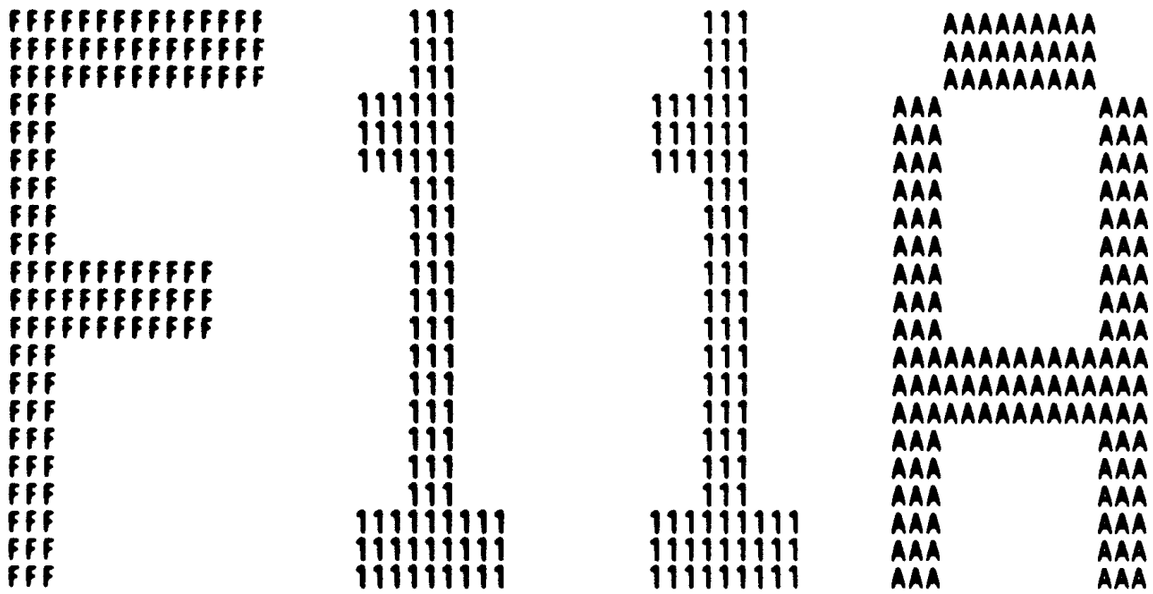
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
111
111
111
111111
111111
111111
111
111
111
111
111
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**MODIFY

```
MM      MM    OOOOOO    DDDDDDDD      IIIIII    FFFFFFFFFF  YY      YY
MM      MM    OOOOOO    DDDDDDDD      IIIIII    FFFFFFFFFF  YY      YY
MMMM  MMMM  OO      OO  DD      DD      II      FF          YY      YY
MMMM  MMMM  OO      OO  DD      DD      II      FF          YY      YY
MM  MM  MM  OO      OO  DD      DD      II      FF            YY  YY
MM  MM  MM  OO      OO  DD      DD      II      FF            YY  YY
MM      MM  OO      OO  DD      DD      II      FFFFFFFF        YY
MM      MM  OO      OO  DD      DD      II      FFFFFFFF        YY
MM      MM  OO      OO  DD      DD      II      FF              YY
MM      MM  OO      OO  DD      DD      II      FF              YY
MM      MM  OO      OO  DD      DD      II      FF              YY          ....
MM      MM  OO      OO  DD      DD      II      FF              YY          ....
MM      MM    OOOOOO    DDDDDDDD      IIIIII    FF              YY          ....
MM      MM    OOOOOO    DDDDDDDD      IIIIII    FF              YY          ....

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

```
;     1          0001  0 MODULE MODIFY (
;     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 implements the MODIFY function.
;    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  23:03
;    48          0048  1 !
;    49          0049  1 ! MODIFIED BY:
;    50          0050  1 !
;    51          0051  1 !       V03-004 LMP0154         L. Mark Pilant,         14-Sep-1983  15:38
;    52          0052  1 !               Do access conflict checks here rather than in WRITE_ATTRIB.
;    53          0053  1 !
;    54          0054  1 !       V03-003 ACG0343         Andrew C. Goldstein,    19-Jul-1983  16:50
;    55          0055  1 !               Inhibit revision count if NORECORD is specified
;    56          0056  1 !
;    57          0057  1 !       V03-002 ACG56916        Andrew C. Goldstein,    21-Jun-1983  12:05
```

```
;    58          0058  1 !                 Update revision date on MODIFY operations
;    59          0059  1 !
;    60          0060  1 !       V03-001 ACG0282         Andrew C. Goldstein,    6-Apr-1982  16:08
;    61          0061  1 !               Check for device write-locked before attempting operations
;    62          0062  1 !
;    63          0063  1 !       V02-002 ACG0171         Andrew C. Goldstein,    7-May-1980  18:55
;    64          0064  1 !               Condition check of truncate lock on presence of FCB
;    65          0065  1 !
;    66          0066  1 !       V02-001 ACG0167         Andrew C. Goldstein,    7-May-1980  18:51
;    67          0067  1 !               Previous revision history moved to F11A.REV
;    68          0068  1 !**
;    69          0069  1
;    70          0070  1
;    71          0071  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    72          0072  1 REQUIRE 'SRC$:FCPDEF.B32';
;    73          0387  1
;    74          0388  1
;    75          0389  1 FORWARD ROUTINE
;    76          0390  1         MODIFY,                         ! MODIFY function routine
;    77          0391  1         INCREMENT;                      ! increment write count in WCB
```

```
;    79          0392  1 GLOBAL ROUTINE MODIFY  =
;    80          0393  1
;    81          0394  1 !++                                                                                          ;
;    82          0395  1 !
;    83          0396  1 ! FUNCTIONAL DESCRIPTION:
;    84          0397  1 !                                                                                            ;
;    85          0398  1 !       This routine implements the MODIFY function.                                         ;
;    86          0399  1 !                                                                                            ;
;    87          0400  1 ! CALLING SEQUENCE:
;    88          0401  1 !       MODIFY ()
;    89          0402  1 !
;    90          0403  1 ! INPUT PARAMETERS:
;    91          0404  1 !       NONE
;    92          0405  1 !
;    93          0406  1 ! IMPLICIT INPUTS:                                                                           ;
;    94          0407  1 !       CURRENT_WINDOW: window for file                                                      ;
;    95          0408  1 !       PRIMARY_FCB: FCB of file                                                             ;
;    96          0409  1 !       IO_PACKET: I/O packet in process                                                     ;
;    97          0410  1 !       FILE_HEADER: address of current file header                                          ;
;    98          0411  1 !
;    99          0412  1 ! OUTPUT PARAMETERS:
;   100          0413  1 !       NONE
;   101          0414  1 !
;   102          0415  1 ! IMPLICIT OUTPUTS:                                                                          ;
;   103          0416  1 !       NONE                                                                                 ;
;   104          0417  1 !
;   105          0418  1 ! ROUTINE VALUE:                                                                             ;
;   106          0419  1 !       NONE                                                                                 ;
;   107          0420  1 !
;   108          0421  1 ! SIDE EFFECTS:                                                                              ;
;   109          0422  1 !       file and FCB modified
;   110          0423  1 !
;   111          0424  1 !--
;   112          0425  1
;   113          0426  2 BEGIN
;   114          0427  2
;   115          0428  2 LOCAL                                                                                        ;
;   116          0429  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH],
;   117          0430  2                                         ! buffer descriptors                                 ;
;   118          0431  2         FIB             : REF BBLOCK,   ! FIB                                                ;
;   119          0432  2         FCB             : REF BBLOCK,   ! FCB of file                                        ;
;   120          0433  2         HEADER          : REF BBLOCK,   ! file header                                        ;
;   121          0434  2         IDENT_AREA      : REF BBLOCK;   ! ident area of file header                          ;
;   122          0435  2                                                                                              ;
;   123          0436  2 EXTERNAL                                                                                     ;
;   124          0437  2         IO_PACKET       : REF BBLOCK,   ! I/O packet in process                              ;
;   125          0438  2         FILE_HEADER     : REF BBLOCK,   ! address of currently resident file header          ;
;   126          0439  2         CURRENT_WINDOW  : REF BBLOCK,   ! window of file                                     ;
;   127          0440  2         PRIMARY_FCB     : REF BBLOCK,   ! FCB of file                                        ;
;   128          0441  2         CURRENT_VCB     : REF BBLOCK,   ! VCB of volume                                      ;
;   129          0442  2         CURRENT_UCB     : REF BBLOCK;   ! UCB of volume                                      ;
;   130          0443  2                                                                                              ;
;   131          0444  2 EXTERNAL ROUTINE                                                                             ;
;   132          0445  2         GET_FIB,                        ! get FIB of request                                 ;
;   133          0446  2         FIND,                           ! find name in directory                             ;
;   134          0447  2         SEARCH_FCB,                     ! search FCB list                                    ;
;   135          0448  2         READ_HEADER,                    ! read file header                                   ;
```

```
;  136          0449  2          CHECK_PROTECT,                  ! check file protection
;  137          0450  2          GET_TIME,                       ! get date/time string
;  138          0451  2          MARK_DIRTY,                     ! mark header for write-back
;  139          0452  2          WRITE_ATTRIB,                   ! write file attributes
;  140          0453  2          EXTEND,                         ! extend file
;  141          0454  2          TRUNCATE,                       ! truncate file
;  142          0455  2          CHECKSUM,                       ! checksum the file header
;  143          0456  2          UPDATE_FCB;                     ! update FCB contents
;  144          0457  2
;  145          0458  2
;  146          0459  2 ! First find the buffer descriptor, FIB, FCB, etc., and read the header.
;  147          0460  2 !
;  148          0461  2
;  149          0462  2                                          ! pointer to buffer descriptors
;  150          0463  2 ABD = .BBLOCK [.IO_PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPT];
;  151          0464  2 FIB = GET_FIB (.ABD);
;  152          0465  2
;  153          0466  2 ! If a directory ID is present, do a directory search first.
;  154          0467  2 !
;  155          0468  2
;  156          0469  2 IF .FIB[FIB$W_DID_NUM] NEQ 0
;  157          0470  2 THEN FIND (.ABD, .FIB, 0);
;  158          0471  2
;  159          0472  2 ! If there is a file open on the channel, check the file ID returned by the
;  160          0473  2 ! FIND against the file ID that is open. If they are different, drop the FCB
;  161          0474  2 ! and window addresses on the floor.
;  162          0475  2 !
;  163          0476  2
;  164          0477  2 IF .PRIMARY_FCB NEQ 0
;  165          0478  2 THEN
;  166          0479  2     IF .PRIMARY_FCB[FCB$W_FID_NUM] NEQ .FIB[FIB$W_FID_NUM]
;  167          0480  2     OR .PRIMARY_FCB[FCB$W_FID_RVN] NEQ .FIB[FIB$W_FID_RVN]
;  168          0481  2     THEN
;  169          0482  3         BEGIN
;  170          0483  3         PRIMARY_FCB = 0;
;  171          0484  3         CURRENT_WINDOW = 0;
;  172          0485  2         END;
;  173          0486  2
;  174          0487  2 FCB = SEARCH_FCB (FIB[FIB$W_FID]);
;  175          0488  2 PRIMARY_FCB = .FCB;
;  176          0489  2 HEADER = READ_HEADER (FIB[FIB$W_FID], .FCB);
;  177          0490  2
;  178          0491  2 ! Check that the volume is write enabled.
;  179          0492  2 !
;  180          0493  2
;  181          0494  2 IF .BBLOCK [CURRENT_UCB[UCB$L_DEVCHAR], DEV$V_SWL]
;  182          0495  2 THEN ERR_EXIT (SS$_WRITLCK);
;  183          0496  2
;  184          0497  2 ! Arbitrate access interlocks. If this is the accessor, then the file
;  185          0498  2 ! must be write accessed. Count a write to the file.
;  186          0499  2 !
;  187          0500  2
;  188          0501  2 IF .CURRENT_WINDOW NEQ 0
;  189          0502  2 THEN
;  190          0503  3     BEGIN
;  191          0504  3     IF NOT .CURRENT_WINDOW[WCB$V_WRITE]
;  192          0505  3     THEN ERR_EXIT (SS$_NOPRIV);
```

```
;  193          0506  3     IF NOT .FIB[FIB$V_NORECORD]
;  194          0507  3     THEN KERNEL_CALL (INCREMENT, CURRENT_WINDOW[WCB$L_WRITES]);
;  195          0508  3     END
;  196          0509  3
;  197          0510  3 ! If it is not, then the file must not be locked against modification
;  198          0511  3 ! and the caller must pass file protection.
;  199          0512  3 !
;  200          0513  3
;  201          0514  2 ELSE
;  202          0515  3     BEGIN
;  203          0516  3     IF .FCB NEQ 0
;  204          0517  3     THEN
;  205          0518  3         IF .FCB[FCB$V_EXCL]
;  206          0519  3         OR .FCB[FCB$W_LCNT] NEQ 0
;  207          0520  3         THEN ERR_EXIT (SS$_ACCONFLICT);
;  208          0521  3     CHECK_PROTECT (WRITE_ACCESS, .HEADER, .FCB);
;  209          0522  3
;  210          0523  3 ! If the file is not open, count a revision and update the revision
;  211          0524  3 ! date now.
;  212          0525  3 !
;  213          0526  3
;  214          0527  3     IF NOT .FIB[FIB$V_NORECORD]
;  215          0528  3     THEN
;  216          0529  4         BEGIN
;  217          0530  4         IDENT_AREA = .HEADER + .HEADER[FH1$B_IDOFFSET]*2;
;  218          0531  4         IDENT_AREA[FI1$W_REVISION] = .IDENT_AREA[FI1$W_REVISION] + 1;
;  219          0532  4         GET_TIME (IDENT_AREA[FI1$T_REVDATE]);
;  220          0533  4         MARK_DIRTY (.HEADER);
;  221          0534  3         END;
;  222          0535  2     END;
;  223          0536  2
;  224          0537  2
;  225          0538  2 ! If an attribute list exists, perform the write attributes operation.
;  226          0539  2 !
;  227          0540  2
;  228          0541  2 IF .IO_PACKET[IRP$W_BCNT] GTR ABD$C_ATTRIB
;  229          0542  2 THEN
;  230          0543  3     BEGIN
;  231          0544  3     WRITE_ATTRIB (.HEADER, .ABD, 1);
;  232          0545  3     CHECKSUM (.HEADER);
;  233          0546  2     END;
;  234          0547  2
;  235          0548  2 ! If the extend enable bit is on, perform the extend operation.
;  236          0549  2 ! If the truncate bit is on, perform the truncate operation. If both are
;  237          0550  2 ! on, it is an error.
;  238          0551  2 !
;  239          0552  2
;  240          0553  3 IF (.FIB[FIB$V_EXTEND] OR .FIB[FIB$V_TRUNC])
;  241          0554  2 AND .CURRENT_VCB[VCB$V_NOALLOC]
;  242          0555  2 THEN ERR_EXIT (SS$_WRITLCK);
;  243          0556  2
;  244          0557  2 IF .FIB[FIB$V_EXTEND]
;  245          0558  2 THEN
;  246          0559  3     BEGIN
;  247          0560  3     IF .FIB[FIB$V_TRUNC] THEN ERR_EXIT (SS$_BADPARAM);
;  248          0561  3     EXTEND (.FIB, .HEADER);
;  249          0562  2     END;
```

```
;  250          0563  2
;  251          0564  2 IF .FIB[FIB$V_TRUNC]
;  252          0565  2 THEN
;  253          0566  3     BEGIN
;  254          0567  3     IF .FCB NEQ 0
;  255          0568  3     THEN
;  256          0569  4         BEGIN
;  257          0570  4         IF .FCB[FCB$L_TCNT] NEQ 0
;  258          0571  4         THEN ERR_EXIT(SS$_ACCONFLICT);
;  259          0572  3         END;
;  260          0573  3     TRUNCATE (.FIB, .HEADER, DEALLOC_BLOCKS);
;  261          0574  2     END;
;  262          0575  2
;  263          0576  2 HEADER = .FILE_HEADER;
;  264          0577  2 CHECKSUM (.HEADER);                          ! checksum the file header
;  265          0578  2 KERNEL_CALL (UPDATE_FCB, .HEADER);
;  266          0579  2
;  267          0580  2 RETURN 1;
;  268          0581  2
;  269          0582  1 END;                                         ! end of routine MODIFY

                                        .TITLE  MODIFY
                                        .IDENT  \V04-000\

                                        .EXTRN  IO_PACKET, FILE_HEADER
                                        .EXTRN  CURRENT_WINDOW, PRIMARY_FCB
                                        .EXTRN  CURRENT_VCB, CURRENT_UCB
                                        .EXTRN  GET_FIB, FIND, SEARCH_FCB
                                        .EXTRN  READ_HEADER, CHECK_PROTECT
                                        .EXTRN  GET_TIME, MARK_DIRTY
                                        .EXTRN  WRITE_ATTRIB, EXTEND
                                        .EXTRN  TRUNCATE, CHECKSUM
                                        .EXTRN  UPDATE_FCB, SYS$CMKRNL

                                        .PSECT  $CODE$,NOWRT,2

                        01FC 00000      .ENTRY  MODIFY, Save R2,R3,R4,R5,R6,R7,R8        ; 0392
          58     0000G  CF   9E 00002    MOVAB   CURRENT_WINDOW, R8                      ;
          57     0000G  CF   9E 00007    MOVAB   PRIMARY_FCB, R7                         ;
          56 00000000G  9F   9E 0000C    MOVAB   @#SYS$CMKRNL, R6                        ;
          50     0000G  CF   D0 00013    MOVL    IO_PACKET, R0                           ; 0463
          55       2C   B0   D0 00018    MOVL    @44(R0), ABD                            ;
                        55   DD 0001C    PUSHL   ABD                                     ; 0464
0000G     CF                  01   FB 0001E    CALLS   #1, GET_FIB                     ;
          52                  50   D0 00023    MOVL    R0, FIB                         ;
                        0A    A2   B5 00026    TSTW    10(FIB)                         ; 0469
                              0B   13 00029    BEQL    1$                              ;
                              7E   D4 0002B    CLRL    -(SP)                           ; 0470
                              52   DD 0002D    PUSHL   FIB                             ;
                              55   DD 0002F    PUSHL   ABD                             ;
0000G     CF                  03   FB 00031    CALLS   #3, FIND                        ;
          50                  67   D0 00036 1$: MOVL    PRIMARY_FCB, R0                ; 0477
                              12   13 00039    BEQL    3$                              ;
    04    A2        24    A0   B1 0003B    CMPW    36(R0), 4(FIB)                  ; 0479
                              07   12 00040    BNEQ    2$                              ;
    08    A2        28    A0   B1 00042    CMPW    40(R0), 8(FIB)                  ; 0480
```

```
                                04  13 00047                BEQL    3$                                      ;
                                67  D4 00049 2$:            CLRL    PRIMARY_FCB                             ; 0483
                                68  D4 00049                CLRL    CURRENT_WINDOW                          ; 0484
                              04    A2 0004D 3$:            PUSHAB  4(FIB)                                  ; 0487
                     0000G    CF    01 00050                CALLS   #1, SEARCH_FCB                          ;
                              53    50 00055                MOVL    R0, FCB                                 ;
                              67    53 00058                MOVL    FCB, PRIMARY_FCB                        ; 0488
                                    53 0005B                PUSHL   FCB                                     ; 0489
                              04    A2 0005D                PUSHAB  4(FIB)                                  ;
                     0000G    CF    02 00060                CALLS   #2, READ_HEADER                         ;
                              54    50 00065                MOVL    R0, HEADER                              ;
                              50  0000G CF 00068            MOVL    CURRENT_UCB, R0                         ; 0494
                03       3B   A0    01 0006D                BBC     #1, 59(R0), 4$                          ;
                                 0089 31 00072              BRW     12$                                     ;
                              50    68 00075 4$:            MOVL    CURRENT_WINDOW, R0                      ; 0501
                                    1D 00078                BEQL    6$                                      ;
                03       0B   A0    01 0007A                BBS     #1, 11(R0), 5$                          ; 0504
                                    24 0007F                CHMU    #36                                     ; 0505
                                    04 00081                RET                                             ;
                4A           62     15 00082 5$:            BBS     #21, (FIB), 9$                          ; 0506
                7E           68     28 00086                ADDL3   #40, CURRENT_WINDOW, -(SP)              ; 0507
                                    01 0008A                PUSHL   #1                                      ;
                                    5E 0008C                PUSHL   SP                                      ;
                              0000V CF 9F 0008E             PUSHAB  INCREMENT                               ;
                              66    04 00092                CALLS   #4, SYS$CMKRNL                          ;
                                    39 00095                BRB     9$                                      ; 0501
                                    53 00097 6$:            TSTL    FCB                                     ; 0516
                                    0D 00099                BEQL    8$                                      ;
                03       22   A3    03 0009B                BBC     #3, 34(FCB), 7$                         ; 0518
                                 0080 31 000A0              BRW     16$                                     ;
                              1E    A3 B5 000A3 7$:         TSTW    30(FCB)                                 ; 0519
                                    7B 12 000A6             BNEQ    16$                                     ;
                                    53 DD 000A8 8$:         PUSHL   FCB                                     ; 0521
                                    54 DD 000AA             PUSHL   HEADER                                  ;
                                    01 DD 000AC             PUSHL   #1                                      ;
                     0000G    CF    03 FB 000AE             CALLS   #3, CHECK_PROTECT                       ;
                19           62    15 E0 000B3             BBS     #21, (FIB), 9$                          ; 0527
                              50    64 9A 000B7             MOVZBL  (HEADER), R0                            ; 0530
                              50  6440 3E 000BA              MOVAW   (HEADER)[R0], IDENT_AREA                ;
                                0A  A0 B6 000BE             INCW    10(IDENT_AREA)                          ; 0531
                                0C  A0 9F 000C1             PUSHAB  12(IDENT_AREA)                          ; 0532
                     0000G    CF    01 FB 000C4             CALLS   #1, GET_TIME                            ;
                                    54 DD 000C9             PUSHL   HEADER                                  ; 0533
                     0000G    CF    01 FB 000CB             CALLS   #1, MARK_DIRTY                          ;
                              50  0000G CF DO 000D0 9$:     MOVL    IO_PACKET, R0                           ; 0541
                              05    32  A0 B1 000D5         CMPW    50(R0), #5                              ;
                                    10 1B 000D9             BLEQU   10$                                     ;
                                    01 DD 000DB             PUSHL   #1                                      ; 0544
                                    30 BB 000DD             PUSHR   #^M<R4,R5>                              ;
                     0000G    CF    03 FB 000DF             CALLS   #3, WRITE_ATTRIB                        ;
                                    54 DD 000E4             PUSHL   HEADER                                  ; 0545
                     0000G    CF    01 FB 000E6             CALLS   #1, CHECKSUM                            ;
                              16    A2 95 000EB 10$:        TSTB    22(FIB)                                 ; 0553
                                    04 19 000EE             BLSS    11$                                     ;
                              0F  17  A2 E9 000F0           BLBC    23(FIB), 13$                            ;
                              50  0000G CF DO 000F4 11$:    MOVL    CURRENT_VCB, R0                         ; 0554
                05       0B   A0    04 E1 000F9             BBC     #4, 11(R0), 13$                         ;
```

```
                          025C   8F   BF 000FE 12$:    CHMU    #604                                    ; 0555
                                      04 00102         RET                                             ;
                            16   A2   95 00103 13$:    TSTB    22(FIB)                                 ; 0557
                                 0E   18 00106         BGEQ    15$                                     ;
                       03        17   A2   E9 00108         BLBC    23(FIB), 14$                        ; 0560
                                      14   BF 0010C         CHMU    #20                                 ;
                                      04 0010E         RET                                             ;
                                      14   BB 0010F 14$:    PUSHR   #^M<R2,R4>                          ; 0561
                0000G CF              02   FB 00111         CALLS   #2, EXTEND                          ;
                      17        17   A2   E9 00116 15$:    BLBC    23(FIB), 18$                        ; 0564
                                      53   D5 0011A         TSTL    FCB                                 ; 0567
                                      0A   13 0011C         BEQL    17$                                 ;
                            20   A3   B5 0011E         TSTW    32(FCB)                                 ; 0570
                                      05   13 00121         BEQL    17$                                 ;
                          0800   8F   BF 00123 16$:    CHMU    #2048                                   ; 0571
                                      04 00127         RET                                             ;
                                      01   DD 00128 17$:    PUSHL   #1                                  ; 0573
                                      14   BB 0012A         PUSHR   #^M<R2,R4>                          ;
                0000G CF              03   FB 0012C         CALLS   #3, TRUNCATE                        ;
                      54        0000G CF   D0 00131 18$:    MOVL    FILE_HEADER, HEADER                 ; 0576
                                      54   DD 00136         PUSHL   HEADER                              ; 0577
                0000G CF              01   FB 00138         CALLS   #1, CHECKSUM                        ;
                                      54   DD 0013D         PUSHL   HEADER                              ; 0578
                                      01   DD 0013F         PUSHL   #1                                  ;
                                      5E   DD 00141         PUSHL   SP                                  ;
                          0000G CF   9F   00143         PUSHAB  UPDATE_FCB                              ;
                       66             04   FB 00147         CALLS   #4, SYS$CMKRNL                      ;
                       50             01   D0 0014A         MOVL    #1, R0                              ; 0580
                                      04 0014D         RET                                             ; 0582

; Routine Size:  334 bytes,    Routine Base:  $CODE$ + 0000

;  270          0583  1
;  271          0584  1 !
;  272          0585  1 ! Subroutine to increment the write count in the window. This operation
;  273          0586  1 ! must be done in kernel mode.
;  274          0587  1 !
;  275          0588  1
;  276          0589  1 ROUTINE INCREMENT (ADDRESS) =
;  277          0590  2 BEGIN
;  278          0591  2
;  279          0592  2 .ADDRESS = ..ADDRESS + 1;
;  280          0593  2 1
;  281          0594  2
;  282          0595  1 END;


                                    0000 00000 INCREMENT:
                                                .WORD   Save nothing                                    ; 0589
                                 04   BC   D6 00002         INCL    @ADDRESS                            ; 0592
                          50             01   D0 00005         MOVL    #1, R0                              ; 0595
                                              04 00008         RET                                         ;
```

```
; Routine Size:  9 bytes,    Routine Base:  $CODE$ + 014E

;  283          0596  1
;  284          0597  1 END
;  285          0598  0 ELUDOM




;                                  PSECT SUMMARY
;
;       Name                     Bytes                          Attributes
;
;  $CODE$                         343  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        30         0      1000          00:01.9




;                                        COMMAND QUALIFIERS

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

; Size:          343 code + 0 data bytes
; Run Time:          00:11.1
; Elapsed Time:      00:29.6
; Lines/CPU Min:      3241
; Lexemes/CPU-Min:  14601
; Memory Used:  157 pages
; Compilation Complete
```

REQUEU
LIS

RWATTR
LIS

MODIFY
LIS

SCHFCB
LIS

MAKAEC
LIS

MPWIND
LIS

MAPVBN
LIS

PMS
LIS

RDHEDR
LIS

RWVB
LIS

RETDIR
LIS

RDBLOK
LIS

SMALOC
LIS

MOUNT
LIS

MAKNMB
LIS

MAKSTR
LIS

MXTHDR
LIS