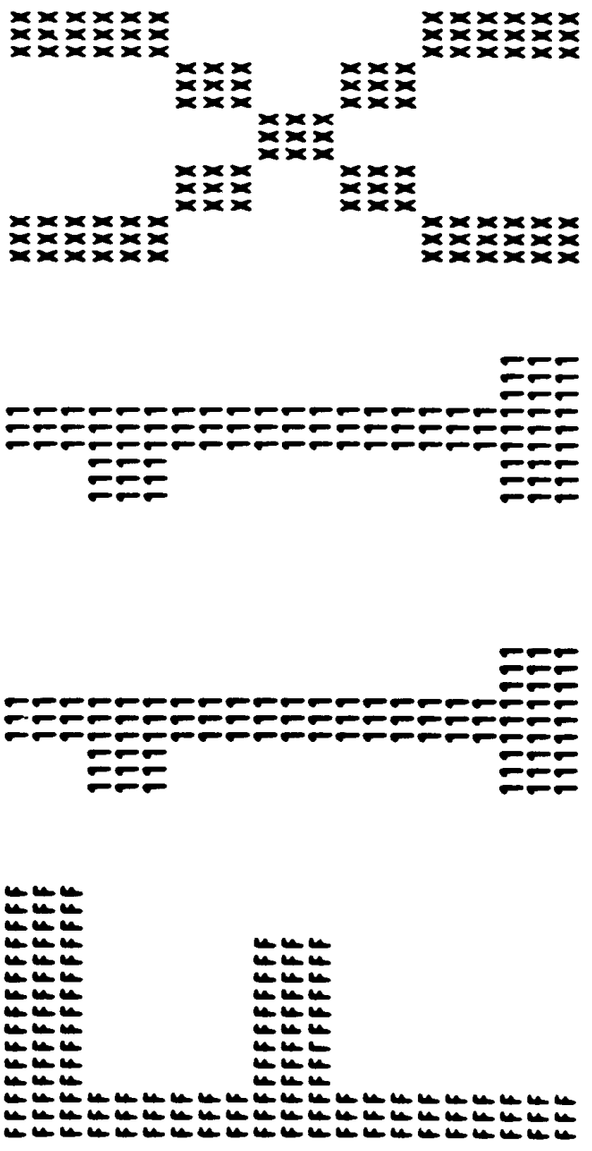

```
DDDDDDDD    EEEEEEEEEE    AAAAAA      CCCCCCCC    CCCCCCCC    SSSSSSSS
DDDDDDDD    EEEEEEEEEE    AAAAAA      CCCCCCCC    CCCCCCCC    SSSSSSSS
DD      DD  EE            AA  AA    CC          CC          SS
DD      DD  EE            AA  AA    CC          CC          SS
DD      DD  EE            AA  AA    CC          CC          SS
DD      DD  EE            AA  AA    CC          CC          SS
DD      DD  EEEEEEEE      AA  AA    CC          CC            SSSSSS
DD      DD  EEEEEEEE      AA  AA    CC          CC            SSSSSS
DD      DD  EE            AAAAAAAAAA  CC          CC                  SS
DD      DD  EE            AAAAAAAAAA  CC          CC                  SS
DD      DD  EE            AA  AA    CC          CC                  SS      ....
DD      DD  EE            AA  AA    CC          CC                  SS      ....
DDDDDDDD    EEEEEEEEEE    AA  AA      CCCCCCCC    CCCCCCCC  SSSSSSSS        ....
DDDDDDDD    EEEEEEEEEE    AA  AA      CCCCCCCC    CCCCCCCC  SSSSSSSS        ....

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
```

DEA
V04

: R

;

```
;     1          0001  0 MODULE DEACCS (
;     2          0002  0                 LANGUAGE (BLISS32),
;001  !CDS0009   0003  0                 IDENT = 'V04-001'
;     4-1        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 2
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       This routine implements the DEACCESS functior.
;    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:  6-Jan-1977  23:29
;    48          0048  1 !
;    49          0049  1 ! MODIFIED BY:
;    50          0050  1 !
;001  !CDS0009   0051  1 !       V04-001 CDS0009         Christian D. Saether     9-Nov-1984
;002  !CDS0009   0052  1 !               Mark FCB stale when deferring truncation.
;003  !CDS0009   0053  1 !
;    51          0054  1 !       V03-012 CDS0008         Christian D. Saether    21-Aug-1984
;    52          0055  1 !               Changes to handle stale fcbs.
;    53          0056  1 !
;    54          0057  1 !       V03-011 CDS0007         Christian D. Saether    19-Apr-1984
```

```
;  55          0058  1 !               Many changes to reflect modified access lock handling.
;  56          0059  1 !
;  57          0060  1 !       V03-010 CDS0006         Christian D. Saether    29-Dec-1983
;  58          0061  1 !               Use L_NORM linkage and BIND_COMMON macro.
;  59          0062  1 !
;  60          0063  1 !       V03-009 CDS0005         Christian D. Saether    23-Sep-1983
;  61          0064  1 !               Manually merge in ACG0343, ACG59616, STJ3109.
;  62          0065  1 !
;  63          0066  1 !       V03-008 ACG0343         Andrew C. Goldstein,    19-Jul-1983  16:46
;  64          0067  1 !               Inhibit revision date count if NORECORD is specified
;  65          0068  1 !
;  66          0069  1 !       V03-007 ACG59616        Andrew C. Goldstein,    21-Jun-1983  15:53
;  67          0070  1 !               Create common subroutine for revision and expiration dates
;  68          0071  1 !
;  69          0072  1 !       V03-006 STJ3109         Steven T. Jeffreys,     06-Jun-1983
;  70          0073  1 !               Copy FHWM from FCB to file header.
;  71          0074  1 !
;  72          0075  1 !       V03-005 CDS0004         Christian D. Saether    14-Sep-1983
;  73          0076  1 !               Modify SERIAL_FILE interface.
;  74          0077  1 !
;  75          0078  1 !       V03-004 LMP0149         L. Mark Pilant,         13-Sep-1983  11:26
;  76          0079  1 !               Correct a logic problem that caused problems during the
;  77          0080  1 !               protection check of a write attribute operation.
;  78          0081  1 !
;  79          0082  1 !       V03-003 CDS0003         Christian D. Saether     4-May-1983
;  80          0083  1 !               Sychronize processing by FID using SERIAL_FILE.
;  81          0084  1 !
;  82          0085  1 !       V03-002 CDS0002         Christian D. Saether    21-Apr-1983
;  83          0086  1 !               Modify truncate access arbitration checks to permit
;  84          0087  1 !               cluster operation.  Possibly defer truncation or
;  85          0088  1 !               perform a deferred truncate operation.
;  86          0089  1 !
;  87          0090  1 !       V03-001 CDS0001         Christian D. Saether     7-Apr-1983
;  88          0091  1 !               Make mark-for-delete checks work in a cluster.
;  89          0092  1 !
;  90          0093  1 !       V02-006 ACG0258         Andrew C. Goldstein,    26-Jan-1982  16:56
;  91          0094  1 !               Fix reference to RVN 1 in expiration date processing
;  92          0095  1 !
;  93          0096  1 !       V02-005 ACG0230         Andrew C. Goldstein,    23-Dec-1981  23:46
;  94          0097  1 !               Add expiration date support
;  95          0098  1 !
;  96          0099  1 !       V02-004 ACG0247         Andrew C. Goldstein,    23-Dec-1981  20:49
;  97          0100  1 !               Update revision count only if written
;  98          0101  1 !
;  99          0102  1 !       V02-003 ACG0245         Andrew C. Goldstein,    23-Dec-1981  20:48
; 100          0103  1 !               Move queueing of spool file to cleanup
; 101          0104  1 !
; 102          0105  1 !       V02-002 ACG0167         Andrew C. Goldstein,    16-Apr-1980  19:25
; 103          0106  1 !               Previous revision history moved to [F11B.SRC]F11B.REV
; 104          0107  1 !--
; 105          0108  1
; 106          0109  1
; 107          0110  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
; 108          0111  1 REQUIRE 'SRC$:FCPDEF.B32';
; 109          1102  1
; 110          1103  1 FORWARD ROUTINE
; 111          1104  1         SET_REVISION    : L_NORM NOVALUE, ! set revision and expiration date
```

```
;  112          1105  1          TRUNC_HANDLER;                  . handler for delayed truncate
```

```
;  114          1106  1 GLOBAL ROUTINE DEACCESS  : L_NORM =
;  115          1107  1
;  116          1108  1 !++
;  117          1109  1 !
;  118          1110  1 ! FUNCTIONAL DESCRIPTION:
;  119          1111  1 !
;  120          1112  1 !       This routine implements the DEACCESS function.
;  121          1113  1 !       If an attribute list is present, attributes are written.
;  122          1114  1 !
;  123          1115  1 ! CALLING SEQUENCE:
;  124          1116  1 !       DEACCESS ()
;  125          1117  1 !
;  126          1118  1 ! INPUT PARAMETERS:
;  127          1119  1 !       NONE
;  128          1120  1 !
;  129          1121  1 ! IMPLICIT INPUTS:
;  130          1122  1 !       IO_PACKET: I/O packet in process
;  131          1123  1 !       CURRENT_WINDOW: window of file
;  132          1124  1 !       PRIMARY_FCB: FCB of file
;  133          1125  1 !
;  134          1126  1 ! OUTPUT PARAMETERS:
;  135          1127  1 !       NONE
;  136          1128  1 !
;  137          1129  1 ! IMPLICIT OUTPUTS:
;  138          1130  1 !       NONE
;  139          1131  1 !
;  140          1132  1 ! ROUTINE VALUE:
;  141          1133  1 !       NONE
;  142          1134  1 !
;  143          1135  1 ! SIDE EFFECTS:
;  144          1136  1 !       file deaccessed
;  145          1137  1 !       FCB may be deleted
;  146          1138  1 !       header may be modified
;  147          1139  1 !
;  148          1140  1 !--
;  149          1141  1
;  150          1142  2 BEGIN
;  151          1143  2
;  152          1144  2 LABEL
;  153          1145  2         DELAY_TRUNC;                    ! truncation delay block
;  154          1146  2
;  155          1147  2 LOCAL
;  156          1148  2         DO_EXPIRE,                      ! flag indicating expiration to be updated
;  157          1149  2         MODIFIED,                       ! flag indicating file has been modified
;  158          1150  2         K,                              ! local copy of truncate lock count
;  159          1151  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH],
;  160          1152  2         FIB             : REF BBLOCK,   ! FIB
;  161          1153  2         FCB             : REF BBLOCK,   ! pointer to FCB
;  162          1154  2         HEADER          : REF BBLOCK;   ! file header
;  163          1155  2
;  164          1156  2 BIND_COMMON;
;  165          1157  2
;  166          1158  2 EXTERNAL ROUTINE
;001  !CDS0009  1159  2         MAKE_FCB_STALE  : L_NORM NOVALUE, ! mark FCB as stale
;  167          1160  2         REBLD_PRIM_FCB  : L_NORM NOVALUE, ! rebuild primary fcb from header
;  168          1161  2         BUILD_EXT_FCBS  : L_NORM NOVALUE, ! build extension fcb chain
;  169          1162  2         CONV_ACCLOCK    : L_NORM,       ! convert access lock.
```

```
;  170          1163  2          LOCK_COUNT      : L_NORM,       ! get count of granted locks.
;  171          1164  2          SERIAL_FILE     : L_NORM,       ! interlock file processing
;  172          1165  2          TRUNC_CHECKS    : L_JSB_2ARGS NOVALUE, ! parameter checks
;  173          1166  2          GET_FIB         : L_NORM,       ! get FIB of request
;  174          1167  2          READ_HEADER     : L_NORM,       ! read file header
;  175          1168  2          MARK_DIRTY      : L_NORM,       ! mark buffer for write-back
;  176          1169  2          WRITE_ATTRIB    : L_NORM,       ! write attributes routine
;  177          1170  2          TRUNCATE        : L_NORM,       ! truncate file
;  178          1171  2          UPDATE_FCB      : L_NORM,       ! update contents of FCB
;  179          1172  2          CHECKSUM        : L_NORM;       ! compute file header checksum
;  180          1173  2
;  181          1174  2  ! Set the cleanup flags to cause the deaccess to occurr.
;  182          1175  2  ! Find the buffer descriptor and FIB.
;  183          1176  2  !
;  184          1177  2
;  185          1178  2  CLEANUP_FLAGS[CLF_ZCHANNEL] = 1;
;  186          1179  2  CLEANUP_FLAGS[CLF_DEACCESS] = 1;
;  187          1180  2  CLEANUP_FLAGS[CLF_DELWINDOW] = 1;
;  188          1181  2
;  189          1182  2                                          ! pointer to buffer descriptors
;  190          1183  2  ABD = .BBLOCK [.IO_PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPT];
;  191          1184  2  FIB = GET_FIB (.ABD);
;  192          1185  2  FCB = .PRIMARY_FCB;
;  193          1186  2
;  194          1187  2  ! Synchronize further file processing.
;  195          1188  2  !
;  196          1189  2
;  197          1190  2  PRIM_LCKINDX = SERIAL_FILE (FCB [FCB$W_FID]);
;  198          1191  2
;  199          1192  2  ! Make sure irrelevant parameters are not present.
;  200          1193  2  !
;  201          1194  2
;  202          1195  2  IF .FIB[FIB$V_EXTEND]
;  203          1196  2  THEN ERR_STATUS (SS$_BADPARAM);
;  204          1197  2
;  205          1198  2  ! If the access lock is held in NL mode, and this file is cluster
;  206          1199  2  ! accessible, then set the stale flag to force rebuild of the fcbs
;  207          1200  2  ! from the header(s).
;  208          1201  2  !
;  209          1202  2
;  210          1203  2  IF .FCB [FCB$B_ACCLKMODE] EQL LCK$K_NLMODE
;  211          1204  2      AND .FCB [FCB$L_ACCLKID] NEQ 0
;  212          1205  2  THEN
;  213          1206  2      FCB [FCB$V_STALE] = 1;
;  214          1207  2
;  215          1208  2  ! Determine if the expiration date is to be updated, and if the file has
;  216          1209  2  ! actually been modified.
;  217          1210  2  !
;  218          1211  2
;  219          1212  2  DO_EXPIRE = .CURRENT_WINDOW[WCB$V_EXPIRE]
;  220          1213  2             AND NOT .FIB[FIB$V_NORECORD]
;  221          1214  3             AND (.CURRENT_WINDOW[WCB$L_WRITES] NEQ 0
;  222          1215  3                 OR .CURRENT_WINDOW[WCB$L_READS] NEQ 0
;  223          1216  2                 OR .FCB[FCB$L_EFBLK] EQL 0);
;  224          1217  2  MODIFIED =   .CURRENT_WINDOW[WCB$V_WRITE]
;  225          1218  2             AND NOT .FIB[FIB$V_NORECORD]
;  226          1219  3             AND (.CURRENT_WINDOW[WCB$L_WRITES] NEQ 0
```

```
;  227          1220  3              OR .IO_PACKET[IRP$W_BCNT] GTRU ABD$C_ATTRIB
;  228          1221  2              OR .FIB[FIB$V_TRUNC]);
;  229          1222  2
;  230          1223  2 ! If the file is accessed for write, if we must update the expiration
;  231          1224  2 ! date, or if the file is marked for delete or is marked bad and this
;  232          1225  2 ! is the last access, read the header.
;  233          1226  2 !
;  234          1227  2
;  235          1228  2 IF .CURRENT_WINDOW[WCB$V_WRITE]
;  236          1229  2 OR .DO_EXPIRE
;  237          1230  4 OR ((.FCB[FCB$V_MARKDEL]
;  238          1231  4           OR .FCB [FCB$V_DELAYTRNC]
;  239          1232  4           OR .FCB [FCB$V_STALE]
;  240          1233  4           OR .FCB[FCB$V_BADBLK]
;  241          1234  4           OR .CLEANUP_FLAGS[CLF_SPOOLFILE])
;  242          1235  3       AND .FCB[FCB$W_REFCNT] EQL 1)
;  243          1236  2 THEN
;  244          1237  3       BEGIN
;  245          1238  3       HEADER = READ_HEADER (0, .FCB);
;  246          1239  3
;  247          1240  3       IF .FCB [FCB$V_STALE]
;  248          1241  3       THEN
;  249          1242  4           BEGIN
;  250          1243  4           REBLD_PRIM_FCB (.FCB, .HEADER);
;  251          1244  4           BUILD_EXT_FCBS (.HEADER);
;  252          1245  3           END;
;  253          1246  2       END;
;  254          1247  2
;  255          1248  2 ! If this the last deaccess from a file marked for delete, delete the file.
;  256          1249  2 ! If the file is a spool file, send it to the job controller.
;  257          1250  2 !
;  258          1251  2
;  259          1252  2 IF .FCB[FCB$W_REFCNT] EQL 1
;  260          1253  2 THEN
;  261          1254  3       BEGIN
;  262          1255  3       IF .FCB[FCB$V_MARKDEL]
;  263          1256  3       THEN
;  264          1257  3
;  265          1258  3 ! Make sure we are the only accessor left in the entire cluster.
;  266          1259  3 !
;  267          1260  3
;  268          1261  3           IF LOCK_COUNT (.FCB [FCB$L_ACCLKID]) EQL 1
;  269          1262  3           THEN
;  270          1263  3               CLEANUP_FLAGS [CLF_DELFILE] = 1;
;  271          1264  3
;  272          1265  3       IF .CLEANUP_FLAGS[CLF_SPOOLFILE]
;  273          1266  3       THEN CLEANUP_FLAGS[CLF_DOSPOOL] = 1;
;  274          1267  3
;  275          1268  3 ! If the FCB is marked bad, now set the bad block bit in the file header.
;  276          1269  3 !
;  277          1270  3
;  278          1271  3       IF .FCB[FCB$V_BADBLK]
;  279          1272  3       THEN
;  280          1273  4           BEGIN
;  281          1274  4           HEADER[FH2$V_BADBLOCK] = 1;
;  282          1275  4           MARK_DIRTY (.HEADER);
;  283          1276  3           END;
```

```
;  284     1277  2      END;
;  285     1278  2
;  286     1279  2  ! Update revision count, date, and expiration date as appropriate.
;  287     1280  2  !
;  288     1281  2
;  289     1282  2  IF .MODIFIED
;  290     1283  2  OR .DO_EXPIRE
;  291     1284  2  THEN SET_REVISION (.HEADER, .MODIFIED);
;  292     1285  2
;  293     1286  2  ! Do deaccess processing for a write accessed file. If a deaccess lock
;  294     1287  2  ! was requested on the file, set the lock bit. Then process the write
;  295     1288  2  ! attributes, if any. If attributes were written, then clear the
;  296     1289  2  ! lock bit.
;  297     1290  2  !
;  298     1291  2
;  299     1292  2  IF .CURRENT_WINDOW[WCB$V_WRITE]
;  300     1293  2  THEN
;  301     1294  3      BEGIN
;  302     1295  3      MARK_DIRTY (.HEADER);
;  303     1296  3
;  304     1297  3      ! Update the FHWM in the file header.
;  305     1298  3      !
;  306     1299  3      !   o If the FHWM is not supported in this header, do nothing.
;  307     1300  3      !
;  308     1301  3      !   o If the volume FHWM attribute is disabled, then set the FHWM
;  309     1302  3      !     to the file size + 1.  This will protect the contents of the
;  310     1303  3      !     file should it be opened and modified some time in the future
;  311     1304  3      !     when the volume's FHWM attribute is enabled.
;  312     1305  3      !
;  313     1306  3      !   o If the FCB FHWM is 0, and the file header supports FHWM, then
;  314     1307  3      !     set the header's FHWM to the file size + 1.  This will likewise
;  315     1308  3      !     protect the file contents.
;  316     1309  3      !
;  317     1310  3      !   o If the FCB FHWM is nonzero, and the file header supports FHWM, and
;  318     1311  3      !     the volume FHWM attribute is enabled, simply copy the FCB FHWM to
;  319     1312  3      !     the file header.
;  320     1313  3      !
;  321     1314  3      ! For FHWM to be supported, the 'highwater' field in the
;  322     1315  3      ! header must be present.  All files created on version 4
;  323     1316  3      ! or later systems will have this characteristic.
;  324     1317  3      !
;  325     1318  3
;  326     1319  3      IF  .HEADER[FH2$B_IDOFFSET] GEQU ($BYTEOFFSET(FH2$L_HIGHWATER)+4)/2
;  327     1320  3      THEN
;  328     1321  3          IF .CURRENT_VCB[VCB$V_NOHIGHWATER]
;  329     1322  3          OR .FCB[FCB$L_HIGHWATER] EQL 0
;  330     1323  3          THEN
;  331     1324  3              HEADER[FH2$L_HIGHWATER] = .FCB[FCB$L_FILESIZE] + 1
;  332     1325  3          ELSE
;  333     1326  3              HEADER[FH2$L_HIGHWATER] = .FCB[FCB$L_HIGHWATER];
;  334     1327  3
;  335     1328  3
;  336     1329  3      IF .CURRENT_WINDOW[WCB$V_DLOCK]
;  337     1330  3      THEN HEADER[FH2$V_LOCKED] = 1;
;  338     1331  3
;  339     1332  3      IF .IO_PACKET[IRP$W_BCNT] GTR ABD$C_ATTRIB
;  340     1333  3      AND .USER_STATUS[0]
```

```
;  341          1334  3      THEN
;  342          1335  4          BEGIN
;  343          1336  4          WRITE_ATTRIB (.HEADER, .ABD, 0);
;  344          1337  4          HEADER = .FILE_HEADER;
;  345          1338  4          HEADER[FH2$V_LOCKED] = 0;
;  346          1339  3          END;
;  347          1340  3
;  348          1341  3      ! If a truncate is requested, do it, if the file was write accessed and
;  349          1342  3      ! there are not other accessors now, else delay the truncation until
;  350          1343  3      ! the last reader deaccess.
;  351          1344  3      !
;  352          1345  3
;  353          1346  3      IF .FIB[FIB$V_TRUNC]
;  354          1347  3          AND NOT .FCB [FCB$V_MARKDEL]
;  355          1348  3      THEN
;  356          1349  4          BEGIN
;  357          1350  4          IF .CURRENT_VCB[VCB$V_NOALLOC]
;  358          1351  4          THEN ERR_EXIT (SS$_WRITLCK);
;  359          1352  4
;  360          1353  4          IF .FCB [FCB$W_REFCNT] EQL 1
;  361          1354  4              AND LOCK_COUNT (.FCB [FCB$L_ACCLKID]) EQL 1
;  362          1355  4          THEN
;  363          1356  5              BEGIN
;  364          1357  5
;  365          1358  5              CHECKSUM (.HEADER);
;  366          1359  5              TRUNCATE (.FIB, .HEADER, .FIB [FIB$L_EXVBN]);
;  367          1360  5              CLEANUP_FLAGS[CLF_FIXFCB] = 0;
;  368          1361  5              UPDATE_FCB (.FILE_HEADER);
;  369          1362  5              END
;  370          1363  4          ELSE
;  371          1364  4              IF .FCB [FCB$W_WCNT] EQL 1  ! 1 is just us.
;  372          1365  5                  AND (.FCB [FCB$V_EXCL]  ! must be a NOLOCK somewhere
;  373          1366  5                  OR .FCB [FCB$W_LCNT] NEQ 0 ! must be us
;  374          1367  5                  OR CONV_ACCLOCK (LCK$K_PWMODE, .FCB))
;  375          1368  5                                          ! lock will be converted back
;  376          1369  5                                          ! in MAKE_DEACCESS
;  377          1370  5
;  378          1371  5      ! There are other readers, but no writers, accessing the file, so we will make
;  379          1372  5      ! checks to see if the truncation arguments make sense, and if so,
;  380          1373  5      ! store appropriate context in the FCB so that the last reader to deaccess
;  381          1374  5      ! the file will perform the truncation.
;  382          1375  5      !
;  383          1376  5
;  384          1377  4              THEN
;  385          1378  5                  BEGIN
;  386          1379  5                  LOCAL
;  387          1380  5                      TRNVBN;
;  388          1381  5
;  389          1382  5                  TRNVBN = .FIB [FIB$L_EXVBN];
;  390          1383  5                  TRUNC_CHECKS (.FIB, .HEADER);
;  391          1384  5
;  392          1385  5      ! lock will be converted when new lock mode is calcuated and lock
;  393          1386  5      ! converted in MAKE_DEACCESS.  Even if it was not converted up to
;  394          1387  5      ! PW above (i.e., was already held in either), it will have to be
;  395          1388  5      ! lowered because this thread means the last writer on this node is
;  396          1389  5      ! going away.
;  397          1390  5      !
```

```
;  398          1391  5
;  399          1392  5                FCB [FCB$V_DELAYTRNC] = 1;
;  400          1393  5                FCB [FCB$L_TRUNCVBN] = .TRNVBN;
;  401          1394  5                FIB [FIB$L_EXVBN] = .FCB [FCB$L_FILESIZE] + 1;
;001  !CDS0009   1395  5
;002  !CDS0009   1396  5 ! Mark FCB as stale so that other nodes will pick up the fact that
;003  !CDS0009   1397  5 ! a deferred truncation remains to be done when the last access goes away.
;004  !CDS0009   1398  5 !
;005  !CDS0009   1399  5
;006  !CDS0009   1400  5                MAKE_FCB_STALE (.FCB);
;  402          1401  5                END
;  403          1402  4            ELSE
;  404          1403  4                ERR_EXIT (SS$_ACCONFLICT);
;  405          1404  4
;  406          1405  3        END;                ! of wanted to do a truncation
;  407          1406
;  408          1407  3    END                     ! of was write accessed.
;  409          1408  3
;  410          1409  2 ELSE
;  411          1410  2 DELAY_TRUNC:
;  412          1411  3    BEGIN                   ! not write accessd
;  413          1412  3
;  414          1413  3    BUILTIN FP;
;  415          1414  3
;  416          1415  3    LOCAL SAVE_US1;
;  417          1416  3
;  418          1417  3    IF NOT .FCB [FCB$V_DELAYTRNC]
;  419          1418  3    THEN LEAVE DELAY_TRUNC;
;  420          1419  3
;  421          1420  3    IF .FCB [FCB$W_REFCNT] NEQ 1
;  422          1421  3        OR .FCB [FCB$V_MARKDEL]
;  423          1422  3        OR .FCB [FCB$L_TRUNCVBN] EQL 0
;  424          1423  3        OR LOCK_COUNT (.FCB [FCB$L_ACLKID]) NEQ 1
;  425          1424  3    THEN LEAVE DELAY_TRUNC;
;  426          1425  3
;  427          1426  3    SAVE_US1 = .USER_STATUS [1];
;  428          1427  3    CHECKSUM (.HEADER);
;  429          1428  3
;  430          1429  3    .FP = TRUNC_HANDLER;
;  431          1430  3    TRUNCATE (SECOND_FIB, .HEADER, .FCB [FCB$L_TRUNCVBN]);
;  432          1431  3    .FP = 0;
;  433          1432  3
;  434          1433  3    USER_STATUS [1] = .SAVE_US1;
;  435          1434  3
;  436          1435  2    END;
;  437          1436  2
;  438          1437  2 ! Return failure to let the error cleanup do the actual deaccessing.
;  439          1438  2 !
;  440          1439  2
;  441          1440  2 RETURN 0;
;  442          1441  2
;  443          1442  1 END;                                    ! end of routine DEACCESS

                                                        .TITLE  DEACCS
                                                        .IDENT  \V04-001\
```

```
                                        .EXTRN  MAKE_FCB_STALE, REBLD_PRIM_FCB
                                        .EXTRN  BUILD_EXT_FCBS, CONV_ACCLOCK
                                        .EXTRN  LOCK_COUNT, SERIAL_FILE
                                        .EXTRN  TRUNC_CHECKS, GET_FIB
                                        .EXTRN  READ_HEADER, MARK_DIRTY
                                        .EXTRN  WRITE_ATTRIB, TRUNCATE
                                        .EXTRN  UPDATE_FCB, CHECKSUM

                                        .PSECT  $CODE$,NOWRT,2

                          0BFC 00000    .ENTRY  DEACCESS, Save R2,R3,R4,R5,R6,R7,R8,R9,R11      ; 1106
          5B      0000G   CF   9E 00002 MOVAB   LOCK_COUNT, R11
          56        80    AA   9E 00007 MOVAB   -128(BASE), R6                                  ; 1154
          58        0C    AA   9E 0000B MOVAB   12(BASE), R8
    02    AA      0403    8F   A8 0000F BISW2   #1027, 2(BASE)                                  ; 1180
          50        90    AA   D0 00015 MOVL    -112(BASE), R0                                  ; 1183
          59        2C    B0   D0 00019 MOVL    @44(R0), ABD
                          59   DD 0001D PUSHL   ABD                                           ; 1184
  0000G   CF              01   FB 0001F CALLS   #1, GET_FIB
          54              50   D0 00024 MOVL    R0, FIB
          52        08    AA   D0 00027 MOVL    8(BASE), FCB                                    ; 1185
                    24    A2   9F 0002B PUSHAB  36(FCB)                                         ; 1190
  0000G   CF              01   FB 0002E CALLS   #1, SERIAL_FILE
    18    AA              50   D0 00033 MOVL    R0, 24(BASE)
                    16    A4   95 00037 TSTB    22(FIB)                                         ; 1195
                          06   18 0003A BGEQ    1$
          03              66   E9 0003C BLBC    (R6), 1$                                        ; 1196
          66              14   B0 0003F MOVW    #20, (R6)
                    0B    A2   95 00042 1$:     TSTB    11(FCB)                                 ; 1203
                          09   12 00045 BNEQ    2$
                    48    A2   D5 00047 TSTL    72(FCB)                                         ; 1204
                          04   13 0004A BEQL    2$
                23  A2        01   88 0004C BISB2   #1, 35(FCB)                                 ; 1206
                    50        68   D0 00050 2$:     MOVL    (R8), R0                            ; 1212
    55      0B  A0            07   EF 00053 EXTZV   #7, #1, 11(R0), R5                          ; 1213
    51          64            15   EF 00059 EXTZV   #21, #1, (FIB), R1
                55            51   CA 0005E BICL2   R1, R5
                              53   D4 00061 CLRL    R3                                          ; 1214
                    28    A0   D5 00063 TSTL    40(R0)
                          02   13 00066 BEQL    3$
                          53   D6 00068 INCL    R3
                              51   D4 0006A 3$:     CLRL    R1                                  ; 1215
                    24    A0   D5 0006C TSTL    36(R0)
                          02   13 0006F BEQL    4$
                          51   D6 00071 INCL    R1
                50            53   C8 00073 4$:     BISL2   R3, R1
                              50   D4 00076 CLRL    R0                                          ; 1216
                    3C    A2   D5 00078 TSTL    60(FCB)
                          02   12 0007B BNEQ    5$
                          50   D6 0007D INCL    R0
                50            51   C8 0007F 5$:     BISL2   R1, R0
                57            55   D2 00082 MCOML   R5, DO_EXPIRE                               ; 1214
          57                  57   CB 00085 BICL3   DO_EXPIRE, R0, DO_EXPIRE
                50            68   D0 00089 MOVL    (R8), R0                                    ; 1217
    55      0B  A0            01   EF 0008C EXTZV   #1, #1, 11(R0), R5                          ; 1218
    51          64            15   EF 00092 EXTZV   #21, #1, (FIB), R1
                55            51   CA 00097 BICL2   R1, R5
```

```
                                   53  D4 0009A         CLRL    R3                                     ; 1219
                           28      A0  D5 0009C         TSTL    40(R0)                                 ;
                                   02  13 0009F         BEQL    6$                                     ;
                                   53  D6 000A1         INCL    R3                                     ;
                   50      90      AA  D0 000A3 6$:     MOVL    -112(BASE), R0                         ; 1220
                                   51  D4 000A7         CLRL    R1                                     ;
                   05      32      A0  B1 000A9         CMPW    50(R0), #5                             ;
                                   02  1B 000AD         BLEQU   7$                                     ;
                                   51  D6 000AF         INCL    R1                                     ;
                   51              53  C8 000B1 7$:     BISL2   R3, R1                                 ;
50     17   A4     01              00  EF 000B4         EXTZV   #0, #1, 23(FIB), R0                    ; 1221
                   51              50  C8 000BA         BISL2   R0, R1                                 ;
                   53              55  D2 000BD         MCOML   R5, MODIFIED                           ; 1219
            53     51              53  CB 000C0         BICL3   MODIFIED, R1, MODIFIED                 ;
                   50              68  D0 000C4         MOVL    (R8), R0                               ; 1226
            20  0B A0              01  E0 000C7         BBS     #1, 11(R0), 9$                         ;
                   1D              57  E8 000CC         BLBS    DO_EXPIRE, 9$                          ; 1229
            12  22 A2              01  E0 000CF         BBS     #1, 34(FCB), 8$                        ; 1230
            0D  23 A2              01  E0 000D4         BBS     #1, 35(FCB), 8$                        ; 1231
                   09      23      A2  E8 000D9         BLBS    35(FCB), 8$                            ; 1232
            04  22 A2              02  E0 000DD         BBS     #2, 34(FCB), 8$                        ; 1233
                                   6A  95 000E2         TSTB    (BASE)                                 ; 1234
                                   24  18 000E4         BGEQ    10$                                    ;
                   01      18      A2  B1 000E6 8$:     CMPW    24(FCB), #1                            ; 1235
                                   1E  12 000EA         BNEQ    10$                                    ;
                                   52  DD 000EC 9$:     PUSHL   FCB                                    ; 1238
                                   7E  D4 000EE         CLRL    -(SP)                                  ;
             0000G CF              02  FB 000F0         CALLS   #2, READ_HEADER                        ;
                   55              50  D0 000F5         MOVL    R0, HEADER                             ;
                   0E      23      A2  E9 000F8         BLBC    35(FCB), 10$                           ; 1240
                                   24  BB 000FC         PUSHR   #^M<R2,R5>                             ; 1243
             0000G CF              02  FB 000FE         CALLS   #2, REBLD_PRIM_FCB                     ;
                                   55  DD 00103         PUSHL   HEADER                                 ; 1244
             0000G CF              01  FB 00105         CALLS   #1, BUILD_EXT_FCBS                     ;
                   01      18      A2  B1 0010A 10$:    CMPW    24(FCB), #1                            ; 1252
                                   2C  12 0010E         BNEQ    13$                                    ;
            0F  22 A2              01  E1 00110         BBC     #1, 34(FCB), 11$                       ; 1255
                           48      A2  DD 00115         PUSHL   72(FCB)                                ; 1261
                   6B              01  FB 00118         CALLS   #1, LOCK_COUNT                         ;
                   01              50  D1 0011B         CMPL    R0, #1                                 ;
                                   04  12 0011E         BNEQ    11$                                    ;
                02 AA              20  88 00120         BISB2   #32, 2(BASE)                           ; 1263
                                   6A  95 00124 11$:    TSTB    (BASE)                                 ; 1265
                                   03  18 00126         BGEQ    12$                                    ;
                   6A              04  88 00128         BISB2   #4, (BASE)                             ; 1266
            0C  22 A2              02  E1 0012B 12$:    BBC     #2, 34(FCB), 13$                       ; 1271
                35 A5      40      8F  88 00130         BISB2   #64, 53(HEADER)                        ; 1274
                                   55  DD 00135         PUSHL   HEADER                                 ; 1275
             0000G CF              01  FB 00137         CALLS   #1, MARK_DIRTY                         ;
                   03              53  E8 0013C 13$:    BLBS    MODIFIED, 14$                          ; 1282
                   09              57  E9 0013F         BLBC    DO_EXPIRE, 15$                         ; 1283
                                   53  DD 00142 14$:    PUSHL   MODIFIED                               ; 1284
                                   55  DD 00144         PUSHL   HEADER                                 ;
             0000V CF              02  FB 00146         CALLS   #2, SET_REVISION                       ;
                   50              68  D0 0014B 15$:    MOVL    (R8), R0                               ; 1292
            03  0B A0              01  E0 0014E         BBS     #1, 11(R0), 16$                        ;
                                 00DD  31 00153         BRW     27$                                    ;
```

```
                                                55  DD 00156 16$:   PUSHL   HEADER                          ; 1295
                                      0000G     CF                 CALLS   #1, MARK_DIRTY
                                                01  FB 00158
                                                28  65 91 0015D    CMPB    (HEADER), #40                   ; 1319
                                                    1B 1F 00160    BLSSU   19$
                                                50      98  AA D0 00162    MOVL    -104(BASE), R0          ; 1321
                                           05       53  A0 04 E0 00166    BBS     #4, 83(R0), 17$
                                                        44 A2 D5 0016B    TSTL    68(FCB)                 ; 1322
                                                           08 12 0016E    BNEQ    18$
                                           4C   A5      38 A2 C1 C1 00170 17$:   ADDL3   #1, 56(FCB), 76(HEADER) ; 1324
                                                           05 11 00176    BRB     19$
                                                4C     A5      44 A2 D0 00178 18$:   MOVL    68(FCB), 76(HEADER) ; 1326
                                                50             68 D0 0017D 19$:   MOVL    (R8), R0         ; 1329
                                           05       14  A0 01 E1 00180    BBC     #1, 20(R0), 20$
                                                34     A5      40 8F 8B 00185    BISB2   #64, 52(HEADER)  ; 1330
                                                50             90 AA D0 0018A 20$:   MOVL    -112(BASE), R0   ; 1332
                                                05             32 A0 B1 0018E    CMPW    50(R0), #5
                                                               17 1B 00192    BLEQU   21$
                                                14             66 E9 00194    BLBC    (R6), 21$        ; 1333
                                                               7E D4 00197    CLRL    -(SP)            ; 1336
                                                         0220  8F BB 00199    PUSHR   #^M<R5,R9>
                                                0000G    CF                 CALLS   #3, WRITE_ATTRIB
                                                55      04  AA D0 001A2    MOVL    4(BASE), HEADER  ; 1337
                                                  34    A5      40 8F 8A 001A6    BICB2   #64, 52(HEADER)  ; 1338
                                                7D             17 A4 E9 001AB 21$:   BLBC    23(FIB), 25$     ; 1346
                                           78       22  A2 01 E0 001AF    BBS     #1, 34(FCB), 25$         ; 1347
                                                50             98 AA D0 001B4    MOVL    -104(BASE), R0   ; 1350
                                           05       0B  A0 04 E1 001B8    BBC     #4, 11(R0), 22$
                                                         025C  8F BF 001BD    CHMU    #604                     ; 1351
                                                               04 001C1    RET
                                                01             18 A2 B1 001C2 22$:   CMPW    24(FCB), #1      ; 1353
                                                               29 12 001C6    BNEQ    23$
                                                              48 A2 DD 001C8    PUSHL   72(FCB)          ; 1354
                                                         6B                 CALLS   #1, LOCK_COUNT
                                                01             50 D1 001CE    CMPL    R0, #1
                                                               1E 12 001D1    BNEQ    23$
                                                               55 DD 001D3    PUSHL   HEADER           ; 1358
                                                0000G    CF                 CALLS   #1, CHECKSUM
                                                              1C A4 DD 001DA    PUSHL   28(FIB)          ; 1359
                                                               30 BB 001DD    PUSHR   #^M<R4,R5>
                                                0000G    CF                 CALLS   #3, TRUNCATE
                                                         6A                 BICB2   #2, (BASE)       ; 1360
                                                              04 AA DD 001E7    PUSHL   4(BASE)          ; 1361
                                                0000G    CF                 CALLS   #1, UPDATE_FCB
                                                               3B 11 001EF    BRB     25$              ; 1353
                                                01             1C A2 B1 001F1 23$:   CMPW    28(FCB), #1      ; 1367
                                                               37 12 001F5    BNEQ    26$
                                           11       22  A2 03 E0 001F7    BBS     #3, 34(FCB), 24$         ; 1365
                                                              1E A2 B5 001FC    TSTW    30(FCB)          ; 1366
                                                               0C 12 001FF    BNEQ    24$
                                                               52 DD 00201    PUSHL   FCB              ; 1367
                                                               04 DD 00203    PUSHL   #4
                                                0000G    CF                 CALLS   #2, CONV_ACCLOCK
                                                21             50 E9 0020A    BLBC    R0, 26$
                                                53             1C A4 D0 0020D 24$:   MOVL    28(FIB), TRNVBN  ; 1382
                                                50               54 7D 00211    MOVQ    FIB, R0          ; 1383
                                                           0000G 30 00214    BSBW    TRUNC_CHECKS
                                                  23     A2      02 88 00217    BISB2   #2, 35(FCB)      ; 1392
```

```
                50   A2              53   D0 0021B          MOVL    TRNVBN, 80(FCB)                 ; 1393
          1C    38   A2              01   C1 0021F          ADDL3   #1, 56(FCB), 28(FIB)            ; 1394
                                     52   DD 00225          PUSHL   FCB                             ; 1400
              0000G    CF            01   FB 00227          CALLS   #1, MAKE_FCB_STALE
                                     49   11 0022C 25$:     BRB     28$                             ; 1364
                          0800       8F   BF 0022E 26$:     CHMU    #2048                           ; 1403
                                     04   00232            RET
                3F    23   A2        01   E1 00233 27$:     BBC     #1, 35(FCB), 28$                ; 1417
                           01   18    A2   B1 00238          CMPW    24(FCB), #1                     ; 1420
                                     39   12 0023C          BNEQ    28$
                34    22   A2        01   E0 0023E          BBS     #1, 34(FCB), 28$                ; 1421
                               50    A2   D5 00243          TSTL    80(FCB)                         ; 1422
                                     2F   13 00246          BEQL    28$
                               48    A2   DD 00248          PUSHL   72(FCB)                         ; 1423
                           6B              01   FB 0024B          CALLS   #1, LOCK_COUNT
                           01              50   D1 0024E          CMPL    R0, #1
                                           24   12 00251          BNEQ    28$
                           53        04    A6   D0 00253          MOVL    4(R6), SAVE_US1         ; 1426
                                           55   DD 00257          PUSHL   HEADER                  ; 1427
              0000G    CF                  01   FB 00259          CALLS   #1, CHECKSUM
                           6D      0000V    CF   9E 0025E          MOVAB   TRUNC_HANDLER, (FP)     ; 1429
                                     50    A2   DD 00263          PUSHL   80(FCB)                 ; 1430
                                           55   DD 00266          PUSHL   HEADER
                                     0244  CA   9F 00268          PUSHAB  580(BASE)
              0000G    CF                  03   FB 0026C          CALLS   #3, TRUNCATE
                                           6D   D4 00271          CLRL    (FP)                    ; 1431
                             04    A6            53   D0 00273          MOVL    SAVE_US1, 4(R6)         ; 1433
                                           50   D4 00277 28$:     CLRL    R0                      ; 1440
                                           04   00279            RET                             ; 1442

; Routine Size:  634 bytes,    Routine Base:  $CODE$ + 0000

;  444          1443  1
```

```
;  446          1444  1 ROUTINE TRUNC_HANDLER (SIGNAL, MECHANISM) =
;  447          1445  1
;  448          1446  1 !++
;  449          1447  1 !
;  450          1448  1 !--
;  451          1449  1
;  452          1450  2 BEGIN
;  453          1451  2
;  454          1452  2 MAP
;  455          1453  2         SIGNAL          : REF BBLOCK,
;  456          1454  2         MECHANISM       : REF BBLOCK;
;  457          1455  2
;  458          1456  2 IF .SIGNAL [CHF$L_SIG_NAME] EQL SS$_CMODUSER
;  459          1457  2 THEN
;  460          1458  2     $UNWIND (DEPADR = MECHANISM [CHF$L_MCH_DEPTH]);
;  461          1459  2
;  462          1460  2 SS$_RESIGNAL
;  463          1461  1 END;


                                                        .EXTRN  SYS$UNWIND

                                        0000 00000 TRUNC_HANDLER:
                                                        .WORD   Save nothing                    ;  1444
                        50      04  AC  D0 00002         MOVL    SIGNAL, R0                      ;  1456
            00000424    8F      04  A0  D1 00006         CMPL    4(R0), #1060                    ;
                                    0E  12 0000E         BNEQ    1$                              ;
                                    7E  D4 00010         CLRL    -(SP)                           ;  1458
                7E      08  AC      08  C1 00012         ADDL3   #8, MECHANISM, -(SP)            ;
            00000000G   00          02  FB 00017         CALLS   #2, SYS$UNWIND                  ;
                        50  0918  8F  3C 0001E 1$:       MOVZWL  #2328, R0                       ;  1461
                                    04 00023            RET                                     ;

; Routine Size:  36 bytes,    Routine Base:  $CODE$ + 027A
```

```
;  465     1462  1 GLOBAL ROUTINE SET_REVISION (HEADER, MODE) : L_NORM NOVALUE =
;  466     1463  1
;  467     1464  1 !++
;  468     1465  1 !
;  469     1466  1 ! FUNCTIONAL DESCRIPTION:
;  470     1467  1 !
;  471     1468  1 !       This routine updates the revision count and date, and the
;  472     1469  1 !       expiration date in the file header as specified.
;  473     1470  1 !
;  474     1471  1 ! CALLING SEQUENCE:
;  475     1472  1 !       SET_REVISION (HEADER, MODE)
;  476     1473  1 !
;  477     1474  1 ! INPUT PARAMETERS:
;  478     1475  1 !       HEADER: address of file header to operate on
;  479     1476  1 !       MODE: 0 to just update expiration date
;  480     1477  1 !             1 to set revision and expiration date
;  481     1478  1 !             3 to do above and clear backup date
;  482     1479  1 !
;  483     1480  1 ! IMPLICIT INPUTS:
;  484     1481  1 !       NONE
;  485     1482  1 !
;  486     1483  1 ! OUTPUT PARAMETERS:
;  487     1484  1 !       NONE
;  488     1485  1 !
;  489     1486  1 ! IMPLICIT OUTPUTS:
;  490     1487  1 !       NONE
;  491     1488  1 !
;  492     1489  1 ! ROUTINE VALUE:
;  493     1490  1 !       NONE
;  494     1491  1 !
;  495     1492  1 ! SIDE EFFECTS:
;  496     1493  1 !       file header modified and marked dirty
;  497     1494  1 !
;  498     1495  1 !--
;  499     1496  1
;  500     1497  2 BEGIN
;  501     1498  2
;  502     1499  2 LABEL
;  503     1500  2         CHECK_EXPIRE;                   ! check file expiration date
;  504     1501  2
;  505     1502  2 MAP
;  506     1503  2         HEADER          : REF BBLOCK,   ! file header
;  507     1504  2         MODE            : BITVECTOR;    ! routine mode flags
;  508     1505  2
;  509     1506  2 LOCAL
;  510     1507  2         DAY_TIME        : VECTOR [2],   ! time of day
;  511     1508  2         DAY_TIME2       : VECTOR [2],   ! time of day
;  512     1509  2         UCB             : REF BBLOCK,   ! UCB of RVN 1
;  513     1510  2         PRIMARY_VCB     : REF BBLOCK,   ! VCB of RVN 1
;  514     1511  2         IDENT_AREA      : REF BBLOCK;   ! header ident area
;  515     1512  2
;  516     1513  2 BIND_COMMON;
;  517     1514  2
;  518     1515  2 EXTERNAL ROUTINE
;  519     1516  2         MARK_DIRTY      : L_NORM;       ! mark buffer for write-back
;  520     1517  2
;  521     1518  2
```

```
;  522          1519  2 ! Locate the ident area and check that the date fields are present.
;  523          1520  2 !
;  524          1521  2
;  525          1522  2 IDENT_AREA = .HEADER + .HEADER[FH2$B_IDOFFSET]*2;
;  526          1523  2 IF .HEADER[FH2$B_MPOFFSET] - .HEADER[FH2$B_IDOFFSET] LSSU
;  527          1524  2     ($BYTEOFFSET (FI2$Q_EXPDATE) + FI2$S_EXPDATE) / 2
;  528          1525  2 THEN RETURN;
;  529          1526  2
;  530          1527  2 ! Update the expiration date of the file.
;  531          1528  2 !
;  532          1529  2
;  533          1530  2 MARK_DIRTY (.HEADER);
;  534          1531  3 CHECK_EXPIRE: BEGIN
;  535          1532  3 PRIMARY_VCB = .CURRENT_VCB;
;  536          1533  3 IF .PRIMARY_VCB[VCB$W_RVN] NEQ 0
;  537          1534  3 THEN
;  538          1535  4     BEGIN
;  539          1536  4     UCB = .VECTOR [CURRENT_RVT[RVT$L_UCBLST], 0];
;  540          1537  4     IF .UCB EQL 0
;  541          1538  4     THEN LEAVE CHECK_EXPIRE;
;  542          1539  4     PRIMARY_VCB = .UCB[UCB$L_VCB];
;  543          1540  3     END;
;  544          1541  3
;  545          1542  3 $GETTIM (TIMADR = DAY_TIME);
;  546          1543  3 IF .(PRIMARY_VCB[VCB$Q_RETAINMAX]+4) NEQ 0
;  547          1544  3 THEN
;  548          1545  4     BEGIN
;  549          1546  4     SUBQ (PRIMARY_VCB[VCB$Q_RETAINMAX], DAY_TIME, DAY_TIME2);
;  550          1547  5     IF CMPQ (IDENT_AREA[FI2$Q_EXPDATE], GEQ, DAY_TIME2)
;  551          1548  4     THEN LEAVE CHECK_EXPIRE;
;  552          1549  4     CH$MOVE (8, DAY_TIME2, IDENT_AREA[FI2$Q_EXPDATE]);
;  553          1550  3     END;
;  554          1551  2 END;                                ! end of block CHECK_EXPIRE
;  555          1552  2
;  556          1553  2 ! Increment the revision count of the file if specified.
;  557          1554  2 !
;  558          1555  2
;  559          1556  2 IF .MODE[0]
;  560          1557  2 THEN
;  561          1558  3     BEGIN
;  562          1559  3     IDENT_AREA[FI2$W_REVISION] = .IDENT_AREA[FI2$W_REVISION] + 1;
;  563          1560  3     CH$MOVE (8, DAY_TIME, IDENT_AREA[FI2$Q_REVDATE]);
;  564          1561  2     END;
;  565          1562  2
;  566          1563  2 ! Clear the backup date if requested.
;  567          1564  2 !
;  568          1565  2
;  569          1566  2 IF .MODE[1]
;  570          1567  2 THEN
;  571          1568  3     BEGIN
;  572          1569  3     (IDENT_AREA[FI2$Q_BAKDATE]) = 0;
;  573          1570  3     (IDENT_AREA[FI2$Q_BAKDATE])+4 = 0;
;  574          1571  2     END;
;  575          1572  2
;  576          1573  1 END;                                    ! end of routine SET_REVISION
```

```
                                                        .EXTRN  SYS$GETTIM

                                        007C 00000      .ENTRY  SET_REVISION, Save R2,R3,R4,R5,R6                ; 1462
                                5E        10  C2 00002     SUBL2   #16, SP
                                51    04  AC  D0 00005     MOVL    HEADER, R1                                   ; 1522
                                50        61  9A 00009     MOVZBL  (R1), R0
                                56      6140  3E 0000C     MOVAW   (R1)[R0], IDENT_AREA
                                50    01  A1  9A 00010     MOVZBL  1(R1), R0                                    ; 1523
                                52        61  9A 00014     MOVZBL  (R1), R2
                                50        52  C2 00017     SUBL2   R2, R0
                                17        50  D1 0001A     CMPL    R0, #23                                      ; 1524
                                          7D  1F 0001D     BLSSU   9$
                                51        DD  1F 0001F     PUSHL   R1                                           ; 1530
                          0000G CF        01  FB 00021     CALLS   #1, MARK_DIRTY
                                52    98  AA  D0 00026     MOVL    -104(BASE), PRIMARY_VCB                      ; 1532
                                      0E  A2  B5 0002A     TSTW    14(PRIMARY_VCB)                              ; 1533
                                          0E  13 0002D     BEQL    1$
                                50    9C  AA  D0 0002F     MOVL    -100(BASE), R0                               ; 1536
                                50    44  A0  D0 00033     MOVL    68(R0), UCB
                                          4E  13 00037     BEQL    7$                                           ; 1537
                                52    34  A0  D0 00039     MOVL    52(UCB), PRIMARY_VCB                         ; 1539
                                      08  AE  9F 0003D 1$:  PUSHAB  DAY_TIME                                    ; 1542
                    00000000G   00        01  FB 00040     CALLS   #1, SYS$GETTIM
                                      78  A2  D5 00047     TSTL    120(PRIMARY_VCB)                             ; 1543
                                          3B  13 0004A     BEQL    7$
                          6E    08  AE    74  A2  C3 0004C     SUBL3   116(PRIMARY_VCB), DAY_TIME, DAY_TIME2    ; 1546
                                04  AE    0C  AE  D0 00052     MOVL    DAY_TIME, DAY_TIME2
                                04  AE    78  A2  D9 00057     SBWC    120(PRIMARY_VCB), DAY_TIME2
                                50            01  CE 0005C     MNEGL   #1, R0                                   ; 1547
                                04  AE    2A  A6  D1 0005F     CMPL    42(IDENT_AREA), DAY_TIME2
                                          0E  19 00064     BLSS    4$
                                          08  14 00066     BGTR    2$
                                      6E      26  A6  D1 00068     CMPL    38(IDENT_AREA), DAY_TIME2
                                              04  13 0006C     BEQL    3$
                                              04  1F 0006E     BLSSU   4$
                                              50  D6 00070 2$:  INCL    R0
                                              50  D6 00072 3$:  INCL    R0
                                02 FFFFFFFF   8F      50  CF 00074 4$:  CASEL   R0, #-1, #2
                                0008              0006      0007C 5$:  .WORD   6$-5$,-
                                                                       7$-5$,-
                                                                       7$-5$
                          26    A6      6E          08  28 00082 6$:  MOVC3   #8, DAY_TIME2, 38(IDENT_AREA)   ; 1549
                                      09      08    AC  E9 00087 7$:  BLBC    MODE, 8$                         ; 1556
                                              14    A6  B6 00088      INCW    20(IDENT_AREA)                   ; 1559
                          1E    A6      08    AE      08  28 0008E     MOVC3   #8, DAY_TIME, 30(IDENT_AREA)     ; 1560
                                03      08    AC      01  E1 00094 8$:  BBC     #1, MODE, 9$                     ; 1566
                                                  2E  A6  7C 00099      CLRQ    46(IDENT_AREA)                   ; 1569
                                                      04 0009C 9$:      RET                                      ; 1573

; Routine Size:  157 bytes,    Routine Base:  $CODE$ + 029E

;  577          1574  1
;  578          1575  1 END
;  579          1576  0 ELUDOM
```

```
;                                PSECT SUMMARY

;       Name                    Bytes                           Attributes

;  $CODE$                         827  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                              Library Statistics

;                                        --------  Symbols  --------      Pages       Processing
;       File                             Total     Loaded   Percent       Mapped      Time

;  _$255$DUA18:[SYSLIB]LIB.L32;1          18619         76         0         1000        00:02.0


;                                COMMAND QUALIFIERS

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

; Size:          827 code + 0 data bytes
; Run Time:          00:37.1
; Elapsed Time:      00:53.9
; Lines/CPU Min:      2551
; Lexemes/CPU-Min:  45464
; Memory Used:   345 pages
; Compilation Complete
```

0443 AH-EF71A-SE
VAX/VMS V4.1 SRC LST MCRF UPD

DELBAD
LIS

QUOTAUTIL
LIS

CREHDR
LIS

RDBLOK
LIS

MAKACC
LIS

CREATE
LIS

DEACCS
LIS

EXTHDR
LIS