```
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSSSSS  LLL                 IIIIIIIII     BBBBBBBBBBBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMMMMM   MMMMMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM   MMM   MMM  SSS              LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM     SSSSSSSSS     LLL                    III        BBBBBBBBBBBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
VVV         VVV  MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
   VVV   VVV     MMM         MMM              SSS  LLL                    III        BBB         BBB
      VVV        MMM         MMM  SSSSSSSSSSSS     LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
      VVV        MMM         MMM  SSSSSSSSSSSS     LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
      VVV        MMM         MMM  SSSSSSSSSSSS     LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB
```

```
**FILE**ID**LIBEXTCON                                                    J 2




LL            IIIIII    BBBBBBBB    EEEEEEEEEE  XX      XX  TTTTTTTTTT    CCCCCCCC    OOOOOO    NN      NN
LL            IIIIII    BBBBBBBB    EEEEEEEEEE  XX      XX  TTTTTTTTTT    CCCCCCCC    OOOOOO    NN      NN
LL              II      BB      BB  EE          XX      XX      TT      CC          OO      OO  NN      NN
LL              II      BB      BB  EE          XX      XX      TT      CC          OO      OO  NN      NN
LL              II      BB      BB  EE            XX  XX        TT      CC          OO      OO  NNNN    NN
LL              II      BB      BB  EE            XX  XX        TT      CC          OO      OO  NNNN    NN
LL              II      BBBBBBBB    EEEEEEEE        XX          TT      CC          OO      OO  NN  NN  NN
LL              II      BBBBBBBB    EEEEEEEE        XX          TT      CC          OO      OO  NN  NN  NN
LL              II      BB      BB  EE            XX  XX        TT      CC          OO      OO  NN    NNNN
LL              II      BB      BB  EE            XX  XX        TT      CC          OO      OO  NN    NNNN
LL              II      BB      BB  EE          XX      XX      TT      CC          OO      OO  NN      NN      ....
LL              II      BB      BB  EE          XX      XX      TT      CC          OO      OO  NN      NN      ....
LLLLLLLLLL    IIIIII    BBBBBBBB    EEEEEEEEEE  XX      XX      TT        CCCCCCCC    OOOOOO    NN      NN      ....
LLLLLLLLLL    IIIIII    BBBBBBBB    EEEEEEEEEE  XX      XX      TT        CCCCCCCC    OOOOOO    NN      NN      ....


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
```

```
0001  0 %TITLE 'LIB$EXTRACT_CONCEALED - Extract concealed device and root directory'
0002  0 MODULE LIB$EXTRACT_CONCEALED (                  ! Extract concealed device and root directory
0003  0                 IDENT = 'V04-000'                       ! File: LIBEXTCON.B32 Edit: 1-001
0004  0                 ) =
0005  1 BEGIN
0006  1 !
0007  1 !****************************************************************************
0008  1 !*                                                                          *
0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0011  1 !*  ALL RIGHTS RESERVED.                                                    *
0012  1 !*                                                                          *
0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0014  1 !*  ONLY IN  ACCORDANCE WITH  THF  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0018  1 !*  TRANSFERRED.                                                            *
0019  1 !*                                                                          *
0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0022  1 !*  CORPORATION.                                                            *
0023  1 !*                                                                          *
0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0026  1 !*                                                                          *
0027  1 !*                                                                          *
0028  1 !****************************************************************************
0029  1 !
0030  1
0031  1 !++
0032  1 ! FACILITY:     General Utility Library
0033  1 !
0034  1 ! ABSTRACT:
0035  1 !
0036  1 !       This routine determines whether the device component of a file
0037  1 !       specification is a concealed device, and if so, whether it also
0038  1 !       specifies a root directory, and returns the concealed device and
0039  1 !       root directory.
0040  1 !
0041  1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
0042  1 !
0043  1 ! AUTHOR: Martin L. Jack, CREATION DATE: 19-Dec-1981
0044  1 !
0045  1 ! MODIFIED BY:
0046  1 !
0047  1 ! 1-001 - Original.  MLJ 19-Dec-1981
0048  1 !--
0049  1
```

```
;   51        0050  1 %SBTTL 'Declarations'
;   52        0051  1 !
;   53        0052  1 ! SWITCHES:
;   54        0053  1 !
;   55        0054  1
;   56        0055  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;   57        0056  1
;   58        0057  1 !
;   59        0058  1 ! LINKAGES:
;   60        0059  1 !
;   61        0060  1 LINKAGE
;   62        0061  1         LINKAGE_JSB_3_6 = JSB(REGISTER=0,REGISTER=1,REGISTER=2):
;   63        0062  1                           NOPRESERVE(3,4,5,6),
;   64        0063  1
;   65        0064  1         LINKAGE_JSB_2_2 = JSB(REGISTER=0;REGISTER=1,REGISTER=2);
;   66        0065  1 !
;   67        0066  1 ! TABLE OF CONTENTS:
;   68        0067  1 !
;   69        0068  1
;   70        0069  1 FORWARD ROUTINE
;   71        0070  1     LIB$EXTRACT_CONCEALED;                        ! Extract concealed device and root directory
;   72        0071  1
;   73        0072  1 !
;   74        0073  1 ! INCLUDE FILES:
;   75        0074  1 !
;   76        0075  1
;   77        0076  1 LIBRARY 'SYS$LIBRARY:LIB';                      ! System symbols
;   78        0077  1
;   79        0078  1 !*! REQUIRE 'RTLIN:RTLPSECT';                    ! Define PSECT declarations macros
;   80        0079  1
;   81        0080  1 !
;   82        0081  1 ! MACROS:
;   83        0082  1 !
;   84        0083  1 !       NONE
;   85        0084  1 !
;   86        0085  1 ! EQUATED SYMBOLS:
;   87        0086  1 !
;   88        0087  1 !       NONE
;   89        0088  1 !
;   90        0089  1 ! FIELDS:
;   91        0090  1 !
;   92        0091  1 !       NONE
;   93        0092  1 !
;   94        0093  1 ! PSECTS:
;   95        0094  1 !
;   96        0095  1 !*! DECLARE_PSECTS (LIB);                                ! Declare PSECTs for LIB$ facility
;   97        0096  1     PSECT
;   98        0097  1             CODE = _LIB$CODE (READ, NOWRITE, EXECUTE, SHARE, PIC, ADDRESSING_MODE (WORD_RELATIVE)),
;   99        0098  1             PLIT = _LIB$CODE (READ, NOWRITE, EXECUTE, SHARE, PIC, ADDRESSING_MODE (WORD_RELATIVE)),
;  100        0099  1             OWN  = _LIB$DATA (READ, WRITE, NOEXECUTE, NOSHARE, PIC, ADDRESSING_MODE (LONG_RELATIVE)),
;  101        0100  1             GLOBAL = _LIB$DATA (READ, WRITE, NOEXECUTE, NOSHARE, PIC, ADDRESSING_MODE (LONG_RELATIVE)) ;
;  102        0101  1 !
;  103        0102  1 ! OWN STORAGE:
;  104        0103  1 !
;  105        0104  1 !       NONE
;  106        0105  1 !
;  107        0106  1 ! EXTERNAL REFERENCES:
```

```
;  108          0107  1 !
;  109          0108  1
;  110          0109  1 EXTERNAL ROUTINE
;  111          0110  1     LIB$SCOPY_R_DX6:            LINKAGE_JSB_3_6,        ! String copy
;  112          0111  1     LIB$ANALYZE_SDESC_R2:       LINKAGE_JSB_2_2;        ! Analyze descriptor
;  113          0112  1
;  114          0113  1 EXTERNAL LITERAL                            ! Condition value symbols
;  115          0114  1     LIB$_INVARG,                            ! Invalid argument
;  116          0115  1     LIB$_STRTRU,                            ! String truncated
;  117          0116  1     LIB$_INVFILSPE;                         ! Invalid file specification
;   119          0117   1 %SBTTL 'LIB$EXTRACT_CONCEALED - Extract concealed device and root directory'
;   120          0118   1 GLOBAL ROUTINE LIB$EXTRACT_CONCEALED (           ! Extract concealed device and root directory
;   121          0119   1         FILE_SPEC,                              ! File specification
;   122          0120   1         CONCEALED_DEVICE,                       ! concealed device name
;   123          0121   1         ROOT_DIRECTORY,                         ! Root directory name
;   124          0122   1         CONCEALED_DEVICE_LENGTH,                ! Length of CONCEALED_DEVICE
;   125          0123   1         ROOT_DIRECTORY_LENGTH                   ! Length of ROOT_DIRECTORY
;   126          0124   1     ) =
;   127          0125   1
;   128          0126   1 !++
;   129          0127   1 ! FUNCTIONAL DESCRIPTION:
;   130          0128   1 !
;   131          0129   1 !       This routine determines whether the device component of a file
;   132          0130   1 !       specification is a concealed device, and if so, whether it also
;   133          0131   1 !       specifies a root directory, and returns the concealed device and
;   134          0132   1 !       root directory.
;   135          0133   1 !
;   136          0134   1 ! CALLING SEQUENCE:
;   137          0135   1 !
;   138          0136   1 !       ret_status.wlc.v = LIB$EXTRACT_CONCEALED (file-spec.rt.dx,
;   139          0137   1 !               [concealed-device.wt.dx], [root-directory.wt.dx],
;   140          0138   1 !               [concealed-device-length.wwu.r], [root-directory-length.wwu.r])
;   141          0139   1 !
;   142          0140   1 ! FORMAL PARAMETERS:
;   143          0141   1 !
;   144          0142   1 !       FILE_SPEC               Address of a descriptor for the file
;   145          0143   1 !                               specification to be analyzed.  The string must
;   146          0144   1 !                               not be longer than 255 characters.
;   147          0145   1 !
;   148          0146   1 !       CONCEALED_DEVICE        Address of a descriptor to receive the
;   149          0147   1 !                               concealed device name.  This is an optional
;   150          0148   1 !                               output parameter.
;   151          0149   1 !
;   152          0150   1 !       ROOT_DIRECTORY          Address of a descriptor to receive the root
;   153          0151   1 !                               directory name, without brackets or dot.  This
;   154          0152   1 !                               is an optional output parameter.  If the file
;   155          0153   1 !                               specification does not specify a root
;   156          0154   1 !                               directory, this parameter receives a null
;   157          0155   1 !                               string.
;   158          0156   1 !
;   159          0157   1 !       CONCEALED_DEVICE_LENGTH Address of a word to receive the number of
;   160          0158   1 !                               characters written into concealed-device, not
;   161          0159   1 !                               counting padding in the case of a fixed-length
;   162          0160   1 !                               string.  If the output string is truncated to
;   163          0161   1 !                               the size specified in the concealed-device
;   164          0162   1 !                               string, concealed-device-length is set to this
;   165          0163   1 !                               size.  Therefore, concealed-device-length can
;   166          0164   1 !                               always be used by the calling program to access
;   167          0165   1 !                               a valid substring of concealed-device.  This is
;   168          0166   1 !                               an optional output parameter, passed by
;   169          0167   1 !                               reference.
;   170          0168   1 !
;   171          0169   1 !       ROOT_DIRECTORY_LENGTH   Address of a word to receive the number of
;   172          0170   1 !                               characters written into root-directory, not
;   173          0171   1 !                               counting padding in the case of a fixed-length
;   174          0172   1 !                               string.  If the output string is truncated to
;   175          0173   1 !                               the size specified in the root-directory
;  176         0174  1 !                                  string, root-directory-length is set to this
;  177         0175  1 !                                  size.  Therefore, root-directory-length can
;  178         0176  1 !                                  always be used by the calling program to access
;  179         0177  1 !                                  a valid substring of root-directory.  This is
;  180         0178  1 !                                  an optional output parameter, passed by
;  181         0179  1 !                                  reference.
;  182         0180  1 !
;  183         0181  1 !        The output parameters are guaranteed to be stored only if the routine
;  184         0182  1 !        value is true.
;  185         0183  1 !
;  186         0184  1 ! IMPLICIT INPUTS:
;  187         0185  1 !
;  188         0186  1 !        NONE
;  189         0187  1 !
;  190         0188  1 ! IMPLICIT OUTPUTS:
;  191         0189  1 !
;  192         0190  1 !        NONE
;  193         0191  1 !
;  194         0192  1 ! COMPLETION STATUS:
;  195         0193  1 !
;  196         0194  1 !        SS$_NORMAL      Normal successful completion
;  197         0195  1 !
;  198         0196  1 !        LIB$_INVARG     Required argument omitted, or file specification longer
;  199         0197  1 !                        than 255 characters
;  200         0198  1 !
;  201         0199  1 !        LIB$_STRTRU     String truncated (qualified success)
;  202         0200  1 !
;  203         0201  1 !        LIB$_INVFILSPE  String does not represent a valid concealed device with
;  204         0202  1 !                        optional root directory
;  205         0203  1 !
;  206         0204  1 !        LIB$ANALYZE_SDESC errors
;  207         0205  1 !        $TRNLOG errors
;  208         0206  1 !        LIB$SCOPY errors
;  209         0207  1 !
;  210         0208  1 ! SIDE EFFECTS:
;  211         0209  1 !
;  212         0210  1 !        NONE
;  213         0211  1 !
;  214         0212  1 !--
;  215         0213  1
;  216         0214  2 BEGIN
;  217         0215  2 LOCAL
;  218         0216  2          FAB:            $FAB_DECL,                 ! FAB for $PARSE
;  219         0217  2          NAM:            $NAM_DECL,                 ! NAM block for $PARSE
;  220         0218  2          ESA_BUFFER:     VECTOR[NAM$C_MAXRSS,BYTE], ! Expanded string area
;  221         0219  2          INPUT_DESC:     BLOCK[DSC$K_Z_BLN,BYTE],! Descriptor for input string
;  222         0220  2          TRNLOG_DESC:    BLOCK[DSC$K_Z_BLN,BYTE],! Descriptor for translated input string
;  223         0221  2          TRNLOG_BUFFER:  VECTOR[LOG$C_NAMLENGTH,BYTE], ! Buffer for translated input string
;  224         0222  2          INPUT_LENGTH,                              ! Length of compressed input
;  225         0223  2          INPUT_ADDRESS:  REF VECTOR[,BYTE],         ! Address of compressed input
;  226         0224  2          COMPRESS_CURSOR:REF VECTOR[,BYTE],         ! Temporary for upcase/compress
;  227         0225  2          DEV_LENGTH:     WORD,                      ! Length of concealed device
;  228         0226  2          DIR_LENGTH:     WORD,                      ! Length of root directory
;  229         0227  2          DIR_ADDRESS,                               ! Address of root directory
;  230         0228  2          DELIMITER:      BYTE,                      ! Closing directory delimiter
;  231         0229  2          STATUS_1,                                  ! Status return
;  232         0230  2          STATUS_2,                                  ! Status return
```

```
:  233          0231  2             STATUS_3,                               ! Status return
:  234          0232  2             STATUS_4,                               ! Status return
:  235          0233  2             ROUTINE_VALUE;                          . Final routine value
:  236          0234  2     BUILTIN
:  237          0235  2             ACTUALCOUNT,                            ! Determine argument count
:  238          0236  2             LOCC,                                   ! LOCC instruction
:  239          0237  2             NULLPARAMETER;                          ! Test for null parameter
:  240          0238  2
:  241          0239  2     !+
:  242          0240  2     ! Ensure that the required parameter is present.
:  243          0241  2     !-
:  244          0242  2
:  245          0243  2     IF ACTUALCOUNT() EQL 0 THEN RETURN LIB$_INVARG;
:  246          0244  2
:  247          0245  2     !+
:  248          0246  2     ! Initialize RMS structures required to do a $PARSE.
:  249          0247  2     !-
:  250          0248  2
:  251        P 0249  2     $FAB_INIT(FAB=FAB,
:  252          0250  2         NAM=NAM);
:  253        P 0251  2     $NAM_INIT(NAM=NAM,
:  254        P 0252  2         ESA=ESA_BUFFER,
:  255          0253  2         ESS=NAM$C_MAXRSS);
:  256          0254  2
:  257          0255  2     !+
:  258          0256  2     ! Analyze the input descriptor and set up the FAB filename descriptor.
:  259          0257  2     !-
:  260          0258  2
:  261          0259  3     BEGIN ! block to use output registers
:  262          0260  3     REGISTER
:  263          0261  3             R1 = 1,
:  264          0262  3             R2 = 2;
:  265          0263  3
:  266          0264  3     STATUS_1 = LIB$ANALYZE_SDESC_R2(.FILE_SPEC; R1, R2);
:  267          0265  3     IF NOT .STATUS_1 THEN RETURN .STATUS_1;
:  268          0266  3     IF .R1 GTRU 255 THEN RETURN LIB$_INVARG;
:  269          0267  3     FAB[FAB$B_FNS] = .R1;
:  270          0268  3     FAB[FAB$L_FNA] = .R2;
:  271          0269  2     END; ! block to use output registers
:  272          0270  2
:  273          0271  2     !+
:  274          0272  2     ! Parse the input string to obtain the expanded name string.  Ignore errors
:  275          0273  2     ! provided that a device name was actually returned.
:  276          0274  2     !-
:  277          0275  2
:  278          0276  2     $PARSE(FAB=FAB);
:  279          0277  2     IF .NAM[NAM$B_DEV] EQL 0 THEN RETURN LIB$_INVFILSPE;
:  280          0278  2
:  281          0279  2     !+
:  282          0280  2     ! Get a descriptor for the input device specification.  The high order word is
:  283          0281  2     ! not initialized because system services ignore this word.
:  284          0282  2     !-
:  285          0283  2
:  286          0284  2     INPUT_DESC[DSC$W_LENGTH] = .NAM[NAM$B_DEV];
:  287          0285  2     INPUT_DESC[DSC$A_POINTER] = .NAM[NAM$L_DEV];
:  288          0286  2
:  289          0287  2     !+
2 ! If the string contained a wildcard or a node name, internal RMS resources
2 ! have been consumed by $PARSE.  Execute another $PARSE using the same FAB
2 ! on a null string to release these resources.
2 !-
2
2 FAB[FAB$B_FNS] = 0;
2 NAM[NAM$B_ESS] = 0;
2 $PARSE(FAB=FAB);
2
2 !+
2 ! Strip text following a leading colon from the specification.
2 !-
2
3 BEGIN ! block to use output registers
3 REGISTER
3         R0 = 0;
3
3 LOCC(%REF(%C':'), INPUT_DESC[DSC$W_LENGTH], .INPUT_DESC[DSC$A_POINTER]; R0);
3 INPUT_DESC[DSC$W_LENGTH] = .INPUT_DESC[DSC$W_LENGTH] - .R0;
2 END; ! block to use output registers
2
2 !+
2 ! Translate the input device specification.
2 !-
2
2 INPUT_ADDRESS = TRNLOG_BUFFER;
2 TRNLOG_DESC[DSC$B_CLASS] = DSC$K_CLASS_Z;
2 TRNLOG_DESC[DSC$B_DTYPE] = DSC$K_DTYPE_Z;
2 TRNLOG_DESC[DSC$W_LENGTH] = LOG$C_NAMLENGTH;
2 TRNLOG_DESC[DSC$A_POINTER] = .INPUT_ADDRESS;
2 STATUS_2 = $TRNLOG(LOGNAM=INPUT_DESC, RSLLEN=TRNLOG_DESC, RSLBUF=TRNLOG_DESC);
2 IF NOT .STATUS_2 THEN RETURN .STATUS_2;
2
2 !+
2 ! If the string has no translation, it does not represent a concealed device
2 ! or root directory specification.
2 !-
2
2 IF .STATUS_2 EQL SS$_NOTRAN THEN RETURN LIB$_INVFILSPE;
2
2 !+
2 ! Upcase and remove blanks, tabs, and nulls from the string.
2 !-
2
2 INPUT_LENGTH = .TRNLOG_DESC[DSC$W_LENGTH];
2 COMPRESS_CURSOR = .INPUT_ADDRESS;
2 INCR N FROM 0 TO .INPUT_LENGTH-1 DO
3     BEGIN
3     LOCAL
3         CHARACTER: BYTE;                ! Character from string
3
3     CHARACTER = .INPUT_ADDRESS[.N];
3     IF .CHARACTER NEQ %C' ' AND .CHARACTER NEQ %O'011' AND .CHARACTER NEQ 0
3     THEN
4         BEGIN
4         IF .CHARACTER GEQU %C'a' AND .CHARACTER LEQU %C'z'
4             THEN CHARACTER = .CHARACTER - %C'a' + %C'A';
```

```
4         COMPRESS_CURSOR[0] = .CHARACTER;
4         COMPRESS_CURSOR = .COMPRESS_CURSOR + 1;
3         END;
2     END;
2 INPUT_LENGTH = .COMPRESS_CURSOR - .INPUT_ADDRESS;
2 DIR_LENGTH = 0;
2
2
3 BEGIN ! block to use output registers
3 REGISTER
3         R0 = 0,
3         R1 = 1;
3
3 !+
3 ! Test the string for the form "__any:any".  If found, the portion of the
3 ! string ending with the colon is a valid concealed device.
3 !-
3
3 IF .INPUT_LENGTH LSSU 3 THEN RETURN LIB$_INVFILSPE; ! Not long enough
3 IF .(.INPUT_ADDRESS)<0,16> NEQ '__'
3 THEN RETURN LIB$_INVFILSPE;             ! No double underline found
3 IF NOT LOCC(%REF(%C':'), INPUT_LENGTH, .INPUT_ADDRESS; R0, R1)
3 THEN RETURN LIB$_INVFILSPE;             ! No colon found
3 R0 = .R0 - 1;                           ! Adjust count/pointer to include colon
3 R1 = .R1 + 1;
3 DEV_LENGTH = .R1 - .INPUT_ADDRESS;      ! Length up to and including colon
3 INPUT_ADDRESS = .R1;                    ! Prune device from string
3 INPUT_LENGTH = .R0;
2 END; ! block to use output registers
2
2 !+
2 ! If there is no remaining string, exit with success, and no root
2 ! directory.
2 !-
2
2 IF .INPUT_LENGTH NEQ 0
2 THEN
3     BEGIN
3
4     BEGIN ! block to use output registers
4     REGISTER
4         R0 = 0,
4         R1 = 1:                 REF VECTOR[,BYTE];
4
4     !+
4     ! Test the remaining string for the form "[any.]", where angle brackets
4     ! may replace the square brackets.  If found, this string is a valid
4     ! root directory.
4     !-
4
4     IF .INPUT_ADDRESS[0] NEQ %C'[' AND .INPUT_ADDRESS[0] NEQ %C'<'
4     THEN RETURN LIB$_INVFILSPE;                 ! No directory delimiter
4     DELIMITER = .INPUT_ADDRESS[0] - %C'[' + %C']'; ! Get closing delimiter
4     INPUT_ADDRESS = .INPUT_ADDRESS + 1;         ! Prune delimiter from string
4     INPUT_LENGTH = .INPUT_LENGTH - 1;
4     DIR_ADDRESS = .INPUT_ADDRESS;               ! Get pointer to beginning
4     LOCC(%REF(%C'.'), INPUT_LENGTH, .INPUT_ADDRESS; R0, R1);
```

```
;  404          0402  4        IF .R0 NEQ 2 THEN RETURN LIB$_INVFILSPE;     ! Dot followed by one character
;  405          0403  4        IF .R1[1] NEQ .DELIMITER THEN RETURN LIB$_INVFILSPE; ! Wrong delimiter
;  406          0404  4        DIR_LENGTH = .R1 - .INPUT_ADDRESS;           ! Get just intervening part
;  407          0405  3        END; ! block to use output registers
;  408          0406  3
;  409          0407  2        END;
;  410          0408  2
;  411          0409  2   !+
;  412          0410  2   ! Set up to detect string truncation.
;  413          0411  2   !-
;  414          0412  2
;  415          0413  2   ROUTINE_VALUE = SS$_NORMAL;
;  416          0414  2
;  417          0415  2   !+
;  418          0416  2   ! Re urn the concealed device specification, if requested, and determine if
;  419          0417  2   ! truncation occurred.
;  420          0418  2   !-
;  421          0419  2
;  422          0420  2   IF NOT NULLPARAMETER(2)
;  423          0421  2   THEN
;  424          0422  3        BEGIN
;  425          0423  3        REGISTER
;  426          0424  3            R1 = 1:         WORD;
;  427          0425  3
;  428          0426  3        STATUS_3 = LIB$SCOPY_R_DX6(.DEV_LENGTH, TRNLOG_BUFFER, .CONCEALED_DEVICE);
;  429          0427  3        IF NOT .STATUS_3 THEN RETURN .STATUS_3;
;  430          0428  3        LIB$ANALYZE_SDESC_R2(.CONCEALED_DEVICE; R1);
;  431          0429  3        IF .R1 LSSU .DEV_LENGTH
;  432          0430  3        THEN
;  433          0431  4            BEGIN
;  434          0432  4            DEV_LENGTH = .R1;
;  435          0433  4            ROUTINE_VALUE = LIB$_STRTRU;
;  436          0434  3            END;
;  437          0435  2        END;
;  438          0436  2
;  439          0437  2   !+
;  440          0438  2   ! Return the length of the concealed device specification if requested.
;  441          0439  2   !-
;  442          0440  2
;  443          0441  2   IF NOT NULLPARAMETER(4)
;  444          0442  2   THEN
;  445          0443  2        (.CONCEALED_DEVICE_LENGTH)<0,16> = .DEV_LENGTH;
;  446          0444  2
;  447          0445  2   !+
;  448          0446  2   ! Return the root directory specification, if requested, and determine if
;  449          0447  2   ! truncation occurred.
;  450          0448  2   !-
;  451          0449  2
;  452          0450  2   IF NOT NULLPARAMETER(3)
;  453          0451  2   THEN
;  454          0452  3        BEGIN
;  455          0453  3        REGISTER
;  456          0454  3            R1 = 1:         WORD;
;  457          0455  3
;  458          0456  3        STATUS_4 = LIB$SCOPY_R_DX6(.DIR_LENGTH, .DIR_ADDRESS, .ROOT_DIRECTORY);
;  459          0457  3        IF NOT .STATUS_4 THEN RETURN .STATUS_4;
;  460          0458  3        LIB$ANALYZE_SDESC_R2(.ROOT_DIRECTORY; R1);
;  461           0459  3      IF .R1 LSSU .DIR_LENGTH
;  462           0460  3      THEN
;  463           0461  4          BEGIN
;  464           0462  4          DEV_LENGTH = .R1;
;  465           0463  4          ROUTINE_VALUE = LIB$_STRTRU;
;  466           0464  3          END;
;  467           0465  2      END;
;  468           0466  2
;  469           0467  2  !+
;  470           0468  2  ! Return the length of the root directory specification if requested.
;  471           0469  2  !-
;  472           0470  2
;  473           0471  2  IF NOT NULLPARAMETER(5)
;  474           0472  2  THEN
;  475           0473  2      (.ROOT_DIRECTORY_LENGTH)<0,16> = .DIR_LENGTH;
;  476           0474  2
;  477           0475  2  !+
;  478           0476  2  ! Return appropriate status.
;  479           0477  2  !-
;  480           0478  2
;  481           0479  2  RETURN .ROUTINE_VALUE;
;  482           0480  1  END;                                     ! End of routine LIB$EXTRACT_CONCEALED


                                                                              .TITLE  LIB$EXTRACT_CONCEALED LIB$EXTRACT_CONCEALED - E
;                                                                                                           xtract concealed devic
                                                                              .IDENT  \V04-000\

                                                                              .EXTRN  LIB$SCOPY_R_DX6
                                                                              .EXTRN  LIB$ANALYZE_SDESC_R2
                                                                              .EXTRN  LIB$_INVARG, LIB$_STRTRU
                                                                              .EXTRN  LIB$_INVFILSPE, SYS$PARSE
                                                                              .EXTRN  SYS$TRNLOG

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

                                                         0FFC 00000           .ENTRY  LIB$EXTRACT_CONCEALED, Save R2,R3,R4,R5,R6,-; 0118
                                                                                      R7,R8,R9,R10,R11                            ;
                                   5B 00000000G  00  9E 00002                 MOVAB   LIB$ANALYZE_SDESC_R2, R11                   ;
                                   5E      FE00  CE  9E 00009                 MOVAB   -512(SP), SP                                ;
                                                 6C  95 0000E                 TSTB    (AP)                                        ; 0243
                                                 4B  13 00010                 BEQL    1$                                          ;
   0050    8F          00          6E            00  2C 00012                 MOVC5   #0, (SP), #0, #80, $RMS_PTR                 ; 0250
                                             B0  AD     00019                                                                     ;
                              B0   AD      5003  8F  B0 0001B                 MOVW    #20483, $RMS_PTR                            ;
                              C6   AD            02  90 00021                 MOVB    #2, $RMS_PTR+22                             ;
                              CF   AD            02  90 00025                 MOVB    #2, $RMS_PTR+31                             ;
                              D8   AD      FF50  CD  9E 00029                 MOVAB   NAM, $RMS_PTR+40                            ;
   0060    8F          00          6E            00  2C 0002F                 MOVC5   #0, (SP), #0, #96, $RMS_PTR                 ; 0253
                                           FF50  CD     00036                                                                     ;
                            FF50   CD      6002  8F  B0 00039                 MOVW    #24578, $RMS_PTR                            ;
                            FF5A   CD            01  8E 00040                 MNEGB   #1, $RMS_PTR+10                             ;
                            FF5C   CD        50  AE  9E 00045                 MOVAB   ESA_BUFFER, $RMS_PTR+12                     ;
                                   50        04  AC  D0 0004B                 MOVL    FILE_SPEC, R0                               ; 0264
                                                 6B  16 0004F                 JSB     LIB$ANALYZE_SDESC_R2                        ;
                                   6D            50  E9 00051                 BLBC    STATUS_1, 3$                                ; 0265
                        000000FF   8F            51  D1 00054                 CMPL    R1, #255                                    ; 0266
```

```
                                        08  1B 0005B        BLEQU   2$                                      ;
                 50 00000000G           8F  D0 0005D 1$:    MOVL    #LIB$_INVARG, R0                        ;
                                            04 00064        RET                                             ;
                E4      AD              51  90 00065 2$:    MOVB    R1, FAB+52                              ; 0267
                DC      AD              52  D0 00069        MOVL    R2, FAB+44                              ; 0268
                                B0      AD  9F 0006D        PUSHAB  FAB                                     ; 0276
          00000000G     00              01  FB 00070        CALLS   #1, SYS$PARSE                           ;
                                89      AD  95 00077        TSTB    NAM+57                                  ; 0277
                                        50  13 0007A        BEQL    5$                                      ;
                48      AE      89      AD  9B 0007C        MOVZBW  NAM+57, INPUT_DESC                      ; 0284
                4C      AE      94      AD  D0 00081        MOVL    NAM+68, INPUT_DESC+4                    ; 0285
                                E4      AD  94 00086        CLRB    FAB+52                                  ; 0293
                              FF5A      CD  94 00089        CLRB    NAM+10                                  ; 0294
                                B0      AD  9F 0008D        PUSHAB  FAB                                     ; 0295
          00000000G     00              01  FB 00090        CALLS   #1, SYS$PARSE                           ;
    4C   BE     48      AE              3A  3A 00097        LOCC    #58, INPUT_DESC, @INPUT_DESC+4          ; 0305
                48      AE              50  A2 0009D        SUBW2   R0, INPUT_DESC                          ; 0306
                        52              6E  9E 000A1        MOVAB   TRNLOG_BUFFER, INPUT_ADDRESS            ; 0313
                40      AE      40      8F  9A 000A4        MOVZBL  #64, TRNLOG_DESC                        ; 0316
                44      AE              52  D0 000A9        MOVL    INPUT_ADDRESS, TRNLOG_DESC+4            ; 0317
                                        7E  7C 000AD        CLRQ    -(SP)                                   ; 0318
                                        7E  D4 000AF        CLRL    -(SP)                                   ;
                                4C      AE  9F 000B1        PUSHAB  TRNLOG_DESC                             ;
                                50      AE  9F 000B4        PUSHAB  TRNLOG_DESC                             ;
                                5C      AE  9F 000B7        PUSHAB  INPUT_DESC                              ;
          00000000G     00              06  FB 000BA        CALLS   #6, SYS$TRNLOG                          ;
                        01              50  E8 000C1 3$:    BLBS    STATUS_2, 4$                            ; 0319
                                            04 000C4        RET                                             ;
          00000629      8F              50  D1 000C5 4$:    CMPL    STATUS_2, #1577                         ; 0326
                                        7D  13 000CC 5$:    BEQL    10$                                     ;
                        53      40      AE  3C 000CE        MOVZWL  TRNLOG_DESC, INPUT_LENGTH               ; 0332
                        54              52  D0 000D2        MOVL    INPUT_ADDRESS, COMPRESS_CURSOR          ; 0333
                        51              01  CE 000D5        MNEGL   #1, N                                   ; 0334
                                        24  11 000D8        BRB     8$                                      ;
                        50            6142  90 000DA 6$:    MOVB    (N)[INPUT_ADDRESS], CHARACTER           ; 0339
                        20              50  91 000DE        CMPB    CHARACTER, #32                          ; 0340
                                        1B  13 000E1        BEQL    8$                                      ;
                        09              50  91 000E3        CMPB    CHARACTER, #9                           ;
                                        16  13 000E6        BEQL    8$                                      ;
                                        50  95 000E8        TSTB    CHARACTER                               ;
                                        12  13 000EA        BEQL    8$                                      ;
                61      8F              50  91 000EC        CMPB    CHARACTER, #97                          ; 0343
                                        09  1F 000F0        BLSSU   7$                                      ;
                7A      8F              50  91 000F2        CMPB    CHARACTER, #122                         ;
                                        03  1A 000F6        BGTRU   7$                                      ;
                        50              20  82 000F8        SUBB2   #32, CHARACTER                          ; 0344
                        84              50  90 000FB 7$:    MOVB    CHARACTER, (COMPRESS_CURSOR)+           ; 0345
         D8             51              53  F2 000FE 8$:    AOBLSS  INPUT_LENGTH, N, 6$                     ; 0334
         53             54              52  C3 00102        SUBL3   INPUT_ADDRESS, COMPRESS_CURSOR, -       ; 0349
                                                                    INPUT_LENGTH
                                        59  B4 00106        CLRW    DIR_LENGTH                              ; 0350
                        03              53  D1 00108        CMPL    INPUT_LENGTH, #3                        ; 0363
                                        3E  1F 0010B        BLSSU   10$                                     ;
              5F5F      8F              62  B1 0010D        CMPW    (INPUT_ADDRESS), #24415                 ; 0364
                                        37  12 00112        BNEQ    10$                                     ;
         62             53              3A  3A 00114        LOCC    #58, INPUT_LENGTH, (INPUT_ADDRESS)      ; 0366
                                        31  13 00118        BEQL    10$                                     ;
```

```
                     51 D6 0011A       INCL     R1                                              ; 0369
58      51           52 A3 0011C       SUBW3    INPUT_ADDRESS, R1, DEV_LENGTH                   ; 0370
        52           51 D0 00120       MOVL     R1, INPUT_ADDRESS                               ; 0371
        53           70 9E 00123       MOVAB    -(R0), INPUT_LENGTH                             ; 0372
                     2F 13 00126       BEQL     12$                                             ; 0380
    5B  8F           62 91 00128       CMPB     (INPUT_ADDRESS), #91                            ; 0395
                     05 13 0012C       BEQL     9$
        3C           62 91 0012E       CMPB     (INPUT_ADDRESS), #60
                     18 12 00131       BNEQ     10$
54      82           02 81 00133 9$:   ADDB3    #2, (INPUT_ADDRESS)+, DELIMITER                 ; 0397
                     53 D7 00137       DECL     INPUT_LENGTH                                    ; 0399
        5A           52 D0 00139       MOVL     INPUT_ADDRESS, DIR_ADDRESS                      ; 0400
62      53           2E 3A 0013C       LOCC     #46, INPUT_LENGTH, (INPUT_ADDRESS)              ; 0401
        02           50 D1 00140       CMPL     R0, #2                                          ; 0402
                     06 12 00143       BNEQ     10$
        54        01 A1 91 00145       CMPB     1(R1), DELIMITER                                ; 0403
                     08 13 00149       BEQL     11$
        50 00000000G 8F D0 0014B 10$:  MOVL     #LIB$_INVFILSPE, R0
                     04 00152          RET
59      51           52 A3 00153 11$:  SUBW3    INPUT_ADDRESS, R1, DIR_LENGTH                   ; 0404
        57           01 D0 00157 12$:  MOVL     #1, ROUTINE_VALUE                               ; 0413
        02           6C 91 0015A       CMPB     (AP), #2                                        ; 0420
                     2D 1F 0015D       BLSSU    13$
                  08 AC D5 0015F       TSTL     8(AP)
                     28 13 00162       BEQL     13$
        51           6E 9E 00164       MOVAB    TRNLOG_BUFFER, R1                               ; 0426
        52        08 AC D0 00167       MOVL     CONCEALED_DEVICE, R2
        50           58 3C 0016B       MOVZWL   DEV_LENGTH, R0
           00000000G 00 16 0016E       JSB      LIB$SCOPY_R_DX6
        66           50 E9 00174       BLBC     STATUS_3, 17$                                   ; 0427
        50        08 AC D0 00177       MOVL     CONCEALED_DEVICE, R0                            ; 0428
                     6B 16 0017B       JSB      LIB$ANALYZE_SDESC_R2
        58           51 B1 0017D       CMPW     R1, DEV_LENGTH                                  ; 0429
                     0A 1E 00180       BGEQU    13$
        58           51 B0 00182       MOVW     R1, DEV_LENGTH                                  ; 0432
        57 00000000G 8F D0 00185       MOVL     #LIB$_STRTRU, ROUTINE_VALUE                     ; 0433
        04           6C 91 0018C 13$:  CMPB     (AP), #4                                        ; 0441
                     09 1F 0018F       BLSSU    14$
                  10 AC D5 00191       TSTL     16(AP)
                     04 13 00194       BEQL     14$
    10  BC           58 B0 00196       MOVW     DEV_LENGTH, @CONCEALED_DEVICE_LENGTH            ; 0443
        03           6C 91 0019A 14$:  CMPB     (AP), #3                                        ; 0450
                     2D 1F 0019D       BLSSU    15$
                  0C AC D5 0019F       TSTL     12(AP)
                     28 13 001A2       BEQL     15$
        52        0C AC D0 001A4       MOVL     ROOT_DIRECTORY, R2                              ; 0456
        51           5A D0 001A8       MOVL     DIR_ADDRESS, R1
        50           59 3C 001AB       MOVZWL   DIR_LENGTH, R0
           00000000G 00 16 001AE       JSB      LIB$SCOPY_R_DX6
        26           50 E9 001B4       BLBC     STATUS_4, 17$                                   ; 0457
        50        0C AC D0 001B7       MOVL     ROOT_DIRECTORY, R0                              ; 0458
                     6B 16 001BB       JSB      LIB$ANALYZE_SDESC_R2
        59           51 B1 001BD       CMPW     R1, DIR_LENGTH                                  ; 0459
                     0A 1E 001C0       BGEQU    15$
        58           51 B0 001C2       MOVW     R1, DEV_LENGTH                                  ; 0462
        57 00000000G 8F D0 001C5       MOVL     #LIB$_STRTRU, ROUTINE_VALUE                     ; 0463
        05           6C 91 001CC 15$:  CMPB     (AP), #5                                        ; 0471
```

```
                                   09  1F 001CF          BLSSU   16$                                              ;
                              14   AC  D5 001D1          TSTL    20(AP)                                           ;
                                   04  13 001D4          BEQL    16$                                              ;
                    14   BC        59  B0 001D6          MOVW    DIR_LENGTH, @ROOT_DIRECTORY_LENGTH               ; 0473
                         50        57  D0 001DA 16$:     MOVL    ROUTINE_VALUE, R0                                ; 0479
                                       04 001DD 17$:     RET                                                      ; 0480

; Routine Size:  478 bytes,    Routine Base:  _LIB$CODE + 0000
;  484          0481  1 END                                       . End of module LIB$EXTRACT_CONCEALED
;  485          0482  0 ELUDOM



;                                         PSECT SUMMARY
;
;       Name                          Bytes                        Attributes
;
;  _LIB$CODE                               478  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        69         0        1000           00:02.0



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

; Size:          478 code + 0 data bytes
; Run Time:         00:14.5
; Elapsed Time:     00:16.7
; Lines/CPU Min:      1994
; Lexemes/CPU-Min: 20238
; Memory Used:  238 pages
; Compilation Complete
```

0436 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

READOBJ
LIS

SCREEN
LIS

LIBEXECLI
LIS

MOUNTMSG
LIS

PASMSG
LIS

LIBFILPRO
LIS

LIBNETCON
LIS

LIBEXTCON
LIS

LIBMERGE
LIS

LNKMSG
LIS

REPORTIOE
LIS

LIBUNLFIL
LIS

LIBFIDNAM
LIS

MATCHNAME
LIS

OPCMSG
LIS

PLIMSG
LIS