```
FFFFFFFFFFFFFFFF        111          111     XXX         XXX
FFFFFFFFFFFFFFFF        111          111     XXX         XXX
FFFFFFFFFFFFFFFF        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
FFFFFFFF.FFF            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_H
LB_L
LOCA
LOCA
LOCK

LOCK
LOCK
LOCK
LOC_
LOC_
L_CC
L_CC
L_DA
L_DA
MAIN
MAKE
MAKE
MAKE
MAKE
MAKE
MAKE

MAKE
MAKE
MAP_
MAP_

MAP
MARK
MARK
MARK
MARK
MARK
```

FILEID**MAKACC

```
MM      MM    AAAAAA    KK      KK    AAAAAA      CCCCCCCC    CCCCCCCC
MM      MM    AAAAAA    KK      KK    AAAAAA      CCCCCCCC    CCCCCCCC
MMMM  MMMM  AA      AA  KK      KK  AA      AA  CC          CC
MMMM  MMMM  AA      AA  KK      KK  AA      AA  CC          CC
MM  MM  MM  AA      AA  KK    KK    AA      AA  CC          CC
MM  MM  MM  AA      AA  KK    KK    AA      AA  CC          CC
MM      MM  AA      AA  KKKKKK      AA      AA  CC          CC
MM      MM  AA      AA  KKKKKK      AA      AA  CC          CC
MM      MM  AAAAAAAAAA  KK    KK    AAAAAAAAAA  CC          CC
MM      MM  AAAAAAAAAA  KK    KK    AAAAAAAAAA  CC          CC
MM      MM  AA      AA  KK      KK  AA      AA  CC          CC              ....
MM      MM  AA      AA  KK      KK  AA      AA  CC          CC              ....
MM      MM  AA      AA  KK      KK  AA      AA    CCCCCCCC    CCCCCCCC      ....
MM      MM  AA      AA  KK      KK  AA      AA    CCCCCCCC    CCCCCCCC      ....

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 MAKACC (
;  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 makes the necessary hookups in the I/O database to
; 38        0038  1 !       reflect a new file access.
; 39        0039  1 ! ENVIRONMENT:
; 40        0040  1 !
; 41        0041  1 !       STARLET operating system, including privileged system services
; 42        0042  1 !       and internal exec routines. This routine must be called
; 43        0043  1 !       in kernel mode.
; 44        0044  1 !
; 45        0045  1 !--
; 46        0046  1 !
; 47        0047  1 !
; 48        0048  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  20-Dec-1976  17:28
; 49        0049  1 !
; 50        0050  1 ! MODIFIED BY:
; 51        0051  1 !
; 52        0052  1 !       V03-004 CDS0003         Christian D. Saether    19-Apr-1984
; 53        0053  1 !               Bump REFCNT in Fcb.  Do not bump other counts if
; 54        0054  1 !               this is NOACCLOCK.  Remove reference to old dirfcb index.
; 55        0055  1 !
; 56        0056  1 !       V03-003 CDS0002         Christian D. Saether     2-Mar-1984
; 57        0057  1 !               Set WRITE_TURN flag in WCB if index file, storage bitmap,
```

```
;    58          0058  1 !                 or a directory is being write accessed.
;    59          0059  1 !
;    60          0060  1 !       V03-002 CDS0001         Christian D. Saether    30-Dec-1983
;    61          0061  1 !                 Use L_NORM linkage and BIND_COMMON macro.
;    62          0062  1 !
;    63          0063  1 !       V03-001 LMP0059         L. Mark Pilant,         4-Jan-1983  12:28
;    64          0064  1 !                 Don't insert the FCB into the queue as it is done when the
;    65          0065  1 !                 FCB is created.
;    66          0066  1 !
;    67          0067  1 !       V02-002 LMP0003         L. Mark Pilant,         20-Nov-1981  9:30
;    68          0068  1 !                 Modify so that all the segments to a window get inserted into
;    69          0069  1 !                 the window queue.
;    70          0070  1 !
;    71          0071  1 !       V02-001 ACG0167         Andrew C. Goldstein,    16-Apr-1980  19:26
;    72          0072  1 !                 Previous revision history moved to F11B.REV
;    73          0073  1 !--
;    74          0074  1
;    75          0075  1
;    76          0076  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    77          0077  1 REQUIRE 'SRC$:FCPDEF.B32';
```

```
;    79          1068  1 GLOBAL ROUTINE MAKE_ACCESS (FCB, WINDOW, ABD) : L_NORM NOVALUE =
;    80          1069  1
;    81          1070  1 !++
;    82          1071  1 !
;    83          1072  1 ! FUNCTIONAL DESCRIPTION:
;    84          1073  1 !
;    85          1074  1 !       This routine makes the necessary hookups in the I/O database to
;    86          1075  1 !       reflect a new file access.
;    87          1076  1 !
;    88          1077  1 ! CALLING SEQUENCE:
;    89          1078  1 !       MAKE_ACCESS (ARG1, ARG2, ARG3)
;    90          1079  1 !
;    91          1080  1 ! INPUT PARAMETERS:
;    92          1081  1 !       ARG1: address of FCB to access
;    93          1082  1 !       ARG2: address of window to link up
;    94          1083  1 !       ARG3: address of buffer descriptors
;    95          1084  1 !
;    96          1085  1 ! IMPLICIT INPUTS:
;    97          1086  1 !       CURRENT_VCB: VCB of volume in process
;    98          1087  1 !
;    99          1088  1 ! OUTPUT PARAMETERS:
;   100          1089  1 !       NONE
;   101          1090  1 !
;   102          1091  1 ! IMPLICIT OUTPUTS:
;   103          1092  1 !       NONE
;   104          1093  1 !
;   105          1094  1 ! ROUTINE VALUE.
;   106          1095  1 !       NONE
;   107          1096  1 !
;   108          1097  1 ! SIDE EFFECTS:
;   109          1098  1 !       VCB transaction count bumped, access counts in FCB adjusted,
;   110          1099  1 !       FCB and window hooked up.
;   111          1100  1 !
;   112          1101  1 !--
;   113          1102  1
;   114          1103  2 BEGIN
;   115          1104  2
;   116          1105  2 MAP
;   117          1106  2         FCB             : REF BBLOCK,   ! FCB arg
;   118          1107  2         WINDOW          : REF BBLOCK,   ! window arg
;   119          1108  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH];
;   120          1109  2                                         ! buffer descriptor arg
;   121          1110  2
;   122          1111  2 LOCAL
;   123          1112  2         WINDOW_SEGMENT  : REF BBLOCK;   ! address of the current window segment
;   124          1113  2
;   125          1114  2 BIND_COMMON;
;   126          1115  2
;   127          1116  2 EXTERNAL
;   128          1117  2         PMS$GL_OPEN     : ADDRESSING_MODE (ABSOLUTE),
;   129          1118  2                                         ! system count of currently open files
;   130          1119  2         PMS$GL_OPENS    : ADDRESSING_MODE (ABSOLUTE);
;   131          1120  2                                         ! system count of files opened
;   132          1121  2
;   133          1122  2 ! Now hook the window onto the FCB and adjust the access counts
;   134          1123  2 ! according to the access control bits in the window.
;   135          1124  2 !
```

```
;  136          1125  2
;  137          1126  2 WINDOW_SEGMENT = .WINDOW;
;  138          1127  2 DO
;  139          1128  2     BEGIN
;  140          1129  2     INSQUE (.WINDOW_SEGMENT, .FCB[FCB$L_WLBL]);
;  141          1130  2     WINDOW_SEGMENT = .WINDOW_SEGMENT[WCB$L_LINK];
;  142          1131  2     END
;  143          1132  2 UNTIL .WINDOW_SEGMENT EQL 0;
;  144          1133  2
;  145          1134  2 FCB [FCB$W_REFCNT] = .FCB [FCB$W_REFCNT] + 1;    ! bump reference count
;  146          1135  2
;  147          1136  2 IF NOT .WINDOW [WCB$V_NOACCLOCK]
;  148          1137  2 THEN
;  149          1138  2     BEGIN
;  150          1139  2
;  151          1140  2     FCB[FCB$W_ACNT] = .FCB[FCB$W_ACNT] + 1;      ! bump access count
;  152          1141  2
;  153          1142  2     IF .WINDOW[WCB$V_NOREAD]
;  154          1143  2     THEN FCB[FCB$V_EXCL] = 1;            ! set exclusive access
;  155          1144  2
;  156          1145  2     IF .WINDOW[WCB$V_NOWRITE]
;  157          1146  2     THEN FCB[FCB$W_LCNT] = .FCB[FCB$W_LCNT] + 1; ! no writers
;  158          1147  2
;  159          1148  2     IF .WINDOW[WCB$V_NOTRUNC]
;  160          1149  2     THEN FCB[FCB$W_TCNT] = .FCB[FCB$W_TCNT] + 1; ! no truncates
;  161          1150  2
;  162          1151  2     END;
;  163          1152  2
;  164          1153  2 ! For a write access, bump the writer count. If this is the
;  165          1154  2 ! first write, and the file is the index file or the storage map, set
;  166          1155  2 ! the appropriate flag in the VCB.
;  167          1156  2 !
;  168          1157  2
;  169          1158  2 IF .WINDOW[WCB$V_WRITE]
;  170          1159  2 THEN
;  171          1160  3     BEGIN
;  172          1161  3     IF .FCB [FCB$B_FID_NMX] EQL 0
;  173          1162  3     THEN
;  174          1163  4         BEGIN
;  175          1164  4         IF .FCB[FCB$W_FID_NUM] EQL 1
;  176          1165  4         THEN
;  177          1166  5             BEGIN
;  178          1167  5             CURRENT_VCB[VCB$V_WRITE_IF] = 1;
;  179          1168  5             WINDOW [WCB$V_WRITE_TURN] = 1;
;  180          1169  4             END;
;  181          1170  4
;  182          1171  4         IF .FCB[FCB$W_FID_NUM] EQL 2
;  183          1172  4         THEN
;  184          1173  5             BEGIN
;  185          1174  5             CURRENT_VCB[VCB$V_WRITE_SM] = 1;
;  186          1175  5             WINDOW [WCB$V_WRITE_TURN] = 1;
;  187          1176  4             END;
;  188          1177  3         END;
;  189          1178  3
;  190          1179  3     IF .FCB[FCB$V_DIR]
;  191          1180  3     THEN
;  192          1181  4         BEGIN
```

```
4           FCB[FCB$W_DIRSEQ] = .FCB[FCB$W_DIRSEQ] + 1;
4           WINDOW [WCB$V_WRITE_TURN] = 1;
3           END;
3
3       IF NOT .WINDOW [WCB$V_NOACCLOCK]
3       THEN
3           FCB[FCB$W_WCNT] = .FCB[FCB$W_WCNT] + 1;
3
2       END;
2
2 ! Count the access in the volume transaction count and return
2 ! the window address for the user's CCB.
2 !
2
2 PMS$GL_OPEN = .PMS$GL_OPEN + 1;          ! bump open file count
2 PMS$GL_OPENS = .PMS$GL_OPENS + 1;        ! bump count of opens
2 CURRENT_VCB[VCB$W_TRANS] = .CURRENT_VCB[VCB$W_TRANS] + 1;
2
2 ABD[ABD$C_WINDOW, ABD$W_COUNT] = 4;      ! enable write-back
2 .ABD[ABD$C_WINDOW, ABD$Q_TEXT] + ABD[ABD$C_WINDOW, ABD$W_TEXT] + 1 = .WINDOW;
2                                          ! put window address in buffer text
2
2 ! Mark the access complete in the cleanup action flags.
2 !
2
2 CLEANUP_FLAGS[CLF_DEACCESS] = 1;
2
1 END;                                     ! end of routine MAKE_ACCESS

                                                .TITLE   MAKACC
                                                .IDENT   \V04-000\

                                                .EXTRN   PMS$GL_OPEN, PMS$GL_OPENS

                                                .PSECT   $CODE$,NOWRT,2

                           0000 00000           .ENTRY   MAKE_ACCESS, Save nothing                    ; 1068
          51    08   AC   D0 00002              MOVL     WINDOW, WINDOW_SEGMENT                        ; 1126
          50    04   AC   D0 00006 1$:          MOVL     FCB, R0                                       ; 1129
     14   B0         61   0E 0000A              INSQUE   (WINDOW_SEGMENT), @20(R0)                     ;
          51    20   A1   D0 0000E              MOVL     32(WINDOW_SEGMENT), WINDOW_SEGMENT            ; 1130
                     F2   12 00012              BNEQ     1$                                            ; 1132
          50    04   AC   D0 00014              MOVL     FCB, R0                                       ; 1134
                18   A0   B6 00018              INCW     24(R0)                                        ;
          50    08   AC   D0 0001B              MOVL     WINDOW, R0                                    ; 1136
37   14   A0         02   E0 0001F              BBS      #2, 20(R0), 4$                                ;
          50    04   AC   D0 00024              MOVL     FCB, R0                                       ; 1140
                1A   A0   B6 00028              INCW     26(R0)                                        ;
          50    08   AC   D0 0002B              MOVL     WINDOW, R0                                    ; 1142
08   15   A0         02   E1 0002F              BBC      #2, 21(R0), 2$                                ;
          50    04   AC   D0 00034              MOVL     FCB, R0                                       ; 1143
     22   A0         08   88 00038              BISB2    #8, 34(R0)                                    ;
          50    08   AC   D0 0003C 2$:          MOVL     WINDOW, R0                                    ; 1145
          07    14   A0   E9 00040              BLBC     20(R0), 3$                                    ;
          50    04   AC   D0 00044              MOVL     FCB, R0                                       ; 1146
                1E   A0   B6 00048              INCW     30(R0)                                        ;
```

```
                50      08  AC  D0 0004B 3$:    MOVL    WINDOW, R0                              ; 1148
07      15      A0          03  E1 0004F         BBC     #3, 21(R0), 4$
                50      04  AC  D0 00054         MOVL    FCB, R0                                 ; 1149
                        20  A0  B6 00058         INCW    32(R0)
                50      08  AC  D0 0005B 4$:    MOVL    WINDOW, R0                              ; 1158
5C      0B      A0          01  E1 0005F         BBC     #1, 11(R0), 8$
                50      04  AC  D0 00064         MOVL    FCB, R0                                 ; 1161
                        29  A0  95 00068         TSTB    41(R0)
                            30  12 0006B         BNEQ    6$
                01      24  A0  B1 0006D         CMPW    36(R0), #1                              ; 1164
                            10  12 00071         BNEQ    5$
                50      98  AA  D0 00073         MOVL    -104(BASE), R0                          ; 1167
            0B  A0          01  88 00077         BISB2   #1, 11(R0)
                50      08  AC  D0 0007B         MOVL    WINDOW, R0                              ; 1168
            15  A0          10  88 0007F         BISB2   #16, 21(R0)
                50      04  AC  D0 00083 5$:    MOVL    FCB, R0                                 ; 1171
                02      24  A0  B1 00087         CMPW    36(R0), #2
                            10  12 0008B         BNEQ    6$
                50      98  AA  D0 0008D         MOVL    -104(BASE), R0                          ; 1174
            0B  A0          02  88 00091         BISB2   #2, 11(R0)
                50      08  AC  D0 00095         MOVL    WINDOW, R0                              ; 1175
            15  A0          10  88 00099         BISB2   #16, 21(R0)
                50      04  AC  D0 0009D 6$:    MOVL    FCB, R0                                 ; 1179
                0B      22  A0  E9 000A1         BLBC    34(R0), 7$
                        42  A0  B6 000A5         INCW    66(R0)                                  ; 1182
                50      08  AC  D0 000A8         MOVL    WINDOW, R0                              ; 1183
            15  A0          10  88 000AC         BISB2   #16, 21(R0)
                50      08  AC  D0 000B0 7$:    MOVL    WINDOW, R0                              ; 1186
07      14  A0          02  E0 000B4         BBS     #2, 20(R0), 8$
                50      04  AC  D0 000B9         MOVL    FCB, R0                                 ; 1188
                        1C  A0  B6 000BD         INCW    28(R0)
                00000000G   9F  D6 000C0 8$:    INCL    @#PMS$GL_OPEN                           ; 1196
                00000000G   9F  D6 000C6         INCL    @#PMS$GL_OPENS                          ; 1197
                50      98  AA  D0 000CC         MOVL    -104(BASE), R0                          ; 1198
                        0C  A0  B6 000D0         INCW    12(R0)
                50      0C  AC  D0 000D3         MOVL    ABD, R0                                 ; 1200
            02  A0          04  B0 000D7         MOVW    #4, 2(R0)
                50      0C  BC  3C 000DB         MOVZWL  @ABD, R0                                ; 1201
                50      0C  AC  C0 000DF         ADDL2   ABD, R0
            01  A0      08  AC  D0 000E3         MOVL    WINDOW, 1(R0)
            02  AA          01  88 000E8         BISB2   #1, 2(BASE)                             ; 1207
                            04  00 000EC         RET                                             ; 1209

; Routine Size:  237 bytes,    Routine Base:  $CODE$ + 0000

;  221          1210  1
;  222          1211  1 END
;  223          1212  0 ELUDOM



;                                PSECT SUMMARY
;
```

```
;       Name                          Bytes                       Attributes
;
;  $CODE$                                   237  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        42         0       1000          00:01.9



;                                  COMMAND QUALIFIERS

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

; Size:          237 code + 0 data bytes
; Run Time:         00:18.3
; Elapsed Time:     00:38.3
; Lines/CPU Min:     3984
; Lexemes/CPU-Min: 49384
; Memory Used:  239 pages
; Compilation Complete
```

PMS
LIS

QUOTAUTIL
LIS

MAKPTR
LIS

MATCHNAME
LIS

MPWIND
LIS

LOCKDB
LIS

LOCKERS
LIS

MAKACC
LIS

PARSNM
LIS

MAPVBN
LIS

IODONE
LIS

LOCKDN
LIS

MODIFY
LIS

MOUNT
LIS

NXTHDR
LIS

MAKNMB
LIS

MAKSTR
LIS