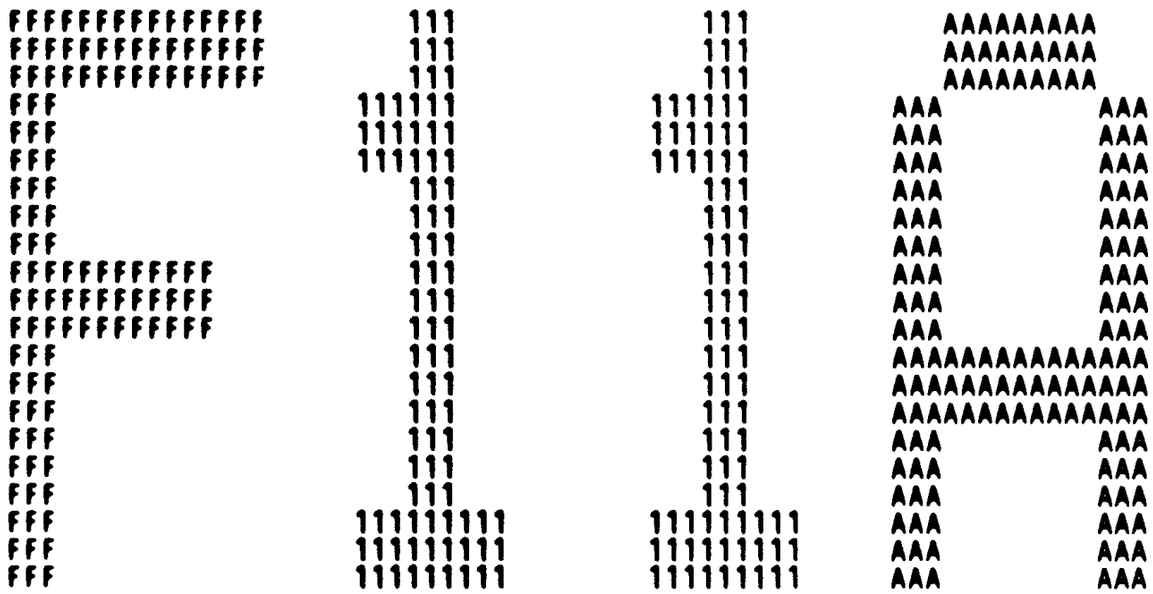
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
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**TRUNC

```
TTTTTTTTTT  RRRRRRRR    UU      UU  NN      NN    CCCCCCCC
TTTTTTTTTT  RRRRRRRR    UU      UU  NN      NN    CCCCCCCC
    TT      RR      RR  UU      UU  NN      NN  CC
    TT      RR      RR  UU      UU  NN      NN  CC
    TT      RR      RR  UU      UU  NNNN    NN  CC
    TT      RR      RR  UU      UU  NNNN    NN  CC
    TT      RRRRRRRR    UU      UU  NN  NN  NN  CC
    TT      RRRRRRRR    UU      UU  NN  NN  NN  CC
    TT      RR  RR      UU      UU  NN    NNNN  CC
    TT      RR  RR      UU      UU  NN    NNNN  CC
    TT      RR    RR    UU      UU  NN      NN  CC              ....
    TT      RR    RR    UU      UU  NN      NN  CC              ....
    TT      RR      RR  UUUUUUUUUU  NN      NN    CCCCCCCC      ....
    TT      RR      RR  UUUUUUUUUU  NN      NN    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
```

TRU
V04

;

;

; S
; R
; E
; L
; L
; M
; C

```
;     1          0001  0 MODULE TRUNC (
;     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 truncates a file by deallocating the indicated blocks.
;    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:  21-Mar-1977  10:41
;    48          0048  1 !
;    49          0049  1 ! MODIFIED BY:
;    50          0050  1 !
;    51          0051  1 !       V03-002 ACG0287         Andrew C. Goldstein,    14-Apr-1982  17:18
;    52          0052  1 !               Check for index file ID in header rather than FCB
;    53          0053  1 !
;    54          0054  1 !       V03-001 LMP0023         L. Mark Pilant,          7-Apr-1982  16:45
;    55          0055  1 !               Give a privilege violation if attempting to truncate the
;    56          0056  1 !               index file (INDEXF.SYS).
;    57          0057  1 !
```

```
;    58          0058  1 !       V02-003 ACG35898        Andrew C. Goldstein,    11-Mar-1981  16:31
;    59          0059  1 !               Update HIBLK in primary file header
;    60          0060  1 !
;    61          0061  1 !       V02-002 ACG0167         Andrew C. Goldstein,    7-May-1980  18:52
;    62          0062  1 !               Previous revision history moved to F11A.REV
;    63          0063  1 !**
;    64          0064  1
;    65          0065  1
;    66          0066  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    67          0067  1 REQUIRE 'SRC$:FCPDEF.B32';
;    68          0382  1
;    69          0383  1
;    70          0384  1 FORWARD ROUTINE
;    71          0385  1         TRUNCATE        : NOVALUE,      ! truncate file
;    72          0386  1         TRUNCATE_HEADER : NOVALUE;      ! truncate individual file header
```

```
;    74          0387  1 GLOBAL ROUTINE TRUNCATE (FIB, FILE_HEADER, DEALLOCATE) : NOVALUE =
;    75          0388  1
;    76          0389  1 !++
;    77          0390  1 !
;    78          0391  1 ! FUNCTIONAL DESCRIPTION:
;    79          0392  1 !
;    80          0393  1 !       This routine truncates a file to the size indicated in the FIB by
;    81          0394  1 !       deallocating the necessary blocks. The erase flag controls whether
;    82          0395  1 !       the retrieval pointers are erased in the header. The deallocate flag
;    83          0396  1 !       controls whether or not the blocks are actually returned to the
;    84          0397  1 !       storage map.
;    85          0398  1 !
;    86          0399  1 ! CALLING SEQUENCE:
;    87          0400  1 !       TRUNCATE (ARG1, ARG2, ARG3)
;    88          0401  1 !
;    89          0402  1 ! INPUT PARAMETERS:
;    90          0403  1 !       ARG1: address of FIB for operation
;    91          0404  1 !       ARG2: address of file header
;    92          0405  1 !       ARG3: = 1 to return space to storage map
;    93          0406  1 !             = 0 to just clean out header
;    94          0407  1 !
;    95          0408  1 ! IMPLICIT INPUTS:
;    96          0409  1 !       CURRENT_VCB: VCB of volume
;    97          0410  1 !
;    98          0411  1 ! OUTPUT PARAMETERS:
;    99          0412  1 !       NONE
;   100          0413  1 !
;   101          0414  1 ! IMPLICIT OUTPUTS:
;   102          0415  1 !       NONE
;   103          0416  1 !
;   104          0417  1 ! ROUTINE VALUE:
;   105          0418  1 !       NONE
;   106          0419  1 !
;   107          0420  1 ! SIDE EFFECTS:
;   108          0421  1 !       storage bitmap altered
;   109          0422  1 !       file header altered
;   110          0423  1 !
;   111          0424  1 !--
;   112          0425  1
;   113          0426  2 BEGIN
;   114          0427  2
;   115          0428  2 MAP
;   116          0429  2         FIB             : REF BBLOCK,   ! FIB for operation
;   117          0430  2         FILE_HEADER     : REF BBLOCK;   ! file header
;   118          0431  2
;   119          0432  2 LOCAL
;   120          0433  2         FCB             : REF BBLOCK,   ! FCB of current file header
;   121          0434  2         HEADER          : REF BBLOCK,   ! address of current file header
;   122          0435  2         NEW_HEADER      : REF BBLOCK,   ! address of extension file header
;   123          0436  2         MAP_AREA        : REF BBLOCK,   ! file header map area
;   124          0437  2         MAP_POINTER     : REF BBLOCK,   ! pointer to scan map
;   125          0438  2         MAP_COUNT,                      ! retrieval pointer count
;   126          0439  2         VBN,                            ! relative VBN of operation
;   127          0440  2         COUNT,                          ! count of blocks returned
;   128          0441  2         EXT_FID         : BBLOCK [FID$C_LENGTH], ! file ID of extension header
;   129          0442  2         EX_SEGNUM,                      ! segment number of ext header
;   130          0443  2         REREAD,                         ! flag to reread primary header
```

```
;  131          0444  2          REREAD2;                        ! flag to update primary header
;  132          0445  2
;  133          0446  2 LABEL
;  134          0447  2          DO_TRUNCATE,                    ! main truncate operation body
;  135          0448  2          VBN_LOOP;                       ! main loop to scan for starting VBN
;  136          0449  2
;  137          0450  2 EXTERNAL
;  138          0451  2          CLEANUP_FLAGS    : BITVECTOR,   ! cleanup action flags
;  139          0452  2          USER_STATUS      : VECTOR,      ! I/O status block of user
;  140          0453  2          CURRENT_VCB      : REF BBLOCK,  ! VCB of volume
;  141          0454  2          PRIMARY_FCB      : REF BBLOCK;  ! FCB of file
;  142          0455  2
;  143          0456  2 EXTERNAL ROUTINE
;  144          0457  2          PMS_START_SUB,                  ! start subfunction metering
;  145          0458  2          PMS_END_SUB,                    ! end subfunction metering
;  146          0459  2          SEARCH_FCB,                     ! search FCB list for FCB
;  147          0460  2          MARK_DIRTY,                     ! mark buffer for write-back
;  148          0461  2          CHECKSUM,                       ! checksum file header
;  149          0462  2          NEXT_HEADER,                    ! read next extension header
;  150          0463  2          UPDATE_FILESIZE,                ! update file size in primary FCB
;  151          0464  2          INIT_FCB,                       ! initialize FCB
;  152          0465  2          WRITE_HEADER,                   ! write file header
;  153          0466  2          READ_HEADER,                    ! read file header
;  154          0467  2          DEL_EXTFCB,                     ! delete extension FCB's
;  155          0468  2          DELETE_FILE;                    ! delete remainder of file
;  156          0469  2
;  157          0470  2
;  158          0471  2 ! Start metering for this subfunction.
;  159          0472  2 !
;  160          0473  2
;  161          0474  2 PMS_START_SUB (PMS_ALLOC);
;  162          0475  2
;  163          0476  2 ! Establish the basic pointers. Round up the starting VBN to the next cluster
;  164          0477  2 ! boundary and adjust it to a zero start. The block count must be zero (default).
;  165          0478  2 !
;  166          0479  2
;  167          0480  2 HEADER = .FILE_HEADER;
;  168          0481  2 FCB = .PRIMARY_FCB;
;  169          0482  2
;  170          0483  2 ! Check for the index file INDEXF.SYS
;  171          0484  2 !
;  172          0485  2
;  173          0486  2 IF  .HEADER[FH1$W_FID_NUM] EQL FID$C_INDEXF
;  174          0487  2 AND .HEADER[FH1$W_FID_SEQ] EQL FID$C_INDEXF
;  175          0488  2 THEN ERR_EXIT (SS$_NOPRIV);
;  176          0489  2
;  177          0490  4 VBN = ((.FIB[FIB$L_EXVBN] - 1 + .CURRENT_VCB[VCB$W_CLUSTER] - 1)
;  178          0491  2           / .CURRENT_VCB[VCB$W_CLUSTER]) * .CURRENT_VCB[VCB$W_CLUSTER];
;  179          0492  2
;  180          0493  2 IF .FIB[FIB$L_EXSZ] NEQ 0 THEN ERR_EXIT (SS$_BADPARAM);
;  181          0494  2
;  182          0495  2 ! Init the user's return parameters.
;  183          0496  2 !
;  184          0497  2
;  185          0498  2 USER_STATUS[1] = .VBN + 1 - .FIB[FIB$L_EXVBN];
;  186          0499  2 FIB[FIB$L_EXVBN] = .VBN + 1;
;  187          0500  2 FIB[FIB$L_EXSZ] = 0;
```

```
;  188          0501  2 REREAD = 0;
;  189          0502  2
;  190          0503  2 ! Now scan the file headers for the retrieval pointer containing the starting
;  191          0504  2 ! VBN. If the VBN is off the end of file, report the error; if it coincides,
;  192          0505  2 ! the operation is a noop.
;  193          0506  2 !
;  194          0507  2
;  195          0508  2 DO_TRUNCATE:
;  196          0509  3 BEGIN
;  197          0510  3
;  198          0511  3 VBN_LOOP:
;  199          0512  4 BEGIN
;  200          0513  4 WHILE 1 DO
;  201          0514  5     BEGIN
;  202          0515  5     MAP_AREA = .HEADER + .HEADER[FH1$B_MPOFFSET]*2;
;  203          0516  5     MAP_POINTER = .MAP_AREA + FM1$C_POINTERS;
;  204          0517  5     MAP_COUNT = .MAP_AREA[FM1$B_INUSE] / 2;
;  205          0518  5
;  206          0519  5     WHILE .MAP_COUNT GTR 0 DO
;  207          0520  6         BEGIN
;  208          0521  6         COUNT = .MAP_POINTER[FM1$B_COUNT] + 1;
;  209          0522  6         IF .COUNT GTRU .VBN THEN LEAVE VBN_LOOP;
;  210          0523  6         VBN = .VBN - .COUNT;
;  211          0524  6         MAP_POINTER = .MAP_POINTER + 4;
;  212          0525  6         MAP_COUNT = .MAP_COUNT - 1;
;  213          0526  5         END;
;  214          0527  5
;  215          0528  5 ! We have scanned through an entire header. Chain to the next header if it
;  216          0529  5 ! exists. First check for the boundary condition of starting the truncate
;  217          0530  5 ! at the header boundary to avoid leaving an empty header.
;  218          0531  5 !
;  219          0532  5
;  220          0533  5     IF .VBN EQL 0
;  221          0534  5     THEN
;  222          0535  6         BEGIN
;  223          0536  6         IF .MAP_AREA[FM1$W_EX_FILNUM] EQL 0
;  224          0537  6         THEN LEAVE DO_TRUNCATE          ! truncate coincides with EOF
;  225          0538  6         ELSE LEAVE VBN_LOOP;            ! dump the extension headers
;  226          0539  5         END;
;  227          0540  5
;  228          0541  5     NEW_HEADER = NEXT_HEADER (.HEADER, .FCB);
;  229          0542  5     IF .NEW_HEADER EQL 0 THEN EXITLOOP;
;  230          0543  5     REREAD = 1;
;  231          0544  5     HEADER = .NEW_HEADER;
;  232          0545  5
;  233          0546  5     IF .FCB NEQ 0
;  234          0547  5     THEN FCB = .FCB[FCB$L_EXFCB]
;  235          0548  5     ELSE IF SEARCH_FCB (HEADER[FH1$W_FID]) NEQ 0
;  236          0549  5     THEN ERR_EXIT (SS$_ACCONFLICT);
;  237          0550  4     END;                                ! end of header scan loop
;  238          0551  4
;  239          0552  5 ERR_EXIT (SS$_ENDOFFILE)
;  240          0553  3 END;                                    ! end of VBN_LOOP
;  241          0554  3
;  242          0555  3 ! We are now pointing at the retrieval pointer in which the truncation starts.
;  243          0556  3 ! Clean out the rest of this header. If there are extension headers, use
;  244          0557  3 ! the delete routine to dispose of them. Also flush extension FCB's, if they
```

```
;  245          0558  3 ! exist.
;  246          0559  3 !
;  247          0560  3
;  248          0561  3 MARK_DIRTY (.HEADER);
;  249          0562  3 CLEANUP_FLAGS[CLF_FIXFCB] = 1;
;  250          0563  3 CLEANUP_FLAGS[CLF_INVWINDOW] = 1;
;  251          0564  3 CLEANUP_FLAGS[CLF_CLEANTRUNC] = 1;
;  252          0565  3 KERNEL_CALL (UPDATE_FILESIZE, .FIB[FIB$L_EXVBN] - 1);
;  253          0566  3
;  254          0567  3 ! Update the HIBLK field in the record attributes to reflect the new file
;  255          0568  3 ! size.
;  256          0569  3 !
;  257          0570  3
;  258          0571  3 IF NOT .REREAD
;  259          0572  3 THEN BBLOCK [HEADER[FH1$W_RECATTR], FAT$L_HIBLK] = ROT (.FIB[FIB$L_EXVBN]-1, 16);
;  260          0573  3
;  261          0574  3 TRUNCATE_HEADER (.FIB, .HEADER, .DEALLOCATE, .MAP_POINTER, .VBN);
;  262          0575  3 EX_SEGNUM = .MAP_AREA[FM1$B_EX_SEGNUM] + 1;
;  263          0576  3 EXT_FID[FID$W_NUM] = .MAP_AREA[FM1$W_EX_FILNUM];
;  264          0577  3 EXT_FID[FID$W_SEQ] = .MAP_AREA[FM1$W_EX_FILSEQ];
;  265          0578  3 EXT_FID[FID$W_RVN] = 0;
;  266          0579  3 MAP_AREA[FM1$W_EX_FILNUM] = 0;
;  267          0580  3 MAP_AREA[FM1$W_EX_FILSEQ] = 0;
;  268          0581  3
;  269          0582  3 IF .FCB NEQ 0 AND .FCB NEQ .PRIMARY_FCB
;  270          0583  3 THEN KERNEL_CALL (INIT_FCB, .FCB, .HEADER);
;  271          0584  3
;  272          0585  3 CHECKSUM (.HEADER);
;  273          0586  3 WRITE_HEADER ();
;  274          0587  3
;  275          0588  3 REREAD2 = .REREAD;
;  276          0589  4 IF .DEALLOCATE AND (.EXT_FID[FID$W_NUM] NEQ 0)
;  277          0590  3 THEN
;  278          0591  4     BEGIN
;  279          0592  4     REREAD = 1;
;  280          0593  4     HEADER = NEXT_HEADER (0, .FCB, EXT_FID, .EX_SEGNUM);
;  281          0594  4     KERNEL_CALL (DEL_EXTFCB, .FCB);
;  282          0595  4     DELETE_FILE (.FIB, .HEADER);
;  283          0596  3     END;
;  284          0597  3
;  285          0598  2 END;                                    ! end of block DO_TRUNCATE
;  286          0599  2
;  287          0600  2 ! If this was a truncate of a multi-header file, reread the primary header
;  288          0601  2 ! and update the HIBLK field in the record attributes to reflect the new file
;  289          0602  2 ! size.
;  290          0603  2 !
;  291          0604  2
;  292          0605  2 IF .REREAD
;  293          0606  2 THEN HEADER = READ_HEADER (FIB[FIB$W_FID], .PRIMARY_FCB);
;  294          0607  2
;  295          0608  2 IF .REREAD2
;  296          0609  2 THEN
;  297          0610  3     BEGIN
;  298          0611  3     BBLOCK [HEADER[FH1$W_RECATTR], FAT$L_HIBLK] = ROT (.FIB[FIB$L_EXVBN]-1, 16);
;  299          0612  3     MARK_DIRTY (.HEADER);
;  300          0613  2     END;
;  301          0614  2
```

```
;  302          0615  2
;  303          0616  2 ! Stop metering of this subfunction
;  304          0617  2 !
;  305          0618  2
;  306          0619  2 PMS_END_SUB ();
;  307          0620  2
;  308          0621  1 END;                                    ! end of routine TRUNCATE

                                                        .TITLE  TRUNC
                                                        .IDENT  \V04-000\

                                                        .EXTRN  CLEANUP_FLAGS, USER_STATUS
                                                        .EXTRN  CURRENT_VCB, PRIMARY_FCB
                                                        .EXTRN  PMS_START_SUB, PMS_END_SUB
                                                        .EXTRN  SEARCH_FCB, MARK_DIRTY
                                                        .EXTRN  CHECKSUM, NEXT_HEADER
                                                        .EXTRN  UPDATE_FILESIZE
                                                        .EXTRN  INIT_FCB, WRITE_HEADER
                                                        .EXTRN  READ_HEADER, DEC_EXTFCB
                                                        .EXTRN  DELETE_FILE, SYS$CMKRNL

                                                        .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000          .ENTRY  TRUNCATE, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-; 0387
                                                                R11
                          5B 00000000G  9F  9E 00002    MOVAB   @#SYS$CMKRNL, R11
                                     5E     08  C2 00009    SUBL2   #8, SP
                                            08  DD 0000C    PUSHL   #8                                      ; 0474
                              0000G  CF     01  FB 0000E    CALLS   #1, PMS_START_SUB
                                     55 08  AC  D0 00013    MOVL    FILE_HEADER, HEADER                     ; 0480
                                     56 0000G CF D0 00017   MOVL    PRIMARY_FCB, FCB                        ; 0481
                                     01 02  A5  B1 0001C    CMPW    2(HEADER), #1                           ; 0486
                                            09  12 00020    BNEQ    1$
                                     01 04  A5  B1 00022    CMPW    4(HEADER), #1                           ; 0487
                                            03  12 00026    BNEQ    1$
                                            24  BF 00028    CHMU    #36                                     ; 0488
                                            04  0002A      RET
                                     51 04  AC  D0 0002B 1$: MOVL    FIB, R1                                 ; 0490
                                     50 0000G CF D0 0002F   MOVL    CURRENT_VCB, R0
                                     52 3C  A0  3C 00034    MOVZWL  60(R0), R2
                                     52 1C  A1  C0 00038    ADDL2   28(R1), R2
                                     52     02  C2 0003C    SUBL2   #2, R2
                                     53 3C  A0  3C 0003F    MOVZWL  60(R0), R3                              ; 0491
                                     52     53  C6 00043    DIVL2   R3, R2
                                     54 3C  A0  3C 00046    MOVZWL  60(R0), VBN
                                     54     52  C4 0004A    MULL2   R2, VBN
                                        18  A1  D5 0004D    TSTL    24(R1)                                  ; 0493
                                            03  13 00050    BEQL    2$
                                            14  BF 00052    CHMU    #20
                                            04  00054      RET
                                     50 04  AC  D0 00055 2$: MOVL    FIB, R0                                 ; 0498
                              51     54 1C  A0  C3 00059    SUBL3   28(R0), VBN, R1
                              0000G  CF 01  A1  9E 0005E    MOVAB   1(R1), USER_STATUS+4
                                1C   A0 01  A4  9E 00064    MOVAB   1(R4), 28(R0)                           ; 0499
                                        18  A0  D4 00069    CLRL    24(R0)                                  ; 0500
                                            59  D4 0006C    CLRL    REREAD                                  ; 0501
```

```
                    50       01    A5  9A 0006E 3$:     MOVZBL  1(HEADER), R0                           ; 0515
                    52     6540    3E  00072         MOVAW   (HEADER)[R0], MAP_AREA
                    53       0A    A2  9E 00076         MOVAB   10(R2), MAP_POINTER                     ; 0516
                    58       08    A2  9A 0007A         MOVZBL  8(MAP_AREA), MAP_COUNT                  ; 0517
                    58              02  C6 0007E         DIVL2   #2, MAP_COUNT
                                    15  15 00081 4$:     BLEQ    5$                                      ; 0519
                    57       01    A3  9A 00083         MOVZBL  1(MAP_POINTER), COUNT                   ; 0521
                                    57  D6 00087         INCL    COUNT
                    54              57  D1 00089         CMPL    COUNT, VBN                              ; 0522
                                    49  1A 0008C         BGTRU   9$
                    54              57  C2 0008E         SUBL2   COUNT, VBN                              ; 0523
                    53              04  C0 00091         ADDL2   #4, MAP_POINTER                         ; 0524
                                    58  D7 00094         DECL    MAP_COUNT                               ; 0525
                                    E9  11 00096         BRB     4$                                      ; 0519
                                    54  D5 00098 5$:     TSTL    VBN                                     ; 0533
                                    08  12 0009A         BNEQ    6$
                            02      A2  B5 0009C         TSTW    2(MAP_AREA)                             ; 0536
                                    36  12 0009F         BNEQ    9$
                                 00D9   31 000A1         BRW     12$                                     ; 0537
                    7E              55  7D 000A4 6$:     MOVQ    HEADER, -(SP)                           ; 0541
          0000G     CF              02  FB 000A7         CALLS   #2, NEXT_HEADER
                    5A              50  D0 000AC         MOVL    R0, NEW_HEADER
                                    21  13 000AF         BEQL    8$                                      ; 0542
                    59              01  D0 000B1         MOVL    #1, REREAD                              ; 0543
                    55              5A  D0 000B4         MOVL    NEW_HEADER, HEADER                      ; 0544
                                    56  D5 000B7         TSTL    FCB                                     ; 0546
                                    06  13 000B9         BEQL    7$
                    56       0C     A6  D0 000BB         MOVL    12(FCB), FCB                            ; 0547
                                    AD  11 000BF         BRB     3$
                            02      A5  9F 000C1 7$:     PUSHAB  2(HEADER)                               ; 0548
          0000G     CF              01  FB 000C4         CALLS   #1, SEARCH_FCB
                                    50  D5 000C9         TSTL    R0
                                    A1  13 000CB         BEQL    3$
                                0800    8F  BF 000CD         CHMU    #2048                               ; 0549
                                        04  000D1         RET
                                0870    8F  BF 000D2 8$:     CHMU    #2160                               ; 0552
                                        04  000D6         RET
                                    55  DD 000D7 9$:     PUSHL   HEADER                                  ; 0561
                    0000G       CF      01  FB 000D9         CALLS   #1, MARK_DIRTY
                    0000G       CF  00080012    8F  C8 000DE         BISL2   #524306, CLEANUP_FLAGS          ; 0564
                                57      04      AC  D0 000E7         MOVL    FIB, R7                         ; 0565
                58       1C     A7              01  C3 000EB         SUBL3   #1, 28(R7), R8
                                                58  DD 000F0         PUSHL   R8
                                                01  DD 000F2         PUSHL   #1
                                                5E  DD 000F4         PUSHL   SP
                                            0000G   CF  9F 000F6         PUSHAB  UPDATE_FILESIZE
                                    6B              04  FB 000FA         CALLS   #4, SYS$CMKRNL
                                    05              59  E8 000FD         BLBS    REREAD, 10$             ; 0571
                12  A5              58              10  9C 00100         ROTL    #16, R8, 18(HEADER)     ; 0572
                                                    18  BB 00105 10$:     PUSHR   #^M<R3,R4>              ; 0574
                                            0C      AC  DD 00107         PUSHL   DEALLOCATE
                                                    55  DD 0010A         PUSHL   HEADER
                                                    57  DD 0010C         PUSHL   R7
                    0000V       CF                  05  FB 0010E         CALLS   #5, TRUNCATE_HEADER
                                    53              62  9A 00113         MOVZBL  (MAP_AREA), EX_SEGNUM   ; 0575
                                                    53  D6 00116         INCL    EX_SEGNUM
                                    6E      02      A2  D0 00118         MOVL    2(MAP_AREA), EXT_FID    ; 0576
```

```
                        04   AE  B4 0011C          CLRW    EXT_FID+4                                       ; 0578
                        02   A2  D4 0011F          CLRL    2(MAP_AREA)                                     ; 0579
                             56  D5 00122          TSTL    FCB                                             ; 0582
                             16  13 00124          BEQL    11$
                0000G   CF   56  D1 00126          CMPL    FCB, PRIMARY_FCB
                             0F  13 0012B          BEQL    11$
                             55  DD 0012D          PUSHL   HEADER                                          ; 0583
                             56  DD 0012F          PUSHL   FCB
                             02  DD 00131          PUSHL   #2
                             5E  DD 00133          PUSHL   SP
                        0000G CF  9F 00135          PUSHAB  INIT_FCB
                    6B       05  FB 00139          CALLS   #5, SYS$CMKRNL
                             55  DD 0013C 11$:      PUSHL   HEADER                                          ; 0585
                0000G   CF   01  FB 0013E          CALLS   #1, CHECKSUM
                0000G   CF   00  FB 00143          CALLS   #0, WRITE_HEADER                                ; 0586
                    52       59  D0 00148          MOVL    REREAD, REREAD2                                 ; 0588
                    2E   0C  AC  E9 0014B          BLBC    DEALLOCATE, 12$                                 ; 0589
                             6E  B5 0014F          TSTW    EXT_FID
                             2A  13 00151          BEQL    12$
                    59       01  D0 00153          MOVL    #1, REREAD                                      ; 0592
                             53  DD 00156          PUSHL   EX_SEGNUM                                       ; 0593
                        04   AE  9F 00158          PUSHAB  EXT_FID
                             56  DD 0015B          PUSHL   FCB
                             7E  D4 0015D          CLRL    -(SP)
                0000G   CF   04  FB 0015F          CALLS   #4, NEXT_HEADER
                    55       50  D0 00164          MOVL    R0, HEADER
                             56  DD 00167          PUSHL   FCB                                             ; 0594
                             01  DD 00169          PUSHL   #1
                             5E  DD 0016B          PUSHL   SP
                        0000G CF  9F 0016D          PUSHAB  DEL_EXTFCB
                    6B       04  FB 00171          CALLS   #4, SYS$CMKRNL
                             55  DD 00174          PUSHL   HEADER                                          ; 0595
                             57  DD 00176          PUSHL   R7
                0000G   CF   02  FB 00178          CALLS   #2, DELETE_FILE
                    11       59  E9 0017D 12$:      BLBC    REREAD, 13$                                     ; 0605
                        0000G CF  DD 00180          PUSHL   PRIMARY_FCB                                     ; 0606
                7E       04   AC  04  C1 00184     ADDL3   #4, FIB, -(SP)
                0000G   CF   02  FB 00189          CALLS   #2, READ_HEADER
                             55  D0 0018E          MOVL    R0, HEADER
                             15  52  E9 00191 13$:  BLBC    REREAD2, 14$                                    ; 0608
                             50   04  AC  D0 00194 MOVL    FIB, R0                                         ; 0611
                50       1C   A0  01  C3 00198     SUBL3   #1, 28(R0), R0
            2   A5           50  10  9C 0019D     ROTL    #16, R0, 18(HEADER)
                             55  DD 001A2          PUSHL   HEADER                                          ; 0612
                0000G   CF   01  FB 001A4          CALLS   #1, MARK_DIRTY
                0000G   CF   00  FB 001A9 14$:      CALLS   #0, PMS_END_SUB                                 ; 0619
                             04  001AE             RET                                                     ; 0621

; Routine Size:  431 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  310     0622  1 GLOBAL ROUTINE TRUNCATE_HEADER (FIB, HEADER, DEALLOCATE, POINTER, LAST_COUNT) : NOVALUE =
;  311     0623  1
;  312     0624  1 !++
;  313     0625  1 !
;  314     0626  1 ! FUNCTIONAL DESCRIPTION:
;  315     0627  1 !
;  316     0628  1 !       This routine returns the indicated retrieval pointers in the given
;  317     0629  1 !       file header to the storage map and erases them in the file header.
;  318     0630  1 !
;  319     0631  1 !
;  320     0632  1 ! CALLING SEQUENCE:
;  321     0633  1 !       TRUNCATE_HEADER (ARG1, ARG2, ARG3, ARG4, ARG5)
;  322     0634  1 !
;  323     0635  1 ! INPUT PARAMETERS:
;  324     0636  1 !       ARG1: address of FIB of operation
;  325     0637  1 !       ARG2: address of file header
;  326     0638  1 !       ARG3: = 1 to return space to the storage map
;  327     0639  1 !             = 0 to just erase the retrieval pointers
;  328     0640  1 !       ARG4: address of first retrieval pointer to process, if present
;  329     0641  1 !       ARG5: new count field of first pointer, if present
;  330     0642  1 !
;  331     0643  1 ! IMPLICIT INPUTS:
;  332     0644  1 !       NONE
;  333     0645  1 !
;  334     0646  1 ! OUTPUT PARAMETERS:
;  335     0647  1 !       NONE
;  336     0648  1 !
;  337     0649  1 ! IMPLICIT OUTPUTS:
;  338     0650  1 !       NONE
;  339     0651  1 !
;  340     0652  1 ! ROUTINE VALUE:
;  341     0653  1 !       NONE
;  342     0654  1 !
;  343     0655  1 ! SIDE EFFECTS:
;  344     0656  1 !       file header altered, storage map altered
;  345     0657  1 !
;  346     0658  1 !--
;  347     0659  1
;  348     0660  2 BEGIN
;  349     0661  2
;  350     0662  2 MAP
;  351     0663  2         FIB             : REF BBLOCK,   ! user FIB
;  352     0664  2         HEADER          : REF BBLOCK;   ! file header
;  353     0665  2
;  354     0666  2 LOCAL
;  355     0667  2         MAP_AREA        : REF BBLOCK,   ! address of file header map area
;  356     0668  2         MAP_POINTER     : REF BBLOCK,   ! pointer to map pointers
;  357     0669  2         MAP_END,                        ! address of end of map area
;  358     0670  2         COUNT,                          ! count field of current retrieval pointer
;  359     0671  2         LBN,                            ! LBN field of current retrieval pointer
;  360     0672  2         RP_COUNT;                       ! new count field for retrieval pointer
;  361     0673  2
;  362     0674  2 EXTERNAL ROUTINE
;  363     0675  2         RETURN_BLOCKS;                  ! return blocks to storage map
;  364     0676  2
;  365     0677  2
;  366     0678  2 ! Establish pointers into the file header. If explicit args are supplied, use
```

```
;  367          0679  2 ! them; else default to releasing the entire file header.
;  368          0680  2 !
;  369          0681  2
;  370          0682  2 MAP_AREA = .HEADER + .HEADER[FH1$B_MPOFFSET]*2;
;  371          0683  2 MAP_POINTER = .MAP_AREA + FM1$C_POINTERS;
;  372          0684  2 MAP_END = .MAP_POINTER + .MAP_AREA[FM1$B_INUSF]*2;
;  373          0685  2 RP_COUNT = 0;
;  374          0686  2
;  375          0687  2 IF ACTUALCOUNT GEQ 4
;  376          0688  2 THEN
;  377          0689  3     BEGIN
;  378          0690  3     MAP_POINTER = .POINTER;
;  379          0691  3     RP_COUNT = .LAST_COUNT;
;  380          0692  2     END;
;  381          0693  2
;  382          0694  2 ! Now scan the map area, cleaning out pointers and releasing blocks.
;  383          0695  2 !
;  384          0696  2
;  385          0697  2 UNTIL .MAP_POINTER GEQA .MAP_END DO
;  386          0698  3     BEGIN
;  387          0699  3     COUNT = .MAP_POINTER[FM1$B_COUNT] + 1;
;  388          0700  3     LBN = .MAP_POINTER[FM1$W_LOWLBN];
;  389          0701  3     LBN<16,8> = .MAP_POINTER[FM1$B_HIGHLBN];
;  390          0702  3     IF .RP_COUNT NEQ 0
;  391          0703  3     THEN
;  392          0704  3         MAP_POINTER[FM1$B_COUNT] = .RP_COUNT - 1
;  393          0705  3     ELSE
;  394          0706  4         BEGIN
;  395          0707  4         MAP_POINTER[FM1$B_HIGHLBN] = 0;
;  396          0708  4         MAP_POINTER[FM1$B_COUNT] = 0;
;  397          0709  4         MAP_POINTER[FM1$W_LOWLBN] = 0;
;  398          0710  4         MAP_AREA[FM1$B_INUSE] = .MAP_AREA[FM1$B_INUSE] - 2;
;  399          0711  3         END;
;  400          0712  3
;  401          0713  3     IF .DEALLOCATE THEN RETURN_BLOCKS (.LBN + .RP_COUNT, .COUNT - .RP_COUNT);
;  402          0714  3     FIB[FIB$L_EXSZ] = .FIB[FIB$L_EXSZ] + .COUNT - .RP_COUNT;
;  403          0715  3     RP_COUNT = 0;
;  404          0716  3     MAP_POINTER = .MAP_POINTER + 4;
;  405          0717  2     END;
;  406          0718  2
;  407          0719  1 END;                                    ! end of routine TRUNCATE_HEADER


                                                .EXTRN  RETURN_BLOCKS

                                    01FC 00000  .ENTRY  TRUNCATE_HEADER, Save R2,R3,R4,R5,R6,R7,R8  ; 0622
                          51    08    AC  D0 00002  MOVL   HEADER, R1                               ; 0682
                          50    01    A1  9A 00006  MOVZBL 1(R1), R0                                ;
                          53        6140  3E 0000A  MOVAW  (R1)[R0], MAP_AREA                       ;
                          52    0A    A3  9E 0000E  MOVAB  10(R3), MAP_POINTER                      ; 0683
                          50    08    A3  9A 00012  MOVZBL 8(MAP_AREA), R0                          ; 0684
                          58        6240  3E 00016  MOVAW  (MAP_POINTER)[R0], MAP_END               ;
                                      56  D4 0001A  CLRL   RP_COUNT                                 ; 0685
                          04          6C  91 0001C  CMPB   (AP), #4                                 ; 0687
                                      08  1F 0001F  BLSSU  1$                                       ;
                          52    10    AC  D0 00021  MOVL   POINTER, MAP_POINTER                     ; 0690
```

```
                                56        14    AC   D0 00025          MOVL    LAST_COUNT, RP_COUNT                            ; 0691
                                54        04    AC   D0 00029 1$:      MOVL    FIB, R4                                         ; 0714
                                59              52   D1 0002D 2$:      CMPL    MAP_POINTER, MAP_END                            ; 0697
                                                41   1E 00030          BGEQU   6$                                              ;
                                57        01    A2   9A 00032          MOVZBL  1(MAP_POINTER), COUNT                           ; 0699
                                                57   D6 00036          INCL    COUNT                                           ;
                                55        02    A2   3C 00038          MOVZWL  2(MAP_POINTER), LBN                             ; 0700
          55            08              10          62   F0 0003C      INSV    (MAP_POINTER), #16, #8, LBN                     ; 0701
                                                    56   D5 00041      TSTL    RP_COUNT                                        ; 0702
                                                    07   13 00043      BEQL    3$                                              ;
                  01    A2                56          01   83 00045      SUBB3   #1, RP_COUNT, 1(MAP_POINTER)                    ; 0704
                                                    06   11 0004A      BRB     4$                                              ;
                                                    62   D4 0004C 3$:   CLRL    (MAP_POINTER)                                   ; 0707
                      08    A3                    02   82 0004E      SUBB2   #2, 8(MAP_AREA)                                 ; 0710
                                0C        0C    AC   E9 00052 4$:      BLBC    DEALLOCATE, 5$                                  ; 0713
                      7E                57          56   C3 00056      SUBL3   RP_COUNT, COUNT, -(SP)                          ;
                                                  6645   9F 0005A      PUSHAB  (RP_COUNT)[LBN]                                 ;
                  0000G CF                        02   FB 0005D      CALLS   #2, RETURN_BLOCKS                               ;
                      50                57        18    A4   C1 00062 5$:      ADDL3   24(R4), COUNT, R0                       ; 0714
                  18  A4                50                56   C3 00067      SUBL3   RP_COUNT, R0, 24(R4)                    ;
                                                    56   D4 0006C      CLRL    RP_COUNT                                        ; 0715
                                52                  04   C0 0006E      ADDL2   #4, MAP_POINTER                                 ; 0716
                                                    BA   11 00071      BRB     2$                                              ; 0697
                                                        04 00073 6$:      RET                                                     ; 0719

; Routine Size:  116 bytes,    Routine Base:  $CODE$ + 01AF

;  408          0720  1
;  409          0721  1 END
;  410          0722  0 ELUDOM


;                                       PSECT SUMMARY
;
;      Name                             Bytes                          Attributes
;
;  $CODE$                                547  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        29         0        1000          00:02.0
```

```
;                                       COMMAND QUALIFIERS

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

; Size:         547 code + 0 data bytes
; Run Time:        00:15.3
; Elapsed Time:    00:33.6
; Lines/CPU Min:    2835
; Lexemes/CPU-Min: 14811
; Memory Used:  184 pages
; Compilation Complete
```

FCPDEF
B32

ACLCNTRL
LIS

ACLSUBR
LIS

SNDERL
LIS

ACPCNTRL
LIS

F11X

F11BXQP
MAP

ACCESS
LIS

TRUNC
LIS

FILESERV
MAP

WITURN
LIS

SNDSMB
LIS