D 16

```
FFFFFFFFFFFFFFF        111              111           AAAAAAAAA
FFFFFFFFFFFFFFF        111              111           AAAAAAAAA
FFFFFFFFFFFFFFF        111              111           AAAAAAAAA
FFF                 111111           111111        AAA         AAA
FFF                 111111           111111        AAA         AAA
FFF                 111111           111111        AAA         AAA
FFF                    111              111        AAA         AAA
FFF                    111              111        AAA         AAA
FFF                    111              111        AAA         AAA
FFFFFFFFFFFF           111              111        AAA         AAA
FFFFFFFFFFFF           111              111        AAA         AAA
FFFFFFFFFFFF           111              111        AAA         AAA
FFF                    111              111        AAAAAAAAAAAAAAA
FFF                    111              111        AAAAAAAAAAAAAAA
FFF                    111              111        AAAAAAAAAAAAAAA
FFF                    111              111        AAA         AAA
FFF                    111              111        AAA         AAA
FFF                    111              111        AAA         AAA
FFF                 111111111        111111111     AAA         AAA
FFF                 111111111        111111111     AAA         AAA
FFF                 111111111        111111111     AAA         AAA
```

```
**FILE**ID**NXTHDR                                   N 5
```

```
NN      NN  XX      XX  TTTTTTTTTT  HH      HH  DDDDDDDD    RRRRRRRR
NN      NN  XX      XX  TTTTTTTTTT  HH      HH  DDDDDDDD    RRRRRRRR
NN      NN  XX      XX      TT      HH      HH  DD      DD  RR      RR
NN      NN  XX      XX      TT      HH      HH  DD      DD  RR      RR
NNNN    NN    XX  XX        TT      HH      HH  DD      DD  RR      RR
NNNN    NN    XX  XX        TT      HH      HH  DD      DD  RR      RR
NN  NN  NN      XX          TT      HHHHHHHHHH  DD      DD  RRRRRRRR
NN  NN  NN      XX          TT      HHHHHHHHHH  DD      DD  RRRRRRRR
NN    NNNN    XX  XX        TT      HH      HH  DD      DD  RR  RR
NN    NNNN    XX  XX        TT      HH      HH  DD      DD  RR  RR
NN      NN  XX      XX      TT      HH      HH  DD      DD  RR    RR        ....
NN      NN  XX      XX      TT      HH      HH  DD      DD  RR    RR        ....
NN      NN  XX      XX      TT      HH      HH  DDDDDDDD    RR      RR      ....
NN      NN  XX      XX      TT      HH      HH  DDDDDDDD    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
```

```
PM
VO

;
```

```
0001 0 MODULE NXTHDR (
0002 0                 LANGUAGE (BLISS32),
0003 0                 IDENT = 'V04-000'
0004 0                 ) =
0005 1 BEGIN
0006 1
0007 1 !
0008 1 !****************************************************************************
0009 1 !*                                                                          *
0010 1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0011 1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0012 1 !*  ALL RIGHTS RESERVED.                                                    *
0013 1 !*                                                                          *
0014 1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0015 1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0016 1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0017 1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0018 1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0019 1 !*  TRANSFERRED.                                                            *
0020 1 !*                                                                          *
0021 1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0022 1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0023 1 !*  CORPORATION.                                                            *
0024 1 !*                                                                          *
0025 1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0026 1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0027 1 !*                                                                          *
0028 1 !*                                                                          *
0029 1 !****************************************************************************
0030 1
0031 1 !++
0032 1 !
0033 1 ! FACILITY:  F11ACP Structure Level 1
0034 1 !
0035 1 ! ABSTRACT:
0036 1 !
0037 1 !       This routine reads the next extension header, if any, of the
0038 1 !       given file.
0039 1 !
0040 1 ! ENVIRONMENT:
0041 1 !
0042 1 !       STARLET operating system, including privileged system services
0043 1 !       and internal exec routines.
0044 1 !
0045 1 !--
0046 1 !
0047 1 !
0048 1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  22-Jul-1977  17:40
0049 1 !
0050 1 ! MODIFIED BY:
0051 1 !
0052 1 !       A0100   ACG0001         Andrew C. Goldstein,  10-Oct-1978  20:01
0053 1 !               Previous revision history moved to F11A.REV
0054 1 !
0055 1 !**
0056 1
0057 1
```

```
;   58          0058  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;   59          0059  1 REQUIRE 'SRC$:FCPDEF.B32';
```

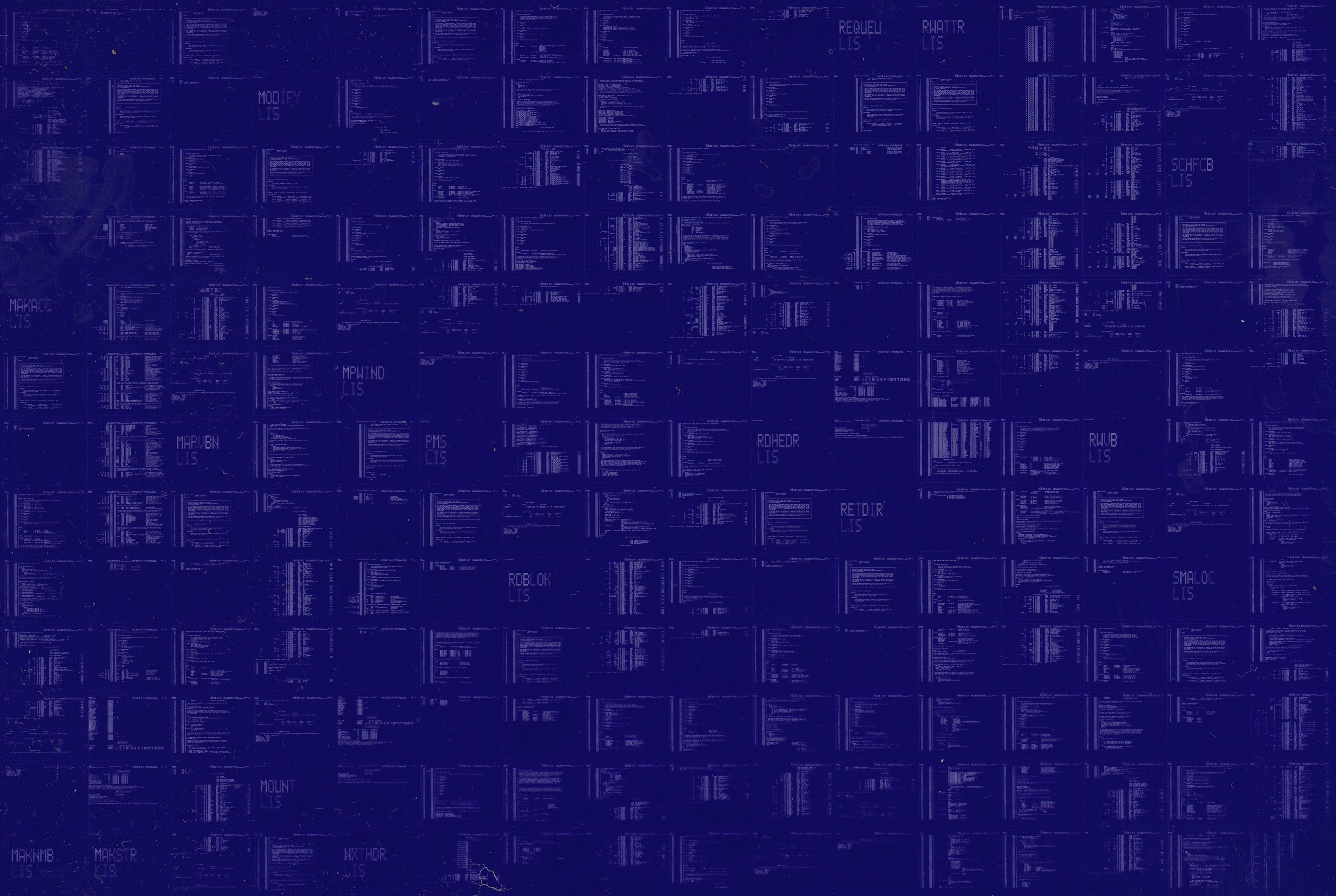
```
1 GLOBAL ROUTINE NEXT_HEADER (HEADER, FCB, EXT_FID, SEGNUM) =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine reads the next extension header, if any, of the
1 !       indicated file. Extension data is taked from either the indicated
1 !       file header or the arguments.
1 !
1 !
1 ! CALLING SEQUENCE:
1 !       NEXT_HEADER (ARG1, ARG2, ARG3, ARG4)
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1: address of current file header or 0
1 !       ARG2: address of corresponding FCB or zero
1 !       ARG3: extension file ID, if present
1 !       ARG4: extension segment number, if present
1 !
1 ! IMPLICIT INPUTS:
1 !       NONE
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       Address of header read or 0 if none
1 !
1 ! SIDE EFFECTS:
1 !       File header may be read
1 !
1 !--
1
2 BEGIN
2
2 MAP
2       HEADER          : REF BBLOCK,   ! file header arg
2       FCB             : REF BBLOCK,   ! FCB arg
2       EXT_FID         : REF BBLOCK;   ! extension file ID arg
2
2 LOCAL
2       NEW_HEADER      : REF BBLOCK,   ! address of extension file header read
2       MAP_AREA        : REF BBLOCK,   ! address of header map area
2       EXT_FCB         : REF BBLOCK,   ! address of extension FCB
2       FILE_ID         : BBLOCK [FID$C_LENGTH], ! file ID of extension header
2       SEG_NUMBER      : BYTE;         ! segment number of file header
2
2 EXTERNAL ROUTINE
2       READ_HEADER;                    ! read a file header
2
2
2 ! Get the extension file number of the file header. If it is zero, then
2 ! there is no extension header. If it is non-zero, read the header, using
```

```
;  118         0431  2 ! the extension FCB if one exists.
;  119         0432  2 !
;  120         0433  2
;  121         0434  2 IF ACTUALCOUNT LSS 4
;  122         0435  2 THEN
;  123         0436  3     BEGIN
;  124         0437  3     MAP_AREA = .HEADER + .HEADER[FH1$B_MPOFFSET]*2;
;  125         0438  3     FILE_ID[FID$W_NUM] = .MAP_AREA[FM1$W_EX_FILNUM];
;  126         0439  3     FILE_ID[FID$W_SEQ] = .MAP_AREA[FM1$W_EX_FILSEQ];
;  127         0440  3     FILE_ID[FID$W_RVN] = 0;
;  128         0441  3     SEG_NUMBER = .MAP_AREA[FM1$B_EX_SEGNUM] + 1;
;  129         0442  3     END
;  130         0443  2 ELSE
;  131         0444  3     BEGIN
;  132         0445  3     CH$MOVE (FID$C_LENGTH, .EXT_FID, FILE_ID);
;  133         0446  3     SEG_NUMBER = .SEGNUM;
;  134         0447  2     END;
;  135         0448  2
;  136         0449  2 IF .FILE_ID[FID$W_NUM] EQL 0 THEN RETURN 0;
;  137         0450  2 EXT_FCB =
;  138         0451  3     (IF .FCB NEQ 0
;  139         0452  3      THEN .FCB[FCB$L_EXFCB]
;  140         0453  3      ELSE 0
;  141         0454  2     );
;  142         0455  2 NEW_HEADER = READ_HEADER (FILE_ID, .EXT_FCB);
;  143         0456  2
;  144         0457  2 ! Check the segment number of the header read for consistency.
;  145         0458  2 !
;  146         0459  2
;  147         0460  2 MAP_AREA = .NEW_HEADER + .NEW_HEADER[FH1$B_MPOFFSET]*2;
;  148         0461  2 IF .SEG_NUMBER NEQ .MAP_AREA[FM1$B_EX_SEGNUM]
;  149         0462  2 THEN ERR_EXIT (SS$_BADFILEHDR);
;  150         0463  2
;  151         0464  2 RETURN .NEW_HEADER;
;  152         0465  2
;  153         0466  1 END;                                          ! end of routine NEXT_HEADER


                                                                    .TITLE  NXTHDR
                                                                    .IDENT  \V04-000\

                                                                    .EXTRN  READ_HEADER

                                                                    .PSECT  $CODE$,NOWRT,2

                                        003C 00000                  .ENTRY  NEXT_HEADER, Save R2,R3,R4,R5                    ; 0374
                          5E              08  C2 00002              SUBL2   #8, SP                                           ;
                          04              6C  91 00005              CMPB    (AP), #4                                         ; 0434
                                          19  1E 00008              BGEQU   1$                                               ;
                          51        04    AC  D0 0000A              MOVL    HEADER, R1                                       ; 0437
                          50        01    A1  9A 0000E              MOVZBL  1(R1), R0                                        ;
                          52            6140  3E 00012              MOVAW   (R1)[R0], MAP_AREA                               ;
                          6E        02    A2  D0 00016              MOVL    2(MAP_AREA), FILE_ID                             ; 0438
                                    04    AE  B4 0001A              CLRW    FILE_ID+4                                        ; 0440
          53              62              01  81 0001D              ADDB3   #1, (MAP_AREA), SEG_NUMBER                       ; 0441
                                          09  11 00021              BRB     2$                                               ; 0434
          6E        0C    BC              06  28 00023 1$:          MOVC3   #6, @EXT_FID, FILE_ID                            ; 0445
```

```
             53       10   AC  90 00028         MOVB    SEGNUM, SEG_NUMBER                          ; 0446
                           6E  B5 0002C  2$:    TSTW    FILE_ID                                     ; 0449
                           31  13 0002E         BEQL    6$                                          ;
             50       08   AC  D0 00030         MOVL    FCB, R0                                     ; 0451
                           06  13 00034         BEQL    3$                                          ;
             50       0C   A0  D0 00036         MOVL    12(R0), EXT_FCB                             ; 0452
                           02  11 0003A         BRB     4$                                          ;
                           50  D4 0003C  3$:    CLRL    EXT_FCB                                     ; 0451
                           50  DD 0003E  4$:    PUSHL   EXT_FCB                                     ; 0455
                      04   AE  9F 00040         PUSHAB  FILE_ID                                     ;
        0000G  CF          02  FB 00043         CALLS   #2, READ_HEADER                             ;
               51          50  D0 00048         MOVL    R0, NEW_HEADER                              ;
               50     01   A1  9A 0004B         MOVZBL  1(NEW_HEADER), R0                           ; 0460
               52       6140   3E 0004F         MOVAW   (NEW_HEADER)[R0], MAP_AREA                  ;
               62          53  91 00053         CMPB    SEG_NUMBER, (MAP_AREA)                      ; 0461
                           05  13 00056         BEQL    5$                                          ;
                  0810     8F  BF 00058         CHMU    #2064                                       ; 0462
                               04 0005C         RET                                                 ;
             50            51  D0 0005D  5$:    MOVL    NEW_HEADER, R0                              ; 0464
                               04 00060         RET                                                 ;
                           50  D4 00061  6$:    CLRL    R0                                          ; 0466
                               04 00063         RET                                                 ;

; Routine Size:  100 bytes,    Routine Base:  $CODE$ + 0000


;   154          0467  1
;   155          0468  1 END
;   156          0469  0 ELUDOM




;                         PSECT SUMMARY
;
;      Name                      Bytes                      Attributes
;
;   $CODE$                           100  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                        Library Statistics
;
;                                        -------- Symbols --------          Pages          Processing
;      File                              Total    Loaded   Percent          Mapped         Time
;
;   _$255$DUA28:[SYSLIB]LIB.L32;1        18619        11         0          1000             00:01.9
;                                 COMMAND QUALIFIERS

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

; Size:          100 code + 0 data bytes
; Run Time:         00:06.5
; Elapsed Time:     00:17.9
; Lines/CPU Min:     4315
; Lexemes/CPU-Min: 13628
; Memory Used:  88 pages
; Compilation Complete
```

0166 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



REQUEU
LIS

RWATTR
LIS

MODIFY
LIS

SCHFCB
LIS

MAKACC
LIS

MPWIND
LIS

MAPVBN
LIS

PMS
LIS

RDHEDR
LIS

RWVB
LIS

RETDIR
LIS

RDBLOK
LIS

SMALOC
LIS

MOUNT
LIS

MAKNMB
LIS

MAKSTR
LIS

NXTHDR
LIS