$                                             ;
                        0828    8F   BF 00165 13$:  CHMU    #2088                                           ; 0575
                                     04 00169       RET                                                     ;
                                1D   12 0016A 14$:  BNEQ    15$                                             ; 0583
                          4C    AE   B4 0016C       CLRW    NAMEBLOCK+16                                    ; 0586
      62    AE    56            01   A3 0016F       SUBW3   #1, SAVED_CONTEXT, NAMEBLOCK+38                 ; 0587
                                7E   D4 00174       CLRL    -(SP)                                           ; 0588
                          40    AE   9F 00176       PUSHAB  NAMEBLOCK                                       ;
                 0000G    CF    02   FB 00179       CALLS   #2, DIR_SCAN                                    ;
                          58    50   D0 0017E       MOVL    R0, RECADDR                                     ;
                                10   12 00181       BNEQ    16$                                             ; 0589
                                   FEFF 00183       BUGW                                                    ; 0590
                                  0000* 00185       .WORD   <BUG$_DIRENTRY!4>                               ;
                                0A   11 00187       BRB     16$                                             ; 0585
            4C    AE            5B   B0 00189 15$:  MOVW    SAVED_FLAGS, NAMEBLOCK+16                       ; 0598
            62    AE            56   B0 0018D       MOVW    SAVED_CONTEXT, NAMEBLOCK+38                     ; 0599
                                83   11 00191       BRB     10$                                             ; 0547
                  57      08    AC   D0 00193 16$:  MOVL    FIB, R7                                         ; 0606
      04    A7    68            06   28 00197       MOVC3   #6, (RECADDR), 4(R7)                            ;
                  02      15    A7   E8 0019C       BLBS    21(R7), 17$                                     ; 0609
```

```
                                       56  D4 001A0        CLRL    R6                                  ;
               10   A7                 56  D0 001A2  17$:  MOVL    R6, 16(R7)                          ; 0608
                             4100      8F  BB 001A6        PUSHR   #^M<R8,SP>                          ; 0613
            0000G   CF                 02  FB 001AA        CALLS   #2, MAKE_STRING                     ;
                    59                 50  D0 001AF        MOVL    R0, COUNT                           ;
                               04      AC  DD 001B2        PUSHL   ABD                                 ; 0614
                               04      AE  9F 001B5        PUSHAB  STRING                              ;
                                       59  DD 001B8        PUSHL   COUNT                               ;
                                       03  DD 001BA        PUSHL   #3                                  ;
                                       5E  DD 001BC        PUSHL   SP                                  ;
                             0000G     CF  9F 001BE        PUSHAB  RETURN_DIR                          ;
        00000000G   9F                 06  FB 001C2        CALLS   #6, @#SYS$CMKRNL                    ;
                    32         0C      AC  E9 001C9        BLBC    REMOVE, 18$                         ; 0623
   0000G  CF        68                 06  28 001CD        MOVC3   #6, (RECADDR), SUPER_FID            ; 0626
                                       68  B4 001D3        CLRW    (RECADDR)                           ; 0627
                                       58  DD 001D5        PUSHL   RECADDR                             ; 0628
            0000G   CF                 01  FB 001D7        CALLS   #1, DIRPUT                          ;
            0000G   CF         80      8F  88 001DC        BISB2   #128, CLEANUP_FLAGS+2               ; 0629
             1A7A   8F         0C      A8  B1 001E2        CMPW    12(RECADDR), #6778                  ; 0631
                                       15  12 001E8        BNEQ    18$                                 ;
                    01         0E      A8  B1 001EA        CMPW    14(RECADDR), #1                     ; 0632
                                       0F  12 001EE        BNEQ    18$                                 ;
                                       7E  D4 001F0        CLRL    -(SP)                               ; 0633
                                       5E  DD 001F2        PUSHL   SP                                  ;
                             0000V     CF  9F 001F4        PUSHAB  UPDATE_DIRSEQ                       ;
        00000000G   9F                 03  FB 001F8        CALLS   #3, @#SYS$CMKRNL                    ;
            0000G   CF                 00  FB 001FF  18$:  CALLS   #0, PMS_END_SUB                     ; 0639
                                           04 00204        RET                                         ; 0641

; Routine Size:  517 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   330     0642  1 GLOBAL ROUTINE UPDATE_DIRSEQ =
;   331     0643  1
;   332     0644  1 !++
;   333     0645  1 !
;   334     0646  1 ! FUNCTIONAL DESCRIPTION:
;   335     0647  1 !
;   336     0648  1 !       This routine bumps the directory sequence count in the UCB to invalidate
;   337     0649  1 !       RMS directory caches on the volume.
;   338     0650  1 !
;   339     0651  1 !
;   340     0652  1 ! CALLING SEQUENCE:
;   341     0653  1 !       UPDATE_DIRSEQ ()
;   342     0654  1 !
;   343     0655  1 ! INPUT PARAMETERS:
;   344     0656  1 !       NONE
;   345     0657  1 !
;   346     0658  1 ! IMPLICIT INPUTS:
;   347     0659  1 !       CURRENT_UCB: UCB of device in use
;   348     0660  1 !
;   349     0661  1 ! OUTPUT PARAMETERS:
;   350     0662  1 !       NONE
;   351     0663  1 !
;   352     0664  1 ! IMPLICIT OUTPUTS:
;   353     0665  1 !       directory sequence count incremented
;   354     0666  1 !
;   355     0667  1 ! ROUTINE VALUE:
;   356     0668  1 !       1
;   357     0669  1 !
;   358     0670  1 ! SIDE EFFECTS:
;   359     0671  1 !       NONE
;   360     0672  1 !
;   361     0673  1 !--
;   362     0674  1
;   363     0675  2 BEGIN
;   364     0676  2
;   365     0677  2 EXTERNAL
;   366     0678  2         CURRENT_UCB     : REF BBLOCK;   ! UCB of device
;   367     0679  2
;   368     0680  2
;   369     0681  2 CURRENT_UCB[UCB$W_DIRSEQ] = .CURRENT_UCB[UCB$W_DIRSEQ] + 1;
;   370     0682  2
;   371     0683  2 RETURN 1;
;   372     0684  2
;   373     0685  1 END;                                    ! end of routine UPDATE_DIRSEQ


                                                        .EXTRN  CURRENT_UCB

                                        0000 00000      .ENTRY  UPDATE_DIRSEQ, Save nothing             ; 0642
                              50  0000G  CF   D0 00002  MOVL    CURRENT_UCB, R0                         ; 0681
                                  00AC  C0   B6 00007  INCW    172(R0)                                 ;
                              50        01   D0 0000B  MOVL    #1, R0                                  ; 0683
                                             04 0000E  RET                                             ; 0685

; Routine Size:  15 bytes,    Routine Base:  $CODE$ + 0205
```

```
;  374          0686  1
;  375          0687  1 END
;  376          0688  0 ELUDOM



;                               PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $CODE$                        532  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        33         0        1000        00:01.9



;                                       COMMAND QUALIFIERS

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

; Size:          532 code + 0 data bytes
; Run Time:          00:14.3
; Elapsed Time:      00:44.1
; Lines/CPU Min:      2884
; Lexemes/CPU-Min:  15949
; Memory Used:   195 pages
; Compilation Complete
```

EXTFCB.LIS

DELFIL.LIS

DIRGET.LIS

EXTIDX.LIS

IODONE.LIS

LOCKDN.LIS

ENTER.LIS

GETREQ.LIS

GETTIM.LIS

DISPAT.LIS

INIFCP.LIS

DIRFCB.LIS

EXTHDR.LIS

DIRSCN.LIS

LOGDEL.LIS

LOCKDB.LIS

FIND.LIS

GETFIB.LIS

DIRACC.LIS

INIFCB.LIS

EXTDIR.LIS

EXTEND.LIS