```
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
```

```
PPPPPPPP      AAAAAA      SSSSSSSS  FFFFFFFFFF    IIIIII    NN      NN  DDDDDDDD      222222
PPPPPPPP      AAAAAA      SSSSSSSS  FFFFFFFFFF    IIIIII    NN      NN  DDDDDDDD      222222
PP      PP   AA      AA  SS        FF              II      NN      NN  DD      DD  22      22
PP      PP   AA      AA  SS        FF              II      NN      NN  DD      DD  22      22
PP      PP   AA      AA  SS        FF              II      NNNN    NN  DD      DD          22
PP      PP   AA      AA  SS        FF              II      NNNN    NN  DD      DD          22
PPPPPPPP     AA      AA    SSSSSS  FFFFFFF         II      NN  NN  NN  DD      DD        22
PPPPPPPP     AA      AA    SSSSSS  FFFFFFF         II      NN  NN  NN  DD      DD        22
PP           AAAAAAAAAA          SS  FF            II      NN    NNNN  DD      DD      22
PP           AAAAAAAAAA          SS  FF            II      NN    NNNN  DD      DD      22
PP           AA      AA          SS  FF            II      NN      NN  DD      DD    22          ....
PP           AA      AA          SS  FF            II      NN      NN  DD      DD    22          ....
PP           AA      AA  SSSSSSSS    FF          IIIIII    NN      NN  DDDDDDDD  2222222222    ....
PP           AA      AA  SSSSSSSS    FF          IIIIII    NN      NN  DDDDDDDD  2222222222    ....

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

```
;     1          0001  0 MODULE PAS$FIND2 ( %TITLE 'FIND procedure'
;     2          0002  0                  IDENT = '1-002'                ! File: PASFIND2.B32 Edit: SBL1002
;     3          0003  0                  ) =
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !**************************************************************************
;     7          0007  1 !*                                                                        *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                               *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                   *
;    11          0011  1 !*                                                                        *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  1 !*  TRANSFERRED.                                                           *
;    18          0018  1 !*                                                                        *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  1 !*  CORPORATION.                                                           *
;    22          0022  1 !*                                                                        *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
;    25          0025  1 !*                                                                        *
;    26          0026  1 !*                                                                        *
;    27          0027  1 !**************************************************************************
;    28          0028  1 !
;    29          0029  1
;    30          0030  1 !++
;    31          0031  1 ! FACILITY:     Pascal Language Support
;    32          0032  1 !
;    33          0033  1 ! ABSTRACT:
;    34          0034  1 !
;    35          0035  1 !       This module contains PAS$FIND2, which implements the
;    36          0036  1 !       VAX-11 Pascal FIND procedure.
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  User mode - AST reentrant
;    39          0039  1 !
;    40          0040  1 ! AUTHOR: Steven B. Lionel, CREATION DATE: 16-December-1981
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original.  SBL 16-December-1981
;    45          0045  1 ! 1-002 - Store record length in length word for FILE OF VARYING.
;    46          0046  1 !         SBL 20-Oct-1982
;    47          0047  1 !--
;    48          0048  1
```

```
;    50          0049  1 %SBTTL 'Declarations'
;    51          0050  1 !
;    52          0051  1 ! PROLOGUE DEFINITIONS:
;    53          0052  1 !
;    54          0053  1
;    55          0054  1 REQUIRE 'RTLIN:PASPROLOG';                    ! Externals, linkages, PSECTs, structures
;    56          0118  1
;    57          0119  1 !
;    58          0120  1 ! TABLE OF CONTENTS:
;    59          0121  1 !
;    60          0122  1
;    61          0123  1 FORWARD ROUTINE
;    62          0124  1     PAS$FIND2: NOVALUE;                      ! Do a FIND
;    63          0125  1
;    64          0126  1 !
;    65          0127  1 ! MACROS:
;    66          0128  1 !
;    67          0129  1 !       NONE
;    68          0130  1 !
;    69          0131  1 ! EQUATED SYMBOLS:
;    70          0132  1 !
;    71          0133  1 !       NONE
;    72          0134  1 !
;    73          0135  1 ! FIELDS:
;    74          0136  1 !
;    75          0137  1 !       NONE
;    76          0138  1 !
;    77          0139  1 ! OWN STORAGE:
;    78          0140  1 !
;    79          0141  1 !       NONE
```

```
;    81          0142  1 %SBTTL 'PAS$FIND2 - FIND procedure'
;    82          0143  1 GLOBAL ROUTINE PAS$FIND2 (                          ! Do a FIND
;    83          0144  1         PFV: REF $PAS$PFV_FILE_VARIABLE,            ! File variable
;    84          0145  1         COMPONENT : REF VECTOR [, LONG],            ! Component number
;    85          0146  1         ERROR                                       ! Unwind address if error
;    86          0147  1     ): NOVALUE =
;    87          0148  1
;    88          0149  1 !++
;    89          0150  1 ! FUNCTIONAL DESCRIPTION:
;    90          0151  1 !
;    91          0152  1 !       PAS$FIND2 implements the VAX-11 Pascal FIND procedure.  It
;    92          0153  1 !       finds the specified file component and places its contents in
;    93          0154  1 !       the file buffer.
;    94          0155  1 !
;    95          0156  1 ! CALLING SEQUENCE:
;    96          0157  1 !
;    97          0158  1 !       CALL PAS$FIND2 (PFV.mr.r, COMPONENT.rlu.r [, ERROR.j.r])
;    98          0159  1 !
;    99          0160  1 ! FORMAL PARAMETERS:
;   100          0161  1 !
;   101          0162  1 !       PFV             - The Pascal File Variable (PFV) passed by reference.
;   102          0163  1 !                         The structure of the PFV is defined in PASPFV.REQ.
;   103          0164  1 !
;   104          0165  1 !       COMPONENT       - The number of the file component to read.
;   105          0166  1 !
;   106          0167  1 !       ERROR           - Optional.  If specified, the address to unwind to
;   107          0168  1 !                         if an error occurs.
;   108          0169  1 !
;   109          0170  1 ! IMPLICIT INPUTS:
;   110          0171  1 !
;   111          0172  1 !       NONE
;   112          0173  1 !
;   113          0174  1 ! IMPLICIT OUTPUTS:
;   114          0175  1 !
;   115          0176  1 !       NONE
;   116          0177  1 !
;   117          0178  1 ! ROUTINE VALUE:
;   118          0179  1 !
;   119          0180  1 !       NONE
;   120          0181  1 !
;   121          0182  1 ! SIDE EFFECTS·
;   122          0183  1 !
;   123          0184  1 !       Places file in Inspection mode
;   124          0185  1 !
;   125          0186  1 ! SIGNALLED ERRORS:
;   126          0187  1 !
;   127          0188  1 !       FILNOTOPE - File not open
;   128          0189  1 !       FILNOTDIR - File not opened for direct access
;   129          0190  1 !       FAIGETLOC - Failed to GET locked record
;   130          0191  1 !       ERRDURFIN - Error during FIND or FINDK
;   131          0192  1 !
;   132          0193  1 !
;   133          0194  1 !--
;   134          0195  1
;   135          0196  2     BEGIN
;   136          0197  2
;   137          0198  2     LOCAL
```

```
;  138          0199  2          FCB: REF $PAS$FCB_CONTROL_BLOCK,                . File control block
;  139          0200  2          GET_STATUS,                                     ! Status from $GET
;  140          0201  2          PFV_ADDR: VOLATILE,                             . Enable argument
;  141          0202  2          UNWIND_ACT: VOLATILE,                           . Unwind action
;  142          0203  2          ERROR_ADDR: VOLATILE;                           . Enable argument
;  143          0204  2
;  144          0205  2      BIND
;  145          0206  2          RAB = FCB: REF BLOCK [, BYTE];                  ! RMS RAB
;  146          0207  2
;  147          0208  2      BUILTIN
;  148          0209  2          ACTUALCOUNT;
;  149          0210  2
;  150          0211  2      ENABLE
;  151          0212  2          PAS$$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);
;  152          0213  2
;  153          0214  2      IF ACTUALCOUNT () GEQU 3
;  154          0215  2      THEN
;  155          0216  2          ERROR_ADDR = .ERROR;                            ! Set unwind address
;  156          0217  2
;  157          0218  2      !+
;  158          0219  2      ! Set PFV address enable argument.
;  159          0220  2      !-
;  160          0221  2
;  161          0222  2      PFV_ADDR = PFV [PFV$R_PFV];
;  162          0223  2
;  163          0224  2      !+
;  164          0225  2      ! Validate and lock PFV
;  165          0226  2      !-
;  166          0227  2
;  167          0228  2      PAS$$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
;  168          0229  2
;  169          0230  2      !+
;  170          0231  2      ! Set unwind action to unlock file.
;  171          0232  2      !-
;  172          0233  2
;  173          0234  2      UNWIND_ACT = PAS$K_UNWIND_UNLOCK;
;  174          0235  2
;  175          0236  2      !+
;  176          0237  2      ! Open file if it should be implicitly opened
;  177          0238  2      !-
;  178          0239  2
;  179          0240  2      IF NOT .PFV [PFV$V_VALID]
;  180          0241  2      THEN
;  181          0242  2          PAS$$OPEN_IMPLICIT (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
;  182          0243  2
;  183          0244  2      !+
;  184          0245  2      ! Verify that the file is open.
;  185          0246  2      !-
;  186          0247  2
;  187          0248  2      IF NOT .PFV [PFV$V_OPEN]
;  188          0249  2      THEN
;  189          0250  2          $PAS$IO_ERROR (PAS$_FILNOTOPE,0);
;  190          0251  2
;  191          0252  2      !+
;  192          0253  2      ! Verify that the file is opened for direct access
;  193          0254  2      !-
;  194          0255  2
```

```
;  195     0256  2      IF NOT .FCB [FCB$V_DIRECT]
;  196     0257  2      THEN
;  197     0258  2          $PAS$IO_ERROR (PAS$_FILNOTDIR,0);
;  198     0259  2
;  199     0260  2      !+
;  200     0261  2      ! Assume that file buffer will be defined
;  201     0262  2      !-
;  202     0263  2
;  203     0264  2      PFV [PFV$V_DFB] = 1;
;  204     0265  2
;  205     0266  2      !+
;  206     0267  2      ! Get the record.
;  207     0268  2      !-
;  208     0269  2
;  209     0270  2      RAB [RAB$B_RAC] = RAB$C_KEY;                    ! Set "keyed" (direct) access
;  210     0271  2      RAB [RAB$V_NLK] = 0;                            ! Turn off "no lock" bit
;  211     0272  2      FCB [FCB$L_COMPONENT] = .COMPONENT [0];         ! Set component number
;  212     0273  2      GET_STATUS = $PAS$RMS_OP ($GET (RAB = RAB [0,0,0,0]));
;  213     0274  2
;  214     0275  2      !+
;  215     0276  2      ! If varying, put component length in first word of user buffer.
;  216     0277  2      ! Do this before we check for errors!
;  217     0278  2      !-
;  218     0279  2
;  219     0280  2      IF .FCB [FCB$V_VARYING]
;  220     0281  2      THEN
;  221     0282  2          (.PFV [PFV$A_BUFFER])<0,16,0> = .RAB [RAB$W_RSZ];
;  222     0283  2
;  223     0284  2      !+
;  224     0285  2      ! Check for errors.
;  225     0286  2      !-
;  226     0287  2
;  227     0288  2      IF NOT .GET_STATUS
;  228     0289  2      THEN
;  229     0290  3          BEGIN
;  230     0291  3          !+
;  231     0292  3          ! $GET failed.  Determine if the error was "component not found"
;  232     0293  3          ! or another error.   No error is signalled if component not found,
;  233     0294  3          ! but buffer is undefined.
;  234     0295  3          !-
;  235     0296  3
;  236     0297  3          PFV [PFV$V_DFB] = 0;     ! Undefine file buffer
;  237     0298  3          IF .GET_STATUS EQL RMS$_RLK
;  238     0299  3          THEN
;  239     0300  4              $PAS$IO_ERROR (PAS$_FAIGETLOC)      ! Failed to GET locked record
;  240     0301  4          ELSE IF (.GET_STATUS NEQU RMS$_EOF) AND (.GET_STATUS NEQU RMS$_RNF)
;  241     0302  3          THEN
;  242     0303  3              $PAS$IO_ERROR (PAS$_ERRDURFIN);     ! Error during FIND
;  243     0304  2          END;
;  244     0305  2
;  245     0306  2      !+
;  246     0307  2      ! Set Inspection mode
;  247     0308  2      ! Indicate not at EOF
;  248     0309  2      ! Cancel previous LOCATE
;  249     0310  2      ! Indicate successful completion
;  250     0311  2      ! Unlock the file
;  251     0312  2      !-
```

```
;  252          0313  2
;  253          0314  2      FCB [FCB$V_INSPECTION] = 1;
;  254          0315  2      FCB [FCB$V_GENERATION] = 0;
;  255          0316  2      FCB [FCB$V_EOF] = 0;
;  256          0317  2      FCB [FCB$V_LOCATE] = 0;
;  257          0318  2      FCB [FCB$L_STATUS] = 0;
;  258          0319  2      PFV [PFV$V_LOCK] = 0;
;  259          0320  2
;  260          0321  2      RETURN;
;  261          0322  2
;  262          0323  1      END;                                    ! End of routine PAS$FIND2


                                                                    .TITLE  PAS$FIND2 FIND procedure
                                                                    .IDENT  \1-002\

                                                                    .EXTRN  PAS$FIND2, PAS$$IO_HANDLER
                                                                    .EXTRN  PAS$$VALIDATE_PFV
                                                                    .EXTRN  PAS$$OPEN_IMPLICIT
                                                                    .EXTRN  PAS$$SIGNAL, PAS$K_FILNOTOPE
                                                                    .EXTRN  PAS$K_FILNOTDIR
                                                                    .EXTRN  SYS$GET, PAS$K_FAIGETLOC
                                                                    .EXTRN  PAS$K_ERRDURFIN

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

                                                00FC 00000          .ENTRY  PAS$FIND2, Save R2,R3,R4,R5,R6,R7           ; 0143
                                    55 00000000G    00  9E 00002    MOVAB   PAS$$SIGNAL, R5                             ;
                                    5E              08  C2 00009    SUBL2   #8, SP                                      ;
                                                    7E  D4 0000C    CLRL    ERROR_ADDR                                  ; 0196
                                                04  AE  7C 0000E    CLRQ    UNWIND_ACT                                  ;
                                    6D          00C2  CF  DE 00011  MOVAL   12$, (FP)                                   ;
                                    03                6C  91 00016  CMPB    (AP), #3                                    ; 0214
                                                      04  1F 00019  BLSSU   1$                                          ;
                                    6E            0C  AC  D0 0001B  MOVL    ERROR, ERROR_ADDR                           ; 0216
                                    56            04  AC  D0 0001F 1$:  MOVL    PFV, R6                                 ; 0222
                                            08  AE        56  D0 00023  MOVL    R6, PFV_ADDR                            ;
                                                00000000G   00  16 00027  JSB     PAS$$VALIDATE_PFV                       ; 0228
                                            04  AE        01  D0 0002D  MOVL    #1, UNWIND_ACT                          ; 0234
                                                54    04  A6  9E 00031  MOVAB   4(R6), R4                               ; 0240
                                                06    02  A4  E8 00035  BLBS    2(R4), 2$                               ;
                                                00000000G   00  16 00039  JSB     PAS$$OPEN_IMPLICIT                      ; 0242
                                        08          64        1D  E0 0003F 2$:  BBS     #29, (R4), 3$                   ; 0248
                                                              7E  D4 00043  CLRL    -(SP)                               ; 0250
                                                7E    00G   8F  9A 00045  MOVZBL  #PAS$K_FILNOTOPE, -(SP)                 ;
                                                            0E  11 00049  BRB     4$                                      ;
                                                53    FC    A7  9E 0004B 3$:  MOVAB   -4(FCB), R3                       ; 0256
                                        0A          63        01  E0 0004F  BBS     #1, (R3), 5$                        ;
                                                              7E  D4 00053  CLRL    -(SP)                               ; 0258
                                                7E    00G   8F  9A 00055  MOVZBL  #PAS$K_FILNOTDIR, -(SP)                ;
                                                65          02  FB 00059 4$:  CALLS   #2, PAS$$SIGNAL                     ;
                                                              04  0005C  RET                                           ;
                                            02  A4          02  88 0005D 5$:  BISB2   #2, 2(R4)                       ; 0264
                                            1E  A7          01  90 00061  MOVB    #1, 30(FCB)                         ; 0270
                                            06  A7          10  8A 00065  BICB2   #16, 6(FCB)                         ; 0271
                                            D8  A7    08    BC  D0 00069  MOVL    @COMPONENT, -40(FCB)                ; 0272
                                                              57  DD 0006E 6$:  PUSHL   FCB                            ; 0273
```

```
                 00000000G  00              01  FB 00070          CALLS   #1, SYS$GET                                  ;
                            0D              50  E8 00077          BLBS    $$STATUS, 7$                                  ;
                 0001825A   8F              50  D1 0007A          CMPL    $$STATUS, #98906                              ;
                                            04  12 00081          BNEQ    7$                                            ;
                            E7        03    A3  E8 00083          BLBS    3(R3), 6$                                     ;
                            52              50  D0 00087 7$:      MOVL    $$STATUS, GET_STATUS                          ;
              05       F8   A7              02  E1 0008A          BBC     #2, -8(FCB), 8$                               ; 0280
                       00   B6        22    A7  B0 0008F          MOVW    34(FCB), @0(R6)                               ; 0282
                            2D              52  E8 00094 8$:      BLBS    GET_STATUS, 11$                               ; 0288
                       02   A4              02  8A 00097          BICB2   #2, 2(R4)                                     ; 0297
                 000182AA   8F              52  D1 0009B          CMPL    GET_STATUS, #98986                            ; 0298
                                            06  12 000A2          BNEQ    9$                                            ;
                            7E        00G   8F  9A 000A4          MOVZBL  #PAS$K_FAIGETLOC, -(SP)                       ; 0300          ; F
                                            16  11 000A8          BRB     10$                                           ;
                 0001827A   8F              52  D1 000AA 9$:      CMPL    GET_STATUS, #98938                            ; 0301
                                            11  13 000B1          BEQL    11$                                           ;               ;
                 000182B2   8F              52  D1 000B3          CMPL    GET_STATUS, #98994                            ;               ;
                                            08  13 000BA          BEQL    11$                                           ;
                            7E        00G   8F  9A 000BC          MOVZBL  #PAS$K_ERRDURFIN, -(SP)                       ; 0303
                            65              01  FB 000C0 10$:      CALLS   #1, PAS$$SIGNAL                               ;
                                            04  000C3              RET                                                   ;
                       01   A3              08  88 000C4 11$:      BISB2   #8, 1(R3)                                     ; 0314
                       01   A3        0130  8F  AA 000C8          BICW2   #304, 1(R3)                                   ; 0317
                                      D4    A7  D4 000CE          CLRL    -44(FCB)                                      ; 0318
                       03   A4        80    8F  8A 000D1          BICB2   #128, 3(R4)                                   ; 0319
                                            04  000D6              RET                                                   ; 0323
                                      0000      000D7 12$:        .WORD   Save nothing                                  ; 0196
                            50        08    AC  D0 000D9          MOVL    8(AP), R0                                     ;
                            50        04    A0  D0 000DD          MOVL    4(R0), R0                                     ;
                                      F4    A0  9F 000E1          PUSHAB  ERROR_ADDR                                    ;
                                      F8    A0  9F 000E4          PUSHAB  UNWIND_ACT                                    ;
                                      FC    A0  9F 000E7          PUSHAB  PFV_ADDR                                      ;
                                            03  DD 000EA          PUSHL   #3                                            ;
                                            5E  DD 000EC          PUSHL   SP                                            ;
                                  7E        04  AC  7D 000EE      MOVQ    4(AP), -(SP)                                  ;
                      00000000G   00              03  FB 000F2    CALLS   #3, PAS$$IO_HANDLER                           ;
                                                  04  000F9      RET                                                   ;

; Routine Size:  250 bytes,    Routine Base:  _PAS$CODE + 0000

;  263          0324  1
;  264          0325  1 !<BLF/PAGE>
```

```
;  266          0326  1 END
;  267          0327  1
;  268          0328  0 ELUDOM

                                                        . end of module PAS$FIND2
```

PSECT SUMMARY

Name	Bytes	Attributes
_PAS$CODE	250	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	Symbols Total	Symbols Loaded	Symbols Percent	Pages Mapped	Processing Time
_255DUA28:[SYSLIB]STARLET.L32;1	9776	12	0	581	00:01.0
_255DUA28:[PASRTL.OBJ]PASLIB.L32;1	427	94	22	33	00:00.4

COMMAND QUALIFIERS

```
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:PASFIND2/OBJ=OBJ$:PASFIND2 MSRC$:PASFIND2/UPDATE=(ENH$:PASFIND2)

; Size:         250 code + 0 data bytes
; Run Time:       00:07.5
; Elapsed Time:   00:21.2
; Lines/CPU Min:   2624
; Lexemes/CPU-Min: 15832
; Memory Used:  110 pages
; Compilation Complete
```

PASFVOUTP
LIS

PASEOLN2
LIS

PASHEAP
LIS

PASHANDLE
LIS

PASFAB
LIS

PASGET
LIS

PASCVTRT
LIS

PASDATE
LIS

PASEOF2
LIS

PASFINDK
LIS

PASFVINPU
LIS

PASEXPO
LIS

PASGOTO
LIS

PASFILEUT
LIS

PASHALT
LIS

PASDELETE
LIS

PASFIND2
LIS