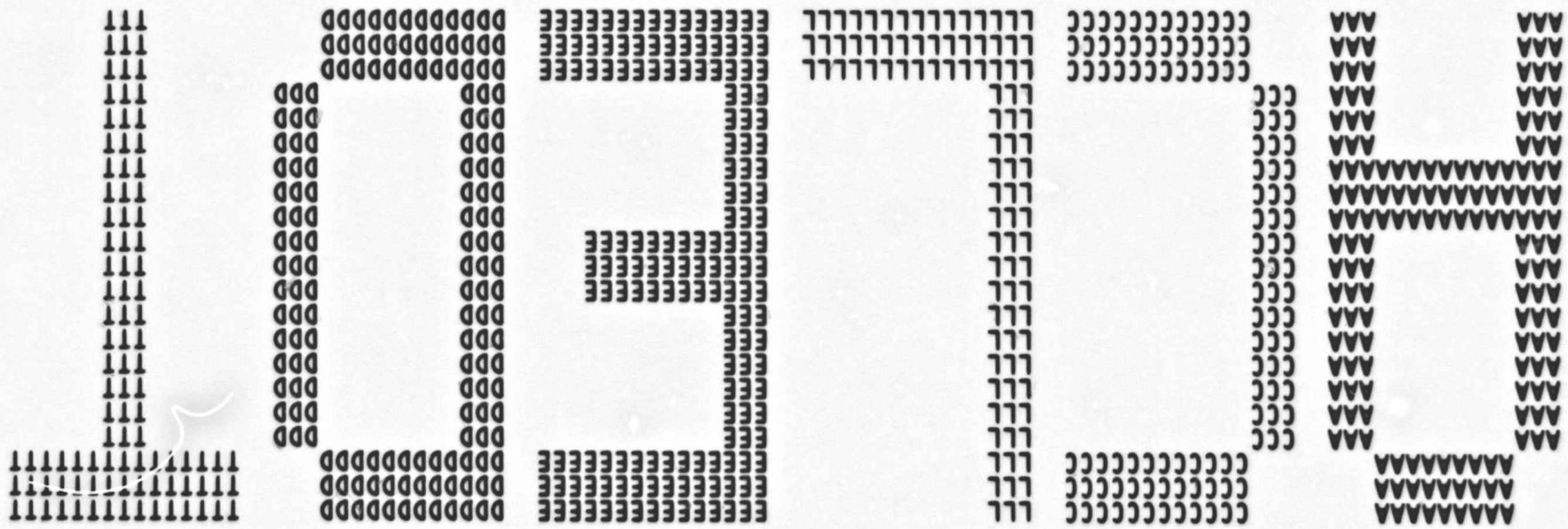
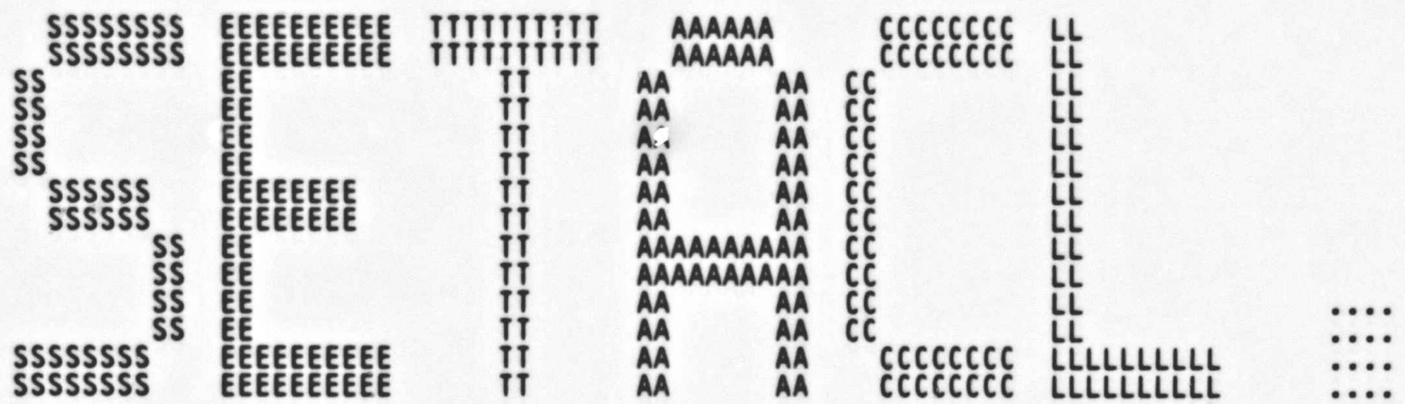


FILEID**SETACL



```
;      1     0001  0 MODULE AED$SETACL (
;      2     0002  0                     LANGUAGE (BLISS32),
;      3     0003  0                     IDENT = 'V04-000'
;      4     0004  0                     ADDRESSING_MODE (EXTERNAL = GENERAL)
;      5     0005  0                     ) =
;      6     0006  1 BEGIN
;      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:     SET utility
;     34     0034  1 !
;     35     0035  1 ! ABSTRACT:
;     36     0036  1 !
;     37     0037  1 !       This module contains all the routines necessary to support the
;     38     0038  1 !       DCL commands SET FILE/ACL, SET DIRECTORY/ACL, SET DEVICE/ACL,
;     39     0039  1 !       and SET ACL with the exception of the /EDIT qualifier.
;     40     0040  1 !
;     41     0041  1 ! ENVIRONMENT:
;     42     0042  1 !
;     43     0043  1 !       VAX/VMS operating system, user mode utilities.
;     44     0044  1 !
;     45     0045  1 !--
;     46     0046  1 !
;     47     0047  1 !
;     48     0048  1 ! AUTHOR:       L. Mark Pilant          CREATION DATE:  4-May-1983  9:20
;     49     0049  1 !
;     50     0050  1 ! MODIFIED BY:
;     51     0051  1 !
;     52     0052  1 !       V03-019 LMP0296         L. Mark Pilant,         6-Aug-1984  15:02
;     53     0053  1 !               Change the location of the code that determines if the target
;     54     0054  1 !               file is a directory file to correct a bug where the default
;     55     0055  1 !               option was being cleared.
;     56     0056  1 !
;     57     0057  1 !       V03-018 LMP0283         L. Mark Pilant,         25-Jul-1984  12:40
```

```
;    58          0058  1 !                Make sure the default object type is a file.
;    59          0059  1 !
;    60          0060  1 !        V03-017 LMP0260         L. Mark Pilant,         27-Jun-1984  9:11
;    61          0061  1 !                Add support for the /DEFAULT qualifier.
;    62          0062  1 !
;    63          0063  1 !        V03-016 LMP0253         L. Mark Pilant,         4-Jun-1984  10:41
;    64          0064  1 !                Fix the error handling in COPY_ACL so that SS$_NOMOREACE
;    65          0065  1 !                and SS$_ACLEMPTY are (again) turned into SS$_NORMAL.
;    66          0066  1 !
;    67          0067  1 !        V03-015 LMP0244         L. Mark Pilant,         1-May-1984  16:02
;    68          0068  1 !                Fix a bug intruduced by LMP0238 that caused the wrong
;    69          0069  1 !                item code to be used.
;    70          0070  1 !
;    71          0071  1 !        V03-014 LMP0238         L. Mark Pilant,         19-Apr-1984  13:35
;    72          0072  1 !                Use the size of the ACE being twiddled, when possible.
;    73          0073  1 !
;    74          0074  1 !        V03-013 LMP0236         L. Mark Pilant,         18-Apr-1984  13:25
;    75          0075  1 !                Correct a bug that caused an ACCVIO to be returned from the
;    76          0076  1 !                $CHANGE_ACL system service when an attempt was made to lock
;    77          0077  1 !                a file's ACL for writing.
;    78          0078  1 !
;    79          0079  1 !        V03-012 LMP0230         L. Mark Pilant,         16-Apr-1984  10:45
;    80          0080  1 !                Track interface changes to $CHANGE_ACL system service.
;    81          0081  1 !
;    82          0082  1 !        V03-011 LMP0226         L. Mark Pilant,         9-Apr-1984  9:32
;    83          0083  1 !                Make sure all ACEs to be modified exist and are in the
;    84          0084  1 !                correct order (if more than one).
;    85          0085  1 !
;    86          0086  1 !        V03-010 LMP0224         L. Mark Pilant,         7-Apr-1984  13:50
;    87          0087  1 !                Use enhanced lib$file_scan features for stickyness.
;    88          0088  1 !
;    89          0089  1 !        V03-009 LMP0223         L. Mark Pilant,         6-Apr-1984  12:49
;    90          0090  1 !                Use the correct amount of storage for the $CHANGE_ACL
;    91          0091  1 !                lock block.
;    92          0092  1 !
;    93          0093  1 !        V03-008 LMP0213         L. Mark Pilant,         24-Mar-1984  12:23
;    94          0094  1 !                Add support for locking and unlocking the object's ACL.
;    95          0095  1 !                Also, modify it so that the DCL commands SET ACL and SHOW
;    96          0096  1 !                ACL call the same image.
;    97          0097  1 !
;    98          0098  1 !        V03-007 LMP0210         L. Mark Pilant,         23-Mar-1984  14:33
;    99          0099  1 !                Change the /MODIFY qualifier to /REPLACE.
;   100          0100  1 !
;   101          0101  1 !        V03-006 LMP0198         L. Mark Pilant,         28-Feb-1984  12:05
;   102          0102  1 !                Open the object specified by the /LIKE qualifier for
;   103          0103  1 !                shared read access.
;   104          0104  1 !
;   105          0105  1 !        V03-005 LMP0185         L. Mark Pilant,         4-Feb-1984  12:15
;   106          0106  1 !                Add support for device ACLs.
;   107          0107  1 !
;   108          0108  1 !        V03-004 LMP0181         L. Mark Pilant,         15-Dec-1983  9:54
;   109          0109  1 !                Change code to use $CHANGE_ACL instead of the ACP to do
;   110          0110  1 !                the ACL twiddling.
;   111          0111  1 !
;   112          0112  1 !        V03-003 LMP0168         L. Mark Pilant,         11-Nov-1983  10:58
;   113          0113  1 !                Make use of the HIDDEN ACE option illegal.
;   114          0114  1 !
```

```
;  115          0115  1 !       V03-002 LMP0137         L. Mark Pilant,         12-Aug-1983  9:36
;  116          0116  1 !               Add support for the qualifiers: /BEFORE, /SINCE,
;  117          0117  1 !               and /CREATED.
;  118          0118  1 !
;  119          0119  1 !       V03-001 LMP0126         L. Mark Pilant,         5-Jul-1983  11:00
;  120          0120  1 !               Correctly use a 'sticky' input file-spec.  Also, handle
;  121          0121  1 !               errors while processing multiple files correctly.
;  122          0122  1 !
;  123          0123  1 !**
;  124          0124  1
;  125          0125  1 LIBRARY 'SYS$LIBRARY:LIB';
;  126          0126  1 LIBRARY 'SYS$LIBRARY:TPAMAC';
```

```
;  128          0127  1 !  Routines contained within this module.
;  129          0128  1
;  130          0129  1 FORWARD ROUTINE
;  131          0130  1         SET_ACL,                                ! Main processing routine
;  132          0131  1         GET_FILE,                               ! Get next output file spec
;  133          0132  1         PROCESS_FILE,                           ! Act upon the specified file
;  134          0133  1         ADD_ACL,                                ! Add to an existing ACL
;  135          0134  1         DELETE_ACL,                             ! Delete ACEs or an ACL
;  136          0135  1         REPLACE_ACL,                            ! Modify existing ACEs
;  137          0136  1         COPY_ACL,                               ! Copy a object's ACL
;  138          0137  1         INPUT_ERROR,                            ! Signal file scanning error
;  139          0138  1         FILE_ERROR;                             ! Signal general file error
;  140          0139  1
;  141          0140  1 ! Define common error message codes.
;  142          0141  1
;  143        P 0142  1 $SHR_MSGDEF     (SET, 119, LOCAL,
;  144        P 0143  1                 (SYNTAX, SEVERE),
;  145        P 0144  1                 (OPENIN, ERROR),
;  146        P 0145  1                 (CLOSEIN, ERROR),
;  147        P 0146  1                 (OPENOUT, ERROR),
;  148        P 0147  1                 (CLOSEOUT, ERROR),
;  149        P 0148  1                 (READERR, SEVERE),
;  150        P 0149  1                 (WRITEERR, SEVERE)
;  151          0150  1                 );
;  152          0151  1
;  153          0152  1 ! Define necessary macros.
;  154          0153  1
;  155          0154  1 MACRO
;  156        M 0155  1         SIGNAL  (ARG) =
;  157        M 0156  1                 BEGIN
;  158        M 0157  1                 EXTERNAL ROUTINE        LIB$SIGNAL;
;  159        M 0158  1                 LIB$SIGNAL (ARG %IF %LENGTH-1 GTR 0 %THEN, %REMAINING %FI);
;  160        M 0159  1                 IF NOT ARG AND
;  161        M 0160  1                    (.WORST_ERROR AND STS$M_SEVERITY) LSS
;  162        M 0161  1                    (ARG AND STS$M_SEVERITY) THEN WORST_ERROR = ARG OR
;  163        M 0162  1                                                   STS$M_INHIB_MSG;
;  164        M 0163  1                 END
;  165          0164  1                 %;
;  166          0165  1
;  167          0166  1 MACRO
;  168        M 0167  1         ALLOCATE (SIZE, ADDRESS) =
;  169        M 0168  1                 BEGIN
;  170        M 0169  1                 EXTERNAL ROUTINE        LIB$GET_VM;
;  171        M 0170  1                 LOCAL   VM_STATUS;
;  172        M 0171  1                 VM_STATUS = LIB$GET_VM (%REF (SIZE), ADDRESS);
;  173        M 0172  1                 IF .VM_STATUS THEN CH$FILL (0, SIZE, .ADDRESS);
;  174        M 0173  1                 .VM_STATUS
;  175        M 0174  1                 END
;  176          0175  1                 %;
;  177          0176  1
;  178          0177  1 ! Various needed flags.
;  179          0178  1
;  180          0179  1 MACRO
;  181          0180  1         QUAL_AFTER      = 0, 0, 1, 0 %,         ! /AFTER qualifier seen
;  182          0181  1         QUAL_DELETE     = 0, 1, 1, 0 %,         ! /DELETE qualifier seen
;  183          0182  1         QUAL_LIKE       = 0, 2, 1, 0 %,         ! /LIKE qualifier seen
;  184          0183  1         QUAL_LOG        = 0, 3, 1, 0 %,         ! /LOG qualifier seen
```

```
;  185          0184  1          QUAL_REPLACE    = 0, 4, 1, 0 %,          ! /REPLACE qualifier seen
;  186          0185  1          QUAL_NEW        = 0, 5, 1, 0 %,          ! /NEW qualifier seen
;  187          0186  1          QUAL_DEFAULT    = 0, 6, 1, 0 %,          ! /DEFAULT qualifier seen
;  188          0187  1          DIRECTORY       = 0, 10, 1, 0 %,         ! Target file is a directory file
;  189          0188  1          IN_ELLIPSE      = 0, 11, 1, 0 %,         ! In ellipse processing
;  190          0189  1          SET_DEV_CMD     = 0, 12, 1, 0 %,         ! SET DEVICE command
;  191          0190  1          SET_FILE_CMD    = 0, 13, 1, 0 %,         ! SET FILE command
;  192          0191  1          SET_DIR_CMD     = 0, 14, 1, 0 %,         ! SET DIRECTORY command
;  193          0192  1          SET_ACL_CMD     = 0, 15, 1, 0 %;         ! SET ACL command
;  194          0193  1
;  195          0194  1  ! Structure definition for the old and new ACE queues.
;  196          0195  1
;  197          0196  1  MACRO
;  198          0197  1          ACEQ_L_FLINK    = 0, 0, 32, 0 %,         ! Forward link
;  199          0198  1          ACEQ_L_BLINK    = 4, 0, 32, 0 %,         ! Backward link
;  200          0199  1          ACEQ_T_ACE      = 8, 0, 32, 0 %;         ! Start of the actual ACE
;  201          0200  1
;  202          0201  1  LITERAL
;  203          0202  1          ACEQ_C_LENGTH   = 8;                     ! Length of the overhead area
;  204          0203  1
;  205          0204  1  ! Semi-permanent storage.
;  206          0205  1
;  207          0206  1  OWN
;  208          0207  1          FLAGS           : $BBLOCK [2],           ! Needed flags
;  209          0208  1          WORST_ERROR,                             ! Worst error encountered
;  210          0209  1          ACL_LOCKID      : $BBLOCK [ACL$S_RLOCK_ACL],    ! Lock-id for ACL lock
;  211          0210  1          OBJECT_TYPE,                             ! Object type code
;  212          0211  1          OBJECT_NAME     : $BBLOCK [DSC$C_S_BLN],        ! Object name descriptor
;  213          0212  1          OBJECT_FAB      : $FAB_DECL,             ! Output object FAB
;  214          0213  1          OBJECT_NAM      : $NAM_DECL,             ! Output object NAMe block
;  215          0214  1          OBJECT_EXP_NAME : $BBLOCK [NAM$C_MAXRSS],       ! Expanded name string
;  216          0215  1          OBJECT_RES_NAME : $BBLOCK [NAM$C_MAXRSS],       ! Resultant name string
;  217          0216  1          RELATED_NAM     : $NAM_DECL,             ! Related object spec
;  218          0217  1          CHAN,                                    ! Input object channel
;  219          0218  1          ACL_CONTEXT,                             ! ACL context used by $CHANGE_ACL
;  220          0219  1          SACL_LOCKID     : $BBLOCK [ACL$S_RLOCK_ACL],    ! Lock-id for ACL lock
;  221          0220  1          SOBJECT_TYPE,                            ! Source object type code
;  222          0221  1          SOBJECT_DESC    : $BBLOCK [DSC$C_S_BLN],        ! Source object descr
;  223          0222  1          SOBJECT_FAB     : $FAB_DECL,             ! Source object FAB
;  224          0223  1          SOBJECT_NAM     : $NAM_DECL,             ! Source object NAMe block
;  225          0224  1          SOBJECT_EXP_NAME        : $BBLOCK [NAM$C_MAXRSS],       ! Expanded name string
;  226          0225  1          SOBJECT_RES_NAME        : $BBLOCK [NAM$C_MAXRSS],       ! Resultant name string
;  227          0226  1          SCHAN,                                   ! Source object channel
;  228          0227  1          SACL_CONTEXT,                            ! ACL context for $CHANGE_ACL
;  229          0228  1          SDEVICE_DESC    : $BBLOCK [DSC$C_S_BLN],        ! Source device desc
;  230          0229  1          SFIB_DESC       : $BBLOCK [DSC$C_S_BLN],        ! Source file FIB desc
;  231          0230  1          SFILE_FIB       : $BBLOCK [FIB$C_LENGTH],       ! Source file FIB
;  232          0231  1          COMMON_CTX,                              ! Common qual context
;  233          0232  1          ATR_ARGLIST     : BLOCKVECTOR [3, ITM$S_ITEM, BYTE],    ! ACP attribute descr
;  234          0233  1          CLI_ACE_DESC    : $BBLOCK [DSC$C_S_BLN],        ! ACE string from CLI
;  235          0234  1          ERROR_POS,                               ! Error position parsing ACE
;  236          0235  1          ACE_DESC        : $BBLOCK [DSC$C_S_BLN],        ! Binary ACE descriptor
;  237          0236  1          ACE             : $BBLOCK [ACL$S_READACL],      ! Binary ACE storage
;  238          0237  1          ACE_POINTER     : REF $BBLOCK,           ! Pointer to ACE queue entry
;  239          0238  1          ACE_TEXT_DESC   : $BBLOCK [DSC$C_S_BLN],        ! Text ACE descriptor
;  240          0239  1          ACE_TEXT        : $BBLOCK [3072],        ! AE text storage
;  241          0240  1          OLD_ACE_HEAD    : $BBLOCK [ACEQ_C_LENGTH],      ! Old ACE queue head
```

```
;  242     0241  1          NEW_ACE_HEAD    : $BBLOCK [ACEQ_C_LENGTH],      ! New ACE queue head
;  243     0242  1          DIR_GROUP,                                      ! Group of UIC format directory
;  244     0243  1          DIR_MEMBER;                                     ! Member of UIC format directory
;  245     0244  1
;  246     0245  1  EXTERNAL
;  247     0246  1          SET$_NOHIDDEN,                                  ! No HIDDEN ACEs allowed
;  248     0247  1          SET$_OBJLOCKED,                                 ! Object locked by another user
;  249     0248  1          SET$_IVORDER,                                   ! Incorrect ordering of ACEs to be modified
;  250     0249  1          SET$_NOSUCHACE,                                 ! Specified ACE doesn't exist
;  251     0250  1          SET$_MODIFIED;                                  ! Object modified message
;  252     0251  1
;  253     0252  1  EXTERNAL ROUTINE
;  254     0253  1          CLI$GET_VALUE,                                  ! Get qualifier value
;  255     0254  1          CLI$PRESENT,                                    ! See if qualifier present
;  256     0255  1          LIB$FID_TO_NAME,                                ! Translate FID to file-spec
;  257     0256  1          LIB$FILE_SCAN,                                  ! Search wildcard file spec
;  258     0257  1          LIB$QUAL_FILE_MATCH,                            ! Check for match
;  259     0258  1          LIB$QUAL_FILE_PARSE,                            ! Set match context
;  260     0259  1          LIB$TPARSE;                                     ! General purpose parser
;  261     0260  1
;  262     0261  1  ! TPARSE table for UIC format directory names.
;  263     0262  1
;  264     0263  1  $INIT_STATE     (DIR_STATE, DIR_KEYS);
;  265     0264  1
;  266     0265  1  $STATE  (,(TPA$_OCTAL,,,DIR_GROUP));
;  267     0266  1  $STATE  (,('.'));
;  268     0267  1  $STATE  (,(TPA$_OCTAL,,,DIR_MEMBER));
```

```
;  270     0268  1 GLOBAL ROUTINE SET_ACL =
;  271     0269  1
;  272     0270  1 !++
;  273     0271  1 !
;  274     0272  1 ! FUNCTIONAL DESCRIPTION:
;  275     0273  1 !
;  276     0274  1 !       This routine is the main routine.  It parses the command line to
;  277     0275  1 !       determine what modifications to the object (or objects) ACL are to
;  278     0276  1 !       occur.
;  279     0277  1 !
;  280     0278  1 !--
;  281     0279  1
;  282     0280  2 BEGIN
;  283     0281  2
;  284     0282  2 BUILTIN
;  285     0283  2         INSQUE;
;  286     0284  2
;  287     0285  2 LOCAL
;  288     0286  2         SCAN_CONTEXT,                           ! LIB$FILE_SCAN context storage
;  289     0287  2         CMD_DESC        : $BBLOCK [DSC$C_S_BLN],        ! DCL command descr
;  290     0288  2         STATUS,                                 ! Local routine return status
;  291     0289  2         IO_STATUS       : VECTOR [4, WORD];     ! I/O status block
;  292     0290  2
;  293     0291  2 ! Initialize local storage.
;  294     0292  2
;  295     0293  2 CH$FILL (0, 3*ITM$S_ITEM, ATR_ARGLIST);
;  296     0294  2 CH$FILL (0, FIB$C_LENGTH, SFILE_FIB);
;  297     0295  2 CH$FILL (0, DSC$C_S_BLN, CLI_ACE_DESC);
;  298     0296  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, ACE_DESC);
;  299     0297  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, ACE_TEXT_DESC);
;  300     0298  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, OBJECT_NAME);
;  301     0299  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, SOBJECT_DESC);
;  302     0300  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, CMD_DESC);
;  303     0301  2 CH$MOVE (DSC$C_S_BLN, CLI_ACE_DESC, SFIB_DESC);
;  304     0302  2
;  305     0303  2 FLAGS = 0;
;  306     0304  2 SCAN_CONTEXT = 0;
;  307     0305  2 OBJECT_TYPE = SOBJECT_TYPE = 0;
;  308     0306  2 CHAN = SCHAN = 0;
;  309     0307  2 WORST_ERROR = SS$_NORMAL;
;  310     0308  2 CLI_ACE_DESC[DSC$B_CLASS] = DSC$K_CLASS_D;
;  311     0309  2 OBJECT_NAME[DSC$B_CLASS] = DSC$K_CLASS_D;
;  312     0310  2 SOBJECT_DESC[DSC$B_CLASS] = DSC$K_CLASS_D;
;  313     0311  2 CMD_DESC[DSC$B_CLASS] = DSC$K_CLASS_D;
;  314     0312  2 SFIB_DESC[DSC$W_LENGTH] = 10;
;  315     0313  2 SFIB_DESC[DSC$A_POINTER] = SFILE_FIB;
;  316     0314  2 ACE_DESC[DSC$A_POINTER] = ACE;
;  317     0315  2 OLD_ACE_HEAD[ACEQ_L_FLINK] = OLD_ACE_HEAD[ACEQ_L_BLINK]
;  318     0316  2                           = OLD_ACE_HEAD[ACEQ_L_FLINK];         ! Null queue
;  319     0317  2 NEW_ACE_HEAD[ACEQ_L_FLINK] = NEW_ACE_HEAD[ACEQ_L_BLINK]
;  320     0318  2                           = NEW_ACE_HEAD[ACEQ_L_FLINK];         ! Null queue
;  321     0319  2
;  322     0320  2 ! Determine what DCL command was used to invoke this image.  Also, set the
;  323     0321  2 ! appropriate default object type code.
;  324     0322  2
;  325     0323  2 CLI$GET_VALUE ($DESCRIPTOR ('OPTION'), CMD_DESC);
;  326     0324  2 IF CH$EQL (.CMD_DESC[DSC$W_LENGTH], .CMD_DESC[DSC$A_POINTER],
```

```
;   327          0325  2              MINU (.CMD_DESC[DSC$W_LENGTH], %CHARCOUNT ('FILE')), UPLIT ('FILE'),
;   328          0326  2              0)
;   329          0327  2          THEN
;   330          0328  3              BEGIN
;   331          0329  3              FLAGS[SET_FILE_CMD] = 1;
;   332          0330  3              OBJECT_TYPE = ACL$C_FILE;
;   333          0331  3              SOBJECT_TYPE = ACL$C_FILE;
;   334          0332  2              END;
;   335          0333  2
;   336          0334  2          IF CH$EQL (.CMD_DESC[DSC$W_LENGTH], .CMD_DESC[DSC$A_POINTER],
;   337          0335  2                     MINU (.CMD_DESC[DSC$W_LENGTH], %CHARCOUNT ('DIRECTORY')), UPLIT ('DIRECTORY'),
;   338          0336  2                     0)
;   339          0337  2          THEN
;   340          0338  3              BEGIN
;   341          0339  3              FLAGS[SET_DIR_CMD] = 1;
;   342          0340  3              OBJECT_TYPE = ACL$C_FILE;
;   343          0341  3              SOBJECT_TYPE = ACL$C_FILE;
;   344          0342  2              END;
;   345          0343  2
;   346          0344  2          IF CH$EQL (.CMD_DESC[DSC$W_LENGTH], .CMD_DESC[DSC$A_POINTER],
;   347          0345  2                     MINU (.CMD_DESC[DSC$W_LENGTH], %CHARCOUNT ('DEVICE')), UPLIT ('DEVICE'),
;   348          0346  2                     0)
;   349          0347  2          THEN
;   350          0348  3              BEGIN
;   351          0349  3              FLAGS[SET_DEV_CMD] = 1;
;   352          0350  3              OBJECT_TYPE = ACL$C_DEVICE;
;   353          0351  3              SOBJECT_TYPE = ACL$C_DEVICE;
;   354          0352  2              END;
;   355          0353  2
;   356          0354  2          IF CH$EQL (.CMD_DESC[DSC$W_LENGTH], .CMD_DESC[DSC$A_POINTER],
;   357          0355  2                     MINU (.CMD_DESC[DSC$W_LENGTH], %CHARCOUNT ('ACL')), UPLIT ('ACL'),
;   358          0356  2                     0)
;   359          0357  2          THEN
;   360          0358  3              BEGIN
;   361          0359  3              FLAGS[SET_ACL_CMD] = 1;
;   362          0360  3              OBJECT_TYPE = ACL$C_FILE;
;   363          0361  3              SOBJECT_TYPE = ACL$C_FILE;
;   364          0362  2              END;
;   365          0363  2
;   366          0364  2          ! Determine what qualifiers are present.
;   367          0365  2
;   368          0366  2          FLAGS[QUAL_AFTER] = CLI$PRESENT ($DESCRIPTOR ('AFTER'));
;   369          0367  2          FLAGS[QUAL_DEFAULT] = CLI$PRESENT ($DESCRIPTOR ('DEFAULT'));
;   370          0368  2          FLAGS[QUAL_DELETE] = CLI$PRESENT ($DESCRIPTOR ('DELETE'));
;   371          0369  2          FLAGS[QUAL_LOG] = CLI$PRESENT ($DESCRIPTOR ('LOG'));
;   372          0370  2          FLAGS[QUAL_REPLACE] = CLI$PRESENT ($DESCRIPTOR ('REPLACE'));
;   373          0371  2          FLAGS[QUAL_NEW] = CLI$PRESENT ($DESCRIPTOR ('NEW'));
;   374          0372  2
;   375          0373  2          ! If the /LIKE qualifier is present, get the source object type and name.  If it
;   376          0374  2          ! is a file, access it for later use.
;   377          0375  2
;   378          0376  3          IF (FLAGS[QUAL_LIKE] = CLI$PRESENT ($DESCRIPTOR ('LIKE')))
;   379          0377  2          THEN
;   380          0378  3              BEGIN
;   381          0379  3
;   382          0380  3          ! Determine the characteristics of the source object.
;   383          0381  3
```

```
;  384          0382  3      IF .FLAGS[SET_ACL_CMD]
;  385          0383  3      THEN
;  386          0384  4          BEGIN
;  387          0385  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.FILE')) THEN SOBJECT_TYPE = ACL$C_FILE;
;  388          0386  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.DEVICE')) THEN SOBJECT_TYPE = ACL$C_DEVICE;
;  389          0387  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.QUEUE')) THEN SOBJECT_TYPE = ACL$C_JOBCTL_QUEUE;
;  390          0388  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.EVENT_CLUSTER')) THEN SOBJECT_TYPE = ACL$C_COMMON_EF
;  391          0389  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.LOGICAL_NAME_TABLE')) THEN SOBJECT_TYPE = ACL$C_LOGIC
;  392          0390  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.PROCESS')) THEN SOBJECT_TYPE = ACL$C_PROCESS;
;  393          0391  4          IF CLI$PRESENT ($DESCRIPTOR ('LIKE.OBJECT_TYPE.GLOBAL_SECTION')) THEN SOBJECT_TYPE = ACL$C_GLOBAL_SE
;  394          0392  4          CLI$GET_VALUE ($DESCRIPTOR ('LIKE.OBJECT_NAME'), SOBJECT_DESC);
;  395          0393  4          END
;  396          0394  3      ELSE CLI$GET_VALUE ($DESCRIPTOR ('LIKE'), SOBJECT_DESC);
;  397          0395  3
;  398          0396  3  ! Attempt to obtain a read lock for the source object.
;  399          0397  3
;  400          0398  3      ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_RLOCK_ACL;
;  401          0399  3      ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_RLOCK_ACL;
;  402          0400  3      ATR_ARGLIST[0, ITM$L_BUFADR] = SACL_LOCKID;
;  403        P 0401  3      STATUS = $CHANGE_ACL (CHAN = .SCHAN,
;  404        P 0402  3                            OBJTYP = SOBJECT_TYPE,
;  405        P 0403  3                            OBJNAM = SOBJECT_DESC,
;  406          0404  3                            ITMLST = ATR_ARGLIST);
;  407          0405  3      IF NOT .STATUS
;  408          0406  3      THEN
;  409          0407  4          BEGIN
;  410          0408  4          IF .STATUS EQL SS$_NOTQUEUED
;  411          0409  5          THEN SIGNAL (SET$_OBJLOCKED)
;  412          0410  4          ELSE SIGNAL (.STATUS);
;  413          0411  4          RETURN .WORST_ERROR;
;  414          0412  4          END;
;  415          0413  3
;  416          0414  3  ! Open the source object to get the ACL being copied; if it is a file.
;  417          0415  3
;  418          0416  3      IF .SOBJECT_TYPE EQL ACL$C_FILE
;  419          0417  3      THEN
;  420          0418  4          BEGIN
;  421        P 0419  4          $FAB_INIT (FAB = SOBJECT_FAB,
;  422        P 0420  4                     FAC = GET,
;  423        P 0421  4                     FNA = .SOBJECT_DESC[DSC$A_POINTER],
;  424        P 0422  4                     FNS = .SOBJECT_DESC[DSC$W_LENGTH],
;  425        P 0423  4                     FOP = UFO,
;  426        P 0424  4                     NAM = SOBJECT_NAM,
;  427          0425  4                     SHR = <GET, UPI>);
;  428        P 0426  4          $NAM_INIT (NAM = SOBJECT_NAM,
;  429        P 0427  4                     ESA = SOBJECT_EXP_NAME,
;  430        P 0428  4                     ESS = NAM$C_MAXRSS,
;  431        P 0429  4                     RSA = SOBJECT_RES_NAME,
;  432          0430  4                     RSS = NAM$C_MAXRSS);
;  433          0431  5          IF NOT $OPEN (FAB = SOBJECT_FAB)
;  434          0432  4          THEN
;  435          0433  5              BEGIN
;  436          0434  5              FILE_ERROR (SET$_OPENIN, SOBJECT_FAB, .SOBJECT_FAB[FAB$L_STS],
;  437          0435  5                                                    .SOBJECT_FAB[FAB$L_STV]);
;  438          0436  5              RETURN SET$_OPENIN OR STS$M_INHIB_MSG;
;  439          0437  4              END;
;  440          0438  4          SCHAN = .SOBJECT_FAB[FAB$L_STV];
```

```
;  441          0439  3          END;
;  442          0440  3      END;
;  443          0441  2
;  444          0442  2  ! Determine the characteristics of the target object.
;  445          0443  2
;  446          0444  2  IF .FLAGS[SET_ACL_CMD]
;  447          0445  2  THEN
;  448          0446  3      BEGIN
;  449          0447  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.FILE')) THEN OBJECT_TYPE = ACL$C_FILE;
;  450          0448  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.DEVICE')) THEN OBJECT_TYPE = ACL$C_DEVICE;
;  451          0449  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.QUEUE')) THEN OBJECT_TYPE = ACL$C_JOBCTL_QUEUE;
;  452          0450  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.EVENT_CLUSTER')) THEN OBJECT_TYPE = ACL$C_COMMON_EF_CLUSTER;
;  453          0451  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.LOGICAL_NAME_TABLE')) THEN OBJECT_TYPE = ACL$C_LOGICAL_NAME_TA
;  454          0452  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.PROCESS')) THEN OBJECT_TYPE = ACL$C_PROCESS;
;  455          0453  3      IF CLI$PRESENT ($DESCRIPTOR ('OBJECT_TYPE.GLOBAL_SECTION')) THEN OBJECT_TYPE = ACL$C_GLOBAL_SECTION;
;  456          0454  3      END;
;  457          0455  2
;  458          0456  2  ! Now get any ACEs specified on the /ACL qualifier.
;  459          0457  2
;  460          0458  2  WHILE CLI$GET_VALUE ($DESCRIPTOR ('ACL'), CLI_ACE_DESC)
;  461          0459  2  DO
;  462          0460  3      BEGIN
;  463          0461  3      ACE_DESC[DSC$W_LENGTH] = ACL$S_READACL;              ! Reset buffer size
;  464        P 0462  3      STATUS = $PARSE_ACL (ACLSTR = CLI_ACE_DESC,
;  465        P 0463  3                           ACLENT = ACE_DESC,
;  466          0464  3                           ERRPOS = ERROR_POS);
;  467          0465  3      IF NOT .STATUS
;  468          0466  3      THEN
;  469          0467  4          BEGIN
;  470          0468  4          CLI_ACE_DESC[DSC$A_POINTER] = .CLI_ACE_DESC[DSC$A_POINTER] + .ERROR_POS;
;  471          0469  4          CLI_ACE_DESC[DSC$W_LENGTH] = .CLI_ACE_DESC[DSC$W_LENGTH] - .ERROR_POS;
;  472          0470  4          SIGNAL (SET$_SYNTAX, 1, CLI_ACE_DESC, .STATUS, 0);
;  473          0471  4          RETURN .WORST_ERROR;
;  474          0472  3          END;
;  475          0473  3      IF .ACE[ACE$V_HIDDEN]
;  476          0474  3      THEN
;  477          0475  4          BEGIN
;  478          0476  4          SIGNAL (SET$_NOHIDDEN);
;  479          0477  4          RETURN .WORST_ERROR;
;  480          0478  3          END;
;  481          0479  3      STATUS = ALLOCATE (.ACE[ACE$B_SIZE] + ACEQ_C_LENGTH, ACE_POINTER);
;  482          0480  3      IF NOT .STATUS
;  483          0481  3      THEN
;  484          0482  4          BEGIN
;  485          0483  4          SIGNAL (.STATUS);
;  486          0484  4          RETURN .WORST_ERROR;
;  487          0485  3          END;
;  488          0486  3      CH$MOVE (.ACE[ACE$B_SIZE], ACE, ACE_POINTER[ACEQ_T_ACE]);
;  489          0487  4      INSQUE (.ACE_POINTER, (IF .FLAGS[QUAL_DELETE] OR .FLAGS[QUAL_REPLACE]
;  490          0488  4                             THEN .OLD_ACE_HEAD[ACEQ_L_BLINK]
;  491          0489  3                             ELSE .NEW_ACE_HEAD[ACEQ_L_BLINK]));
;  492          0490  2      END;
;  493          0491  2
;  494          0492  2  ! Now get any ACEs specified on the /REPLACE or /AFTER qualifiers.
;  495          0493  2
;  496          0494  3  WHILE CLI$GET_VALUE ((IF .FLAGS[QUAL_REPLACE]
;  497          0495  3                        THEN $DESCRIPTOR ('REPLACE')
```

```
;  498          0496  2                          ELSE $DESCRIPTOR ('AFTER')), CLI_ACE_DESC)
;  499          0497  2 DO
;  500          0498  3     BEGIN
;  501          0499  3     ACE_DESC[DSC$W_LENGTH] = ACL$S_READACL;                 ! Reset buffer size
;  502        P 0500  3     STATUS = $PARSE_ACL (ACLSTR = CLI_ACE_DESC,
;  503        P 0501  3                          ACLENT = ACE_DESC,
;  504          0502  3                          ERRPOS = ERROR_POS);
;  505          0503  3     IF NOT .STATUS
;  506          0504  3     THEN
;  507          0505  4         BEGIN
;  508          0506  4         CLI_ACE_DESC[DSC$A_POINTER] = .CLI_ACE_DESC[DSC$A_POINTER] + .ERROR_POS;
;  509          0507  4         CLI_ACE_DESC[DSC$W_LENGTH] = .CLI_ACE_DESC[DSC$W_LENGTH] - .ERROR_POS;
;  510          0508  4         SIGNAL (SET$_SYNTAX, 1, CLI_ACE_DESC, .STATUS, 0);
;  511          0509  4         RETURN .WORST_ERROR;
;  512          0510  3         END;
;  513          0511  3     IF .ACE[ACE$V_HIDDEN]
;  514          0512  3     THEN
;  515          0513  4         BEGIN
;  516          0514  4         SIGNAL (SET$_NOHIDDEN);
;  517          0515  4         RETURN .WORST_ERROR;
;  518          0516  3         END;
;  519          0517  3     STATUS = ALLOCATE (.ACE[ACE$B_SIZE] + ACEQ_C_LENGTH, ACE_POINTER);
;  520          0518  3     IF NOT .STATUS
;  521          0519  3     THEN
;  522          0520  4         BEGIN
;  523          0521  4         SIGNAL (.STATUS);
;  524          0522  4         RETURN .WORST_ERROR;
;  525          0523  3         END;
;  526          0524  3     CH$MOVE (.ACE[ACE$B_SIZE], ACE, ACE_POINTER[ACEQ_T_ACE]);
;  527          0525  4     INSQUE (.ACE_POINTER, (IF .FLAGS[QUAL_REPLACE]
;  528          0526  4                            THEN .NEW_ACE_HEAD[ACEQ_L_BLINK]
;  529          0527  3                            ELSE .OLD_ACE_HEAD[ACEQ_L_BLINK]));
;  530          0528  2     END;
;  531          0529  2
;  532          0530  2 ! Check for syntax errors on the command.
;  533          0531  2
;  534          0532  2 IF .OLD_ACE_HEAD[ACEQ_L_FLINK] EQLA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  535          0533  2 AND .NEW_ACE_HEAD[ACEQ_L_FLINK] EQLA NEW_ACE_HEAD[ACEQ_L_FLINK]
;  536          0534  2 THEN
;  537          0535  3     BEGIN
;  538          0536  3     IF .FLAGS[QUAL_AFTER] OR .FLAGS[QUAL_REPLACE]
;  539          0537  4     OR (.FLAGS[QUAL_NEW] AND NOT .FLAGS[QUAL_LIKE])
;  540          0538  3     THEN
;  541          0539  4         BEGIN
;  542          0540  4         SIGNAL (SET$_SYNTAX, 1, $DESCRIPTOR ('command line'));
;  543          0541  4         RETURN .WORST_ERROR;
;  544          0542  3         END;
;  545          0543  3     END
;  546          0544  2 ELSE
;  547          0545  3     BEGIN
;  548          0546  3     IF .FLAGS[QUAL_LIKE]
;  549          0547  3     THEN
;  550          0548  4         BEGIN
;  551          0549  4         SIGNAL (SET$_SYNTAX, 1, $DESCRIPTOR ('command line'));
;  552          0550  4         RETURN .WORST_ERROR;
;  553          0551  3         END;
;  554          0552  2     END;
```

```
;  555          0553  2
;  556          0554  2 ! If the object is a file, loop through all the specifications supplied.
;  557          0555  2 ! For any other object, simply dispatch to the appropriate routine from here.
;  558          0556  2
;  559          0557  2 IF .OBJECT_TYPE EQL ACL$C_FILE
;  560          0558  2 THEN
;  561          0559  2     BEGIN
;  562        P 0560  2     $FAB_INIT (FAB = OBJECT_FAB,
;  563        P 0561  2                FAC = <GET, PUT>,
;  564        P 0562  2                FOP = UFO,
;  565        P 0563  2                NAM = OBJECT_NAM,
;  566          0564  2                SHR = <GET, UPI>);
;  567        P 0565  2     $NAM_INIT (NAM = OBJECT_NAM,
;  568        P 0566  2                ESA = OBJECT_EXP_NAME,
;  569        P 0567  2                ESS = NAM$C_MAXRSS,
;  570        P 0568  2                RSA = OBJECT_RES_NAME,
;  571          0569  2                RSS = NAM$C_MAXRSS);
;  572          0570  2
;  573          0571  2 ! LIB$QUAL_FILE_PARSE is called to parse the common qualifiers.  It sets up
;  574          0572  2 ! a data base which describes the results for LIB$QUAL_FILE_MATCH to use.
;  575          0573  2
;  576          0574  2     STATUS = LIB$QUAL_FILE_PARSE (%REF (LIB$M_CQF_BEFORE OR
;  577          0575  2                                         LIB$M_CQF_BYOWNER OR
;  578          0576  2                                         LIB$M_CQF_CONFIRM OR
;  579          0577  2                                         LIB$M_CQF_CREATED OR
;  580          0578  2                                         LIB$M_CQF_EXCLUDE OR
;  581          0579  2                                         LIB$M_CQF_SINCE), COMMON_CTX);
;  582          0580  2     IF NOT .STATUS
;  583          0581  3     THEN
;  584          0582  4         BEGIN
;  585          0583  4         SIGNAL (.STATUS);
;  586          0584  4         RETURN .WORST_ERROR;
;  587          0585  3         END;
;  588          0586  3
;  589          0587  3 ! Sit in a loop processing each 'input' file specified.  For the copy
;  590          0588  3 ! operation, the 'input' file is really the output file.
;  591          0589  3
;  592          0590  3     FLAGS[IN_ELLIPSE] = 0;                          ! For initial directory processing
;  593          0591  3     WHILE GET_FILE (OBJECT_FAB)
;  594          0592  3     DO
;  595          0593  4         BEGIN
;  596          0594  4
;  597          0595  4 ! If this is the /DEFAULT processing, and a channel has been assigned,
;  598          0596  4 ! deaccess the directory file, and deassign the channel.
;  599          0597  4
;  600          0598  4         IF .FLAGS[QUAL_DEFAULT] AND .SCHAN NEQ 0
;  601          0599  4         THEN
;  602          0600  5             BEGIN
;  603        P 0601  5             STATUS = $QIOW (CHAN = .SCHAN,
;  604        P 0602  5                             FUNC = IO$_DEACCESS,
;  605          0603  5                             IOSB = IO_STATUS);
;  606          0604  5             IF .STATUS THEN STATUS = .IO_STATUS[0];
;  607          0605  5             IF NOT .STATUS THEN SIGNAL (SET$_CLOSEIN, 1, SOBJECT_DESC, .STATUS, 0);
;  608          0606  5             STATUS = $DASSGN (CHAN = .SCHAN);
;  609          0607  5             IF NOT .STATUS THEN SIGNAL (SET$_CLOSEIN, 1, SOBJECT_DESC, .STATUS, 0);
;  610          0608  5
;  611          0609  5 ! Now release the read lock that was taken out for the directory file.
```

```
;  612          0610  5
;  613          0611  5              ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_UNLOCK_ACL;
;  614          0612  5              ATR_ARGLIST[0, ITM$W_BUFSIZ] = 4;
;  615          0613  5              ATR_ARGLIST[0, ITM$L_BUFADR] = SACL_LOCKID;
;  616        P 0614  5              STATUS = $CHANGE_ACL (CHAN = .SCHAN,
;  617        P 0615  5                                    OBJTYP = SOBJECT_TYPE,
;  618        P 0616  5                                    OBJNAM = SOBJECT_DESC,
;  619          0617  5                                    ITMLST = ATR_ARGLIST);
;  620          0618  5              IF NOT .STATUS THEN SIGNAL (SET$_CLOSEIN, 1, SOBJECT_DESC, .STATUS, 0);
;  621          0619  5              SCHAN = 0;
;  622          0620  4              END;
;  623          0621  4          LIB$FILE_SCAN (OBJECT_FAB,
;  624          0622  4                         PROCESS_FILE,            ! File found action routine
;  625          0623  4                         INPUT_ERROR,             ! Input error action routine
;  626          0624  4                         SCAN_CONTEXT);           ! Stickiness context
;  627          0625  4              END;
;  628          0626  3      END
;  629          0627  2  ELSE
;  630          0628  2      BEGIN
;  631          0629  2
;  632          0630  2  ! Get the object's name.
;  633          0631  2
;  634          0632  2      CLI$GET_VALUE ($DESCRIPTOR ('INPUT'), OBJECT_NAME);
;  635          0633  2
;  636          0634  2  ! Attempt to obtain a write lock for the target object.
;  637          0635  2
;  638          0636  2      ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_WLOCK_ACL;
;  639          0637  2      ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_WLOCK_ACL;
;  640          0638  2      ATR_ARGLIST[0, ITM$L_BUFADR] = ACL_LOCKID;
;  641        P 0639  2      STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  642        P 0640  2                            OBJTYP = OBJECT_TYPE,
;  643        P 0641  2                            OBJNAM = OBJECT_NAME,
;  644          0642  2                            ITMLST = ATR_ARGLIST);
;  645          0643  2      IF NOT .STATUS
;  646          0644  2      THEN
;  647          0645  3          BEGIN
;  648          0646  3          IF .STATUS EQL SS$_NOTQUEUED
;  649          0647  4          THEN SIGNAL (SET$_OBJLOCKED)
;  650          0648  3          ELSE SIGNAL (.STATUS);
;  651          0649  3          RETURN .WORST_ERROR;
;  652          0650  3          END;
;  653          0651  2
;  654          0652  2  ! Call the necessary routine based upon the command line qualifiers.
;  655          0653  2
;  656          0654  2      IF .FLAGS[QUAL_LIKE] THEN STATUS = COPY_ACL (OBJECT_NAME)           ! /LIKE
;  657          0655  2      ELSE IF .FLAGS[QUAL_DELETE] THEN STATUS = DELETE_ACL (OBJECT_NAME)  ! /DELETE
;  658          0656  2           ELSE IF .FLAGS[QUAL_REPLACE] THEN STATUS = REPLACE_ACL (OBJECT_NAME)   ! /REPLACE
;  659          0657  2                ELSE STATUS = ADD_ACL (OBJECT_NAME);                   ! /AFTER, /NEW, or just /ACL
;  660          0658  2
;  661          0659  2  ! If logging is being done, indicate that the object has been modified.
;  662          0660  2
;  663          0661  2      IF .FLAGS[QUAL_LOG] AND .STATUS
;  664          0662  2      THEN SIGNAL (SET$_MODIFIED, 1, OBJECT_NAME);
;  665          0663  2      END;
;  666          0664  1
;  667          0665  2  RETURN .WORST_ERROR;
;  668          0666  2
```

```
;  669          0667  1 END;                    ! End of routine SET_ACL

                                                .TITLE  AED$SETACL
                                                .IDENT  \V04-000\

                                                .PSECT  _LIB$STATE$,NOWRT,  SHR,  PIC,1

                                        00000 DIR_STATE::
                                                .BLKB   0
                                    45F4 00000 ;TPA$TYPE
                                        U.2:    .WORD   17908                           ;
                                00000000* 00002 ;TPA$ADDR
                                        U.3:    .LONG   <<DIR_GROUP-U.3>-4>             ;
                                    042C 00006 ;TPA$TYPE
                                        U.4:    .WORD   1068                            ;
                                    45F4 00008 ;TPA$TYPE
                                        U.5:    .WORD   17908                           ;
                                00000000* 0000A ;TPA$ADDR
                                        U.6:    .LONG   <<DIR_MEMBER-U.6>-4>            ;

                                                .PSECT  _LIB$KEY0$,NOWRT,  SHR,  PIC,1

                                        00000 DIR_KEYS::
                                                .BLKB   0
                                        00000 ;TPA$KEY0
                                        U.1:    .BLKB   0

                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                4E  4F  49  54  50  4F  00000 P.AAB:   .ASCII  \OPTION\                 ;
                                                        00006         .BLKB   2
                                                    00000006  00008 P.AAA:   .LONG   6                  ;
                                                    00000000' 0000C         .ADDRESS P.AAB              ;
                                            45  4C  49  46  00010 P.AAC:   .ASCII  \FILE\               ;
00  00  00  59  52  4F  54  43  45  52  49  44  00014 P.AAD:   .ASCII  \DIRECTORY\<0><0><0>             ;
                    00  00  45  43  49  56  45  44  00020 P.AAE:   .ASCII  \DEVICE\<0><0>               ;
                                        00  4C  43  41  00028 P.AAF:   .ASCII  \ACL\<0>                 ;
                                            52  45  54  46  41  0002C P.AAH:   .ASCII  \AFTER\          ;
                                                        00031         .BLKB   3
                                                    00000005  00034 P.AAG:   .LONG   5                  ;
                                                    00000000' 00038         .ADDRESS P.AAH              ;
                                    54  4C  55  41  46  45  44  0003C P.AAJ:   .ASCII  \DEFAULT\         ;
                                                        00043         .BLKB   1
                                                    00000007  00044 P.AAI:   .LONG   7                  ;
                                                    00000000' 00048         .ADDRESS P.AAJ              ;
                                        45  54  45  4C  45  44  0004C P.AAL:   .ASCII  \DELETE\         ;
                                                        00052         .BLKB   2
                                                    00000006  00054 P.AAK:   .LONG   6                  ;
                                                    00000000' 00058         .ADDRESS P.AAL              ;
                                                47  4F  4C  0005C P.AAN:   .ASCII  \LOG\                ;
                                                        0005F         .BLKB   1
                                                    00000003  00060 P.AAM:   .LONG   3                  ;
                                                    00000000' 00064         .ADDRESS P.AAN              ;
                                    45  43  41  4C  50  45  52  00068 P.AAP:   .ASCII  \REPLACE\         ;
                                                        0006F         .BLKB   1
                                                    00000007  00070 P.AAO:   .LONG   7                  ;
```

```
                                                   00000000'  00074          .ADDRESS P.AAP                              ;
                                                 57  45  4E  00078 P.AAR:   .ASCII  \NEW\                                ;
                                                              0007B          .BLKB   1
                                                   00000003   0007C P.AAQ:   .LONG   3                                    ;
                                                   00000000'  00080          .ADDRESS P.AAR                              ;
                                             45  4B  49  4C  00084 P.AAT:   .ASCII  \LIKE\                               ;
                                                   00000004   00088 P.AAS:   .LONG   4                                    ;
                                                   00000000'  0008C          .ADDRESS P.AAT                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  00090 P.AAV:   .ASCII  \LIKE.OBJECT_TYPE.FILE\              ;
                                45  4C  49  46  2E  45  0009F                                                        ;
                                                              000A5          .BLKB   3
                                                   00000015   000A8 P.AAU:   .LONG   21                                   ;
                                                   00000000'  000AC          .ADDRESS P.AAV                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  000B0 P.AAX:   .ASCII  \LIKE.OBJECT_TYPE.DEVICE\            ;
                            45  43  49  56  45  44  2E  45  000BF                                                        ;
                                                              000C7          .BLKB   1
                                                   00000017   000C8 P.AAW:   .LONG   23                                   ;
                                                   00000000'  000CC          .ADDRESS P.AAX                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  000D0 P.AAZ:   .ASCII  \LIKE.OBJECT_TYPE.QUEUE\             ;
                                45  55  45  55  51  2E  45  000DF                                                        ;
                                                              000E6          .BLKB   2
                                                   00000016   000E8 P.AAY:   .LONG   22                                   ;
                                                   00000000'  000EC          .ADDRESS P.AAZ                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  000F0 P.ABB:   .ASCII  \LIKE.OBJECT_TYPE.EVENT_CLUSTER\     ;
52  45  54  53  55  4C  43  5F  54  4E  45  56  45  2E  45  000FF                                                        ;
                                                              0010E          .BLKB   2
                                                   0000001E   00110 P.ABA:   .LONG   30                                   ;
                                                   00000000'  00114          .ADDRESS P.ABB                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  00118 P.ABD:   .ASCII  \LIKE.OBJECT_TYPE.LOGICAL_NAME_TABLE\ ;
5F  45  4D  41  4E  5F  4C  41  43  49  47  4F  4C  2E  45  00127                                                        ;
                                        45  4C  42  41  54  00136                                                        ;
                                                              0013B          .BLKB   1
                                                   00000023   0013C P.ABC:   .LONG   35                                   ;
                                                   00000000'  00140          .ADDRESS P.ABD                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  00144 P.ABF:   .ASCII  \LIKE.OBJECT_TYPE.PROCESS\           ;
                        53  53  45  43  4F  52  50  2E  45  00153                                                        ;
                                                   00000018   0015C P.ABE:   .LONG   24                                   ;
                                                   00000000'  00160          .ADDRESS P.ABF                              ;
50  59  54  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  00164 P.ABH:   .ASCII  \LIKE.OBJECT_TYPE.GLOBAL_SECTION\    ;
4F  49  54  43  45  53  5F  4C  41  42  4F  4C  47  2E  45  00173                                                        ;
                                                          4E  00182                                                        ;
                                                              00183          .BLKB   1
                                                   0000001F   00184 P.ABG:   .LONG   31                                   ;
                                                   00000000'  00188          .ADDRESS P.ABH                              ;
4D  41  4E  5F  54  43  45  4A  42  4F  2E  45  4B  49  4C  0018C P.ABJ:   .ASCII  \LIKE.OBJECT_NAME\                   ;
                                                          45  0019B                                                        ;
                                                   00000010   0019C P.ABI:   .LONG   16                                   ;
                                                   00000000'  001A0          .ADDRESS P.ABJ                              ;
                                             45  4B  49  4C  001A4 P.ABL:   .ASCII  \LIKE\                               ;
                                                   00000004   001A8 P.ABK:   .LONG   4                                    ;
                                                   00000000'  001AC          .ADDRESS P.ABL                              ;
4C  49  46  2E  45  50  59  54  5F  54  43  45  4A  42  4F  001B0 P.ABN:   .ASCII  \OBJECT_TYPE.FILE\                   ;
                                                          45  001BF                                                        ;
                                                   00000010   001C0 P.ABM:   .LONG   16                                   ;
                                                   00000000'  001C4          .ADDRESS P.ABN                              ;
56  45  44  2E  45  50  59  54  5F  54  43  45  4A  42  4F  001C8 P.ABP:   .ASCII  \OBJECT_TYPE.DEVICE\                 ;
                                              45  43  49  001D7                                                        ;
```

```
                                                        001DA                   .BLKB   2
                                                00000012.001DC P.ABO:   .LONG   18
                                                00000000'001E0          .ADDRESS P.ABP
45 55 51 2E 45 50 59 54 5F 54 43 45 4A 42 4F    001E4 P.ABR:   .ASCII  \OBJECT_TYPE.QUEUE\
                                        45 55   001F3
                                                001F5                   .BLKB   3
                                                00000011.001F8 P.ABQ:   .LONG   17
                                                00000000'001FC          .ADDRESS P.ABR
45 56 45 2E 45 50 59 54 5F 54 43 45 4A 42 4F    00200 P.ABT:   .ASCII  \OBJECT_TYPE.EVENT_CLUSTER\
                  52 45 54 53 55 4C 43 5F 54 4E 0020F
                                                00219                   .BLKB   3
                                                00000019.0021C P.ABS:   .LONG   25
                                                00000000'00220          .ADDRESS P.ABT
47 4F 4C 2E 45 50 59 54 5F 54 43 45 4A 42 4F    00224 P.ABV:   .ASCII  \OBJECT_TYPE.LOGICAL_NAME_TABLE\
45 4C 42 41 54 5F 45 4D 41 4E 5F 4C 41 43 49    00233
                                                00242                   .BLKB   2
                                                0000001E.00244 P.ABU:   .LONG   30
                                                00000000'00248          .ADDRESS P.ABV
4F 52 50 2E 45 50 59 54 5F 54 43 45 4A 42 4F    0024C P.ABX:   .ASCII  \OBJECT_TYPE.PROCESS\
                                        53 53 45 43 0025B
                                                0025F                   .BLKB   1
                                                00000013.00260 P.ABW:   .LONG   19
                                                00000000'00264          .ADDRESS P.ABX
4F 4C 47 2E 45 50 59 54 5F 54 43 45 4A 42 4F    00268 P.ABZ:   .ASCII  \OBJECT_TYPE.GLOBAL_SECTION\
                  4E 4F 49 54 43 45 53 5F 4C 41 42 00277
                                                00282                   .BLKB   2
                                                0000001A.00284 P.ABY:   .LONG   26
                                                00000000'00288          .ADDRESS P.ABZ
                                        4C 43 41    0028C P.ACB:   .ASCII  \ACL\
                                                0028F                   .BLKB   1
                                                00000003.00290 P.ACA:   .LONG   3
                                                00000000'00294          .ADDRESS P.ACB
                        45 43 41 4C 50 45 52    00298 P.ACD:   .ASCII  \REPLACE\
                                                0029F                   .BLKB   1
                                                00000007.002A0 P.ACC:   .LONG   7
                                                00000000'002A4          .ADDRESS P.ACD
                              52 45 54 46 41    002A8 P.ACF:   .ASCII  \AFTER\
                                                002AD                   .BLKB   3
                                                00000005.002B0 P.ACE:   .LONG   5
                                                00000000'002B4          .ADDRESS P.ACF
65 6E 69 6C 20 64 6E 61 6D 6D 6F 63             002B8 P.ACH:   .ASCII  \command line\
                                                0000000C.002C4 P.ACG:   .LONG   12
                                                00000000'002C8          .ADDRESS P.ACH
65 6E 69 6C 20 64 6E 61 6D 6D 6F 63             002CC P.ACJ:   .ASCII  \command line\
                                                0000000C.002D8 P.ACI:   .LONG   12
                                                00000000'002DC          .ADDRESS P.ACJ
                              54 55 50 4E 49    002E0 P.ACL:   .ASCII  \INPUT\
                                                002E5                   .BLKB   3
                                                00000005.002E8 P.ACK:   .LONG   5
                                                00000000'002EC          .ADDRESS P.ACL

                                                                        .PSECT  $OWN$,NOEXE,2

                                                00000 FLAGS:    .BLKB   2
                                                00002           .BLKB   2
                                                00004 WORST_ERROR:
                                                                .BLKB   4
```

```
00008 ACL_LOCKID:
                .BLKB   4
0000C OBJECT_TYPE:
                .BLKB   4
00010 OBJECT_NAME:
                .BLKB   8
00018 OBJECT_FAB:
                .BLKB   80
00068 OBJECT_NAM:
                .BLKB   96
000C8 OBJECT_EXP_NAME:
                .BLKB   255
001C7           .BLKB   1
001C8 OBJECT_RES_NAME:
                .BLKB   255
002C7           .BLKB   1
002C8 RELATED_NAM:
                .BLKB   96
00328 CHAN:     .BLKB   4
0032C ACL_CONTEXT:
                .BLKB   4
00330 SACL_LOCKID:
                .BLKB   4
00334 SOBJECT_TYPE:
                .BLKB   4
00338 SOBJECT_DESC:
                .BLKB   8
00340 SOBJECT_FAB:
                .BLKB   80
00390 SOBJECT_NAM:
                .BLKB   96
003F0 SOBJECT_EXP_NAME:
                .BLKB   255
004EF           .BLKB   1
004F0 SOBJECT_RES_NAME:
                .BLKB   255
005EF           .BLKB   1
005F0 SCHAN:    .BLKB   4
005F4 SACL_CONTEXT:
                .BLKB   4
005F8 SDEVICE_DESC:
                .BLKB   8
00600 SFIB_DESC:
                .BLKB   8
00608 SFILE_FIB:
                .BLKB   64
00648 COMMON_CTX:
                .BLKB   4
0064C ATR_ARGLIST:
                .BLKB   36
00670 CLI_ACE_DESC:
                .BLKB   8
00678 ERROR_POS:
                .BLKB   4
0067C ACE_DESC:
                .BLKB   8
00684 ACE:      .BLKB   512
```

```
00884 ACE_POINTER:
                .BLKB   4
00888 ACE_TEXT_DESC:
                .BLKB   8
00890 ACE_TEXT:
                .BLKB   3072
01490 OLD_ACE_HEAD:
                .BLKB   8
01498 NEW_ACE_HEAD:
                .BLKB   8
014A0 DIR_GROUP:
                .BLKB   4
014A4 DIR_MEMBER:
                .BLKB   4

        $RMS_PTR=           SOBJECT_FAB
        $RMS_PTR=           SOBJECT_NAM
        $RMS_PTR=           OBJECT_FAB
        $RMS_PTR=           OBJECT_NAM
                .EXTRN  SET$_NOHIDDEN, SET$_OBJLOCKED
                .EXTRN  SET$_IVORDER, SET$_NOSUCHACE
                .EXTRN  SET$_MODIFIED, CLI$GET_VALUE
                .EXTRN  CLI$PRESENT, LIB$FID_TO_NAME
                .EXTRN  LIB$FILE_SCAN, LIB$QUAL_FILE_MATCH
                .EXTRN  LIB$QUAL_FILE_PARSE
                .EXTRN  LIB$TPARSE, SYS$CHANGE_ACL
                .EXTRN  LIB$SIGNAL, SYS$OPEN
                .EXTRN  SYS$PARSE_ACL, LIB$GET_VM
                .EXTRN  SYS$QIOW, SYS$DASSGN

                .PSECT  $CODE$,NOWRT,2

                                        0FFC 00000      .ENTRY  SET_ACL, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,- ; 0268
                                                                R11
                                5B 00000000G    00  9E 00002  MOVAB   LIB$SIGNAL, R11
                                5A      0000'   CF  9E 00009  MOVAB   P.AAA, R10
                                59 00000000G    00  9E 0000E  MOVAB   CLI$PRESENT, R9
                                58      0000'   CF  9E 00015  MOVAB   FLAGS, R8
                                5E              18  C2 0001A  SUBL2   #24, SP
        24              00      6E              00  2C 0001D  MOVC5   #0, (SP), #0, #36, ATR_ARGLIST          ; 0293
                                        064C    C8      00022
0040    8F              00      6E              00  2C 00025  MOVC5   #0, (SP), #0, #64, SFILE_FIB           ; 0294
                                        0608    C8      0002C
        08              00      6E              00  2C 0002F  MOVC5   #0, (SP), #0, #8, CLI_ACE_DESC        ; 0295
                                        0670    C8      00034
                067C    C8  0670    C8          08  28 00037  MOVC3   #8, CLI_ACE_DESC, ACE_DESC            ; 0296
                0888    C8  0670    C8          08  28 0003F  MOVC3   #8, CLI_ACE_DESC, ACE_TEXT_DESC       ; 0297
                  10    A8  0670    C8          08  28 00047  MOVC3   #8, CLI_ACE_DESC, OBJECT_NAME         ; 0298
                0338    C8  0670    C8          08  28 0004E  MOVC3   #8, CLI_ACE_DESC, SOBJECT_DESC        ; 0299
                  10    AE  0670    C8          08  28 00056  MOVC3   #8, CLI_ACE_DESC, CMD_DESC           ; 0300
                0600    C8  0670    C8          08  28 0005D  MOVC3   #8, CLI_ACE_DESC, SFIB_DESC          ; 0301
                                                68  B4 00065  CLRW    FLAGS                               ; 0303
                                          04    AE  D4 00067  CLRL    SCAN_CONTEXT                        ; 0304
                                        0334    C8  D4 0006A  CLRL    SOBJECT_TYPE                        ; 0305
                                          0C    A8  D4 0006E  CLRL    OBJECT_TYPE
                                        05F0    C8  D4 00071  CLRL    SCHAN                               ; 0306
                                        0328    C8  D4 00075  CLRL    CHAN
```

```
                        04    A8          01   D0  00079          MOVL    #1, WORST_ERROR                              ; 0307
                      0673    C8          02   90  0007D          MOVB    #2, CLI_ACE_DESC+3                           ; 0308
                        13    A8          02   90  00082          MOVB    #2, OBJECT_NAME+3                            ; 0309
                      033B    C8          02   90  00086          MOVB    #2, SOBJECT_DESC+3                           ; 0310
                        13    AE          02   90  0008B          MOVB    #2, CMD_DESC+3                               ; 0311
                      0600    C8          0A   B0  0008F          MOVW    #10, SFIB_DESC                               ; 0312
                      0604    C8   0608   C8   9E  00094          MOVAB   SFILE_FIB, SFIB_DESC+4                       ; 0313
                      0680    C8   0684   C8   9E  0009B          MOVAB   ACE, ACE_DESC+4                              ; 0314
                        50         1490   C8   9E  000A2          MOVAB   OLD_ACE_HEAD, R0                             ; 0316
                      1494    C8          50   D0  000A7          MOVL    R0, OLD_ACE_HEAD+4                           ;
                      1490    C8          50   D0  000AC          MOVL    R0, OLD_ACE_HEAD                             ;
                        50         1498   C8   9E  000B1          MOVAB   NEW_ACE_HEAD, R0                             ; 0318
                      149C    C8          50   D0  000B6          MOVL    R0, NEW_ACE_HEAD+4                           ;
                      1498    C8          50   D0  000BB          MOVL    R0, NEW_ACE_HEAD                             ;
                                     10   AE   9F  000C0          PUSHAB  CMD_DESC                                     ; 0323
                                          5A   DD  000C3          PUSHL   R10                                          ;
                 00000000G    00          02   FB  000C5          CALLS   #2, CLI$GET_VALUE                            ;
                              54     10   AE   3C  000CC          MOVZWL  CMD_DESC, R4                                 ; 0324
                              50          54   D0  000D0          MOVL    R4, R0                                       ; 0325
                              04          50   B1  000D3          CMPW    R0, #4                                       ;
                                          03   1B  000D6          BLEQU   1$                                           ;
                              50          04   D0  000D8          MOVL    #4, R0                                       ;
  50          00        14    BE          54   2D  000DB  1$:     CMPC5   R4, @CMD_DESC+4, #0, R0, P.AAC               ; 0324
                                     08   AA       000E1                                                               ;
                                          0D   12  000E3          BNEQ    2$                                           ;
                        01    A8          20   88  000E5          BISB2   #32, FLAGS+1                                 ; 0329
                        0C    A8          01   D0  000E9          MOVL    #1, OBJECT_TYPE                              ; 0330
                      0334    C8          01   D0  000ED          MOVL    #1, SOBJECT_TYPE                             ; 0331
                              50          54   D0  000F2  2$:     MOVL    R4, R0                                       ; 0335
                              09          50   B1  000F5          CMPW    R0, #9                                       ;
                                          03   1B  000F8          BLEQU   3$                                           ;
                              50          09   D0  000FA          MOVL    #9, R0                                       ;
  50          00        14    BE          54   2D  000FD  3$:     CMPC5   R4, @CMD_DESC+4, #0, R0, P.AAD               ; 0334
                                     0C   AA       00103                                                               ;
                                          0E   12  00105          BNEQ    4$                                           ;
                        01    A8     40   8F   88  00107          BISB2   #64, FLAGS+1                                 ; 0339
                        0C    A8          01   D0  0010C          MOVL    #1, OBJECT_TYPE                              ; 0340
                      0334    C8          01   D0  00110          MOVL    #1, SOBJECT_TYPE                             ; 0341
                              50          54   D0  00115  4$:     MOVL    R4, R0                                       ; 0345
                              06          50   B1  00118          CMPW    R0, #6                                       ;
                                          03   1B  0011B          BLEQU   5$                                           ;
                              50          06   D0  0011D          MOVL    #6, R0                                       ;
  50          00        14    BE          54   2D  00120  5$:     CMPC5   R4, @CMD_DESC+4, #0, R0, P.AAE               ; 0344
                                     18   AA       00126                                                               ;
                                          0D   12  00128          BNEQ    6$                                           ;
                        01    A8          10   88  0012A          BISB2   #16, FLAGS+1                                 ; 0349
                        0C    A8          02   D0  0012E          MOVL    #2, OBJECT_TYPE                              ; 0350
                      0334    C8          02   D0  00132          MOVL    #2, SOBJECT_TYPE                             ; 0351
                              50          54   D0  00137  6$:     MOVL    R4, R0                                       ; 0355
                              03          50   B1  0013A          CMPW    R0, #3                                       ;
                                          03   1B  0013D          BLEQU   7$                                           ;
                              50          03   D0  0013F          MOVL    #3, R0                                       ;
  50          00        14    BE          54   2D  00142  7$:     CMPC5   R4, @CMD_DESC+4, #0, R0, P.AAF               ; 0354
                                     20   AA       00148                                                               ;
                                          0E   12  0014A          BNEQ    8$                                           ;
                        01    A8     80   8F   88  0014C          BISB2   #128, FLAGS+1                                ; 0359
                        0C    A8          01   D0  00151          MOVL    #1, OBJECT_TYPE                              ; 0360
```

```
                                0334   C8              01   D0  00155          MOVL    #1, SOBJECT_TYPE                        ; 0361
                                              2C        AA   9F  0015A  8$:     PUSHAB  P.AAG                                   ; 0366
                                       69             01   FB  0015D          CALLS   #1, CLI$PRESENT
                                       68       01    00   F0  00160          INSV    R0, #0, #1, FLAGS
                                              3C        AA   9F  00165          PUSHAB  P.AAI                                   ; 0367
                                       69             01   FB  00168          CALLS   #1, CLI$PRESENT
                                       68       01    06   F0  0016B          INSV    R0, #6, #1, FLAGS
                                              4C        AA   9F  00170          PUSHAB  P.AAK                                   ; 0368
                                       69             01   FB  00173          CALLS   #1, CLI$PRESENT
                                       68       01    01   F0  00176          INSV    R0, #1, #1, FLAGS
                                              58        AA   9F  0017B          PUSHAB  P.AAM                                   ; 0369
                                       69             01   FB  0017E          CALLS   #1, CLI$PRESENT
                                       68       01    03   F0  00181          INSV    R0, #3, #1, FLAGS
                                              68        AA   9F  00186          PUSHAB  P.AAO                                   ; 0370
                                       69             01   FB  00189          CALLS   #1, CLI$PRESENT
                                       68       01    04   F0  0018C          INSV    R0, #4, #1, FLAGS
                                              74        AA   9F  00191          PUSHAB  P.AAQ                                   ; 0371
                                       69             01   FB  00194          CALLS   #1, CLI$PRESENT
                                       68       01    05   F0  00197          INSV    R0, #5, #1, FLAGS
                                            0080        CA   9F  0019C          PUSHAB  P.AAS                                   ; 0376
                                       69             01   FB  001A0          CALLS   #1, CLI$PRESENT
                                       68       01    02   F0  001A3          INSV    R0, #2, #1, FLAGS
                                                      03   50  E8  001A8          BLBS    R0, 9$
                                                           015A  31  001AB          BRW     22$
                                              01        A8   95  001AE  9$:     TSTB    FLAGS+1                                 ; 0382
                                                           73  18  001B1          BGEQ    17$
                                            00A0        CA   9F  001B3          PUSHAB  P.AAU                                   ; 0385
                                       69             01   FB  001B7          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  001BA          BLBC    R0, 10$
                                0334   C8              01   D0  001BD          MOVL    #1, SOBJECT_TYPE
                                            00C0        CA   9F  001C2  10$:     PUSHAB  P.AAW                                   ; 0386
                                       69             01   FB  001C6          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  001C9          BLBC    R0, 11$
                                0334   C8              02   D0  001CC          MOVL    #2, SOBJECT_TYPE
                                            00E0        CA   9F  001D1  11$:     PUSHAB  P.AAY                                   ; 0387
                                       69             01   FB  001D5          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  001D8          BLBC    R0, 12$
                                0334   C8              03   D0  001DB          MOVL    #3, SOBJECT_TYPE
                                            0108        CA   9F  001E0  12$:     PUSHAB  P.ABA                                   ; 0388
                                       69             01   FB  001E4          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  001E7          BLBC    R0, 13$
                                0334   C8              04   D0  001EA          MOVL    #4, SOBJECT_TYPE
                                            0134        CA   9F  001EF  13$:     PUSHAB  P.ABC                                   ; 0389
                                       69             01   FB  001F3          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  001F6          BLBC    R0, 14$
                                0334   C8              05   D0  001F9          MOVL    #5, SOBJECT_TYPE
                                            0154        CA   9F  001FE  14$:     PUSHAB  P.ABE                                   ; 0390
                                       69             01   FB  00202          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  00205          BLBC    R0, 15$
                                0334   C8              06   D0  00208          MOVL    #6, SOBJECT_TYPE
                                            017C        CA   9F  0020D  15$:     PUSHAB  P.ABG                                   ; 0391
                                       69             01   FB  00211          CALLS   #1, CLI$PRESENT
                                                      05   50  E9  00214          BLBC    R0, 16$
                                0334   C8              07   D0  00217          MOVL    #7, SOBJECT_TYPE
                                            0338        C8   9F  0021C  16$:     PUSHAB  SOBJECT_DESC                            ; 0392
                                            0194        CA   9F  00220          PUSHAB  P.ABI
                                                           08  11  00224          BRB     18$
```

```
                                                0338   C8  9F 00226 17$:    PUSHAB  SOBJECT_DESC                          ; 0394
                                                01A0   CA  9F 0022A       PUSHAB  P.ABK                                 ;
                                  00000000G     00       02  FB 0022E 18$:    CALLS   #2, CLI$GET_VALUE                 ;
                                      064C   C8 000A0004   8F  D0 00235       MOVL    #655364, ATR_ARGLIST              ; 0399
                                      0650   C8     0330   C8  9E 0023E       MOVAB   SACL_LOCKID, ATR_ARGLIST+4        ; 0400
                                                           7E  7C 00245       CLRQ    -(SP)                             ; 0404
                                                           7E  D4 00247       CLRL    -(SP)                             ;
                                                0064C  C8  9F  00249       PUSHAB  ATR_ARGLIST                       ;
                                                0338   C8  9F  0024D       PUSHAB  SOBJECT_DESC                      ;
                                                0334   C8  9F  00251       PUSHAB  SOBJECT_TYPE                      ;
                                                05F0   C8  DD  00255       PUSHL   SCHAN                             ;
                                  00000000G     00       07  FB 00259       CALLS   #7, SYS$CHANGE_ACL                ;
                                                57       50  D0 00260       MOVL    R0, STATUS                        ;
                                                03       57  E8 00263       BLBS    STATUS, 19$                       ; 0405
                                                      04B9  31  00266       BRW     71$                               ;
                                                01     0334   C8  D1 00269 19$:    CMPL    SOBJECT_TYPE, #1          ; 0416
                                                           03  13 0026E       BEQL    20$                               ;
                                                      0095  31  00270       BRW     22$                               ;
    0050    8F          00                        6E       00  2C 00273 20$:    MOVC5   #0, (SP), #0, #80, $RMS_PTR       ; 0425
                                                      0340   C8      0027A       ;
                                          0340   C8     5003   8F  B0 0027D       MOVW    #20483, $RMS_PTR                  ;
                                          0344   C8 00020000   8F  D0 00284       MOVL    #131072, $RMS_PTR+4               ;
                                          0356   C8     4202   8F  B0 0028D       MOVW    #16898, $RMS_PTR+22               ;
                                          035F   C8            02  90 00294       MOVB    #2, $RMS_PTR+31                   ;
                                          0368   C8     0390   C8  9E 00299       MOVAB   SOBJECT_NAM, $RMS_PTR+40          ;
                                          036C   C8     033C   C8  D0 002A0       MOVL    SOBJECT_DESC+4, $RMS_PTR+44       ;
                                          0374   C8     0338   C8  90 002A7       MOVB    SOBJECT_DESC, $RMS_PTR+52         ;
    0060    8F          00                        6E       00  2C 002AE       MOVC5   #0, (SP), #0, #96, $RMS_PTR       ; 0430
                                                      0390   C8      002B5       ;
                                          0390   C8     6002   8F  B0 002B8       MOVW    #24578, $RMS_PTR                  ;
                                          0392   C8            01  8E 002BF       MNEGB   #1, $RMS_PTR+2                    ;
                                          0394   C8     04F0   C8  9E 002C4       MOVAB   SOBJECT_RES_NAME, $RMS_PTR+4      ;
                                          039A   C8            01  8E 002CB       MNEGB   #1, $RMS_PTR+10                   ;
                                          039C   C8     03F0   C8  9E 002D0       MOVAB   SOBJECT_EXP_NAME, $RMS_PTR+12     ;
                                                0340   C8  9F  002D7       PUSHAB  SOBJECT_FAB                       ; 0431
                                  00000000G     00       01  FB 002DB       CALLS   #1, SYS$OPEN                      ;
                                                1C       50  E8 002E2       BLBS    R0, 21$                           ;
                                                7E     0348   C8  7D 002E5       MOVQ    SOBJECT_FAB+8, -(SP)              ; 0434
                                                0340   C8  9F  002EA       PUSHAB  SOBJECT_FAB                       ;
                                          0077109A     8F  DD  002EE       PUSHL   #7803034                          ;
                                          0000V  CF        04  FB 002F4       CALLS   #4, FILE_ERROR                    ;
                                                50 1077109A   8F  D0 002F9       MOVL    #276238490, R0                    ; 0436
                                                           04  00300       RET                                       ;
                                          05F0   C8     034C   C8  D0 00301 21$:    MOVL    SOBJECT_FAB+12, SCHAN             ; 0438
                                                   01     A8  95 00308 22$:    TSTB    FLAGS+1                           ; 0444
                                                           62  18 0030B       BGEQ    29$                               ;
                                                0001B8 CA  9F  0030D       PUSHAB  P.ABM                             ; 0447
                                                69         01  FB 00311       CALLS   #1, CLI$PRESENT                   ;
                                                04         50  E9 00314       BLBC    R0, 23$                           ;
                                          0C     A8            01  D0 00317       MOVL    #1, OBJECT_TYPE                   ;
                                                      01D4   CA  9F  0031B 23$:    PUSHAB  P.ABO                             ; 0448
                                                69         01  FB 0031F       CALLS   #1, CLI$PRESENT                   ;
                                                04         50  E9 00322       BLBC    R0, 24$                           ;
                                          0C     A8            02  D0 00325       MOVL    #2, OBJECT_TYPE                   ;
                                                      01F0   CA  9F  00329 24$:    PUSHAB  P.ABQ                             ; 0449
                                                69         01  FB 0032D       CALLS   #1, CLI$PRESENT                   ;
                                                04         50  E9 00330       BLBC    R0, 25$                           ;
```

```
                    0C   A8              03   D0 00333              MOVL    #3, OBJECT_TYPE                                 ;
                                  0214  CA   9F 00337 25$:          PUSHAB  P.ABS                                           ; 0450
                    69                  01   FB 0033B              CALLS   #1, CLI$PRESENT                                 ;
                    04                  50   E9 0033E              BLBC    R0, 26$                                         ;
                    0C   A8              04   D0 00341              MOVL    #4, OBJECT_TYPE                                 ;
                                  023C  CA   9F 00345 26$:          PUSHAB  P.ABU                                           ; 0451
                    69                  01   FB 00349              CALLS   #1, CLI$PRESENT                                 ;
                    04                  50   E9 0034C              BLBC    R0, 27$                                         ;
                    0C   A8              05   D0 0034F              MOVL    #5, OBJECT_TYPE                                 ;
                                  0258  CA   9F 00353 27$:          PUSHAB  P.ABW                                           ; 0452
                    69                  01   FB 00357              CALLS   #1, CLI$PRESENT                                 ;
                    04                  50   E9 0035A              BLBC    R0, 28$                                         ;
                    0C   A8              06   D0 0035D              MOVL    #6, OBJECT_TYPE                                 ;
                                  027C  CA   9F 00361 28$:          PUSHAB  P.ABY                                           ; 0453
                    69                  01   FB 00365              CALLS   #1, CLI$PRESENT                                 ;
                    04                  50   E9 00368              BLBC    R0, 29$                                         ;
                    0C   A8              07   D0 0036B              MOVL    #7, OBJECT_TYPE                                 ;
                                  0670  C8   9F 0036F 29$:          PUSHAB  CLI_ACE_DESC                                    ; 0458
                                  0288  CA   9F 00373              PUSHAB  P.ACA                                           ;
              00000000G           00        02   FB 00377              CALLS   #2, CLI$GET_VALUE                               ;
                                    03        50   E8 0037E              BLBS    R0, 30$                                         ;
                                            00DB   31   00381              BRW     41$                                             ;
                         067C   C8      0200   8F   B0 00384 30$:          MOVW    #512, ACE_DESC                                  ; 0461
                                               7E   D4 0038B              CLRL    -(SP)                                           ; 0464
                                         0678  C8   9F 0038D              PUSHAB  ERROR_POS                                       ;
                                         067C  C8   9F 00391              PUSHAB  ACE_DESC                                        ;
                                         0670  C8   9F 00395              PUSHAB  CLI_ACE_DESC                                    ;
              00000000G           00        04   FB 00399              CALLS   #4, SYS$PARSE_ACL                               ;
                                    57        50   D0 003A0              MOVL    R0, STATUS                                      ;
                                    24        57   E8 003A3              BLBS    STATUS, 32$                                     ; 0465
                         0674   C8      0678   C8   C0 003A6 31$:          ADDL2   ERROR_POS, CLI_ACE_DESC+4                       ; 0468
                         0670   C8      0678   C8   A2 003AD              SUBW2   ERROR_POS, CLI_ACE_DESC                         ; 0469
                                               7E   D4 003B4              CLRL    -(SP)                                           ; 0470
                                               57   DD 003B6              PUSHL   STATUS                                          ;
                                         0670  C8   9F 003B8              PUSHAB  CLI_ACE_DESC                                    ;
                                               01   DD 003BC              PUSHL   #1                                              ;
                                    007710FC   8F   DD 003BE              PUSHL   #7803132                                        ;
                                    6B              05   FB 003C4              CALLS   #5, LIB$SIGNAL                                  ;
                                                  00DD   31   003C7              BRW     45$                                             ;
                              2D      0687   C8          02   E1 003CA 32$:          BBC     #2, ACE+3, 35$                          ; 0473
                                    00000000G   00   9F 003D0 33$:          PUSHAB  SET$_NOHIDDEN                           ; 0476
                                    6B          01   FB 003D6              CALLS   #1, LIB$SIGNAL                          ;
                                    50 00000000G   00   9E 003D9              MOVAB   SET$_NOHIDDEN, R0                       ;
                                    17          50   E8 003E0              BLBS    R0, 34$                                 ;
                                    50 00000000*   00   9E 003E3              MOVAB   <SET$_NOHIDDEN&7>, R0                   ;
                50        04   A8          03          00   ED 003EA              CMPZV   #0, #3, WORST_ERROR, R0                 ;
                                                    08   18 003F0              BGEQ    34$                                     ;
                                    04   A8 00000000*   00   9E 003F2              MOVAB   <SET$_NOHIDDEN!268435456>, WORST_ERROR  ;
                                                  03E5   31   003FA 34$:          BRW     79$                                     ; 0477
                                         0884  C8   9F 003FD 35$:          PUSHAB  ACE_POINTER                             ; 0479
                                    04   AE      0684   C8   9A 00401              MOVZBL  ACE, 4(SP)                              ;
                                    04   AE           C8   00 00407              ADDL2   #8, 4(SP)                               ;
                                               04   AE   9F 0040B              PUSHAB  4(SP)                                   ;
                              00000000G   00          02   FB 0040E              CALLS   #2, LIB$GET_VM                          ;
                                    56          50   D0 00415              MOVL    R0, VM_STATUS                           ;
                                    10          56   E9 00418              BLBC    VM_STATUS, 36$                          ;
                                    50      0684   C8   9A 0041B              MOVZBL  ACE, R0                                 ;
```

```
                                50             08  C0 00420          ADDL2   #8, R0                          :
  50            00              6E             00  2C 00423          MOVC5   #0, (SP), #0, R0, @ACE_POINTER  :
                                      0884     D8     00428                                                  :
                                57             56  D0 0042B  36$:    MOVL    VM_STATUS, STATUS               :
                                03             57  E8 0042E          BLBS    STATUS, 37$                     : 0480
                                      0323     31     00431          BRW     72$                             :
                                50    0684     C8  9A 00434  37$:    MOVZBL  ACE, R0                         : 0486
                                56    0884     C8  D0 00439          MOVL    ACE_POINTER, R6                 :
          08    A6      0684    C8             50  28 0043E          MOVC3   R0, ACE, 8(R6)                  :
                04              68             01  E0 00445          BBS     #1, FLAGS, 38$                  : 0487
                07              68             04  E1 00449          BBC     #4, FLAGS, 39$                  :
                                50    1494     C8  D0 0044D  38$:    MOVL    OLD_ACE_HEAD+4, R0              : 0488
                                               05  11 00452          BRB     40$                             :
                                50    149C     C8  D0 00454  39$:    MOVL    NEW_ACE_HEAD+4, R0              : 0489
                                60             66  0E 00459  40$:    INSQUE  (R6), (R0)                      : 0487
                                      FF10     31     0045C          BRW     29$                             : 0458
                                      0670     C8  9F 0045F  41$:    PUSHAB  CLI_ACE_DESC                    : 0494
                07              68             04  E1 00463          BBC     #4, FLAGS, 42$                  :
                                50    0298     CA  9E 00467          MOVAB   P.ACC, R0                       : 0495
                                               05  11 0046C          BRB     43$                             :
                                50    02A8     CA  9E 0046E  42$:    MOVAB   P.ACE, R0                       : 0496
                                               50  DD 00473  43$:    PUSHL   R0                              :
                      00000000G 00             02  FB 00475          CALLS   #2, CLI$GET_VALUE               : 0494
                                03             50  E8 0047C          BLBS    R0, 44$                         :
                                      009A     31     0047F          BRW     53$                             :
                         067C   C8    0200     8F  B0 00482  44$:    MOVW    #512, ACE_DESC                  : 0499
                                               7E  D4 00489          CLRL    -(SP)                           : 0502
                                      0678     C8  9F 0048B          PUSHAB  ERROR_POS                       :
                                      067C     C8  9F 0048F          PUSHAB  ACE_DESC                        :
                                      0670     C8  9F 00493          PUSHAB  CLI_ACE_DESC                    :
                      00000000G 00             04  FB 00497          CALLS   #4, SYS$PARSE_ACL               :
                                57             50  D0 0049E          MOVL    R0, STATUS                      :
                                11             57  E8 004A1          BLBS    STATUS, 47$                     : 0503
                                      FEFF     31     004A4          BRW     31$                             : 0506
  04      04    A8              03             00  ED 004A7  45$:    CMPZV   #0, #3, WORST_ERROR, #4         : 0508
                                               03  18 004AD          BGEQ    46$                             :
                                      00AD     31     004AF          BRW     57$                             :
                                      032D     31     004B2  46$:    BRW     79$                             :
          03            0687    C8             02  E1 004B5  47$:    BBC     #2, ACE+3, 48$                  : 0511
                                      FF12     31     004BB          BRW     33$                             :
                                      0884     C8  9F 004BE  48$:    PUSHAB  ACE_POINTER                     : 0517
                           04   AE    0684     C8  9A 004C2          MOVZBL  ACE, 4(SP)                      :
                           04   AE             08  C0 004C8          ADDL2   #8, 4(SP)                       :
                                        04     AE  9F 004CC          PUSHAB  4(SP)                           :
                      00000000G 00             02  FB 004CF          CALLS   #2, LIB$GET_VM                  :
                                56             50  D0 004D6          MOVL    R0, VM_STATUS                   :
                                10             56  E9 004D9          BLBC    VM_STATUS, 49$                  :
                                50    0684     C8  9A 004DC          MOVZBL  ACE, R0                         :
                                50             08  C0 004E1          ADDL2   #8, R0                          :
  50            00              6E             00  2C 004E4          MOVC5   #0, (SP), #0, R0, @ACE_POINTER  :
                                      0884     D8     004E9                                                  :
                                57             56  D0 004EC  49$:    MOVL    VM_STATUS, STATUS               :
                                03             57  E8 004EF          BLBS    STATUS, 50$                     : 0518
                                      0262     31     004F2          BRW     72$                             :
                                50    0684     C8  9A 004F5  50$:    MOVZBL  ACE, R0                         : 0524
                                56    0884     C8  D0 004FA          MOVL    ACE_POINTER, R6                 :
          08    A6      0684    C8             50  28 004FF          MOVC3   R0, ACE, 8(R6)                  :
```

```
             07        68          04  E1 00506          BBC     #4, FLAGS, 51$                         ; 0525
                       50    149C  C8  D0 0050A          MOVL    NEW_ACE_HEAD+4, R0                     ; 0526
                                   05  11 0050F          BRB     52$                                    ;
                       50    1494  C8  D0 00511  51$:    MOVL    OLD_ACE_HEAD+4, R0                     ; 0527
                       60          66  0E 00516  52$:    INSQUE  (R6), (R0)                             ; 0525
                                 FF43  31 00519          BRW     41$                                    ; 0494
                       50    1490  C8  9E 0051C  53$:    MOVAB   OLD_ACE_HEAD, R0                       ; 0532
                       50    1490  C8  D1 00521          CMPL    OLD_ACE_HEAD, R0                       ;
                                   2D  12 00526          BNEQ    56$                                    ;
                       50    1498  C8  9E 00528          MOVAB   NEW_ACE_HEAD, R0                       ; 0533
                       50    1498  C8  D1 0052D          CMPL    NEW_ACE_HEAD, R0                       ;
                                   21  12 00532          BNEQ    56$                                    ;
                       0C          68  E8 00534          BLBS    FLAGS, 54$                             ; 0536
             08        68          04  E0 00537          BBS     #4, FLAGS, 54$                         ;
             2B        68          05  E1 0053B          BBC     #5, FLAGS, 58$                         ; 0537
             27        68          02  E0 0053F          BBS     #2, FLAGS, 58$                         ;
                             02BC  CA  9F 00543  54$:    PUSHAB  P.ACG                                  ; 0540
                                   01  DD 00547  55$:    PUSHL   #1                                     ;
                         007710FC  8F  DD 00549          PUSHL   #7803132                               ;
                       6B          03  FB 0054F          CALLS   #3, LIB$SIGNAL                         ;
                                 FF52  31 00552          BRW     45$                                    ;
             11        68          02  E1 00555  56$:    BBC     #2, FLAGS, 58$                         ; 0546
                             02D0  CA  9F 00559          PUSHAB  P.ACI                                  ; 0549
                                   E8  11 0055D          BRB     55$                                    ;
                  04   A8 107710FC  8F  D0 0055F  57$:    MOVL    #276238588, WORST_ERROR                ;
                                 0278  31 00567          BRW     79$                                    ; 0550
                  01         0C    A8  D1 0056A  58$:    CMPL    OBJECT_TYPE, #1                        ; 0557
                                   03  13 0056E          BEQL    59$                                    ;
                                 0173  31 00570          BRW     70$                                    ;
 0050   8F       00        6E          00  2C 00573  59$:    MOVC5   #0, (SP), #0, #80, $RMS_PTR            ; 0564
                                   18  A8    0057A                                                      ;
                  18   A8    5003  8F  B0 0057C          MOVW    #20483, $RMS_PTR                       ;
                  1C   A8 00020000  8F  D0 00582          MOVL    #131072, $RMS_PTR+4                    ;
                  2E   A8    4203  8F  B0 0058A          MOVW    #16899, $RMS_PTR+22                    ;
                  37   A8          02  90 00590          MOVB    #2, $RMS_PTR+31                        ;
                  40   A8      68  A8  9E 00594          MOVAB   OBJECT_NAM, $RMS_PTR+40                ;
 0060   8F       00        6E          00  2C 00599          MOVC5   #0, (SP), #0, #96, $RMS_PTR            ; 0569
                                   68  A8    005A0                                                      ;
                  68   A8    6002  8F  B0 005A2          MOVW    #24578, $RMS_PTR                       ;
                  6A   A8          01  8E 005A8          MNEGB   #1, $RMS_PTR+2                         ;
                  6C   A8    01C8  C8  9E 005AC          MOVAB   OBJECT_RES_NAME, $RMS_PTR+4            ;
                  72   A8          01  8E 005B2          MNEGB   #1, $RMS_PTR+10                        ;
                  74   A8    00C8  C8  9E 005B6          MOVAB   OBJECT_EXP_NAME, $RMS_PTR+12           ;
                             0648  C8  9F 005BC          PUSHAB  COMMON_CTX                             ; 0574
                  04   AE    011F  8F  3C 005C0          MOVZWL  #287, 4(SP)                            ; 0578
                               04  AE  9F 005C6          PUSHAB  4(SP)                                  ; 0574
           00000000G   00          02  FB 005C9          CALLS   #2, LIB$QUAL_FILE_PARSE                ;
                       57          50  D0 005D0          MOVL    R0, STATUS                             ;
                       03          57  E8 005D3          BLBS    STATUS, 60$                            ; 0580
                                 017E  31 005D6          BRW     72$                                    ;
                  01   A8          08  8A 005D9  60$:    BICB2   #8, FLAGS+1                            ; 0590
                               18  A8  9F 005DD  61$:    PUSHAB  OBJECT_FAB                             ; 0591
               0000V   CF          01  FB 005E0          CALLS   #1, GET_FILE                           ;
                       03          50  E8 005E5          BLBS    R0, 62$                                ;
                                 01F7  31 005E8          BRW     79$                                    ;
             03        68          06  E0 005EB  62$:    BBS     #6, FLAGS, 64$                         ; 0598
                                 00DC  31 005EF  63$:    BRW     69$                                    ;
```

```
                                05F0    C8  D5 005F2 64$:     TSTL    SCHAN                                   ;
                                        F7  13 005F6         BEQL    63$                                     ;
                                        7E  7C 005F8         CLRQ    -(SP)                                   ; 0603
                                        7E  7C 005FA         CLRQ    -(SP)                                   ;
                                        7E  7C 005FC         CLRQ    -(SP)                                   ;
                                        7E  7C 005FE         CLRQ    -(SP)                                   ;
                                  28    AE  9F 00600         PUSHAB  IO_STATUS                               ;
                                        34  DD 00603         PUSHL   #52                                     ;
                                05F0    C8  DD 00605         PUSHL   SCHAN                                   ;
                                        7E  D4 00609         CLRL    -(SP)                                   ;
                00000000G   00          0C  FB 0060B         CALLS   #12, SYS$QIOW                           ;
                            57          50  D0 00612         MOVL    R0, STATUS                              ;
                            07          57  E9 00615         BLBC    STATUS, 65$                             ; 0604
                            57    08    AE  3C 00618         MOVZWL  IO_STATUS, STATUS                       ;
                            23          57  E8 0061C         BLBS    STATUS, 66$                             ; 0605
                                        7E  D4 0061F 65$:    CLRL    -(SP)                                   ;
                                        57  DD 00621         PUSHL   STATUS                                  ;
                                0338    C8  9F 00623         PUSHAB  SOBJECT_DESC                            ;
                                        01  DD 00627         PUSHL   #1                                      ;
                            00771052    8F  DD 00629         PUSHL   #7802962                                ;
                            6B          05  FB 0062F         CALLS   #5, LIB$SIGNAL                          ;
02      04  A8                   03      00  ED 00632         CMPZV   #0, #3, WORST_ERROR, #2                 ;
                                        08  18 00638         BGEQ    66$                                     ;
                          04    A8 10771052 8F  D0 0063A         MOVL    #276238418, WORST_ERROR         ;
                                05F0    C8  DD 00642 66$:    PUSHL   SCHAN                                   ; 0606
                00000000G   00          01  FB 00646         CALLS   #1, SYS$DASSGN                          ;
                            57          50  D0 0064D         MOVL    R0, STATUS                              ;
                            23          57  E8 00650         BLBS    STATUS, 67$                             ; 0607
                                        7E  D4 00653         CLRL    -(SP)                                   ;
                                        57  DD 00655         PUSHL   STATUS                                  ;
                                0338    C8  9F 00657         PUSHAB  SOBJECT_DESC                            ;
                                        01  DD 0065B         PUSHL   #1                                      ;
                            00771052    8F  DD 0065D         PUSHL   #7802962                                ;
                            6B          05  FB 00663         CALLS   #5, LIB$SIGNAL                          ;
02      04  A8                   03      00  ED 00666         CMPZV   #0, #3, WORST_ERROR, #2                 ;
                                        08  18 0066C         BGEQ    67$                                     ;
                          04    A8 10771052 8F  D0 0066E         MOVL    #276238418, WORST_ERROR         ;
                        064C    C8 000C0004 8F  D0 00676 67$:    MOVL    #786436, ATR_ARGLIST            ; 0612
                        0650    C8      0330 C8  9E 0067F        MOVAB   SACL_LOCKID, ATR_ARGLIST+4      ; 0613
                                        7E  7C 00686         CLRQ    -(SP)                                   ; 0617
                                        7E  D4 00688         CLRL    -(SP)                                   ;
                                064C    C8  9F 0068A         PUSHAB  ATR_ARGLIST                             ;
                                0338    C8  9F 0068E         PUSHAB  SOBJECT_DESC                            ;
                                0334    C8  9F 00692         PUSHAB  SOBJECT_TYPE                            ;
                                05F0    C8  DD 00696         PUSHL   SCHAN                                   ;
                00000000G   00          07  FB 0069A         CALLS   #7, SYS$CHANGE_ACL                      ;
                            57          50  D0 006A1         MOVL    R0, STATUS                              ;
                            23          57  E8 006A4         BLBS    STATUS, 68$                             ; 0618
                                        7E  D4 006A7         CLRL    -(SP)                                   ;
                                        57  DD 006A9         PUSHL   STATUS                                  ;
                                0338    C8  9F 006AB         PUSHAB  SOBJECT_DESC                            ;
                                        01  DD 006AF         PUSHL   #1                                      ;
                            00771052    8F  DD 006B1         PUSHL   #7802962                                ;
                            6B          05  FB 006B7         CALLS   #5, LIB$SIGNAL                          ;
02      04  A8                   03      00  ED 006BA         CMPZV   #0, #3, WORST_ERROR, #2                 ;
                                        08  18 006C0         BGEQ    68$                                     ;
                          04    A8 10771052 8F  D0 006C2         MOVL    #276238418, WORST_ERROR         ;
```

```
                                05F0   C8  D4  006CA 68$:    CLRL    SCHAN                                   ; 0619
                                  04   AE  9F  006CE 69$:    PUSHAB  SCAN_CONTEXT                            ; 0621
                              0000V    CF  9F  006D1         PUSHAB  INPUT_ERROR                             ;
                              0000V    CF  9F  006D5         PUSHAB  PROCESS_FILE                            ;
                                  18   A8  9F  006D9         PUSHAB  OBJECT_FAB                              ;
                00000000G     00           04  FB  006DC    CALLS   #4, LIB$FILE_SCAN                       ;
                                       FEF7   31  006E3    BRW     61$                                     ; 0591
                                  10   A8  9F  006E6 70$:    PUSHAB  OBJECT_NAME                             ; 0632
                              02E0     CA  9F  006E9         PUSHAB  P.ACK                                   ;
                00000000G     00           02  FB  006ED    CALLS   #2, CLI$GET_VALUE                       ;
                    064C   C8  000B0004     8F  D0  006F4    MOVL    #720900, ATR_ARGLIST                    ; 0637
                    0650   C8          08   A8  9E  006FD    MOVAB   ACL_LOCKID, ATR_ARGLIST+4               ; 0638
                                            7E  7C  00703    CLRQ    -(SP)                                   ; 0642
                                            7E  D4  00705    CLRL    -(SP)                                   ;
                                    064C    C8  9F  00707    PUSHAB  ATR_ARGLIST                             ;
                                      10    A8  9F  0070B    PUSHAB  OBJECT_NAME                             ;
                                      0C    A8  9F  0070E    PUSHAB  OBJECT_TYPE                             ;
                                    0328    C8  DD  00711    PUSHL   CHAN                                    ;
                00000000G     00           07  FB  00715    CALLS   #7, SYS$CHANGE_ACL                      ;
                              57           50  D0  0071C    MOVL    R0, STATUS                              ;
                              55           57  E8  0071F    BLBS    STATUS, 74$                             ; 0643
                000009B8      8F           57  D1  00722 71$:    CMPL    STATUS, #2488                           ; 0646
                                           2C  12  00729    BNEQ    72$                                     ;
                                 00000000G 00  9F  0072B    PUSHAB  SET$_OBJLOCKED                          ; 0647
                                    6B     01  FB  00731    CALLS   #1, LIB$SIGNAL                          ;
                                 50 00000000G 00  9E  00734    MOVAB   SET$_OBJLOCKED, R0                      ;
                                    37     50  E8  0073B    BLBS    R0, 73$                                 ;
                                 50 00000000* 00  9E  0073E    MOVAB   <SET$_OBJLOCKED&7>, R0                  ;
        50      04   A8                03     00  ED  00745    CMPZV   #0, #3, WORST_ERROR, R0                 ;
                                           28  18  0074B    BGEQ    73$                                     ;
                                    04   A8 00000000* 00  9E  0074D    MOVAB   <SET$_OBJLOCKED!268435456>, WORST_ERROR ;
                                           1E  11  00755    BRB     73$                                     ; 0646
                                           57  DD  00757 72$:    PUSHL   STATUS                              ; 0648
                                    6B     01  FB  00759    CALLS   #1, LIB$SIGNAL                          ;
                                    16     57  E8  0075C    BLBS    STATUS, 73$                             ;
        50          57                 03     00  EF  0075F    EXTZV   #0, #3, STATUS, R0                      ;
        50      04  A8                 03     00  ED  00764    CMPZV   #0, #3, WORST_ERROR, R0                 ;
                                           76  18  0076A    BGEQ    79$                                     ;
                    04   A8            57 10000000  8F  C9  0076C    BISL3   #268435456, STATUS, WORST_ERROR  ;
                                           6B  11  00775 73$:    BRB     79$                                     ; 0649
                    0A                 68     02  E1  00777 74$:    BBC     #2, FLAGS, 75$                          ; 0654
                                    10   A8  9F  0077B         PUSHAB  OBJECT_NAME                             ;
                              0000V    CF     01  FB  0077E    CALLS   #1, COPY_ACL                            ;
                                           24  11  00783    BRB     78$                                     ;
                    0A                 68     01  E1  00785 75$:    BBC     #1, FLAGS, 76$                          ; 0655
                                    10   A8  9F  00789         PUSHAB  OBJECT_NAME                             ;
                              0000V    CF     01  FB  0078C    CALLS   #1, DELETE_ACL                          ;
                                           16  11  00791    BRB     78$                                     ;
                    0A                 68     04  E1  00793 76$:    BBC     #4, FLAGS, 77$                          ; 0656
                                    10   A8  9F  00797         PUSHAB  OBJECT_NAME                             ;
                              0000V    CF     01  FB  0079A    CALLS   #1, REPLACE_ACL                         ;
                                           08  11  0079F    BRB     78$                                     ;
                                    10   A8  9F  007A1 77$:    PUSHAB  OBJECT_NAME                             ; 0657
                              0000V    CF     01  FB  007A4    CALLS   #1, ADD_ACL                             ;
                                    57     50  D0  007A9 78$:    MOVL    R0, STATUS                              ;
                    32              68     03  E1  007AC    BBC     #3, FLAGS, 79$                          ; 0661
                                    2F     57  E9  007B0    BLBC    STATUS, 79$                             ;
```

```
                                    10   A8  9F 007B3          PUSHAB  OBJECT_NAME                                          ; 0662
                                         01  DD 007B6          PUSHL   #1                                                  ;
                              00000000G  00  9F 007B8          PUSHAB  SET$_MODIFIED                                       ;
                                    6B   03  FB 007BE          CALLS   #3, LIB$SIGNAL                                      ;
                                    50 00000000G 00  9E 007C1  MOVAB   SET$_MODIFIED, R0                                   ;
                                    17   50  E8 007C8          BLBS    R0, 79$                                             ;
                                    50 00000000* 00  9E 007CB  MOVAB   <SET$_MODIFIED&7>, R0                               ;
          50        04   A8          03   00  ED 007D2          CMPZV   #0, #3, WORST_ERROR, R0                             ;
                                         08  18 007D8          BGEQ    79$                                                 ;
                              04   A8 00000000* 00  9E 007DA    MOVAB   <SET$_MODIFIED!268435456>, WORST_ERROR              ;
                                    50   04  A8  D0 007E2 79$:  MOVL    WORST_ERROR, R0                                     ; 0665
                                              04 007E6          RET                                                         ; 0667

; Routine Size:  2023 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  671     0668  1 ROUTINE GET_FILE =
;  672     0669  1
;  673     0670  1 !++
;  674     0671  1 !
;  675     0672  1 ! FUNCTIONAL DESCRIPTION:
;  676     0673  1 !
;  677     0674  1 !       This routine gets the next file specification in the command line.
;  678     0675  1 !       If there are no more specifications, the routine returns zero.
;  679     0676  1 !       Otherwise, the next file specification is placed in the specified
;  680     0677  1 !       FAB for later searching and parsing.
;  681     0678  1 !
;  682     0679  1 ! CALLING SEQUENCE:
;  683     0680  1 !
;  684     0681  1 !       GET_FILE
;  685     0682  1 !
;  686     0683  1 ! INPUT PARAMETERS:
;  687     0684  1 !       none
;  688     0685  1 !
;  689     0686  1 ! INPLICIT INPUTS:
;  690     0687  1 !       none
;  691     0688  1 !
;  692     0689  1 ! OUTPUT PARAMETERS:
;  693     0690  1 !       none
;  694     0691  1 !
;  695     0692  1 ! IMPLICIT OUTPUTS:
;  696     0693  1 !       none
;  697     0694  1 !
;  698     0695  1 ! ROUTINE VALUE:
;  699     0696  1 !       1 if a specification was found
;  700     0697  1 !       0 otherwise
;  701     0698  1 !
;  702     0699  1 ! SIDE EFFECTS:
;  703     0700  1 !       The retrieved file specification is placed into the specified FAB.
;  704     0701  1 !
;  705     0702  1 !--
;  706     0703  1
;  707     0704  2 BEGIN
;  708     0705  2
;  709     0706  2 OWN
;  710     0707  2         FILE_DESC       : $BBLOCK [DSC$C_S_BLN] ! File name descr
;  711     0708  2                           INITIAL (REP DSC$C_S_BLN OF (BYTE (0)));
;  712     0709  2
;  713     0710  2 LOCAL
;  714     0711  2         DESC            : $BBLOCK [DSC$C_S_BLN],        ! Temp descriptor
;  715     0712  2         ENDCHAR         : BYTE,                         ! Dir spec terminator
;  716     0713  2         EOS,                                            ! End addr of dir spec
;  717     0714  2         PTR,                                            ! Moving pointer in dir spec
;  718     0715  2         STR_PTR,                                        ! Pointer to remainder of spec
;  719     0716  2         STR_LEN,                                        ! Remaining length of dir spec
;  720     0717  2         TEMP_STRING     : VECTOR [NAM$C_MAXRSS, BYTE],  ! Temp dir spec storage
;  721     0718  2         TEMP,                                           ! Location of string to find
;  722     0719  2         STATUS;                                         ! Local routine exit status
;  723     0720  2
;  724     0721  2 ! Determine whether or not it is necessary to get another input specification.
;  725     0722  2
;  726     0723  3 IF NOT .FLAGS[SET_DIR_CMD] OR (.FLAGS[SET_DIR_CMD] AND NOT .FLAGS[IN_ELLIPSE])
;  727     0724  2 THEN
```

```
;  728          0725  3      BEGIN
;  729          0726  3
;  730          0727  3  ! If there are no more specifications, return 0.
;  731          0728  3
;  732          0729  3      FILE_DESC[DSC$B_CLASS] = DSC$K_CLASS_D;
;  733          0730  3      IF NOT CLI$GET_VALUE ($DESCRIPTOR ('INPUT'), FILE_DESC) THEN RETURN 0;
;  734          0731  3
;  735          0732  3  ! Fill in the FAB fields for the normal (or simple) case.
;  736          0733  3
;  737          0734  3      OBJECT_FAB[FAB$L_FNA] = .FILE_DESC[DSC$A_POINTER];
;  738          0735  3      OBJECT_FAB[FAB$B_FNS] = .FILE_DESC[DSC$W_LENGTH];
;  739          0736  2      END;
;  740          0737  2
;  741          0738  2  ! If this is a SET DIRECTORY command, it is necessary to do some additional
;  742          0739  2  ! processing of the input file specification.  In other words, it will be
;  743          0740  2  ! necessary to turn the directory specification into a file specification.
;  744          0741  2
;  745          0742  2  IF .FLAGS[SET_DIR_CMD]
;  746          0743  2  THEN
;  747          0744  3      BEGIN
;  748          0745  3
;  749          0746  3  ! Check here to see if a trailing ellipse is being treated.  If so,
;  750          0747  3  ! then FLAGS[IN_ELLIPSE] will be set to 1, and there's no need
;  751          0748  3  ! to search and see if such a trailing ellipse is present.  However,
;  752          0749  3  ! if the value is set to 0, then get a new directory spec.
;  753          0750  3
;  754          0751  3      IF NOT .FLAGS[IN_ELLIPSE]                    ! If not processing an ellipse
;  755          0752  3      THEN                                         ! then get the next directory
;  756          0753  4          BEGIN
;  757          0754  4          OBJECT_FAB[FAB$L_FNA] = .FILE_DESC[DSC$A_POINTER];
;  758          0755  4          OBJECT_FAB[FAB$B_FNS] = .FILE_DESC[DSC$W_LENGTH];
;  759          0756  4
;  760          0757  4  ! Since this is a new entry, it must be checked for a trailing ellipse.
;  761          0758  4
;  762          0759  4          CH$MOVE (.FILE_DESC[DSC$W_LENGTH],       ! Move this many chars
;  763          0760  4                   .FILE_DESC[DSC$A_POINTER],      ! From the CLI area
;  764          0761  4                   TEMP_STRING);                   ! To the temp string
;  765          0762  4          STR_PTR = TEMP_STRING;                   ! Set up pointer
;  766          0763  4          STR_LEN = .FILE_DESC[DSC$W_LENGTH];      ! and length.
;  767          0764  4
;  768          0765  4  ! Look for ellipses.
;  769          0766  4
;  770          0767  4          WHILE NOT CH$FAIL (TEMP = CH$FIND_SUB (.STR_LEN, .STR_PTR,
;  771          0768  4                                                 3, UPLIT ('...')))
;  772          0769  4          DO
;  773          0770  5              BEGIN
;  774          0771  5              STR_PTR = .TEMP + 3;                 ! Update pointer
;  775          0772  5              STR_LEN = .STR_LEN - (.TEMP - .STR_PTR) - 3;
;  776          0773  4              END;
;  777          0774  4
;  778          0775  4  ! After the final ellipse, check to see if it is at the end of the
;  779          0776  4  ! directory specification.  If so, then change the context field of
;  780          0777  4  ! the fab, and insert an end bracket at the beginning of the ellipse.
;  781          0778  4
;  782          0779  5          IF (.STR_PTR EQL TEMP_STRING + .OBJECT_FAB[FAB$B_FNS] -1)
;  783          0780  4          THEN
;  784          0781  5              BEGIN
```

```
;  785          0782  5              FLAGS[IN_ELLIPSE] = 1;                  ! Show that there's a trailing ellipse
;  786          0783  5              CH$WCHAR (..STR_PTR, .STR_PTR - 3); ! Put the end bracket in place
;  787          0784  5              OBJECT_FAB[FAB$L_FNA] = TEMP_STRING;        ! Set up FAB fields
;  788          0785  5              OBJECT_FAB[FAB$B_FNS] = .STR_PTR - 3 - TEMP_STRING + 1;
;  789          0786  4              END;
;  790          0787  4          END
;  791          0788  4
;  792          0789  4  ! If here, then the trailing ellipse has been processed, and this is the
;  793          0790  4  ! second time thru.  Restore the original file name.
;  794          0791  4
;  795          0792  3      ELSE
;  796          0793  4          BEGIN
;  797          0794  4          OBJECT_FAB[FAB$L_FNA] = .FILE_DESC[DSC$A_POINTER];      ! Original filename
;  798          0795  4          OBJECT_FAB[FAB$B_FNS] = .FILE_DESC[DSC$W_LENGTH];       ! Original length
;  799          0796  4          FLAGS[IN_ELLIPSE] = 0;                                  ! Ellipse processed
;  800          0797  3          END;
;  801          0798  3
;  802          0799  3  ! Parse the input string
;  803          0800  3
;  804          0801  3      $NAM_INIT (NAM = RELATED_NAM);                          ! Re-init the RLF
;  805          0802  4      IF (.OBJECT_NAM[NAM$B_DEV] NEQ 0)                       ! If a device was
;  806          0803  3      THEN                                                    ! specified, then
;  807          0804  4          BEGIN
;  808          0805  4          OBJECT_FAB[FAB$L_DNA] = .OBJECT_NAM[NAM$L_DEV]; ! Make device sticky
;  809          0806  4          OBJECT_FAB[FAB$B_DNS] = .OBJECT_NAM[NAM$B_DEV];
;  810          0807  3          END;
;  811          0808  4      IF NOT (STATUS = $PARSE (FAB = OBJECT_FAB))
;  812          0809  3      THEN
;  813          0810  4          BEGIN
;  814          0811  4          DESC[DSC$W_LENGTH] = .OBJECT_FAB[FAB$B_FNS];
;  815          0812  4          DESC[DSC$A_POINTER] = .OBJECT_FAB[FAB$L_FNA];
;  816          0813  4          FILE_ERROR (SET$_SYNTAX, OBJECT_FAB, .STATUS, 0);
;  817          0814  3          END;
;  818          0815  3
;  819          0816  3  ! Check the parsed string for legality, i.e. nothing after the directory
;  820          0817  3
;  821          0818  4      IF (.OBJECT_NAM[NAM$B_NAME] NEQ 0 OR
;  822          0819  4          .OBJECT_NAM[NAM$B_TYPE] NEQ 1 OR
;  823          0820  4          .OBJECT_NAM[NAM$B_VER]  NEQ 1 )
;  824          0821  3      THEN
;  825          0822  4          BEGIN
;  826          0823  4          DESC[DSC$W_LENGTH] = .OBJECT_FAB[FAB$B_FNS];
;  827          0824  4          DESC[DSC$A_POINTER] = .OBJECT_FAB[FAB$L_FNA];
;  828          0825  4          FILE_ERROR (SET$_SYNTAX, OBJECT_FAB, SS$_BADIRECTORY, 0);
;  829          0826  3          END;
;  830          0827  3
;  831          0828  3  ! Determine what the directory terminator character was, and save it.
;  832          0829  3
;  833          0830  3      ENDCHAR = .(.OBJECT_NAM[NAM$L_DIR] + .OBJECT_NAM[NAM$B_DIR] - 1);
;  834          0831  3
;  835          0832  3  ! The directory string must now be analyzed and manipulated so that the
;  836          0833  3  ! final directory entry becomes a file.  First, initialize some pointers.
;  837          0834  3
;  838          0835  3      DESC[DSC$W_LENGTH] = .OBJECT_NAM[NAM$B_ESL] - 2;
;  839          0836  3      DESC[DSC$A_POINTER] = .OBJECT_NAM[NAM$L_ESA];
;  840          0837  3      STR_PTR = .DESC[DSC$A_POINTER];
;  841          0838  3      STR_LEN = .DESC[DSC$W_LENGTH];
```

```
;   842          0839  3     PTR = 0;
;   843          0840  3     EOS = .DESC[DSC$A_POINTER] + .DESC[DSC$W_LENGTH] -1;
;   844          0841  3
;   845          0842  3 ! Look for wildcard ellipses
;   846          0843  3
;   847          0844  3     WHILE NOT CH$FAIL (TEMP = CH$FIND_SUB (.STR_LEN, .STR_PTR,
;   848          0845  3                                            3, UPLIT ('...')))
;   849          0846  3     DO
;   850          0847  4         BEGIN
;   851          0848  4
;   852          0849  4 ! Make PTR point to the beginning of the "...", and advance the string
;   853          0850  4 ! pointer to the character just past the "..."
;   854          0851  4
;   855          0852  4         PTR = .TEMP;
;   856          0853  4         STR_LEN = .STR_LEN - (.TEMP - .STR_PTR) -3;
;   857          0854  4         STR_PTR = .TEMP + 3;
;   858          0855  4         END;
;   859          0856  3
;   860          0857  3 ! If there was any occurrence of "...", point just past it.
;   861          0858  3
;   862          0859  3     IF .PTR NEQ 0 THEN PTR = .PTR + 3;
;   863          0860  3
;   864          0861  3 ! Find the last directory in the specification
;   865          0862  3
;   866          0863  3     WHILE NOT CH$FAIL ( TEMP = CH$FIND_CH (.STR_LEN, .STR_PTR, '.'))
;   867          0864  3     DO
;   868          0865  4         BEGIN
;   869          0866  4
;   870          0867  4 ! Make PTR point to the ".", and advance the string pointer to
;   871          0868  4 ! the first character after the "."
;   872          0869  4
;   873          0870  4         PTR = .TEMP;
;   874          0871  4         STR_LEN = .STR_LEN - (.TEMP - .STR_PTR) - 1;
;   875          0872  4         STR_PTR = .TEMP + 1;
;   876          0873  3         END;
;   877          0874  3
;   878          0875  3     IF .PTR NEQ 0
;   879          0876  3     THEN
;   880          0877  4         BEGIN
;   881          0878  4
;   882          0879  4 ! If here, then either a trailing ellipse, or a final sub-directory
;   883          0880  4 ! was specified.  If the pointer is at the bracket, then there is a
;   884          0881  4 ! trailing ellipse, in which case only a "*" is required.
;   885          0882  4
;   886          0883  4         IF .PTR EQL .EOS
;   887          0884  4         THEN
;   888          0885  5             BEGIN
;   889          0886  5             CH$A_WCHAR ('*', PTR);              ! Stick an asterisk after the bracket.
;   890          0887  5             PTR = .PTR + 1;                     ! Adjust the pointer.
;   891          0888  5             END
;   892          0889  5
;   893          0890  5 ! If the pointer is inside the bracket, then the last directory name
;   894          0891  5 ! must be moved out of the brackets.
;   895          0892  5
;   896          0893  4         ELSE
;   897          0894  5             BEGIN
;   898          0895  5
```

```
;  899          0896  5 ! Check to see if the directory is [main.sub] or [main...sub]
;  900          0897  5
;  901          0898  5             IF .PTR EQLU .STR_PTR
;  902          0899  5             THEN
;  903          0900  6                 BEGIN                                   ! [main...sub] form
;  904          0901  6                 STR_LEN = .EOS - .PTR;
;  905          0902  6                 CH$MOVE (.STR_LEN, .PTR, .PTR+1);
;  906          0903  6                 CH$WCHAR (.ENDCHAR, .PTR);
;  907          0904  6                 PTR = .PTR + .STR_LEN + 1;
;  908          0905  6                 END                                     ! end of [main...sub] processing
;  909          0906  5             ELSE
;  910          0907  6                 BEGIN                                   ! [main.sub] form
;  911          0908  6                 STR_LEN = .EOS - .STR_PTR;
;  912          0909  6                 CH$WCHAR_A (.ENDCHAR, PTR);
;  913          0910  6                 PTR = .PTR + .STR_LEN;
;  914          0911  5                 END;                                    ! end of [main.sub] processing
;  915          0912  4             END;
;  916          0913  4         END                                             ! End of non-zero pointer stuff
;  917          0914  3     ELSE
;  918          0915  4         BEGIN
;  919          0916  4
;  920          0917  4 ! If the pointer is still zero, then there is either a wildcard, a main
;  921          0918  4 ! directory, or a [g,m] directory.  In all such cases, a main directory
;  922          0919  4 ! of [000000] must be fabricated.
;  923          0920  4
;  924          0921  4         STATUS = CH$FIND_CH (.STR_LEN, .STR_PTR, ',');  ! Save for later
;  925          0922  4
;  926          0923  4 ! Move the string out seven spaces and insert "000000]"
;  927          0924  4
;  928          0925  4         STR_PTR = .DESC[DSC$A_POINTER] + .OBJECT_NAM[NAM$B_DEV] + 1;
;  929          0926  4         TEMP = CH$MOVE (.EOS - .STR_PTR, .STR_PTR, .STR_PTR + 7);
;  930          0927  4         STR_PTR = CH$MOVE (6, UPLIT ('000000'), .STR_PTR);
;  931          0928  4         CH$MOVE (1, ENDCHAR, .STR_PTR);
;  932          0929  4
;  933          0930  4 ! If no comma was found, then all that is required is to update the
;  934          0931  4 ! pointer.
;  935          0932  4
;  936          0933  4         IF CH$FAIL (.STATUS) THEN PTR = .TEMP
;  937          0934  4
;  938          0935  4 ! Otherwise, it's a [g,m] directory.  Convert it.
;  939          0936  4
;  940          0937  4         ELSE
;  941          0938  5             BEGIN
;  942          0939  5
;  943          0940  5             LOCAL TPARSE_BLOCK : $BBLOCK[TPA$K_LENGTH0];        ! Define a TPARSE block
;  944          0941  5
;  945          0942  5             CH$FILL (0, TPA$K_LENGTH0, TPARSE_BLOCK);   ! Zero it.
;  946          0943  5             TPARSE_BLOCK[TPA$L_COUNT] = TPA$K_COUNT0;   ! Fill in size
;  947          0944  5
;  948          0945  5             TPARSE_BLOCK[TPA$L_STRINGCNT] = .EOS - .STR_PTR;
;  949          0946  5             TPARSE_BLOCK[TPA$L_STRINGPTR] = .STR_PTR + 7;
;  950          0947  6             IF NOT (STATUS = LIB$TPARSE (TPARSE_BLOCK,
;  951          0948  6                                          DIR_STATE,
;  952          0949  6                                          DIR_KEYS))
;  953          0950  5             THEN FILE_ERROR (SET$_SYNTAX, OBJECT_FAB, .STATUS, 0)
;  954          0951  5             ELSE
;  955          0952  6                 BEGIN
```

```
;  956          0953  6                  LOCAL TEMP_DESC : $BBLOCK[DSC$C_S_BLN];
;  957          0954  6                  TEMP_DESC[DSC$W_LENGTH] = 6;
;  958          0955  6                  TEMP_DESC[DSC$A_POINTER] = .STR_PTR + 7;
;  959        P 0956  7                  IF NOT (STATUS = $FAO ($DESCRIPTOR('!2(3OW)'),
;  960        P 0957  7                                        TEMP_DESC,
;  961        P 0958  7                                        TEMP_DESC,
;  962        P 0959  7                                        .DIR_GROUP,
;  963          0960  7                                        .DIR_MEMBER))
;  964          0961  6                  THEN FILE_ERROR (SET$_SYNTAX, OBJECT_FAB, .STATUS, 0)
;  965          0962  6                  ELSE PTR = .STR_PTR + 14;
;  966          0963  5                  END;
;  967          0964  4              END;
;  968          0965  3          END;
;  969          0966  3      PTR = CH$MOVE (4, UPLIT ('.DIR'), .PTR);
;  970          0967  3      OBJECT_FAB[FAB$B_FNS] = .PTR - .DESC[DSC$A_POINTER];
;  971          0968  3      OBJECT_FAB[FAB$L_FNA] = .DESC[DSC$A_POINTER];
;  972          0969  2      END;
;  973          0970  2
;  974          0971  2  RETURN 1;
;  975          0972  2
;  976          0973  1  END;                                   ! End of routine GET_FILE


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                  54  55  50  4E  49   002F0 P.ACN:  .ASCII  \INPUT\                    ;
                                                 002F5          .BLKB   3
                                      00000005   002F8 P.ACM:  .LONG   5                              ;
                                      00000000'  002FC          .ADDRESS P.ACN                       ;
                                  00  2E  2E  2E   00300 P.ACO:  .ASCII  \...\<0>                     ;
                                  00  2E  2E  2E   00304 P.ACP:  .ASCII  \...\<0>                     ;
                              00  00  30  30  30   00308 P.ACQ:  .ASCII  \000000\<0><0>               ;
                                  29  57  4F  33   00310 P.ACS:  .ASCII  \!2(3OW)\                    ;
                                                 00317          .BLKB   1
                                      00000007   00318 P.ACR:  .LONG   7                              ;
                                      00000000'  0031C          .ADDRESS P.ACS                       ;
                                  52  49  44  2E   00320 P.ACT:  .ASCII  \.DIR\                      ;

                                                        .PSECT  $OWN$,NOEXE,2

                                          00  014A8 FILE_DESC:
                                                        .BYTE   0                                    ;
                                          00  014A9             .BYTE   0                            ;
                                          00  014AA             .BYTE   0                            ;
                                          00  014AB             .BYTE   0                            ;
                                          00  014AC             .BYTE   0                            ;
                                          00  014AD             .BYTE   0                            ;
                                          00  014AE             .BYTE   0                            ;
                                          00  014AF             .BYTE   0                            ;

                                                  $RMS_PTR=             RELATED_NAM
                                                        .EXTRN  SYS$PARSE, SYS$FAO

                                                        .PSECT  $CODE$,NOWRT,2

                                      0FFC 00000 GET_FILE:
```

```
                                                        .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11         ; 0668
                      5E      FEC4    CE   9E 00002             MOVAB   -316(SP), SP
              06   0000'  CF           06   E1 00007            BBC     #6, FLAGS+1, 1$                           ; 0723
              28   0000'  CF           03   E0 0000D            BBS     #3, FLAGS+1, 3$
                   0000'  CF           02   90 00013 1$:        MOVB    #2, FILE_DESC+3                           ; 0729
                              0000'    CF   9F 00018            PUSHAB  FILE_DESC                                 ; 0730
                              0000'    CF   9F 0001C            PUSHAB  P.ACM
              00000000G   00           02   FB 00020            CALLS   #2, CLI$GET_VALUE
                          03           50   E8 00027            BLBS    R0, 2$
                                     0294   31 0002A            BRW     30$
                   0000'  CF  0000'    CF   D0 0002D 2$:        MOVL    FILE_DESC+4, OBJECT_FAB+44                ; 0734
                   0000'  CF  0000'    CF   90 00034            MOVB    FILE_DESC, OBJECT_FAB+52                  ; 0735
              03   0000'  CF           06   E0 0003B 3$:        BBS     #6, FLAGS+1, 4$                           ; 0742
                                     0279   31 00041            BRW     29$
                          50  0000'    CF   D0 00044 4$:        MOVL    FILE_DESC+4, R0                           ; 0754
                   0000'  CF           50   D0 00049            MOVL    R0, OBJECT_FAB+44
                          56  0000'    CF   3C 0004E            MOVZWL  FILE_DESC, R6                             ; 0755
              65   0000'  CF           03   E0 00053            BBS     #3, FLAGS+1, 8$                           ; 0751
                   0000'  CF           56   90 00059            MOVB    R6, OBJECT_FAB+52                         ; 0755
         34   AE   0000'  DF           56   28 0005E            MOVC3   R6, @FILE_DESC+4, TEMP_STRING             ; 0759
                          5A      34   AE   9E 00065            MOVAB   TEMP_STRING, STR_PTR                      ; 0762
                          57           56   D0 00069            MOVL    R6, STR_LEN                               ; 0763
    6A        57   0000'  CF           03   39 0006C 5$:        MATCHC  #3, P.ACO, STR_LEN, (STR_PTR)             ; 0768
                                       03   13 00073            BEQL    6$
                          53           03   D0 00075            MOVL    #3, R3
                                       53   D7 00078 6$:        DECL    R3
                          5B           73   3E 0007A            MOVAW   -(R3), TEMP
                                       0F   13 0007D            BEQL    7$
                          5A      03   AB   9E 0007F            MOVAB   3(R11), STR_PTR                           ; 0771
              53          5A           5B   C3 00083            SUBL3   TEMP, STR_PTR, R3                         ; 0772
                          57   FD A347      9E 00087            MOVAB   -3(R3)[STR_LEN], STR_LEN
                                       DE   11 0008C            BRB     5$                                        ; 0767
                          50  0000'    CF   9A 0008E 7$:        MOVZBL  OBJECT_FAB+52, R0                         ; 0779
                          50   33 AE40      9E 00093            MOVAB   TEMP_STRING-1[R0], R0
                          50           5A   D1 00098            CMPL    STR_PTR, R0
                                       2B   12 0009B            BNEQ    9$
                   0000'  CF           08   88 0009D            BISB2   #8, FLAGS+1                               ; 0782
                          51      FD   AA   9E 000A2            MOVAB   -3(STR_PTR), R1                           ; 0783
                          61           6A   90 000A6            MOVB    (STR_PTR), (R1)
                   0000'  CF      34   AE   9E 000A9            MOVAB   TEMP_STRING, OBJECT_FAB+44                ; 0784
                          50      34   AE   9E 000AF            MOVAB   TEMP_STRING, R0                           ; 0785
                          50           51   C2 000B3            SUBL2   R1, R0
         0000'  CF        01           50   83 000B6            SUBB3   R0, #1, OBJECT_FAB+52
                                       0A   11 000BC            BRB     9$                                        ; 0751
                   0000'  CF           56   90 000BE 8$:        MOVB    R6, OBJECT_FAB+52                         ; 0795
                   0000'  CF           08   8A 000C3            BICB2   #8, FLAGS+1                               ; 0796
0060     8F           00               6E   00 2C 000C8 9$:     MOVC5   #0, (SP), #0, #96, $RMS_PTR               ; 0801
                              0000'    CF      000CF
                   0000'  CF  6002     8F   B0 000D2            MOVW    #24578, $RMS_PTR
                          50  0000'    CF   9A 000D9            MOVZBL  OBJECT_NAM+57, R0                         ; 0802
                                       0C   13 000DE            BEQL    10$
                   0000'  CF  0000'    CF   D0 000E0            MOVL    OBJECT_NAM+68, OBJECT_FAB+48              ; 0805
                   0000'  CF           50   90 000E7            MOVB    R0, OBJECT_FAB+53                         ; 0806
                              0000'    CF   9F 000EC 10$:       PUSHAB  OBJECT_FAB                                ; 0808
              00000000G   00           01   FB 000F0            CALLS   #1, SYS$PARSE
                          6E           50   D0 000F7            MOVL    R0, STATUS
                          20           6E   E8 000FA            BLBS    STATUS, 11$
```

```
                F8   AD      0000'   CF   9B 000FD          MOVZBW  OBJECT_FAB+52, DESC                        ;0811
                FC   AD      0000'   CF   D0 00103          MOVL    OBJECT_FAB+44, DESC+4                      ;0812
                                     7E   D4 00109          CLRL    -(SP)                                      ;0813
                               04    AE   DD 0010B          PUSHL   STATUS                                     ;
                           0000'     CF   9F 0010E          PUSHAB  OBJECT_FAB                                 ;
                       007710FC      8F   DD 00112          PUSHL   #7803132                                   ;
            0000V      CF                 04   FB 00118          CALLS   #4, FILE_ERROR                     ;
                                0000'     CF   95 0011D 11$:     TSTB    OBJECT_NAM+59                      ;0818
                                     0E   12   00121          BNEQ    12$                                        ;
                          01        0000'     CF   91 00123          CMPB    OBJECT_NAM+60, #1                  ;0819
                                     07   12   00128          BNEQ    12$                                        ;
                          01        0000'     CF   91 0012A          CMPB    OBJECT_NAM+61, #1                  ;0820
                                     22   13   0012F          BEQL    13$                                        ;
                F8   AD      0000'   CF   9B 00131 12$:     MOVZBW  OBJECT_FAB+52, DESC                        ;0823
                FC   AD      0000'   CF   D0 00137          MOVL    OBJECT_FAB+44, DESC+4                      ;0824
                                     7E   D4 0013D          CLRL    -(SP)                                      ;0825
                          7E        0828      8F   3C 0013F          MOVZWL  #2088, -(SP)                       ;
                                0000'     CF   9F 00144          PUSHAB  OBJECT_FAB                         ;
                       007710FC      8F   DD 00148          PUSHL   #7803132                                   ;
            0000V      CF                 04   FB 0014E          CALLS   #4, FILE_ERROR                     ;
                          50        0000'     CF   9A 00153 13$:     MOVZBL  OBJECT_NAM+58, R0                  ;0830
                          50        0000'     CF   C0 00158          ADDL2   OBJECT_NAM+72, R0                  ;
                04    AE        FF        A0   90 0015D          MOVB    -1(R0), ENDCHAR                    ;
                F8   AD      0000'   CF   9B 00162          MOVZBW  OBJECT_NAM+11, DESC                        ;0835
                F8   AD                  02   A2 00168          SUBW2   #2, DESC                           ;
                FC   AD      0000'   CF   D0 0016C          MOVL    OBJECT_NAM+12, DESC+4                      ;0836
                          59        FC   AD   D0 00172          MOVL    DESC+4, R9                         ;0837
                          5A             59   D0 00176          MOVL    R9, STR_PTR                        ;
                          50        F8   AD   3C 00179          MOVZWL  DESC, R0                           ;0838
                          57             50   D0 0017D          MOVL    R0, STR_LEN                        ;
                                         58   D4 00180          CLRL    PTR                                ;0839
                          56        FF A049   9E 00182          MOVAB   -1(R0)[R9], EOS                    ;0840
      6A        57      0000'     CF        03   39 00187 14$:     MATCHC  #3, P.ACP, STR_LEN, (STR_PTR)      ;0845
                                         03   13   0018E          BEQL    15$                                ;
                          53             03   D0 00190          MOVL    #3, R3                             ;
                                         53   D7 00193 15$:     DECL    R3                                 ;
                          5B             73   3E 00195          MOVAW   -(R3), TEMP                        ;
                                         12   13   00198          BEQL    16$                                ;
                          58             5B   D0 0019A          MOVL    TEMP, PTR                          ;0852
                53        5A             5B   C3 0019D          SUBL3   TEMP, STR_PTR, R3                  ;0853
                          57        FD A347   9E 001A1          MOVAB   -3(R3)[STR_LEN], STR_LEN           ;
                          5A        03   AB   9E 001A6          MOVAB   3(R11), STR_PTR                    ;0854
                                         DB   11   001AA          BRB     14$                                ;0844
                                         58   D5 001AC 16$:     TSTL    PTR                                ;0859
                                         03   13   001AE          BEQL    17$                                ;
                          58             03   C0 001B0          ADDL2   #3, PTR                            ;
                6A        57             2E   3A 001B3 17$:     LOCC    #46, STR_LEN, (STR_PTR)            ;0863
                                         02   12   001B7          BNEQ    18$                                ;
                                         51   D4 001B9          CLRL    R1                                 ;
                          5B             51   D0 001BB 18$:     MOVL    R1, TEMP                           ;
                                         12   13   001BE          BEQL    19$                                ;
                          58             5B   D0 001C0          MOVL    TEMP, PTR                          ;0870
                53        5A             5B   C3 001C3          SUBL3   TEMP, STR_PTR, R3                  ;0871
                          57        FF A347   9E 001C7          MOVAB   -1(R3)[STR_LEN], STR_LEN           ;
                          5A        01   AB   9E 001CC          MOVAB   1(R11), STR_PTR                    ;0872
                                         E1   11   001D0          BRB     17$                                ;0863
                                         58   D5 001D2 19$:     TSTL    PTR                                ;0875
```

```
                                32  13 001D4          BEQL    22$                                  ;
                    56          58  D1 001D6          CMPL    PTR, EOS                             ; 0883
                                07  12 001D9          BNEQ    20$                                  ;
                                58  D6 001DB          INCL    PTR                                  ; 0886
                    88          2A  90 001DD          MOVB    #42, (PTR)+                          ;
                                5B  11 001E0          BRB     24$                                  ; 0883
                    5A          58  D1 001E2 20$:     CMPL    PTR, STR_PTR                         ; 0898
                                14  12 001E5          BNEQ    21$                                  ;
         57         56          58  C3 001E7          SUBL3   PTR, EOS, STR_LEN                    ; 0901
   01    A8         68          57  28 001EB          MOVC3   STR_LEN, (PTR), 1(PTR)               ; 0902
                    68      04  AE  90 001F0          MOVB    ENDCHAR, (PTR)                       ; 0903
                    58   01 A748    9E 001F4          MOVAB   1(STR_LEN)[PTR], PTR                 ; 0904
                                42  11 001F9          BRB     24$                                  ; 0898
         57         56          5A  C3 001FB 21$:     SUBL3   STR_PTR, EOS, STR_LEN                ; 0908
                    88      04  AE  90 001FF          MOVB    ENDCHAR, (PTR)+                      ; 0909
                    58          57  C0 00203          ADDL2   STR_LEN, PTR                         ; 0910
                                35  11 00206          BRB     24$                                  ; 0875
         6A         57          2C  3A 00208 22$:     LOCC    #44, STR_LEN, (STR_PTR)              ; 0921
                                02  12 0020C          BNEQ    23$                                  ;
                                51  D4 0020E          CLRL    R1                                   ;
                    6E          51  D0 00210 23$:     MOVL    R1, STATUS                           ;
                    50   0000'  CF  9A 00213          MOVZBL  OBJECT_NAM+57, R0                    ; 0925
                    5A   01 A049    9E 00218          MOVAB   1(R0)[R9], STR_PTR                   ;
         50         56          5A  C3 0021D          SUBL3   STR_PTR, EOS, R0                     ; 0926
   07    AA         6A          50  28 00221          MOVC3   R0, (STR_PTR), 7(STR_PTR)            ;
                    5B          53  D0 00226          MOVL    R3, TEMP                             ;
         6A   0000' CF          06  28 00229          MOVC3   #6, P.ACQ, (STR_PTR)                 ; 0927
                    5A          53  D0 0022F          MOVL    R3, STR_PTR                          ;
                    6A      04  AE  90 00232          MOVB    ENDCHAR, (STR_PTR)                   ; 0928
                                6E  D5 00236          TSTL    STATUS                               ; 0933
                                05  12 00238          BNEQ    25$                                  ;
                    58          5B  D0 0023A          MOVL    TEMP, PTR                            ;
                                6E  11 0023D 24$:     BRB     28$                                  ;
24       00         6E          00  2C 0023F 25$:     MOVC5   #0, (SP), #0, #36, TPARSE_BLOCK      ; 0942
                            10  AE     00244                                                       ;
              10    AE          08  D0 00246          MOVL    #8, TPARSE_BLOCK                     ; 0943
   18    AE         56          5A  C3 0024A          SUBL3   STR_PTR, EOS, TPARSE_BLOCK+8         ; 0945
                    53      07  AA  9E 0024F          MOVAB   7(R10), R3                           ; 0946
              1C    AE          53  D0 00253          MOVL    R3, TPARSE_BLOCK+12                  ;
                         0000'  CF  9F 00257          PUSHAB  DIR_KEYS                             ; 0947
                         0000'  CF  9F 0025B          PUSHAB  DIR_STATE                            ;
                            18  AE  9F 0025F          PUSHAB  TPARSE_BLOCK                         ;
       00000000G    00          03  FB 00262          CALLS   #3, LIB$TPARSE                       ;
                    6E          50  D0 00269          MOVL    R0, STATUS                           ;
                    24          6E  E9 0026C          BLBC    STATUS, 26$                          ;
              08    AE          06  B0 0026F          MOVW    #6, TEMP_DESC                        ; 0954
              0C    AE          53  D0 00273          MOVL    R3, TEMP_DESC+4                      ; 0955
                    7E   0000'  CF  7D 00277          MOVQ    DIR_GROUP, -(SP)                     ; 0960
                            10  AE  9F 0027C          PUSHAB  TEMP_DESC                            ;
                            14  AE  9F 0027F          PUSHAB  TEMP_DESC                            ;
                         0000'  CF  9F 00282          PUSHAB  P.ACR                                ;
       00000000G    00          05  FB 00286          CALLS   #5, SYS$FAO                          ;
                    6E          50  D0 0028D          MOVL    R0, STATUS                           ;
                    16          6E  E8 00290          BLBS    STATUS, 27$                          ;
                                7E  D4 00293 26$:     CLRL    -(SP)                                ; 0961
                            04  AE  DD 00295          PUSHL   STATUS                               ;
                         0000'  CF  9F 00298          PUSHAB  OBJECT_FAB                           ;
```

```
                          007710FC   8F   DD 0029C         PUSHL   #7803132                      ;
                  0000V   CF         04   FB 002A2         CALLS   #4, FILE_ERROR                ;
                                     04   11 002A7         BRB     28$                           ;
                          58    0E   AA   9E 002A9 27$:    MOVAB   14(R10), PTR                  ; 0962
                          88  0000'  CF   D0 002AD 28$:    MOVL    P.ACT, (PTR)+                 ; 0966
      0000'  CF           58         59   83 002B2         SUBB3   R9, PTR, OBJECT_FAB+52        ; 0967
                  0000'   CF         59   D0 002B8         MOVL    R9, OBJECT_FAB+44             ; 0968
                          50         01   D0 002BD 29$:    MOVL    #1, R0                        ; 0971
                                        04 002C0           RET                                   ;
                                     50   D4 002C1 30$:    CLRL    R0                            ; 0973
                                        04 002C3           RET                                   ;

; Routine Size:  708 bytes,    Routine Base:  $CODE$ + 07E7
```

```
;  978          0974  1 ROUTINE PROCESS_FILE =
;  979          0975  1
;  980          0976  1 !++
;  981          0977  1 !
;  982          0978  1 ! FUNCTIONAL DESCRIPTION:
;  983          0979  1 !
;  984          0980  1 !       This routine takes the spec from LIB$FILE_SCAN, and calls the
;  985          0981  1 !       appropriate routine based upon the command line qualifiers.
;  986          0982  1 !
;  987          0983  1 ! CALLING SEQUENCE:
;  988          0984  1 !       PROCESS_FILE
;  989          0985  1 !
;  990          0986  1 ! INPUT PARAMETERS:
;  991          0987  1 !       none
;  992          0988  1 !
;  993          0989  1 ! IMPLICIT INPUTS:
;  994          0990  1 !       none
;  995          0991  1 !
;  996          0992  1 ! OUTPUT PARAMETERS:
;  997          0993  1 !       none
;  998          0994  1 !
;  999          0995  1 ! IMPLICIT OUTPUTS:
; 1000          0996  1 !       none
; 1001          0997  1 !
; 1002          0998  1 ! ROUTINE VALUE:
; 1003          0999  1 !       1 if successful
; 1004          1000  1 !       error code otherwise
; 1005          1001  1 !
; 1006          1002  1 ! SIDE EFFECTS:
; 1007          1003  1 !       none
; 1008          1004  1 !
; 1009          1005  1 !--
; 1010          1006  1
; 1011          1007  2 BEGIN
; 1012          1008  2
; 1013          1009  2 LOCAL
; 1014          1010  2         FILE_NAME       : $BBLOCK [DSC$C_S_BLN],        ! File name to log
; 1015          1011  2         FAB             : $FAB_DECL,                    ! Storage for the FAB
; 1016          1012  2         NAM             : $NAM_DECL,                    ! Storage for the NAMe block
; 1017          1013  2         XABDAT          : $XABDAT_DECL,                 ! Date XAB storage
; 1018          1014  2         XABPRO          : $XABPRO_DECL,                 ! Protection XAB storage
; 1019          1015  2         FILE_CHAR       : $BBLOCK [4],                  ! Target file characteristics
; 1020          1016  2         IO_STATUS       : VECTOR [4, WORD],             ! I/O status block
; 1021          1017  2         STATUS,                                         ! Local routine return status
; 1022          1018  2         STATUS1;                                        ! Second local routine exit status
; 1023          1019  2
; 1024          1020  2 ! Open the the specified file.
; 1025          1021  2
; 1026          1022  2 CH$FILL (0, 3 * ITM$S_ITEM, ATR_ARGLIST);
; 1027          1023  2 CH$MOVE (NAM$C_BLN, .OBJECT_FAB[FAB$L_NAM], NAM);
; 1028        P 1024  2 $FAB_INIT (FAB = FAB,
; 1029        P 1025  2            FAC = <GET, PUT>,
; 1030        P 1026  2            FOP = <NAM, UFO>,
; 1031        P 1027  2            NAM = NAM,
; 1032        P 1028  2            SHR = NIL,
; 1033          1029  2            XAB = XABDAT);
; 1034        P 1030  2 $XABDAT_INIT (XAB = XABDAT,
```

```
;  1035     1031  2                    NXT = XABPRO);
;  1036     1032  2 $XABPRO_INIT (XAB = XABPRO);
;  1037     1033  2
;  1038     1034  2 STATUS = $OPEN (FAB = FAB);
;  1039     1035  2
;  1040     1036  2 ! Set up the actual file name.
;  1041     1037  2
;  1042     1038  2 CH$FILL (0, DSC$C_S_BLN, FILE_NAME);
;  1043     1039  2 IF .NAM[NAM$B_RSL] NEQ 0
;  1044     1040  2 THEN
;  1045     1041  3     BEGIN
;  1046     1042  3     FILE_NAME[DSC$W_LENGTH] = .NAM[NAM$B_RSL];
;  1047     1043  3     FILE_NAME[DSC$A_POINTER] = .NAM[NAM$L_RSA];
;  1048     1044  3     END
;  1049     1045  2 ELSE IF .NAM[NAM$B_ESL] NEQ 0
;  1050     1046  2 THEN
;  1051     1047  3     BEGIN
;  1052     1048  3     FILE_NAME[DSC$W_LENGTH] = .NAM[NAM$B_ESL];
;  1053     1049  3     FILE_NAME[DSC$A_POINTER] = .NAM[NAM$L_ESA];
;  1054     1050  3     END
;  1055     1051  2 ELSE
;  1056     1052  3     BEGIN
;  1057     1053  3     FILE_NAME[DSC$W_LENGTH] = .FAB[FAB$B_FNS];
;  1058     1054  3     FILE_NAME[DSC$A_POINTER] = .FAB[FAB$L_FNA];
;  1059     1055  2     END;
;  1060     1056  2
;  1061     1057  2 ! If there are any errors on the open, note them.
;  1062     1058  2
;  1063     1059  2 IF NOT .STATUS
;  1064     1060  2 THEN
;  1065     1061  3     BEGIN
;  1066     1062  3
;  1067     1063  3 ! If the error is a "file locked by another user" error and the file-id of the
;  1068     1064  3 ! source and tarket files match, simply ignore the error and go process the next
;  1069     1065  3 ! in line.  Otherwise, note the error.
;  1070     1066  3
;  1071     1067  3     IF .FAB[FAB$L_STS] NEQ RMS$_FLK
;  1072     1068  3     OR CH$NEQ (6, SOBJECT_NAM[NAM$W_FID], 6, OBJECT_NAM[NAM$W_FID], 0)
;  1073     1069  3     THEN FILE_ERROR (SET$_OPENIN, FAB, .FAB[FAB$L_STS], .FAB[FAB$L_STV]);
;  1074     1070  3     RETURN 1;
;  1075     1071  2     END;
;  1076     1072  2
;  1077     1073  2 CHAN = .FAB[FAB$L_STV];
;  1078     1074  2
;  1079     1075  2 ! See if the file matches the criteria specified by the common command
;  1080     1076  2 ! qualifiers.
;  1081     1077  2
;  1082     1078  2 IF NOT LIB$QUAL_FILE_MATCH (COMMON_CTX,
;  1083     1079  2                             FAB,
;  1084     1080  2                             0,
;  1085     1081  2                             $DESCRIPTOR ('%SET-I-MODIFY, modify ACL on !AS [N]:'),
;  1086     1082  2                             %REF (FILE_NAME),
;  1087     1083  2                             0) THEN RETURN 1;
;  1088     1084  2
;  1089     1085  2 ! Determine whether or not the target file is a directory file.
;  1090     1086  2
;  1091     1087  2 ATR_ARGLIST[0, ITM$W_ITMCOD] = ATR$C_UCHAR;
```

```
;  1092          1088  2 ATR_ARGLIST[0, ITM$W_BUFSIZ] = ATR$S_UCHAR;
;  1093          1089  2 ATR_ARGLIST[0, ITM$L_BUFADR] = FILE_CHAR;
;  1094        P 1090  2 STATUS = $QIOW (CHAN = .CHAN,
;  1095        PP 1091 2                 FUNC = IO$_ACCESS,
;  1096        P 1092  2                 IOSB = IO_STATUS,
;  1097          1093  2                 P5 = ATR_ARGLIST);
;  1098          1094  2 IF .STATUS THEN STATUS = .IO_STATUS[0];
;  1099          1095  2 IF NOT .STATUS
;  1100          1096  2 THEN
;  1101          1097  2     BEGIN
;  1102          1098  2     SIGNAL (SET$_OPENIN, 1, FILE_NAME, .STATUS, 0);
;  1103          1099  2     RETURN 1;                               ! Return without doing anything
;  1104          1100  2     END;
;  1105          1101  2 FLAGS[DIRECTORY] = .FILE_CHAR[FCH$V_DIRECTORY];
;  1106          1102  2
;  1107          1103  2 ! If the /DEFAULT qualifier is being processed, make sure that the parent
;  1108          1104  2 ! directory of the current file is accessed on the source object channel.
;  1109          1105  2
;  1110          1106  2 IF .FLAGS[QUAL_DEFAULT]
;  1111          1107  2 THEN
;  1112          1108  3     BEGIN
;  1113          1109  3
;  1114          1110  3 ! If a channel has not been assigned to the source object, assign a channel
;  1115          1111  3 ! to the device for the  parent directory.
;  1116          1112  3
;  1117          1113  3     IF .SCHAN EQL 0
;  1118          1114  3     THEN
;  1119          1115  4         BEGIN
;  1120          1116  4         CH$FILL (0, DSC$C_S_BLN, SDEVICE_DESC);
;  1121          1117  4         SDEVICE_DESC[DSC$W_LENGTH] = .VECTOR [NAM[NAM$T_DVI], 0;, BYTE];
;  1122          1118  4         SDEVICE_DESC[DSC$A_POINTER] = NAM[NAM$T_DVI] + 1;
;  1123          1119  4         STATUS = $ASSIGN (DEVNAM = SDEVICE_DESC, CHAN = SCHAN);
;  1124          1120  4         IF NOT .STATUS
;  1125          1121  4         THEN
;  1126          1122  5             BEGIN
;  1127          1123  5             SIGNAL (SET$_OPENIN, 1, SDEVICE_DESC, .STATUS, 0);
;  1128          1124  5             RETURN 1;
;  1129          1125  4             END;
;  1130          1126  3         END;
;  1131          1127  3
;  1132          1128  3 ! If there is already a directory accessed on the source object channel, and
;  1133          1129  3 ! the file-IDs are not the same, deaccess the directory file.
;  1134          1130  3
;  1135          1131  3     IF .SFILE_FIB[FIB$W_FID_NUM] NEQ 0
;  1136          1132  3     AND CH$NEQ (FIB$S_FID, SFILE_FIB[FIB$W_FID], FIB$S_FID, NAM[NAM$W_DID], 0)
;  1137          1133  3     THEN
;  1138          1134  4         BEGIN
;  1139        P 1135  4         STATUS = $QIOW (CHAN = .SCHAN,
;  1140        P 1136  4                         FUNC = IO$_DEACCESS,
;  1141          1137  4                         IOSB = IO_STATUS);
;  1142          1138  4         IF .STATUS THEN STATUS = .IO_STATUS[0];
;  1143          1139  4         IF NOT .STATUS THEN SIGNAL (SET$_CLOSEIN, 1, SOBJECT_DESC, .STATUS, 0);
;  1144          1140  4         SFILE_FIB[FIB$W_FID_NUM] = 0;           ! To force access below
;  1145          1141  4
;  1146          1142  4 ! Now release the read lock that was taken out for the directory file.
;  1147          1143  4
;  1148          1144  4         ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_UNLOCK_ACL;
```

```
;  1145          1145  4          ATR_ARGLIST[0, ITM$W_BUFSIZ] = 4;
;  1146          1146  4          ATR_ARGLIST[0, ITM$L_BUFADR] = SACL_LOCKID;
;  1147        P 1147  4          STATUS = $CHANGE_ACL (CHAN = .SCHAN,
;  1148        PP 1148  4                                OBJTYP = SOBJECT_TYPE,
;  1149        P 1149  4                                 OBJNAM = SOBJECT_DESC,
;  1150          1150  4                                 ITMLST = ATR_ARGLIST);
;  1151          1151  4          IF NOT .STATUS THEN SIGNAL (SET$_CLOSEIN, 1, SOBJECT_DESC, .STATUS, 0);
;  1152          1152  3          END;
;  1153          1153  3
;  1154          1154  3  ! If there is not a directory file currently accessed, do so now.
;  1155          1155  3
;  1156          1156  3      IF .SFILE_FIB[FIB$W_FID_NUM] EQL 0
;  1157          1157  3      THEN
;  1158          1158  4          BEGIN
;  1159          1159  4          SFILE_FIB[FIB$L_ACCTL] = 0;
;  1160          1160  4          CH$MOVE (FIB$S_FID, NAM[NAM$W_DID], SFILE_FIB[FIB$W_FID]);
;  1161        P 1161  4          STATUS = $QIOW (CHAN = .SCHAN,
;  1162        PP 1162  4                          FUNC = IO$_ACCESS OR IO$M_ACCESS,
;  1163        P 1163  4                          IOSB = IO_STATUS,
;  1164          1164  4                          P1 = SFIB_DESC);
;  1165          1165  4          IF .STATUS THEN STATUS = .IO_STATUS[0];
;  1166          1166  4          IF NOT .STATUS
;  1167          1167  4          THEN
;  1168          1168  5              BEGIN
;  1169          1169  5              SIGNAL (SET$_OPENIN, 1, SDEVICE_DESC, .STATUS, 0);
;  1170          1170  5              RETURN 1;
;  1171          1171  4              END;
;  1172          1172  4
;  1173          1173  4  ! Get the file spec for the parent directory file, in case any errors occur.
;  1174          1174  4
;  1175          1175  4          LIB$FID_TO_NAME (SDEVICE_DESC, SFILE_FIB[FIB$W_FID],
;  1176          1176  4                           SOBJECT_DESC, SOBJECT_DESC,
;  1177          1177  4                           0, STATUS1);
;  1178          1178  4
;  1179          1179  4  ! Attempt to obtain a read lock for the source object.
;  1180          1180  4
;  1181          1181  4          ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_RLOCK_ACL;
;  1182          1182  4          ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_RLOCK_ACL;
;  1183          1183  4          ATR_ARGLIST[0, ITM$L_BUFADR] = SACL_LOCKID;
;  1184        P 1184  4          STATUS = $CHANGE_ACL (CHAN = .SCHAN,
;  1185        P 1185  4                                OBJTYP = SOBJECT_TYPE,
;  1186        P 1186  4                                OBJNAM = SOBJECT_DESC,
;  1187          1187  4                                ITMLST = ATR_ARGLIST);
;  1188          1188  4          IF NOT .STATUS
;  1189          1189  4          THEN
;  1190          1190  5              BEGIN
;  1191          1191  5              IF .STATUS EQL SS$_NOTQUEUED
;  1192          1192  6              THEN SIGNAL (SET$_OBJLOCKED)
;  1193          1193  5              ELSE SIGNAL (.STATUS);
;  1194          1194  5              RETURN 1;
;  1195          1195  4              END;
;  1196          1196  3          END;
;  1197          1197  2      END;
;  1198          1198  2
;  1199          1199  2  ! Attempt to obtain a write lock for the target object.
;  1200          1200  2
;  1201          1201  2  ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_WLOCK_ACL;
```

```
;  1206          1202  2 ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_WLOCK_ACL;
;  1207          1203  2 ATR_ARGLIST[0, ITM$L_BUFADR] = ACL_LOCKID;
;  1208        P 1204  2 STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1209        P 1205  2                       OBJTYP = OBJECT_TYPE,
;  1210        P 1206  2                       OBJNAM = FILE_NAME,
;  1211          1207  2                       ITMLST = ATR_ARGLIST);
;  1212          1208  2 IF NOT .STATUS
;  1213          1209  2 THEN
;  1214          1210  3     BEGIN
;  1215          1211  3     IF .STATUS EQL SS$_NOTQUEUED
;  1216          1212  4     THEN SIGNAL (SET$_OBJLOCKED)
;  1217          1213  3     ELSE SIGNAL (.STATUS);
;  1218          1214  3     RETURN 1;
;  1219          1215  2     END;
;  1220          1216  2
;  1221          1217  2 ! Call the necessary routine based upon the command line qualifiers.
;  1222          1218  2
;  1223          1219  2 IF .FLAGS[QUAL_LIKE] OR .FLAGS[QUAL_DEFAULT] THEN STATUS = COPY_ACL (FILE_NAME)
;  1224          1220  2 ELSE IF .FLAGS[QUAL_DELETE] THEN STATUS = DELETE_ACL (FILE_NAME)
;  1225          1221  2      ELSE IF .FLAGS[QUAL_REPLACE] THEN STATUS = REPLACE_ACL (FILE_NAME)
;  1226          1222  2           ELSE STATUS = ADD_ACL (FILE_NAME);
;  1227          1223  2
;  1228          1224  2 ! Now release the write lock that was taken out.
;  1229          1225  2
;  1230          1226  2 ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_UNLOCK_ACL;
;  1231          1227  2 ATR_ARGLIST[0, ITM$W_BUFSIZ] = 4;
;  1232          1228  2 ATR_ARGLIST[0, ITM$L_BUFADR] = ACL_LOCKID;
;  1233        P 1229  2 STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1234        P 1230  2                       OBJTYP = OBJECT_TYPE,
;  1235        P 1231  2                       OBJNAM = FILE_NAME,
;  1236          1232  2                       ITMLST = ATR_ARGLIST);
;  1237          1233  2
;  1238          1234  2 ! If logging is being done, indicate that the object has been modified.
;  1239          1235  2
;  1240          1236  2 IF .FLAGS[QUAL_LOG] AND .STATUS
;  1241          1237  2 THEN SIGNAL (SET$_MODIFIED, 1, FILE_NAME);
;  1242          1238  2
;  1243          1239  2 ! Tie off the opened input file, if necessary.
;  1244          1240  2
;  1245          1241  2 IF .STATUS
;  1246          1242  2 THEN
;  1247          1243  3     BEGIN
;  1248        P 1244  3     STATUS = $QIOW (CHAN = .CHAN,
;  1249        P 1245  3                     FUNC = IO$_DEACCESS,
;  1250          1246  3                     IOSB = IO_STATUS);
;  1251          1247  3     IF .STATUS THEN STATUS = .IO_STATUS[0];
;  1252          1248  3     IF NOT .STATUS
;  1253          1249  3     THEN
;  1254          1250  4         BEGIN
;  1255          1251  4         FILE_ERROR (SET$_CLOSEIN, FAB, .STATUS, 0);
;  1256          1252  4         RETURN 1;
;  1257          1253  3         END;
;  1258          1254  3     STATUS = $DASSGN (CHAN = .CHAN);
;  1259          1255  3     IF NOT .STATUS
;  1260          1256  3     THEN
;  1261          1257  4         BEGIN
;  1262          1258  4         FILE_ERROR (SET$_CLOSEIN, FAB, .STATUS, 0);
```

```
; 1263            1259 4          RETURN 1;
; 1264            1260 3          END;
; 1265            1261 2       END;
; 1266            1262 2
; 1267            1263 2 RETURN 1;
; 1268            1264 2
; 1269            1265 1 END;                                          ! End of routine PROCESS_FILE


                                                                         .PSECT  $PLIT$,NOWRT,NOEXE,2

20 2C 59 46 49 44 4F 4D 2D 49 2D 54 45 53 25 00324 P.ACV:     .ASCII  \%SET-I-MODIFY, modify ACL on !AS [N]:\       ;
21 20 6E 6F 20 4C 43 41 20 79 66 69 64 6F 6D 00333                                                                  ;
                        3A 5D 4E 5B 20 53 41 00342                                                                  ;
                                             00349                  .BLKB   3
                                   00000025  0034C P.ACU:           .LONG   37                                       ;
                                   00000000' 00350                  .ADDRESS P.ACV                                   ;

                                                                    .EXTRN  SYS$ASSIGN

                                                                    .PSECT  $CODE$,NOWRT,2

                                      0FFC 00000 PROCESS_FILE:
                                                                    .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11    ; 0974
                    5B 00000000G  00  9E 00002                      MOVAB   SET$_OBJLOCKED, R11                     ;
                    5A 00000000G  00  9E 00009                      MOVAB   SYS$CHANGE_ACL, R10                     ;
                    59 00000000G  00  9E 00010                      MOVAB   SYS$QIOW, R9                            ;
                    58 00000000G  00  9E 00017                      MOVAB   LIB$SIGNAL, R8                          ;
                    57     0000'  CF  9E 0001E                      MOVAB   WORST_ERROR, R7                         ;
                    5E     FEB0   CE  9E 00023                      MOVAB   -336(SP), SP                            ;
      24        00          6E    00  2C 00028                      MOVC5   #0, (SP), #0, #36, ATR_ARGLIST          ; 1022
                           0648   C7     0002D                                                                      ;
           0098 CE    3C B7  0060  8F  28 00030                      MOVC3   #96, @OBJECT_FAB+40, NAM                ; 1023
   0050  8F     00          6E    00  2C 00039                      MOVC5   #0, (SP), #0, #80, $RMS_PTR             ; 1029
                             A8   AD     00040                                                                      ;
                 A8 AD     5003   8F  B0 00042                      MOVW    #20483, $RMS_PTR                        ;
                 AC AD 01020000   8F  D0 00048                      MOVL    #16908288, $RMS_PTR+4                   ;
                 BE AD     2003   8F  B0 00050                      MOVW    #8195, $RMS_PTR+22                      ;
                 C7 AD       02   90     00056                      MOVB    #2, $RMS_PTR+31                         ;
                 CC AD       6C   AE  9E 0005A                      MOVAB   XABDAT, $RMS_PTR+36                     ;
                 D0 AD     0098   CE  9E 0005F                      MOVAB   NAM, $RMS_PTR+40                        ;
      2C        00          6E    00  2C 00065                      MOVC5   #0, (SP), #0, #44, $RMS_PTR             ; 1031
                             6C   AE     0006A                                                                      ;
                 6C AE     2C12   8F  B0 0006C                      MOVW    #11282, $RMS_PTR                        ;
                 70 AE       14   AE  9E 00072                      MOVAB   XABPRO, $RMS_PTR+4                      ;
   0058  8F     00          6E    00  2C 00077                      MOVC5   #0, (SP), #0, #88, $RMS_PTR             ; 1032
                             14   AE     0007E                                                                      ;
                 14 AE     5813   8F  B0 00080                      MOVW    #22547, $RMS_PTR                        ;
                             A8   AD  9F 00086                      PUSHAB  FAB                                     ; 1034
          00000000G 00            01  FB 00089                      CALLS   #1, SYS$OPEN                            ;
                    56            50  D0 00090                      MOVL    R0, STATUS                              ;
      08        00          6E    00  2C 00093                      MOVC5   #0, (SP), #0, #8, FILE_NAME             ; 1038
                             F8   AD     00098                                                                      ;
                    50     009B   CE  9A 0009A                      MOVZBL  NAM+3, R0                               ; 1039
                                  0C  13 0009F                      BEQL    1$                                      ;
                 F8 AD            50  B0 000A1                      MOVW    R0, FILE_NAME                           ; 1042
```

```
                        FC   AD      009C    CE   D0 000A5          MOVL    NAM+4, FILE_NAME+4                                      ; 1043
                                             1D   11 000AB          BRB     3$                                                      ; 1039
                           50      00A3      CE   9A 000AD 1$:      MOVZBL  NAM+11, R0                                              ; 1045
                                             0C   13 000B2          BEQL    2$
                        F8   AD              50   B0 000B4          MOVW    R0, FILE_NAME                                           ; 1048
                        FC   AD      00A4    CE   D0 000B8          MOVL    NAM+12, FILE_NAME+4                                     ; 1049
                                             0A   11 000BE          BRB     3$                                                      ; 1045
                        F8   AD        DC    AD   9B 000C0 2$:      MOVZBW  FAB+52, FILE_NAME                                       ; 1053
                        FC   AD        D4    AD   D0 000C5          MOVL    FAB+44, FILE_NAME+4                                     ; 1054
                             27              56   E8 000CA 3$:      BLBS    STATUS, 6$                                              ; 1059
                0001828A     8F        B0    AD   D1 000CD          CMPL    FAB+8, #98954                                           ; 1067
                                             0D   12 000D5          BNEQ    5$
          0088  C7      03B0  C7              06   29 000D7          CMPC3   #6, SOBJECT_NAM+36, OBJECT_NAM+36                       ; 1068
                                             03   12 000DF          BNEQ    5$
                                        0376 31 000E1 4$:           BRW     34$
                           7E              B0   AD   7D 000E4 5$:      MOVQ    FAB+8, -(SP)                                        ; 1069
                                           A8   AD   9F 000E8          PUSHAB  FAB
                                0077109A   8F   DD 000EB              PUSHL   #7803034
                                        0361 31 000F1               BRW     33$
                        0324    C7        B4   AD   D0 000F4 6$:      MOVL    FAB+12, CHAN                                        ; 1073
                                             7E   D4 000FA          CLRL    -(SP)                                                   ; 1078
                           04   AE        F8   AD   9E 000FC          MOVAB   FILE_NAME, 4(SP)                                    ; 1082
                                           04   AE   9F 00101          PUSHAB  4(SP)
                                      0000' CF   9F 00104              PUSHAB  P.ACU                                               ; 1081
                                             7E   D4 00108          CLRL    -(SP)                                                   ; 1078
                                           A8   AD   9F 0010A          PUSHAB  FAB
                                      0644    C7   9F 0010D          PUSHAB  COMMON_CTX
                00000000G     00              06   FB 00111          CALLS   #6, LIB$QUAL_FILE_MATCH
                              C6              50   E9 00118          BLBC    R0, 4$
                        0648  C7  00030004    8F   D0 0011B          MOVL    #196612, ATR_ARGLIST                                    ; 1088
                        064C  C7        04    AE   9E 00124          MOVAB   FILE_CHAR, ATR_ARGLIST+4                            ; 1089
                                             7E   D4 0012A          CLRL    -(SP)                                                   ; 1093
                                      0648    C7   9F 0012C          PUSHAB  ATR_ARGLIST
                                             7E   7C 00130          CLRQ    -(SP)
                                             7E   7C 00132          CLRQ    -(SP)
                                             7E   7C 00134          CLRQ    -(SP)
                                        2C   AE   9F 00136          PUSHAB  IO_STATUS
                                             32   DD 00139          PUSHL   #50
                                      0324    C7   DD 0013B          PUSHL   CHAN
                                             7E   D4 0013F          CLRL    -(SP)
                                   69        0C   FB 00141          CALLS   #12, SYS$QIOW
                                   56        50   D0 00144          MOVL    R0, STATUS
                                   07        56   E9 00147          BLBC    STATUS, 7$                                              ; 1094
                                   56     0C   AE   3C 0014A          MOVZWL  IO_STATUS, STATUS
                                   0A        56   E8 0014E          BLBS    STATUS, 8$                                              ; 1095
                                             7E   D4 00151 7$:      CLRL    -(SP)                                                   ; 1098
                                             56   DD 00153          PUSHL   STATUS
                                          F8 AD   9F 00155          PUSHAB  FILE_NAME
                                        0139 31 00158               BRW     18$
          50        05    AE                01   05   EF 0015B 8$:      EXTZV   #5, #1, FILE_CHAR+1, R0                             ; 1101
    FD    A7              01                02   50   F0 00161          INSV    R0, #2, #1, FLAGS+1
                          03        FC    A7        06   E0 00167          BBS     #6, FLAGS, 9$                                   ; 1106
                                        0195 31 0016C               BRW     21$
                                      05EC  C7   D5 0016F 9$:       TSTL    SCHAN                                               ; 1113
                                             30   12 00173          BNEQ    10$
          08              00                6E   00   2C 00175          MOVC5   #0, (SP), #0, #8, SDEVICE_DESC                  ; 1116
                                      05F4  C7        0017A
```

```
                05F4   C7      FF5C   CD  9B 0017D          MOVZBW  NAM+20, SDEVICE_DESC                    ; 1117
                05F8   C7      FF5D   CD  9E 00184          MOVAB   NAM+21, SDEVICE_DESC+4                  ; 1118
                                      7E  7C 0018B          CLRQ    -(SP)                                   ; 1119
                                05EC  C7  9F 0018D          PUSHAB  SCHAN                                   ;
                                05F4  C7  9F 00191          PUSHAB  SDEVICE_DESC                            ;
            00000000G   00            04  FB 00195          CALLS   #4, SYS$ASSIGN                          ;
                        56            50  D0 0019C          MOVL    R0, STATUS                              ;
                        03            56  E8 0019F          BLBS    STATUS, 10$                             ; 1120
                                    00E7  31 001A2          BRW     17$                                     ;
                                0608  C7  B5 001A5  10$:     TSTW    SFILE_FIB+4                             ; 1131
                                      08  13 001A9          BEQL    11$                                     ;
    FF72    CD      0608    C7            06  29 001AB      CMPC3   #6, SFILE_FIB+4, NAM+42                 ; 1132
                                      03  12 001B3  11$:     BNEQ    12$                                     ;
                                    0096  31 001B5          BRW     15$                                     ;
                                      7E  7C 001B8  12$:     CLRQ    -(SP)                                   ; 1137
                                      7E  7C 001BA          CLRQ    -(SP)                                   ;
                                      7E  7C 001BC          CLRQ    -(SP)                                   ;
                                      7E  7C 001BE          CLRQ    -(SP)                                   ;
                                  2C  AE  9F 001C0          PUSHAB  IO_STATUS                               ;
                                      34  DD 001C3          PUSHL   #52                                     ;
                                05EC  C7  DD 001C5          PUSHL   SCHAN                                   ;
                                      7E  D4 001C9          CLRL    -(SP)                                   ;
                        69            0C  FB 001CB          CALLS   #12, SYS$QIOW                           ;
                        56            50  D0 001CE          MOVL    R0, STATUS                              ;
                        07            56  E9 001D1          BLBC    STATUS, 13$                             ; 1138
                        56      0C    AE  3C 001D4          MOVZWL  IO_STATUS, STATUS                       ;
                        21            56  E8 001D8          BLBS    STATUS, 14$                             ; 1139
                                      7E  D4 001DB  13$:     CLRL    -(SP)                                   ;
                                      56  DD 001DD          PUSHL   STATUS                                  ;
                                  0334  C7  9F 001DF          PUSHAB  SOBJECT_DESC                            ;
                                      01  DD 001E3          PUSHL   #1                                      ;
                            00771052      8F  DD 001E5      PUSHL   #7802962                                ;
                        68                05  FB 001EB      CALLS   #5, LIB$SIGNAL                          ;
    02          67      03                00  ED 001EE      CMPZV   #0, #3, WORST_ERROR, #2                 ;
                                          07  18 001F3      BGEQ    14$                                     ;
                        67  10771052       8F  D0 001F5      MOVL    #276238418, WORST_ERROR                 ;
                                0608      C7  B4 001FC  14$:     CLRW    SFILE_FIB+4                             ; 1140
                        0648    C7  000C0004  8F  D0 00200      MOVL    #786436, ATR_ARGLIST                    ; 1145
                        064C    C7      032C  C7  9E 00209      MOVAB   SACL_LOCKID, ATR_ARGLIST+4              ; 1146
                                      7E  7C 00210          CLRQ    -(SP)                                   ; 1150
                                      7E  D4 00212          CLRL    -(SP)                                   ;
                                  0648  C7  9F 00214          PUSHAB  ATR_ARGLIST                             ;
                                  0334  C7  9F 00218          PUSHAB  SOBJECT_DESC                            ;
                                  0330  C7  9F 0021C          PUSHAB  SOBJECT_TYPE                            ;
                                  05EC  C7  DD 00220          PUSHL   SCHAN                                   ;
                        6A            07  FB 00224          CALLS   #7, SYS$CHANGE_ACL                      ;
                        56            50  D0 00227          MOVL    R0, STATUS                              ;
                        21            56  E8 0022A          BLBS    STATUS, 15$                             ; 1151
                                      7E  D4 0022D          CLRL    -(SP)                                   ;
                                      56  DD 0022F          PUSHL   STATUS                                  ;
                                  0334  C7  9F 00231          PUSHAB  SOBJECT_DESC                            ;
                                      01  DD 00235          PUSHL   #1                                      ;
                            00771052      8F  DD 00237      PUSHL   #7802962                                ;
                        68                05  FB 0023D      CALLS   #5, LIB$SIGNAL                          ;
    02          67      03                00  ED 00240      CMPZV   #0, #3, WORST_ERROR, #2                 ;
                                          07  18 00245      BGEQ    15$                                     ;
                        67  10771052       8F  D0 00247      MOVL    #276238418, WORST_ERROR                 ;
```

```
                                0608   C7  B5 0024E 15$:    TSTW    SFILE_FIB+4                                    ; 1156
                                       03  13 00252         BEQL    16$
                                   00AD 31  00254         BRW     21$                                            ;
                                0604   C7  D4 00257 16$:    CLRL    SFILE_FIB                                      ; 1159
        0608    C7    FF72  CD           06  28 0025B         MOVC3   #6, NAM+42, SFILE_FIB+4                        ; 1160
                                       7E  7C 00263         CLRQ    -(SP)                                          ; 1164
                                       7E  7C 00265         CLRQ    -(SP)                                          ;
                                       7E  D4 00267         CLRL    -(SP)                                          ;
                                05FC   C7  9F 00269         PUSHAB  SFIB_DESC                                      ;
                                       7E  7C 0026D         CLRQ    -(SP)                                          ;
                                  2C   AE  9F 0026F         PUSHAB  IO_STATUS                                      ;
                              7E    72   8F  9A 00272         MOVZBL  #1T4, -(SP)                                    ;
                                05EC   C7  DD 00276         PUSHL   SCHAN                                          ;
                                       7E  D4 0027A         CLRL    -(SP)                                          ;
                              69       0C  FB 0027C         CALLS   #12, SYS$QIOW                                  ;
                              56       50  D0 0027F         MOVL    R0, STATUS                                     ;
                              07       56  E9 00282         BLBC    STATUS, 17$                                    ; 1165
                              56    0C AE  3C 00285         MOVZWL  IO_STATUS, STATUS                              ;
                              24       56  E8 00289         BLBS    STATUS, 20$                                    ; 1166
                                       7E  D4 0028C 17$:    CLRL    -(SP)                                          ; 1169
                                       56  DD 0028E         PUSHL   STATUS                                         ;
                                05F4   C7  9F 00290         PUSHAB  SDEVICE_DESC                                   ;
                                       01  DD 00294 18$:    PUSHL   #1                                             ;
                             0077109A  8F  DD 00296         PUSHL   #7803034                                       ;
                              68       05  FB 0029C         CALLS   #5, LIB$SIGNAL                                 ;
        02          67       03       00  ED 0029F         CMPZV   #0, #3, WORST_ERROR, #2                        ;
                                       07  18 002A4         BGEQ    19$                                            ;
                              67 1077109A  8F  D0 002A6         MOVL    #276238490, WORST_ERROR                        ;
                                   01AA 31  002AD 19$:    BRW     34$                                            ; 1170
                                   08   AE  9F 002B0 20$:    PUSHAB  STATUS1                                        ; 1175
                                       7E  D4 002B3         CLRL    -(SP)                                          ;
                                0334   C7  9F 002B5         PUSHAB  SOBJECT_DESC                                   ;
                                0334   C7  9F 002B9         PUSHAB  SOBJECT_DESC                                   ;
                                0608   C7  9F 002BD         PUSHAB  SFILE_FIB+4                                    ;
                                05F4   C7  9F 002C1         PUSHAB  SDEVICE_DESC                                   ;
  00000000G     00                   06  FB 002C5         CALLS   #6, LIB$FID_TO_NAME                            ;
      0648      C7 000A0004  8F      D0 002CC         MOVL    #655364, ATR_ARGLIST                           ; 1182
      064C      C7      032C C7      9E 002D5         MOVAB   SACL_LOCKID, ATR_ARGLIST+4                     ; 1183
                                       7E  7C 002DC         CLRQ    -(SP)                                          ; 1187
                                       7E  D4 002DE         CLRL    -(SP)                                          ;
                                0648   C7  9F 002E0         PUSHAB  ATR_ARGLIST                                    ;
                                0334   C7  9F 002E4         PUSHAB  SOBJECT_DESC                                   ;
                                0330   C7  9F 002E8         PUSHAB  SOBJECT_TYPE                                   ;
                                05EC   C7  DD 002EC         PUSHL   SCHAN                                          ;
                              6A       07  FB 002F0         CALLS   #7, SYS$CHANGE_ACL                             ;
                              56       50  D0 002F3         MOVL    R0, STATUS                                     ;
                              0B       56  E8 002F6         BLBS    STATUS, 21$                                    ; 1188
      000009B8  8F                     56  D1 002F9         CMPL    STATUS, #2488                                  ; 1191
                                       35  13 00300         BEQL    22$                                            ;
                                       55  11 00302         BRB     23$                                            ; 1193
      0648      C7 000B0004  8F      D0 00304 21$:    MOVL    #720900, ATR_ARGLIST                           ; 1202
      064C      C7      04   A7      9E 0030D         MOVAB   ACL_LOCKID, ATR_ARGLIST+4                      ; 1203
                                       7E  7C 00313         CLRQ    -(SP)                                          ; 1207
                                       7E  D4 00315         CLRL    -(SP)                                          ;
                                0648   C7  9F 00317         PUSHAB  ATR_ARGLIST                                    ;
                                  F8   AD  9F 0031B         PUSHAB  FILE_NAME                                      ;
                                  08   A7  9F 0031E         PUSHAB  OBJECT_TYPE                                    ;
```

```
                               0324   C7  DD 00321           PUSHL    CHAN                                          ;
                    6A                07  FB 00325           CALLS    #7, SYS$CHANGE_ACL                            ;
                    56                50  D0 00328           MOVL     R0, STATUS                                    ;
                    4A                56  E8 0032B           BLBS     STATUS, 25$                                   ; 1208
          000009B8  8F                56  D1 0032E           CMPL     STATUS, #2488                                 ; 1211
                    22                12  00335              BNEQ     23$                                           ;
                    5B                DD  00337  22$:        PUSHL    R11                                           ; 1212
                    68                01  FB 00339           CALLS    #1, LIB$SIGNAL                                ;
                    50                6B  9E 0033C           MOVAB    SET$_OBJLOCKED, R0                            ;
                    33                50  E8 0033F           BLBS     R0, 24$                                       ;
                    50 00000000*      00  9E 00342           MOVAB    <SET$_OBJLOCKED&7>, R0                        ;
     50        67   03                00  ED 00349           CMPZV    #0, #3, WORST_ERROR, R0                       ;
                    25                18  0034E              BGEQ     24$                                           ;
                    67 00000000*      00  9E 00350           MOVAB    <SET$_OBJLOCKED!268435456>, WORST_ERROR       ;
                    1C                11  00357              BRB      24$                                           ; 1211
                    56                DD  00359  23$:        PUSHL    STATUS                                        ; 1213
                    68                01  FB 0035B           CALLS    #1, LIB$SIGNAL                                ;
                    14                56  E8 0035E           BLBS     STATUS, 24$                                   ;
     50        56   03                00  EF 00361           EXTZV    #0, #3, STATUS, R0                            ;
     50        67   03                00  ED 00366           CMPZV    #0, #3, WORST_ERROR, R0                       ;
                    08                18  0036B              BGEQ     24$                                           ;
               67   56 10000000       8F  C9 0036D           BISL3    #268435456, STATUS, WORST_ERROR               ;
                               00E2   31  00375  24$:        BRW      34$                                           ; 1214
     05      FC     A7                02  E0 00378  25$:     BBS      #2, FLAGS, 26$                                ; 1219
     0A      FC     A7                06  E1 0037D           BBC      #6, FLAGS, 27$                                ;
                                 F8   AD  9F 00382  26$:     PUSHAB   FILE_NAME                                     ;
          0000V     CF                01  FB 00385           CALLS    #1, COPY_ACL                                  ;
                    26                11  0038A              BRB      30$                                           ;
     0A      FC     A7                01  E1 0038C  27$:     BBC      #1, FLAGS, 28$                                ; 1220
                                 F8   AD  9F 00391           PUSHAB   FILE_NAME                                     ;
          0000V     CF                01  FB 00394           CALLS    #1, DELETE_ACL                                ;
                    17                11  00399              BRB      30$                                           ;
     0A      FC     A7                04  E1 0039B  28$:     BBC      #4, FLAGS, 29$                                ; 1221
                                 F8   AD  9F 003A0           PUSHAB   FILE_NAME                                     ;
          0000V     CF                01  FB 003A3           CALLS    #1, REPLACE_ACL                               ;
                    08                11  003A8              BRB      30$                                           ;
                                 F8   AD  9F 003AA  29$:     PUSHAB   FILE_NAME                                     ; 1222
          0000V     CF                01  FB 003AD           CALLS    #1, ADD_ACL                                   ;
                    56                50  D0 003B2  30$:     MOVL     R0, STATUS                                    ;
          0648      C7 000C0004       8F  D0 003B5           MOVL     #786436, ATR_ARGLIST                          ; 1227
          064C      C7           04   A7  9E 003BE           MOVAB    ACL_LOCKID, ATR_ARGLIST+4                     ; 1228
                                      7E  7C 003C4           CLRQ     -(SP)                                         ; 1232
                                      7E  D4 003C6           CLRL     -(SP)                                         ;
                               0648   C7  9F 003C8           PUSHAB   ATR_ARGLIST                                   ;
                                 F8   AD  9F 003CC           PUSHAB   FILE_NAME                                     ;
                                 08   A7  9F 003CF           PUSHAB   OBJECT_TYPE                                   ;
                               0324   C7  DD 003D2           PUSHL    CHAN                                          ;
                    6A                07  FB 003D6           CALLS    #7, SYS$CHANGE_ACL                            ;
                    56                50  D0 003D9           MOVL     R0, STATUS                                    ;
     30      FC     A7                03  E1 003DC           BBC      #3, FLAGS, 31$                                ; 1236
                    76                56  E9 003E1           BLBC     STATUS, 34$                                   ;
                                 F8   AD  9F 003E4           PUSHAB   FILE_NAME                                     ; 1237
                                      01  DD 003E7           PUSHL    #1                                            ;
                          00000000G   00  9F 003E9           PUSHAB   SET$_MODIFIED                                 ;
                    68                03  FB 003EF           CALLS    #3, LIB$SIGNAL                                ;
                    50 00000000G      00  9E 003F2           MOVAB    SET$_MODIFIED, R0                             ;
                    15                50  E8 003F9           BLBS     R0, 31$                                       ;
```

```
                          50 00000000*  00  9E  003FC              MOVAB   <SET$_MODIFIED&7>, R0                          ;
          50        67    03            00  ED  00403              CMPZV   #0, #3, WORST_ERROR, R0                       ;
                                        07  18  00408              BGEQ    31$                                           ;
                          67 00000000*  00  9E  0040A              MOVAB   <SET$_MODIFIED!268435456>, WORST_ERROR        ;
                          46            56  E9  00411  31$:       BLBC    STATUS, 34$                                   ; 1241
                                        7E  7C  00414              CLRQ    -(SP)                                         ; 1246
                                        7E  7C  00416              CLRQ    -(SP)                                         ;
                                        7E  7C  00418              CLRQ    -(SP)                                         ;
                                        7E  7C  0041A              CLRQ    -(SP)                                         ;
                                2C      AE  9F  0041C              PUSHAB  IO_STATUS                                     ;
                                        34  DD  0041F              PUSHL   #52                                           ;
                              0324      C7  DD  00421              PUSHL   CHAN                                          ;
                                        7E  D4  00425              CLRL    -(SP)                                         ;
                          69            0C  FB  00427              CALLS   #12, SYS$QIOW                                 ;
                          56            50  D0  0042A              MOVL    R0, STATUS                                    ;
                          18            56  E9  0042D              BLBC    STATUS, 32$                                   ; 1247
                          56        0C  AE  3C  00430              MOVZWL  IO_STATUS, STATUS                             ;
                          11            56  E9  00434              BLBC    STATUS, 32$                                   ; 1248
                              0324      C7  DD  00437              PUSHL   CHAN                                          ; 1254
                00000000G 00            01  FB  0043B              CALLS   #1, SYS$DASSGN                                ;
                          56            50  D0  00442              MOVL    R0, STATUS                                    ;
                          12            56  E8  00445              BLBS    STATUS, 34$                                   ; 1255
                                        7E  D4  00448  32$:       CLRL    -(SP)                                         ; 1258
                                        56  DD  0044A              PUSHL   STATUS                                        ;
                                A8      AD  9F  0044C              PUSHAB  FAB                                           ;
                          00771052      8F  DD  0044F              PUSHL   #7802962                                      ;
                    0000V CF            04  FB  00455  33$:       CALLS   #4, FILE_ERROR                                ;
                          50            01  D0  0045A  34$:       MOVL    #1, R0                                        ; 1263
                                        04  0045D                  RET                                                   ; 1265

; Routine Size:  1118 bytes,    Routine Base:  $CODE$ + 0AAB
```

```
;  1271     1266  1 ROUTINE ADD_ACL (OBJECT_NAME_DESC) =
;  1272     1267  1
;  1273     1268  1 !++
;  1274     1269  1 !
;  1275     1270  1 ! FUNCTIONAL DESCRIPTION:
;  1276     1271  1 !
;  1277     1272  1 !       This routine adds ACEs to the end of the ACL or inserts ACEs into
;  1278     1273  1 !       various points within the ACL.
;  1279     1274  1 !
;  1280     1275  1 ! CALLING SEQUENCE:
;  1281     1276  1 !       ADD_ACL (ARG1)
;  1282     1277  1 !
;  1283     1278  1 ! INPUT PARAMETERS:
;  1284     1279  1 !       ARG1: address of the FAB
;  1285     1280  1 !
;  1286     1281  1 ! IMPLICIT INPUTS:
;  1287     1282  1 !       none
;  1288     1283  1 !
;  1289     1284  1 ! OUTPUT PARAMETERS:
;  1290     1285  1 !       none
;  1291     1286  1 !
;  1292     1287  1 ! IMPLICIT OUTPUTS:
;  1293     1288  1 !       none
;  1294     1289  1 !
;  1295     1290  1 ! ROUTINE VALUE:
;  1296     1291  1 !       1 if successful
;  1297     1292  1 !       error code otherwise
;  1298     1293  1 !
;  1299     1294  1 ! SIDE EFFECTS:
;  1300     1295  1 !       none
;  1301     1296  1 !
;  1302     1297  1 !--
;  1303     1298  1
;  1304     1299  2 BEGIN
;  1305     1300  2
;  1306     1301  2 LOCAL
;  1307     1302  2         STATUS;                                         ! Local routine return status
;  1308     1303  2
;  1309     1304  2 ! Preset the context to start adding ACEs at the beginning of the ACL.
;  1310     1305  2
;  1311     1306  2 ACL_CONTEXT = 0;
;  1312     1307  2
;  1313     1308  2 ! If this is a new ACL, delete any ACL that currently exists on the object.
;  1314     1309  2
;  1315     1310  2 IF .FLAGS[QUAL_NEW]
;  1316     1311  2 THEN
;  1317     1312  3     BEGIN
;  1318     1313  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_DELETEACL;
;  1319     1314  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_DELETEACL;
;  1320     1315  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1321   P 1316  3     STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1322   P 1317  3                           OBJTYP = OBJECT_TYPE,
;  1323   P 1318  3                           OBJNAM = .OBJECT_NAME_DESC,
;  1324   P 1319  3                           ITMLST = ATR_ARGLIST,
;  1325     1320  3                           CONTXT = ACL_CONTEXT);
;  1326     1321  3     IF NOT .STATUS
;  1327     1322  3     THEN
```

```
;  1328          1323  4          BEGIN
;  1329          1324  4          SIGNAL (SET$_WRITEERR, .OBJECT_NAME_DESC, .STATUS, 0);
;  1330          1325  4          RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1331          1326  3          END;
;  1332          1327  2      END;
;  1333          1328  2
;  1334          1329  2  ! For an insert, first locate the ACE after which the new ACEs will be added.
;  1335          1330  2
;  1336          1331  2  IF .FLAGS[QUAL_AFTER]
;  1337          1332  2  THEN
;  1338          1333  3      BEGIN
;  1339          1334  3      ACE_POINTER = .OLD_ACE_HEAD[ACEQ_L_FLINK];
;  1340          1335  3      CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1341          1336  3               ACE_POINTER[ACEQ_T_ACE], ACE);
;  1342          1337  3      ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_FNDACLENT;
;  1343          1338  3      ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1344          1339  3      ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1345        P 1340  3      STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1346        P 1341  3                            OBJTYP = OBJECT_TYPE,
;  1347        P 1342  3                            OBJNAM = .OBJECT_NAME_DESC,
;  1348        P 1343  3                            ITMLST = ATR_ARGLIST,
;  1349          1344  3                            CONTXT = ACL_CONTEXT);
;  1350          1345  3      IF NOT .STATUS
;  1351          1346  3      THEN
;  1352          1347  4          BEGIN
;  1353          1348  4          SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1354          1349  4          RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1355          1350  3          END;
;  1356          1351  3      IF .ACE[ACE$B_SIZE] EQL 0
;  1357          1352  3      THEN
;  1358          1353  4          BEGIN
;  1359          1354  4          IF .ACE[ACE$W_FLAGS] NEQ SS$_ACLEMPTY
;  1360          1355  4          THEN
;  1361          1356  5              BEGIN
;  1362          1357  5              SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .ACE[ACE$W_FLAGS], 0);
;  1363          1358  5              RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1364          1359  4              END;
;  1365          1360  3          END;
;  1366          1361  3      ACL_CONTEXT = .ACL_CONTEXT + 1;
;  1367          1362  2      END;
;  1368          1363  2
;  1369          1364  2  ! Now that the context has been set, add the new ACEs.
;  1370          1365  2
;  1371          1366  2  ACE_POINTER = .NEW_ACE_HEAD[ACEQ_L_FLINK];
;  1372          1367  2  UNTIL .ACE_POINTER EQLA NEW_ACE_HEAD[ACEQ_L_FLINK]
;  1373          1368  2  DO
;  1374          1369  3      BEGIN
;  1375          1370  3      CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1376          1371  3               ACE_POINTER[ACEQ_T_ACE], ACE);
;  1377          1372  3      ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_ADDACLENT;
;  1378          1373  3      ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1379          1374  3      ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1380        P 1375  3      STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1381        P 1376  3                            OBJTYP = OBJECT_TYPE,
;  1382        P 1377  3                            OBJNAM = .OBJECT_NAME_DESC,
;  1383        P 1378  3                            ITMLST = ATR_ARGLIST,
;  1384          1379  3                            CONTXT = ACL_CONTEXT);
```

```
;  1385          1380  3      IF NOT .STATUS
;  1386          1381  3      THEN
;  1387          1382  4          BEGIN
;  1388          1383  4          SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1389          1384  4          RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1390          1385  3          END;
;  1391          1386  3      ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1392          1387  2      END;
;  1393          1388  2
;  1394          1389  2  RETURN 1;
;  1395          1390  2
;  1396          1391  1  END;                                    ! End of routine ADD_ACL


                                        03FC 00000 ADD_ACL:.WORD   Save R2,R3,R4,R5,R6,R7,R8,R9          ; 1266
                              59 00000000G   00  9E 00002         MOVAB   LIB$SIGNAL, R9                  ;
                              58 00000000G   00  9E 00009         MOVAB   SYS$CHANGE_ACL, R8              ;
                              57      0000'  CF  9E 00010         MOVAB   ACE, R7                         ;
                                     FCA8    C7  D4 00015         CLRL    ACL_CONTEXT                     ; 1306
                        48         F97C    C7     05  E1 00019   BBC     #5, FLAGS, 3$                   ; 1310
                                     C8    A7 000600FF   8F  D0 0001F   MOVL    #393471, ATR_ARGLIST    ; 1314
                                     CC    A7            67  9E 00027   MOVAB   ACE, ATR_ARGLIST+4       ; 1315
                                           FCA8    C7  9F 0002B         PUSHAB  ACL_CONTEXT             ; 1320
                                                   7E  7C 0002F         CLRQ    -(SP)                   ;
                                             C8    A7  9F 00031         PUSHAB  ATR_ARGLIST             ;
                                             04    AC  DD 00034         PUSHL   OBJECT_NAME_DESC        ;
                                           F988    C7  9F 00037         PUSHAB  OBJECT_TYPE             ;
                                           FCA4    C7  DD 0003B         PUSHL   CHAN                    ;
                              68                   07  FB 0003F         CALLS   #7, SYS$CHANGE_ACL      ;
                              56                   50  D0 00042         MOVL    R0, STATUS              ;
                              1F                   56  E8 00045         BLBS    STATUS, 3$              ; 1321
                                                   7E  D4 00048         CLRL    -(SP)                   ; 1324
                                                   56  DD 0004A         PUSHL   STATUS                  ;
                                             04    AC  DD 0004C         PUSHL   OBJECT_NAME_DESC        ;
                                    007710D4       8F  DD 0004F         PUSHL   #7803092                ;
                              69                   04  FB 00055         CALLS   #4, LIB$SIGNAL          ;
                        04         F980    C7     03     00  ED 00058 1$:   CMPZV   #0, #3, WORST_ERROR, #4 ;
                                                   03  19 0005F         BLSS    2$                      ;
                                                 00DE  31 00061         BRW     10$                     ;
                                                 00D2  31 00064 2$:     BRW     9$                      ;
                                    6A         F97C    C7  E9 00067 3$:   BLBC    FLAGS, 7$               ; 1331
                                 0200    C7    0E0C    C7  D0 0006C     MOVL    OLD_ACE_HEAD, ACE_POINTER   ; 1334
                                    50       0200    C7  D0 00073     MOVL    ACE_POINTER, R0             ; 1335
                                    51         08    A0  9A 00078     MOVZBL  8(R0), R1                   ;
                        67          08    A0                 51  28 0007C   MOVC3   R1, 8(R0), ACE      ; 1336
                                    CA    A7                 04  B0 00081   MOVW    #4, ATR_ARGLIST+2   ; 1337
                                    C8    A7                 67  9B 00085   MOVZBW  ACE, ATR_ARGLIST    ; 1338
                                    CC    A7                 67  9E 00089   MOVAB   ACE, ATR_ARGLIST+4  ; 1339
                                                   FCA8    C7  9F 0008D     PUSHAB  ACL_CONTEXT         ; 1344
                                                           7E  7C 00091     CLRQ    -(SP)               ;
                                                     C8    A7  9F 00093     PUSHAB  ATR_ARGLIST         ;
                                                     04    AC  DD 00096     PUSHL   OBJECT_NAME_DESC    ;
                                                   F988    C7  9F 00099     PUSHAB  OBJECT_TYPE         ;
                                                   FCA4    C7  DD 0009D     PUSHL   CHAN                ;
```

```
                                68              07  FB 000A1          CALLS   #7, SYS$CHANGE_ACL                      ;
                                56              50  D0 000A4          MOVL    R0, STATUS                             ;
                                14              56  E8 000A7          BLBS    STATUS, 5$                             ; 1345
                                                7E  D4 000AA          CLRL    -(SP)                                  ; 1348
                                                56  DD 000AC          PUSHL   STATUS                                 ;
                                        04      AC  DD 000AE 4$:      PUSHL   OBJECT_NAME_DESC                       ;
                                                01  DD 000B1          PUSHL   #1                                     ;
                                007710D4        8F  DD 000B3          PUSHL   #7803092                               ;
                                69              05  FB 000B9          CALLS   #5, LIB$SIGNAL                         ;
                                                9A  11 000BC          BRB     1$                                     ;
                                                67  95 000BE 5$:      TSTB    ACE                                    ; 1351
                                                10  12 000C0          BNEQ    6$                                     ;
                        09D0    8F      02      A7  B1 000C2          CMPW    ACE+2, #2512                           ; 1354
                                                08  13 000C8          BEQL    6$                                     ;
                                                7E  D4 000CA          CLRL    -(SP)                                  ; 1357
                                7E      02      A7  3C 000CC          MOVZWL  ACE+2, -(SP)                           ;
                                                DC  11 000D0          BRB     4$                                     ;
                                        FCA8    C7  D6 000D2 6$:      INCL    ACL_CONTEXT                            ; 1361
                        0200    C7      0E14    C7  D0 000D6 7$:      MOVL    NEW_ACE_HEAD, ACE_POINTER              ; 1366
                                50      0200    C7  D0 000DD 8$:      MOVL    ACE_POINTER, R0                        ; 1367
                                51      0E14    C7  9E 000E2          MOVAB   NEW_ACE_HEAD, R1                       ;
                                51              50  D1 000E7          CMPL    R0, R1                                 ;
                                                67  13 000EA          BEQL    12$                                    ;
                                51      08      A0  9A 000EC          MOVZBL  8(R0), R1                              ; 1370
                        67      08      A0          51  28 000F0      MOVC3   R1, 8(R0), ACE                         ; 1371
                                CA      A7          01  B0 000F5      MOVW    #1, ATR_ARGLIST+2                      ; 1372
                                C8      A7          67  9B 000F9      MOVZBW  ACE, ATR_ARGLIST                       ; 1373
                                CC      A7          67  9E 000FD      MOVAB   ACE, ATR_ARGLIST+4                     ; 1374
                                        FCA8        C7  9F 00101      PUSHAB  ACL_CONTEXT                            ; 1379
                                                    7E  7C 00105      CLRQ    -(SP)                                  ;
                                        C8          A7  9F 00107      PUSHAB  ATR_ARGLIST                            ;
                                        04          AC  DD 0010A      PUSHL   OBJECT_NAME_DESC                       ;
                                        F988        C7  9F 0010D      PUSHAB  OBJECT_TYPE                            ;
                                        FCA4        C7  DD 00111      PUSHL   CHAN                                   ;
                                68                      07  FB 00115 CALLS   #7, SYS$CHANGE_ACL                    ;
                                56                      50  D0 00118 MOVL    R0, STATUS                           ;
                                2C                      56  E8 0011B BLBS    STATUS, 11$                          ; 1380
                                                        7E  D4 0011E CLRL    -(SP)                                ; 1383
                                                        56  DD 00120 PUSHL   STATUS                               ;
                                                04      AC  DD 00122 PUSHL   OBJECT_NAME_DESC                     ;
                                                        01  DD 00125 PUSHL   #1                                   ;
                                            007710D4    8F  DD 00127 PUSHL   #7803092                             ;
                                69                      05  FB 0012D CALLS   #5, LIB$SIGNAL                       ;
                04      F980    C7              03      00  ED 00130 CMPZV   #0, #3, WORST_ERROR, #4                ;
                                                        09  18 00137 BGEQ    10$                                  ;
                                        F980    C7  107710D4    8F  D0 00139 9$:  MOVL    #276238548, WORST_ERROR  ;
                                50              C7  107710D4    8F  D0 00142 10$: MOVL    #276238548, R0           ; 1384
                                                                    04 00149 RET                                   ;
                                        0200    C7      0200    D7  D0 0014A 11$: MOVL    @ACE_POINTER, ACE_POINTER ; 1386
                                                                8A  11 00151 BRB     8$                            ; 1367
                                50                      01  D0 00153 12$: MOVL    #1, R0                       ; 1389
                                                            04 00156 RET                                           ; 1391

; Routine Size:  343 bytes,    Routine Base:  $CODE$ + 0F09
```

```
;  1398          1392  1 ROUTINE DELETE_ACL (OBJECT_NAME_DESC) =
;  1399          1393  1
;  1400          1394  1 !++
;  1401          1395  1 !
;  1402          1396  1 ! FUNCTIONAL DESCRIPTION:
;  1403          1397  1 !
;  1404          1398  1 !       This routine deletes one or more ACEs (or the entire ACL) from
;  1405          1399  1 !       the specified object.
;  1406          1400  1 !
;  1407          1401  1 ! CALLING SEQUENCE:
;  1408          1402  1 !       ADD_ACL (ARG1)
;  1409          1403  1 !
;  1410          1404  1 ! INPUT PARAMETERS:
;  1411          1405  1 !       ARG1: address of the FAB
;  1412          1406  1 !
;  1413          1407  1 ! IMPLICIT INPUTS:
;  1414          1408  1 !       none
;  1415          1409  1 !
;  1416          1410  1 ! OUTPUT PARAMETERS:
;  1417          1411  1 !       none
;  1418          1412  1 !
;  1419          1413  1 ! IMPLICIT OUTPUTS:
;  1420          1414  1 !       none
;  1421          1415  1 !
;  1422          1416  1 ! ROUTINE VALUE:
;  1423          1417  1 !       1 if successful
;  1424          1418  1 !       error code otherwise
;  1425          1419  1 !
;  1426          1420  1 ! SIDE EFFECTS:
;  1427          1421  1 !       none
;  1428          1422  1 !
;  1429          1423  1 !--
;  1430          1424  1
;  1431          1425  2 BEGIN
;  1432          1426  2
;  1433          1427  2 LOCAL
;  1434          1428  2         STATUS;                                 ! Local routine return status
;  1435          1429  2
;  1436          1430  2 ! If there were ACEs given on the /ACL qualifier, just those specified ACEs
;  1437          1431  2 ! are deleted.  Otherwise, the entire ACL is deleted.
;  1438          1432  2
;  1439          1433  2 IF .OLD_ACE_HEAD[ACEQ_L_FLINK] NEQA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  1440          1434  2 THEN
;  1441          1435  3     BEGIN
;  1442          1436  3
;  1443          1437  3 ! Before deleting any of the given ACEs, make sure that they all exist.
;  1444          1438  3
;  1445          1439  3     ACE_POINTER = .OLD_ACE_HEAD[ACEQ_L_FLINK];
;  1446          1440  3     UNTIL .ACE_POINTER EQLA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  1447          1441  3     DO
;  1448          1442  4         BEGIN
;  1449          1443  4         CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1450          1444  4                  ACE_POINTER[ACEQ_T_ACE], ACE);
;  1451          1445  4         ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_FNDACLENT;
;  1452          1446  4         ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1453          1447  4         ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1454        P 1448  4         STATUS = $CHANGE_ACL (CHAN = .CHAN,
```

```
;  1455        P 1449  4                                OBJTYP = OBJECT_TYPE,
;  1456        P 1450  4                                OBJNAM = .OBJECT_NAME_DESC,
;  1457        P 1451  4                                ITMLST = ATR_ARGLIST,
;  1458          1452  4                                CONTXT = ACL_CONTEXT);
;  1459          1453  4            IF NOT .STATUS
;  1460          1454  4            THEN
;  1461          1455  5                BEGIN
;  1462          1456  5                IF .STATUS NEQ SS$_ACLEMPTY
;  1463          1457  5                AND .STATUS NEQ SS$_NOENTRY
;  1464          1458  5                THEN
;  1465          1459  6                    BEGIN
;  1466          1460  6                    SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1467          1461  6                    RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1468          1462  5                    END;
;  1469          1463  5                ACE_DESC[DSC$W_LENGTH] = .$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE];
;  1470          1464  5                ACE_DESC[DSC$A_POINTER] = ACE_POINTER[ACEQ_T_ACE];
;  1471          1465  5                ACE_TEXT_DESC[DSC$W_LENGTH] = 3072;
;  1472          1466  5                ACE_TEXT_DESC[DSC$A_POINTER] = ACE_TEXT;
;  1473        P 1467  5                $FORMAT_ACL (ACLENT = ACE_DESC,
;  1474        P 1468  5                             ACLLEN = ACE_TEXT_DESC[DSC$W_LENGTH],
;  1475        P 1469  5                             ACLSTR = ACE_TEXT_DESC,
;  1476        P 1470  5                             WIDTH = %REF (80),
;  1477        P 1471  5                             TRMDSC = $DESCRIPTOR (%CHAR (13), %CHAR (10)),
;  1478          1472  5                             INDENT = %REF (4));
;  1479          1473  5                SIGNAL (SET$_NOSUCHACE, 2, .OBJECT_NAME_DESC, ACE_TEXT_DESC);
;  1480          1474  4                END;
;  1481          1475  4            ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1482          1476  3            END;
;  1483          1477  3
;  1484          1478  3    ! Delete the specified ACEs.
;  1485          1479  3
;  1486          1480  3        ACE_POINTER = .OLD_ACE_HEAD[ACEQ_L_FLINK];
;  1487          1481  3        UNTIL .ACE_POINTER EQLA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  1488          1482  3        DO
;  1489          1483  4            BEGIN
;  1490          1484  4            CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1491          1485  4                     ACE_POINTER[ACEQ_T_ACE], ACE);
;  1492          1486  4            ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_DELACLENT;
;  1493          1487  4            ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1494          1488  4            ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1495        P 1489  4            STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1496        P 1490  4                                  OBJTYP = OBJECT_TYPE,
;  1497        P 1491  4                                  OBJNAM = .OBJECT_NAME_DESC,
;  1498        P 1492  4                                  ITMLST = ATR_ARGLIST,
;  1499          1493  4                                  CONTXT = ACL_CONTEXT);
;  1500          1494  4            IF NOT .STATUS
;  1501          1495  4            THEN
;  1502          1496  5                BEGIN
;  1503          1497  5                SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1504          1498  5                RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1505          1499  4                END;
;  1506          1500  4            ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1507          1501  3            END;
;  1508          1502  3        END
;  1509          1503  2    ELSE
;  1510          1504  3        BEGIN
;  1511          1505  3
```

```
;  1512          1506  3 ! Delete any ACL that currently exists on the object.
;  1513          1507  3
;  1514          1508  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_DELETEACL;
;  1515          1509  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_DELETEACL;
;  1516          1510  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1517        P 1511  3     STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1518        P 1512  3                           OBJTYP = OBJECT_TYPE,
;  1519        P 1513  3                           OBJNAM = .OBJECT_NAME_DESC,
;  1520        P 1514  3                           ITMLST = ATR_ARGLIST,
;  1521          1515  3                           CONTXT = ACL_CONTEXT);
;  1522          1516  3     IF NOT .STATUS
;  1523          1517  3     THEN
;  1524          1518  4         BEGIN
;  1525          1519  4         SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1526          1520  4         RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1527          1521  3         END;
;  1528          1522  2     END;
;  1529          1523  2
;  1530          1524  2 RETURN 1;
;  1531          1525  2
;  1532          1526  1 END;                                   ! End of routine DELETE_ACL


                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

                                              0D  00354 P.ACX:  .ASCII  <13>                                    ;
                                              0A  00355         .ASCII  <10>                                    ;
                                                  00356         .BLKB   2
                                          00000002  00358 P.ACW:  .LONG   2                                     ;
                                          00000000' 0035C         .ADDRESS P.ACX                                ;

                                                    .EXTRN  SYS$FORMAT_ACL

                                                    .PSECT  $CODE$,NOWRT,2

                                                  0FFC 00000 DELETE_ACL:
                                                             .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11       ; 1392
                                      5B 00000000G   00  9E 00002         MOVAB   SET$_NOSUCHACE, R11           ;
                                      5A 00000000G   00  9E 00009         MOVAB   LIB$SIGNAL, R10               ;
                                      59 00000000G   00  9E 00010         MOVAB   SYS$CHANGE_ACL, R9            ;
                                      58      0000'  CF  9E 00017         MOVAB   ACE_POINTER, R8               ;
                                      5E             08  C2 0001C         SUBL2   #8, SP                        ;
                                      56          04 AC  D0 0001F         MOVL    OBJECT_NAME_DESC, R6          ; 1452
                                      50      0C0C   C8  9E 00023         MOVAB   OLD_ACE_HEAD, R0              ; 1433
                                      50      0C0C   C8  D1 00028         CMPL    OLD_ACE_HEAD, R0              ;
                                                     03  12 0002D         BNEQ    1$                            ;
                                                 014D    31 0002F         BRW     13$                           ;
                                      68      0C0C   C8  D0 00032 1$:     MOVL    OLD_ACE_HEAD, ACE_POINTER     ; 1439
                                      50             68  D0 00037 2$:     MOVL    ACE_POINTER, R0               ; 1440
                                      51      0C0C   C8  9E 0003A         MOVAB   OLD_ACE_HEAD, R1              ;
                                      51             50  D1 0003F         CMPL    R0, R1                        ;
                                                     03  12 00042         BNEQ    3$                            ;
                                                 00E0    31 00044         BRW     9$                            ;
                                      51          08 A0  9A 00047 3$:     MOVZBL  8(R0), R1                     ; 1443
                                FE00  C8      08   A0      51  28 0004B   MOVC3   R1, 8(R0), ACE                 ; 1444
                                            FDCA   C8      04  B0 00052   MOVW    #4, ATR_ARGLIST+2             ; 1445
```

```
                                FDC8   C8      FE00    C8  9B 00057          MOVZBW  ACE, ATR_ARGLIST                    ; 1446
                                FDCC   C8      FE00    C8  9E 0005E          MOVAB   ACE, ATR_ARGLIST+4                  ; 1447
                                               FAA8    C8  9F 00065          PUSHAB  ACL_CONTEXT                         ; 1452
                                                       7E  7C 00069          CLRQ    -(SP)                               ;
                                               FDC8    C8  9F 0006B          PUSHAB  ATR_ARGLIST                         ;
                                                       56  DD 0006F          PUSHL   R6                                  ;
                                               F788    C8  9F 00071          PUSHAB  OBJECT_TYPE                         ;
                                               FAA4    C8  DD 00075          PUSHL   CHAN                                ;
                                          69           07  FB 00079          CALLS   #7, SYS$CHANGE_ACL                  ;
                                          57           50  D0 0007C          MOVL    R0, STATUS                          ;
                                          03           57  E9 0007F          BLBC    STATUS, 4$                          ; 1453
                                                     009C  31 00082          BRW     8$                                  ;
                                000009D0  8F           57  D1 00085 4$:      CMPL    STATUS, #2512                       ; 1456
                                                       28  13 0008C          BEQL    7$                                  ;
                                000009D8  8F           57  D1 0008E          CMPL    STATUS, #2520                       ; 1457
                                                       1F  13 00095          BEQL    7$                                  ;
                                                       7E  D4 00097 5$:      CLRL    -(SP)                               ; 1460
                                          7E           56  7D 00099          MOVQ    R6, -(SP)                           ;
                                                       01  DD 0009C          PUSHL   #1                                  ;
                                             007710D4  8F  DD 0009E          PUSHL   #7803092                            ;
                                          6A           05  FB 000A4          CALLS   #5, LIB$SIGNAL                      ;
                04     F780    C8                    03  00  ED 000A7          CMPZV   #0, #3, WORST_ERROR, #4             ;
                                                       03  19 000AE          BLSS    6$                                  ;
                                                     011B  31 000B0          BRW     15$                                 ;
                                                     010F  31 000B3 6$:      BRW     14$                                 ;
                                          50           68  D0 000B6 7$.      MOVL    ACE_POINTER, R0                     ; 1463
                                FDF8   C8        08    A0  9B 000B9          MOVZBW  8(R0), ACE_DESC                     ;
                                FDFC   C8        08    A0  9E 000BF          MOVAB   8(R0), ACE_DESC+4                   ; 1464
                                  04   A8      0C00    8F  B0 000C5          MOVW    #3072, ACE_TEXT_DESC                ; 1465
                                  08   A8        0C    A8  9E 000CB          MOVAB   ACE_TEXT, ACE_TEXT_DESC+4           ; 1466
                                                       7E  D4 000D0          CLRL    -(SP)                               ; 1472
                                  08   AE                04  D0 000D2          MOVL    #4, 8(SP)                           ;
                                            08     AE      04  9F 000D6          PUSHAB  8(SP)                               ;
                                             0000' CF      04  9F 000D9          PUSHAB  P.ACW                               ;
                                  0C   AE        50    8F  9A 000DD          MOVZBL  #80, 12(SP)                         ;
                                            0C     AE      04  9F 000E2          PUSHAB  12(SP)                              ;
                                            04     A8      04  9F 000E5          PUSHAB  ACE_TEXT_DESC                       ;
                                            04     A8      04  9F 000E8          PUSHAB  ACE_TEXT_DESC                       ;
                                               FDF8    C8  9F 000EB          PUSHAB  ACE_DESC                            ;
                                00000000G  00           07  FB 000EF          CALLS   #7, SYS$FORMAT_ACL                  ;
                                            04     A8      04  9F 000F6          PUSHAB  ACE_TEXT_DESC                       ; 1473
                                                       56  DD 000F9          PUSHL   R6                                  ;
                                                       02  DD 000FB          PUSHL   #2                                  ;
                                                       5B  DD 000FD          PUSHL   R11                                 ;
                                          6A           04  FB 000FF          CALLS   #4, LIB$SIGNAL                      ;
                                          50           6B  9E 00102          MOVAB   SET$_NOSUCHACE, R0                  ;
                                          19           50  E8 00105          BLBS    R0, 8$                              ;
                                          50 00000000*  00  9E 00108          MOVAB   <SET$_NOSUCHACE&7>, R0              ;
                50     F780    C8                    03  00  ED 0010F          CMPZV   #0, #3, WORST_ERROR, R0             ;
                                                       09  18 00116          BGEQ    8$                                  ;
                                F780   C8 00000000*    00  9E 00118          MOVAB   <SET$_NOSUCHACE!268435456>, WORST_ERROR ;
                                          78           98  D0 00121 8$:      MOVL    @ACE_POINTER, ACE_POINTER           ; 1475
                                                     FF10  31 00124          BRW     2$                                  ; 1440
                                          68     0C0C    C8  D0 00127 9$:      MOVL    OLD_ACE_HEAD, ACE_POINTER           ; 1480
                                          50           68  D0 0012C 10$:     MOVL    ACE_POINTER, R0                     ; 1481
                                          51     0C0C    C8  9E 0012F          MOVAB   OLD_ACE_HEAD, R1                    ;
                                          51           50  D1 00134          CMPL    R0, R1                              ;
```

```
                                     03  12 00137            BNEQ    11$                                    ;
                                   009A  31 00139            BRW     16$                                    ;
                 51            08    A0  9A 0013C  11$:      MOVZBL  8(R0), R1                              ; 1484
    FE00    C8         08    A0          51  28 00140            MOVC3   R1, 8(R0), ACE                         ; 1485
                     FDCA    C8          02  B0 00147            MOVW    #2, ATR_ARGLIST+2                      ; 1486
                     FDC8    C8    FE00  C8  9B 0014C            MOVZBW  ACE, ATR_ARGLIST                       ; 1487
                     FDCC    C8    FE00  C8  9E 00153            MOVAB   ACE, ATR_ARGLIST+4                     ; 1488
                                   FAA8  C8  9F 0015A            PUSHAB  ACL_CONTEXT                            ; 1493
                                         7E  7C 0015E            CLRQ    -(SP)                                  ;
                                   FDC8  C8  9F 00160            PUSHAB  ATR_ARGLIST                            ;
                                         56  DD 00164            PUSHL   R6                                     ;
                                   F788  C8  9F 00166            PUSHAB  OBJECT_TYPE                            ;
                                   FAA4  C8  DD 0016A            PUSHL   CHAN                                   ;
                             69          07  FB 0016E            CALLS   #7, SYS$CHANGE_ACL                     ;
                             57          50  D0 00171            MOVL    R0, STATUS                             ;
                             03          57  E8 00174            BLBS    STATUS, 12$                            ; 1494
                                       FF1D  31 00177            BRW     5$                                     ;
                             78          98  D0 0017A  12$:      MOVL    @ACE_POINTER, ACE_POINTER              ; 1500
                                         AD  11 0017D            BRB     10$                                    ; 1481
                     FDC8    C8 000600FF 8F  D0 0017F  13$:      MOVL    #393471, ATR_ARGLIST                   ; 1509
                     FDCC    C8    FE00  C8  9E 00188            MOVAB   ACE, ATR_ARGLIST+4                     ; 1510
                                   FAA8  C8  9F 0018F            PUSHAB  ACL_CONTEXT                            ; 1515
                                         7E  7C 00193            CLRQ    -(SP)                                  ;
                                   FDC8  C8  9F 00195            PUSHAB  ATR_ARGLIST                            ;
                                         56  DD 00199            PUSHL   R6                                     ;
                                   F788  C8  9F 0019B            PUSHAB  OBJECT_TYPE                            ;
                                   FAA4  C8  DD 0019F            PUSHL   CHAN                                   ;
                             69          07  FB 001A3            CALLS   #7, SYS$CHANGE_ACL                     ;
                             57          50  D0 001A6            MOVL    R0, STATUS                             ;
                             2A          57  E8 001A9            BLBS    STATUS, 16$                            ; 1516
                                         7E  D4 001AC            CLRL    -(SP)                                  ; 1519
                             7E          56  7D 001AE            MOVQ    R6, -(SP)                              ;
                                         01  DD 001B1            PUSHL   #1                                     ;
                               007710D4  8F  DD 001B3            PUSHL   #7803092                               ;
                             6A          05  FB 001B9            CALLS   #5, LIB$SIGNAL                         ;
        04    F780    C8     03          00  ED 001BC            CMPZV   #0, #3, WORST_ERROR, #4                ;
                                         09  18 001C3            BGEQ    15$                                    ;
                     F780    C8 107710D4 8F  D0 001C5  14$:      MOVL    #276238548, WORST_ERROR                ;
                             50 107710D4 8F  D0 001CE  15$:      MOVL    #276238548, R0                         ; 1520
                                             04 001D5            RET                                            ;
                             50          01  D0 001D6  16$:      MOVL    #1, R0                                 ; 1524
                                             04 001D9            RET                                            ; 1526

; Routine Size:  474 bytes,    Routine Base:  $CODE$ + 1060
```

```
;  1534          1527  1 ROUTINE REPLACE_ACL (OBJECT_NAME_DESC) =
;  1535          1528  1
;  1536          1529  1 !++
;  1537          1530  1 !
;  1538          1531  1 ! FUNCTIONAL DESCRIPTION:
;  1539          1532  1 !
;  1540          1533  1 !       This routine deletes the indicated ACEs, and then replaces them
;  1541          1534  1 !       with the new ones specified on the /REPLACE qualifier.
;  1542          1535  1 !
;  1543          1536  1 ! CALLING SEQUENCE:
;  1544          1537  1 !       ADD_ACL (ARG1)
;  1545          1538  1 !
;  1546          1539  1 ! INPUT PARAMETERS:
;  1547          1540  1 !       ARG1: address of the FAB
;  1548          1541  1 !
;  1549          1542  1 ! IMPLICIT INPUTS:
;  1550          1543  1 !       none
;  1551          1544  1 !
;  1552          1545  1 ! OUTPUT PARAMETERS:
;  1553          1546  1 !       none
;  1554          1547  1 !
;  1555          1548  1 ! IMPLICIT OUTPUTS:
;  1556          1549  1 !       none
;  1557          1550  1 !
;  1558          1551  1 ! ROUTINE VALUE:
;  1559          1552  1 !       1 if successful
;  1560          1553  1 !       error code otherwise
;  1561          1554  1 !
;  1562          1555  1 ! SIDE EFFECTS:
;  1563          1556  1 !       none
;  1564          1557  1 !
;  1565          1558  1 !--
;  1566          1559  1
;  1567          1560  2 BEGIN
;  1568          1561  2
;  1569          1562  2 LOCAL
;  1570          1563  2         OLD_ACLCTX,                                 ! Old ACL context
;  1571          1564  2         STATUS;                                     ! Local routine return status
;  1572          1565  2
;  1573          1566  2 ! Before deleting any of the given ACEs, make sure that they all exist and
;  1574          1567  2 ! the order is correct.
;  1575          1568  2
;  1576          1569  2 OLD_ACLCTX = 0;
;  1577          1570  2 ACE_POINTER = .OLD_ACE_HEAD[ACEQ_L_FLINK];
;  1578          1571  2 UNTIL .ACE_POINTER EQLA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  1579          1572  2 DO
;  1580          1573  3     BEGIN
;  1581          1574  3     CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1582          1575  3              ACE_POINTER[ACEQ_T_ACE], ACE);
;  1583          1576  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_FNDACLENT;
;  1584          1577  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1585          1578  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1586        P 1579  3     STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1587        P 1580  3                           OBJTYP = OBJECT_TYPE,
;  1588        P 1581  3                           OBJNAM = .OBJECT_NAME_DESC,
;  1589        P 1582  3                           ITMLST = ATR_ARGLIST,
;  1590          1583  3                           CONTXT = ACL_CONTEXT);
```

```
;  1591          1584  3     IF NOT .STATUS
;  1592          1585  3     THEN
;  1593          1586  4         BEGIN
;  1594          1587  4         IF .STATUS NEQ SS$_ACLEMPTY
;  1595          1588  4         AND .STATUS NEQ SS$_NOENTRY
;  1596          1589  4         THEN
;  1597          1590  5             BEGIN
;  1598          1591  5             SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1599          1592  5             RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1600          1593  4             END;
;  1601          1594  4         ACE_DESC[DSC$W_LENGTH] = .$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE];
;  1602          1595  4         ACE_DESC[DSC$A_POINTER] = ACE_POINTER[ACEQ_T_ACE];
;  1603          1596  4         ACE_TEXT_DESC[DSC$W_LENGTH] = 3072;
;  1604          1597  4         ACE_TEXT_DESC[DSC$A_POINTER] = ACE_TEXT;
;  1605        P 1598  4         $FORMAT_ACL (ACLENT = ACE_DESC,
;  1606        P 1599  4                      ACLLEN = ACE_TEXT_DESC[DSC$W_LENGTH],
;  1607        P 1600  4                      ACLSTR = ACE_TEXT_DESC,
;  1608        P 1601  4                      WIDTH = %REF (80),
;  1609        P 1602  4                      TRMDSC = $DESCRIPTOR (%CHAR (13), %CHAR (10)),
;  1610          1603  4                      INDENT = %REF (4));
;  1611          1604  4         SIGNAL (SET$_NOSUCHACE, 2, .OBJECT_NAME_DESC, ACE_TEXT_DESC);
;  1612          1605  4         RETURN SET$_NOSUCHACE OR STS$M_INHIB_MSG;
;  1613          1606  3         END;
;  1614          1607  3
;  1615          1608  3 ! The ACE exists.  Is the ordering correct?
;  1616          1609  3
;  1617          1610  3     IF .OLD_ACLCTX NEQ 0
;  1618          1611  3     THEN
;  1619          1612  4         BEGIN
;  1620          1613  4         IF .OLD_ACLCTX<0,24> + 1 NEQ .ACL_CONTEXT
;  1621          1614  4         THEN
;  1622          1615  5             BEGIN
;  1623          1616  5             SIGNAL (SET$_IVORDER, 1, .OBJECT_NAME_DESC);
;  1624          1617  5             RETURN SET$_IVORDER OR STS$M_INHIB_MSG;
;  1625          1618  4             END;
;  1626          1619  3         END;
;  1627          1620  3     OLD_ACLCTX = .ACL_CONTEXT;
;  1628          1621  3     ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1629          1622  2     END;
;  1630          1623  2
;  1631          1624  2 ! Delete any ACEs specified on the /ACL qualifier.
;  1632          1625  2
;  1633          1626  2 ACE_POINTER = .OLD_ACE_HEAD[ACEQ_L_FLINK];
;  1634          1627  2 UNTIL .ACE_POINTER EQLA OLD_ACE_HEAD[ACEQ_L_FLINK]
;  1635          1628  2 DO
;  1636          1629  3     BEGIN
;  1637          1630  3     CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1638          1631  3              ACE_POINTER[ACEQ_T_ACE], ACE);
;  1639          1632  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_DELACLENT;
;  1640          1633  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1641          1634  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1642        P 1635  3     STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1643        P 1636  3                           OBJTYP = OBJECT_TYPE,
;  1644        P 1637  3                           OBJNAM = .OBJECT_NAME_DESC,
;  1645        P 1638  3                           ITMLST = ATR_ARGLIST,
;  1646          1639  3                           CONTXT = ACL_CONTEXT);
;  1647          1640  3     IF NOT .STATUS
```

```
;  1648          1641  3     THEN
;  1649          1642  4         BEGIN
;  1650          1643  4         SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1651          1644  4         RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1652          1645  3         END;
;  1653          1646  3     IF .ACE[ACE$B_SIZE] EQL 0
;  1654          1647  3     THEN
;  1655          1648  4         BEGIN
;  1656          1649  4         IF .ACE[ACE$W_FLAGS] EQL SS$_ACLEMPTY
;  1657          1650  4         THEN EXITLOOP
;  1658          1651  4         ELSE
;  1659          1652  5             BEGIN
;  1660          1653  5             SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .ACE[ACE$W_FLAGS], 0);
;  1661          1654  5             RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1662          1655  4             END;
;  1663          1656  3         END;
;  1664          1657  3     ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1665          1658  2     END;
;  1666          1659  2
;  1667          1660  2 ! Add the new ACEs specified on the /REPLACE qualifier.
;  1668          1661  2
;  1669          1662  2 ACE_POINTER = .NEW_ACE_HEAD[ACEQ_L_FLINK];
;  1670          1663  2 UNTIL .ACE_POINTER EQLA NEW_ACE_HEAD[ACEQ_L_FLINK]
;  1671          1664  2 DO
;  1672          1665  3     BEGIN
;  1673          1666  3     CH$MOVE (.$BBLOCK[ACE_POINTER[ACEQ_T_ACE], ACE$B_SIZE],
;  1674          1667  3              ACE_POINTER[ACEQ_T_ACE], ACE);
;  1675          1668  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_ADDACLENT;
;  1676          1669  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1677          1670  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1678        P 1671  3     STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1679        P 1672  3                           OBJTYP = OBJECT_TYPE,
;  1680        P 1673  3                           OBJNAM = .OBJECT_NAME_DESC,
;  1681        P 1674  3                           ITMLST = ATR_ARGLIST,
;  1682          1675  3                           CONTXT = ACL_CONTEXT);
;  1683          1676  3     IF NOT .STATUS
;  1684          1677  3     THEN
;  1685          1678  4         BEGIN
;  1686          1679  4         SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1687          1680  4         RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1688          1681  3         END;
;  1689          1682  3     ACE_POINTER = .ACE_POINTER[ACEQ_L_FLINK];
;  1690          1683  2     END;
;  1691          1684  2
;  1692          1685  2 RETURN 1;
;  1693          1686  2
;  1694          1687  1 END;                                   ! End of routine REPLACE_ACL


                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

                                          0D  00360 P.ACZ:  .ASCII  <13>                    ;
                                          0A  00361         .ASCII  <10>                    ;
                                              00362         .BLKB   2
                                00000002  00364 P.ACY:  .LONG   2                           ;
                                00000000' 00368         .ADDRESS P.ACZ                      ;
```

```
                                                        .PSECT  $CODE$,NOWRT,2

                                                OFFC 00000 REPLACE_ACL:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11            ; 1527
                                    5B 00000000G   00   9E 00002        MOVAB   SYS$CHANGE_ACL, R11             ;
                                    5A 00000000G   00   9E 00009        MOVAB   LIB$SIGNAL, R10                 ;
                                    59      0000'  CF   9E 00010        MOVAB   ACE_POINTER, R9                 ;
                                    5E             08   C2 00015        SUBL2   #8, SP                          ;
                                                   58   D4 00018        CLRL    OLD_ACLCTX                      ; 1569
                                    69      0C0C   C9   D0 0001A        MOVL    OLD_ACE_HEAD, ACE_POINTER       ; 1570
                                    56        04   AC   D0 0001F        MOVL    OBJECT_NAME_DESC, R6            ; 1583
                                    50             69   D0 00023 1$:    MOVL    ACE_POINTER, R0                 ; 1571
                                    51      0C0C   C9   9E 00026        MOVAB   OLD_ACE_HEAD, R1                ;
                                    51             50   D1 0002B        CMPL    R0, R1                          ;
                                                   03   12 0002E        BNEQ    2$                              ;
                                                 0140   31 00030        BRW     12$                             ;
                                    51        08   A0   9A 00033 2$:    MOVZBL  8(R0), R1                       ; 1574
                FE00    C9        08   A0             51   28 00037    MOVC3   R1, 8(R0), ACE                  ; 1575
                              FDCA  C9                 04   B0 0003E    MOVW    #4, ATR_ARGLIST+2               ; 1576
                              FDC8  C9      FE00       C9   9B 00043    MOVZBW  ACE, ATR_ARGLIST                ; 1577
                              FDCC  C9      FE00       C9   9E 0004A    MOVAB   ACE, ATR_ARGLIST+4              ; 1578
                                            FAA8       C9   9F 00051    PUSHAB  ACL_CONTEXT                     ; 1583
                                                       7E   7C 00055    CLRQ    -(SP)                           ;
                                            FDC8       C9   9F 00057    PUSHAB  ATR_ARGLIST                     ;
                                                       56   DD 0005B    PUSHL   R6                              ;
                                            F788       C9   9F 0005D    PUSHAB  OBJECT_TYPE                     ;
                                            FAA4       C9   DD 00061    PUSHL   CHAN                            ;
                                    6B                 07   FB 00065    CALLS   #7, SYS$CHANGE_ACL              ;
                                    57                 50   D0 00068    MOVL    R0, STATUS                      ;
                                    03                 57   E9 0006B    BLBC    STATUS, 3$                      ; 1584
                                                     00AD   31 0006E    BRW     9$                              ;
                              000009D0  8F                 57   D1 00071 3$:    CMPL    STATUS, #2512           ; 1587
                                                           29   13 00078    BEQL    7$                          ;
                              000009D8  8F                 57   D1 0007A    CMPL    STATUS, #2520               ; 1588
                                                           20   13 00081    BEQL    7$                          ;
                                                           7E   D4 00083 4$:    CLRL    -(SP)                   ; 1591
                                                           57   DD 00085    PUSHL   STATUS                      ;
                                                           56   DD 00087 5$:    PUSHL   R6                      ;
                                                           01   DD 00089    PUSHL   #1                          ;
                                         007710D4          8F   DD 0008B    PUSHL   #7803092                    ;
                                    6A                     05   FB 00091    CALLS   #5, LIB$SIGNAL              ;
                04      F780    C9  03                     00   ED 00094    CMPZV   #0, #3, WORST_ERROR, #4     ;
                                                           03   19 0009B    BLSS    6$                          ;
                                                         01B0   31 0009D    BRW     19$                         ;
                                                         01A4   31 000A0 6$:    BRW     18$                     ;
                                    50                     69   D0 000A3 7$:    MOVL    ACE_POINTER, R0         ; 1594
                              FDF8  C9        08           A0   9B 000A6    MOVZBW  8(R0), ACE_DESC             ;
                              FDFC  C9        08           A0   9E 000AC    MOVAB   8(R0), ACE_DESC+4           ; 1595
                                04  A9      0C00           8F   B0 000B2    MOVW    #3072, ACE_TEXT_DESC        ; 1596
                                08  A9        0C           A9   9E 000B8    MOVAB   ACE_TEXT, ACE_TEXT_DESC+4   ; 1597
                                                           7E   D4 000BD    CLRL    -(SP)                       ; 1603
                                08  AE                     04   D0 000BF    MOVL    #4, 8(SP)                   ;
                                                     08    AE   9F 000C3    PUSHAB  8(SP)                       ;
                                                   0000'   CF   9F 000C6    PUSHAB  P.ACY                       ;
```

```
                        0C   AE        50   8F  9A 000CA          MOVZBL  #80, 12(SP)                                          ;
                                       0C   AE  9F 000CF          PUSHAB  12(SP)                                               ;
                                       04   A9  9F 000D2          PUSHAB  ACE_TEXT_DESC                                        ;
                                       04   A9  9F 000D5          PUSHAB  ACE_TEXT_DESC                                        ;
                                     FDF8   C9  9F 000D8          PUSHAB  ACE_DESC                                             ;
                  00000000G          00       07  FB 000DC          CALLS   #7, SYS$FORMAT_ACL                                   ;
                                       04   A9  9F 000E3          PUSHAB  ACE_TEXT_DESC                                        ; 1604
                                                56  DD 000E6          PUSHL   R6                                                   ;
                                                02  DD 000E8          PUSHL   #2                                                   ;
                             00000000G          00  9F 000EA          PUSHAB  SET$_NOSUCHACE                                       ;
                           6A                   04  FB 000F0          CALLS   #4, LIB$SIGNAL                                       ;
                           50  00000000G        00  9E 000F3          MOVAB   SET$_NOSUCHACE, R0                                   ;
                           19                   50  E8 000FA          BLBS    R0, 8$                                               ;
                           50  00000000*        00  9E 000FD          MOVAB   <SET$_NOSUCHACE&7>, R0                               ;
        50      F780    C9         03          00  ED 00104          CMPZV   #0, #3, WORST_ERROR, R0                              ;
                                               09  18 0010B          BGEQ    8$                                                   ;
                        F780    C9 00000000*   00  9E 0010D          MOVAB   <SET$_NOSUCHACE!268435456>, WORST_ERROR              ;
                           50  00000000*        00  9E 00116 8$:      MOVAB   <SET$_NOSUCHACE!268435456>, R0                       ; 1605
                                               04  00119            RET                                                          ;
                                               58  D5  0011E 9$:      TSTL    OLD_ACLCTX                                           ; 1610
                                               46  13  00120          BEQL    11$                                                  ;
        50              58         18          00  EF  00122          EXTZV   #0, #24, OLD_ACLCTX, R0                              ; 1613
                                               50  D6  00127          INCL    R0                                                   ;
                        FAA8    C9             50  D1  00129          CMPL    R0, ACL_CONTEXT                                      ;
                                               38  13  0012E          BEQL    11$                                                  ;
                                               56  DD  00130          PUSHL   R6                                                   ; 1616
                                               01  DD  00132          PUSHL   #1                                                   ;
                             00000000G          00  9F  00134          PUSHAB  SET$_IVORDER                                         ;
                           6A                   03  FB  0013A          CALLS   #3, LIB$SIGNAL                                       ;
                           50  00000000G        00  9E  0013D          MOVAB   SET$_IVORDER, R0                                     ;
                           19                   50  E8  00144          BLBS    R0, 10$                                              ;
                           50  00000000*        00  9E  00147          MOVAB   <SET$_IVORDER&7>, R0                                 ;
        50      F780    C9         03          00  ED  0014E          CMPZV   #0, #3, WORST_ERROR, R0                              ;
                                               09  18  00155          BGEQ    10$                                                  ;
                        F780    C9 00000000*   00  9E  00157          MOVAB   <SET$_IVORDER!268435456>, WORST_ERROR                ;
                           50  00000000*        00  9E  00160 10$:     MOVAB   <SET$_IVORDER!268435456>, R0                         ; 1617
                                               04  00167            RET                                                          ;
                           58      FAA8    C9     D0  00168 11$:     MOVL    ACL_CONTEXT, OLD_ACLCTX                              ; 1620
                           79                 99     D0  0016D          MOVL    @ACE_POINTER, ACE_POINTER                            ; 1621
                                           FEB0   31  00170          BRW     1$                                                   ; 1571
                           69      0C0C    C9     D0  00173 12$:     MOVL    OLD_ACE_HEAD, ACE_POINTER                            ; 1626
                           50                 69     D0  00178 13$:     MOVL    ACE_POINTER, R0                                      ; 1627
                           51      0C0C    C9     9E  0017B          MOVAB   OLD_ACE_HEAD, R1                                     ;
                           51                 50     D1  00180          CMPL    R0, R1                                               ;
                                              5C     13  00183          BEQL    16$                                                  ;
                           51        08       A0     9A  00185          MOVZBL  8(R0), R1                                            ; 1630
                    FE00    C9          08    A0         51  28  00189  MOVC3   R1, 8(R0), ACE                                     ; 1631
                                    FDCA    C9         02  B0  00190  MOVW    #2, ATR_ARGLIST+2                                    ; 1632
                                    FDC8    C9     FE00    C9  9B  00195  MOVZBW  ACE, ATR_ARGLIST                                     ; 1633
                                    FDCC    C9     FE00    C9  9E  0019C  MOVAB   ACE, ATR_ARGLIST+4                                   ; 1634
                                           FAA8    C9         9F  001A3  PUSHAB  ACL_CONTEXT                                          ; 1639
                                                              7E  7C  001A7  CLRQ    -(SP)                                            ;
                                           FDC8    C9         9F  001A9  PUSHAB  ATR_ARGLIST                                          ;
                                                              56  DD  001AD  PUSHL   R6                                               ;
                                           F788    C9         9F  001AF  PUSHAB  OBJECT_TYPE                                          ;
                                           FAA4    C9         DD  001B3  PUSHL   CHAN                                                 ;
                           6B                                 07  FB  001B7  CALLS   #7, SYS$CHANGE_ACL                              ;
```

```
                                57              50  D0 001BA          MOVL    R0, STATUS                                      ;
                                03              57  E8 001BD          BLBS    STATUS, 14$                                     ; 1640
                                              FEC0  31 001C0          BRW     4$                                              ;
                                          FE00  C9  95 001C3 14$:     TSTB    ACE                                             ; 1646
                                                13  12 001C7          BNEQ    15$                                             ;
                            09D0    8F    FE02  C9  B1 001C9          CMPW    ACE+2, #2512                                    ; 1649
                                                0F  13 001D0          BEQL    16$                                             ;
                                                7E  D4 001D2          CLRL    -(SP)                                           ; 1653
                                7E      FE02    C9  3C 001D4          MOVZWL  ACE+2, -(SP)                                    ;
                                              FEAB  31 001D9          BRW     5$                                              ;
                                79              99  D0 001DC 15$:     MOVL    @ACE_POINTER, ACE_POINTER                       ; 1657
                                                97  11 001DF          BRB     13$                                             ; 1627
                                69      0C14    C9  D0 001E1 16$:     MOVL    NEW_ACE_HEAD, ACE_POINTER                       ; 1662
                                50              69  D0 001E6 17$:     MOVL    ACE_POINTER, R0                                 ; 1663
                                51      0C14    C9  9E 001E9          MOVAB   NEW_ACE_HEAD, R1                                ;
                                51              50  D1 001EE          CMPL    R0, R1                                          ;
                                                6A  13 001F1          BEQL    21$                                             ;
                                51        08    A0  9A 001F3          MOVZBL  8(R0), R1                                       ; 1666
                    FE00    C9      08    A0      51  28 001F7          MOVC3   R1, 8(R0), ACE                                  ; 1667
                            FDCA    C9              01  B0 001FE          MOVW    #1, ATR_ARGLIST+2                               ; 1668
                            FDC8    C9      FE00    C9  9B 00203          MOVZBW  ACE, ATR_ARGLIST                                ; 1669
                            FDCC    C9      FE00    C9  9E 0020A          MOVAB   ACE, ATR_ARGLIST+4                              ; 1670
                                    FAA8    C9  9F 00211          PUSHAB  ACL_CONTEXT                                             ; 1675
                                            7E  7C 00215          CLRQ    -(SP)                                           ;
                                    FDC8    C9  9F 00217          PUSHAB  ATR_ARGLIST                                     ;
                                            56  DD 0021B          PUSHL   R6                                              ;
                                    F788    C9  9F 0021D          PUSHAB  OBJECT_TYPE                                     ;
                                    FAA4    C9  DD 00221          PUSHL   CHAN                                            ;
                                6B              07  FB 00225          CALLS   #7, SYS$CHANGE_ACL                              ;
                                57              50  D0 00228          MOVL    R0, STATUS                                      ;
                                2A              57  E8 0022B          BLBS    STATUS, 20$                                     ; 1676
                                                7E  D4 0022E          CLRL    -(SP)                                           ; 1679
                                7E              56  7D 00230          MOVQ    R6, -(SP)                                       ;
                                                01  DD 00233          PUSHL   #1                                              ;
                                    007710D4    8F  DD 00235          PUSHL   #7803092                                        ;
                                6A              05  FB 0023B          CALLS   #5, LIB$SIGNAL                                  ;
                    04      F780    C9          03  00  ED 0023E          CMPZV   #0, #3, WORST_ERROR, #4                         ;
                                                09  18 00245          BGEQ    19$                                             ;
                            F780    C9  107710D4    8F  D0 00247 18$:     MOVL    #276238548, WORST_ERROR                         ;
                                50      107710D4    8F  D0 00250 19$:     MOVL    #276238548, R0                                  ; 1680
                                                    04  00257          RET                                                     ;
                                79              99  D0 00258 20$:     MOVL    @ACE_POINTER, ACE_POINTER                       ; 1682
                                                89  11 0025B          BRB     17$                                             ; 1663
                                50              01  D0 0025D 21$:     MOVL    #1, R0                                          ; 1685
                                                    04  00260          RET                                                     ; 1687

; Routine Size:  609 bytes,    Routine Base:  $CODE$ + 123A
```

```
;  1696          1688  1 ROUTINE COPY_ACL (OBJECT_NAME_DESC) =
;  1697          1689  1
;  1698          1690  1 !++
;  1699          1691  1 !
;  1700          1692  1 ! FUNCTIONAL DESCRIPTION:
;  1701          1693  1 !
;  1702          1694  1 !       This routine is called to copy the ACL from the specified input object
;  1703          1695  1 !       to the selected output object.  It is also used to delete the ACL of
;  1704          1696  1 !       a object.
;  1705          1697  1 !
;  1706          1698  1 ! CALLING SEQUENCE:
;  1707          1699  1 !       COPY_ACL (ARG1)
;  1708          1700  1 !
;  1709          1701  1 ! INPUT PARAMETERS:
;  1710          1702  1 !       ARG1: address of the FAB
;  1711          1703  1 !
;  1712          1704  1 ! IMPLICIT INPUTS:
;  1713          1705  1 !       none
;  1714          1706  1 !
;  1715          1707  1 ! OUTPUT PARAMETERS:
;  1716          1708  1 !       none
;  1717          1709  1 !
;  1718          1710  1 ! IMPLICIT OUTPUTS:
;  1719          1711  1 !       none
;  1720          1712  1 !
;  1721          1713  1 ! ROUTINE VALUE:
;  1722          1714  1 !       1 if successful
;  1723          1715  1 !       error code otherwise
;  1724          1716  1 !
;  1725          1717  1 ! SIDE EFFECTS:
;  1726          1718  1 !       The ACL is copied from one object to another.
;  1727          1719  1 !
;  1728          1720  1 !--
;  1729          1721  1
;  1730          1722  2 BEGIN
;  1731          1723  2
;  1732          1724  2 LOCAL
;  1733          1725  2         DEVICE_DESC     : $BBLOCK [DSC$C_S_BLN],        ! Device name descr
;  1734          1726  2         DEVICE          : $BBLOCK [NAM$C_DVI],  ! Device name storage
;  1735          1727  2         OBJECT_FIB_DESC : $BBLOCK [DSC$C_S_BLN],        ! Object's FIB descr
;  1736          1728  2         OBJECT_FIB      : $BBLOCK [FIB$C_LENGTH],       ! Object's FIB
;  1737          1729  2         STATUS;                                         ! Local routine return status
;  1738          1730  2
;  1739          1731  2 ! Delete any ACL that currently exists on the object.
;  1740          1732  2
;  1741          1733  2 ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_DELETEACL;
;  1742          1734  2 ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_DELETEACL;
;  1743          1735  2 ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1744        P 1736  2 STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1745        P 1737  2                       OBJTYP = OBJECT_TYPE,
;  1746        P 1738  2                       OBJNAM = .OBJECT_NAME_DESC,
;  1747        P 1739  2                       ITMLST = ATR_ARGLIST,
;  1748          1740  2                       CONTXT = ACL_CONTEXT);
;  1749          1741  2 IF NOT .STATUS
;  1750          1742  2 THEN
;  1751          1743  3     BEGIN
;  1752          1744  3     SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
```

```
;  1753          1745  3     RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1754          1746  2     END;
;  1755          1747  2
;  1756          1748  2 ! Now that the input and output objects are open, copy the ACL if necessary.
;  1757          1749  2
;  1758          1750  2 SACL_CONTEXT = 0;
;  1759          1751  2
;  1760          1752  2 WHILE 1
;  1761          1753  2 DO
;  1762          1754  3     BEGIN
;  1763          1755  3     ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_READACE;
;  1764          1756  3     ATR_ARGLIST[0, ITM$W_BUFSIZ] = ACL$S_READACE;
;  1765          1757  3     ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1766        P 1758  3     STATUS = $CHANGE_ACL (CHAN = .SCHAN,
;  1767        P 1759  3                           OBJTYP = SOBJECT_TYPE,
;  1768        P 1760  3                           OBJNAM = SOBJECT_DESC,
;  1769        P 1761  3                           ITMLST = ATR_ARGLIST,
;  1770          1762  3                           CONTXT = SACL_CONTEXT);
;  1771          1763  3     IF NOT .STATUS
;  1772          1764  3     THEN
;  1773          1765  4         BEGIN
;  1774          1766  4
;  1775          1767  4 ! Check for the end of the ACL.
;  1776          1768  4
;  1777          1769  4         IF .STATUS EQL SS$_ACLEMPTY OR .STATUS EQL SS$_NOMOREACE THEN EXITLOOP;
;  1778          1770  4
;  1779          1771  4 ! Not the end, return the error.
;  1780          1772  4
;  1781          1773  4         SIGNAL (SET$_READERR, 1, .SOBJECT_DESC, .STATUS, 0);
;  1782          1774  4         RETURN SET$_READERR OR STS$M_INHIB_MSG;
;  1783          1775  3         END;
;  1784          1776  3
;  1785          1777  3 ! If possible, copy the ACE to the target object.
;  1786          1778  3
;  1787          1779  3     IF NOT .ACE[ACE$V_NOPROPAGATE]
;  1788          1780  4     AND (IF .FLAGS[QUAL_DEFAULT]
;  1789          1781  4          THEN .ACE[ACE$V_DEFAULT] OR .FLAGS[DIRECTORY]
;  1790          1782  4          ELSE NOT .ACE[ACE$V_HIDDEN])
;  1791          1783  3     THEN
;  1792          1784  4         BEGIN
;  1793          1785  4
;  1794          1786  4 ! If this is a default ACE and the target is not a directory file, clear the
;  1795          1787  4 ! default option in the ACE.
;  1796          1788  4
;  1797          1789  4         IF .FLAGS[QUAL_DEFAULT]
;  1798          1790  4         THEN IF .ACE[ACE$V_DEFAULT]
;  1799          1791  4              AND NOT .FLAGS[DIRECTORY]
;  1800          1792  4              THEN ACE[ACE$V_DEFAULT] = 0;
;  1801          1793  4
;  1802          1794  4 ! Now add the ACE to the object's ACL.
;  1803          1795  4
;  1804          1796  4         ACL_CONTEXT = -1;
;  1805          1797  4         ATR_ARGLIST[0, ITM$W_ITMCOD] = ACL$C_ADDACLENT;
;  1806          1798  4         ATR_ARGLIST[0, ITM$W_BUFSIZ] = .ACE[ACE$B_SIZE];
;  1807          1799  4         ATR_ARGLIST[0, ITM$L_BUFADR] = ACE;
;  1808        P 1800  4         STATUS = $CHANGE_ACL (CHAN = .CHAN,
;  1809        P 1801  4                               OBJTYP = OBJECT_TYPE,
```

```
;  1810        P 1802  4                       OBJNAM = .OBJECT_NAME_DESC,
;  1811        P 1803  4                       ITMLST = ATR_ARGLIST,
;  1812          1804  4                       CONTXT = ACL_CONTEXT);
;  1813          1805  4           IF NOT .STATUS
;  1814          1806  4           THEN
;  1815          1807  5               BEGIN
;  1816          1808  5               SIGNAL (SET$_WRITEERR, 1, .OBJECT_NAME_DESC, .STATUS, 0);
;  1817          1809  5               RETURN SET$_WRITEERR OR STS$M_INHIB_MSG;
;  1818          1810  4               END;
;  1819          1811  3           END;
;  1820          1812  2       END;
;  1821          1813  2
;  1822          1814  2 ! Now that the ACL has been copied, return to clean things up.
;  1823          1815  2
;  1824          1816  2 RETURN 1;
;  1825          1817  2
;  1826          1818  1 END;                                   ! End of routine COPY_ACL


                                        003C 00000 COPY_ACL:
                                                    .WORD   Save R2,R3,R4,R5                ; 1688
                        55 00000000G   00    9E 00002       MOVAB   LIB$SIGNAL, R5          ;
                        54 00000000G   00    9E 00009       MOVAB   SYS$CHANGE_ACL, R4      ;
                        53      0000'  CF    9E 00010       MOVAB   ATR_ARGLIST, R3         ;
                        5E        A0   AE    9E 00015       MOVAB   -96(SP), SP             ;
                        63 000600FF    8F    D0 00019       MOVL    #393471, ATR_ARGLIST    ; 1734
                     04 A3        38   A3    9E 00020       MOVAB   ACE, ATR_ARGLIST+4      ; 1735
                              FCE0     C3    9F 00025       PUSHAB  ACL_CONTEXT             ; 1740
                                       7E    7C 00029       CLRQ    -(SP)                   ;
                                       53    DD 0002B       PUSHL   R3                      ;
                                 04    AC    DD 0002D       PUSHL   OBJECT_NAME_DESC        ;
                              F9C0     C3    9F 00030       PUSHAB  OBJECT_TYPE             ;
                              FCDC     C3    DD 00034       PUSHL   CHAN                    ;
                           64         07    FB 00038       CALLS   #7, SYS$CHANGE_ACL      ;
                           52         50    D0 0003B       MOVL    R0, STATUS              ;
                           21         52    E8 0003E       BLBS    STATUS, 2$              ; 1741
                                      7E    D4 00041       CLRL    -(SP)                   ; 1744
                                      52    DD 00043       PUSHL   STATUS                  ;
                                 04    AC    DD 00045       PUSHL   OBJECT_NAME_DESC        ;
                                       01    DD 00048       PUSHL   #1                      ;
                           007710D4    8F    DD 0004A       PUSHL   #7803092                ;
                           65          05    FB 00050       CALLS   #5, LIB$SIGNAL          ;
04      F9B8    C3         03          00    ED 00053       CMPZV   #0, #3, WORST_ERROR, #4 ;
                                       03    19 0005A       BLSS    1$                      ;
                                 00F4  31    11 0005C       BRW     13$                     ;
                                 00E8  31    11 0005F 1$:   BRW     12$                     ;
                                 A8    A3    D4 00062 2$:   CLRL    SACL_CONTEXT            ; 1750
                           63 000900FF    8F    D0 00065 3$:   MOVL    #590079, ATR_ARGLIST    ; 1756
                        04 A3        38   A3    9E 0006C       MOVAB   ACE, ATR_ARGLIST+4      ; 1757
                                 A8    A3    9F 00071       PUSHAB  SACL_CONTEXT            ; 1762
                                       7E    7C 00074       CLRQ    -(SP)                   ;
                                       53    DD 00076       PUSHL   R3                      ;
                                 FCEC  C3    9F 00078       PUSHAB  SOBJECT_DESC            ;
                                 FCE8  C3    9F 0007C       PUSHAB  SOBJECT_TYPE            ;
```

```
                                                A4   A3   DD 00080          PUSHL   SCHAN                                    ;
                                             64      07   FB 00083          CALLS   #7, SYS$CHANGE_ACL                       ;
                                             52      50   D0 00086          MOVL    R0, STATUS                               ;
                                             42      52   E8 00089          BLBS    STATUS, 7$                               ; 1763
                                 000009D0    8F      52   D1 0008C          CMPL    STATUS, #2512                            ; 1769
                                                     07   13 00093          BEQL    4$                                       ;
                                 000009E0    8F      52   D1 00095          CMPL    STATUS, #2528                            ;
                                                     03   12 0009C 4$:       BNEQ    5$                                       ;
                                                  00BA   31 0009E          BRW     14$                                      ;
                                                     7E   D4 000A1 5$:       CLRL    -(SP)                                    ; 1773
                                                     52   DD 000A3          PUSHL   STATUS                                   ;
                                                FCEC C3   DD 000A5          PUSHL   SOBJECT_DESC                             ;
                                                     01   DD 000A9          PUSHL   #1                                       ;
                                             007710B4    8F   DD 000AB          PUSHL   #7803060                             ;
                                             65          05   FB 000B1          CALLS   #5, LIB$SIGNAL                       ;
04      F9B8    C3                           03          00   ED 000B4          CMPZV   #0, #3, WORST_ERROR, #4              ;
                                                         09   18 000BB          BGEQ    6$                                   ;
                                      F9B8   C3 107710B4 8F   D0 000BD          MOVL    #276238516, WORST_ERROR              ;
                                             50 107710B4 8F   D0 000C6 6$:       MOVL    #276238516, R0                       ; 1774
                                                         04   000CD          RET                                          ;
                                92       3B   A3          03   E0 000CE 7$:       BBS     #3, ACE+3, 3$                        ; 1779
                                0C      F9B4  C3          06   E1 000D3          BBC     #6, FLAGS, 8$                        ; 1780
                                             10      3B   A3   E8 000D9          BLBS    ACE+3, 10$                           ; 1781
                                82      F9B5  C3          02   E1 000DD          BBC     #2, FLAGS+1, 3$                      ;
                                                         08   11 000E3          BRB     10$                                  ;
                                03       3B   A3          02   E1 000E5 8$:       BBC     #2, ACE+3, 10$                       ; 1782
                                                      FF78   31 000EA 9$:       BRW     3$                                   ;
                                0E      F9B4  C3          06   E1 000ED 10$:      BBC     #6, FLAGS, 11$                       ; 1789
                                             0A      3B   A3   E9 000F3          BLBC    ACE+3, 11$                           ; 1790
                                04      F9B5  C3          02   E0 000F7          BBS     #2, FLAGS+1, 11$                     ; 1791
                                         3B   A3          01   8A 000FD          BICB2   #1, ACE+3                            ; 1792
                                      FCE0   C3          01   CE 00101 11$:      MNEGL   #1, ACL_CONTEXT                      ; 1796
                                         02   A3          01   B0 00106          MOVW    #1, ATR_ARGLIST+2                    ; 1797
                                             63      38   A3   9B 0010A          MOVZBW  ACE, ATR_ARGLIST                     ; 1798
                                         04  A3      38   A3   9E 0010E          MOVAB   ACE, ATR_ARGLIST+4                   ; 1799
                                                FCE0 C3   9F 00113          PUSHAB  ACL_CONTEXT                              ; 1804
                                                     7E   7C 00117          CLRQ    -(SP)                                    ;
                                                     53   DD 00119          PUSHL   R3                                       ;
                                                   04     AC   DD 0011B          PUSHL   OBJECT_NAME_DESC                     ;
                                                F9C0 C3   9F 0011E          PUSHAB  OBJECT_TYPE                              ;
                                                FCDC C3   DD 00122          PUSHL   CHAN                                     ;
                                             64          07   FB 00126          CALLS   #7, SYS$CHANGE_ACL                   ;
                                             52          50   D0 00129          MOVL    R0, STATUS                           ;
                                             BB          52   E8 0012C          BLBS    STATUS, 9$                           ; 1805
                                                         7E   D4 0012F          CLRL    -(SP)                                ; 1808
                                                         52   DD 00131          PUSHL   STATUS                               ;
                                                   04     AC   DD 00133          PUSHL   OBJECT_NAME_DESC                     ;
                                                         01   DD 00136          PUSHL   #1                                   ;
                                             007710D4    8F   DD 00138          PUSHL   #7803092                             ;
                                             65          05   FB 0013E          CALLS   #5, LIB$SIGNAL                       ;
04      F9B8    C3                           03          00   ED 00141          CMPZV   #0, #3, WORST_ERROR, #4              ;
                                                         09   18 00148          BGEQ    13$                                  ;
                                      F9B8   C3 107710D4 8F   D0 0014A 12$:      MOVL    #276238548, WORST_ERROR              ;
                                             50 107710D4 8F   D0 00153 13$:      MOVL    #276238548, R0                       ; 1809
                                                         04   0015A          RET                                          ;
                                             50          01   D0 0015B 14$:      MOVL    #1, R0                               ; 1816
                                                         04   0015E          RET                                          ; 1818
```

; Routine Size: 351 bytes,   Routine Base: $CODE$ + 149B

```
;  1828          1819  1 ROUTINE INPUT_ERROR (FILE_FAB) =
;  1829          1820  1
;  1830          1821  1 !++
;  1831          1822  1 !
;  1832          1823  1 ! FUNCTIONAL DESCRIPTION:
;  1833          1824  1 !
;  1834          1825  1 !       This routine is used to signal errors received on the file scan.
;  1835          1826  1 !
;  1836          1827  1 ! CALLING SEQUENCE:
;  1837          1828  1 !       INPUT_ERROR (ARG1)
;  1838          1829  1 !
;  1839          1830  1 ! INPUT PARAMETERS:
;  1840          1831  1 !       ARG1: address of the FAB
;  1841          1832  1 !
;  1842          1833  1 ! IMPLICIT INPUTS:
;  1843          1834  1 !       none
;  1844          1835  1 !
;  1845          1836  1 ! OUTPUT PARAMETERS:
;  1846          1837  1 !       none
;  1847          1838  1 !
;  1848          1839  1 ! IMPLICIT OUTPUTS:
;  1849          1840  1 !       none
;  1850          1841  1 !
;  1851          1842  1 ! ROUTINE VALUE:
;  1852          1843  1 !       1
;  1853          1844  1 !
;  1854          1845  1 ! SIDE EFFECTS:
;  1855          1846  1 !       The error is signaled by placing the appropriate message into
;  1856          1847  1 !       the output file.
;  1857          1848  1 !
;  1858          1849  1 !--
;  1859          1850  1
;  1860          1851  2 BEGIN
;  1861          1852  2
;  1862          1853  2 MAP
;  1863          1854  2         FILE_FAB        : REF $BBLOCK;          ! FAB address
;  1864          1855  2
;  1865          1856  2 LOCAL
;  1866          1857  2         STATUS;                                 ! Error to signal;
;  1867          1858  2
;  1868          1859  2 STATUS = SET$_OPENOUT;
;  1869          1860  2 IF .FILE_FAB[FAB$L_STS] EQL RMS$_FNF
;  1870          1861  2 THEN STATUS = SET$_OPENOUT AND NOT STS$M_SEVERITY OR STS$K_WARNING;
;  1871          1862  2
;  1872          1863  2 FILE_ERROR (.STATUS, .FILE_FAB, .FILE_FAB[FAB$L_STS],
;  1873          1864  2                                 .FILE_FAB[FAB$L_STV]);
;  1874          1865  2
;  1875          1866  2 RETURN 1;
;  1876          1867  2
;  1877          1868  1 END;                                             ! End of routine INPUT_ERROR


                                0000 00000 INPUT_ERROR:
                                                .WORD   Save nothing                    ; 1819
```

```
                    51 007710A2   8F   D0 00002          MOVL    #7803042, STATUS                        ; 1859
                    50         04   AC   D0 00009          MOVL    FILE_FAB, R0                            ; 1860
          00018292  8F         08   A0   D1 0000D          CMPL    8(R0), #98962                           ;
                                    07   12 00015          BNEQ    1$                                      ;
                    51 007710A0   8F   D0 00017          MOVL    #7803040, STATUS                        ; 1861
                    7E         08   A0   7D 0001E 1$:     MOVQ    8(R0), -(SP)                            ; 1863
                                    50   DD 00022          PUSHL   R0                                      ;
                                    51   DD 00024          PUSHL   STATUS                                  ;
              0000V CF                  04   FB 00026          CALLS   #4, FILE_ERROR                          ;
                    50                  01   D0 0002B          MOVL    #1, R0                                  ; 1866
                                        04   0002E          RET                                             ; 1868

; Routine Size:  47 bytes,    Routine Base:  $CODE$ + 15FA
```

```
;  1879          1869  1 ROUTINE FILE_ERROR (ERROR_CODE, FILE_FAB, STS, STV) =
;  1880          1870  1
;  1881          1871  1 !++
;  1882          1872  1 !
;  1883          1873  1 ! FUNCTIONAL DESCRIPTION:
;  1884          1874  1 !
;  1885          1875  1 !       This routine is used to signal errors received on files.
;  1886          1876  1 !
;  1887          1877  1 ! CALLING SEQUENCE:
;  1888          1878  1 !       FILE_ERROR (ARG1, ARG2, ARG3, ARG4)
;  1889          1879  1 !
;  1890          1880  1 ! INPUT PARAMETERS:
;  1891          1881  1 !       ARG1: error code
;  1892          1882  1 !       ARG2: address of the FAB
;  1893          1883  1 !       ARG3: primary error status
;  1894          1884  1 !       ARG4: secondary error status
;  1895          1885  1 !
;  1896          1886  1 ! IMPLICIT INPUTS:
;  1897          1887  1 !       none
;  1898          1888  1 !
;  1899          1889  1 ! OUTPUT PARAMETERS:
;  1900          1890  1 !       none
;  1901          1891  1 !
;  1902          1892  1 ! IMPLICIT OUTPUTS:
;  1903          1893  1 !       none
;  1904          1894  1 !
;  1905          1895  1 ! ROUTINE VALUE:
;  1906          1896  1 !       1
;  1907          1897  1 !
;  1908          1898  1 ! SIDE EFFECTS:
;  1909          1899  1 !       none
;  1910          1900  1 !
;  1911          1901  1 !--
;  1912          1902  1
;  1913          1903  2 BEGIN
;  1914          1904  2
;  1915          1905  2 MAP
;  1916          1906  2         FILE_FAB        : REF $BBLOCK;          ! FAB address
;  1917          1907  2
;  1918          1908  2 BIND
;  1919          1909  2         FILE_NAM        = .FILE_FAB[FAB$L_NAM]  : $BBLOCK;      ! NAMe block address
;  1920          1910  2
;  1921          1911  2 LOCAL
;  1922          1912  2         FILE_NAME       : $BBLOCK [DSC$C_S_BLN];        ! Local file name descr
;  1923          1913  2
;  1924          1914  2 CH$FILL (0, DSC$C_S_BLN, FILE_NAME);
;  1925          1915  2 IF .FILE_NAM[NAM$B_RSL] NEQ 0
;  1926          1916  2 THEN
;  1927          1917  3     BEGIN
;  1928          1918  3     FILE_NAME[DSC$W_LENGTH] = .FILE_NAM[NAM$B_RSL];
;  1929          1919  3     FILE_NAME[DSC$A_POINTER] = .FILE_NAM[NAM$C_RSA];
;  1930          1920  3     END
;  1931          1921  2 ELSE IF .FILE_NAM[NAM$B_ESL] NEQ 0
;  1932          1922  2 THEN
;  1933          1923  3     BEGIN
;  1934          1924  3     FILE_NAME[DSC$W_LENGTH] = .FILE_NAM[NAM$B_ESL];
;  1935          1925  3     FILE_NAME[DSC$A_POINTER] = .FILE_NAM[NAM$C_ESA];
```

```
;  1936          1926  3      END
;  1937          1927  2 ELSE
;  1938          1928  3      BEGIN
;  1939          1929  3      FILE_NAME[DSC$W_LENGTH] = .FILE_FAB[FAB$B_FNS];
;  1940          1930  3      FILE_NAME[DSC$A_POINTER] = .FILE_FAB[FAB$L_FNA];
;  1941          1931  2      END;
;  1942          1932  2
;  1943          1933  2 SIGNAL (.ERROR_CODE, 1, FILE_NAME, .STS, .STV);
;  1944          1934  2
;  1945          1935  2 RETURN 1;
;  1946          1936  2
;  1947          1937  1 END;                                    ! End of routine FILE_ERROR


                                        00FC 00000 FILE_ERROR:
                                                   .WORD   Save R2,R3,R4,R5,R6,R7                          ; 1869
                                5E          08  C2 00002       SUBL2   #8, SP                              ;
                                57      08  AC  D0 00005       MOVL    FILE_FAB, R7                        ; 1909
                                56      28  A7  D0 00009       MOVL    40(R7), R6                          ;
        08          00          6E          00  2C 0000D       MOVC5   #0, (SP), #0, #8, FILE_NAME         ; 1914
                                                6E 00012       ;                                           ;
                                        03  A6  95 00013       TSTB    3(R6)                               ; 1915
                                        0B  13 00016           BEQL    1$                                  ;
                                6E      03  A6  9B 00018       MOVZBW  3(R6), FILE_NAME                    ; 1918
                            04  AE      04  A6  D0 0001C       MOVL    4(R6), FILE_NAME+4                  ; 1919
                                        19  11 00021           BRB     3$                                  ; 1915
                                        0B  A6  95 00023 1$:   TSTB    11(R6)                              ; 1921
                                        0B  13 00026           BEQL    2$                                  ;
                                6E      0B  A6  9B 00028       MOVZBW  11(R6), FILE_NAME                   ; 1924
                            04  AE      0C  A6  D0 0002C       MOVL    12(R6), FILE_NAME+4                 ; 1925
                                        09  11 00031           BRB     3$                                  ; 1921
                                6E      34  A7  9B 00033 2$:   MOVZBW  52(R7), FILE_NAME                   ; 1929
                            04  AE      2C  A7  D0 00037       MOVL    44(R7), FILE_NAME+4                 ; 1930
                                7E      0C  AC  7D 0003C 3$:   MOVQ    STS, -(SP)                          ; 1933
                                        08  AE  9F 00040       PUSHAB  FILE_NAME                           ;
                                        01  DD 00043          PUSHL   #1                                  ;
                                        04  AC  DD 00045      PUSHL   ERROR_CODE                          ;
                        00000000G       00      FB 00048       CALLS   #5, LIB$SIGNAL                      ;
                                1A      04  AC  E8 0004F       BLBS    ERROR_CODE, 4$                      ;
        50      04  AC              03      00  EF 00053       EXTZV   #0, #3, ERROR_CODE, R0              ;
        50  0000'  CF              03          00  ED 00059    CMPZV   #0, #3, WORST_ERROR, R0             ;
                                            0B  18 00060       BGEQ    4$                                  ;
            0000'  CF      04  AC 10000000  8F  C9 00062       BISL3   #268435456, ERROR_CODE, WORST_ERROR ;
                           50               01  D0 0006D 4$:   MOVL    #1, R0                              ; 1935
                                            04  00070         RET                                         ; 1937

; Routine Size:  113 bytes,    Routine Base:  $CODE$ + 1629

;  1948          1938  1
;  1949          1939  1 END
;  1950          1940  0 ELUDOM
```

```
;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $OWN$                         5296  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  _LIB$KEY0$                       0  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(1)
;  _LIB$STATE$                     14  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(1)
;  $PLIT$                         876  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                        5786  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       189         1        1000        00:01.9
;  _$255$DUA28:[SYSLIB]TPAMAC.L32;1        42        15        35          14        00:00.2


;                                   COMMAND QUALIFIERS

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

; Size:          5786 code + 6186 data bytes
; Run Time:         01:37.2
; Elapsed Time:     04:37.6
; Lines/CPU Min:     1197
; Lexemes/CPU-Min: 27511
; Memory Used:  578 pages
; Compilation Complete
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

AEDMESSAG
LIS

AEDPROMPT
LIS

SETACL
LIS

AEDSUBR
LIS

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SETSHOAL
CLS

OBJEXEREQ
REQ

EXEFIXUP
CLS

ANALYZRMS
MAP

SHOWACL
CLS

EXESTUFF
CLS

ANALYZ

EXEINPUT
CLS

ANALYZOBJ
MAP

EXEDRIVE
CLS

RMSREQ
REQ