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        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
```

FILEID**LIBCREDIR

```
LL          IIIIII   BBBBBBBB    CCCCCCCC  RRRRRRRR    EEEEEEEEEE  DDDDDDDD     IIIIII   RRRRRRRR
LL          IIIIII   BBBBBBBB    CCCCCCCC  RRRRRRRR    EEEEEEEEEE  DDDDDDDD     IIIIII   RRRRRRRR
LL            II     BB      BB  CC        RR      RR  EE          DD      DD     II     RR      RR
LL            II     BB      BB  CC        RR      RR  EE          DD      DD     II     RR      RR
LL            II     BB      BB  CC        RR      RR  EE          DD      DD     II     RR      RR
LL            II     BB      BB  CC        RR      RR  EE          DD      DD     II     RR      RR
LL            II     BBBBBBBB    CC        RRRRRRRR    EEEEEEEE    DD      DD     II     RRRRRRRR
LL            II     BBBBBBBB    CC        RRRRRRRR    EEEEEEEE    DD      DD     II     RRRRRRRR
LL            II     BB      BB  CC        RR  RR      EE          DD      DD     II     RR  RR
LL            II     BB      BB  CC        RR  RR      EE          DD      DD     II     RR  RR
LL            II     BB      BB  CC        RR    RR    EE          DD      DD     II     RR    RR
LL            II     BB      BB  CC        RR    RR    EE          DD      DD     II     RR    RR        ....
LLLLLLLLLL  IIIIII   BBBBBBBB    CCCCCCCC  RR      RR  EEEEEEEEEE  DDDDDDDD     IIIIII   RR      RR      ....
LLLLLLLLLL  IIIIII   BBBBBBBB    CCCCCCCC  RR      RR  EEEEEEEEEE  DDDDDDDD     IIIIII   RR      RR      ....

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
```

LIB
V03

```
0 %TITLE 'LIB$CREATE_DIR - Create directory'
0 MODULE LIB$CREATE_DIR (                                  ! Create directory
0                 IDENT = 'V03-005'                        ! File: LIBCREDIR.B32 Edit: V03-001
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 ! FACILITY:     General Utility Library
1 !
1 ! ABSTRACT:
1 !
1 !       This routine creates a directory file.
1 !
1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
1 !
1 ! AUTHOR: Martin L. Jack, CREATION DATE: 23-Dec-1981
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-005 LMP0189         L. Mark Pilant,         6-Feb-1984  14:10
1 !               Rip out the ACL propagation done here.  It is now done by
1 !               the disk ACP.
1 !
1 !       V03-004 RAS0249         Ron Schaefer            4-Feb-1984
1 !               Fix this routine to cope with searchlists.  The directory
1 !               will be created in the first entry of the list just
1 !               like a file create.
1 !               We suppress the device assignment and directory lookups.
1 !               Thus we need to do a $GETDVI in order to determine that
1 !               the device is a disk.
1 !
1 !       V03-003 LMP0143         L. Mark Pilant,         24-Aug-1983  3:44
1 !               Propagate the directory default protection ACE to the
1 !               created directories.
```

```
:   58          0058  1 !
:   59          0059  1 !       V03-002 KBT0567         Keith B. Thompson       26-Jul-1983
:   60          0060  1 !               New RMS file naming features
:   61          0061  1 !
:   62          0062  1 !       V03-001 ACG0275         Andrew C. Goldstein,    26-Mar-1982  13:50
:   63          0063  1 !               Fix read references to top level directories
:   64          0064  1 !
:   65          0065  1 !--
```

```
1 %SBTTL 'Declarations'
1 !
1 ! SWITCHES:
1 !
1
1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
1
1 !
1 ! LINKAGES:
1 !
1 LINKAGE
1         LINKAGE_JSB_2_2 = JSB(REGISTER=0;REGISTER=1,REGISTER=2);
1 !
1 ! TABLE OF CONTENTS:
1 !
1
1 FORWARD ROUTINE
1     LIB$CREATE_DIR;                            ! Create directory
1
1 !
1 ! INCLUDE FILES:
1 !
1
1 LIBRARY 'SYS$LIBRARY:LIB';                     ! System symbols
1
1 !*! REQUIRE 'RTLIN:RTLPSECT';                  ! Define PSECT declarations macros
1
1 !
1 ! MACROS:
1 !
1         NONE
1 !
1 ! EQUATED SYMBOLS:
1 !
1         NONE
1 !
1 ! FIELDS:
1 !
1         NONE
1 !
1 ! PSECTS:
1 !
1 !*! DECLARE_PSECTS (LIB);                                    ! Declare PSECTs for LIB$ facility
1     PSECT
1             CODE = _LIB$CODE (READ, NOWRITE, EXECUTE, SHARE, PIC, ADDRESSING_MODE (WORD_RELATIVE)),
1             PLIT = _LIB$CODE (READ, NOWRITE, EXECUTE, SHARE, PIC, ADDRESSING_MODE (WORD_RELATIVE)),
1             OWN  = _LIB$DATA (READ, WRITE, NOEXECUTE, NOSHARE, PIC, ADDRESSING_MODE (LONG_RELATIVE)),
1             GLOBAL = _LIB$DATA (READ, WRITE, NOEXECUTE, NOSHARE, PIC, ADDRESSING_MODE (LONG_RELATIVE)) ;
1 !
1 ! OWN STORAGE:
1 !
1         NONE
1 !
1 ! EXTERNAL REFERENCES:
1 !
1
1 EXTERNAL ROUTINE
```

```
;  124         0123  1      LIB$ANALYZE_SDESC_R2:        LINKAGE_JSB_2_2,        ! Analyze descriptor
;  125         0124  1      LIB$CVT_OTB,                                   ! Convert octal to binary
;  126         0125  1      LIB$FREE_EF,                                   ! Deallocate an event flag
;  127         0126  1      LIB$GET_EF;                                    ! Allocate an event flag
;  128         0127  1
;  129         0128  1 EXTERNAL LITERAL                                    ! Completion status codes
;  130         0129  1      LIB$_INVARG,                                   ! Invalid argument
;  131         0130  1      LIB$_INVFILSPE;                                ! Invalid file specification
```

```
1 %SBTTL 'LIB$CREATE_DIR - Create directory'
1 GLOBAL ROUTINE LIB$CREATE_DIR (                     ! Create directory
1               DEV_DIR_SPEC,                         ! Device and directory string
1               OWNER_UIC,                            ! Owner UIC
1               PROT_ENABLE,                          ! File protection enables
1               PROT_VALUE,                           ! File protection value
1               MAX_VERSIONS,                         ! Maximum version count
1               RVN                                   ! Relative volume number
1       ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !             This routine creates directory files - it
1 !             creates one directory (spec) at a time.  The
1 !             directory may have up to 7 levels of sub-directories.
1 !
1 ! CALLING SEQUENCE:
1 !
1 !             ret_status.wlc.v = LIB$CREATE_DIR (dev-dir-spec.rt.dx,
1 !                     [owner-UIC.rlu.r], [prot-enable.rwu.r], [prot-value.rwu.r],
1 !                     [max-versions.rwu.r], [rvn.rwu.r])
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !             DEV_DIR_SPEC    Address of a descriptor for the device and directory
1 !                             specification.  This string is a standard RMS file
1 !                             specification; it must not contain a node name, file
1 !                             name, file type, file version, or wild card characters;
1 !                             it must contain an explicit directory; it must
1 !                             reference a disk device.  The string must be no longer
1 !                             than 255 characters.
1 !
1 !             OWNER_UIC       Address of a longword that specifies the owner UIC of
1 !                             the created directories.  If specified with a zero
1 !                             value, the owner UIC is that of the parent directory.
1 !                             This is an optional parameter.  The default is the
1 !                             current process UIC, except that if the directory is in
1 !                             UIC format, that UIC is the default.
1 !
1 !             PROT_ENABLE     Address of a word containing a mask to specify the bits
1 !                             of prot-value to be used.  Bits of the file protection
1 !                             corresponding to set bits in prot-enable are set to the
1 !                             value of the corresponding bit of the prot-value
1 !                             parameter; bits of the file protection corresponding to
1 !                             clear bits in prot-enable are set to the value of the
1 !                             corresponding bit of the parent directory's file
1 !                             protection with delete access dropped for all access
1 !                             categories.  This is an optional parameter.  The
1 !                             default is a mask of all zero bits, which results in
1 !                             propagating the parent directory's file protection.
1 !                             If prot-enable is all zero, prot-value is ignored.
1 !
1 !             PROT_VALUE      Address of a word containing a mask to specify the
1 !                             value of the file protection.  Bits of the file
1 !                             protection corresponding to set bits in prot-enable are
1 !                             set to the value of the corresponding bit of the
```

```
1 !                             prot-value parameter.  This is an optional parameter.
1 !                             The default is a mask of all zero bits, which specifies
1 !                             full access for all access categories.  In typical
1 !                             usage, prot-value is not omitted unless prot-enable is
1 !                             also omitted; in this case, prot-value is ignored.
1 !
1 !      MAX_VERSIONS           Address of a word that specifies the default maximum
1 !                             number of versions for files cataloged in the created
1 !                             directories.  This is an optional parameter.  The
1 !                             default is the parent directory's default version
1 !                             limit.  If specified as zero, the maximum number of
1 !                             versions is not limited.
1 !
1 !      RVN                    Address of a word that specifies the relative volume
1 !                             number within a volume set on which the directories
1 !                             must be created.  This is an optional parameter.  The
1 !                             default is arbitrary placement.
1 !
1 !      The format of PROT_ENABLE and PROT_VALUE is:
1 !
1 !                    1 1 1 1 1 1
1 !                    5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0
1 !                   +-------+-------+-------+-------+
1 !                   |World  |Group  |Owner  |System |
1 !                   |D E W R|D E W R|D E W R|D E W R|
1 !                   +-------+-------+-------+-------+
1 !
1 !      Set bits deny access and clear bits grant access.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !      NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !      NONE
1 !
1 ! COMPLETION STATUS:
1 !
1 !      SS$_NORMAL             Normal successful completion; all specified directories
1 !                             already exist
1 !
1 !      SS$_CREATED            Normal successful completion; one or more directories
1 !                             created
1 !
1 !      LIB$_INVARG            Required argument omitted, or dev-dir-spec longer than
1 !                             255 characters
1 !
1 !      LIB$_INVFILSPE         File specification did not contain an explicit
1 !                             directory or contained a node name, file name, file
1 !                             type, file version, or wildcard, or device not a disk
1 !
1 !      LIB$ANALYZE_SDESC errors
1 !      $PARSE errors
1 !      $ASSIGN errors
1 !      LIB$GET_EF errors
1 !      $QIO errors
```

```
1 !             $DASSGN errors
1 !             LIB$FREE_EF errors
1 !
1 ! SIDE EFFECTS:
1 !
1 !             Directory files created as requested.
1 !
1 ! NOTES:
1 !
1 !             LIB$CREATE_DIR does nothing with the ACL associated with a
1 !             directory.  This is the job of the file system (XQP).  There is a bit
1 !             defined in the FIB (FIB$x_DIRACL) that tells the file system (ACP)
1 !             that the file being created is going to be a directory file.  The
1 !             file system (ACP)
1 !             then copies the ACL from the parent directory (excluding ACEs
1 !             marked as NOPROPAGATE).  If the ACL of the sub-directory is
1 !             to be different than the parent directory, it is necessary to
1 !             alter it after the directory is created.
1 !
1 !--
1
2 BEGIN
2 LOCAL
2               FAB:            $FAB_DECL,        ! FAB for $PARSE
2               NAM:            $NAM_DECL,        ! NAM block for $PARSE
2               ESA_BUFFER:     VECTOR[NAM$C_MAXRSS,BYTE], ! Expanded string area
2               TEMP_DESC:      BLOCK[DSC$K_S_BLN,BYTE], ! Utility descriptor
2               NAME_BUFFER:    VECTOR[15,BYTE], ! Directory name buffer
2               NAME_DESC:      BLOCK[DSC$K_Z_BLN,BYTE], ! Directory name descriptor
2               DEV_ATTR:       BLOCK[4,BYTE],   ! Device attributes from $GETDVI
2               DVI_ITMLST:     BLOCK[4,LONG]    ! One itemlist entry
2                               INITIAL(DVI$_DEVCHAR*65536+4,DEV_ATTR,0,0),
2 RECATTR:              BLOCK[ATR$S_RECATTR,BYTE]
2                               VOLATILE,        ! Record attributes
2               UCHAR:          BLOCK[ATR$S_UCHAR,BYTE]
2                               VOLATILE,        ! File characteristics
2               FPRO:           BLOCK[ATR$S_FPRO,BYTE]
2                               VOLATILE,        ! File protection
2               UIC:            BLOCK[ATR$S_UIC,BYTE]
2                               VOLATILE,        ! File owner UIC
2               HEADER:         BLOCK[ATR$S_HEADER,BYTE]
2                               VOLATILE,        ! File header
2               FIB:            BLOCK[FIB$C_LENGTH,BYTE], ! FIB
2               FIB_DESC:       VECTOR[2],       ! Descriptor for FIB
2               ATR:            BLOCKVECTOR[7,8,BYTE], ! Attribute descriptors
2               IOSB:           VECTOR[4,WORD],  ! I/O status block
2               CHANNEL:        WORD,            ! Channel number
2               EFN,                             ! Event flag number
2               GROUP,                           ! Binary group number
2               MEMBER,                          ! Binary member number
2               LOCAL_ENABLE:   WORD,            ! Value of PROT_ENABLE after defaulting
2               LOCAL_VALUE:    WORD,            ! Value of PROT_VALUE after defaulting
2               DIR_LENGTH,                      ! Length of residual directory string
2               DIR_ADDRESS,                     ! Address of residual directory string
2               REALDIR_ADDRESS,                 ! Real (not root) directory pointer
2               STATUS_1,                        ! Status return
2               STATUS_2,                        ! Status return
```

```
;  304         0302  2          STATUS_4,                       ! Status return
;  305         0303  2          STATUS_5,                       ! Status return
;  306         0304  2          STATUS_6,                       ! Status return
;  307         0305  2          STATUS_7,                       ! Status return
;  308         0306  2          FINAL_STATUS;                   ! Status return
;  309         0307  2
;  310         0308  2  BIND
;  311         0309  2          DIR_TYP_VER = UPLIT BYTE ('.DIR;1');
;  312         0310  2                                          ! File type and version string
;  313         0311  2  LABEL
;  314         0312  2          PROCESS;                        ! Block exited when processing complete
;  315         0313  2  BUILTIN
;  316         0314  2          ACTUALCOUNT,                    ! Return number of arguments
;  317         0315  2          LOCC,                           ! LOCC instruction
;  318         0316  2          NULLPARAMETER,                  ! Test if parameter specified
;  319         0317  2          ROT;                            ! Rotate longword
;  320         0318  2
;  321         0319  2  ! Ensure that the required parameter is present.
;  322         0320  2  !
;  323         0321  2  IF ACTUALCOUNT() EQL 0
;  324         0322  2  THEN
;  325         0323  2      RETURN LIB$_INVARG;
;  326         0324  2
;  327         0325  2  ! Initialize RMS structures required to do a $PARSE.
;  328         0326  2  !
;  329       P 0327  2  $FAB_INIT( FAB = FAB,
;  330         0328  2             NAM = NAM );
;  331       P 0329  2  $NAM_INIT( NAM = NAM,
;  332       P 0330  2             NOP = <SYNCHK,NOCONCEAL>,
;  333       P 0331  2             ESA = ESA_BUFFER,
;  334         0332  2             ESS = NAM$C_MAXRSS );
;  335         0333  2
;  336         0334  2  ! Analyze the input descriptor and set up the FAB filename descriptor.
;  337         0335  2  !
;  338         0336  3  BEGIN ! block to use output registers
;  339         0337  3  REGISTER
;  340         0338  3          R1 = 1,
;  341         0339  3          R2 = 2;
;  342         0340  3
;  343         0341  3  STATUS_1 = LIB$ANALYZE_SDESC_R2(.DEV_DIR_SPEC; R1, R2);
;  344         0342  3  IF NOT .STATUS_1
;  345         0343  3  THEN
;  346         0344  3      RETURN .STATUS_1;
;  347         0345  3
;  348         0346  3  IF .R1 GTRU 255
;  349         0347  3  THEN
;  350         0348  3      RETURN LIB$_INVARG;
;  351         0349  3
;  352         0350  3  FAB [ FAB$B_FNS ] = .R1;
;  353         0351  3  FAB [ FAB$L_FNA ] = .R2
;  354         0352  2  END; ! block to use output registers
;  355         0353  2
;  356         0354  2  !+
;  357         0355  2  ! Parse the file specification to obtain the expanded name string.  RMS will
;  358         0356  2  ! usually return RMS$_DNF (directory not found), but all that is needed is
;  359         0357  2  ! the expanded string.
;  360         0358  2  !-
```

```
2
2 STATUS_2 = $PARSE(FAB=FAB);
2 IF NOT .STATUS_2
2 THEN
2     RETURN STATUS_2;
2
2 ! Perform various error checks on the file specification.  It must not have
2 ! a node name, file name, file type, or file version; it must have a directory
2 ! name that does not contain wildcards.
2 !
2 IF  NOT .NAM[NAM$V_EXP_DIR] OR
3     (.NAM[NAM$L_FNB] AND
4         (NAM$M_WILDCARD OR
3          NAM$M_NODE OR NAM$M_EXP_NAME OR NAM$M_EXP_TYPE OR NAM$M_EXP_VER))
2         NEQ 0
2 THEN
2     RETURN LIB$_INVFILSPE;
2
2 ! Get the length and address of the directory string without brackets.
2 !
2 DIR_LENGTH = .NAM [ NAM$B_DIR ] - 2;             ! Length without brackets
2 DIR_ADDRESS = .NAM [ NAM$L_DIR ] + 1;            ! Address without bracket
2
2 ! If there is a root directory locate the real directory and squish
2 ! them together
2 !
2 IF .NAM [ NAM$V_ROOT_DIR ]
2 THEN
3     BEGIN
3
3     REGISTER
3         R0 = 0,
3         R1 = 1;
3
3     LOCAL
3         TERMINATOR:     BYTE;
3
3     TERMINATOR = ..NAM [ NAM$L_DIR ] + 2;               ! Close = Open+2
3
3     IF NOT LOCC( TERMINATOR, DIR_LENGTH, .DIR_ADDRESS; R0, R1 )
3     THEN
3         RETURN LIB$_INVFILSPE;                 ! No Root Found
3
3     REALDIR_ADDRESS = .R1;
3
3     ! Found the terminator of the root directory, the real directory
3     ! will be 2 past it,  i.e.  "[ROOT.][DIR]"
3     !                   REALDIR_ADDRESS-^
3     !
3     ! Move what is left over 2 in order to get rid of the "][" to get:
3     !                             "[ROOT.DIR]"
3     !                   REALDIR_ADDRESS-^
3     !
3     CH$MOVE( .R0, .REALDIR_ADDRESS+2, .REALDIR_ADDRESS );
3
3     ! Adjust the direcory length
3     !
```

```
3     DIR_LENGTH = .DIR_LENGTH - 2
3
3     END
2 ELSE
2
2     ! If no root then the real directory is the uic directory
2     !
2     REALDIR_ADDRESS = .DIR_ADDRESS;
2
2 ! If the directory is in UIC format, convert it to normal format.
2 !
2 IF .NAM [ NAM$V_GRP_MBR ]
2 THEN
3     BEGIN
3
3     ! Convert the group part
3     !
3     IF NOT LIB$CVT_OTB( 3, .REALDIR_ADDRESS, GROUP)
3     THEN
3         RETURN LIB$_INVFILSPE;                  ! Invalid group number
3
3     ! Convert the member part
3     !
3     IF NOT LIB$CVT_OTB( 3, .REALDIR_ADDRESS + 3, MEMBER)
3     THEN
3         RETURN LIB$_INVFILSPE                   ! Invalid member number
3
2     END;
2
2 TEMP_DESC [ DSC$B_CLASS ] = DSC$K_CLASS_S;
2 TEMP_DESC [ DSC$B_DTYPE ] = DSC$K_DTYPE_T;
2 TEMP_DESC [ DSC$W_LENGTH ] = .NAM [ NAM$B_DEV];
2 TEMP_DESC [ DSC$A_POINTER ] = .NAM [ NAM$L_DEV ];
2
2 ! Set up the FIB to look up the MFD.
2 !
2 CH$FILL(0, FIB$C_LENGTH, FIB);
2 FIB[FIB$L_ACCTL] = FIB$M_WRITE OR FIB$M_NOREAD OR FIB$M_NOWRITE;
2 FIB[FIB$W_FID_NUM] = FID$C_MFD;
2 FIB[FIB$W_FID_SEQ] = FID$C_MFD;
2
2 ! Set up the FIB descriptor.
2 !
2 FIB_DESC[0] = FIB$C_LENGTH;
2 FIB_DESC[1] = FIB;
2
2 ! Set up the name descriptor.  The length is filled in as need be.
2 !
2 NAME_DESC[DSC$A_POINTER] = NAME_BUFFER;
2
2 !+
2 ! Assign a channel to the device.  The descriptor is already set up from
2 ! a preceding operation.
2 !-
2
2 STATUS_4 = $ASSIGN(DEVNAM=TEMP_DESC, CHAN=CHANNEL);
2 IF NOT .STATUS_4 THEN RETURN .STATUS_4;
```

```
2
2 !+
2 ! Get the device characteristics and make sure this is a disk-device
2 ! and not mounted foreign.
2 !-
2
2 STATUS_4 = $GETDVIW(CHAN=.CHANNEL,ITMLST=DVI_ITMLST);
2 IF NOT .STATUS_4 THEN RETURN .STATUS_4;
2 IF NOT .DEV_ATTR[DEV$V_RND] OR .DEV_ATTR[DEV$V_FOR]
2 THEN
3     BEGIN
3     $DASSGN(CHAN=.CHANNEL);
3     RETURN LIB$_INVFILSPE;
2     END;
2
2 !+
2 ! Allocate an event flag.
2 !-
2
2 STATUS_5 = LIB$GET_EF(EFN);
2 IF NOT .STATUS_5
2 THEN
3     BEGIN
3     $DASSGN(CHAN=.CHANNEL);
3     RETURN .STATUS_5;
2     END;
2
2 !+
2 ! Beginning of block that is exited when processing is complete.  FINAL_STATUS
2 ! contains the status to be returned to caller.
2 !-
2
3 PROCESS:  BEGIN
3
3 !+
3 ! Loop to look up directories.
3 !-
3
3 WHILE 1 DO
4     BEGIN
4
4     !+
4     ! Copy the file ID to FIB$W_DID so that the next lookup is done in that
4     ! directory.
4     !-
4
4     FIB[FIB$W_DID_NUM] = .FIB[FIB$W_FID_NUM];
4     FIB[FIB$W_DID_SEQ] = .FIB[FIB$W_FID_SEQ];
4     FIB[FIB$W_DID_RVN] = .FIB[FIB$W_FID_RVN];
4
4     !+
4     ! Locate the next directory name.
4     !-
4
5     BEGIN ! block to use output registers
5     REGISTER
5         R0 = 0,
;  532          0530  5           R1 = 1;
;  533          0531  5       LOCAL
;  534          0532  5           NAME_LENGTH,
;  535          0533  5           NAME_ADDRESS;
;  536          0534  5
;  537          0535  5       NAME_ADDRESS = .DIR_ADDRESS;                  ! Save starting point
;  538          0536  5       LOCC(%REF(%C'.'), DIR_LENGTH, .DIR_ADDRESS; R0, R1);
;  539          0537  5       NAME_LENGTH = .DIR_LENGTH - .R0;              ! Length preceding dot or end
;  540          0538  5       DIR_ADDRESS = .R1 + 1;                        ! Prune to string following dot
;  541          0539  5       R0 = .R0 - 1;
;  542          0540  5       DIR_LENGTH = .R0;
;  543          0541  5
;  544          0542  5       !+
;  545          0543  5       ! Construct the directory name concatenated with '.DIR;1' in the name
;  546          0544  5       ! buffer, and a descriptor for this name in the name descriptor.
;  547          0545  5       !-
;  548          0546  5
;  549          0547  5       NAME_DESC[DSC$W_LENGTH] = .NAME_LENGTH + 6;
;  550          0548  5       CH$MOVE(6, DIR_TYP_VER, CH$MOVE(.NAME_LENGTH, .NAME_ADDRESS, NAME_BUFFER));
;  551          0549  4       END; ! block to use output registers
;  552          0550  4
;  553          0551  4       !+
;  554          0552  4       ! Look up the directory at the current level.  If the directory
;  555          0553  4       ! does not exist, exit the loop to begin creating directories.
;  556          0554  4       !-
;  557          0555  4
;  558        P 0556  4       FINAL_STATUS = $QIOW(
;  559        P 0557  4           FUNC=IO$_ACCESS,
;  560        P 0558  4           CHAN=.CHANNEL,
;  561        P 0559  4           EFN=.EFN,
;  562        P 0560  4           IOSB=IOSB,
;  563        P 0561  4           P1=FIB_DESC,
;  564          0562  4           P2=NAME_DESC);
;  565          0563  4       IF .FINAL_STATUS THEN FINAL_STATUS = .IOSB[0];
;  566          0564  4       IF .FINAL_STATUS EQL SS$_NOSUCHFILE THEN EXITLOOP;
;  567          0565  4       IF NOT .FINAL_STATUS THEN LEAVE PROCESS;
;  568          0566  4
;  569          0567  4       !+
;  570          0568  4       ! If no more directory levels were specified, all specified directories
;  571          0569  4       ! already exist, so return with success.
;  572          0570  4       !-
;  573          0571  4
;  574          0572  4       IF .DIR_LENGTH LEQ 0
;  575          0573  4       THEN
;  576          0574  5           BEGIN
;  577          0575  5           FINAL_STATUS = SS$_NORMAL;
;  578          0576  5           LEAVE PROCESS;
;  579          0577  4           END;
;  580          0578  3       END;
;  581          0579  3
;  582          0580  3 !+
;  583          0581  3 ! We have reached the level at which directories do not yet exist.  FIB$W_DID
;  584          0582  3 ! now contains the file ID of the directory in which the new directory must be
;  585          0583  3 ! cataloged and the filename descriptor contains the name of the new directory.
;  586          0584  3 !-
;  587          0585  3
;  588          0586  3 !+
;  589          0587  3 ! Set up the attribute list.  Because of dependencies later in the routine,
;  590          0588  3 ! the file header attribute must be last, preceded by the owner UIC attribute.
;  591          0589  3 !-
;  592          0590  3
;  593          0591  3 ATR[0, ATR$W_TYPE] = ATR$C_RECATTR;     ! Record attributes
;  594          0592  3 ATR[0, ATR$W_SIZE] = ATR$S_RECATTR;
;  595          0593  3 ATR[0, ATR$L_ADDR] = RECATTR;
;  596          0594  3 ATR[1, ATR$W_TYPE] = ATR$C_UCHAR;       ! File characteristics
;  597          0595  3 ATR[1, ATR$W_SIZE] = ATR$S_UCHAR;
;  598          0596  3 ATR[1, ATR$L_ADDR] = UCHAR;
;  599          0597  3 ATR[2, ATR$W_TYPE] = ATR$C_FPRO;        ! File protection
;  600          0598  3 ATR[2, ATR$W_SIZE] = ATR$S_FPRO;
;  601          0599  3 ATR[2, ATR$L_ADDR] = FPRO;
;  602          0600  3 ATR[3, ATR$W_TYPE] = ATR$C_UIC;         ! File owner UIC
;  603          0601  3 ATR[3, ATR$W_SIZE] = ATR$S_UIC;
;  604          0602  3 ATR[3, ATR$L_ADDR] = UIC;
;  605          0603  3 ATR[4, ATR$W_TYPE] = ATR$C_HEADER;      ! File header
;  606          0604  3 ATR[4, ATR$W_SIZE] = ATR$S_HEADER;
;  607          0605  3 ATR[4, ATR$L_ADDR] = HEADER;
;  608          0606  3 ATR[5, 0,0,32,0] = 0;                   ! End of list
;  609          0607  3
;  610          0608  3
;  611          0609  3 !+
;  612          0610  3 ! Copy the file ID back to FIB$W_FID to do the read attributes on the
;  613          0611  3 ! last directory file.
;  614          0612  3 !-
;  615          0613  3
;  616          0614  3 FIB[FIB$W_FID_NUM] = .FIB[FIB$W_DID_NUM];
;  617          0615  3 FIB[FIB$W_FID_SEQ] = .FIB[FIB$W_DID_SEQ];
;  618          0616  3 FIB[FIB$W_FID_RVN] = .FIB[FIB$W_DID_RVN];
;  619          0617  3 FIB[FIB$W_DID_NUM] = 0;
;  620          0618  3 FIB[FIB$W_DID_SEQ] = 0;
;  621          0619  3 FIB[FIB$W_DID_RVN] = 0;
;  622          0620  3
;  623          0621  3 !+
;  624          0622  3 ! Read the attributes of the last directory file found so that they
;  625          0623  3 ! may be propagated to the directories created.
;  626          0624  3 !-
;  627          0625  3
;  628        P 0626  3 FINAL_STATUS = $QIOW(
;  629        P 0627  3     FUNC=IO$_ACCESS,
;  630        P 0628  3     CHAN=.CHANNEL,
;  631        P 0629  3     EFN=.EFN,
;  632        P 0630  3     IOSB=IOSB,
;  633        P 0631  3     P1=FIB_DESC,
;  634          0632  3     P5=ATR);
;  635          0633  3 IF .FINAL_STATUS THEN FINAL_STATUS = .IOSB[0];
;  636          0634  3 IF NOT .FINAL_STATUS THEN LEAVE PROCESS;
;  637          0635  3
;  638          0636  3
;  639          0637  3 !+
;  640          0638  3 ! Delete the file header attribute from the attribute list, since it is not
;  641          0639  3 ! valid (or necessary) for creates.
;  642          0640  3 !-
;  643          0641  3
;  644          0642  3 ATR[4, 0,0,32,0] = 0;
;  645          0643  3
```

```
3 !+
3 ! Copy the file ID to FIB$W_DID to create the directory.
3 !-
3
3 FIB[FIB$W_DID_NUM] = .FIB[FIB$W_FID_NUM];
3 FIB[FIB$W_DID_SEQ] = .FIB[FIB$W_FID_SEQ];
3 FIB[FIB$W_DID_RVN] = .FIB[FIB$W_FID_RVN];
3
3 !+
3 ! Establish the allocation of the created directories.  A Structure Level 1
3 ! directory is allocated zero blocks; a Structure Level 2 directory is
3 ! allocated one block.  (This block is later initialized.)  In both cases,
3 ! the file is marked contiguous.
3 !-
3
3 FIB[FIB$W_EXCTL] = FIB$M_EXTEND OR FIB$M_FILCON OR FIB$M_ALCON;
3 IF .HEADER[FH2$B_STRUCLEV] EQL 2 THEN FIB[FIB$L_EXSZ] = 1;
3
3 !+
3 ! Set up the end of file pointer.  It points to the highest allocated block
3 ! plus one (with a first free byte of zero).  Note that EFBLK is stored in
3 ! inverted format.
3 !-
3
3 RECATTR[FAT$L_EFBLK] = ROT(.FIB[FIB$L_EXSZ] + 1, 16);
3
3 !+
3 ! Establish the owner UIC of the created directories.  If the process default
3 ! UIC is to be used, delete the owner UIC attribute from the attribute list
3 ! to cause the ACP to use the default.
3 !-
3
3 IF NOT NULLPARAMETER(2)
3 THEN
4     BEGIN
4     IF ..OWNER_UIC NEQ 0 THEN UIC = ..OWNER_UIC;
4     END
3 ELSE
3     IF .NAM[NAM$V_GRP_MBR]
3     THEN
4         BEGIN
4         UIC<16,16> = .GROUP;
4         UIC<0,16> = .MEMBER;
4         END
3     ELSE
3         ATR[3, 0,0,32,0] = 0;
3
3 !+
3 ! Establish the file protection of the created directories.
3 !-
3
3 FPRO = .FPRO OR %X'8888';
3 LOCAL_ENABLE = 0;
3 IF NOT NULLPARAMETER(3)
3 THEN
3     LOCAL_ENABLE = .(.PROT_ENABLE)<0,16>;
3
```

```
  3 LOCAL_VALUE = 0;
  3 IF NOT NULLPARAMETER(4)
  3 THEN
  3     LOCAL_VALUE = .(.PROT_VALUE)<0,16>;
  3
  3 FPRO = (.FPRO AND NOT .LOCAL_ENABLE) OR (.LOCAL_VALUE AND .LOCAL_ENABLE);
  3
  3 !+
  3 ! Establish the default version limit of the created directories.
  3 !-
  3
  3 IF NOT NULLPARAMETER(5)
  3 THEN
  3     RECATTR[FAT$W_VERSIONS] = .(.MAX_VERSIONS)<0,16>;
  3
  3 !+
  3 ! Establish the placement of the created directories.  Note that if placement
  3 ! is specified, it is required.
  3 !-
  3
  3 IF NOT NULLPARAMETER(6)
  3 THEN
  4     BEGIN
  4     FIB[FIB$V_EXACT] = 1;                       ! Exact placement
  4     FIB[FIB$B_ALALIGN] = FIB$C_LBN;             ! RVN and LBN placement
  4     FIB[FIB$W_LOC_RVN] = .(.RVN)<0,16>;         ! Required RVN
  3     END;
  3
  3 !+
  3 ! Note that the ACL should be copied from parent to child.
  3 !-
  3
  3 FIB[FIB$V_DIRACL] = 1;
  3
  3 !+
  3 ! Loop to create directories.
  3 !-
  3
  3 WHILE 1 DO
  4     BEGIN
  4
  4     !+
  4     ! Create and access the file.
  4     !-
  4
P 4     FINAL_STATUS = $QIOW(
P 4         FUNC=IO$_CREATE OR IO$M_CREATE OR IO$M_ACCESS,
P 4         CHAN=.CHANNEL,
P 4         EFN=.EFN,
P 4         IOSB=IOSB,
P 4         P1=FIB_DESC,
P 4         P2=NAME_DESC,
  4         P5=ATR);
  4     IF .FINAL_STATUS THEN FINAL_STATUS = .IOSB[0];
  4     IF NOT .FINAL_STATUS THEN LEAVE PROCESS;
  4
  4     !+
:  760           0758  4      ! If the directory is Structure Level 2, write the first block of the
:  761           0759  4      ! file.
:  762           0760  4      !-
:  763           0761  4
:  764           0762  4      IF .HEADER[FH2$B_STRUCLEV] EQL 2
:  765           0763  4      THEN
:  766           0764  5          BEGIN
:  767           0765  5          LOCAL
:  768           0766  5              BLOCK_BUFFER:       VECTOR[256,WORD];       ! Block buffer
:  769           0767  5
:  770           0768  5          BLOCK_BUFFER[0] = -1;                       ! End of block marker
:  771           0769  5          CH$FILL(0, 510, BLOCK_BUFFER[1]);           ! Fill rest of block
:  772         P 0770  5          FINAL_STATUS = $QIOW(
:  773         P 0771  5              FUNC=IO$_WRITEVBLK,
:  774         P 0772  5              CHAN=.CHANNEL,
:  775         P 0773  5              EFN=.EFN,
:  776         P 0774  5              IOSB=IOSB,
:  777         P 0775  5              P1=BLOCK_BUFFER,
:  778         P 0776  5              P2=512,
:  779           0777  5              P3=1);
:  780           0778  5          IF .FINAL_STATUS THEN FINAL_STATUS = .IOSB[0];
:  781           0779  5          IF NOT .FINAL_STATUS THEN LEAVE PROCESS;
:  782           0780  4          END;
:  783           0781  4
:  784           0782  4      !+
:  785           0783  4      ! Deaccess the file.
:  786           0784  4      !-
:  787           0785  4
:  788         P 0786  4      FINAL_STATUS = $QIOW(
:  789         P 0787  4          FUNC=IO$_DEACCESS,
:  790         P 0788  4          CHAN=.CHANNEL,
:  791         P 0789  4          EFN=.EFN,
:  792           0790  4          IOSB=IOSB);
:  793           0791  4      IF .FINAL_STATUS THEN FINAL_STATUS = .IOSB[0];
:  794           0792  4      IF NOT .FINAL_STATUS THEN LEAVE PROCESS;
:  795           0793  4
:  796           0794  4      !+
:  797           0795  4      ! If no more directory levels were specified, they have all been created.
:  798           0796  4      !-
:  799           0797  4
:  800           0798  4      IF .DIR_LENGTH LEQ 0
:  801           0799  4      THEN
:  802           0800  5          BEGIN
:  803           0801  5          FINAL_STATUS = SS$_CREATED;
:  804           0802  5          LEAVE PROCESS;
:  805           0803  4          END;
:  806           0804  4
:  807           0805  4      !+
:  808           0806  4      ! Locate the next directory name.
:  809           0807  4      !-
:  810           0808  4
:  811           0809  5      BEGIN ! block to use output registers
:  812           0810  5      REGISTER
:  813           0811  5          R0 = 0,
:  814           0812  5          R1 = 1;
:  815           0813  5      LOCAL
:  816           0814  5          NAME_LENGTH,
;  817        0815  5           NAME_ADDRESS;
;  818        0816  5
;  819        0817  5       NAME_ADDRESS = .DIR_ADDRESS;                 ! Save starting point
;  820        0818  5       LOCC(%REF(%C'.'), DIR_LENGTH, .DIR_ADDRESS; R0, R1);
;  821        0819  5       NAME_LENGTH = .DIR_LENGTH - .R0;             ! Length preceding dot or end
;  822        0820  5       DIR_ADDRESS = .R1 + 1;                       ! Prune to string following dot
;  823        0821  5       R0 = .R0 - 1;
;  824        0822  5       DIR_LENGTH = .R0;
;  825        0823  5
;  826        0824  5       !+
;  827        0825  5       ! Construct the directory name concatenated with '.DIR;1' in the name
;  828        0826  5       ! buffer, and a descriptor for this name in the name descriptor.
;  829        0827  5       !-
;  830        0828  5
;  831        0829  5       NAME_DESC[DSC$W_LENGTH] = .NAME_LENGTH + 6;
;  832        0830  5       CH$MOVE(6, DIR_TYP_VER, CH$MOVE(.NAME_LENGTH, .NAME_ADDRESS, NAME_BUFFER));
;  833        0831  4       END; ! block to use output registers
;  834        0832  4
;  835        0833  4       !+
;  836        0834  4       ! Copy the file ID of the created directory to FIB$W_DID so that the next
;  837        0835  4       ! directory is cataloged in the directory just created.
;  838        0836  4       !-
;  839        0837  4
;  840        0838  4       FIB[FIB$W_DID_NUM] = .FIB[FIB$W_FID_NUM];
;  841        0839  4       FIB[FIB$W_DID_SEQ] = .FIB[FIB$W_FID_SEQ];
;  842        0840  4       FIB[FIB$W_DID_RVN] = .FIB[FIB$W_FID_RVN];
;  843        0841  3       END;
;  844        0842  3
;  845        0843  3   !+
;  846        0844  3   ! End of block that is exited when processing is complete.  FINAL_STATUS
;  847        0845  3   ! contains the status that is to be returned to caller.
;  848        0846  3   !-
;  849        0847  3
;  850        0848  2   END; ! of block PROCESS
;  851        0849  2
;  852        0850  2   !+
;  853        0851  2   ! Deassign the channel and deallocate the event flag.
;  854        0852  2   !-
;  855        0853  2
;  856        0854  2   STATUS_6 = $DASSGN(CHAN=.CHANNEL);
;  857        0855  2   STATUS_7 = LIB$FREE_EF(EFN);
;  858        0856  2   IF NOT .STATUS_7 THEN RETURN .STATUS_7;
;  859        0857  2   IF NOT .STATUS_6 THEN RETURN .STATUS_6;
;  860        0858  2
;  861        0859  2   !+
;  862        0860  2   ! Return the status.
;  863        0861  2   !-
;  864        0862  2
;  865        0863  2   RETURN .FINAL_STATUS;
;  866        0864  1   END;                                     ! End of routine LIB$CREATE_DIR

                                                                  .TITLE  LIB$CREATE_DIR LIB$CREATE_DIR - Create director
;                                                                                        y
                                                                  .IDENT  \V03-005\

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

```
                                              00020004  00000 P.AAA:  .LONG   131076                                        ;
                            00000000 00000000 00000000  00004         .LONG   0, 0, 0                                       ;
                                  31 3B  52  49 44  2E  00010 P.AAB:  .ASCII  \.DIR;1\                                      ;

                                                              DIR_TYP_VER=           P.AAB
                                                                      .EXTRN  LIB$ANALYZE_SDESC_R2
                                                                      .EXTRN  LIB$CVT_OTB, LIB$FREE_EF
                                                                      .EXTRN  LIB$GET_EF, LIB$_INVARG
                                                                      .EXTRN  LIB$_INVFILSPE, SYS$PARSE
                                                                      .EXTRN  SYS$ASSIGN, SYS$GETDVIW
                                                                      .EXTRN  SYS$DASSGN, SYS$QIOW

                                                   0FFC 00000         .ENTRY  LIB$CREATE_DIR, Save R2,R3,R4,R5,R6,R7,R8,-   ; 0132
                                                                              R9,R10,R11                                    ;
                                    5B 00000000G 00  9E 00002         MOVAB   SYS$QIOW, R11                                 ;
                                    5E      F954 CE  9E 00009         MOVAB   -1708(SP), SP                                 ;
                 FE20  CD       D8  AF           10  28 0000E         MOVC3   #16, P.AAA, DVI_ITMLST                        ; 0276
                              FE24  CD           6E  9E 00015         MOVAB   DEV_ATTR, DVI_ITMLST+4                        ; 0266
                                                 6C  95 0001A         TSTB    (AP)                                          ; 0321
                                                 56  13 0001C         BEQL    2$                                            ;
    0050  8F           00           6E           00  2C 0001E         MOVC5   #0, (SP), #0, #80, $RMS_PTR                   ; 0328
                                              B0 AD     00025                                                               ;
                                B0  AD      5003 8F  B0 00027         MOVW    #20483, $RMS_PTR                              ;
                                C6  AD           02  90 0002D         MOVB    #2, $RMS_PTR+22                               ;
                                CF  AD           02  90 00031         MOVB    #2, $RMS_PTR+31                               ;
                                D8  AD      FF50 CD  9E 00035         MOVAB   NAM, $RMS_PTR+40                              ;
    0060  8F           00           6E           00  2C 0003B         MOVC5   #0, (SP), #0, #96, $RMS_PTR                   ; 0332
                                            FF50 CD     00042                                                               ;
                              FF50  CD      6002 8F  B0 00045         MOVW    #24578, $RMS_PTR                              ;
                              FF58  CD           18  90 0004C         MOVB    #24, $RMS_PTR+8                               ;
                              FF5A  CD           01  8E 00051         MNEGB   #1, $RMS_PTR+10                               ;
                              FF5C  CD      FE50 CD  9E 00056         MOVAB   ESA_BUFFER, $RMS_PTR+12                       ;
                                    50        04 AC  D0 0005D         MOVL    DEV_DIR_SPEC, R0                              ; 0341
                                       00000000G 00  16 00061         JSB     LIB$ANALYZE_SDESC_R2                          ;
                                    01           50  E8 00067         BLBS    STATUS_1, 1$                                  ; 0342
                                                     04 0006A         RET                                                   ;
                          000000FF  8F           51  D1 0006B 1$:     CMPL    R1, #255                                      ; 0346
                                                 08  1B 00072         BLEQU   3$                                            ;
                                    50 00000000G 8F  D0 00074 2$:     MOVL    #LIB$_INVARG, R0                              ; 0348
                                                     04 0007B         RET                                                   ;
                                E4  AD           51  90 0007C 3$:     MOVB    R1, FAB+52                                    ; 0350
                                DC  AD           52  D0 00080         MOVL    R2, FAB+44                                    ; 0351
                                              B0 AD  9F 00084         PUSHAB  FAB                                           ; 0360
                         00000000G  00           01  FB 00087         CALLS   #1, SYS$PARSE                                 ;
                                04  AE           50  D0 0008E         MOVL    R0, STATUS_2                                  ;
                                    05        04 AE  E8 00092         BLBS    STATUS_2, 4$                                  ; 0361
                                    50        04 AE  9E 00096         MOVAB   STATUS_2, R0                                  ; 0363
                                                     04 0009A         RET                                                   ;
                       03       84  AD           06  E0 0009B 4$:     BBS     #6, NAM+52, 6$                                ; 0369
                                               00E1  31 000A0 5$:     BRW     13$                                           ;
                          00020107  8F        84 AD  D3 000A3 6$:     BITL    NAM+52, #131335                               ; 0373
                                                 F3  12 000AB         BNEQ    5$                                            ;
                                    59        8A AD  9A 000AD         MOVZBL  NAM+58, DIR_LENGTH                            ; 0379
                                    59           02  C2 000B1         SUBL2   #2, DIR_LENGTH                                ;
                       5A       98  AD           01  C1 000B4         ADDL3   #1, NAM+72, DIR_ADDRESS                       ; 0380
                       18       85  AD           05  E1 000B9         BBC     #5, NAM+53, 7$                                ; 0385
```

```
                    50        98   BD              02  81 000BE           ADDB3   #2, @NAM+72, TERMINATOR                         ; 0396
                    6A             59              50  3A 000C3           LOCC    TERMINATOR, DIR_LENGTH, (DIR_ADDRESS)           ; 0398
                                                   D7  13 000C7           BEQL    5$                                              ;
                                   56              51  D0 000C9           MOVL    R1, REALDIR_ADDRESS                             ; 0402
                    66        02   A6              50  28 000CC           MOVC3   R0, 2(REALDIR_ADDRESS), (REALDIR_ADDRESS)       ; 0412
                                   59              02  C2 000D1           SUBL2   #2, DIR_LENGTH                                  ; 0416
                                                   03  11 000D4           BRB     8$                                              ;
                                   56              5A  D0 000D6 7$:       MOVL    DIR_ADDRESS, REALDIR_ADDRESS                    ; 0423
                    23        86   AD              03  E1 000D9 8$:       BBC     #3, NAM+54, 9$                                  ; 0427
                                             08    AE  9F 000DE           PUSHAB  GROUP                                           ; 0433
                                                   56  DD 000E1           PUSHL   REALDIR_ADDRESS                                 ;
                                                   03  DD 000E3           PUSHL   #3                                              ;
                       00000000G   00              03  FB 000E5           CALLS   #3, LIB$CVT_OTB                                 ;
                                   B1              50  E9 000EC           BLBC    R0, 5$                                          ;
                                             0C    AE  9F 000EF           PUSHAB  MEMBER                                          ; 0439
                                             03    A6  9F 000F2           PUSHAB  3(REALDIR_ADDRESS)                              ;
                                                   03  DD 000F5           PUSHL   #3                                              ;
                       00000000G   00              03  FB 000F7           CALLS   #3, LIB$CVT_OTB                                 ;
                                   9F              50  E9 000FE           BLBC    R0, 5$                                          ;
                            FE4A   CD      010E    8F  B0 00101 9$:       MOVW    #270, TEMP_DESC+2                               ; 0446
                            FE48   CD        89    AD  9B 00108           MOVZBW  NAM+57, TEMP_DESC                               ; 0447
                            FE4C   CD        94    AD  D0 0010E           MOVL    NAM+68, TEMP_DESC+4                             ; 0448
     0040     8F      00             6E            00  2C 00114           MOVC5   #0, (SP), #0, #64, FIB                          ; 0452
                                           0260    CE      0011B
                            0260   CE      0501    8F  3C 0011E           MOVZWL  #1281, FIB                                      ; 0453
                            0264   CE  00040004    8F  D0 00125           MOVL    #262148, FIB+4                                  ; 0454
                            0258   CE        40    8F  9A 0012E           MOVZBL  #64, FIB_DESC                                   ; 0459
                            025C   CE      0260    CE  9E 00134           MOVAB   FIB, FIB_DESC+4                                 ; 0460
                            FE34   CD      FE38    CD  9E 0013B           MOVAB   NAME_BUFFER, NAME_DESC+4                        ; 0464
                                                   7E  7C 00142           CLRQ    -(SP)                                           ; 0471
                                             18    AE  9F 00144           PUSHAB  CHANNEL                                         ;
                                           FE48    CD  9F 00147           PUSHAB  TEMP_DESC                                       ;
                       00000000G   00              04  FB 0014B           CALLS   #4, SYS$ASSIGN                                  ;
                                   19              50  E9 00152           BLBC    STATUS_4, 10$                                   ; 0472
                                                   7E  7C 00155           CLRQ    -(SP)                                           ; 0479
                                                   7E  7C 00157           CLRQ    -(SP)                                           ;
                                           FE20    CD  9F 00159           PUSHAB  DVI_ITMLST                                      ;
                                                   7E  D4 0015D           CLRL    -(SP)                                           ;
                                   57        28    AE  3C 0015F           MOVZWL  CHANNEL, R7                                     ;
                                                   57  DD 00163           PUSHL   R7                                              ;
                                                   7E  D4 00165           CLRL    -(SP)                                           ;
                       00000000G   00              08  FB 00167           CALLS   #8, SYS$GETDVIW                                 ;
                                   01              50  E8 0016E 10$:      BLBS    STATUS_4, 11$                                   ; 0480
                                                       04 00171           RET                                                     ;
                    04        03   AE              04  E1 00172 11$:      BBC     #4, DEV_ATTR+3, 12$                             ; 0481
                                   11        03    AE  E9 00177           BLBC    DEV_ATTR+3, 14$                                 ;
                                                   57  DD 0017B 12$:      PUSHL   R7                                              ; 0484
                       00000000G   00              01  FB 0017D           CALLS   #1, SYS$DASSGN                                  ;
                                   50  00000000G   8F  D0 00184 13$:      MOVL    #LIB$_INVFILSPE, R0                             ; 0485
                                                       04 0018B           RET                                                     ;
                                             14    AE  9F 0018C 14$:      PUSHAB  EFN                                             ; 0492
                       00000000G   00              01  FB 0018F           CALLS   #1, LIB$GET_EF                                  ;
                                   52              50  D0 00196           MOVL    R0, STATUS_5                                    ;
                                   0C              52  E8 00199           BLBS    STATUS_5, 15$                                   ; 0493
                                                   57  DD 0019C           PUSHL   R7                                              ; 0496
                       00000000G   00              01  FB 0019E           CALLS   #1, SYS$DASSGN                                  ;
                                                 02C4  31 001A5           BRW     35$                                             ; 0497
                            58        14  AE  D0 001A8 15$:    MOVL    EFN, R8                                  ; 0562
                  026A      CE      0264  CE  D0 001AC 16$:    MOVL    FIB+4, FIB+10                            ; 0519
                  026E      CE      0268  CE  B0 001B3         MOVW    FIB+8, FIB+14                            ; 0521
                            53            5A  D0 001BA         MOVL    DIR_ADDRESS, NAME_ADDRESS                ; 0535
             6A             59            2E  3A 001BD         LOCC    #46, DIR_LENGTH, (DIR_ADDRESS)           ; 0536
             52             59            50  C3 001C1         SUBL3   R0, DIR_LENGTH, NAME_LENGTH              ; 0537
                            5A        01  A1  9E 001C5         MOVAB   1(R1), DIR_ADDRESS                       ; 0538
                            59            70  9E 001C9         MOVAB   -(R0), DIR_LENGTH                        ; 0540
        FE30   CD           52            06  A1 001CC         ADDW3   #6, NAME_LENGTH, NAME_DESC               ; 0547
        FE38   CD           63            52  28 001D2         MOVC3   NAME_LENGTH, (NAME_ADDRESS), NAME_BUFFER ; 0548
             63      FE1D   CF            06  28 001D8         MOVC3   #6, DIR_TYP_VER, (R3)                    ;
                                          7E  7C 001DE         CLRQ    -(SP)                                    ; 0562
                                          7E  7C 001E0         CLRQ    -(SP)                                    ;
                                    FE30  CD  9F 001E2         PUSHAB  NAME_DESC                                ;
                                    026C  CE  9F 001E6         PUSHAB  FIB_DESC                                 ;
                                          7E  7C 001EA         CLRQ    -(SP)                                    ;
                                    0238  CE  9F 001EC         PUSHAB  IOSB                                     ;
                                          32  DD 001F0         PUSHL   #50                                      ;
                                          57  DD 001F2         PUSHL   R7                                       ;
                                          58  DD 001F4         PUSHL   R8                                       ;
                            6B            0C  FB 001F6         CALLS   #12, SYS$QIOW                            ;
                            56            50  D0 001F9         MOVL    R0, FINAL_STATUS                         ;
                            05            56  E9 001FC         BLBC    FINAL_STATUS, 17$                        ; 0563
                            56      0218  CE  3C 001FF         MOVZWL  IOSB, FINAL_STATUS                       ;
              00000910      8F            56  D1 00204 17$:    CMPL    FINAL_STATUS, #2320                      ; 0564
                                          0D  13 0020B         BEQL    19$                                      ;
                            07            56  E9 0020D         BLBC    FINAL_STATUS, 18$                        ; 0565
                                          59  D5 00210         TSTL    DIR_LENGTH                               ; 0572
                                          98  14 00212         BGTR    16$                                      ;
                            56            01  D0 00214         MOVL    #1, FINAL_STATUS                         ; 0575
                                        0236  31 00217 18$:    BRW     34$                                      ; 0576
                  0220      CE 00040020   8F  D0 0021A 19$:    MOVL    #262176, ATR                             ; 0592
                  0224      CE      FE00  CD  9E 00223         MOVAB   RECATTR, ATR+4                           ; 0593
                  0228      CE 00030004   8F  D0 0022A         MOVL    #196612, ATR+8                           ; 0595
                  022C      CE      FDFC  CD  9E 00233         MOVAB   UCHAR, ATR+12                            ; 0596
                  0230      CE 00160002   8F  D0 0023A         MOVL    #1441794, ATR+16                         ; 0598
                  0234      CE      FDF8  CD  9E 00243         MOVAB   FPRO, ATR+20                             ; 0599
                  0238      CE 00150004   8F  D0 0024A         MOVL    #1376260, ATR+24                         ; 0601
                  023C      CE      FDF4  CD  9E 00253         MOVAB   UIC, ATR+28                              ; 0602
                  0240      CE 000A0200   8F  D0 0025A         MOVL    #655872, ATR+32                          ; 0604
                  0244      CE      02A0  CE  9E 00263         MOVAB   HEADER, ATR+36                           ; 0605
                                    0248  CE  D4 0026A         CLRL    ATR+40                                   ; 0606
                  0264      CE      026A  CE  D0 0026E         MOVL    FIB+10, FIB+4                            ; 0614
                  0268      CE      026E  CE  3C 00275         MOVZWL  FIB+14, FIB+8                            ; 0616
                                    026C  CE  D4 0027C         CLRL    FIB+12                                   ; 0618
                                          7E  D4 00280         CLRL    -(SP)                                    ; 0632
                                    0224  CE  9F 00282         PUSHAB  ATR                                      ;
                                          7E  7C 00286         CLRQ    -(SP)                                    ;
                                          7E  D4 00288         CLRL    -(SP)                                    ;
                                    026C  CE  9F 0028A         PUSHAB  FIB_DESC                                 ;
                                          7E  7C 0028E         CLRQ    -(SP)                                    ;
                                    0238  CE  9F 00290         PUSHAB  IOSB                                     ;
                                          32  DD 00294         PUSHL   #50                                      ;
                                          57  DD 00296         PUSHL   R7                                       ;
                                          58  DD 00298         PUSHL   R8                                       ;
                            6B            0C  FB 0029A         CALLS   #12, SYS$QIOW                            ;
                            56            50  D0 0029D         MOVL    R0, FINAL_STATUS                         ;
```

```
                05          56 E9 002A0         BLBC    FINAL_STATUS, 20$                     ; 0633
                56  0218    CE 3C 002A3         MOVZWL  IOSB, FINAL_STATUS                    ;
                03          56 E8 002A8  20$:   BLBS    FINAL_STATUS, 21$                     ; 0634
                        01A2 31 002AB           BRW     34$                                   ;
                    0240    CE D4 002AE  21$:   CLRL    ATR+32                                ; 0642
          026A  CE  0264    CE D0 002B2         MOVL    FIB+4, FIB+10                         ; 0648
          026E  CE  0268    CE B0 002B9         MOVW    FIB+8, FIB+14                         ; 0650
          0276  CE    85    8F 9B 002C0         MOVZBW  #133, FIB+22                          ; 0659
                02  02A7    CE 91 002C6         CMPB    HEADER+7, #2                          ; 0660
                            05 12 002CB         BNEQ    22$                                   ;
          0278  CE          01 D0 002CD         MOVL    #1, FIB+24                            ;
      50  0278  CE          01 C1 002D2  22$:   ADDL3   #1, FIB+24, R0                        ; 0668
FE08  CD        50          10 9C 002D8         ROTL    #16, R0, RECATTR+8                    ;
                02          6C 91 002DE         CMPB    (AP), #2                              ; 0676
                            12 1F 002E1         BLSSU   23$                                   ;
                      08    AC D5 002E3         TSTL    8(AP)                                 ;
                            0D 13 002E6         BEQL    23$                                   ;
                      08    BC D5 002E8         TSTL    @OWNER_UIC                            ; 0679
                            1F 13 002EB         BEQL    25$                                   ;
          FDF4  CD    08    BC D0 002ED         MOVL    @OWNER_UIC, UIC                       ;
                            17 11 002F3         BRB     25$                                   ; 0676
      0E    86  AD          03 E1 002F5  23$:   BBC     #3, NAM+54, 24$                       ; 0682
          FDF6  CD    08    AE B0 002FA         MOVW    GROUP, UIC+2                          ; 0685
          FDF4  CD    0C    AE B0 00300         MOVW    MEMBER, UIC                           ; 0686
                            04 11 00306         BRB     25$                                   ; 0682
                    0238    CE D4 00308  24$:   CLRL    ATR+24                                ; 0689
          FDF8  CD  8888    8F A8 0030C  25$:   BISW2   #34952, FPRO                          ; 0695
                            50 B4 00313         CLRW    LOCAL_ENABLE                          ; 0696
                03          6C 91 00315         CMPB    (AP), #3                              ; 0697
                            09 1F 00318         BLSSU   26$                                   ;
                      0C    AC D5 0031A         TSTL    12(AP)                                ;
                            04 13 0031D         BEQL    26$                                   ;
                50    0C    BC B0 0031F         MOVW    @PROT_ENABLE, LOCAL_ENABLE            ; 0699
                            51 B4 00323  26$:   CLRW    LOCAL_VALUE                           ; 0701
                04          6C 91 00325         CMPB    (AP), #4                              ; 0702
                            09 1F 00328         BLSSU   27$                                   ;
                      10    AC D5 0032A         TSTL    16(AP)                                ;
                            04 13 0032D         BEQL    27$                                   ;
                51    10    BC B0 0032F         MOVW    @PROT_VALUE, LOCAL_VALUE              ; 0704
                52  FDF8    CD 3C 00333  27$:   MOVZWL  FPRO, R2                              ; 0706
                53          50 3C 00338         MOVZWL  LOCAL_ENABLE, R3                      ;
                52          53 CA 0033B         BICL2   R3, R2                                ;
                53          51 3C 0033E         MOVZWL  LOCAL_VALUE, R3                       ;
                50          50 3C 00341         MOVZWL  LOCAL_ENABLE, R0                      ;
                50          50 D2 00344         MCOML   R0, R0                                ;
      50        53          50 CB 00347         BICL3   R0, R3, R0                            ;
FDF8  CD        50          52 A9 0034B         BISW3   R2, R0, FPRO                          ;
                05          6C 91 00351         CMPB    (AP), #5                              ; 0712
                            0B 1F 00354         BLSSU   28$                                   ;
                      14    AC D5 00356         TSTL    20(AP)                                ;
                            06 13 00359         BEQL    28$                                   ;
          FE1E  CD    14    BC B0 0035B         MOVW    @MAX_VERSIONS, RECATTR+30             ; 0714
                06          6C 91 00361  28$:   CMPB    (AP), #6                              ; 0721
                            15 1F 00364         BLSSU   29$                                   ;
                      18    AC D5 00366         TSTL    24(AP)                                ;
                            10 13 00369         BEQL    29$                                   ;
          0280  CE          01 88 0036B         BISB2   #1, FIB+32                            ; 0724
```

```
                          0281    CE              02  90 00370          MOVB    #2, FIB+33                                        ; 0725
                          0286    CE        18    BC  B0 00375          MOVW    @RVN, FIB+38                                      ; 0726
                          0298    CE              04  88 0037B 29$:     BISB2   #4, FIB+56                                        ; 0733
                                                  7E  D4 00380 30$:     CLRL    -(SP)                                             ; 0753
                                          0224    CE  9F 00382          PUSHAB  ATR                                               ;
                                                  7E  7C 00386          CLRQ    -(SP)                                             ;
                                          FE30    CD  9F 00388          PUSHAB  NAME_DESC                                         ;
                                          026C    CE  9F 0038C          PUSHAB  FIB_DESC                                          ;
                                                  7E  7C 00390          CLRQ    -(SP)                                             ;
                                          0238    CE  9F 00392          PUSHAB  IOSB                                              ;
                                  7E        F3    8F  9A 00396          MOVZBL  #243, -(SP)                                       ;
                                                  57  DD 0039A          PUSHL   R7                                                ;
                                                  58  DD 0039C          PUSHL   R8                                                ;
                                  6B              0C  FB 0039E          CALLS   #12, SYS$QIOW                                     ;
                                  56              50  D0 003A1          MOVL    R0, FINAL_STATUS                                  ;
                                  72              56  E9 003A4          BLBC    FINAL_STATUS, 32$                                 ; 0754
                                  56      0218    CE  3C 003A7          MOVZWL  IOSB, FINAL_STATUS                                ;
                                  6A              56  E9 003AC          BLBC    FINAL_STATUS, 32$                                 ; 0755
                                  02      02A7    CE  91 003AF          CMPB    HEADER+7, #2                                      ; 0762
                                                  37  12 003B4          BNEQ    31$                                               ;
                             18   AE              01  AE 003B6          MNEGW   #1, BLOCK_BUFFER                                  ; 0768
      01FE    8F       00         6E              00  2C 003BA          MOVC5   #0, (SP), #0, #510, BLOCK_BUFFER+2                ; 0769
                                            1A    AE     003C1                                                                    ;
                                                  7E  7C 003C3          CLRQ    -(SP)                                             ; 0777
                                  7E              01  7D 003C5          MOVQ    #1, -(SP)                                         ;
                                  7E      0200    8F  3C 003C8          MOVZWL  #512, -(SP)                                       ;
                                            2C    AE  9F 003CD          PUSHAB  BLOCK_BUFFER                                      ;
                                                  7E  7C 003D0          CLRQ    -(SP)                                             ;
                                          0238    CE  9F 003D2          PUSHAB  IOSB                                              ;
                                                  30  DD 003D6          PUSHL   #48                                               ;
                                                  57  DD 003D8          PUSHL   R7                                                ;
                                                  58  DD 003DA          PUSHL   R8                                                ;
                                  6B              0C  FB 003DC          CALLS   #12, SYS$QIOW                                     ;
                                  56              50  D0 003DF          MOVL    R0, FINAL_STATUS                                  ;
                                  6B              56  E9 003E2          BLBC    FINAL_STATUS, 34$                                 ; 0778
                                  56      0218    CE  3C 003E5          MOVZWL  IOSB, FINAL_STATUS                                ;
                                  63              56  E9 003EA          BLBC    FINAL_STATUS, 34$                                 ; 0779
                                                  7E  7C 003ED 31$:     CLRQ    -(SP)                                             ; 0790
                                                  7E  7C 003EF          CLRQ    -(SP)                                             ;
                                                  7E  7C 003F1          CLRQ    -(SP)                                             ;
                                                  7E  7C 003F3          CLRQ    -(SP)                                             ;
                                          0238    CE  9F 003F5          PUSHAB  IOSB                                              ;
                                                  34  DD 003F9          PUSHL   #52                                               ;
                                                  57  DD 003FB          PUSHL   R7                                                ;
                                                  58  DD 003FD          PUSHL   R8                                                ;
                                  6B              0C  FB 003FF          CALLS   #12, SYS$QIOW                                     ;
                                  56              50  D0 00402          MOVL    R0, FINAL_STATUS                                  ;
                                  48              56  E9 00405          BLBC    FINAL_STATUS, 34$                                 ; 0791
                                  56      0218    CE  3C 00408          MOVZWL  IOSB, FINAL_STATUS                                ;
                                  40              56  E9 0040D          BLBC    FINAL_STATUS, 34$                                 ; 0792
                                                  59  D5 00410          TSTL    DIR_LENGTH                                        ; 0798
                                                  07  14 00412          BGTR    33$                                               ;
                                  56      0619    8F  3C 00414          MOVZWL  #1561, FINAL_STATUS                               ; 0801
                                                  35  11 00419 32$:     BRB     34$                                               ; 0802
                                  53              5A  D0 0041B 33$:     MOVL    DIR_ADDRESS, NAME_ADDRESS                         ; 0817
                       6A         59              2E  3A 0041E          LOCC    #46, DIR_LENGTH, (DIR_ADDRESS)                    ; 0818
                       52         59              50  C3 0042?          SUBL3   R0, DIR_LENGTH, NAME_LENGTH                       ; 0819
```

```
                                   5A      01    A1  9E 00426          MOVAB   1(R1), DIR_ADDRESS                             ; 0820
                                   59            70  9E 0042A          MOVAB   -(R0), DIR_LENGTH                              ; 0822
            FE30    CD             52            06  A1 0042D          ADDW3   #6, NAME_LENGTH, NAME_DESC                     ; 0829
            FE38    CD             63            52  28 00433          MOVC3   NAME_LENGTH, (NAME_ADDRESS), NAME_BUFFER       ; 0830
                    63       FBBC  CF            06  28 00439          MOVC3   #6, DIR_TYP_VER, (R3)                          ;
                             026A  CE    0264    CE  D0 0043F          MOVL    FIB+4, FIB+10                                  ; 0838
                             026E  CE    0268    CE  B0 00446          MOVW    FIB+8, FIB+14                                  ; 0840
                                               FF30  31 0044D          BRW     30$                                            ; 0739
                                                 57  DD 00450 34$:     PUSHL   R7                                             ; 0854
                         00000000G  00           01  FB 00452          CALLS   #1, SYS$DASSGN                                 ;
                                   52            50  D0 00459          MOVL    R0, STATUS_6                                   ;
                                           14    AE  9F 0045C          PUSHAB  EFN                                            ; 0855
                         00000000G  00           01  FB 0045F          CALLS   #1, LIB$FREE_EF                                ;
                                   0A            50  E9 00466          BLBC    STATUS_7, 37$                                  ; 0856
                                   04            52  E8 00469          BLBS    STATUS_6, 36$                                  ; 0857
                                   50            52  D0 0046C 35$:     MOVL    STATUS_6, R0                                   ;
                                                     04 0046F          RET                                                    ;
                                   50            56  D0 00470 36$:     MOVL    FINAL_STATUS, R0                               ; 0863
                                                     04 00473 37$:     RET                                                    ; 0864

; Routine Size:  1140 bytes,    Routine Base:  _LIB$CODE + 0016


;  867          0865  1
;  868          0866  1 END                                         ! End of module LIB$CREATE_DIR
;  869          0867  0 ELUDOM



;                                  PSECT SUMMARY
;
;       Name                     Bytes                        Attributes
;
;   _LIB$CODE                        1162  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]LIB.L32;1          18619       133           0     1000          00:01.4




;                                   COMMAND QUALIFIERS

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

; Size:           1140 code + 22 data bytes
; Run Time:          00:19.0
; Elapsed Time:      01:18.7
; Lines/CPU Min:      2735
; Lexemes/CPU-Min:  33495
; Memory Used:   387 pages
; Compilation Complete
```

0204 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LIBCVTDX
LIS

LIBCVTMAC
LIS

LIBCRCTAB
LIS

LIBCVTDF
LIS

LIBCUSTOM
LIS

LIBCREDIR
LIS

LIBCVTATB
LIS

LIBCOMMON
LIS

LIBCRC
LIS