D 8

```
LLL                  IIIIIIIII    BBBBBBBBBBBB      RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                  IIIIIIIII    BBBBBBBBBBBB      RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                  IIIIIIIII    BBBBBBBBBBBB      RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                     III       BBB          BBB  RRR         RRR        TTT        LLL
LLL                     III       BBB          BBB  RRR         RRR        TTT        LLL
LLL                     III       BBB          BBB  RRR         RRR        TTT        LLL
LLL                     III       BBB          BBB  RRR         RRR        TTT        LLL
LLL                     III       BBB          BBB  RRR         RRR        TTT        LLL
LLL                     III       BBBBBBBBBBBB      RRRRRRRRRRRR           TTT        LLL
LLL                     III       BBBBBBBBBBBB      RRRRRRRRRRRR           TTT        LLL
LLL                     III       BBBBBBBBBBBB      RRRRRRRRRRRR           TTT        LLL
LLL                     III       BBB          BBB  RRR   RRR              TTT        LLL
LLL                     III       BBB          BBB  RRR   RRR              TTT        LLL
LLL                     III       BBB          BBB  RRR   RRR              TTT        LLL
LLL                     III       BBB          BBB  RRR      RRR           TTT        LLL
LLL                     III       BBB          BBB  RRR      RRR           TTT        LLL
LLL                     III       BBB          BBB  RRR      RRR           TTT        LLL
LLLLLLLLLLLLLLL      IIIIIIIII    BBBBBBBBBBBB      RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL      IIIIIIIII    BBBBBBBBBBBB      RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL      IIIIIIIII    BBBBBBBBBBBB      RRR         RRR        TTT        LLLLLLLLLLLLLLL
```

```
**FILE**ID**LIBRENAME                                   D  7

LL            IIIIII    BBBBBBBB    RRRRRRRR    EEEEEEEEEE  NN      NN    AAAAAA    MM      MM  EEEEEEEEEE
LL            IIIIII    BBBBBBBB    RRRRRRRR    EEEEEEEEEE  NN      NN    AAAAAA    MM      MM  EEEEEEEEEE
LL              II      BB      BB  RR      RR  EE          NN      NN  AA      AA  MMMM  MMMM  EE
LL              II      BB      BB  RR      RR  EE          NN      NN  AA      AA  MMMM  MMMM  EE
LL              II      BB      BB  RR      RR  EE          NNNN    NN  AA      AA  MM  MM  MM  EE
LL              II      BB      BB  RR      RR  EE          NNNN    NN  AA      AA  MM  MM  MM  EE
LL              II      BBBBBBBB    RRRRRRRR    EEEEEEEE    NN  NN  NN  AA      AA  MM      MM  EEEEEEEE
LL              II      BBBBBBBB    RRRRRRRR    EEEEEEEE    NN  NN  NN  AA      AA  MM      MM  EEEEEEEE
LL              II      BB      BB  RR  RR      EE          NN    NNNN  AAAAAAAAAA  MM      MM  EE
LL              II      BB      BB  RR  RR      EE          NN    NNNN  AAAAAAAAAA  MM      MM  EE
LL              II      BB      BB  RR    RR    EE          NN      NN  AA      AA  MM      MM  EE              ....
LL              II      BB      BB  RR    RR    EE          NN      NN  AA      AA  MM      MM  EE              ....
LLLLLLLLLL    IIIIII    BBBBBBBB    RR      RR  EEEEEEEEEE  NN      NN  AA      AA  MM      MM  EEEEEEEEEE      ....
LLLLLLLLLL    IIIIII    BBBBBBBB    RR      RR  EEEEEEEEEE  NN      NN  AA      AA  MM      MM  EEEEEEEEEE      ....


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
```

```
0 MODULE LIB$RENAME_FILE ( %TITLE 'Rename one or more files'
0                 IDENT = '1-006'         ! File: LIBRENAME.B32 Edit: BLS0331
0                 ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1
1 !++
1 ! FACILITY:     General Utility Library
1 !
1 ! ABSTRACT:
1 !
1 !       LIB$RENAME_FILE renames one or more files.
1 !
1 ! ENVIRONMENT:  User mode - AST reentrant
1 !
1 ! AUTHOR: Steven B. Lionel, CREATION DATE: 13-July-1982
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  SBL 13-July-1982
1 ! 1-002 - Add related-filespec argument.  Pass FAB to confirm-routine.
1 !         SBL 1-Oct-1982
1 ! 1-003 - Pass error source code to error-routine.  SBL 11-Nov-1982
1 ! 1-004 - Allow for new argument to LIB$FILE_SCAN.  BLS  6-FEB-1984
1 ! 1-005 - Correct related name string handling.  Add new argument
1 !       for filescan context. BLS  5-MAR-1984
1 ! 1-006 - Parse the null string after calling file_scan to free up
1 !       internal RMS context. BLS  9-JUL-1984
1 !--
1
```

```
;    55          0054  1 %SBTTL 'Declarations'
;    56          0055  1 !
;    57          0056  1 ! PROLOGUE FILE:
;    58          0057  1 !
;    59          0058  1
;    60          0059  1 REQUIRE 'RTLIN:LIBPROLOG';                          ! Switches, PSECTs, macros, etc.
;    61          0130  1
;    62          0131  1 !
;    63          0132  1 ! LINKAGES:
;    64          0133  1 !
;    65          0134  1 !       NONE
;    66          0135  1 !
;    67          0136  1 ! TABLE OF CONTENTS:
;    68          0137  1 !
;    69          0138  1
;    70          0139  1 FORWARD ROUTINE
;    71          0140  1     LIB$RENAME_FILE,                                ! Rename one or more files
;    72          0141  1     DO_RENAME: NOVALUE,                             ! Rename a file
;    73          0142  1     RENAME_ERROR: NOVALUE,                          ! Error routine
;    74          0143  1     COPY_RESULTANT_NAME,                            ! Copy resultant filename
;    75          0144  1     RENAME_HANDLER;                                 ! Condition handler
;    76          0145  1
;    77          0146  1 !
;    78          0147  1 ! MACROS:
;    79          0148  1 !
;    80          0149  1 !       NONE
;    81          0150  1 !
;    82          0151  1 ! EQUATED SYMBOLS:
;    83          0152  1 !
;    84          0153  1 !       NONE
;    85          0154  1 !
;    86          0155  1 ! FIELDS:
;    87          0156  1 !
;    88          0157  1
;    89          0158  1 !+
;    90          0159  1 ! Define bit fields of FLAGS longword.
;    91          0160  1 !-
;    92          0161  1
;    93          0162  1 FIELD
;    94          0163  1     FLAGS_FIELDS =
;    95          0164  1         SET
;    96          0165  1
;    97          0166  1         V_NONEW_VERSION = [0,0,1,0],                ! Don't create new version
;    98          0167  1         V_RESERVED      = [0,1,31,0]                ! Reserved - MBZ
;    99          0168  1
;   100          0169  1         TES;
;   101          0170  1
;   102          0171  1 !
;   103          0172  1 ! OWN STORAGE:
;   104          0173  1 !
;   105          0174  1 !       NONE
;   106          0175  1 !
;   107          0176  1 ! EXTERNALS:
;   108          0177  1 !
;   109          0178  1
;   110          0179  1 EXTERNAL ROUTINE
;   111          0180  1     LIB$ANALYZE_SDESC_R2: LIB$ANALYZE_SDESC_R2$LINKAGE,
;   112         0181  1       LIB$FILE_SCAN,
;   113         0182  1       LIB$GET_VM,
;   114         0183  1       LIB$FREE_VM,
;   115         0184  1       LIB$SCOPY_DXDX,
;   116         0185  1       LIB$SCOPY_R_DX,
;   117         0186  1       LIB$SIG_TO_RET;
;   118         0187  1
;   119         0188  1 EXTERNAL LITERAL
;   120         0189  1       LIB$_ERRROUCAL,
;   121         0190  1       LIB$_INVARG,
;   122         0191  1       LIB$_INVFILSPE;
```

```
1 %SBTTL 'LIB$RENAME_FILE - Rename one or more files'
1 GLOBAL ROUTINE LIB$RENAME_FILE (
1       OLD_FILESPEC: REF BLOCK [, BYTE],               ! Old file specification
1       NEW_FILESPEC: REF BLOCK [, BYTE],               ! New file specification
1       DEFAULT_FILESPEC: REF BLOCK [, BYTE],           ! Default old file specification
1       RELATED_FILESPEC: REF BLOCK [, BYTE],           ! Related old file specification
1       FLAGS: REF BLOCK [4, BYTE] FIELD (FLAGS_FIELDS),    ! Option flags
1       SUCCESS_ROUTINE,                                ! Called on successful rename
1       ERROR_ROUTINE,                                  ! Called on error
1       CONFIRM_ROUTINE,                                ! Called for conformation
1       USER_ARG,                                       ! User argument
1       OLD_RESULTANT_NAME: REF BLOCK [, BYTE],         ! Returned old filename
1       NEW_RESULTANT_NAME: REF BLOCK [, BYTE],         ! Returned new filename
1       FILE_SCAN_CONTEXT                               ! Context for filescan
1       ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !         LIB$RENAME_FILE changes the name(s) of one or more files.  It is
1 !         similar in function to the DCL RENAME command.  The specification
1 !         of the file(s) to be renamed may include wild cards.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !         ret_status.wlc.v = LIB$RENAME_FILE (
1 !                 OLD_FILESPEC.rt.dx, NEW_FILESPEC.rt.dx
1 !                 [, [DEFAULT_FILESPEC.rt.dx]
1 !                 [, [RELATED_FILESPEC.rt.dx]
1 !                 [, [FLAGS.rlu.r]
1 !                 [, [SUCCESS_ROUTINE.szem.r]
1 !                 [, [ERROR_ROUTINE.fzemlc.r]
1 !                 [, [CONFIRM_ROUTINE.fzemlc.r]
1 !                 [, [USER_ARG.rz]
1 !                 [, [OLD_RESULTANT_NAME.wt.dx]
1 !                 [, [NEW_RESULTANT_NAME.wt.dx]
1 !                 [, FILE_SCAN_CONTEXT.rlu.w]]]]]]]]])
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !         OLD_FILESPEC            - The file specification of the file(s)
1 !                                   to be renamed.  Passed by descriptor.
1 !                                   The specification may include wild cards.
1 !
1 !         NEW_FILESPEC            - The file specification for the new file
1 !                                   name(s).  Passed by descriptor.
1 !
1 !         DEFAULT_FILESPEC        - The default file specification of the file(s)
1 !                                   to be renamed.  Passed by descriptor.  This
1 !                                   is an optional parameter; if omitted, the
1 !                                   default is the null string.
1 !
1 !         RELATED_FILESPEC        - The related file specification of the
1 !                                   files to be renamed.  Passed by descriptor.
1 !                                   If omitted, the default is the null string.
1 !                                   "Input file parsing" is used.
1 !
```

```
;  181        0249  1 !         FLAGS                   - A longword of flag bits specifying optional
;  182        0250  1 !                                   behavior.  This is an optional parameter,
;  183        0251  1 !                                   the default is that all flags are clear.
;  184        0252  1 !
;  185        0253  1 !                                     V_NONEW_VERSION - Bit 0
;  186        0254  1 !                                         In the case where NEW_FILESPEC does
;  187        0255  1 !                                         not specify a version number, controls
;  188        0256  1 !                                         whether or not a new version number for
;  189        0257  1 !                                         the output file is to be assigned.
;  190        0258  1 !
;  191        0259  1 !                                         If clear, the default, the output file
;  192        0260  1 !                                         has a version number one higher than
;  193        0261  1 !                                         any previously existing file of the same
;  194        0262  1 !                                         file name and file type.  If set, the
;  195        0263  1 !                                         version number of the input file is used;
;  196        0264  1 !                                         If a file already exists with the same
;  197        0265  1 !                                         file name, type and version number, an
;  198        0266  1 !                                         error is returned.
;  199        0267  1 !
;  200        0268  1 !                                         This flag is equivalent to the
;  201        0269  1 !                                         /NONEW_VERSION qualifier of the
;  202        0270  1 !                                         DCL RENAME command.
;  203        0271  1 !
;  204        0272  1 !         SUCCESS_ROUTINE         - The entry mask of a routine to call for
;  205        0273  1 !                                   each successful rename, passed by reference.
;  206        0274  1 !                                   The calling format of the SUCCESS_ROUTINE is
;  207        0275  1 !                                   as follows:
;  208        0276  1 !                                         CALL SUCCESS_ROUTINE (
;  209        0277  1 !                                             old_filespec.rt.ds,
;  210        0278  1 !                                             new_filespec.rt.ds,
;  211        0279  1 !                                             user_arg.rz)
;  212        0280  1 !
;  213        0281  1 !                                     old_filespec - The RMS resultant file specification
;  214        0282  1 !                                                    of the file being renamed.  If
;  215        0283  1 !                                                    OLD_RESULTANT_NAME was specified,
;  216        0284  1 !                                                    it is used to pass the string
;  217        0285  1 !                                                    to SUCCESS_ROUTINE.  Otherwise,
;  218        0286  1 !                                                    a class S, type T string is passed.
;  219        0287  1 !
;  220        0288  1 !                                     new_filespec - The RMS resultant file specification
;  221        0289  1 !                                                    of the newly renamed file.  If
;  222        0290  1 !                                                    NEW_RESULTANT_NAME was specified,
;  223        0291  1 !                                                    it is used to pass the string
;  224        0292  1 !                                                    to SUCCESS_ROUTINE.  Otherwise,
;  225        0293  1 !                                                    a class S, type T string is passed.
;  226        0294  1 !
;  227        0295  1 !                                     user_arg     - The value of user_arg passed
;  228        0296  1 !                                                    to LIB$RENAME_FILE is passed
;  229        0297  1 !                                                    to SUCCESS_ROUTINE using the
;  230        0298  1 !                                                    same mechanism as was used to
;  231        0299  1 !                                                    pass it to LIB$RENAME_FILE.
;  232        0300  1 !
;  233        0301  1 !         ERROR_ROUTINE           - The entry mask of a routine to call when
;  234        0302  1 !                                   a file error is detected, passed by reference.
;  235        0303  1 !                                   The function value returned by the routine
;  236        0304  1 !                                   determines whether or not more files will
;  237        0305  1 !                                   be processed.
;  238      0306  1 !
;  239      0307  1 !                      The calling format of the ERROR_ROUTINE is
;  240      0308  1 !                      as follows:
;  241      0309  1 !                            ret_status.wlc.v = ERROR_ROUTINE (
;  242      0310  1 !                                old_filespec.rt.ds,
;  243      0311  1 !                                new_filespec.rt.ds,
;  244      0312  1 !                                RMS_sts.rlc.r
;  245      0313  1 !                                RMS_stv.rlc.r,
;  246      0314  1 !                                error_source.rl.r,
;  247      0315  1 !                                user_arg.rz)
;  248      0316  1 !
;  249      0317  1 !                        old_filespec - The RMS resultant file specification
;  250      0318  1 !                                       of the file being renamed
;  251      0319  1 !                                       when the error occured.  If
;  252      0320  1 !                                       OLD_RESULTANT_NAME was specified,
;  253      0321  1 !                                       it is used to pass the string
;  254      0322  1 !                                       to ERROR_ROUTINE.  Otherwise,
;  255      0323  1 !                                       a class S, type T string is passed.
;  256      0324  1 !
;  257      0325  1 !                        new_filespec - The RMS resultant file specification
;  258      0326  1 !                                       of the new file name being used
;  259      0327  1 !                                       when the error occured.  If
;  260      0328  1 !                                       NEW_RESULTANT_NAME was specified,
;  261      0329  1 !                                       it is used to pass the string
;  262      0330  1 !                                       to ERROR_ROUTINE.  Otherwise,
;  263      0331  1 !                                       a class S, type T string is passed.
;  264      0332  1 !
;  265      0333  1 !                        RMS_sts      - The primary condition code
;  266      0334  1 !                                       which describes the error
;  267      0335  1 !                                       that occurred.
;  268      0336  1 !
;  269      0337  1 !                        RMS_stv      - The secondary condition code
;  270      0338  1 !                                       which describes the error
;  271      0339  1 !                                       that occurred.
;  272      0340  1 !
;  273      0341  1 !                        error_source - An integer code that indicates
;  274      0342  1 !                                       at what point the error was
;  275      0343  1 !                                       found.  The values are:
;  276      0344  1 !                                         0 = Error searching for
;  277      0345  1 !                                             old-file
;  278      0346  1 !                                         1 = Error parsing new
;  279      0347  1 !                                             filespec
;  280      0348  1 !                                         2 = Error renaming file
;  281      0349  1 !
;  282      0350  1 !                        user_arg     - The value of user_arg passed
;  283      0351  1 !                                       to LIB$RENAME_FILE is passed
;  284      0352  1 !                                       to ERROR_ROUTINE using the
;  285      0353  1 !                                       same mechanism as was used to
;  286      0354  1 !                                       pass it to LIB$RENAME_FILE.
;  287      0355  1 !
;  288      0356  1 !                      If ERROR_ROUTINE returns a success status,
;  289      0357  1 !                      then processing of files will continue.  If
;  290      0358  1 !                      a failure status is returned, then processing
;  291      0359  1 !                      will cease immediately and LIB$RENAME_FILE
;  292      0360  1 !                      will return with an error status.
;  293      0361  1 !
;  294      0362  1 !                      If ERROR_ROUTINE is not specified,
1 !                                             LIB$RENAME_FILE will return to its caller with
1 !                                             the most severe of the error statuses encountered
1 !                                             while renaming the files.  Otherwise, if
1 !                                             ERROR_ROUTINE is called for an error, the
1 !                                             success status LIB$_ERRROUCAL is returned.
1 !                                             Note that ERROR_ROUTINE is not called for
1 !                                             errors related to string copying.
1 !
1 !                 CONFIRM_ROUTINE           - The entry mask of a routine to call before
1 !                                             each file is renamed, passed by reference.
1 !                                             The function value returned by the routine
1 !                                             determines whether or not the file will be
1 !                                             be renamed.
1 !
1 !                                             The calling format of CONFIRM_ROUTINE is
1 !                                             as follows:
1 !                                                   ret_status.wlc.v = CONFIRM_ROUTINE (
1 !                                                       old_filespec.rt.ds,
1 !                                                       new_filespec.rt.ds,
1 !                                                       old_FAB.rr.r,
1 !                                                       user_arg.rz)
1 !
1 !                                               old_filespec - The RMS resultant file specification
1 !                                                              of the file about to be
1 !                                                              renamed.  If OLD_RESULTANT_NAME
1 !                                                              was specified, it is used to
1 !                                                              pass the string to CONFIRM_ROUTINE.
1 !                                                              Otherwise, a class S, type T
1 !                                                              string is passed.
1 !
1 !                                               new_filespec - The RMS resultant file specification
1 !                                                              which the file will be given.
1 !                                                              If NEW_RESULTANT_NAME
1 !                                                              was specified, it is used to
1 !                                                              pass the string to CONFIRM_ROUTINE.
1 !                                                              Otherwise, a class S, type T
1 !                                                              string is passed.
1 !
1 !                                               old_FAB      - The address of the RMS FAB
1 !                                                              that describes the file
1 !                                                              being renamed.  You may
1 !                                                              perform an RMS $OPEN on the
1 !                                                              FAB to obtain file attributes
1 !                                                              you need to determine whether
1 !                                                              the file should be renamed,
1 !                                                              but you must close the file
1 !                                                              with $CLOSE before returning
1 !                                                              to LIB$RENAME_FILE
1 !
1 !                                               user_arg     - The value of user_arg passed
1 !                                                              to LIB$RENAME_FILE is passed
1 !                                                              to CONFIRM_ROUTINE using the
1 !                                                              same mechanism as was used to
1 !                                                              pass it to LIB$RENAME_FILE.
1 !
1 !                                             If CONFIRM_ROUTINE returns success, the file
1 !                                             is then renamed, otherwise that file is not
```

```
;  352        0420   1 !                             renamed.
;  353        0421   1 !
;  354        0422   1 !        USER_ARG             - A value passed to SUCCESS_ROUTINE,
;  355        0423   1 !                               ERROR_ROUTINE and CONFIRM_ROUTINE each
;  356        0424   1 !                               time they are called.  Whatever mechanism
;  357        0425   1 !                               is used to pass USER_ARG to LIB$RENAME_FILE is
;  358        0426   1 !                               used to pass it to the action routine.  This
;  359        0427   1 !                               is an optional parameter, if omitted, zero
;  360        0428   1 !                               is passed by immediate value.
;  361        0429   1 !
;  362        0430   1 !        OLD_RESULTANT_NAME   - A string into which is written the old RMS resultant
;  363        0431   1 !                               file specification of the last file processed
;  364        0432   1 !                               by LIB$RENAME_FILE.  Passed by descriptor.
;  365        0433   1 !                               This is an optional parameter. If present,
;  366        0434   1 !                               it is used to store the filespec passed to
;  367        0435   1 !                               the action routines instead of a default
;  368        0436   1 !                               class S, type T string.
;  369        0437   1 !
;  370        0438   1 !        NEW_RESULTANT_NAME   - A string into which is written the new RMS resultant
;  371        0439   1 !                               file specification of the last file processed
;  372        0440   1 !                               by LIB$RENAME_FILE.  Passed by descriptor.
;  373        0441   1 !                               This is an optional parameter. If present,
;  374        0442   1 !                               it is used to store the filespec passed to
;  375        0443   1 !                               the action routines instead of a default
;  376        0444   1 !                               class S, type T string.
;  377        0445   1 !
;  378        0446   1 !        FILE_SCAN_CONTEXT    - The address of a longword, which is
;  379        0447   1 !                               initialized to 0 before calling LIB$RENAME_FILE.
;  380        0448   1 !                               This context is used by LIB$FILE_SCAN to retain
;  381        0449   1 !                               multiple input file related file context, and
;  382        0450   1 !                               need only be specified if dealing with multiple
;  383        0451   1 !                               input files, as the DCL RENAME command does.
;  384        0452   1 !                               The context allocated by LIB$FILE_SCAN while
;  385        0453   1 !                               processing the LIB$RENAME_FILE requests may
;  386        0454   1 !                               be deallocated by calling LIB$FILE_SCAN_END
;  387        0455   1 !                               after all calls to LIB$RENAME_FILE have been
;  388        0456   1 !                               completed.
;  389        0457   1 !
;  390        0458   1 ! IMPLICIT INPUTS:
;  391        0459   1 !
;  392        0460   1 !        NONE
;  393        0461   1 !
;  394        0462   1 ! IMPLICIT OUTPUTS:
;  395        0463   1 !
;  396        0464   1 !        NONE
;  397        0465   1 !
;  398        0466   1 ! COMPLETION STATUS:
;  399        0467   1 !
;  400        0468   1 !        SS$_NORMAL       Normal successful completion
;  401        0469   1 !        LIB$_ERRROUCAL   Success - error routine called.  A file error occurred
;  402        0470   1 !                         but ERROR_ROUTINE was called to handle the condition.
;  403        0471   1 !        LIB$_INVSTRDES   Invalid string descriptor
;  404        0472   1 !        LIB$_INVARG      Invalid argument.  FLAGS has undefined bits set.
;  405        0473   1 !        LIB$_INVFILSPE   Invalid file specification.  OLD_FILESPEC, NEW_FILESPEC
;  406        0474   1 !                         or DEFAULT_OLD_FILESPEC contains more than 255 characters.
;  407        0475   1 !        LIB$_WRONUMARG   Wrong number of arguments.
;  408        0476   1 !        LIB$_xxx         Any error status from LIB$SCOPY_xxx
;  409          0477  1 !       RMS$_xxx        Any error status from RMS.  If ERROR_ROUTINE was not
;  410          0478  1 !                       specified, this is the most severe of the RMS errors
;  411          0479  1 !                       which occurred while renaming the files.
;  412          0480  1 !
;  413          0481  1 ! SIDE EFFECTS:
;  414          0482  1 !
;  415          0483  1 !       The files specified are renamed.
;  416          0484  1 !
;  417          0485  1 !--
;  418          0486  1
;  419          0487  2     BEGIN
;  420          0488  2
;  421          0489  2     !+
;  422          0490  2     ! Because $RENAME does not perform wildcarding, we need to keep two
;  423          0491  2     ! separate FABs (and associated NAMs) for both the old filespec and
;  424          0492  2     ! the new filespec.  OLDFAB (and OLDNAM) are passed to LIB$FILE_SCAN
;  425          0493  2     ! as the basis for the wildcard search.  NEWFAB will be used in
;  426          0494  2     ! DO_RENAME to do two $PARSEs on the new filespec, using OLDFAB as
;  427          0495  2     ! input.  Next the resultant filename from OLDFAB will be copied to
;  428          0496  2     ! OLDFAB_R and the expanded filename from NEWFAB to NEWFAB_R.  Then
;  429          0497  2     ! the $RENAME will be done with OLDFAB_R and NEWFAB_R.
;  430          0498  2     !-
;  431          0499  2
;  432          0500  2     LOCAL
;  433          0501  2         OLDFAB: $FAB_DECL,                          ! Old file's FAB
;  434          0502  2         OLDFAB_R: $FAB_DECL,                        ! Old file's FAB for $RENAME
;  435          0503  2         NEWFAB: $FAB_DECL,                          ! New file's FAB
;  436          0504  2         NEWFAB_R: $FAB_DECL,                        ! New file's FAB for $RENAME
;  437          0505  2         OLDNAM: $NAM_DECL,                          ! Old file's NAM
;  438          0506  2         OLDNAM_R: $NAM_DECL,                        ! Old file's NAM for $RRNAME
;  439          0507  2         NEWNAM: $NAM_DECL,                          ! New file's NAM
;  440          0508  2         NEWNAM_R: $NAM_DECL,                        ! New file's NAM for $RENAME
;  441          0509  2         RLF_NAM: $NAM_DECL,                         ! Related filespec NAM
;  442          0510  2         OLD_ESN: VECTOR [NAM$C_MAXRSS, BYTE],       ! Old Expanded name
;  443          0511  2         OLD_RSN: VECTOR [NAM$C_MAXRSS, BYTE],       ! Old Resultant name
;  444          0512  2         NEW_RSN: VECTOR [NAM$C_MAXRSS, BYTE],       ! New Resultant name
;  445          0513  2         RLF_DESC : $BBLOCK[DSC$C_S_BLN],            ! Temp descriptor for RLF
;  446          0514  2         WORST_ERROR,                                ! Worst error so far
;  447          0515  2         INTERCEPT_FLAG: VOLATILE,                   ! 1 if signals are to be intercepted
;  448          0516  2         RET_STATUS;                                 ! Return status
;  449          0517  2
;  450          0518  2     BUILTIN
;  451          0519  2         ACTUALCOUNT,
;  452          0520  2         NULLPARAMETER;
;  453          0521  2
;  454          0522  2     !+
;  455          0523  2     ! Establish RENAME_HANDLER as a handler which will convert signals
;  456          0524  2     ! to return statuses if INTERCEPT_FLAG is set.  Note that BLISS
;  457          0525  2     ! automatically zeroes INTERCEPT_FLAG as a VOLATILE enable argument.
;  458          0526  2     !-
;  459          0527  2
;  460          0528  2     ENABLE
;  461          0529  2         RENAME_HANDLER (INTERCEPT_FLAG);
;  462          0530  2
;  463          0531  2     !+
;  464          0532  2     ! Validate the argument count.
;  465          0533  2     !-
;   466          0534  2
;   467          0535  2      $LIB$VALIDATE_ARGCOUNT (2,12);
;   468          0536  2
;   469          0537  2      !+
;   470          0538  2      ! Initialize the FAB and NAM blocks.  See DO_RENAME for a description
;   471          0539  2      ! of NEWFAB's DNA field.
;   472          0540  2      !-
;   473          0541  2
;   474          0542  2      $FAB_INIT (FAB=OLDFAB, NAM=OLDNAM);
;   475          0543  2      $FAB_INIT (FAB=OLDFAB_R, NAM=OLDNAM_R, FNA=OLD_RSN);
;   476        P 0544  2      $FAB_INIT (FAB=NEWFAB, NAM=NEWNAM,
;   477          0545  2          DNA = UPLIT BYTE (';*'));
;   478          0546  2      $FAB_INIT (FAB=NEWFAB_R, FNA=NEW_RSN, NAM=NEWNAM, FOP=(OFP));
;   479        P 0547  2      $NAM_INIT (NAM=OLDNAM, ESA=OLD_ESN, ESS=NAM$C_MAXRSS,
;   480          0548  2          RSA=OLD_RSN, RSS=NAM$C_MAXRSS, RLF=RLF_NAM);
;   481        P 0549  2      $NAM_INIT (NAM=OLDNAM_R, ESA=OLD_RSN, ESS=NAM$C_MAXRSS,
;   482          0550  2          RSA=OLD_RSN, RSS=NAM$C_MAXRSS);
;   483        P 0551  2      $NAM_INIT (NAM=NEWNAM, ESA=NEW_RSN, ESS=NAM$C_MAXRSS,
;   484          0552  2          RSA=NEW_RSN, RSS=NAM$C_MAXRSS, RLF=OLDNAM);
;   485        P 0553  2      $NAM_INIT (NAM=NEWNAM_R, ESA=NEW_RSN, ESS=NAM$C_MAXRSS,
;   486          0554  2          RSA=NEW_RSN, RSS=NAM$C_MAXRSS, RLF=OLDNAM);
;   487          0555  2      $NAM_INIT (NAM=RLF_NAM);
;   488          0556  2
;   489          0557  2      !+
;   490          0558  2      ! Initialize WORST_ERROR to zero.  It will be modified if errors occur.
;   491          0559  2      !-
;   492          0560  2
;   493          0561  2      WORST_ERROR = 0;
;   494          0562  2
;   495          0563  2      !+
;   496          0564  2      ! Set up OLDFAB for the old file specification.  Return LIB$_INVFILSPE
;   497          0565  2      ! if the string is longer than 255 characters.
;   498          0566  2      !-
;   499          0567  2
;   500          0568  3          BEGIN
;   501          0569  3
;   502          0570  3          LOCAL
;   503          0571  3              OLD_FILESPEC_LENGTH: WORD,  ! Length of OLD_FILESPEC string
;   504          0572  3              STATUS;                     ! Status from LIB$ANALYZE_SDESC.
;   505          0573  3
;   506          0574  3          STATUS = LIB$ANALYZE_SDESC_R2 (OLD_FILESPEC [0,0,0,0];
;   507          0575  3              OLD_FILESPEC_LENGTH, OLDFAB [FAB$L_FNA]);
;   508          0576  3
;   509          0577  3          IF NOT .STATUS
;   510          0578  3          THEN
;   511          0579  3              RETURN .STATUS;
;   512          0580  3
;   513          0581  3          IF .OLD_FILESPEC_LENGTH GTRU 255
;   514          0582  3          THEN
;   515          0583  3              RETURN LIB$_INVFILSPE;
;   516          0584  3
;   517          0585  3          OLDFAB [FAB$B_FNS] = .OLD_FILESPEC_LENGTH;
;   518          0586  3          END;
;   519          0587  2
;   520          0588  2      !+
;   521          0589  2      ! Set up NEWFAB to refer to NEW_FILENAME.  Return LIB$_INVFILSPE
;   522          0590  2      ! if the string is longer than 255 characters.
```

```
!-

    BEGIN

    LOCAL
        NEW_FILESPEC_LENGTH,
        STATUS;                         ! Status from LIB$ANALYZE_SDESC.

    STATUS = LIB$ANALYZE_SDESC_R2 (NEW_FILESPEC [0,0,0,0];
        NEW_FILESPEC_LENGTH, NEWFAB [FAB$L_FNA]);

    IF NOT .STATUS
    THEN
        RETURN .STATUS;

    IF .NEW_FILESPEC_LENGTH GTRU 255
    THEN
        RETURN LIB$_INVFILSPE;

    NEWFAB [FAB$B_FNS] = .NEW_FILESPEC_LENGTH;
    END;

!+
! If DEFAULT_OLD_FILESPEC is present, set up the default name string
! in OLDFAB to refer to it.
!-

IF NOT NULLPARAMETER (3)
THEN
    BEGIN

    LOCAL
        DEFAULT_FILESPEC_LENGTH: WORD,      ! Length of DEFAULT_FILESPEC string
        STATUS;                         ! Status from LIB$ANALYZE_SDESC.

    STATUS = LIB$ANALYZE_SDESC_R2 (DEFAULT_FILESPEC [0,0,0,0];
        DEFAULT_FILESPEC_LENGTH, OLDFAB [FAB$L_DNA]);

    IF NOT .STATUS
    THEN
        RETURN .STATUS;

    IF .DEFAULT_FILESPEC_LENGTH GTRU 255
    THEN
        RETURN LIB$_INVFILSPE;

    OLDFAB [FAB$B_DNS] = .DEFAULT_FILESPEC_LENGTH;
    END;

!+
! If RELATED_FILESPEC is present, set up the related file specification
! in RLF_NAM to refer to it.
!-

RLF_DESC[DSC$W_LENGTH] = 0;
RLF_DESC[DSC$B_DTYPE] = DSC$K_DTYPE_T;
RLF_DESC[DSC$B_CLASS] = DSC$K_CLASS_D;
```

```
2       RLF_DESC[DSC$A_POINTER] = 0;
2       IF NOT NULLPARAMETER (4)
2           AND NULLPARAMETER (12)
2       THEN
3           BEGIN
3
3           LOCAL
3               RELATED_FILESPEC_LENGTH: WORD, ! Length of RELATED_FILESPEC string
3               RELATED_FILESPEC_ADDR,         ! and it's address
3               STATUS;                        ! Status from LIB$ANALYZE_SDESC.
3
3           STATUS = LIB$ANALYZE_SDESC_R2 (RELATED_FILESPEC [0,0,0,0];
3                   RELATED_FILESPEC_LENGTH, RELATED_FILESPEC_ADDR);
3           IF NOT .STATUS
3           THEN
3               RETURN .STATUS;
3
3           IF .RELATED_FILESPEC_LENGTH GTRU 255
3           THEN
3               RETURN LIB$_INVFILSPE;
3
3           ! Must use LIB$GET_VM since LIB$FILE_SCAN may deallocate with FREE_VM
3           STATUS = LIB$GET_VM(RELATED_FILESPEC_LENGTH,RLF_DESC[DSC$A_POINTER]);
3           RLF_DESC[DSC$W_LENGTH] = .RELATED_FILESPEC_LENGTH;
3           CH$MOVE(.RLF_DESC[DSC$W_LENGTH],.RELATED_FILESPEC_ADDR,
3                       .RLF_DESC[DSC$A_POINTER]);
3           RLF_NAM[NAM$B_RSL] = .RLF_DESC[DSC$W_LENGTH];
3           RLF_NAM[NAM$L_RSA] = .RLF_DESC[DSC$A_POINTER];
2           END;
2
2       !+
2       ! Verify that reserved bits aren't set in FLAGS, if present.
2       !-
2
2       IF NOT NULLPARAMETER (5)
2       THEN
2           IF .FLAGS [V_RESERVED] NEQ 0
2           THEN
2               RETURN LIB$_INVARG;
2
2       !
2       ! If passing context to lib$file_scan, then ensure that we
2       ! don't confuse things with a dummy related nam block
2       !
2       IF NOT NULLPARAMETER(12)
2           THEN OLDNAM[NAM$L_RLF] = 0;
2       !+
2       ! Call LIB$FILE_SCAN, which will call DO_RENAME for each file that is
2       ! to be renamed.  Pass as extra parameters the other FABs.
2       ! These will be passed to DO_RENAME.
2       !-
2
2       LIB$FILE_SCAN (
2           OLDFAB,                        ! Input FAB
2           DO_RENAME,                     ! Success routine
2           RENAME_ERROR,                  ! Error routine
2           (IF ACTUALCOUNT () GEQU 12 THEN .FILE_SCAN_CONTEXT ELSE 0), ! Context
;  637          0705  2           NEWFAB,                              ! FAB for $PARSE of new filename
;  638          0706  2           OLDFAB_R,                            ! Source FAB for $RENAME
;  639          0707  2           NEWFAB_R,                            ! Dest FAB for $RENAME
;  640          0708  2           (IF NOT NULLPARAMETER (5) THEN .FLAGS [0,0,32,0] ELSE 0),! Flags
;  641          0709  2           (IF ACTUALCOUNT () GEQU 6 THEN .SUCCESS_ROUTINE ELSE 0),! Success routine
;  642          0710  2           (IF ACTUALCOUNT () GEQU 7 THEN .ERROR_ROUTINE ELSE 0),  ! Error routine
;  643          0711  2           (IF ACTUALCOUNT () GEQU 8 THEN .CONFIRM_ROUTINE ELSE 0),! Confirm routine
;  644          0712  2           (IF ACTUALCOUNT () GEQU 9 THEN .USER_ARG ELSE 0),       ! User argument
;  645          0713  2           (IF ACTUALCOUNT () GEQU 10 THEN .OLD_RESULTANT_NAME ELSE 0), ! Old resultant name
;  646          0714  2           (IF ACTUALCOUNT () GEQU 11 THEN .NEW_RESULTANT_NAME ELSE 0), ! New resultant name
;  647          0715  2           WORST_ERROR,                         ! Worst error so far
;  648          0716  2           INTERCEPT_FLAG);                     ! Signal intercept flag
;  649          0717  2
;  650          0718  2       LIB$FREE_VM(RLF_DESC[DSC$W_LENGTH],
;  651          0719  2                       RLF_DESC[DSC$A_POINTER]); !Deallocate temp descriptor
;  652          0720  2       !+
;  653          0721  2       ! Parse the null string to deallocate any internal
;  654          0722  2       ! RMS context.
;  655          0723  2       !-
;  656          0724  2
;  657          0725  2       OLDNAM[NAM$V_SVCTX] = 0;
;  658          0726  2       OLDNAM[NAM$V_SYNCHK] = 1;
;  659          0727  2       OLDNAM[NAM$B_ESL] = 0;
;  660          0728  2       OLDNAM[NAM$B_RSL] = 0;
;  661          0729  2       OLDNAM[NAM$B_ESS] = 0;
;  662          0730  2       OLDNAM[NAM$B_RSS] = 0;
;  663          0731  2       OLDNAM[NAM$L_RLF] = 0;
;  664          0732  2       OLDFAB[FAB$B_FNS] = 0;
;  665          0733  2       OLDFAB[FAB$B_DNS] = 0;
;  666          0734  2       $PARSE(FAB=OLDFAB);
;  667          0735  2
;  668          0736  2       !+
;  669          0737  2       ! Return WORST_ERROR or SS$_NORMAL, as appropriate.
;  670          0738  2       !-
;  671          0739  2
;  672          0740  2       IF .WORST_ERROR NEQ 0
;  673          0741  2       THEN
;  674          0742  2           RET_STATUS = .WORST_ERROR
;  675          0743  2       ELSE
;  676          0744  2           RET_STATUS = SS$_NORMAL;
;  677          0745  2
;  678          0746  2       RETURN .RET_STATUS;
;  679          0747  2
;  680          0748  1       END;                                          ! End of routine LIB$RENAME_FILE


                                                                      .TITLE  LIB$RENAME_FILE Rename one or more files
                                                                      .IDENT  \1-006\

                                                                      .PSECT  _LIB$CODE,NOWRT,  SHR,  PIC,2

                                             2A  3B  00000 P.AAA:     .ASCII  \;*\                                    ;

                                                                      .EXTRN  LIB$ANALYZE_SDESC_R2
                                                                      .EXTRN  LIB$FILE_SCAN, LIB$GET_VM
                                                                      .EXTRN  LIB$FREE_VM, LIB$SCOPY_DXDX
                                                                      .EXTRN  LIB$SCOPY_R_DX, LIB$SIG_TO_RET
                                                                      .EXTRN  LIB$_ERRROUCAL, LIB$_INVARG
                                                                      .EXTRN  LIB$_INVFILSPE, LIB$_WRONUMARG
                                                                      .EXTRN  SYS$PARSE

                                                    007C 00000        .ENTRY  LIB$RENAME_FILE, Save R2,R3,R4,R5,R6       ; 0193
                                     56  00000000G00  9E 00002        MOVAB   LIB$ANALYZE_SDESC_R2, R6                   ;
                                     5E      F9CC CE  9E 00009        MOVAB   -1588(SP), SP                              ;
                                               08 AE  D4 0000E        CLRL    INTERCEPT_FLAG                             ; 0487
                                     6D      031D CF  DE 00011        MOVAL   28$, (FP)                                  ;
                        50           6C           02  83 00016        SUBB3   #2, (AP), DIFF                             ; 0535
                                     0A           50  91 0001A        CMPB    DIFF, #10                                  ;
                                                  08  1B 0001D        BLEQU   1$                                         ;
                                     50  00000000G8F  D0 0001F        MOVL    #LIB$_WRONUMARG, R0                        ;
                                                      04 00026        RET                                                ;
    0050   8F           00           6E           00  2C 00027 1$:    MOVC5   #0, (SP), #0, #80, $RMS_PTR                ; 0542
                                               B0 AD     0002E                                                           ;
                                 B0  AD      5003 8F  B0 00030        MOVW    #20483, $RMS_PTR                           ;
                                 C6  AD           02  90 00036        MOVB    #2, $RMS_PTR+22                            ;
                                 CF  AD           02  90 0003A        MOVB    #2, $RMS_PTR+31                            ;
                                 D8  AD      FE60 CD  9E 0003E        MOVAB   OLDNAM, $RMS_PTR+40                        ;
    0050   8F           00           6E           00  2C 00044        MOVC5   #0, (SP), #0, #80, $RMS_PTR                ; 0543
                                             FF60 CD     0004B                                                           ;
                               FF60  CD      5003 8F  B0 0004E        MOVW    #20483, $RMS_PTR                           ;
                               FF76  CD           02  90 00055        MOVB    #2, $RMS_PTR+22                            ;
                               FF7F  CD           02  90 0005A        MOVB    #2, $RMS_PTR+31                            ;
                                 88  AD      FE00 CD  9E 0005F        MOVAB   OLDNAM_R, $RMS_PTR+40                      ;
                                 8C  AD      0114 CE  9E 00065        MOVAB   OLD_RSN, $RMS_PTR+44                       ;
    0050   8F           00           6E           00  2C 0006B        MOVC5   #0, (SP), #0, #80, $RMS_PTR                ; 0545
                                             FF10 CD     00072                                                           ;
                               FF10  CD      5003 8F  B0 00075        MOVW    #20483, $RMS_PTR                           ;
                               FF26  CD           02  90 0007C        MOVB    #2, $RMS_PTR+22                            ;
                               FF2F  CD           02  90 00081        MOVB    #2, $RMS_PTR+31                            ;
                               FF38  CD      FDA0 CD  9E 00086        MOVAB   NEWNAM, $RMS_PTR+40                        ;
                               FF40  CD      FF6D CF  9E 0008D        MOVAB   P.AAA, $RMS_PTR+48                         ;
    0050   8F           00           6E           00  2C 00094        MOVC5   #0, (SP), #0, #80, $RMS_PTR                ; 0546
                                             FEC0 CD     0009B                                                           ;
                               FEC0  CD      5003 8F  B0 0009E        MOVW    #20483, $RMS_PTR                           ;
                               FEC4  CD  20000000 8F  D0 000A5        MOVL    #536870912, $RMS_PTR+4                     ;
                               FED6  CD           02  90 000AE        MOVB    #2, $RMS_PTR+22                            ;
                               FEDF  CD           02  90 000B3        MOVB    #2, $RMS_PTR+31                            ;
                               FEE8  CD      FDA0 CD  9E 000B8        MOVAB   NEWNAM, $RMS_PTR+40                        ;
                               FEEC  CD        14 AE  9E 000BF        MOVAB   NEW_RSN, $RMS_PTR+44                       ;
    0060   8F           00           6E           00  2C 000C5        MOVC5   #0, (SP), #0, #96, $RMS_PTR                ; 0548
                                             FE60 CD     000CC                                                           ;
                               FE60  CD      6002 8F  B0 000CF        MOVW    #24578, $RMS_PTR                           ;
                               FE62  CD           01  8E 000D6        MNEGB   #1, $RMS_PTR+2                             ;
                               FE64  CD      0114 CE  9E 000DB        MOVAB   OLD_RSN, $RMS_PTR+4                        ;
                               FE6A  CD           01  8E 000E2        MNEGB   #1, $RMS_PTR+10                            ;
                               FE6C  CD      0214 CE  9E 000E7        MOVAB   OLD_ESN, $RMS_PTR+12                       ;
                               FE70  CD      0314 CE  9E 000EE        MOVAB   RLF_NAM, $RMS_PTR+16                       ;
    0060   8F           00           6E           00  2C 000F5        MOVC5   #0, (SP), #0, #96, $RMS_PTR                ; 0550
                                             FE00 CD     000FC                                                           ;
                               FE00  CD      6002 8F  B0 000FF        MOVW    #24578, $RMS_PTR                           ;
                               FE02  CD           01  8E 00106        MNEGB   #1, $RMS_PTR+2                             ;
                               FE04  CD      0114 CE  9E 0010B        MOVAB   OLD_RSN, $RMS_PTR+4                        ;
                               FE0A  CD           01  8E 00112        MNEGB   #1, $RMS_PTR+10                            ;
                               FE0C  CD      0114 CE  9E 00117        MOVAB   OLD_RSN, $RMS_PTR+12                       ;
    0060  8F        00            6E             00 2C 0011E       MOVC5   #0, (SP), #0, #96, $RMS_PTR                  ; 0552
                                           FDA0  CD    00125
                            FDA0  CD       6002  8F B0 00128       MOVW    #24578, $RMS_PTR
                            FDA2  CD             01 8E 0012F       MNEGB   #1, $RMS_PTR+2
                            FDA4  CD         14  AE 9E 00134       MOVAB   NEW_RSN, $RMS_PTR+4
                            FDAA  CD             01 8E 0013A       MNEGB   #1, $RMS_PTR+10
                            FDAC  CD         14  AE 9E 0013F       MOVAB   NEW_RSN, $RMS_PTR+12
                            FDB0  CD       FE60  CD 9E 00145       MOVAB   OLDNAM, $RMS_PTR+16
    0060  8F        00            6E             00 2C 0014C       MOVC5   #0, (SP), #0, #96, $RMS_PTR                  ; 0554
                                           FD40  CD    00153
                            FD40  CD       6002  8F B0 00156       MOVW    #24578, $RMS_PTR
                            FD42  CD             01 8E 0015D       MNEGB   #1, $RMS_PTR+2
                            FD44  CD         14  AE 9E 00162       MOVAB   NEW_RSN, $RMS_PTR+4
                            FD4A  CD             01 8E 00168       MNEGB   #1, $RMS_PTR+10
                            FD4C  CD         14  AE 9E 0016D       MOVAB   NEW_RSN, $RMS_PTR+12
                            FD50  CD       FE60  CD 9E 00173       MOVAB   OLDNAM, $RMS_PTR+16
    0060  8F        00            6E             00 2C 0017A       MOVC5   #0, (SP), #0, #96, $RMS_PTR                  ; 0555
                                           0314  CE    00181
                            0314  CE       6002  8F B0 00184       MOVW    #24578, $RMS_PTR
                                             04  AE D4 0018B       CLRL    WORST_ERROR                                  ; 0561
                                  50         04  AC D0 0018E       MOVL    OLD_FILESPEC, R0                             ; 0574
                                                 66 16 00192       JSB     LIB$ANALYZE_SDESC_R2
                              DC  AD             52 D0 00194       MOVL    R2, OLDFAB+44                                ; 0575
                                  71             50 E9 00198       BLBC    STATUS, 4$                                   ; 0577
                            00FF  8F             51 B1 0019B       CMPW    OLD_FILESPEC_LENGTH, #255                    ; 0581
                                                 75 1A 001A0       BGTRU   6$
                              E4  AD             51 90 001A2       MOVB    OLD_FILESPEC_LENGTH, OLDFAB+52               ; 0585
                                  50         08  AC D0 001A6       MOVL    NEW_FILESPEC, R0                             ; 0599
                                                 66 16 001AA       JSB     LIB$ANALYZE_SDESC_R2
                            FF3C  CD             52 D0 001AC       MOVL    R2, NEWFAB+44                                ; 0600
                                  58             50 E9 001B1       BLBC    STATUS, 4$                                   ; 0602
                        000000FF  8F             51 D1 001B4       CMPL    NEW_FILESPEC_LENGTH, #255                    ; 0606
                                                 5A 1A 001BB       BGTRU   6$
                            FF44  CD             51 90 001BD       MOVB    NEW_FILESPEC_LENGTH, NEWFAB+52               ; 0610
                                  03             6C 91 001C2       CMPB    (AP), #3                                     ; 0618
                                                 1D 1F 001C5       BLSSU   2$
                                             0C  AC D5 001C7       TSTL    12(AP)
                                                 18 13 001CA       BEQL    2$
                                  50         0C  AC D0 001CC       MOVL    DEFAULT_FILESPEC, R0                         ; 0626
                                                 66 16 001D0       JSB     LIB$ANALYZE_SDESC_R2
                              E0  AD             52 D0 001D2       MOVL    R2, OLDFAB+48                                ; 0627
                                  33             50 E9 001D6       BLBC    STATUS, 4$                                   ; 0629
                            00FF  8F             51 B1 001D9       CMPW    DEFAULT_FILESPEC_LENGTH, #255                ; 0633
                                                 37 1A 001DE       BGTRU   6$
                              E5  AD             51 90 001E0       MOVB    DEFAULT_FILESPEC_LENGTH, OLDFAB+53           ; 0637
                              0C  AE   020E0000  8F D0 001E4 2$:   MOVL    #34471936, RLF_DESC                          ; 0645
                                             10  AE D4 001EC       CLRL    RLF_DESC+4                                   ; 0648
                                  04             6C 91 001EF       CMPB    (AP), #4                                     ; 0649
                                                 4E 1F 001F2       BLSSU   8$
                                             10  AC D5 001F4       TSTL    16(AP)
                                                 49 13 001F7       BEQL    8$
                                  0C             6C 91 001F9       CMPB    (AP), #12                                    ; 0650
                                                 05 1F 001FC       BLSSU   3$
                                             30  AC D5 001FE       TSTL    48(AP)
                                                 3F 12 00201       BNEQ    8$
                                  50         10  AC D0 00203 3$:   MOVL    RELATED_FILESPEC, R0                         ; 0659
                                                 66 16 00207       JSB     LIB$ANALYZE_SDESC_R2
                            6E             51  D0 00209          MOVL     R1, RELATED_FILESPEC_LENGTH           ;
                            01             50  E8 0020C 4$:      BLBS     STATUS, 5$                            ; 0661
                                               04 0020F          RET                                            ;
                       00FF 8F             6E  31 00210 5$:      CMPW     RELATED_FILESPEC_LENGTH, #255         ; 0665
                                           08  1B 00215          BLEQU    7$                                    ;
                            50 00000000G   8F  D0 00217 6$:      MOVL     #LIB$_INVFILSPE, R0                   ; 0667
                                               04 0021E          RET                                            ;
                                      10   AE  9F 0021F 7$:      PUSHAB   RLF_DESC+4                            ; 0670
                                      04   AE  9F 00222          PUSHAB   RELATED_FILESPEC_LENGTH               ;
                  00000000G 00             02  FB 00225          CALLS    #2, LIB$GET_VM                        ;
                         0C AE             6E  B0 0022C          MOVW     RELATED_FILESPEC_LENGTH, RLF_DESC     ; 0671
          10  BE            62        0C   AE  28 00230          MOVC3    RLF_DESC, (RELATED_FILESPEC_ADDR), -  ; 0673
                                                                          @RLF_DESC+4                           ;
                       0317 CE        0C   AE  90 00236          MOVB     RLF_DESC, RLF_NAM+3                   ; 0674
                       0318 CE        10   AE  D0 0023C          MOVL     RLF_DESC+4, RLF_NAM+4                 ; 0675
                            05             6C  91 00242 8$:      CMPB     (AP), #5                              ; 0682
                                           15  1F 00245          BLSSU    9$                                    ;
                                      14   AC  D5 00247          TSTL     20(AP)                                ;
                                           10  13 0024A          BEQL     9$                                    ;
00        14  BC            1F             01  ED 0024C          CMPZV    #1, #31, @FLAGS, #0                   ; 0684
                                           08  13 00252          BEQL     9$                                    ;
                            50 00000000G   8F  D0 00254          MOVL     #LIB$_INVARG, R0                      ; 0686
                                               04 0025B          RET                                            ;
                            0C             6C  91 0025C 9$:      CMPB     (AP), #12                             ; 0692
                                           09  1F 0025F          BLSSU    10$                                   ;
                                      30   AC  D5 00261          TSTL     48(AP)                                ;
                                           04  13 00264          BEQL     10$                                   ;
                                    FE70   CD  D4 00266          CLRL     OLDNAM+16                             ; 0693
                                      08   AE  9F 0026A 10$:     PUSHAB   INTERCEPT_FLAG                        ; 0700
                                      08   AE  9F 0026D          PUSHAB   WORST_ERROR                           ;
                            0B             6C  91 00270          CMPB     (AP), #11                             ; 0714
                                           05  1F 00273          BLSSU    11$                                   ;
                                      2C   AC  DD 00275          PUSHL    NEW_RESULTANT_NAME                    ;
                                           02  11 00278          BRB      12$                                   ;
                                           7E  D4 0027A 11$:     CLRL     -(SP)                                 ;
                            0A             6C  91 0027C 12$:     CMPB     (AP), #10                             ; 0713
                                           05  1F 0027F          BLSSU    13$                                   ;
                                      28   AC  DD 00281          PUSHL    OLD_RESULTANT_NAME                    ;
                                           02  11 00284          BRB      14$                                   ;
                                           7E  D4 00286 13$:     CLRL     -(SP)                                 ;
                            09             6C  91 00288 14$:     CMPB     (AP), #9                              ; 0712
                                           05  1F 0028B          BLSSU    15$                                   ;
                                      24   AC  DD 0028D          PUSHL    USER_ARG                              ;
                                           02  11 00290          BRB      16$                                   ;
                                           7E  D4 00292 15$:     CLRL     -(SP)                                 ;
                            08             6C  91 00294 16$:     CMPB     (AP), #8                              ; 0711
                                           05  1F 00297          BLSSU    17$                                   ;
                                      20   AC  DD 00299          PUSHL    CONFIRM_ROUTINE                       ;
                                           02  11 0029C          BRB      18$                                   ;
                                           7E  D4 0029E 17$:     CLRL     -(SP)                                 ;
                            07             6C  91 002A0 18$:     CMPB     (AP), #7                              ; 0710
                                           05  1F 002A3          BLSSU    19$                                   ;
                                      1C   AC  DD 002A5          PUSHL    ERROR_ROUTINE                         ;
                                           02  11 002A8          BRB      20$                                   ;
                                           7E  D4 002AA 19$:     CLRL     -(SP)                                 ;
                            06             6C  91 002AC 20$:     CMPB     (AP), #6                              ; 0709
                                           05  1F 002AF          BLSSU    21$                                   ;
```

```
                     18  AC    DD 002B1          PUSHL   SUCCESS_ROUTINE               ;
                         02    11 002B4          BRB     22$                           ;
                         7E    D4 002B6 21$:     CLRL    -(SP)                         ;
           05            6C    91 002B8 22$:     CMPB    (AP), #5                      ; 0708
                         0A    1F 002BB          BLSSU   23$                           ;
                     14  AC    D5 002BD          TSTL    20(AP)                        ;
                         05    13 002C0          BEQL    23$                           ;
                     14  BC    DD 002C2          PUSHL   @FLAGS                        ;
                         02    11 002C5          BRB     24$                           ;
                         7E    D4 002C7 23$:     CLRL    -(SP)                         ;
                   FEC0  CD    9F 002C9 24$:     PUSHAB  NEWFAB_R                      ; 0700
                   FF60  CD    9F 002CD          PUSHAB  OLDFAB_R                      ;
                   FF10  CD    9F 002D1          PUSHAB  NEWFAB                        ;
           0C            6C    91 002D5          CMPB    (AP), #12                     ; 0704
                         05    1F 002D8          BLSSU   25$                           ;
                     30  AC    DD 002DA          PUSHL   FILE_SCAN_CONTEXT             ;
                         02    11 002DD          BRB     26$                           ;
                         7E    D4 002DF 25$:     CLRL    -(SP)                         ;
                  0000V  CF    9F 002E1 26$:     PUSHAB  RENAME_ERROR                  ; 0700
                  0000V  CF    9F 002E5          PUSHAB  DO_RENAME                     ;
                     B0  AD    9F 002E9          PUSHAB  OLDFAB                        ;
00000000G  00            10    FB 002EC          CALLS   #16, LIB$FILE_SCAN            ;
                     10  AE    9F 002F3          PUSHAB  RLF_DESC+4                    ; 0719
                     10  AE    9F 002F6          PUSHAB  RLF_DESC                      ; 0718
00000000G  00            02    FB 002F9          CALLS   #2, LIB$FREE_VM               ;
     FE93  CD        80  8F    8A 00300          BICB2   #128, OLDNAM+51               ; 0725
     FE68  CD            08    88 00306          BISB2   #8, OLDNAM+8                  ; 0726
                   FE62  CD    B4 0030B          CLRW    OLDNAM+2                      ; 0730
                   FE6A  CD    B4 0030F          CLRW    OLDNAM+10                     ; 0729
                   FE70  CD    D4 00313          CLRL    OLDNAM+16                     ; 0731
                     E4  AD    B4 00317          CLRW    OLDFAB+52                     ; 0732
                     B0  AD    9F 0031A          PUSHAB  OLDFAB                        ; 0734
00000000G  00            01    FB 0031D          CALLS   #1, SYS$PARSE                 ;
                     04  AE    D5 00324          TSTL    WORST_ERROR                   ; 0740
                         05    13 00327          BEQL    27$                           ;
           50        04  AE    D0 00329          MOVL    WORST_ERROR, RET_STATUS       ; 0742
                               04 0032D          RET                                   ;
           50            01    D0 0032E 27$:     MOVL    #1, RET_STATUS                ; 0744
                               04 00331          RET                                   ; 0748
                             0000 00332 28$:     .WORD   Save nothing                  ; 0487
           50        08  AC    D0 00334          MOVL    8(AP), R0                     ;
           50        04  A0    D0 00338          MOVL    4(R0), R0                     ;
                   F9D4  C0    9F 0033C          PUSHAB  INTERCEPT_FLAG                ;
                         01    DD 00340          PUSHL   #1                            ;
                         5E    DD 00342          PUSHL   SP                            ;
           7E        04  AC    7D 00344          MOVQ    4(AP), -(SP)                  ;
    0000V  CF            03    FB 00348          CALLS   #3, RENAME_HANDLER            ;
                               04 0034D          RET                                   ;

; Routine Size:  846 bytes,    Routine Base:  _LIB$CODE + 0002
```

```
1 %SBTTL 'DO_RENAME - Rename a file'
1 ROUTINE DO_RENAME (
1     OLDFAB: REF $FAB_DECL,                      ! Next input file FAB
1     UNUSED_1,                                   ! Unused here
1     UNUSED_2,                                   ! Unused here
1     UNUSED_3,                                   ! Unused here
1     NEWFAB: REF $FAB_DECL,                      ! FAB for $PARSE
1     OLDFAB_R: REF $FAB_DECL,                    ! Source for $RENAME
1     NEWFAB_R: REF $FAB_DECL,                    ! Dest for $RENAME
1     FLAGS: BLOCK [4, BYTE] FIELD (FLAGS_FIELDS),! Option flags
1     SUCCESS_ROUTINE,                            ! Success routine address
1     UNUSED_4,                                   ! Unused here
1     CONFIRM_ROUTINE,                            ! Confirm routine address
1     USER_ARG,                                   ! User argument
1     OLD_RESULTANT_NAME: REF BLOCK [, BYTE],     ! Old resultant filename
1     NEW_RESULTANT_NAME: REF BLOCK [, BYTE],     ! New resultant filename
1     UNUSED_5,                                   ! Unused here
1     INTERCEPT_FLAG: REF VECTOR [, LONG]         ! Intercept flag
1     ): NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine is called once for each file that LIB$FILE_SCAN
1 !       finds.  It renames the file described by OLDFAB to the filename
1 !       specified by NEWFAB.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       DO_RENAME is called by LIB$FILE_SCAN with the same arguments as
1 !       were passed to it by LIB$RENAME_FILE.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       See LIB$RENAME_FILE for a description of all parameters.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! COMPLETION STATUS:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       Renames a file
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
```

```
;  739          0806  2             OLDNAM: REF $NAM_DECL,            ! NAM of OLDFAB
;  740          0807  2             OLDNAM_R: REF $NAM_DECL,          ! NAM of OLDFAB_R
;  741          0808  2             NEWNAM: REF $NAM_DECL,            ! NAM of NEWFAB
;  742          0809  2             NEWNAM_R: REF $NAM_DECL,          ! NAM of NEWFAB_R
;  743          0810  2             OLD_LOCAL_DSC: BLOCK [8, BYTE],   ! Local descriptor for old name
;  744          0811  2             NEW_LOCAL_DSC: BLOCK [8, BYTE],   ! Local descriptor for new name
;  745          0812  2             OLD_STRING_PTR,                   ! Pointer to old string
;  746          0813  2             NEW_STRING_PTR;                   ! Pointer to new string
;  747          0814  2
;  748          0815  2         BUILTIN
;  749          0816  2             AP,                               ! Argument pointer
;  750          0817  2             CALLG;                            ! CALLG instruction
;  751          0818  2
;  752          0819  2         !+
;  753          0820  2         ! Set NAM addresses.
;  754          0821  2         !-
;  755          0822  2
;  756          0823  2         OLDNAM = .OLDFAB [FAB$L_NAM];
;  757          0824  2         OLDNAM_R = .OLDFAB_R [FAB$L_NAM];
;  758          0825  2         NEWNAM = .NEWFAB [FAB$L_NAM];
;  759          0826  2         NEWNAM_R = .NEWFAB_R [FAB$L_NAM];
;  760          0827  2
;  761          0828  2         !+
;  762          0829  2         ! Clear old statuses in FABs so that RENAME_ERROR can find the appropriate
;  763          0830  2         ! FAB when an error occurs.  We actually set the statuses to 1 for easy
;  764          0831  2         ! testing.
;  765          0832  2         !-
;  766          0833  2
;  767          0834  2         OLDFAB_R [FAB$L_STS] = 1;
;  768          0835  2         NEWFAB [FAB$L_STS] = 1;
;  769          0836  2         NEWFAB_R [FAB$L_STS] = 1;
;  770          0837  2
;  771          0838  2         !+
;  772          0839  2         ! To make LIB$RENAME look just like DCL RENAME, duplicate its
;  773          0840  2         ! handling of version numbers.  If the old filespec had a wildcard
;  774          0841  2         ! version, or if V_NONEW_VERSION was set, then use a default name of
;  775          0842  2         ! ';*' for the first $PARSE of the new filespec.  This will preserve
;  776          0843  2         ! the version number.  Otherwise use a null default name, which will
;  777          0844  2         ! create the next highest version (if none was specified originally)
;  778          0845  2         ! or will use the specific version in the old filespec.
;  779          0846  2         !
;  780          0847  2         ! Since NEWFAB [FAB$L_DNA] was initialized to the address of the string
;  781          0848  2         ! ';*', all we need to do is set the length if it is to be used.  If
;  782          0849  2         ! not, we need do nothing since its length is already zero!
;  783          0850  2         !-
;  784          0851  2
;  785          0852  2         IF .OLDNAM [NAM$V_WILD_VER] OR .FLAGS [V_NONEW_VERSION]
;  786          0853  2         THEN
;  787          0854  2             NEWFAB [FAB$B_DNS] = %CHARCOUNT (';*');
;  788          0855  2
;  789          0856  2         !+
;  790          0857  2         ! Call RMS to PARSE the new file name.  We must do two parses, the first,
;  791          0858  2         ! with OFP (output file parse) CLEAR, to copy any and all fields left blank
;  792          0859  2         ! in the new name from the old name (which is in the related file block),
;  793          0860  2         ! and the second, with OFP SET, to substitute any names which have been
;  794          0861  2         ! explicitly wildcarded in the output name as these have been ignored
;  795          0862  2         ! during the first parse.
```

```
2       !-
2
3       IF NOT $PARSE (FAB=.NEWFAB)
2       THEN
3           BEGIN
3           CALLG (.AP, RENAME_ERROR);
3           RETURN;
2           END;
2
2       !+
2       ! OFP is set in NEWFAB_R, so do the second parse there.
2       !-
2
2       NEWFAB_R [FAB$B_FNS] = .NEWNAM [NAM$B_ESL];
3       IF NOT $PARSE (FAB=.NEWFAB_R)
2       THEN
3           BEGIN
3           CALLG (.AP, RENAME_ERROR);
3           RETURN;
2           END;
2
2       !+
2       ! Copy the resultant file names to the user's strings or our own.
2       !-
2
2       OLD_STRING_PTR = .OLD_RESULTANT_NAME;
2       IF .OLD_STRING_PTR NEQA 0
2       THEN
3           BEGIN
3           LOCAL
3               COPY_STATUS;
3           COPY_STATUS = LIB$SCOPY_R_DX (%REF(.OLDNAM [NAM$B_RSL]),
3               .OLDNAM [NAM$L_RSA], .OLD_RESULTANT_NAME);
3           IF NOT .COPY_STATUS
3           THEN
4               BEGIN
4               INTERCEPT_FLAG [0] = 1;
4               SIGNAL_STOP (.COPY_STATUS);
4               RETURN;
3               END;
3           END
2       ELSE
3           BEGIN
3           OLD_STRING_PTR = OLD_LOCAL_DSC;          ! Use our string
3           OLD_LOCAL_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
3           OLD_LOCAL_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
3           OLD_LOCAL_DSC [DSC$W_LENGTH] = .OLDNAM [NAM$B_RSL];
3           OLD_LOCAL_DSC [DSC$A_POINTER] = .OLDNAM [NAM$L_RSA];
2           END;
2
2       NEW_STRING_PTR = .NEW_RESULTANT_NAME;
2       IF .NEW_STRING_PTR NEQA 0
2       THEN
3           BEGIN
3           LOCAL
3               COPY_STATUS;
3           COPY_STATUS = LIB$SCOPY_R_DX (%REF(.NEWNAM_R [NAM$B_ESL]),
;  853          0920  3                 .NEWNAM_R [NAM$L_ESA], .NEW_RESULTANT_NAME);
;  854          0921  3             IF NOT .COPY_STATUS
;  855          0922  3             THEN
;  856          0923  4                 BEGIN
;  857          0924  4                 INTERCEPT_FLAG [0] = 1;
;  858          0925  4                 SIGNAL_STOP (.COPY_STATUS);
;  859          0926  4                 RETURN;
;  860          0927  3                 END;
;  861          0928  3             END
;  862          0929  2         ELSE
;  863          0930  3             BEGIN
;  864          0931  3             NEW_STRING_PTR = NEW_LOCAL_DSC;         ! Use our string
;  865          0932  3             NEW_LOCAL_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  866          0933  3             NEW_LOCAL_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
;  867          0934  3             NEW_LOCAL_DSC [DSC$W_LENGTH] = .NEWNAM_R [NAM$B_ESL];
;  868          0935  3             NEW_LOCAL_DSC [DSC$A_POINTER] = .NEWNAM_R [NAM$L_ESA];
;  869          0936  2             END;
;  870          0937  2
;  871          0938  2         !+
;  872          0939  2         ! If there is a CONFIRM_ROUTINE, ask it if it wants the file renamed.
;  873          0940  2         ! Note use of linkage name BLISS to perform a general-routine-call.
;  874          0941  2         !-
;  875          0942  2
;  876          0943  2         IF .CONFIRM_ROUTINE NEQA 0
;  877          0944  2         THEN
;  878          0945  2             IF NOT BLISS (.CONFIRM_ROUTINE, .OLD_STRING_PTR, .NEW_STRING_PTR,
;  879          0946  2                 .OLDFAB, .USER_ARG)
;  880          0947  2             THEN
;  881          0948  2                 RETURN;
;  882          0949  2
;  883          0950  2         !+
;  884          0951  2         ! Now set up OLDFAB_R and NEWFAB_R and do the $RENAME.
;  885          0952  2         !-
;  886          0953  2
;  887          0954  2         OLDFAB_R [FAB$B_FNS] = .OLDNAM [NAM$B_RSL];
;  888          0955  2         NEWFAB_R [FAB$B_FNS] = .NEWNAM_R [NAM$B_ESL];
;  889          0956  3         IF NOT $RENAME (OLDFAB=.OLDFAB_R, NEWFAB=.NEWFAB_R)
;  890          0957  2         THEN
;  891          0958  3             BEGIN
;  892          0959  3             CALLG (.AP, RENAME_ERROR);
;  893          0960  3             RETURN;
;  894          0961  2             END;
;  895          0962  2
;  896          0963  2         !+
;  897          0964  2         ! We need a new "new resultant file name" string.  Copy it to the
;  898          0965  2         ! user's string or reset our own.
;  899          0966  2         !-
;  900          0967  2
;  901          0968  2         IF .NEW_RESULTANT_NAME NEQA 0
;  902          0969  2         THEN
;  903          0970  3             BEGIN
;  904          0971  3             LOCAL
;  905          0972  3                 COPY_STATUS;
;  906          0973  3             COPY_STATUS = LIB$SCOPY_R_DX (%REF(.NEWNAM_R [NAM$B_RSL]),
;  907          0974  3                 .NEWNAM_R [NAM$L_RSA], .NEW_RESULTANT_NAME);
;  908          0975  3             IF NOT .COPY_STATUS
;  909          0976  3             THEN
;   910           0977  4               BEGIN
;   911           0978  4               INTERCEPT_FLAG [0] = 1;
;   912           0979  4               SIGNAL_STOP (.COPY_STATUS);
;   913           0980  4               RETURN;
;   914           0981  3               END;
;   915           0982  3           END
;   916           0983  2       ELSE
;   917           0984  2           NEW_LOCAL_DSC [DSC$W_LENGTH] = .NEWNAM_R [NAM$B_RSL];
;   918           0985  2
;   919           0986  2
;   920           0987  2       !+
;   921           0988  2       ! If there is a success routine, call it.
;   922           0989  2       !-
;   923           0990  2
;   924           0991  2       IF .SUCCESS_ROUTINE NEQA 0
;   925           0992  2       THEN
;   926           0993  2           BLISS (.SUCCESS_ROUTINE, .OLD_STRING_PTR, .NEW_STRING_PTR,
;   927           0994  2               .USER_ARG);
;   928           0995  2
;   929           0996  2       RETURN;
;   930           0997  2
;   931           0998  1       END;                                            ! End of routine DO_RENAME



                                                                  .EXTRN  SYS$RENAME

                                             01FC 00000 DO_RENAME:
                                                                  .WORD   Save R2,R3,R4,R5,R6,R7,R8                     ; 0750
                      58 00000000G   00   9E 00002                MOVAB   SYS$PARSE, R8                                 ;
                      57 00000000G   00   9E 00009                MOVAB   LIB$SCOPY_R_DX, R7                            ;
                      5E            14   C2 00010                 SUBL2   #20, SP                                       ;
                      50       04   AC   D0 00013                 MOVL    OLDFAB, R0                                    ; 0823
                      53       28   A0   D0 00017                 MOVL    40(R0), OLDNAM                                ;
                      52       18   AC   D0 0001B                 MOVL    OLDFAB_R, R2                                  ; 0824
                      50       28   A2   D0 0001F                 MOVL    40(R2), OLDNAM_R                              ;
                      50       14   AC   D0 00023                 MOVL    NEWFAB, R0                                    ; 0825
                      55       28   A0   D0 00027                 MOVL    40(R0), NEWNAM                                ;
                      51       1C   AC   D0 0002B                 MOVL    NEWFAB_R, R1                                  ; 0826
                      54       28   A1   D0 0002F                 MOVL    40(R1), NEWNAM_R                              ;
                 08   A2            01   D0 00033                 MOVL    #1, 8(R2)                                     ; 0834
                 08   A0            01   D0 00037                 MOVL    #1, 8(R0)                                     ; 0835
                 08   A1            01   D0 0003B                 MOVL    #1, 8(R1)                                     ; 0836
         04      34   A3            03   E0 0003F                 BBS     #3, 52(OLDNAM), 1$                            ; 0852
                      04       20   AC   E9 00044                 BLBC    FLAGS, 2$                                     ;
                 35   A0            02   90 00048 1$:             MOVB    #2, 53(R0)                                    ; 0854
                                    50   DD 0004C 2$:             PUSHL   R0                                            ; 0865
                      68            01   FB 0004E                 CALLS   #1, SYS$PARSE                                 ;
                      0E            50   E9 00051                 BLBC    R0, 3$                                        ;
                      50       1C   AC   D0 00054                 MOVL    NEWFAB_R, R0                                  ; 0876
                 34   A0       0B   A5   90 00058                 MOVB    11(NEWNAM), 52(R0)                            ;
                                    50   DD 0005D                 PUSHL   R0                                            ; 0877
                      68            01   FB 0005F                 CALLS   #1, SYS$PARSE                                 ;
                      03            50   E8 00062 3$:             BLBS    R0, 4$                                        ;
                                  00A2   31 00065                 BRW     11$                                           ;
                      56       34   AC   D0 00068 4$:             MOVL    OLD_RESULTANT_NAME, OLD_STRING_PTR            ; 0888
                                    19   13 0006C                 BEQL    5$                                            ; 0889
```

```
                                 34    AC    DD 0006E           PUSHL    OLD_RESULTANT_NAME                          ; 0895
                                 04    A3    DD 00071           PUSHL    4(OLDNAM)                                   ;
                  52             03    A3    9A 00074           MOVZBL   3(OLDNAM), R2                               ; 0894
            08    AE                   52    D0 00078           MOVL     R2, 8(SP)                                   ;
                                 08    AE    9F 0007C           PUSHAB   8(SP)                                       ;
                  67                   03    FB 0007F           CALLS    #3, LIB$SCOPY_R_DX                          ;
                  19                   50    E8 00082           BLBS     COPY_STATUS, 6$                             ; 0896
                                       34    11 00085           BRB      7$                                          ; 0899
                  56             0C    AE    9E 00087 5$:       MOVAB    OLD_LOCAL_DSC, OLD_STRING_PTR               ; 0906
            0E    AE           010E    8F    B0 0008B           MOVW     #270, OLD_LOCAL_DSC+2                       ; 0907
                  52             03    A3    9A 00091           MOVZBL   3(OLDNAM), R2                               ; 0909
            0C    AE                   52    B0 00095           MOVW     R2, OLD_LOCAL_DSC                           ;
            10    AE             04    A3    D0 00099           MOVL     4(OLDNAM), OLD_LOCAL_DSC+4                  ; 0910
                  55             38    AC    D0 0009E 6$:       MOVL     NEW_RESULTANT_NAME, NEW_STRING_PTR          ; 0913
                                       19    13 000A2           BEQL     8$                                          ; 0914
                                 38    AC    DD 000A4           PUSHL    NEW_RESULTANT_NAME                          ; 0920
                                 0C    A4    DD 000A7           PUSHL    12(NEWNAM_R)                                ;
                  53             0B    A4    9A 000AA           MOVZBL   11(NEWNAM_R), R3                            ; 0919
            08    AE                   53    D0 000AE           MOVL     R3, 8(SP)                                   ;
                                 08    AE    9F 000B2           PUSHAB   8(SP)                                       ;
                  67                   03    FB 000B5           CALLS    #3, LIB$SCOPY_R_DX                          ;
                  19                   50    E8 000B8           BLBS     COPY_STATUS, 9$                             ; 0921
                                       6C    11 000BB 7$:       BRB      13$                                         ; 0924
                  55             04    AE    9E 000BD 8$:       MOVAB    NEW_LOCAL_DSC, NEW_STRING_PTR               ; 0931
            06    AE           010E    8F    B0 0C0C1           MOVW     #270, NEW_LOCAL_DSC+2                       ; 0932
                  53             0B    A4    9A 000C7           MOVZBL   11(NEWNAM_R), R3                            ; 0934
            04    AE                   53    B0 000CB           MOVW     R3, NEW_LOCAL_DSC                           ;
            08    AE             0C    A4    D0 000CF           MOVL     12(NEWNAM_R), NEW_LOCAL_DSC+4               ; 0935
                                 2C    AC    D5 000D4 9$:       TSTL     CONFIRM_ROUTINE                             ; 0943
                                       11    13 000D7           BEQL     10$                                         ;
                                 30    AC    DD 000D9           PUSHL    USER_ARG                                    ; 0946
                                 04    AC    DD 000DC           PUSHL    OLDFAB                                      ;
                                       55    DD 000DF           PUSHL    NEW_STRING_PTR                              ; 0945
                                       56    DD 000E1           PUSHL    OLD_STRING_PTR                              ;
            2C    BC                   04    FB 000E3           CALLS    #4, @CONFIRM_ROUTINE                        ;
                  62                   50    E9 000E7           BLBC     R0, 16$                                     ;
                  51             18    AC    D0 000EA 10$:      MOVL     OLDFAB_R, R1                                ; 0954
            34    A1                   52    90 000EE           MOVB     R2, 52(R1)                                  ;
                  50             1C    AC    D0 000F2           MOVL     NEWFAB_R, R0                                ; 0955
            34    A0                   53    90 000F6           MOVB     R3, 52(R0)                                  ;
                                       50    DD 000FA           PUSHL    R0                                          ; 0956
                                       7E    7C 000FC           CLRQ     -(SP)                                       ;
                                       51    DD 000FE           PUSHL    R1                                          ;
     00000000G    00                   04    FB 00100           CALLS    #4, SYS$RENAME                              ;
                  06                   50    E8 00107           BLBS     R0, 12$                                     ;
         0000V    CF                   6C    FA 0010A 11$:      CALLG    (AP), RENAME_ERROR                          ; 0959
                                             04 0010F           RET                                                  ; 0958
                                 38    AC    D5 00110 12$:      TSTL     NEW_RESULTANT_NAME                          ; 0968
                                       22    13 00113           BEQL     14$                                         ;
                                 38    AC    DD 00115           PUSHL    NEW_RESULTANT_NAME                          ; 0974
                                 04    A4    DD 00118           PUSHL    4(NEWNAM_R)                                 ;
            08    AE             03    A4    9A 0011B           MOVZBL   3(NEWNAM_R), 8(SP)                          ; 0973
                                 08    AE    9F 00120           PUSHAB   8(SP)                                       ;
                  67                   03    FB 00123           CALLS    #3, LIB$SCOPY_R_DX                          ;
                  13                   50    E8 00126           BLBS     COPY_STATUS, 15$                            ; 0975
            40    BC                   01    D0 00129 13$:      MOVL     #1, @INTERCEPT_FLAG                         ; 0978
                                       50    DD 0012D           PUSHL    COPY_STATUS                                 ; 0979
```

**F

```
               00000000G  00           01  FB 0012F          CALLS    #1, LIB$STOP                            ;
                                           04 00136          RET                                              ; 0977
                      04  AE       03  A4  9B 00137 14$:     MOVZBW   3(NEWNAM_R), NEW_LOCAL_DSC              ; 0984
                                   24  AC  D5 0013C 15$:     TSTL     SUCCESS_ROUTINE                         ; 0991
                                       0B  13 0013F          BEQL     16$                                     ;
                                   30  AC  DD 00141          PUSHL    USER_ARG                                ; 0994
                                       55  DD 00144          PUSHL    NEW_STRING_PTR                          ; 0993
                                       56  DD 00146          PUSHL    OLD_STRING_PTR                          ;
                      24  BC           03  FB 00148          CALLS    #3, @SUCCESS_ROUTINE                    ;
                                           04 0014C 16$:     RET                                              ; 0998

; Routine Size:  333 bytes,    Routine Base:  _LIB$CODE + 0350
```

```
1 %SBTTL 'RENAME_ERROR - Report error during rename'
1 ROUTINE RENAME_ERROR (
1     OLDFAB: REF $FAB_DECL,                      ! Next input file FAB
1     UNUSED_1,                                   ! Unused here
1     UNUSED_2,                                   ! Unused here
1     UNUSED_3,                                   ! Unused here
1     NEWFAB: REF $FAB_DECL,                      ! FAB for $PARSE
1     OLDFAB_R: REF $FAB_DECL,                    ! Source for $RENAME
1     NEWFAB_R: REF $FAB_DECL,                    ! Dest for $RENAME
1     UNUSED_4,                                   ! Unused here
1     UNUSED_5,                                   ! Unused here
1     ERROR_ROUTINE,                              ! Error routine address
1     UNUSED_6,                                   ! Unused here
1     USER_ARG,                                   ! User argument
1     OLD_RESULTANT_NAME: REF BLOCK [, BYTE],     ! Old resultant filename
1     NEW_RESULTANT_NAME: REF BLOCK [, BYTE],     ! New resultant filename
1     WORST_ERROR: REF BLOCK [4, BYTE],           ! Worst error so far
1     INTERCEPT_FLAG: REF VECTOR [, LONG]         ! Intercept flag
1     ): NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine is called when LIB$FILE_SCAN detects an error.
1 !       It calls the user's error routine, if one exists.
1 !       If the user's error routine returns success, or if there is
1 !       no user error routine, processing of remaining files continues.
1 !       If the user's error routine returns failure, the error is
1 !       signalled.  This signal is converted to the return status of
1 !       LIB$RENAME_FILE.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       Called from LIB$FILE_SCAN and DO_RENAME with the same arguments
1 !       as passed to LIB$FILE_SCAN from LIB$RENAME_FILE.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       See body of LIB$RENAME_FILE for descriptions of arguments.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! COMPLETION STATUS:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       May signal the error
1 !
1 !--
```

```
;  990       1056  1
;  991       1057  2        BEGIN
;  992       1058  2
;  993       1059  2        LOCAL
;  994       1060  2            OLDFAB_PTR: REF $FAB_DECL,            ! Pointer to OLDFAB with error
;  995       1061  2            NEWFAB_PTR: REF $FAB_DECL,            ! Pointer to NEWFAB with error
;  996       1062  2            ERRFAB: REF $FAB_DECL,                ! Pointer to FAB with statuses
;  997       1063  2            ERROR_SOURCE,                         ! Code that indicates source of
;  998       1064  2                                                  ! error: 0 = file scan error
;  999       1065  2                                                  !        1 = $PARSE error
; 1000       1066  2                                                  !        2 = $RENAME error
; 1001       1067  2            OLD_LOCAL_DSC: BLOCK [8, BYTE],       ! Local descriptor for old name
; 1002       1068  2            NEW_LOCAL_DSC: BLOCK [8, BYTE],       ! Local descriptor for new name
; 1003       1069  2            OLD_STRING_PTR,                       ! Pointer to old string
; 1004       1070  2            NEW_STRING_PTR;                       ! Pointer to new string
; 1005       1071  2
; 1006       1072  2        !+
; 1007       1073  2        ! Find out which OLDFAB and NEWFAB has the error.
; 1008       1074  2        !-
; 1009       1075  2
; 1010       1076  2        IF NOT .OLDFAB [FAB$L_STS]
; 1011       1077  2        THEN
; 1012       1078  3            BEGIN
; 1013       1079  3            !+
; 1014       1080  3            ! Error from LIB$FILE_SCAN.
; 1015       1081  3            !-
; 1016       1082  3            LOCAL
; 1017       1083  3                NAM: REF $NAM_DECL;         ! NAM block
; 1018       1084  3
; 1019       1085  3            ERRFAB = .OLDFAB;
; 1020       1086  3            OLDFAB_PTR = .OLDFAB;
; 1021       1087  3            NEWFAB_PTR = .NEWFAB;
; 1022       1088  3            NAM = .NEWFAB_PTR [FAB$L_NAM];
; 1023       1089  3            NAM [NAM$B_RSL] = 0;     ! Just use FNM field in NEWFAB
; 1024       1090  3            NAM [NAM$B_ESL] = 0;
; 1025       1091  3            ERROR_SOURCE = 0;        ! Indicate error finding file
; 1026       1092  3            END
; 1027       1093  2        ELSE
; 1028       1094  2            !+
; 1029       1095  2            ! It's not an error from LIB$FILE_SCAN.  See if it's the error from
; 1030       1096  2            ! one of the $PARSEs or the $RENAME.
; 1031       1097  2            !-
; 1032       1098  2            IF NOT .OLDFAB_R [FAB$L_STS]
; 1033       1099  2            THEN
; 1034       1100  3                BEGIN
; 1035       1101  3                !+
; 1036       1102  3                ! It's the $RENAME.
; 1037       1103  3                !-
; 1038       1104  3                ERRFAB = .OLDFAB_R;
; 1039       1105  3                OLDFAB_PTR = .OLDFAB_R;
; 1040       1106  3                NEWFAB_PTR = .NEWFAB_R;
; 1041       1107  3                ERROR_SOURCE = 2;           ! Indicate RENAME error
; 1042       1108  3                END
; 1043       1109  2            ELSE IF NOT .NEWFAB [FAB$L_STS]
; 1044       1110  2            THEN
; 1045       1111  3                BEGIN
; 1046       1112  3                !+
3             ! It's the first $PARSE.
3             !-
3             ERRFAB = .NEWFAB;
3             OLDFAB_PTR = .OLDFAB;
3             NEWFAB_PTR = .NEWFAB;
3             ERROR_SOURCE = 1;           ! Indicate $PARSE error
3             END
2         ELSE
3             BEGIN
3             !+
3             ! Must be the second $PARSE.
3             !-
3             ERRFAB = .NEWFAB_R;
3             OLDFAB_PTR = .OLDFAB;
3             NEWFAB_PTR = .NEWFAB_R;
3             ERROR_SOURCE = 1;           ! Indicate $PARSE error
3             END;
2
2     !+
2     ! Call COPY_RESULTANT_NAME to copy the resultant name strings into either
2     ! the user's strings or our own.
2     !-
2
2     OLD_STRING_PTR = COPY_RESULTANT_NAME (.OLDFAB_PTR, OLD_LOCAL_DSC,
2         .OLD_RESULTANT_NAME, .INTERCEPT_FLAG);
2     NEW_STRING_PTR = COPY_RESULTANT_NAME (.NEWFAB_PTR, NEW_LOCAL_DSC,
2         .NEW_RESULTANT_NAME, .INTERCEPT_FLAG);
2
2     !+
2     ! If a user error routine has been specified, call it with arguments
2     ! of the filename, STS, STV, error source, and user argument.  If it returns
2     ! failure, signal the error.  Set WORST_ERROR to LIB$_ERRROUCAL so
2     ! that our caller can tell that ERROR_ROUTINE was called.
2     !-
2
2     IF .ERROR_ROUTINE NEQA 0
2     THEN
3         BEGIN
3         IF NOT BLISS (.ERROR_ROUTINE, .OLD_STRING_PTR, .NEW_STRING_PTR,
3             ERRFAB [FAB$L_STS], ERRFAB [FAB$L_STV],
3             ERROR_SOURCE, .USER_ARG)
3         THEN
4             BEGIN
4             INTERCEPT_FLAG [0] = 1;     ! Cause handler to intercept signal
4             SIGNAL_STOP (.ERRFAB [FAB$L_STS]);
4             RETURN;
3             END;
3         WORST_ERROR [0,0,32,0] = LIB$_ERRROUCAL;
3         END
2     ELSE
3         BEGIN
3         !+
3         ! If this error is worse than any previous errors, store it in
3         ! WORST_ERROR.  (Use GEQU so that the initial zero gets replaced if
3         ! the error is a warning.)
3         !-
3
```

```
; 1104          1170  3            IF .BLOCK [ERRFAB [FAB$L_STS], STS$V_SEVERITY;4, BYTE] GEQU
; 1105          1171  3                .WORST_ERROR [STS$V_SEVERITY]
; 1106          1172  3            THEN
; 1107          1173  3                WORST_ERROR [0,0,32,0] = .ERRFAB [FAB$L_STS];
; 1108          1174  2            END;
; 1109          1175  2
; 1110          1176  2        RETURN;
; 1111          1177  2
; 1112          1178  1        END;                                    ! End of routine RENAME_ERROR



                                      001C 00000 RENAME_ERROR:
                                                       .WORD   Save R2,R3,R4                          ; 1000
                  5E          14  C2 00002             SUBL2   #20, SP                                ;
                  51    04    AC  D0 00005             MOVL    OLDFAB, R1                             ; 1076
                  18    08    A1  E8 00009             BLBS    8(R1), 1$                              ;
                  52          51  D0 0000D             MOVL    R1, ERRFAB                             ; 1085
                  54          51  D0 00010             MOVL    R1, OLDFAB_PTR                         ; 1086
                  53    14    AC  D0 00013             MOVL    NEWFAB, NEWFAB_PTR                     ; 1087
                  50    28    A3  D0 00017             MOVL    40(NEWFAB_PTR), NAM                    ; 1088
                        03    A0  94 0001B             CLRB    3(NAM)                                 ; 1089
                        0B    A0  94 0001E             CLRB    11(NAM)                                ; 1090
                              6E  D4 00021             CLRL    ERROR_SOURCE                           ; 1091
                              35  11 00023             BRB     4$                                     ; 1076
                  50    18    AC  D0 00025 1$:         MOVL    OLDFAB_R, R0                           ; 1098
                  0F    08    A0  E8 00029             BLBS    8(R0), 2$                              ;
                  52          50  D0 0002D             MOVL    R0, ERRFAB                             ; 1104
                  54          50  D0 00030             MOVL    R0, OLDFAB_PTR                         ; 1105
                  53    1C    AC  D0 00033             MOVL    NEWFAB_R, NEWFAB_PTR                   ; 1106
                  6E          02  D0 00037             MOVL    #2, ERROR_SOURCE                       ; 1107
                              1E  11 0003A             BRB     4$                                     ; 1098
                  54          51  D0 0003C 2$:         MOVL    R1, OLDFAB_PTR                         ; 1116
                  6E          01  D0 0003F             MOVL    #1, ERROR_SOURCE                       ; 1118
                  50    14    AC  D0 00042             MOVL    NEWFAB, R0                             ; 1109
                  08    08    A0  E8 00046             BLBS    8(R0), 3$                              ;
                  52          50  D0 0004A             MOVL    R0, ERRFAB                             ; 1115
                  53          50  D0 0004D             MOVL    R0, NEWFAB_PTR                         ; 1117
                              08  11 00050             BRB     4$                                     ; 1109
                  52    1C    AC  D0 00052 3$:         MOVL    NEWFAB_R, ERRFAB                       ; 1125
                  53    1C    AC  D0 00056             MOVL    NEWFAB_R, NEWFAB_PTR                   ; 1127
                        40    AC  DD 0005A 4$:         PUSHL   INTERCEPT_FLAG                         ; 1137
                        34    AC  DD 0005D             PUSHL   OLD_RESULTANT_NAME                     ;
                        14    AE  9F 00060             PUSHAB  OLD_LOCAL_DSC                          ; 1136
                              54  DD 00063             PUSHL   OLDFAB_PTR                             ;
            0000V  CF         04  FB 00065             CALLS   #4, COPY_RESULTANT_NAME                ;
                  54          50  D0 0006A             MOVL    R0, OLD_STRING_PTR                     ;
                        40    AC  DD 0006D             PUSHL   INTERCEPT_FLAG                         ; 1139
                        38    AC  DD 00070             PUSHL   NEW_RESULTANT_NAME                     ;
                        0C    AE  9F 00073             PUSHAB  NEW_LOCAL_DSC                          ; 1138
                              53  DD 00076             PUSHL   NEWFAB_PTR                             ;
            0000V  CF         04  FB 00078             CALLS   #4, COPY_RESULTANT_NAME                ;
                        28    AC  D5 0007D             TSTL    ERROR_ROUTINE                          ; 1148
                              2F  13 00080             BEQL    6$                                     ;
                        30    AC  DD 00082             PUSHL   USER_ARG                               ; 1153
```

```
                                                  04   AE   9F 00085         PUSHAB  ERROR_SOURCE                               ; 1152
                                                  0C   A2   9F 00088         PUSHAB  12(ERRFAB)                                 ;
                                                  08   A2   9F 0008B         PUSHAB  8(ERRFAB)                                  ;
                                                       50   DD 0008E         PUSHL   NEW_STRING_PTR                             ;
                                                       54   DD 00090         PUSHL   OLD_STRING_PTR                             ;
                                    28   BC            06   FB 00092         CALLS   #6, @ERROR_ROUTINE                         ;
                                         0F            50   E8 00096         BLBS    R0, 5$                                     ;
                                    40   BC            01   D0 00099         MOVL    #1, @INTERCEPT_FLAG                        ; 1156
                                                  08   A2   DD 0009D         PUSHL   8(ERRFAB)                                  ; 1157
                             00000000G   00            01   FB 000A0         CALLS   #1, LIB$STOP                               ;
                                                       04   000A7            RET                                                ; 1155
                                    3C   BC 00000000G  8F   D0 000A8 5$:     MOVL    #LIB$_ERRROUCAL, @WORST_ERROR              ; 1160
                                                       04   000B0            RET                                                ; 1148
           50        3C   BC             03            00   EF 000B1 6$:     EXTZV   #0, #3, @WORST_ERROR, R0                   ; 1171
           50        08   A2             03            00   ED 000B7         CMPZV   #0, #3, 8(ERRFAB), R0                      ;
                                                       05   1F 000BD         BLSSU   7$                                         ;
                                    3C   BC       08   A2   D0 000BF         MOVL    8(ERRFAB), @WORST_ERROR                    ; 1173
                                                       04   000C4 7$:        RET                                                ; 1178

; Routine Size:  197 bytes,    Routine Base:  _LIB$CODE + 049D
```

```
1 %SBTTL 'COPY_RESULTANT_NAME - Copy resultant filename'
1 ROUTINE COPY_RESULTANT_NAME (
1         FAB: REF $FAB_DECL,                       ! FAB of error
1         LOCAL_DSC: REF BLOCK [, BYTE],            ! Our local descriptor
1         RESULTANT_NAME: REF BLOCK [, BYTE],       ! Caller's string
1         INTERCEPT_FLAG: REF VECTOR [, LONG]       ! Intercept flag
1         ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine is called from RENAME_ERROR to copy the resultant
1 !       file name (or best approximation) to either the user's string
1 !       or our own.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       string_ptr.wa.v = COPY_RESULTANT_NAME (FAB.rr.r, LOCAL_DSC.wq.r,
1 !               [RESULTANT_NAME.wt.dx], INTERCEPT_FLAG.wl.r)
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       FAB             - The FAB referencing the name we want.
1 !
1 !       LOCAL_DSC       - The descriptor which will get filled in with
1 !                         a class S, type T descriptor of the string.
1 !
1 !       RESULTANT_NAME  - The user's string, if any.
1 !
1 !       INTERCEPT_FLAG  - The flag which we will set if we want to signal
1 !                         an error so that RENAME_HANDLER will catch it.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! FUNCTION VALUE:
1 !
1 !       The address of the user's string, if present, else LOCAL_DSC.
1 !
1 ! SIDE EFFECTS:
1 !
1 !       May signal an error
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
2       NAM: REF $NAM_DECL,                 ! RMS NAM
2       STRING_PTR,                         ! Pointer to the string to use
2       COPY_STATUS;                        ! Status from LIB$SCOPY
2
```

```
2       !+
2       ! Set NAM to point to the NAM block.  Fill in LOCAL_DSC.
2
2       NAM = .FAB [FAB$L_NAM];
2       LOCAL_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
2       LOCAL_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
2
2       IF .NAM [NAM$B_RSL] NEQ 0   ! Resultant filename present?
2       THEN
3           BEGIN
3           LOCAL_DSC [DSC$W_LENGTH] = .NAM [NAM$B_RSL];
3           LOCAL_DSC [DSC$A_POINTER] = .NAM [NAM$L_RSA];
3           END
2       ELSE IF .NAM [NAM$B_ESL] NEQ 0      ! Expanded filename present?
2       THEN
3           BEGIN
3           LOCAL_DSC [DSC$W_LENGTH] = .NAM [NAM$B_ESL];
3           LOCAL_DSC [DSC$A_POINTER] = .NAM [NAM$L_ESA];
3           END
2       ELSE                                ! Use filename from FAB
3           BEGIN
3           LOCAL_DSC [DSC$W_LENGTH] = .FAB [FAB$B_FNS];
3           LOCAL_DSC [DSC$A_POINTER] = .FAB [FAB$L_FNA];
2           END;
2
2       !+
2       ! If the user has specified RESULTANT_NAME, copy the filename to it.
2       !-
2
2       STRING_PTR = .RESULTANT_NAME;
2       IF .STRING_PTR NEQA 0
2       THEN
3           BEGIN
3           COPY_STATUS = LIB$SCOPY_DXDX (.LOCAL_DSC, .RESULTANT_NAME);
3           IF NOT .COPY_STATUS
3           THEN
4               BEGIN
4               INTERCEPT_FLAG [0] = 1;     ! Cause handler to intercept signal
4               SIGNAL_STOP (.COPY_STATUS);
4               RETURN .STRING_PTR;         ! Won't get executed
3               END;
3           END
2       ELSE
2           STRING_PTR = .LOCAL_DSC;        ! Use our own string
2
2       RETURN .STRING_PTR;                 ! Return pointer to string
2
1       END;                                ! End of routine COPY_RESULTANT_NAME


                              0004 00000 COPY_RESULTANT_NAME:
                                                 .WORD   Save R2                         ; 1180
          52       04   AC  D0 00002             MOVL    FAB, R2                         ; 1239
          50       28   A2  D0 00006             MOVL    40(R2), NAM                     ;
```

```
                         51        08   AC  D0 0000A          MOVL    LOCAL_DSC, R1                         ; 1240
                   02    A1      010E   8F  B0 0000E          MOVW    #270, 2(R1)                           ;
                                   03   A0  95 00014          TSTB    3(NAM)                                ; 1243
                                        0B  13 00017          BEQL    1$                                    ;
                         61        03   A0  9B 00019          MOVZBW  3(NAM), (R1)                          ; 1246
                   04    A1        04   A0  D0 0001D          MOVL    4(NAM), 4(R1)                         ; 1247
                                        19  11 00022          BRB     3$                                    ; 1243
                                   0B   A0  95 00024 1$:      TSTB    11(NAM)                               ; 1249
                                        0B  13 00027          BEQL    2$                                    ;
                         61        0B   A0  9B 00029          MOVZBW  11(NAM), (R1)                         ; 1252
                   04    A1        0C   A0  D0 0002D          MOVL    12(NAM), 4(R1)                        ; 1253
                                        09  11 00032          BRB     3$                                    ; 1249
                         61        34   A2  9B 00034 2$:      MOVZBW  52(R2), (R1)                          ; 1257
                   04    A1        2C   A2  D0 00038          MOVL    44(R2), 4(R1)                         ; 1258
                         52        0C   AC  D0 0003D 3$:      MOVL    RESULTANT_NAME, STRING_PTR            ; 1265
                                        1E  13 00041          BEQL    4$                                    ; 1266
                                   0C   AC  DD 00043          PUSHL   RESULTANT_NAME                        ; 1269
                                        51  DD 00046          PUSHL   R1                                    ;
            00000000G    00             02  FB 00048          CALLS   #2, LIB$SCOPY_DXDX                    ;
                         12             50  E8 0004F          BLBS    COPY_STATUS, 5$                       ; 1270
                   10    BC             01  D0 00052          MOVL    #1, @INTERCEPT_FLAG                   ; 1273
                                        50  DD 00056          PUSHL   COPY_STATUS                           ; 1274
            00000000G    00             01  FB 00058          CALLS   #1, [IB$STOP                          ;
                                        03  11 0005F          BRB     5$                                    ; 1275
                         52             51  D0 00061 4$:      MOVL    R1, STRING_PTR                        ; 1279
                         50             52  D0 00064 5$:      MOVL    STRING_PTR, R0                        ; 1281
                                            04 00067          RET                                           ; 1283

; Routine Size:  104 bytes,    Routine Base:  _LIB$CODE + 0562
```

```
1 %SBTTL 'RENAME_HANDLER - Local condition handler'
1 ROUTINE RENAME_HANDLER (
1         SIGNAL_ARGS: REF BLOCK [, BYTE],        ! Signal arguments array
1         MECH_ARGS: REF BLOCK [, BYTE],          ! Mechanism arguments array
1         ENABLE_ARGS: REF VECTOR [, LONG]        ! Enable arguments array
1     ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This is the condition handler enabled by LIB$RENAME_FILE.
1 !       If this is not an unwind, and if the INTERCEPT_FLAG enable
1 !       argument is set, then LIB$SIG_TO_RET is called to convert the
1 !       signal to a return status.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !       status.wlc.v = RENAME_HANDLER (SIGNAL_ARGS.rl.ra, MECH_ARGS.rl.ra
1 !                                     , ENABLE_ARGS.rl.ra)
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       SIGNAL_ARGS     - The signal argument list.
1 !
1 !       MECH_ARGS       - The mechanism argument list.
1 !
1 !       ENABLE_ARGS     - The enable argument list.  The one enable
1 !                         argument is the address of INTERCEPT_FLAG;
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !
1 !       SS$_RESIGNAL
1 !
1 ! SIDE EFFECTS:
1 !
1 !       May cause an unwind.
1 !
1 !--
1
2    BEGIN
2
2    BUILTIN
2       AP,     ! Argument pointer
2       CALLG;  ! CALLG instruction
2
2     !+
2     ! Determine if this is an unwind.  If not, then if INTERCEPT_FLAG
2     ! is set, turn this signal into an unwind.
2     !-
```

```
; 1277          1341  2
; 1278          1342  2      IF .SIGNAL_ARGS [CHF$L_SIG_NAME] NEQU SS$_UNWIND
; 1279          1343  2      THEN
; 1280          1344  2          IF ..ENABLE_ARGS [1]    ! Is INTERCEPT_FLAG set?
; 1281          1345  2          THEN
; 1282          1346  2              CALLG (.AP, LIB$SIG_TO_RET);       ! Convert signal to return status
; 1283          1347  2
; 1284          1348  2      RETURN SS$_RESIGNAL;                        ! Resignal error
; 1285          1349  2
; 1286          1350  1      END;                                        ! End of routine RENAME_HANDLER



                                            0000 00000 RENAME_HANDLER:
                                                              .WORD   Save nothing                            ; 1285
                            50       04   AC  D0 00002        MOVL    SIGNAL_ARGS, R0                         ; 1342
                  00000920  8F       04   A0  D1 00006        CMPL    4(R0), #2336                            ;
                                          0F  13 0000E        BEQL    1$                                      ;
                            50       0C   AC  D0 00010        MOVL    ENABLE_ARGS, R0                         ; 1344
                            07       04   B0  E9 00014        BLBC    @4(R0), 1$                              ;
                  00000000G 00            6C  FA 00018        CALLG   (AP), LIB$SIG_TO_RET                    ; 1346
                            50     0918   8F  3C 0001F 1$:    MOVZWL  #2328, R0                               ; 1348
                                              04 00024        RET                                             ; 1350

; Routine Size:  37 bytes,    Routine Base:  _LIB$CODE + 05CA


; 1287          1351  1
; 1289          1352  1 END                                      ! End of module LIB$RENAME_FILE
; 1290          1353  1
; 1291          1354  0 ELUDOM



                                                                 .EXTRN  LIB$STOP

;                             PSECT SUMMARY
;
;       Name                       Bytes                      Attributes
;
;   _LIB$CODE                          1519  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)



;                        Library Statistics
;
;                                                 -------- Symbols --------        Pages       Processing
;       File                                      Total    Loaded   Percent        Mapped      Time
;
;   _$255$DUA28:[SYSLIB]STARLET.L32;1              9776        79         0          581         00:00.7
;   _$255$DUA28:[LIBRTL.OBJ]RTLLIB.L32;1             36         2         5            8         00:00.1




;                              COMMAND QUALIFIERS

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

; Size:          1517 code + 2 data bytes
; Run Time:         00:24.1
; Elapsed Time:     01:30.3
; Lines/CPU Min:     3369
; Lexemes/CPU-Min: 44856
; Memory Used:  283 pages
; Compilation Complete
```

0209 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY