H 5

```
FFFFFFFFFFFFFFF         111              111         XXX         XXX
FFFFFFFFFFFFFFF         111              111         XXX         XXX
FFFFFFFFFFFFFFF         111              111         XXX         XXX
FFF                  111111           111111         XXX         XXX
FFF                  111111           111111         XXX         XXX
FFF                  111111           111111         XXX         XXX
FFF                     111              111            XXX   XXX
FFF                     111              111            XXX   XXX
FFF                     111              111            XXX   XXX
FFFFFFFF.FFF            111              111               XXX
FFFFFFFFFFFF            111              111               XXX
FFFFFFFFFFFF            111              111               XXX
FFF                     111              111            XXX   XXX
FFF                     111              111            XXX   XXX
FFF                     111              111            XXX   XXX
FFF                     111              111         XXX         XXX
FFF                     111              111         XXX         XXX
FFF                     111              111         XXX         XXX
FFF                  111111111        111111111      XXX         XXX
FFF                  111111111        111111111      XXX         XXX
FFF                  111111111        111111111      XXX         XXX
```

```
_$25

Symb
----
IOC$
IO_C
IO_C
IO_D
IO_F
IO_S
KILL

KILL
KILL
LB_E
LB_D
LB_F
LB_H
LB_L
LOCA
LOCA
LOCK

LOCK
LOCK
LOCK
LOC_
LOC_
L_CC
L_CC
L_DA
L_DA
MAIN
MAKE
MAKE
MAKE
MAKE
MAKE
MAKE

MAKE
MAKE
MAP_
MAP_

MAP
MARK
MARK
MARK
MARK
MARK
```

```
 GGGGGGGG   EEEEEEEEEE  TTTTTTTTTT  TTTTTTTTTT    IIIIII    MM      MM
 GGGGGGGG   EEEEEEEEEE  TTTTTTTTTT  TTTTTTTTTT    IIIIII    MM      MM
GG          EE              TT          TT          II      MMMM  MMMM
GG          EE              TT          TT          II      MMMM  MMMM
GG          EE              TT          TT          II      MM  MM  MM
GG          EE              TT          TT          II      MM  MM  MM
GG          EEEEEEEE        TT          TT          II      MM      MM
GG          EEEEEEEE        TT          TT          II      MM      MM
GG  GGGGGG  EE              TT          TT          II      MM      MM
GG  GGGGGG  EE              TT          TT          II      MM      MM
GG      GG  EE              TT          TT          II      MM      MM      ....
GG      GG  EE              TT          TT          II      MM      MM      ....
 GGGGGG     EEEEEEEEEE      TT          TT        IIIIII    MM      MM      ....
 GGGGGG     EEEEEEEEEE      TT          TT        IIIIII    MM      MM      ....

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

```
0 MODULE GETTIM (
0                 LANGUAGE (BLISS32),
0                 IDENT = 'V04-000'
0                 ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 !
1 ! FACILITY:  F11ACP Structure Level 1
1 !
1 ! ABSTRACT:
1 !
1 !       This routine builds a date/time string in the format for file headers
1 !       in the indicated buffer.
1 !
1 ! ENVIRONMENT:
1 !
1 !       STARLET operating system, including privileged system services
1 !       and internal exec routines.
1 !
1 !--
1 !
1 !
1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  2-May-1977  16:07
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-001 CDS0001         Christian D. Saether     4-May-1983
1 !               Make descriptor PIC.
1 !
1 !       A0100   ACG00001        Andrew C. Goldstein,  10-Oct-1978  20:03
1 !       Previous revision history moved to F11A.REV
1 !**
```

```
;    58          0058  1
;    59          0059  1
;    60          0060  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    61          0061  1 REQUIRE 'SRC$:FCPDEF.B32';
```

IN

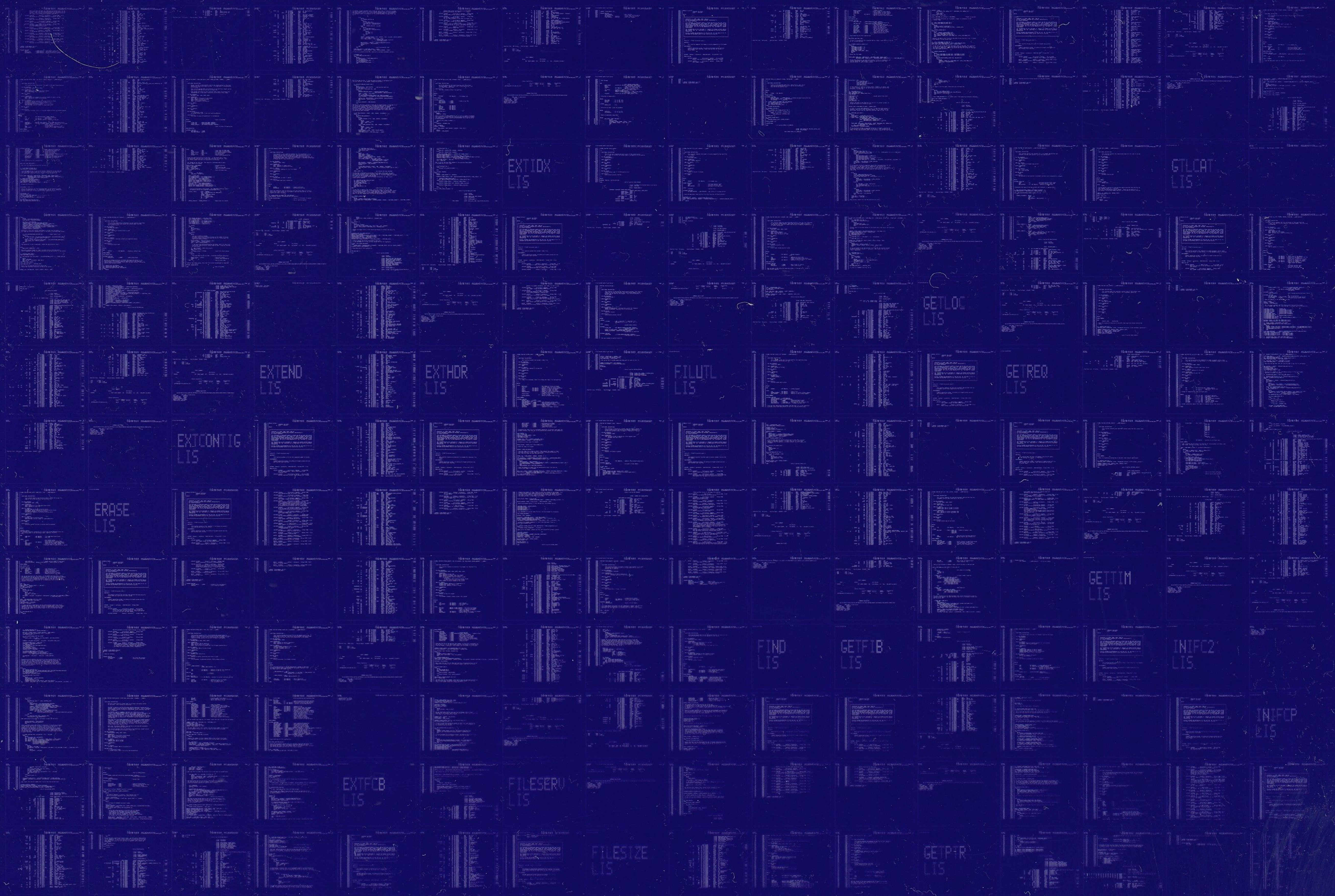
```
1 GLOBAL ROUTINE GET_TIME (BUFFER, TIME) : NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine builds a date/time string in the format for file headers
1 !       in the indicated buffer. If a time is supplied, it is used; if not,
1 !       the current time is used.
1 !
1 ! CALLING SEQUENCE:
1 !       GET_TIME (ARG1, ARG2)
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !       system time of day
1 !
1 ! OUTPUT PARAMETERS:
1 !       ARG1: address of buffer to receive date/time string
1 !       ARG2: optional 64 bit date/time to convert
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2
2 LITERAL                                         ! entries in the numeric date/time buffer
2         YEAR            = 0,
2         MONTH           = 1,
2         DAY             = 2,
2         HOUR            = 3,
2         MINUTE          = 4,
2         SECOND          = 5;
2
2 MAP
2         BUFFER          : REF VECTOR [,BYTE]; ! date/time buffer arg
2
2 LOCAL
2         TIMEADDR,                               ! address of 64 bit date/time
2         DUMMY,                                  ! dummy to receive string length
2         TIMBUF          : VECTOR [7, WORD], ! buffer to receive numeric format time
2         FORMAT          : VECTOR [2],   ! FAO format string descriptor
2         BUF_DESCRIPT    : VECTOR [2];   ! descriptor to pass buffer to FAO
2
2 BIND
2                                                 ! month name table
2         MONTH_TABLE     = UPLIT BYTE
```

```
;  120           1109  2                             ('JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC ')                :
;  121           1110  2                             : VECTOR;                                                           :
;  122           1111  2                                                                                                 :
;  123           1112  2 EXTERNAL ROUTINE                                                                                :
;  124           1113  2         SYS$NUMTIM          : ADDRESSING_MODE (ABSOLUTE),                                       :
;  125           1114  2                                                 ! system time in numeric format                 :
;  126           1115  2         SYS$FAO             : ADDRESSING_MODE (ABSOLUTE);                                       :
;  127           1116  2                                                 ! formatted ASCII output                        :
;  128           1117  2                                                                                                 :
;  129           1118  2                                                                                                 :
;  130           1119  2 ! Execution of this routine simply involves getting the current date and time                 :
;  131           1120  2 ! in numeric format from the system, and then calling FAO to build the ASCII
;  132           1121  2 ! date/time string in the buffer. The string is of the form "DDMMMYYHHMMSS".
;  133           1122  2 !
;  134           1123  2
;  135           1124  2 BUF_DESCRIPT[0] = FI1$S_CREDATE + FI1$S_CRETIME;
;  136           1125  2 BUF_DESCRIPT[1] = .BUFFER;
;  137           1126  2
;  138           1127  2 TIMEADDR = 0;
;  139           1128  2 IF ACTUALCOUNT GEQ 2 THEN TIMEADDR = .TIME;
;  140           1129  2 SYS$NUMTIM (TIMBUF, .TIMEADDR);
;  141           1130  2
;  142           1131  2 PIC_DESC ('!2ZW!AD!4(2ZW)', FORMAT);
;  143           1132  2
;  144           1133  2 SYS$FAO (FORMAT, DUMMY, BUF_DESCRIPT,
;  145           1134  2         .TIMBUF[DAY],
;  146           1135  2         3, MONTH_TABLE[.TIMBUF[MONTH]-1],
;  147           1136  2         .TIMBUF[YEAR] MOD 100,
;  148           1137  2         .TIMBUF[HOUR],
;  149           1138  2         .TIMBUF[MINUTE],
;  150           1139  2         .TIMBUF[SECOND]
;  151           1140  2         );
;  152           1141  2
;  153           1142  1 END;                                             ! end of routine GET_TIME


                                                                                                .TITLE  GETTIM
                                                                                                .IDENT  \V04-000\

                                                                                                .PSECT  $CODE$,NOWRT,2

52  50  41  20  52  41  4D  20  42  45  46  20  4E  41  4A  00000 P.AAA:  .ASCII  \JAN FEB MAR APR MAY JUN JUL AUG SEP OCT \  ;
55  41  20  4C  55  4A  20  4E  55  4A  20  59  41  4D  20  0000F                                                             ;
                    20  54  43  4F  20  50  45  53  20  47  0001E                                                             ;
                            20  43  45  44  20  56  4F  4E  00028          .ASCII  \NOV DEC \                                 ;
00  29  57  5A  32  28  34  21  44  41  21  57  5A  32  21  00030 P.AAB:  .ASCII  \!2ZW!AD!4(2ZW)\<0><0>                     ;
                                                        00  0003F                                                             ;

                                                                  MONTH_TABLE=          P.AAA
                                                                           .EXTRN  SYS$NUMTIM, SYS$FAO

                                                    0000 00000             .ENTRY  GET_TIME, Save nothing                    ; 1052
                                 5E             24  C2 00002               SUBL2   #36, SP                                   ;
                           04    AE             0D  D0 00005               MOVL    #13, BUF_DESCRIPT                         ; 1124
                           08    AE       04    AC  D0 00009               MOVL    BUFFER, BUF_DESCRIPT+4                    ; 1125
                                                50  D4 0000E               CLRL    TIMEADDR                                  ; 1127
                                 02             6C  91 00010               CMPB    (AP), #2                                  ; 1128
```

```
                                                    04  1F 00013         BLSSU   1$                                         ;
                                       50      08   AC  D0 00015         MOVL    TIME, TIMEADDR                             ;
                                                    50  DD 00019 1$:     PUSHL   TIMEADDR                                   ; 1129
                                               18   AE  9F 0001B         PUSHAB  TIMBUF                                     ;
                            00000000G  9F           02  FB 0001E         CALLS   #2, @#SYS$NUMTIM                           ;
                                   0C  AE           0E  D0 00025         MOVL    #14, FORMAT                                ; 1131
                                   10  AE      C4   AF  9E 00029         MOVAB   P.AAB, FORMAT+4                            ;
                                       7E      1E   AE  3C 0002E         MOVZWL  TIMBUF+10, -(SP)                           ; 1139
                                       7E      20   AE  3C 00032         MOVZWL  TIMBUF+8, -(SP)                            ; 1138
                                       7E      22   AE  3C 00036         MOVZWL  TIMBUF+6, -(SP)                            ; 1137
                                       50      20   AE  3C 0003A         MOVZWL  TIMBUF, R0                                 ; 1136
           7E            00            50           01  7A 0003E         EMUL    #1, R0, #0, -(SP)                          ;
           6E            7E            8E 00000064  8F  7B 00043         EDIV    #100, (SP)+, -(SP), (SP)                   ;
                                       50      26   AE  3C 0004C         MOVZWL  TIMBUF+2, R0                               ; 1135
                                             FF67 CF40  DF 00050         PUSHAL  MONTH_TABLE-4[R0]                          ;
                                                    03  DD 00055         PUSHL   #3                                         ;
                                       7E      30   AE  3C 00057         MOVZWL  TIMBUF+4, -(SP)                            ;
                                               20   AE  9F 0005B         PUSHAB  BUF_DESCRIPT                               ; 1133
                                               20   AE  9F 0005E         PUSHAB  DUMMY                                      ;
                                               30   AE  9F 00061         PUSHAB  FORMAT                                     ;
                            00000000G  9F           0A  FB 00064         CALLS   #10, @#SYS$FAO                             ; 1135
                                                        04 0006B         RET                                                ; 1142

; Routine Size:  108 bytes,    Routine Base:  $CODE$ + 0040


;  154          1143  1
;  155          1144  1 END
;  156          1145  0 ELUDOM




;                                     PSECT SUMMARY
;
;       Name                    Bytes                        Attributes
;
;  $CODE$                            172  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        19         0     1000          00:01.9
;                                          COMMAND QUALIFIERS

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

; Size:          108 code + 64 data bytes
; Run Time:         00:08.8
; Elapsed Time:     00:19.1
; Lines/CPU Min:      7789
; Lexemes/CPU-Min: 27802
; Memory Used:  112 pages
; Compilation Complete
```

0170 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



EXTIDX
LIS

GTLCAT
LIS

GETLOC
LIS

EXTEND
LIS

EXTHDR
LIS

FILUTL
LIS

GETREQ
LIS

EXTCONTIG
LIS

ERASE
LIS

GETTIM
LIS

FIND
LIS

GETFIB
LIS

INIFC2
LIS

INIFCP
LIS

EXTFCB
LIS

FILESERV
LIS

FILESIZE
LIS

GETPTR
LIS