F 1

```
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
```

FILEID**COPYCLI

```
  CCCCCCCC    000000    PPPPPPPP    YY      YY    CCCCCCCC  LL            IIIIII
  CCCCCCCC    000000    PPPPPPPP    YY      YY    CCCCCCCC  LL            IIIIII
CC            00    00  PP      PP  YY      YY  CC          LL              II
CC            00    00  PP      PP  YY      YY  CC          LL              II
CC            00    00  PP      PP    YY  YY    CC          LL              II
CC            00    00  PP      PP    YY  YY    CC          LL              II
CC            00    00  PPPPPPPP        YY      CC          LL              II
CC            00    00  PPPPPPPP        YY      CC          LL              II
CC            00    00  PP              YY      CC          LL              II
CC            00    00  PP              YY      CC          LL              II
CC            00    00  PP              YY      CC          LL              II          ....
  CCCCCCCC    000000    PP              YY        CCCCCCCC  LLLLLLLLLL    IIIIII        ....
  CCCCCCCC    000000    PP              YY        CCCCCCCC  LLLLLLLLLL    IIIIII        ....

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

```
;     1          0001  0 MODULE copycli (    ! Declarations of CLI data structures for the COPY command
;     2          0002  0                      LANGUAGE (BLISS32),
;     3          0003  0                      IDENT = 'V04-000'
;     4          0004  0                      ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !*****************************************************************************
;     9          0009  1 !*                                                                           *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                      *
;    13          0013  1 !*                                                                           *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    19          0019  1 !*  TRANSFERRED.                                                              *
;    20          0020  1 !*                                                                           *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    23          0023  1 !*  CORPORATION.                                                              *
;    24          0024  1 !*                                                                           *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*                                                                           *
;    29          0029  1 !*****************************************************************************
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:     COPY
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       This module contains all the routines for processing the COPY command
;    37          0037  1 !       qualifiers.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:
;    40          0040  1 !
;    41          0041  1 !       VAX/VMS operating system, unprivileged user mode utility,
;    42          0042  1 !       operates at non-AST level.
;    43          0043  1 !
;    44          0044  1 !--
;    45          0045  1 !++
;    46          0046  1 !
;    47          0047  1 ! AUTHOR:       Carol Peters,   CREATION DATE:  28 April 1978 07:36
;    48          0048  1 !
;    49          0049  1 ! REVISION HISTORY:
;    50          0050  1 !
;    51          0051  1 !       V3-003  TSK0003         Tamar Krichevsky        9-FEB-1984
;    52          0052  1 !               Change addressing mode for LIB$CVT_DTB and LIB$LOOKUP_KEY
;    53          0053  1 !               to general.
;    54          0054  1 !
;    55          0055  1 !       V3-002  TSK0002         Tamar Krichevsky        10-Aug-1983
;    56          0056  1 !               Fix default for /PROTECTION qualifier so that if fields which
;    57          0057  1 !               have not been specified are left alone.
```

```
;    58          0058  1 !
;    59          0059  1 !       V3-001  TSK0001         Tamar Krichevsky        18-Jan-1983
;    60          0060  1 !               Rework whole module.  Change Command Language Interface over
;    61          0061  1 !               to new CLI.  Create two global routines: COPY$GET_GLOBAL_QUAL
;    62          0062  1 !               and COPY$GET_LOCAL_QUAL.  These routines simulate parts of the
;    63          0063  1 !               CLI so that COPY/QUAL a,b/NOQUAL,c * and COPY a,b/NOQUAL,c */QUAL
;    64          0064  1 !               behave the same.
;    65          0065  1 !
;    66          0066  1 !               Add the common qualifiers (/BEFORE, /SINCE, /CREATED, /MODIFIED
;    67          0067  1 !               /BACKUP, /EXPIRED, /EXCLUDE, /BY_OWNER AND /CONFIRM).
;    68          0068  1 !
;    69          0069  1 !       003     TMH0003         T. Halvorsen    17-Nov-1979
;    70          0070  1 !               Add cleanup2_desc for output parameter cleanup call.
;    71          0071  1 !
;    72          0072  1 !       002     TMH0002         T. Halvorsen    25-Jul-1979
;    73          0073  1 !               Add relative volume placement control
;    74          0074  1 !--
```

```
;    76          0075  1 !
;    77          0076  1 ! Table of contents
;    78          0077  1 !
;    79          0078  1
;    80          0079  1 FORWARD ROUTINE
;    81          0080  1     copy$get_global_qual: NOVALUE,                  ! Get global command qualifiers
;    82          0081  1     copy$get_local_qual : NOVALUE,                  ! Get local command qualifiers
;    83          0082  1     protection_parse    : NOVALUE,                  ! Parse routine for /PROTECTION qualifier
;    84          0083  1     parse_protection_value : NOVALUE;               ! Parse the /PROTECTION keyword values (RWED)
;    85          0084  1
;    86          0085  1 !
;    87          0086  1 ! Include files
;    88          0087  1 !
;    89          0088  1
;    90          0089  1 LIBRARY 'SYS$LIBRARY:STARLET.L32';                  ! Common system definitions
;    91          0090  1 REQUIRE 'SRC$:COPYMSG.REQ';                         ! Put message macros
;    92          0171  1
;    93          0172  1 !
;    94          0173  1 ! Literals
;    95          0174  1 !
;    96          0175  1
;    97          0176  1 BIND
;    98          0177  1
;    99          0178  1     ! Descriptors for the qualifier names, used while parsing the command line.
;   100          0179  1     !
;   101          0180  1     verb_desc           = $DESCRIPTOR('$VERB'),
;   102          0181  1     log_msg_desc        = $DESCRIPTOR('LOG'),
;   103          0182  1     concatenate_desc    = $DESCRIPTOR('CONCATENATE'),
;   104          0183  1     new_version_desc    = $DESCRIPTOR('NEW_VERSION'),
;   105          0184  1     allocation_desc     = $DESCRIPTOR('ALLOCATION'),
;   106          0185  1     contiguous_desc     = $DESCRIPTOR('CONTIGUOUS'),
;   107          0186  1     extension_desc      = $DESCRIPTOR('EXTENSION'),
;   108          0187  1     file_max_desc       = $DESCRIPTOR('FILE_MAXIMUM'),
;   109          0188  1     protection_desc     = $DESCRIPTOR('PROTECTION'),
;   110          0189  1     read_check_desc     = $DESCRIPTOR('READ_CHECK'),
;   111          0190  1     write_check_desc    = $DESCRIPTOR('WRITE_CHECK'),
;   112          0191  1     overlay_desc        = $DESCRIPTOR('OVERLAY'),
;   113          0192  1     volume_desc         = $DESCRIPTOR('VOLUME'),
;   114          0193  1     truncate_desc       = $DESCRIPTOR('TRUNCATE'),
;   115          0194  1     replace_desc        = $DESCRIPTOR('REPLACE')
;   116          0195  1     ;
;   117          0196  1
;   118          0197  1 !
;   119          0198  1 ! Macros
;   120          0199  1 !
;   121          0200  1
;   122          0201  1 MACRO
;   123          0202  1
;   124          0203  1     ! These macros are all used in processing the /PROTECTION qualifier.
;   125          0204  1     !
;   126        M 0205  1     BIT_LOCATION( L, B, S, X) =                   ! Extract a bit from a field definition
;   127          0206  1                 B %,
;   128        M 0207  1     PROT_MASK(DISP,SIZE) =                        ! XAB$W_PRO bit and mask definitions macros
;   129          0208  1                 MASK_DEF(XAB$W_PRO,DISP,SIZE) %,
;   130        M 0209  1     MASK_DEF(L,B,S,X,DISP,SIZE) =
;   131          0210  1                 0, B+DISP, SIZE, X %;
;   132          0211  1
```

```
;  133          0212  1 !
;  134          0213  1 ! External declarations
;  135          0214  1 !
;  136          0215  1
;  137          0216  1 EXTERNAL
;  138          0217  1     copy$prot_value,                        ! Protection keyword value table
;  139          0218  1     copy$cli_status     : $BBLOCK,          ! Results of the command line parse
;  140          0219  1     copy$sem_status     : $BBLOCK           ! Semantics for copy operation
;  141          0220  1     ;
;  142          0221  1
;  143          0222  1 REQUIRE
;  144          0223  1     'SRC$:COPY.REQ';                        ! Field definitions for COPY$CLI_STATUS and COPY$SEM
```

```
; %PRINT:       File: VMSMAC.B32, Version V04-000, Edit 1, WWC, 09-JAN-1978
```

```
;  145          0678  1
;  146          0679  1
;  147          0680  1 EXTERNAL ROUTINE
;  148          0681  1     cli$present,                        ! Determine if a qualifier appears on the command li
;  149          0682  1     cli$get_value,                      ! Retrieve the qualifier's value
;  150          0683  1     lib$qual_file_parse,                ! Parse the common file qualifiers
;  151          0684  1     lib$cvt_dtb : ADDRESSING_MODE(GENERAL),     ! Convert String to binary
;  152          0685  1     lib$lookup_key: ADDRESSING_MODE(GENERAL);   ! Library keyword lookup routine
```

```
;  154          0686  1 !
;  155          0687  1 ! Global variables
;  156          0688  1 !
;  157          0689  1
;  158          0690  1 GLOBAL
;  159          0691  1
;  160          0692  1 !
;  161          0693  1 ! The following variables hold the current qualifier and option values gathered during the
;  162          0694  1 ! CLI processing.  These values may change as local qualifiers are parsed.  The global
;  163          0695  1 ! values are stored in COPY$CLI_STATUS.
;  164          0696  1 !
;  165          0697  1
;  166          0698  1     common_qual_context,                          ! Common qualifier data area
;  167          0699  1     curr_allocation_value,                        ! Binary allocation value
;  168          0700  1     curr_extension_value,                         ! Binary extension value
;  169          0701  1     curr_file_max_value,                          ! Binary file maximum value
;  170          0702  1     curr_protection_or  :                         ! Protection mask to set bits
;  171          0703  1         $BBLOCK[ 2 ]
;  172          0704  1         INITIAL (REP 2 OF BYTE (0)),
;  173          0705  1     curr_protection_and :                         ! Protection mask to clear bits
;  174          0706  1         $BBLOCK[ 2 ]
;  175          0707  1         INITIAL (REP 2 OF BYTE (-1)),
;  176          0708  1     curr_volume_value   : INITIAL (0)             ! Relative volume number
;  177          0709  1     ;
;  178          0710  1
```

```
;  180     0711  1 GLOBAL ROUTINE COPY$GET_GLOBAL_QUAL:  NOVALUE =          ! Retrieve gloabl qualifiers from the CLI
;  181     0712  1
;  182     0713  1 !++
;  183     0714  1 ! FUNCTIONAL DESCRIPTION:
;  184     0715  1 !
;  185     0716  1 !       This routine retrieves the command level qualifiers from the
;  186     0717  1 !       Command Language Interpreter.  It treats any qualifiers found
;  187     0718  1 !       as global, even if they are only locally present.  This ensures
;  188     0719  1 !       that qualifiers which appear on the output file have the same
;  189     0720  1 !       effect as ones which appear on the verb.
;  190     0721  1 !
;  191     0722  1 !
;  192     0723  1 ! FORMAL PARAMETERS:
;  193     0724  1 !
;  194     0725  1 !       None
;  195     0726  1 !
;  196     0727  1 ! IMPLICIT INPUTS:
;  197     0728  1 !
;  198     0729  1 !       None
;  199     0730  1 !
;  200     0731  1 ! IMPLICIT OUTPUTS:
;  201     0732  1 !
;  202     0733  1 !       COPY$CLI_STATUS - Relevant command and qualifier indicators set
;  203     0734  1 !
;  204     0735  1 ! ROUTINE VALUE:
;  205     0736  1 !
;  206     0737  1 !       None
;  207     0738  1 !
;  208     0739  1 ! SIDE EFFECTS:
;  209     0740  1 !
;  210     0741  1 !       None
;  211     0742  1 !
;  212     0743  1 !--
;  213     0744  1
;  214     0745  2     BEGIN
;  215     0746  2
;  216     0747  2     LOCAL
;  217     0748  2         common_qual_flags,                          ! Bits which select the common qualifiers to be pars
;  218     0749  2         rtn_status,                                 ! Status returned from external calls
;  219     0750  2         cli_desc :  $BBLOCK[ dsc$c_s_bln ]          ! Dynamic string descriptor, points to values
;  220     0751  2         ;                                           !   returned from calls to the CLI
;  221     0752  2
;  222     0753  2
;  223     0754  2
;  224     0755  2
;  225     0756  2     ! Initialize descriptor.
;  226     0757  2     !
;  227     0758  2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
;  228     0759  2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
;  229     0760  2
;  230     0761  2
;  231     0762  2     ! Retrieve the verb from the command line.  Determine if it is a COPY or APPEND command.
;  232     0763  2     !
;  233     0764  2     CLI$GET_VALUE( verb_desc, cli_desc);
;  234     0765  2     IF CH$RCHAR( .cli_desc[ DSC$A_POINTER ]) EQL 'A'
;  235     0766  2     THEN
;  236     0767  3         BEGIN
```

```
;  237     0768  3
;  238     0769  3          ! It was an APPEND command.  Set the append command flag and parse the APPEND
;  239     0770  3          ! specific qualifiers.
;  240     0771  3          !
;  241     0772  3          append_command   = TRUE;
;  242     0773  3          new_version_qual = CLI$PRESENT( new_version_desc );
;  243     0774  3          END
;  244     0775  2      ELSE
;  245     0776  3          BEGIN
;  246     0777  3
;  247     0778  3          ! It was a COPY command.  Parse the COPY specific qualifiers.
;  248     0779  3          !
;  249     0780  3          ! /CONCATENATE, /TRUNCATE -- Set the apropriate flags if the qualifier
;  250     0781  3          ! was given or negated.
;  251     0782  3          !
;  252     0783  3          SELECTONE CLI$PRESENT( concatenate_desc ) OF
;  253     0784  3          SET
;  254     0785  4              [ CLI$_PRESENT ] : BEGIN
;  255     0786  4                                 concat_qual          = TRUE;
;  256     0787  4                                 explicit_concat_qual = TRUE;
;  257     0788  4                                 END;
;  258     0789  4              [ CLI$_NEGATED ] : negated_concat_qual  = TRUE;
;  259     0790  3          TES;
;  260     0791  3
;  261     0792  3          SELECTONE CLI$PRESENT( truncate_desc ) OF
;  262     0793  3          SET
;  263     0794  4              [ CLI$_PRESENT,
;  264     0795  4                CLI$_LOCPRES ] : truncate_qual    = TRUE;
;  265     0796  4              [ CLI$_NEGATED ] : truncate_negated = TRUE;
;  266     0797  3          TES;
;  267     0798  3
;  268     0799  3          ! /OVERLAY and  /REPLACE
;  269     0800  3          !
;  270     0801  3          overlay_qual = CLI$PRESENT( overlay_desc );
;  271     0802  3          replace_qual = CLI$PRESENT( replace_desc );
;  272     0803  3
;  273     0804  3
;  274     0805  3          ! /VOLUME
;  275     0806  3          !
;  276     0807  4          IF (volume_qual = CLI$PRESENT( volume_desc ))
;  277     0808  3          THEN
;  278     0809  4              BEGIN
;  279     0810  4
;  280     0811  4              ! Get the value and and convert it from a string into binary.
;  281     0812  4              !
;  282     0813  4              CLI$GET_VALUE( volume_desc, cli_desc );
;  283     0814  5              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  284     0815  5                                               .cli_desc[ DSC$A_POINTER ], volume_value))
;  285     0816  4              THEN
;  286     0817  4                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, volume_desc );
;  287     0818  4              curr_volume_value = .volume_value;
;  288     0819  3              END;
;  289     0820  2          END;
;  290     0821  2
;  291     0822  2      ! Parse the qualifiers which are applicable to both commands.  First,
;  292     0823  2      ! the common qualifiers (/CONFIRM, /BEFORE, /SINCE, /EXCLUDE, /CREATED,
;  293     0824  2      ! /MODIFIED, /BACKUP, /EXPIRED, /BY_OWNER)
```

```
;  294     0825  2     !
;  295     0826  2     ! Initialize the flags longword so that all of the common qualifiers will
;  296     0827  2     ! be parsed.  Then, parse the qualifiers.
;  297     0828  2     !
;  298     0829  2     common_qual_flags = LIB$M_CQF_CONFIRM  OR  LIB$M_CQF_BEFORE  OR
;  299     0830  2                         LIB$M_CQF_SINCE    OR  LIB$M_CQF_CREATED OR
;  300     0831  2                         LIB$M_CQF_MODIFIED OR  LIB$M_CQF_BACKUP  OR
;  301     0832  2                         LIB$M_CQF_EXPIRED  OR  LIB$M_CQF_EXCLUDE OR
;  302     0833  2                         LIB$M_CQF_BYOWNER;
;  303     0834  2
;  304     0835  3     IF NOT (rtn_status = LIB$QUAL_FILE_PARSE( common_qual_flags, common_qual_context ))
;  305     0836  2     THEN
;  306     0837  2         PUT_MESSAGEX( .rtn_status );
;  307     0838  2
;  308     0839  2
;  309     0840  2     ! /LOG, /READ_CHECK, /WRITE_CHECK and /CONTIGUOUS
;  310     0841  2     !
;  311     0842  2     log_msg_qual  = CLI$PRESENT( log_msg_desc );
;  312     0843  2
;  313     0844  2     read_chk_qual = CLI$PRESENT( read_check_desc );
;  314     0845  2
;  315     0846  2     SELECTONE CLI$PRESENT( write_check_desc ) OF
;  316     0847  2     SET
;  317     0848  2         [ CLI$_PRESENT,
;  318     0849  2           CLI$_LOCPRES ] : write_chk_qual    = TRUE;
;  319     0850  2         [ CLI$_NEGATED ] : write_chk_negated = TRUE;
;  320     0851  2     TES;
;  321     0852  2     write_chk_qual = CLI$PRESENT( write_check_desc );
;  322     0853  2
;  323     0854  2     SELECTONE CLI$PRESENT( contiguous_desc ) OF
;  324     0855  2     SET
;  325     0856  2         [ CLI$_PRESENT,
;  326     0857  2           CLI$_LOCPRES ] : contig_qual    = TRUE;
;  327     0858  2         [ CLI$_NEGATED ] : contig_negated = TRUE;
;  328     0859  2     TES;
;  329     0860  2
;  330     0861  2
;  331     0862  2     ! /ALLOCATION
;  332     0863  2     !
;  333     0864  3     IF (alloc_qual = CLI$PRESENT( allocation_desc ))
;  334     0865  2     THEN
;  335     0866  3         BEGIN
;  336     0867  3
;  337     0868  3         ! Get the value and and convert it from a string into binary.
;  338     0869  3         !
;  339     0870  3         CLI$GET_VALUE( allocation_desc, cli_desc );
;  340     0871  4         IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  341     0872  4                                          .cli_desc[ DSC$A_POINTER ], allocation_value))
;  342     0873  3         THEN
;  343     0874  3             PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, allocation_desc );
;  344     0875  3         curr_allocation_value = .allocation_value;
;  345     0876  2         END;
;  346     0877  2
;  347     0878  2
;  348     0879  2     ! /EXTENSION
;  349     0880  2     !
;  350     0881  3     IF (extend_qual = CLI$PRESENT( extension_desc ))
```

```
THEN
    BEGIN

    ! Get the value and and convert it from a string into binary.
    !
    CLI$GET_VALUE( extension_desc, cli_desc );
    IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
                         .cli_desc[ DSC$A_POINTER ], extension_value))
    THEN
        PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, extension_desc );
    curr_extension_value = .extension_value;
    END;


! /FILE_MAXIMUM
!
IF (file_max_qual = CLI$PRESENT( file_max_desc ))
THEN
    BEGIN

    ! Get the value and and convert it from a string into binary.
    !
    CLI$GET_VALUE( file_max_desc, cli_desc );
    IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
                         .cli_desc[ DSC$A_POINTER ], file_max_value))
    THEN
        PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, file_max_desc );
    curr_file_max_value = .file_max_value;
    END;


! /PROTECTION
!
IF (protect_qual = CLI$PRESENT( protection_desc ))
THEN
    BEGIN

    ! Parse the keyword value and save the results.
    !
    protection_parse();
    protection_and = .(curr_protection_and);
    protection_or  = .(curr_protection_or);
    END;
END;                                    ! routine COPY$GET_GLOBAL_QUAL

                                        .TITLE  COPYCLI
                                        .IDENT  \V04-000\

                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

          42  52  45  56  24  00000 P.AAB:  .ASCII  \$VERB\              ;
                              00005         .BLKB   3
                   00000005   00008 P.AAA:  .LONG   5                    ;
                   00000000'  0000C         .ADDRESS P.AAB               ;
                  47  4F  4C  00010 P.AAD:  .ASCII  \LOG\                ;
                              00013         .BLKB   1
```

```
                                00000003  00014 P.AAC:  .LONG   3                                                ;
                                00000000' 00018         .ADDRESS P.AAD                                         ;
                45 54 41 4E 45 54 41 43 4E 4F 43  0001C P.AAF:  .ASCII  \CONCATENATE\                           ;
                                          00027         .BLKB   1                                              ;
                                0000000B  00028 P.AAE:  .LONG   11                                             ;
                                00000000' 0002C         .ADDRESS P.AAF                                         ;
                4E 4F 49 53 52 45 56 5F 57 45 4E  00030 P.AAH:  .ASCII  \NEW_VERSION\                          ;
                                          0003B         .BLKB   1                                              ;
                                0000000B  0003C P.AAG:  .LONG   11                                             ;
                                00000000' 00040         .ADDRESS P.AAH                                         ;
                   4E 4F 49 54 41 43 4F 4C 4C 41  00044 P.AAJ:  .ASCII  \ALLOCATION\                           ;
                                          0004E         .BLKB   2                                              ;
                                0000000A  00050 P.AAI:  .LONG   10                                             ;
                                00000000' 00054         .ADDRESS P.AAJ                                         ;
                   53 55 4F 55 47 49 54 4E 4F 43  00058 P.AAL:  .ASCII  \CONTIGUOUS\                           ;
                                          00062         .BLKB   2                                              ;
                                0000000A  00064 P.AAK:  .LONG   10                                             ;
                                00000000' 00068         .ADDRESS P.AAL                                         ;
                      4E 4F 49 53 4E 45 54 58 45  0006C P.AAN:  .ASCII  \EXTENSION\                            ;
                                          00075         .BLKB   3                                              ;
                                00000009  00078 P.AAM:  .LONG   9                                              ;
                                00000000' 0007C         .ADDRESS P.AAN                                         ;
          4D 55 4D 49 58 41 4D 5F 45 4C 49 46  00080 P.AAP:  .ASCII  \FILE_MAXIMUM\                            ;
                                0000000C  0008C P.AAO:  .LONG   12                                             ;
                                00000000' 00090         .ADDRESS P.AAP                                         ;
                   4E 4F 49 54 43 45 54 4F 52 50  00094 P.AAR:  .ASCII  \PROTECTION\                           ;
                                          0009E         .BLKB   2                                              ;
                                0000000A  000A0 P.AAQ:  .LONG   10                                             ;
                                00000000' 000A4         .ADDRESS P.AAR                                         ;
                   4B 43 45 48 43 5F 44 41 45 52  000A8 P.AAT:  .ASCII  \READ_CHECK\                           ;
                                          000B2         .BLKB   2                                              ;
                                0000000A  000B4 P.AAS:  .LONG   10                                             ;
                                00000000' 000B8         .ADDRESS P.AAT                                         ;
          4B 43 45 48 43 5F 45 54 49 52 57  000BC P.AAV:  .ASCII  \WRITE_CHECK\                                ;
                                          000C7         .BLKB   1                                              ;
                                0000000B  000C8 P.AAU:  .LONG   11                                             ;
                                00000000' 000CC         .ADDRESS P.AAV                                         ;
                      59 41 4C 52 45 56 4F  000D0 P.AAX:  .ASCII  \OVERLAY\                                    ;
                                          000D7         .BLKB   1                                              ;
                                00000007  000D8 P.AAW:  .LONG   7                                              ;
                                00000000' 000DC         .ADDRESS P.AAX                                         ;
                         45 4D 55 4C 4F 56  000E0 P.AAZ:  .ASCII  \VOLUME\                                     ;
                                          000E6         .BLKB   2                                              ;
                                00000006  000E8 P.AAY:  .LONG   6                                              ;
                                00000000' 000EC         .ADDRESS P.AAZ                                         ;
                   45 54 41 43 4E 55 52 54  000F0 P.ABB:  .ASCII  \TRUNCATE\                                   ;
                                00000008  000F8 P.ABA:  .LONG   8                                              ;
                                00000000' 000FC         .ADDRESS P.ABB                                         ;
                      45 43 41 4C 50 45 52  00100 P.ABD:  .ASCII  \REPLACE\                                     ;
                                          00107         .BLKB   1                                              ;
                                00000007  00108 P.ABC:  .LONG   7                                              ;
                                00000000' 0010C         .ADDRESS P.ABD                                         ;

                                                        .PSECT  $GLOBAL$,NOEXE,2

                                          00000 COMMON_QUAL_CONTEXT::
                                                        .BLKB   4
```

```
                        00004 CURR_ALLOCATION_VALUE::
                                      .BLKB   4
                        00008 CURR_EXTENSION_VALUE::
                                      .BLKB   4
                        0000C CURR_FILE_MAX_VALUE::
                                      .BLKB   4
                     00# 00010 CURR_PROTECTION_OR::
                                      .BYTE   0[2]                                                    ;
                        00012         .BLKB   2
                     FF# 00014 CURR_PROTECTION_AND::
                                      .BYTE   -1[2]                                                   ;
                        00016         .BLKB   2
                00000000  00018 CURR_VOLUME_VALUE::
                                      .LONG   0                                                       ;

                                VERB_DESC=          P.AAA
                                LOG_MSG_DESC=       P.AAC
                                CONCATENATE_DESC=   P.AAE
                                NEW_VERSION_DESC=   P.AAG
                                ALLOCATION_DESC=    P.AAI
                                CONTIGUOUS_DESC=    P.AAK
                                EXTENSION_DESC=     P.AAM
                                FILE_MAX_DESC=      P.AAO
                                PROTECTION_DESC=    P.AAQ
                                READ_CHECK_DESC=    P.AAS
                                WRITE_CHECK_DESC=   P.AAU
                                OVERLAY_DESC=       P.AAW
                                VOLUME_DESC=        P.AAY
                                TRUNCATE_DESC=      P.ABA
                                REPLACE_DESC=       P.ABC
                                        .EXTRN  COPY$MSG_NUMBER
                                        .EXTRN  COPY$PROT_VALUE
                                        .EXTRN  COPY$CLI_STATUS
                                        .EXTRN  COPY$SEM_STATUS
                                        .EXTRN  CLI$_PRESENT, CLI$_NEGATED
                                        .EXTRN  CLI$_LOCPRES, CLI$_LOCNEG
                                        .EXTRN  CLI$PRESENT, CLI$GET_VALUE
                                        .EXTRN  LIB$QUAL_FILE_PARSE
                                        .EXTRN  LIB$CVT_DTB, LIB$LOOKUP_KEY

                                        .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000          .ENTRY  COPY$GET_GLOBAL_QUAL, Save R2,R3,R4,R5,R6,- ; 0711
                                                                R7,R8,R9,R10,R11                            ;
                           5B 00000000G   00  9E 00002          MOVAB   LIB$STOP, R11                       ;
                           5A 00000000G   00  9E 00009          MOVAB   LIB$SIGNAL, R10                     ;
                           59     0000G   CF  9E 00010          MOVAB   COPY$MSG_NUMBER, R9                 ;
                           58     0000G   CF  9E 00015          MOVAB   CLI$PRESENT, R8                     ;
                           57     0000'   CF  9E 0001A          MOVAB   VOLUME_DESC, R7                     ;
                           56     0000G   CF  9E 0001F          MOVAB   COPY$CLI_STATUS+4, R6               ;
                           5E             0C  C2 00024          SUBL2   #12, SP                             ;
08              00         6E             00  2C 00027          MOVC5   #0, (SP), #0, #8, CLI_DESC          ; 0758
                                      04   AE      0002C
                                07    AE         02  90 0002E    MOVB    #2, CLI_DESC+3                      ; 0759
                                      04   AE  9F 00032          PUSHAB  CLI_DESC                            ; 0764
                                    FF20   C7  9F 00035          PUSHAB  VERB_DESC                           ;
                             0000G  CF          02  FB 00039    CALLS   #2, CLI$GET_VALUE                   ;
```

```
                        41   8F      08   BE  91 0003E          CMPB    @CLI_DESC+4, #65                        ; 0765
                                         14  12 00043          BNEQ    1$
                        FC   A6          01  88 00045          BISB2   #1, COPY$CLI_STATUS                     ; 0772
                                     FF54  C7  9F 00049          PUSHAB  NEW_VERSION_DESC                        ; 0773
                             68          01  FB 0004D          CALLS   #1, CLI$PRESENT                         ;
    FC   A6          01                 04  50  F0 00050          INSV    R0, #4, #1, COPY$CLI_STATUS             ;
                                         00E1  31 00056          BRW     9$                                      ; 0765
                                     FF40  C7  9F 00059  1$:     PUSHAB  CONCATENATE_DESC                        ; 0783
                             68          01  FB 0005D          CALLS   #1, CLI$PRESENT                         ;
                  00000000G  8F          50  D1 00060          CMPL    R0, #CLI$_PRESENT                       ; 0785
                                         0B  12 00067          BNEQ    2$                                      ;
                       0000G CF          01  88 00069          BISB2   #1, COPY$SEM_STATUS                     ; 0786
                        FC   A6          04  88 0006E          BISB2   #4, COPY$CLI_STATUS                     ; 0787
                                         0D  11 00072          BRB     3$                                      ; 0783
                  00000000G  8F          50  D1 00074  2$:     CMPL    R0, #CLI$_NEGATED                       ; 0789
                                         04  12 0007B          BNEQ    3$                                      ;
                        FC   A6          08  88 0007D          BISB2   #8, COPY$CLI_STATUS                     ;
                                     10  A7  9F 00081  3$:     PUSHAB  TRUNCATE_DESC                           ; 0792
                             68          01  FB 00084          CALLS   #1, CLI$PRESENT                         ;
                  00000000G  8F          50  D1 00087          CMPL    R0, #CLI$_PRESENT                       ; 0794
                                         09  13 0008E          BEQL    4$                                      ;
                  00000000G  8F          50  D1 00090          CMPL    R0, #CLI$_LOCPRES                       ;
                                         06  12 00097          BNEQ    5$                                      ;
                        01   A6          20  88 00099  4$:     BISB2   #32, COPY$CLI_STATUS+5                  ; 0795
                                         0E  11 0009D          BRB     6$                                      ;
                  00000000G  8F          50  D1 0009F  5$:     CMPL    R0, #CLI$_NEGATED                       ; 0796
                                         05  12 000A6          BNEQ    6$                                      ;
                        01   A6      40  8F  88 000A8          BISB2   #64, COPY$CLI_STATUS+5                  ;
                                     F0  A7  9F 000AD  6$:     PUSHAB  OVERLAY_DESC                            ; 0801
                             68          01  FB 000B0          CALLS   #1, CLI$PRESENT                         ;
          66             01          07  50  F0 000B3          INSV    R0, #7, #1, COPY$CLI_STATUS+4           ;
                                     20  A7  9F 000B8          PUSHAB  REPLACE_DESC                            ; 0802
                             68          01  FB 000BB          CALLS   #1, CLI$PRESENT                         ;
    02   A6          01                 01  50  F0 000BE          INSV    R0, #1, #1, COPY$CLI_STATUS+6           ;
                                         57  DD 000C4          PUSHL   R7                                      ; 0807
                             68          01  FB 000C6          CALLS   #1, CLI$PRESENT                         ;
    01   A6          01                 02  50  F0 000C9          INSV    R0, #2, #1, COPY$CLI_STATUS+5           ;
                             68          50  E9 000CF          BLBC    R0, 9$                                  ;
                                      04  AE  9F 000D2          PUSHAB  CLI_DESC                                ; 0813
                                         57  DD 000D5          PUSHL   R7                                      ;
                       0000G CF          02  FB 000D7          CALLS   #2, CLI$GET_VALUE                       ;
                                      14  A6  9F 000DC          PUSHAB  COPY$CLI_STATUS+24                      ; 0815
                                      0C  AE  DD 000DF          PUSHL   CLI_DESC+4                              ;
                             7E       0C  AE  3C 000E2          MOVZWL  CLI_DESC, -(SP)                         ; 0814
                  00000000G  00          03  FB 000E6          CALLS   #3, LIB$CVT_DTB                         ;
                             52          50  D0 000ED          MOVL    R0, RTN_STATUS                          ;
                             41          52  E8 000F0          BLBS    RTN_STATUS, 8$                          ;
                             7E   132C   8F  3C 000F3          MOVZWL  #4908, -(SP)                            ; 0817
                             69          01  FB 000F8          CALLS   #1, COPY$MSG_NUMBER                     ;
          7E             00          50  01  7A 000FB          EMUL    #1, R0, #0, -(SP)                       ;
          50             50          8E  08  7B 00100          EDIV    #8, (SP)+, R0, R0                       ;
                             04          50  D1 00105          CMPL    R0, #4                                  ;
                                         16  13 00108          BEQL    7$                                      ;
                                         57  DD 0010A          PUSHL   R7                                      ;
                                      08  AE  9F 0010C          PUSHAB  CLI_DESC                                ;
                                         02  DD 0010F          PUSHL   #2                                      ;
                             7E   132C   8F  3C 00111          MOVZWL  #4908, -(SP)                            ;
```

```
                                69              01 FB 00116          CALLS   #1, COPY$MSG_NUMBER                                     ;
                                50              DD 00119          PUSHL   R0                                                      ;
                                6A              04 FB 0011B          CALLS   #4, LIB$SIGNAL                                          ;
                                                14 11 0011E          BRB     8$                                                      ;
                                                57 DD 00120 7$:      PUSHL   R7                                                      ;
                                      08        AE 9F 00122          PUSHAB  CLI_DESC                                                ;
                                                02 DD 00125          PUSHL   #2                                                      ;
                                7E    132C      8F 3C 00127          MOVZWL  #4908, -(SP)                                            ;
                                69              01 FB 0012C          CALLS   #1, COPY$MSG_NUMBER                                     ;
                                50              DD 0012F          PUSHL   R0                                                      ;
                                6B              04 FB 00131          CALLS   #4, LIB$STOP                                            ;
                     0000'      CF        14   A6 D0 00134 8$:      MOVL    COPY$CLI_STATUS+24, CURR_VOLUME_VALUE                   ; 0818
                                6E    01FF      8F 3C 0013A 9$:      MOVZWL  #511, COMMON_QUAL_FLAGS                                 ; 0832
                                      0000'     CF 9F 0013F          PUSHAB  COMMON_QUAL_CONTEXT                                     ; 0835
                                        04      AE 9F 00143          PUSHAB  COMMON_QUAL_FLAGS                                       ;
                     0000G      CF              02 FB 00146          CALLS   #2, LIB$QUAL_FILE_PARSE                                 ;
                                52              50 D0 0014B          MOVL    R0, RTN_STATUS                                          ;
                                2A              52 E8 0014E          BLBS    RTN_STATUS, 11$                                         ;
                                                52 DD 00151          PUSHL   RTN_STATUS                                              ; 0837
                                69              01 FB 00153          CALLS   #1, COPY$MSG_NUMBER                                     ;
      7E            00          50              01 7A 00156          EMUL    #1, R0, #0, -(SP)                                       ;
      50            50          8E              08 7B 0015B          EDIV    #8, (SP)+, R0, R0                                       ;
                                04              50 D1 00160          CMPL    R0, #4                                                  ;
                                                0C 13 00163          BEQL    10$                                                     ;
                                                52 DD 00165          PUSHL   RTN_STATUS                                              ;
                                69              01 FB 00167          CALLS   #1, COPY$MSG_NUMBER                                     ;
                                50              DD 0016A          PUSHL   R0                                                      ;
                                6A              01 FB 0016C          CALLS   #1, LIB$SIGNAL                                          ;
                                                0A 11 0016F          BRB     11$                                                     ;
                                                52 DD 00171 10$:     PUSHL   RTN_STATUS                                              ;
                                69              01 FB 00173          CALLS   #1, COPY$MSG_NUMBER                                     ;
                                50              DD 00176          PUSHL   R0                                                      ;
                                6B              01 FB 00178          CALLS   #1, LIB$STOP                                            ;
                                      FF2C      C7 9F 0017B 11$:     PUSHAB  LOG_MSG_DESC                                            ; 0842
                                68              01 FB 0017F          CALLS   #1, CLI$PRESENT                                         ;
      FC    A6          01          01          50 F0 00182          INSV    R0, #1, #1, COPY$CLI_STATUS                             ;
                                      CC        A7 9F 00188          PUSHAB  READ_CHECK_DESC                                         ; 0844
                                68              01 FB 0018B          CALLS   #1, CLI$PRESENT                                         ;
            66          01          00          50 F0 0018E          INSV    R0, #0, #1, COPY$CLI_STATUS+4                           ;
                                      E0        A7 9F 00193          PUSHAB  WRITE_CHECK_DESC                                        ; 0846
                                68              01 FB 00196          CALLS   #1, CLI$PRESENT                                         ;
                          00000000G   8F        50 D1 00199          CMPL    R0, #CLI$_PRESENT                                       ; 0848
                                                09 13 001A0          BEQL    12$                                                     ;
                          00000000G   8F        50 D1 001A2          CMPL    R0, #CLI$_LOCPRES                                       ;
                                                05 12 001A9          BNEQ    13$                                                     ;
                                66              08 88 001AB 12$:     BISB2   #8, COPY$CLI_STATUS+4                                   ; 0849
                                                0C 11 001AE          BRB     14$                                                     ;
                          00000000G   8F        50 D1 001B0 13$:     CMPL    R0, #CLI$_NEGATED                                       ; 0850
                                                03 12 001B7          BNEQ    14$                                                     ;
                                66              10 88 001B9          BISB2   #16, COPY$CLI_STATUS+4                                  ;
                                      E0        A7 9F 001BC 14$:     PUSHAB  WRITE_CHECK_DESC                                        ; 0852
                                68              01 FB 001BF          CALLS   #1, CLI$PRESENT                                         ;
            66          01          03          50 F0 001C2          INSV    R0, #3, #1, COPY$CLI_STATUS+4                           ;
                                      FF7C      C7 9F 001C7          PUSHAB  CONTIGUOUS_DESC                                         ; 0854
                                68              01 FB 001CB          CALLS   #1, CLI$PRESENT                                         ;
                          00000000G   8F        50 D1 001CE          CMPL    R0, #CLI$_PRESENT                                       ; 0856
                                                09 13 001D5          BEQL    15$                                                     ;
```

```
                         00000000G  8F          50  D1 001D7         CMPL    R0, #CLI$_LOCPRES                     ;
                                                06  12 001DE         BNEQ    16$                                   ;
                                FE  A6          08  88 001E0 15$:    BISB2   #8, COPY$CLI_STATUS+2                 ; 0857
                                                0D  11 001E4         BRB     17$                                   ;
                         00000000G  8F          50  D1 001E6 16$:    CMPL    R0, #CLI$_NEGATED                     ; 0858
                                                04  12 001ED         BNEQ    17$                                   ;
                                FE  A6          10  88 001EF         BISB2   #16, COPY$CLI_STATUS+2                ;
                                          FF68  C7  9F 001F3 17$:    PUSHAB  ALLOCATION_DESC                       ; 0864
                                    68          01  FB 001F7         CALLS   #1, CLI$PRESENT                       ;
      FE   A6                 01    00          50  F0 001FA         INSV    R0, #0, #1, COPY$CLI_STATUS+2         ;
                                    6E          50  E9 00200         BLBC    R0, 20$                               ;
                                            04  AE  9F 00203         PUSHAB  CLI_DESC                              ; 0870
                                          FF68  C7  9F 00206         PUSHAB  ALLOCATION_DESC                       ;
                             0000G  CF          02  FB 0020A         CALLS   #2, CLI$GET_VALUE                     ;
                                            04  A6  9F 0020F         PUSHAB  COPY$CLI_STATUS+8                     ; 0872
                                            0C  AE  DD 00212         PUSHL   CLI_DESC+4                            ;
                                    7E      0C  AE  3C 00215         MOVZWL  CLI_DESC, -(SP)                       ; 0871
                         00000000G  00          03  FB 00219         CALLS   #3, LIB$CVT_DTB                       ;
                                    52          50  D0 00220         MOVL    R0, RTN_STATUS                        ;
                                    45          52  E8 00223         BLBS    RTN_STATUS, 19$                       ;
                                    7E    132C  8F  3C 00226         MOVZWL  #4908, -(SP)                          ; 0874
                                    69          01  FB 0022B         CALLS   #1, COPY$MSG_NUMBER                   ;
           7E                 00    50          01  7A 0022E         EMUL    #1, R0, #0, -(SP)                     ;
           50                 50    8E          08  7B 00233         EDIV    #8, (SP)+, R0, R0                     ;
                                    04          50  D1 00238         CMPL    R0, #4                                ;
                                                18  13 0023B         BEQL    18$                                   ;
                                          FF68  C7  9F 0023D         PUSHAB  ALLOCATION_DESC                       ;
                                            08  AE  9F 00241         PUSHAB  CLI_DESC                              ;
                                                02  DD 00244         PUSHL   #2                                    ;
                                    7E    132C  8F  3C 00246         MOVZWL  #4908, -(SP)                          ;
                                    69          01  FB 0024B         CALLS   #1, COPY$MSG_NUMBER                   ;
                                                50  DD 0024E         PUSHL   R0                                    ;
                                    6A          04  FB 00250         CALLS   #4, LIB$SIGNAL                        ;
                                                16  11 00253         BRB     19$                                   ;
                                          FF68  C7  9F 00255 18$:    PUSHAB  ALLOCATION_DESC                       ;
                                            08  AE  9F 00259         PUSHAB  CLI_DESC                              ;
                                                02  DD 0025C         PUSHL   #2                                    ;
                                    7E    132C  8F  3C 0025E         MOVZWL  #4908, -(SP)                          ;
                                    69          01  FB 00263         CALLS   #1, COPY$MSG_NUMBER                   ;
                                                50  DD 00266         PUSHL   R0                                    ;
                                    6B          04  FB 00268         CALLS   #4, LIB$STOP                          ;
                             0000'  CF      04  A6  D0 0026B 19$:    MOVL    COPY$CLI_STATUS+8, CURR_ALLOCATION_VALUE ; 0875
                                            90  A7  9F 00271 20$:    PUSHAB  EXTENSION_DESC                        ; 0881
                                    68          01  FB 00274         CALLS   #1, CLI$PRESENT                       ;
      FE   A6                 01    07          50  F0 00277         INSV    R0, #7, #1, COPY$CLI_STATUS+2         ;
                                    6B          50  E9 0027D         BLBC    R0, 23$                               ;
                                            04  AE  9F 00280         PUSHAB  CLI_DESC                              ; 0887
                                            90  A7  9F 00283         PUSHAB  EXTENSION_DESC                        ;
                             0000G  CF          02  FB 00286         CALLS   #2, CLI$GET_VALUE                     ;
                                            08  A6  9F 0028B         PUSHAB  COPY$CLI_STATUS+12                    ; 0889
                                            0C  AE  DD 0028E         PUSHL   CLI_DESC+4                            ;
                                    7E      0C  AE  3C 00291         MOVZWL  CLI_DESC, -(SP)                       ; 0888
                         00000000G  00          03  FB 00295         CALLS   #3, LIB$CVT_DTB                       ;
                                    52          50  D0 0029C         MOVL    R0, RTN_STATUS                        ;
                                    43          52  E8 0029F         BLBS    RTN_STATUS, 22$                       ;
                                    7E    132C  8F  3C 002A2         MOVZWL  #4908, -(SP)                          ; 0891
                                    69          01  FB 002A7         CALLS   #1, COPY$MSG_NUMBER                   ;
```

```
                7E      00              50              01  7A 002AA          EMUL    #1, R0, #0, -(SP)                                       ;
                5C      50              8E              08  7B 002AF          EDIV    #8, (SP)+, R0, R0                                       ;
                                        04              50  D1 002B4          CMPL    R0, #4                                                  ;
                                                        17  13 002B7          BEQL    21$                                                     ;
                                                    90  A7  9F 002B9          PUSHAB  EXTENSION_DESC                                          ;
                                                    08  AE  9F 002BC          PUSHAB  CLI_DESC                                                ;
                                                        02  DD 002BF          PUSHL   #2                                                      ;
                                                7E  132C    8F  3C 002C1      MOVZWL  #4908, -(SP)                                            ;
                                                69          01  FB 002C6      CALLS   #1, COPY$MSG_NUMBER                                     ;
                                                            50  DD 002C9      PUSHL   R0                                                      ;
                                                6A          04  FB 002CB      CALLS   #4, LIB$SIGNAL                                          ;
                                                            15  11 002CE      BRB     22$                                                     ;
                                                    90  A7  9F 002D0 21$:    PUSHAB  EXTENSION_DESC                                          ;
                                                    08  AE  9F 002D3          PUSHAB  CLI_DESC                                                ;
                                                        02  DD 002D6          PUSHL   #2                                                      ;
                                                7E  132C    8F  3C 002D8      MOVZWL  #4908, -(SP)                                            ;
                                                69          01  FB 002DD      CALLS   #1, COPY$MSG_NUMBER                                     ;
                                                            50  DD 002E0      PUSHL   R0                                                      ;
                                                6B          04  FB 002E2      CALLS   #4, LIB$STOP                                            ;
                                    0000'   CF      08  A6  D0 002E5 22$:    MOVL    COPY$CLI_STATUS+12, CURR_EXTENSION_VALUE        ; 0892
                                                    A4  A7  9F 002EB 23$:    PUSHAB  FILE_MAX_DESC                                   ; 0898
                                                68          01  FB 002EE      CALLS   #1, CLI$PRESENT                                         ;
            FF  A6              01              02          50  F0 002F1      INSV    R0, #2, #1, COPY$CLI_STATUS+3                           ;
                                                6B          50  E9 002F7      BLBC    R0, 26$                                                 ;
                                                    04  AE  9F 002FA          PUSHAB  CLI_DESC                                        ; 0904
                                                    A4  A7  9F 002FD          PUSHAB  FILE_MAX_DESC                                           ;
                                    0000G   CF              02  FB 00300      CALLS   #2, CLI$GET_VALUE                                       ;
                                                    0C  A6  9F 00305          PUSHAB  COPY$CLI_STATUS+16                              ; 0906
                                                    0C  AE  DD 00308          PUSHL   CLI_DESC+4                                              ;
                                                7E  0C  AE  3C 0030B          MOVZWL  CLI_DESC, -(SP)                                 ; 0905
                            00000000G           00          03  FB 0030F      CALLS   #3, LIB$CVT_DTB                                         ;
                                                52          50  D0 00316      MOVL    R0, RTN_STATUS                                          ;
                                                43          52  E8 00319      BLBS    RTN_STATUS, 25$                                         ;
                                                7E  132C    8F  3C 0031C      MOVZWL  #4908, -(SP)                                    ; 0908
                                                69          01  FB 00321      CALLS   #1, COPY$MSG_NUMBER                                     ;
            7E      00              50                  01  7A 00324          EMUL    #1, R0, #0, -(SP)                                       ;
            50      50              8E                  08  7B 00329          EDIV    #8, (SP)+, R0, R0                                       ;
                                    04                  50  D1 0032E          CMPL    R0, #4                                                  ;
                                                        17  13 00331          BEQL    24$                                                     ;
                                                    A4  A7  9F 00333          PUSHAB  FILE_MAX_DESC                                           ;
                                                    08  AE  9F 00336          PUSHAB  CLI_DESC                                                ;
                                                        02  DD 00339          PUSHL   #2                                                      ;
                                                7E  132C    8F  3C 0033B      MOVZWL  #4908, -(SP)                                            ;
                                                69          01  FB 00340      CALLS   #1, COPY$MSG_NUMBER                                     ;
                                                            50  DD 00343      PUSHL   R0                                                      ;
                                                6A          04  FB 00345      CALLS   #4, LIB$SIGNAL                                          ;
                                                            15  11 00348      BRB     25$                                                     ;
                                                    A4  A7  9F 0034A 24$:    PUSHAB  FILE_MAX_DESC                                           ;
                                                    08  AE  9F 0034D          PUSHAB  CLI_DESC                                                ;
                                                        02  DD 00350          PUSHL   #2                                                      ;
                                                7E  132C    8F  3C 00352      MOVZWL  #4908, -(SP)                                            ;
                                                69          01  FB 00357      CALLS   #1, COPY$MSG_NUMBER                                     ;
                                                            50  DD 0035A      PUSHL   R0                                                      ;
                                                6B          04  FB 0035C      CALLS   #4, LIB$STOP                                            ;
                                    0000'   CF      0C  A6  D0 0035F 25$:    MOVL    COPY$CLI_STATUS+16, CURR_FILE_MAX_VALUE         ; 0909
                                                    B8  A7  9F 00365 26$:    PUSHAB  PROTECTION_DESC                                 ; 0915
                                                68          01  FB 00368      CALLS   #1, CLI$PRESENT                                         ;
```

```
      FF    A6          01          05          50  F0 0036B          INSV    R0, #5, #1, COPY$CLI_STATUS+3                    ;
                        11          50  E9 00371          BLBC    R0, 27$                                                  ;
                0000V   CF          00  FB 00374          CALLS   #0, PROTECTION_PARSE                                     ; 0921
                  12    A6    0000' CF  B0 00379          MOVW    CURR_PROTECTION_AND, COPY$CLI_STATUS+22                  ; 0922
                  10    A6    0000' CF  B0 0037F          MOVW    CURR_PROTECTION_OR, COPY$CLI_STATUS+20                   ; 0923
                                      04 00385 27$:       RET                                                              ; 0925

; Routine Size:  902 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  396     0926  1 GLOBAL ROUTINE COPY$GET_LOCAL_QUAL:  NOVALUE =          ! Retrieve local qualifiers from the CLI
;  397     0927  1
;  398     0928  1 !++
;  399     0929  1 ! FUNCTIONAL DESCRIPTION:
;  400     0930  1 !
;  401     0931  1 !       This routine retrieves the local command qualifiers from the command
;  402     0932  1 !       line.
;  403     0933  1 !
;  404     0934  1 ! FORMAL PARAMETERS:
;  405     0935  1 !
;  406     0936  1 !       None
;  407     0937  1 !
;  408     0938  1 ! IMPLICIT INPUTS:
;  409     0939  1 !
;  410     0940  1 !       None
;  411     0941  1 !
;  412     0942  1 ! IMPLICIT OUTPUTS:
;  413     0943  1 !
;  414     0944  1 !       COPY$CLI_STATUS - Relevant command and qualifier indicators set
;  415     0945  1 !
;  416     0946  1 ! ROUTINE VALUE:
;  417     0947  1 !
;  418     0948  1 !       None
;  419     0949  1 !
;  420     0950  1 ! SIDE EFFECTS:
;  421     0951  1 !
;  422     0952  1 !       None
;  423     0953  1 !
;  424     0954  1 !--
;  425     0955  1
;  426     0956  2     BEGIN
;  427     0957  2
;  428     0958  2     LOCAL
;  429     0959  2         rtn_status,                                     ! Status returned from external calls
;  430     0960  2         cli_desc :  $BBLOCK[ dsc$c_s_bln ]              ! Dynamic string descriptor, points to values
;  431     0961  2         ;                                               !   returned from calls to the CLI
;  432     0962  2
;  433     0963  2     BIND
;  434     0964  2
;  435     0965  2         MSG_DESC        = $DESCRIPTOR('can''t change quals in the middle of the command')
;  436     0966  2         ;
;  437     0967  2
;  438     0968  2
;  439     0969  2
;  440     0970  2
;  441     0971  2     ! Initialize descriptor.  Also, if a new output file is being created, then
;  442     0972  2     ! reset the current qualifier values to the global values.  This insures
;  443     0973  2     ! that if a previous local qualifier changed the value and, on this
;  444     0974  2     ! iteration, there is no local qualifier, the value used when creating the
;  445     0975  2     ! output file will be the one given by the global qualifier, not the
;  446     0976  2     ! previous local qualifier.
;  447     0977  2     !
;  448     0978  2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
;  449     0979  2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
;  450     0980  2
;  451     0981  2     IF not .outfile_open
;  452     0982  2     THEN
```

```
;  453          0983  3          BEGIN
;  454          0984  3          curr_allocation_value = .allocation_value;
;  455          0985  3          curr_extension_value  = .extension_value;
;  456          0986  3          curr_file_max_value   = .file_max_value;
;  457          0987  3          curr_protection_or    = .protection_or;
;  458          0988  3          curr_protection_and   = .protection_and;
;  459          0989  3          curr_volume_value     = .volume_value;
;  460          0990  2          END;
;  461          0991  2
;  462          0992  2
;  463          0993  2      ! Determine if this is a COPY or APPEND command.
;  464          0994  2      !
;  465          0995  2      IF NOT .append_command
;  466          0996  2      THEN
;  467          0997  3          BEGIN
;  468          0998  3
;  469          0999  3          ! It is a COPY command.  Parse the COPY specific qualifiers.
;  470          1000  3          !
;  471          1001  3          ! Initialize the flags for the local qualifier states.  Assume that
;  472          1002  3          ! there will be no local qualifier.  See if the qualifier is present.
;  473          1003  3          ! If is is, see if it is locally present or locally negated.  Set the
;  474          1004  3          ! appropriate flags.  The output file can not be open for the local
;  475          1005  3          ! qualifiers to be accepted.  The local qualifiers effect the attributes
;  476          1006  3          ! of the output file at creation time (i.e. allocation, location, etc.).
;  477          1007  3          ! These things cannot change once the file is open.  Therefore, if the
;  478          1008  3          ! output file is open and a local qualifier has been encountered, issue
;  479          1009  3          ! a warning, ignore the qualifier and continue processing.
;  480          1010  3          !
;  481          1011  3          ! /OVERLAY
;  482          1012  3          !
;  483          1013  3          loc_overlay_qual = neg_overlay_qual = FALSE;
;  484          1014  3          rtn_status = CLI$PRESENT( overlay_desc );
;  485          1015  3          SELECTONE .rtn_status OF
;  486          1016  3              SET
;  487          1017  3              [CLI$_LOCPRES] : IF NOT .outfile_open
;  488          1018  3                               THEN loc_overlay_qual = TRUE
;  489          1019  3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  490          1020  3
;  491          1021  3              [CLI$_LOCNEG]  : IF NOT .outfile_open
;  492          1022  3                               THEN neg_overlay_qual = TRUE
;  493          1023  3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  494          1024  3              TES;
;  495          1025  3
;  496          1026  3
;  497          1027  3          ! /REPLACE
;  498          1028  3          !
;  499          1029  3          loc_replace_qual = neg_replace_qual = FALSE;
;  500          1030  3          rtn_status = CLI$PRESENT( replace_desc );
;  501          1031  3          SELECTONE .rtn_status OF
;  502          1032  3              SET
;  503          1033  3              [CLI$_LOCPRES] : IF NOT .outfile_open
;  504          1034  3                               THEN loc_replace_qual = TRUE
;  505          1035  3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  506          1036  3
;  507          1037  3              [CLI$_LOCNEG]  : IF NOT .outfile_open
;  508          1038  3                               THEN neg_replace_qual = TRUE
;  509          1039  3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
```

```
;  510     1040  3            TES;
;  511     1041  3
;  512     1042  3
;  513     1043  3        ! /TRUNCATE
;  514     1044  3        !
;  515     1045  3        loc_truncate_qual = neg_truncate_qual = FALSE;
;  516     1046  3        rtn_status = CLI$PRESENT( truncate_desc );
;  517     1047  3        SELECTONE .rtn_status OF
;  518     1048  3            SET
;  519     1049  3            [CLI$_LOCPRES] : IF NOT .outfile_open
;  520     1050  3                             THEN loc_truncate_qual = TRUE
;  521     1051  3                             ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  522     1052  3
;  523     1053  3            [CLI$_LOCNEG]  : IF NOT .outfile_open
;  524     1054  3                             THEN neg_truncate_qual = TRUE
;  525     1055  3                             ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  526     1056  3            TES;
;  527     1057  3
;  528     1058  3        ! /VOLUME
;  529     1059  3        !
;  530     1060  3        loc_volume_qual = neg_volume_qual = FALSE;
;  531     1061  3        rtn_status = CLI$PRESENT( volume_desc );
;  532     1062  3        SELECTONE .rtn_status OF
;  533     1063  3            SET
;  534     1064  3            [CLI$_LOCPRES] : IF NOT .outfile_open
;  535     1065  3                             THEN
;  536     1066  3                                BEGIN
;  537     1067  4
;  538     1068  4                                ! Get the value and and convert it from a string into binary.
;  539     1069  4                                !
;  540     1070  4                                CLI$GET_VALUE( volume_desc, cli_desc );
;  541     1071  4                                IF NOT (rtn_status = [IB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  542     1072  5                                                                 .cli_desc[ DSC$A_POINTER ], curr_volume_value))
;  543     1073  5                                THEN
;  544     1074  4                                    PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, volume_desc );
;  545     1075  4                                loc_volume_qual = TRUE;
;  546     1076  4                                END
;  547     1077  4                             ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  548     1078  3
;  549     1079  3                [CLI$_LOCNEG]  : IF NOT .outfile_open
;  550     1080  3                                 THEN neg_volume_qual = TRUE
;  551     1081  3                                 ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  552     1082  3                TES;
;  553     1083  3            END;
;  554     1084  2
;  555     1085  2        ! Parse the qualifiers which are applicable to both commands.
;  556     1086  2        !
;  557     1087  2        ! /READ_CHECK, /WRITE_CHECK and /CONTIGUOUS
;  558     1088  2        !
;  559     1089  2        loc_read_chk_qual = neg_read_chk_qual = FALSE;
;  560     1090  2        rtn_status = CLI$PRESENT( read_check_desc );
;  561     1091  2        SELECTONE .rtn_status OF
;  562     1092  2            SET
;  563     1093  2            [CLI$_LOCPRES] : IF NOT .outfile_open
;  564     1094  2                             THEN loc_read_chk_qual = TRUE
;  565     1095  2                             ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  566     1096  2
```

```
;  567          1097  2
;  568          1098  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  569          1099  2                           THEN neg_read_chk_qual = TRUE
;  570          1100  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  571          1101  2          TES;
;  572          1102  2
;  573          1103  2
;  574          1104  2      loc_write_chk_qual = neg_write_chk_qual = FALSE;
;  575          1105  2      rtn_status = CLI$PRESENT( write_check_desc );
;  576          1106  2      SELECTONE .rtn_status OF
;  577          1107  2          SET
;  578          1108  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  579          1109  2                           THEN loc_write_chk_qual = TRUE
;  580          1110  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  581          1111  2
;  582          1112  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  583          1113  2                           THEN neg_write_chk_qual = TRUE
;  584          1114  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  585          1115  2          TES;
;  586          1116  2
;  587          1117  2
;  588          1118  2      loc_contig_qual = neg_contig_qual = FALSE;
;  589          1119  2      rtn_status = CLI$PRESENT( contiguous_desc );
;  590          1120  2      SELECTONE .rtn_status OF
;  591          1121  2          SET
;  592          1122  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  593          1123  2                           THEN loc_contig_qual = TRUE
;  594          1124  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  595          1125  2
;  596          1126  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  597          1127  2                           THEN neg_contig_qual = TRUE
;  598          1128  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  599          1129  2          TES;
;  600          1130  2
;  601          1131  2
;  602          1132  2      ! /ALLOCATION
;  603          1133  2      !
;  604          1134  2      loc_alloc_qual = neg_alloc_qual = FALSE;
;  605          1135  2      rtn_status = CLI$PRESENT( allocation_desc );
;  606          1136  2      SELECTONE .rtn_status OF
;  607          1137  2          SET
;  608          1138  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  609          1139  2                           THEN
;  610          1140  3                              BEGIN
;  611          1141  3
;  612          1142  3                              ! Get the value and and convert it from a string into binary.
;  613          1143  3                              !
;  614          1144  3                              CLI$GET_VALUE( allocation_desc, cli_desc );
;  615          1145  4                              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  616          1146  4                                                               .cli_desc[ DSC$A_POINTER ], curr_allocation_value))
;  617          1147  3                              THEN
;  618          1148  3                                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, allocation_desc );
;  619          1149  3                              loc_alloc_qual = TRUE;
;  620          1150  3                              END
;  621          1151  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  622          1152  2
;  623          1153  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
```

```
;  624          1154  2                          THEN neg_alloc_qual = TRUE
;  625          1155  2                          ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  626          1156  2          TES;
;  627          1157  2
;  628          1158  2
;  629          1159  2      ! /EXTENSION
;  630          1160  2      !
;  631          1161  2      loc_extend_qual = neg_extend_qual = FALSE;
;  632          1162  2      rtn_status = CLI$PRESENT( extension_desc );
;  633          1163  2      SELECTONE .rtn_status OF
;  634          1164  2          SET
;  635          1165  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  636          1166  2                           THEN
;  637          1167  3                              BEGIN
;  638          1168  3
;  639          1169  3                              ! Get the value and and convert it from a string into binary.
;  640          1170  3                              !
;  641          1171  3                              CLI$GET_VALUE( extension_desc, cli_desc );
;  642          1172  4                              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  643          1173  4                                                               .cli_desc[ DSC$A_POINTER ], curr_extension_value))
;  644          1174  3                              THEN
;  645          1175  3                                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, extension_desc );
;  646          1176  3                              loc_extend_qual = TRUE;
;  647          1177  3                              END
;  648          1178  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  649          1179  2
;  650          1180  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  651          1181  2                           THEN neg_extend_qual = TRUE
;  652          1182  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  653          1183  2          TES;
;  654          1184  2
;  655          1185  2
;  656          1186  2      ! /FILE_MAXIMUM
;  657          1187  2      !
;  658          1188  2      loc_file_max_qual = neg_file_max_qual = FALSE;
;  659          1189  2      rtn_status = CLI$PRESENT( file_max_desc );
;  660          1190  2      SELECTONE .rtn_status OF
;  661          1191  2          SET
;  662          1192  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  663          1193  2                           THEN
;  664          1194  3                              BEGIN
;  665          1195  3
;  666          1196  3                              ! Get the value and and convert it from a string into binary.
;  667          1197  3                              !
;  668          1198  3                              CLI$GET_VALUE( file_max_desc, cli_desc );
;  669          1199  4                              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  670          1200  4                                                               .cli_desc[ DSC$A_POINTER ], curr_file_max_value))
;  671          1201  3                              THEN
;  672          1202  3                                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, file_max_desc );
;  673          1203  3                              loc_file_max_qual = TRUE;
;  674          1204  3                              END
;  675          1205  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  676          1206  2
;  677          1207  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  678          1208  2                           THEN neg_extend_qual = TRUE
;  679          1209  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  680          1210  2          TES;
```

```
;  681          1211  2
;  682          1212  2
;  683          1213  2      ! /PROTECTION
;  684          1214  2      !
;  685          1215  2      loc_protect_qual = neg_protect_qual = FALSE;
;  686          1216  2      rtn_status = CLI$PRESENT( protection_desc );
;  687          1217  2      SELECTONE .rtn_status OF
;  688          1218  2          SET
;  689          1219  2          [CLI$_LOCPRES] : IF NOT .outfile_open
;  690          1220  2                           THEN
;  691          1221  3                              BEGIN
;  692          1222  3
;  693          1223  3                              ! Parse the keyword values and save the results.
;  694          1224  3                              !
;  695          1225  3                              protection_parse();
;  696          1226  3                              loc_protect_qual = TRUE;
;  697          1227  3                              END
;  698          1228  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  699          1229  2
;  700          1230  2          [CLI$_LOCNEG]  : IF NOT .outfile_open
;  701          1231  2                           THEN neg_protect_qual = TRUE
;  702          1232  2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
;  703          1233  2          TES;
;  704          1234  2
;  705          1235  1      END;                                        ! routine COPY$GET_GLOBAL_QUAL


                                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

75 71 20 65 67 6E 61 68 63 20 74 22 6E 61 63  00110 P.ABF:  .ASCII  \can''t change quals in the middle of the \  ;
64 64 69 6D 20 65 68 74 20 6E 69 20 73 6C 61  0011F                                                            ;
                20 65 68 74 20 66 6F 20 65 6C  0012E                                                            ;
                   64 6E 61 6D 6D 6F 63        00138          .ASCII  \command\                              ;
                                               0013F          .BLKB   1                                      ;
                                    0000002F   00140 P.ABE:   .LONG   47                                     ;
                                    00000000'  00144          .ADDRESS P.ABF                                 ;

                                               MSG_DESC=            P.ABE


                                                                        .PSECT  $CODE$,NOWRT,2

                                         0FFC  00000          .ENTRY  COPY$GET_LOCAL_QUAL, Save R2,R3,R4,R5,R6,-  ; 0926
                                                                      R7,R8,R9,R10,R11                           ;
                                 5B 00000000G   8F   D0 00002          MOVL    #CLI$_LOCPRES, R11               ;
                                 5A      0000G  CF   9E 00009          MOVAB   COPY$SEM_STATUS, R10             ;
                                 59      0000G  CF   9E 0000E          MOVAB   COPY$MSG_NUMBER, R9              ;
                                 58 00000000G   00   9E 00013          MOVAB   LIB$SIGNAL, R8                   ;
                                 57      0000'  CF   9E 0001A          MOVAB   MSG_DESC, R7                     ;
                                 56      0000G  CF   9E 0001F          MOVAB   COPY$CLI_STATUS+4, R6            ;
                                 5E             08   C2 00024          SUBL2   #8, SP                           ;
              08          00     6E             0C   2C 00027          MOVC5   #0, (SP), #0, #8, CLI_DESC       ; 0978
                                                 6E        0002C                                                ;
                        03   AE             02   90 0002D          MOVB    #2, CLI_DESC+3                   ; 0979
                     1E 02   AA             01   E0 00031          BBS     #1, COPY$SEM_STATUS+2, 1$        ; 0981
                0000' CF        04   A6       7D 00036          MOVQ    COPY$CLI_STATUS+8, CURR_ALLOCATION_VALUE ; 0984
```

```
                0000'  CF        0C    A6  D0 0003C          MOVL    COPY$CLI_STATUS+16, CURR_FILE_MAX_VALUE     ; 0986
                0000'  CF        10    A6  B0 00042          MOVW    COPY$CLI_STATUS+20, CURR_PROTECTION_OR      ; 0987
                0000'  CF        12    A6  B0 00048          MOVW    COPY$CLI_STATUS+22, CURR_PROTECTION_AND     ; 0988
                0000'  CF        14    A6  D0 0004E          MOVL    COPY$CLI_STATUS+24, CURR_VOLUME_VALUE       ; 0989
                       03        FC    A6  E9 00054 1$:      BLBC    COPY$CLI_STATUS, 2$                         ; 0995
                                     017C  31 00058          BRW     17$
                    01   A6            03  8A 0005B 2$:      BICB2   #3, COPY$CLI_STATUS+5                       ; 1013
                                  98   A7  9F 0005F          PUSHAB  OVERLAY_DESC                                ; 1014
                0000G  CF            01  FB 00062          CALLS   #1, CLI$PRESENT
                       52            50  D0 00067          MOVL    R0, RTN_STATUS
                       5B            52  D1 0006A          CMPL    RTN_STATUS, R11                             ; 1017
                                     0B  12 0006D          BNEQ    3$
            1A      02   AA            01  E0 0006F          BBS     #1, COPY$SEM_STATUS+2, 4$
                    01   A6            01  88 00074          BISB2   #1, COPY$CLI_STATUS+5                       ; 1018
                                     25  11 00078          BRB     5$
        00000000G   8F               52  D1 0007A 3$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                    ; 1021
                                     1C  12 00081          BNEQ    5$
            06      02   AA            01  E0 00083          BBS     #1, COPY$SEM_STATUS+2, 4$
                    01   A6            02  88 00088          BISB2   #2, COPY$CLI_STATUS+5                       ; 1022
                                     11  11 0008C          BRB     5$
                                     57  DD 0008E 4$:      PUSHL   R7                                          ; 1023
                                     01  DD 00090          PUSHL   #1
                       7E        10BB  8F  3C 00092          MOVZWL  #4283, -(SP)
                       69            01  FB 00097          CALLS   #1, COPY$MSG_NUMBER
                                     50  DD 0009A          PUSHL   R0
                       68            03  FB 0009C          CALLS   #3, LIB$SIGNAL
                    02   A6            0C  8A 0009F 5$:      BICB2   #12, COPY$CLI_STATUS+6                      ; 1029
                                  C8   A7  9F 000A3          PUSHAB  REPLACE_DESC                                ; 1030
                0000G  CF            01  FB 000A6          CALLS   #1, CLI$PRESENT
                       52            50  D0 000AB          MOVL    R0, RTN_STATUS
                       5B            52  D1 000AE          CMPL    RTN_STATUS, R11                             ; 1033
                                     0B  12 000B1          BNEQ    6$
            1A      02   AA            01  E0 000B3          BBS     #1, COPY$SEM_STATUS+2, 7$
                    02   A6            04  88 000B8          BISB2   #4, COPY$CLI_STATUS+6                       ; 1034
                                     25  11 000BC          BRB     8$
        00000000G   8F               52  D1 000BE 6$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                    ; 1037
                                     1C  12 000C5          BNEQ    8$
            06      02   AA            01  E0 000C7          BBS     #1, COPY$SEM_STATUS+2, 7$
                    02   A6            08  88 000CC          BISB2   #8, COPY$CLI_STATUS+6                       ; 1038
                                     11  11 000D0          BRB     8$
                                     57  DD 000D2 7$:      PUSHL   R7                                          ; 1039
                                     01  DD 000D4          PUSHL   #1
                       7E        10BB  8F  3C 000D6          MOVZWL  #4283, -(SP)
                       69            01  FB 000DB          CALLS   #1, COPY$MSG_NUMBER
                                     50  DD 000DE          PUSHL   R0
                       68            03  FB 000E0          CALLS   #3, LIB$SIGNAL
                    01   A6      0180  8F  AA 000E3 8$:      BICW2   #384, COPY$CLI_STATUS+5                     ; 1045
                                  B8   A7  9F 000E9          PUSHAB  TRUNCATE_DESC                               ; 1046
                0000G  CF            01  FB 000EC          CALLS   #1, CLI$PRESENT
                       52            50  D0 000F1          MOVL    R0, RTN_STATUS
                       5B            52  D1 000F4          CMPL    RTN_STATUS, R11                             ; 1049
                                     0C  12 000F7          BNEQ    9$
            1B      02   AA            01  E0 000F9          BBS     #1, COPY$SEM_STATUS+2, 10$
                    01   A6        80  8F  88 000FE          BISB2   #128, COPY$CLI_STATUS+5                     ; 1050
                                     25  11 00103          BRB     11$
        00000000G   8F               52  D1 00105 9$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                    ; 1053
                                     1C  12 0010C          BNEQ    11$
```

```
                06          02  AA            01  E0 0010E          BBS     #1, COPY$SEM_STATUS+2, 10$                      ;
                            02  A6            01  88 00113          BISB2   #1, COPY$CLI_STATUS+6                          ; 1054
                                              11  11 00117          BRB     11$                                            ;
                                              57  DD 00119 10$:     PUSHL   R7                                             ; 1055
                                              01  DD 0011B          PUSHL   #1                                             ;
                            7E      10BB      8F  3C 0011D          MOVZWL  #4283, -(SP)                                   ;
                            69                01  FB 00122          CALLS   #1, COPY$MSG_NUMBER                            ;
                                              50  DD 00125          PUSHL   R0                                             ;
                            68                03  FB 00127          CALLS   #3, LIB$SIGNAL                                 ;
                              01  A6          18  8A 0012A 11$:     BICB2   #24, COPY$CLI_STATUS+5                         ; 1061
                                A8  A7        9F  0012E          PUSHAB  VOLUME_DESC                                    ; 1062
                          0000G CF            01  FB 00131          CALLS   #1, CLI$PRESENT                                ;
                                52            50  D0 00136          MOVL    R0, RTN_STATUS                                 ;
                                5B            52  D1 00139          CMPL    RTN_STATUS, R11                                ; 1065
                                              74  12 0013C          BNEQ    14$                                            ;
                78          02  AA            01  E0 0013E          BBS     #1, COPY$SEM_STATUS+2, 15$                     ;
                                              5E  DD 00143          PUSHL   SP                                             ; 1071
                                A8  A7        9F  00145          PUSHAB  VOLUME_DESC                                    ;
                          0000G CF            02  FB 00148          CALLS   #2, CLI$GET_VALUE                              ;
                                      0000'   CF  9F 0014D          PUSHAB  CURR_VOLUME_VALUE                              ; 1072
                                        08    AE  DD 00151          PUSHL   CLI_DESC+4                                     ; 1073
                            7E          08    AE  3C 00154          MOVZWL  CLI_DESC, -(SP)                                ; 1072
                   00000000G  00            03  FB 00158          CALLS   #3, LIB$CVT_DTB                                ;
                                52            50  D0 0015F          MOVL    R0, RTN_STATUS                                 ;
                                47            52  E8 00162          BLBS    RTN_STATUS, 13$                                ;
                                7E      132C  8F  3C 00165          MOVZWL  #4908, -(SP)                                   ; 1075
                                69            01  FB 0016A          CALLS   #1, COPY$MSG_NUMBER                            ;
          7E                    50            01  7A 0016D          EMUL    #1, R0, #0, -(SP)                              ;
          50    50              8E            08  7B 00172          EDIV    #8, (SP)+, R0, R0                              ;
                                04            50  D1 00177          CMPL    R0, #4                                         ;
                                              17  13 0017A          BEQL    12$                                            ;
                                A8  A7        9F  0017C          PUSHAB  VOLUME_DESC                                    ;
                                04  AE        9F  0017F          PUSHAB  CLI_DESC                                       ;
                                              02  DD 00182          PUSHL   #2                                             ;
                            7E      132C      8F  3C 00184          MOVZWL  #4908, -(SP)                                   ;
                            69                01  FB 00189          CALLS   #1, COPY$MSG_NUMBER                            ;
                                              50  DD 0018C          PUSHL   R0                                             ;
                            68                04  FB 0018E          CALLS   #4, LIB$SIGNAL                                 ;
                                              19  11 00191          BRB     13$                                            ;
                                A8  A7        9F  00193 12$:     PUSHAB  VOLUME_DESC                                    ;
                                04  AE        9F  00196          PUSHAB  CLI_DESC                                       ;
                                              02  DD 00199          PUSHL   #2                                             ;
                            7E      132C      8F  3C 0019B          MOVZWL  #4908, -(SP)                                   ;
                            69                01  FB 001A0          CALLS   #1, COPY$MSG_NUMBER                            ;
                                              50  DD 001A3          PUSHL   R0                                             ;
                   00000000G  00            04  FB 001A5          CALLS   #4, LIB$STOP                                   ;
                              01  A6          08  88 001AC 13$:     BISB2   #8, COPY$CLI_STATUS+5                          ; 1076
                                              25  11 001B0          BRB     17$                                            ; 1065
                   00000000G  8F                52  D1 001B2 14$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                       ; 1080
                                              1C  12 001B9          BNEQ    17$                                            ;
          06          02  AA                  01  E0 001BB 15$:     BBS     #1, COPY$SEM_STATUS+2, 16$                     ;
                      01  A6                  10  88 001C0          BISB2   #16, COPY$CLI_STATUS+5                         ; 1081
                                              11  11 001C4          BRB     17$                                            ;
                                              57  DD 001C6 16$:     PUSHL   R7                                             ; 1082
                                              01  DD 001C8          PUSHL   #1                                             ;
                            7E      10BB      8F  3C 001CA          MOVZWL  #4283, -(SP)                                   ;
                            69                01  FB 001CF          CALLS   #1, COPY$MSG_NUMBER                            ;
```

```
                                50  DD 001D2          PUSHL   R0                                              ;
                            68  03  FB 001D4          CALLS   #3, LIB$SIGNAL                                  ;
                            66  06  8A 001D7 17$:     BICB2   #6, COPY$CLI_STATUS+4                           ; 1090
                        FF74    C7  9F 001DA          PUSHAB  READ_CHECK_DESC                                 ; 1091
                    0000G   CF  01  FB 001DE          CALLS   #1, CLI$PRESENT                                 ;
                            52  50  D0 001E3          MOVL    R0, RTN_STATUS                                  ;
                            5B  52  D1 001E6          CMPL    RTN_STATUS, R11                                 ; 1094
                                0A  12 001E9          BNEQ    18$                                             ;
                18      02  AA  01  E0 001EB          BBS     #1, COPY$SEM_STATUS+2, 19$                      ;
                            66  02  88 001F0          BISB2   #2, COPY$CLI_STATUS+4                           ; 1095
                                24  11 001F3          BRB     20$                                             ;
                00000000G   8F  52  D1 001F5 18$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                        ; 1098
                                1B  12 001FC          BNEQ    20$                                             ;
                05      02  AA  01  E0 001FE          BBS     #1, COPY$SEM_STATUS+2, 19$                      ;
                            66  04  88 00203          BISB2   #4, COPY$CLI_STATUS+4                           ; 1099
                                11  11 00206          BRB     20$                                             ;
                                57  DD 00208 19$:     PUSHL   R7                                              ; 1100
                                01  DD 0020A          PUSHL   #1                                              ;
                            7E  10BB    8F  3C 0020C  MOVZWL  #4283, -(SP)                                    ;
                            69  01  FB 00211          CALLS   #1, COPY$MSG_NUMBER                             ;
                                50  DD 00214          PUSHL   R0                                              ;
                            68  03  FB 00216          CALLS   #3, LIB$SIGNAL                                  ;
                            66      60  8F  8A 00219 20$:     BICB2   #96, COPY$CLI_STATUS+4                  ; 1104
                                    88  A7  9F 0021D  PUSHAB  WRITE_CHECK_DESC                               ; 1105
                    0000G   CF  01  FB 00220          CALLS   #1, CLI$PRESENT                                 ;
                            52  50  D0 00225          MOVL    R0, RTN_STATUS                                  ;
                            5B  52  D1 00228          CMPL    RTN_STATUS, R11                                 ; 1108
                                0A  12 0022B          BNEQ    21$                                             ;
                19      02  AA  01  E0 0022D          BBS     #1, COPY$SEM_STATUS+2, 22$                      ;
                            66  20  88 00232          BISB2   #32, COPY$CLI_STATUS+4                          ; 1109
                                25  11 00235          BRB     23$                                             ;
                00000000G   8F  52  D1 00237 21$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                        ; 1112
                                1C  12 0023E          BNEQ    23$                                             ;
                06      02  AA  01  E0 00240          BBS     #1, COPY$SEM_STATUS+2, 22$                      ;
                            66      40  8F  88 00245  BISB2   #64, COPY$CLI_STATUS+4                          ; 1113
                                    11  11 00249      BRB     23$                                             ;
                                    57  DD 0024B 22$: PUSHL   R7                                              ; 1114
                                    01  DD 0024D      PUSHL   #1                                              ;
                                7E      10BB    8F  3C 0024F  MOVZWL  #4283, -(SP)                            ;
                                69  01  FB 00254          CALLS   #1, COPY$MSG_NUMBER                         ;
                                    50  DD 00257      PUSHL   R0                                              ;
                                68  03  FB 00259          CALLS   #3, LIB$SIGNAL                              ;
                            FE  A6      60  8F  8A 0025C 23$:     BICB2   #96, COPY$CLI_STATUS+2              ; 1118
                                    FF24    C7  9F 00261  PUSHAB  CONTIGUOUS_DESC                             ; 1119
                    0000G   CF  01  FB 00265          CALLS   #1, CLI$PRESENT                                 ;
                            52  50  D0 0026A          MOVL    R0, RTN_STATUS                                  ;
                            5B  52  D1 0026D          CMPL    RTN_STATUS, R11                                 ; 1122
                                0B  12 00270          BNEQ    24$                                             ;
                1B      02  AA  01  E0 00272          BBS     #1, COPY$SEM_STATUS+2, 25$                      ;
                        FE  A6  20  88 00277          BISB2   #32, COPY$CLI_STATUS+2                          ; 1123
                                26  11 0027B          BRB     26$                                             ;
                00000000G   8F  52  D1 0027D 24$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                        ; 1126
                                1D  12 00284          BNEQ    26$                                             ;
                07      02  AA  01  E0 00286          BBS     #1, COPY$SEM_STATUS+2, 25$                      ;
                        FE  A6      40  8F  88 0028B  BISB2   #64, COPY$CLI_STATUS+2                          ; 1127
                                    11  11 00290      BRB     26$                                             ;
                                    57  DD 00292 25$: PUSHL   R7                                              ; 1128
```

```
                                01 DD 00294          PUSHL    #1                                  :
                  7E     10BB   8F 3C 00296          MOVZWL   #4283, -(SP)                        :
                  69            01 FB 0029B          CALLS    #1, COPY$MSG_NUMBER                 :
                                50 DD 0029E          PUSHL    R0                                  :
                  68            03 FB 002A0          CALLS    #3, LIB$SIGNAL                      :
              FE  A6            06 8A 002A3 26$:     BICB2    #6, COPY$CLI_STATUS+2               : 1134
                         FF10   C7 9F 002A7          PUSHAB   ALLOCATION_DESC                     : 1135
           0000G  CF            01 FB 002AB          CALLS    #1, CLI$PRESENT                     :
                  52            50 D0 002B0          MOVL     R0, RTN_STATUS                      :
                  5B            52 D1 002B3          CMPL     RTN_STATUS, R11                     : 1138
                                77 12 002B6          BNEQ     29$                                 :
        7B    02  AA            01 E0 002B8          BBS      #1, COPY$SEM_STATUS+2, 30$          :
                                5E DD 002BD          PUSHL    SP                                  : 1144
                         FF10   C7 9F 002BF          PUSHAB   ALLOCATION_DESC                     :
           0000G  CF            02 FB 002C3          CALLS    #2, CLI$GET_VALUE                   :
                         0000'  CF 9F 002C8          PUSHAB   CURR_ALLOCATION_VALUE               : 1145
                           08   AE DD 002CC          PUSHL    CLI_DESC+4                          : 1146
                  7E       08   AE 3C 002CF          MOVZWL   CLI_DESC, -(SP)                     : 1145
       00000000G  00            03 FB 002D3          CALLS    #3, LIB$CVT_DTB                     :
                  52            50 D0 002DA          MOVL     R0, RTN_STATUS                      :
                  49            52 E8 002DD          BLBS     RTN_STATUS, 28$                     :
                  7E     132C   8F 3C 002E0          MOVZWL   #4908, -(SP)                        : 1148
                  69            01 FB 002E5          CALLS    #1, COPY$MSG_NUMBER                 :
 7E     00        50            01 7A 002E8          EMUL     #1, R0, #0, -(SP)                   :
 50     50        8E            08 7B 002ED          EDIV     #8, (SP)+, R0, R0                   :
                  04            50 D1 002F2          CMPL     R0, #4                              :
                                18 13 002F5          BEQL     27$                                 :
                         FF10   C7 9F 002F7          PUSHAB   ALLOCATION_DESC                     :
                           04   AE 9F 002FB          PUSHAB   CLI_DESC                            :
                                02 DD 002FE          PUSHL    #2                                  :
                  7E     132C   8F 3C 00300          MOVZWL   #4908, -(SP)                        :
                  69            01 FB 00305          CALLS    #1, COPY$MSG_NUMBER                 :
                                50 DD 00308          PUSHL    R0                                  :
                  68            04 FB 0030A          CALLS    #4, LIB$SIGNAL                      :
                                1A 11 0030D          BRB      28$                                 :
                         FF10   C7 9F 0030F 27$:     PUSHAB   ALLOCATION_DESC                     :
                           04   AE 9F 00313          PUSHAB   CLI_DESC                            :
                                02 DD 00316          PUSHL    #2                                  :
                  7E     132C   8F 3C 00318          MOVZWL   #4908, -(SP)                        :
                  69            01 FB 0031D          CALLS    #1, COPY$MSG_NUMBER                 :
                                50 DD 00320          PUSHL    R0                                  :
       00000000G  00            04 FB 00322          CALLS    #4, LIB$STOP                        :
              FE  A6            02 88 00329 28$:     BISB2    #2, COPY$CLI_STATUS+2               : 1149
                                25 11 0032D          BRB      32$                                 : 1138
       00000000G  8F            52 D1 0032F 29$:     CMPL     RTN_STATUS, #CLI$_LOCNEG            : 1153
                                1C 12 00336          BNEQ     32$                                 :
        06    02  AA            01 E0 00338 30$:     BBS      #1, COPY$SEM_STATUS+2, 31$          :
              FE  A6            04 88 0033D          BISB2    #4, COPY$CLI_STATUS+2               : 1154
                                11 11 00341          BRB      32$                                 :
                                57 DD 00343 31$:     PUSHL    R7                                  : 1155
                                01 DD 00345          PUSHL    #1                                  :
                  7E     10BB   8F 3C 00347          MOVZWL   #4283, -(SP)                        :
                  69            01 FB 0034C          CALLS    #1, COPY$MSG_NUMBER                 :
                                50 DD 0034F          PUSHL    R0                                  :
                  68            03 FB 00351          CALLS    #3, LIB$SIGNAL                      :
              FF  A6            03 8A 00354 32$:     BICB2    #3, COPY$CLI_STATUS+3               : 1161
                         FF38   C7 9F 00358          PUSHAB   EXTENSION_DESC                      : 1162
```

```
                        0000G  CF              01 FB 0035C          CALLS   #1, CLI$PRESENT                                  ;
                               52               50 D0 00361          MOVL    R0, RTN_STATUS                                  ;
                               5B               52 D1 00364          CMPL    RTN_STATUS, R11                                 ; 1165
                                                77 12 00367          BNEQ    35$                                             ;
                    7B    02   AA               01 E0 00369          BBS     #1, COPY$SEM_STATUS+2, 36$                      ;
                                                5E DD 0036E          PUSHL   SP                                              ; 1171
                                          FF38  C7 9F 00370          PUSHAB  EXTENSION_DESC                                  ;
                        0000G  CF               02 FB 00374          CALLS   #2, CLI$GET_VALUE                               ;
                                          0000' CF 9F 00379          PUSHAB  CURR_EXTENSION_VALUE                            ; 1172
                                            08  AE DD 0037D          PUSHL   CLI_DESC+4                                      ; 1173
                               7E           08  AE 3C 00380          MOVZWL  CLI_DESC, -(SP)                                 ; 1172
                 00000000G     00               03 FB 00384          CALLS   #3, LIB$CVT_DTB                                 ;
                               52               50 D0 0038B          MOVL    R0, RTN_STATUS                                  ;
                               49               52 E8 0038E          BLBS    RTN_STATUS, 34$                                 ;
                               7E          132C  8F 3C 00391          MOVZWL  #4908, -(SP)                                    ; 1175
                               69               01 FB 00396          CALLS   #1, COPY$MSG_NUMBER                             ;
          7E        00         50               01 7A 00399          EMUL    #1, R0, #0, -(SP)                               ;
          50        50         8E               08 7B 0039E          EDIV    #8, (SP)+, R0, R0                               ;
                               04               50 D1 003A3          CMPL    R0, #4                                          ;
                                                18 13 003A6          BEQL    33$                                             ;
                                          FF38  C7 9F 003A8          PUSHAB  EXTENSION_DESC                                  ;
                                            04  AE 9F 003AC          PUSHAB  CLI_DESC                                        ;
                                                02 DD 003AF          PUSHL   #2                                              ;
                               7E          132C  8F 3C 003B1          MOVZWL  #4908, -(SP)                                    ;
                               69               01 FB 003B6          CALLS   #1, COPY$MSG_NUMBER                             ;
                                                50 DD 003B9          PUSHL   R0                                              ;
                               68               04 FB 003BB          CALLS   #4, LIB$SIGNAL                                  ;
                                                1A 11 003BE          BRB     34$                                             ;
                                          FF38  C7 9F 003C0 33$:     PUSHAB  EXTENSION_DESC                                  ;
                                            04  AE 9F 003C4          PUSHAB  CLI_DESC                                        ;
                                                02 DD 003C7          PUSHL   #2                                              ;
                               7E          132C  8F 3C 003C9          MOVZWL  #4908, -(SP)                                    ;
                               69               01 FB 003CE          CALLS   #1, COPY$MSG_NUMBER                             ;
                                                50 DD 003D1          PUSHL   R0                                              ;
                 00000000G     00               04 FB 003D3          CALLS   #4, LIB$STOP                                    ;
                         FF    A6               01 88 003DA 34$:     BISB2   #1, COPY$CLI_STATUS+3                           ; 1176
                                                25 11 003DE          BRB     38$                                             ; 1165
                 00000000G     8F               52 D1 003E0 35$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                        ; 1180
                                                1C 12 003E7          BNEQ    38$                                             ;
          06        02    AA                     01 E0 003E9 36$:     BBS     #1, COPY$SEM_STATUS+2, 37$                      ;
                    FF    A6                     02 88 003EE          BISB2   #2, COPY$CLI_STATUS+3                           ; 1181
                                                11 11 003F2          BRB     38$                                             ;
                                                57 DD 003F4 37$:     PUSHL   R7                                              ; 1182
                                                01 DD 003F6          PUSHL   #1                                              ;
                               7E          10BB  8F 3C 003F8          MOVZWL  #4283, -(SP)                                    ;
                               69               01 FB 003FD          CALLS   #1, COPY$MSG_NUMBER                             ;
                                                50 DD 00400          PUSHL   R0                                              ;
                               68               03 FB 00402          CALLS   #3, LIB$SIGNAL                                  ;
                         FF    A6               18 8A 00405 38$:     BICB2   #24, COPY$CLI_STATUS+3                          ; 1188
                                          FF4C  C7 9F 00409          PUSHAB  FILE_MAX_DESC                                   ; 1189
                        0000G  CF               01 FB 0040D          CALLS   #1, CLI$PRESENT                                 ;
                               52               50 D0 00412          MOVL    R0, RTN_STATUS                                  ;
                               5B               52 D1 00415          CMPL    RTN_STATUS, R11                                 ; 1192
                                                77 12 00418          BNEQ    41$                                             ;
          7B        02    AA                     01 E0 0041A          BBS     #1, COPY$SEM_STATUS+2, 42$                      ;
                                                5E DD 0041F          PUSHL   SP                                              ; 1198
                                          FF4C  C7 9F 00421          PUSHAB  FILE_MAX_DESC                                   ;
```

```
                        0000G  CF            02  FB 00425          CALLS   #2, CLI$GET_VALUE                                    ;
                               0000'  CF      9F  0042A          PUSHAB  CURR_FILE_MAX_VALUE                                  ; 1199
                                 08   AE      DD  0042E          PUSHL   CLI_DESC+4                                           ; 1200
                                      7E        08  AE  3C 00431          MOVZWL  CLI_DESC, -(SP)                              ; 1199
                    00000000G         00        03  FB 00435          CALLS   #3, LIB$CVT_DTB                                  ;
                                      52        50  D0 0043C          MOVL    R0, RTN_STATUS                                   ;
                                      49        52  E8 0043F          BLBS    RTN_STATUS, 40$                                  ;
                                      7E   132C  8F  3C 00442          MOVZWL  #4908, -(SP)                                     ; 1202
                                      69        01  FB 00447          CALLS   #1, COPY$MSG_NUMBER                              ;
                7E          00        50        01  7A 0044A          EMUL    #1, R0, #0, -(SP)                                ;
                50          50        8E        08  7B 0044F          EDIV    #8, (SP)+, R0, R0                                ;
                                      04        50  D1 00454          CMPL    R0, #4                                           ;
                                      18        13  00457          BEQL    39$                                              ;
                                 FF4C   C7  9F  00459          PUSHAB  FILE_MAX_DESC                                        ;
                                   04   AE  9F  0045D          PUSHAB  CLI_DESC                                             ;
                                        02  DD  00460          PUSHL   #2                                                   ;
                                      7E   132C  8F  3C 00462          MOVZWL  #4908, -(SP)                                     ;
                                      69        01  FB 00467          CALLS   #1, COPY$MSG_NUMBER                              ;
                                      50        DD  0046A          PUSHL   R0                                               ;
                                      68        04  FB 0046C          CALLS   #4, LIB$SIGNAL                                   ;
                                      1A        11  0046F          BRB     40$                                              ;
                                 FF4C   C7  9F  00471  39$:       PUSHAB  FILE_MAX_DESC                                        ;
                                   04   AE  9F  00475          PUSHAB  CLI_DESC                                             ;
                                        02  DD  00478          PUSHL   #2                                                   ;
                                      7E   132C  8F  3C 0047A          MOVZWL  #4908, -(SP)                                     ;
                                      69        01  FB 0047F          CALLS   #1, COPY$MSG_NUMBER                              ;
                                      50        DD  00482          PUSHL   R0                                               ;
                    00000000G         00        04  FB 00484          CALLS   #4, LIB$STOP                                     ;
                           FF   A6              08  88 0048B  40$:       BISB2   #8, COPY$CLI_STATUS+3                            ; 1203
                                                25  11 0048F          BRB     44$                                              ; 1192
                    00000000G   8F               52  D1 00491  41$:       CMPL    RTN_STATUS, #CLI$_LOCNEG                         ; 1207
                                                 1C  12 00498          BNEQ    44$                                              ;
                06         02   AA               01  E0 0049A  42$:       BBS     #1, COPY$SEM_STATUS+2, 43$                       ;
                           FF   A6              02  88 0049F          BISB2   #2, COPY$CLI_STATUS+3                            ; 1208
                                                11  11 004A3          BRB     44$                                              ;
                                                57  DD 004A5  43$:       PUSHL   R7                                               ; 1209
                                                01  DD 004A7          PUSHL   #1                                               ;
                                      7E   10BB  8F  3C 004A9          MOVZWL  #4283, -(SP)                                     ;
                                      69        01  FB 004AE          CALLS   #1, COPY$MSG_NUMBER                              ;
                                      50        DD  004B1          PUSHL   R0                                               ;
                                      68        03  FB 004B3          CALLS   #3, LIB$SIGNAL                                   ;
                           FF   A6         C0   8F  8A 004B6  44$:       BICB2   #192, COPY$CLI_STATUS+3                          ; 1215
                                      FF60   C7  9F  004BB          PUSHAB  PROTECTION_DESC                                      ; 1216
                                0000G   CF        01  FB 004BF          CALLS   #1, CLI$PRESENT                                  ;
                                      52        50  D0 004C4          MOVL    R0, RTN_STATUS                                   ;
                                      5B        52  D1 004C7          CMPL    RTN_STATUS, R11                                  ; 12'9
                                                10  12 004CA          BNEQ    45$                                              ;
                1F         02   AA               01  E0 004CC          BBS     #1, COPY$SEM_STATUS+2, 46$                       ;
                                0000V   CF        00  FB 004D1          CALLS   #0, PROTECTION_PARSE                             ; 1225
                           FF   A6         40   8F  88 004D6          BISB2   #64, COPY$CLI_STATUS+3                           ; 1226
                                                04  004DB          RET                                                      ; 1219
                    00000000G   8F               52  D1 004DC  45$:       CMPL    RTN_STATUS, #CLI$_LOCNEG                         ; 1230
                                                 1C  12 004E3          BNEQ    47$                                              ;
                06         02   AA               01  E0 004E5          BBS     #1, COPY$SEM_STATUS+2, 46$                       ;
                           FF   A6         80   8F  88 004EA          BISB2   #128, COPY$CLI_STATUS+3                          ; 1231
                                                04  004EF          RET                                                      ;
                                                57  DD 004F0  46$:       PUSHL   R7                                               ; 1232
```

```
                                01  DD 004F2          PUSHL   #1                                                      ;
                          7E    8F  3C 004F4          MOVZWL  #4283, -(SP)                                            ;
                          69    01  FB 004F9          CALLS   #1, COPY$MSG_NUMBER                                     ;
                                50  DD 004FC          PUSHL   R0                                                      ;
                          68    03  FB 004FE          CALLS   #3, LIB$SIGNAL                                          ;
                                04 00501 47$:         RET                                                             ; 1235

; Routine Size:  1282 bytes,    Routine Base:  $CODE$ + 0386
```

```
;  707     1236  1 ROUTINE PROTECTION_PARSE : NOVALUE =
;  708     1237  1
;  709     1238  1 !++
;  710     1239  1 ! FUNCTIONAL DESCRIPTION:
;  711     1240  1 !
;  712     1241  1 !       This routine parses a PROTECTION qualifier value.
;  713     1242  1 !
;  714     1243  1 ! FORMAL PARAMETERS:
;  715     1244  1 !
;  716     1245  1 !       NONE
;  717     1246  1 !
;  718     1247  1 ! IMPLICIT INPUTS:
;  719     1248  1 !
;  720     1249  1 !       NONE
;  721     1250  1 !
;  722     1251  1 ! IMPLICIT OUTPUTS:
;  723     1252  1 !
;  724     1253  1 !       CURR_PROTECTION_OR  - Protection mask storage
;  725     1254  1 !       CURR_PROTECTION_AND - Protection mask storage
;  726     1255  1 !
;  727     1256  1 ! ROUTINE VALUE:
;  728     1257  1 !
;  729     1258  1 !       None
;  730     1259  1 !
;  731     1260  1 ! SIDE EFFECTS:
;  732     1261  1 !
;  733     1262  1 !       None
;  734     1263  1 !
;  735     1264  1 !--
;  736     1265  1
;  737     1266  2     BEGIN
;  738     1267  2
;  739     1268  2     MAP
;  740     1269  2         CURR_PROTECTION_OR  :  $BBLOCK[ 2 ],          ! Protection mask
;  741     1270  2         CURR_PROTECTION_AND :  $BBLOCK[ 2 ];          ! Protection mask
;  742     1271  2
;  743     1272  2     LOCAL
;  744     1273  2         RTN_STATUS,                                   ! Keyword lookup completion code
;  745     1274  2         CLI_DESC :                                    ! Descriptor which points to the keyword value
;  746     1275  2             $BBLOCK[ DSC$C_S_BLN ],
;  747     1276  2         KEY_DISP;                                     ! Displacement of keyword nibble in XAB$W_PRO
;  748     1277  2
;  749     1278  2     BIND
;  750     1279  2         SYSTEM_DESC = $DESCRIPTOR('SYSTEM'),
;  751     1280  2         OWNER_DESC  = $DESCRIPTOR('OWNER'),
;  752     1281  2         GROUP_DESC  = $DESCRIPTOR('GROUP'),
;  753     1282  2         WORLD_DESC  = $DESCRIPTOR('WORLD');
;  754     1283  2
;  755     1284  2
;  756     1285  2
;  757     1286  2     ! Initialize descriptor.
;  758     1287  2     !
;  759     1288  2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
;  760     1289  2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
;  761     1290  2
;  762     1291  2
;  763     1292  2     ! Check for SYSTEM keyword.
```

```
;  764          1293  2     !
;  765          1294  2     IF CLI$PRESENT( SYSTEM_DESC )
;  766          1295  2     THEN
;  767          1296  3         BEGIN
;  768          1297  3
;  769          1298  3         ! Note that this is SYSTEM access.  Initialize the the system protection
;  770          1299  3         ! fields, in case noaccess is specified.
;  771          1300  3         !
;  772          1301  3         KEY_DISP = BIT_LOCATION( XAB$V_SYS );
;  773          1302  3         CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
;  774          1303  3         CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
;  775          1304  3
;  776          1305  3         ! Retrieve the keyword value, if any, and parse it.
;  777          1306  3         !
;  778          1307  3         IF CLI$GET_VALUE( SYSTEM_DESC, CLI_DESC )
;  779          1308  3         THEN
;  780          1309  3             PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;  781          1310  3
;  782          1311  2         END;                                                ! SYSTEM parse
;  783          1312  2
;  784          1313  2
;  785          1314  2     ! Check for OWNER keyword.
;  786          1315  2     !
;  787          1316  2     IF CLI$PRESENT( OWNER_DESC )
;  788          1317  2     THEN
;  789          1318  3         BEGIN
;  790          1319  3
;  791          1320  3         ! Note that this is OWNER access.  Initialize the the OWNER protection
;  792          1321  3         ! fields, in case noaccess is specified.
;  793          1322  3         !
;  794          1323  3         KEY_DISP = BIT_LOCATION( XAB$V_OWN );
;  795          1324  3         CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
;  796          1325  3         CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
;  797          1326  3
;  798          1327  3         ! Retrieve the keyword value, if any, and parse it.
;  799          1328  3         !
;  800          1329  3         IF CLI$GET_VALUE( OWNER_DESC, CLI_DESC )
;  801          1330  3         THEN
;  802          1331  3             PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;  803          1332  3
;  804          1333  2         END;                                                ! OWNER parse
;  805          1334  2
;  806          1335  2
;  807          1336  2     ! Check for GROUP keyword.
;  808          1337  2     !
;  809          1338  2     IF CLI$PRESENT( GROUP_DESC )
;  810          1339  2     THEN
;  811          1340  3         BEGIN
;  812          1341  3
;  813          1342  3         ! Note that this is GROUP access.  Initialize the the GROUP protection
;  814          1343  3         ! fields, in case noaccess is specified.
;  815          1344  3         !
;  816          1345  3         KEY_DISP = BIT_LOCATION( XAB$V_GRP );
;  817          1346  3         CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
;  818          1347  3         CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
;  819          1348  3
;  820          1349  3         ! Retrieve the keyword value, if any, and parse it.
```

```
;  821          1350  3          !
;  822          1351  3          IF CLI$GET_VALUE( GROUP_DESC, CLI_DESC )
;  823          1352  3          THEN
;  824          1353  3              PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;  825          1354  3
;  826          1355  2          END;                                                ! GROUP parse
;  827          1356  2
;  828          1357  2
;  829          1358  2      ! Check for WORLD keyword.
;  830          1359  2      !
;  831          1360  2      IF CLI$PRESENT( WORLD_DESC )
;  832          1361  2      THEN
;  833          1362  3          BEGIN
;  834          1363  3
;  835          1364  3          ! Note that this is WORLD access.  Initialize the the WORLD protection
;  836          1365  3          ! fields, in case noaccess is specified.
;  837          1366  3          !
;  838          1367  3          KEY_DISP = BIT_LOCATION( XAB$V_WLD );
;  839          1368  3          CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
;  840          1369  3          CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
;  841          1370  3
;  842          1371  3          ! Retrieve the keyword value, if any, and parse it.
;  843          1372  3          !
;  844          1373  3          IF CLI$GET_VALUE( WORLD_DESC, CLI_DESC )
;  845          1374  3          THEN
;  846          1375  3              PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;  847          1376  3
;  848          1377  2          END;                                                ! WORLD parse
;  849          1378  2
;  850          1379  2      RETURN;                                                 ! Return to the caller.
;  851          1380  1      END;


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                4D  45  54  53  59  53  00148 P.ABH:   .ASCII  \SYSTEM\                    ;
                                                        0014E          .BLKB   2
                                            00000006    00150 P.ABG:   .LONG   6                         ;
                                            00000000'   00154          .ADDRESS P.ABH                    ;
                                    52  45  4E  57  4F  00158 P.ABJ:   .ASCII  \OWNER\                   ;
                                                        0015D          .BLKB   3
                                            00000005    00160 P.ABI:   .LONG   5                         ;
                                            00000000'   00164          .ADDRESS P.ABJ                    ;
                                    50  55  4F  52  47  00168 P.ABL:   .ASCII  \GROUP\                   ;
                                                        0016D          .BLKB   3
                                            00000005    00170 P.ABK:   .LONG   5                         ;
                                            00000000'   00174          .ADDRESS P.ABL                    ;
                                    44  4C  52  4F  57  00178 P.ABN:   .ASCII  \WORLD\                   ;
                                                        0017D          .BLKB   3
                                            00000005    00180 P.ABM:   .LONG   5                         ;
                                            00000000'   00184          .ADDRESS P.ABN                    ;

                                                    SYSTEM_DESC=        P.ABG
                                                    OWNER_DESC=         P.ABI
                                                    GROUP_DESC=         P.ABK
                                                    WORLD_DESC=         P.ABM
```

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

                                        07FC 00000 PROTECTION_PARSE:
                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10                ; 1236
                                5A      0000V   CF  9E 00002            MOVAB   PARSE_PROTECTION_VALUE, R10      ;
                                59      0000G   CF  9E 00007            MOVAB   CLI$GET_VALUE, R9                ;
                                58      0000G   CF  9E 0000C            MOVAB   CLI$PRESENT, R8                  ;
                                57      0000'   CF  9E 00011            MOVAB   SYSTEM_DESC, R7                  ;
                                56      0000'   CF  9E 00016            MOVAB   CURR_PROTECTION_OR, R6           ;
                                5E              08  C2 0001B            SUBL2   #8, SP                           ;
                08              00      6E          00  2C 0001E        MOVC5   #0, (SP), #0, #8, CLI_DESC       ; 1288
                                                    6E      00023       ;                                        ;
                        03      AE              02  90 00024            MOVB    #2, CLI_DESC+3                   ; 1289
                                                57  DD 00028            PUSHL   R7                               ; 1294
                                68              01  FB 0002A            CALLS   #1, CLI$PRESENT                  ;
                                23              50  E9 0002D            BLBC    R0, 1$                           ;
                                                52  D4 00030            CLRL    KEY_DISP                         ; 1301
                66              04      52 FFFFFFFF 8F  F0 00032        INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR ; 1302
        04      A6              04      52          00  F0 0003B        INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND ; 1303
                                            4080    8F  BB 00041        PUSHR   #^M<R7,SP>                       ; 1307
                                69              02  FB 00045            CALLS   #2, CLI$GET_VALUE                ;
                                08              50  E9 00048            BLBC    R0, 1$                           ;
                                                52  DD 0004B            PUSHL   KEY_DISP                         ; 1309
                                        04      AE  9F 0004D            PUSHAB  CLI_DESC                         ;
                                6A              02  FB 00050            CALLS   #2, PARSE_PROTECTION_VALUE       ;
                                        10      A7  9F 00053 1$:        PUSHAB  OWNER_DESC                       ; 1316
                                68              01  FB 00056            CALLS   #1, CLI$PRESENT                  ;
                                25              50  E9 00059            BLBC    R0, 2$                           ;
                                                52  D4 0005C            MOVL    #4, KEY_DISP                     ; 1323
                66              04      52 FFFFFFFF 8F  F0 0005F        INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR ; 1324
        04      A6              04      52          00  F0 00068        INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND ; 1325
                                                5E  DD 0006E            PUSHL   SP                               ; 1329
                                        10      A7  9F 00070            PUSHAB  OWNER_DESC                       ;
                                69              02  FB 00073            CALLS   #2, CLI$GET_VALUE                ;
                                08              50  E9 00076            BLBC    R0, 2$                           ;
                                                52  DD 00079            PUSHL   KEY_DISP                         ; 1331
                                        04      AE  9F 0007B            PUSHAB  CLI_DESC                         ;
                                6A              02  FB 0007E            CALLS   #2, PARSE_PROTECTION_VALUE       ;
                                        20      A7  9F 00081 2$:        PUSHAB  GROUP_DESC                       ; 1338
                                68              01  FB 00084            CALLS   #1, CLI$PRESENT                  ;
                                25              50  E9 00087            BLBC    R0, 3$                           ;
                                                52  D0 0008A            MOVL    #8, KEY_DISP                     ; 1345
                66              04      52 FFFFFFFF 8F  F0 0008D        INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR ; 1346
        04      A6              04      52          00  F0 00096        INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND ; 1347
                                                5E  DD 0009C            PUSHL   SP                               ; 1351
                                        20      A7  9F 0009E            PUSHAB  GROUP_DESC                       ;
                                69              02  FB 000A1            CALLS   #2, CLI$GET_VALUE                ;
                                08              50  E9 000A4            BLBC    R0, 3$                           ;
                                                52  DD 000A7            PUSHL   KEY_DISP                         ; 1353
                                        04      AE  9F 000A9            PUSHAB  CLI_DESC                         ;
                                6A              02  FB 000AC            CALLS   #2, PARSE_PROTECTION_VALUE       ;
                                        30      A7  9F 000AF 3$:        PUSHAB  WORLD_DESC                       ; 1360
                                68              01  FB 000B2            CALLS   #1, CLI$PRESENT                  ;
                                25              50  E9 000B5            BLBC    R0, 4$                           ;
                                                52  D0 000B8            MOVL    #12, KEY_DISP                    ; 1367
```

```
                66              04              52 FFFFFFFF    8F  F0 000BB          INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR        ; 1368
          04    A6              04              52             00  F0 000C4          INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND       ; 1369
                                                               5E  DD 000CA          PUSHL   SP                                          ; 1373
                                                        30     A7  9F 000CC          PUSHAB  WORLD_DESC                                  ;
                                                  69           02  FB 000CF          CALLS   #2, CLI$GET_VALUE                           ;
                                                  08           50  E9 000D2          BLBC    R0, 4$                                      ;
                                                               52  DD 000D5          PUSHL   KEY_DISP                                    ; 1375
                                                        04     AE  9F 000D7          PUSHAB  CLI_DESC                                    ;
                                                  6A           02  FB 000DA          CALLS   #2, PARSE_PROTECTION_VALUE                  ;
                                                               04  000DD 4$:        RET                                                 ; 1380

; Routine Size:  222 bytes,    Routine Base:  $CODE$ + 0888
```

```
;  853          1381  1 ROUTINE PARSE_PROTECTION_VALUE( DESC : REF $BBLOCK,
;  854          1382  1                                 FIELD_LOCATION ) : NOVALUE =
;  855          1383  1
;  856          1384  1 !++
;  857          1385  1 !
;  858          1386  1 ! FUNCTIONAL DESCRIPTION:
;  859          1387  1 !
;  860          1388  1 !       This routine parses the keyword value given by the /PROTECTION
;  861          1389  1 !       qualifier.  (/PROTECTION=(s:rewd))
;  862          1390  1 !
;  863          1391  1 ! FORMAL PARAMETERS:
;  864          1392  1 !
;  865          1393  1 !       DESC            the address of a descriptor which points to the
;  866          1394  1 !                       keyword value.
;  867          1395  1 !       FIELD_LOCATION  the offset of the appropriate protection field
;  868          1396  1 !
;  869          1397  1 ! IMPLICIT INPUTS:
;  870          1398  1 !
;  871          1399  1 !       None
;  872          1400  1 !
;  873          1401  1 ! IMPLICIT OUTPUTS:
;  874          1402  1 !
;  875          1403  1 !       bits will be set in CURR_PROTECTION_OR
;  876          1404  1 !
;  877          1405  1 ! ROUTINE VALUE:
;  878          1406  1 !
;  879          1407  1 !       None
;  880          1408  1 !
;  881          1409  1 ! COMPLETION CODES:
;  882          1410  1 !
;  883          1411  1 !       None
;  884          1412  1 !
;  885          1413  1 ! SIDE EFFECTS:
;  886          1414  1 !
;  887          1415  1 !       None
;  888          1416  1 !
;  889          1417  1 !--
;  890          1418  1
;  891          1419  2 BEGIN
;  892          1420  2
;  893          1421  2 LOCAL
;  894          1422  2     RTN_STATUS,                           ! status returned from external calls
;  895          1423  2     BIT_DISP,                             ! Location of bit to be set in protection  field
;  896          1424  2     CHAR_DESC : $BBLOCK[ DSC$C_S_BLN]     ! A descriptor
;  897          1425  2     ;
;  898          1426  2
;  899          1427  2
;  900          1428  2 ! The descriptor points to only one character at a time.
;  901          1429  2 !
;  902          1430  2 CH$FILL( 0, DSC$C_S_BLN, char_desc);
;  903          1431  2 CHAR_DESC[ DSC$W_LENGTH ] = 1;
;  904          1432  2
;  905          1433  2
;  906          1434  2 ! Process the keyword value one character at a time.
;  907          1435  2 !
;  908          1436  2 INCR INDEX FROM 0 TO .DESC[ DSC$W_LENGTH ]-1 DO
;  909          1437  3     BEGIN
```

```
3
3          CHAR_DESC[ DSC$A_POINTER ] = .DESC[ DSC$A_POINTER ] + .INDEX;
3
3          ! Look up the keyword in the keyword table.
3          !
4          IF NOT (RTN_STATUS = LIB$LOOKUP_KEY( CHAR_DESC, COPY$PROT_VALUE, BIT_DISP) )
3          THEN
4              BEGIN
4
4              ! No character match was found, signal the error and return to caller.
4              !
4              PUT_MESSAGE( MSG$_BADVALUE, 1, .desc );
4              RETURN;
3              END;
3
3          ! Clear the mask bit which corresponds to the protection attribute.
3          !
3          CURR_PROTECTION_OR[prot_mask( .FIELD_LOCATION + .BIT_DISP, 1)] = NO;
3
2          END;                                         ! End of single character value loop.
1  END;                                                 ! End of routine PARSE_PROTECTION_VAL
```

```
                                           003C 00000 PARSE_PROTECTION_VALUE:
                                                              .WORD   Save R2,R3,R4,R5                        ; 1381
                          5E          0C  C2 00002            SUBL2   #12, SP                                 ;
    08          00        6E          00  2C 00005            MOVC5   #0, (SP), #0, #8, CHAR_DESC             ; 1430
                                  04  AE     0000A
                     04   AE          01  B0 0000C            MOVW    #1, CHAR_DESC                           ; 1431
                          54      04  BC  3C 00010            MOVZWL  @DESC, R4                               ; 1436
                          52      04  AC  D0 00014            MOVL    DESC, R2                                ; 1439
                          53          01  CE 00018            MNEGL   #1, INDEX                               ; 1455
                                      40  11 0001B            BRB     3$                                      ;
                     08   AE    04 B243   9E 0001D 1$:        MOVAB   @4(R2)[INDEX], CHAR_DESC+4              ; 1439
                                      5E  DD 00023            PUSHL   SP                                      ; 1443
                               0000G  CF  9F 00025            PUSHAB  COPY$PROT_VALUE                         ;
                                  0C  AE  9F 00029            PUSHAB  CHAR_DESC                               ;
               00000000G  00          03  FB 0002C            CALLS   #3, LIB$LOOKUP_KEY                      ;
                          55          50  D0 00033            MOVL    R0, RTN_STATUS                          ;
                          19          55  E8 00036            BLBS    RTN_STATUS, 2$                          ;
                                  04  AC  DD 00039            PUSHL   DESC                                    ; 1449
                                      01  DD 0003C            PUSHL   #1                                      ;
                          7E    1114  8F  3C 0003E            MOVZWL  #4372, -(SP)                            ;
                  0000G   CF          01  FB 00043            CALLS   #1, COPY$MSG_NUMBER                     ;
                                      50  DD 00048            PUSHL   R0                                      ;
               00000000G  00          03  FB 0004A            CALLS   #3, LIB$STOP                            ;
                                          04 00051            RET                                             ; 1445
            50      08    AC          6E  C1 00052 2$:        ADDL3   BIT_DISP, FIELD_LOCATION, R0            ; 1455
            00    0000'   CF          50  E5 00057            BBCC    R0, CURR_PROTECTION_OR, 3$              ;
            BC            53          54  F2 0005D 3$:        AOBLSS  R4, INDEX, 1$                           ; 1436
                                          04 00061            RET                                             ; 1458

; Routine Size:  98 bytes,    Routine Base:  $CODE$ + 0966
```

```
;  931          1459  1
;  932          1460  1 END
;  933          1461  0 ELUDOM


                                                        .EXTRN  LIB$SIGNAL, LIB$STOP

;                               PSECT SUMMARY
;
;       Name                    Bytes                   Attributes
;
;  $PLIT$                        392  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $GLOBAL$                       28  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                       2504  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]STARLET.L32;1   9776       65         0       581          00:01.0




;                                   COMMAND QUALIFIERS

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

; Size:         2504 code + 420 data bytes
; Run Time:        00:41.2
; Elapsed Time:    01:25.4
; Lines/CPU Min:    2129
; Lexemes/CPU-Min: 21307
; Memory Used:  369 pages
; Compilation Complete
```

COPYMSG
REQ

COPYSPECS
LIS

COPY

COPY
MAP

VMSMAC
REQ

COPY
REQ

COPYCLI
LIS

COPYMAIN
LIS

COPYSEMAN
LIS