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
```

F 13

FILEID**GETTIM

```
  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 !       V02-001 MLJ0029         Martin L. Jack, 11-Jul-1981  23:49
1 !               Clean up quadword to file header format time conversion so that
1 !               if a zero quadword is supplied, the result is binary zeros,
1 !               which signifies an unspecified time, rather than "17NOV58".
1 !
1 !       A0100   ACG0001         Andrew C. Goldstein,  10-Oct-1978  20:01
```

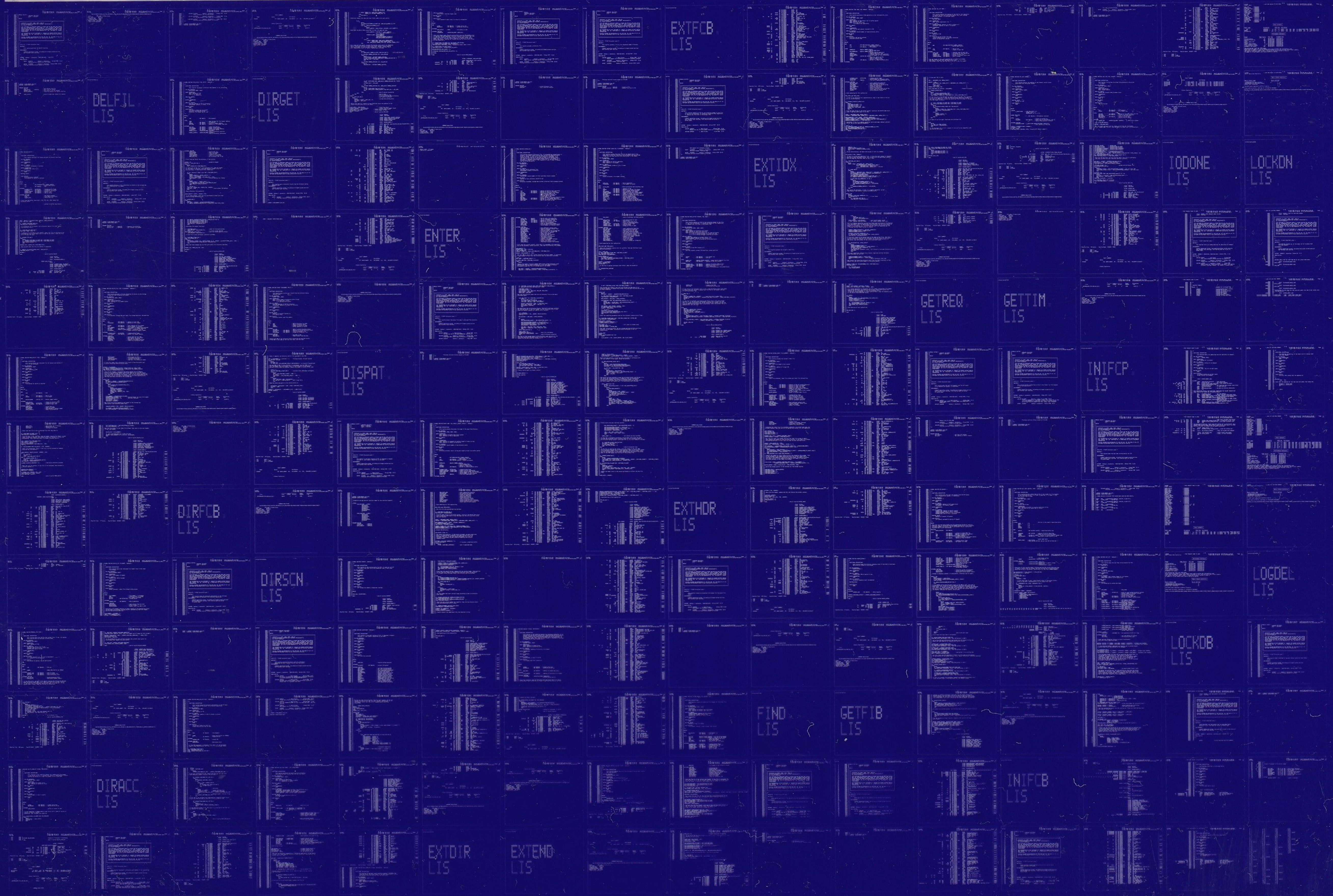
```
:     58          0058  1 !                    Previous revision history moved to F11A.REV
:     59          0059  1 !
:     60          0060  1 !**
:     61          0061  1
:     62          0062  1
:     63          0063  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
:     64          0064  1 REQUIRE 'SRC$:FCPDEF.B32';
```

```
;   66          0379  1 GLOBAL ROUTINE GET_TIME (BUFFER, TIME) : NOVALUE =
;   67          0380  1
;   68          0381  1 !++
;   69          0382  1 !
;   70          0383  1 ! FUNCTIONAL DESCRIPTION:
;   71          0384  1 !
;   72          0385  1 !       This routine builds a date/time string in the format for file headers
;   73          0386  1 !       in the indicated buffer. If a time is supplied, it is used; if not,
;   74          0387  1 !       the current time is used.
;   75          0388  1 !
;   76          0389  1 ! CALLING SEQUENCE:
;   77          0390  1 !       GET_TIME (ARG1, ARG2)
;   78          0391  1 !
;   79          0392  1 ! INPUT PARAMETERS:
;   80          0393  1 !       NONE
;   81          0394  1 !
;   82          0395  1 ! IMPLICIT INPUTS:
;   83          0396  1 !       system time of day
;   84          0397  1 !
;   85          0398  1 ! OUTPUT PARAMETERS:
;   86          0399  1 !       ARG1: address of buffer to receive date/time string
;   87          0400  1 !       ARG2: optional 64 bit date/time to convert
;   88          0401  1 !
;   89          0402  1 ! IMPLICIT OUTPUTS:
;   90          0403  1 !       NONE
;   91          0404  1 !
;   92          0405  1 ! ROUTINE VALUE:
;   93          0406  1 !       NONE
;   94          0407  1 !
;   95          0408  1 ! SIDE EFFECTS:
;   96          0409  1 !       NONE
;   97          0410  1 !
;   98          0411  1 !--
;   99          0412  1
;  100          0413  2 BEGIN
;  101          0414  2
;  102          0415  2 LITERAL                                 ! entries in the numeric date/time buffer
;  103          0416  2         YEAR            = 0,
;  104          0417  2         MONTH           = 1,
;  105          0418  2         DAY             = 2,
;  106          0419  2         HOUR            = 3,
;  107          0420  2         MINUTE          = 4,
;  108          0421  2         SECOND          = 5;
;  109          0422  2
;  110          0423  2 MAP
;  111          0424  2         BUFFER          : REF VECTOR [,BYTE]; ! date/time buffer arg
;  112          0425  2
;  113          0426  2 LOCAL
;  114          0427  2         TIMEADDR        : REF VECTOR,   ! address of 64 bit date/time
;  115          0428  2         DUMMY,                          ! dummy to receive string length
;  116          0429  2         TIMBUF          : VECTOR [7, WORD], ! buffer to receive numeric format time
;  117          0430  2         BUF_DESCRIPT    : VECTOR [2];   ! descriptor to pass buffer to FAO
;  118          0431  2
;  119          0432  2 BIND
;  120          0433  2                                         ! month name table
;  121          0434  2         MONTH_TABLE     = UPLIT BYTE
;  122          0435  2                         ('JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC ')
;  123          0436  2                           : VECTOR,
;  124          0437  2                                           ! FAO format string
;  125          0438  2         FORMAT            = DESCRIPTOR ('!2ZW!AD!4(2ZW)');
;  126          0439  2
;  127          0440  2 EXTERNAL ROUTINE
;  128          0441  2         SYS$NUMTIM        : ADDRESSING_MODE (ABSOLUTE),
;  129          0442  2                                           ! system time in numeric format
;  130          0443  2         SYS$FAO           : ADDRESSING_MODE (ABSOLUTE);
;  131          0444  2                                           ! formatted ASCII output
;  132          0445  2
;  133          0446  2
;  134          0447  2 ! Execution of this routine simply involves getting the current date and time
;  135          0448  2 ! in numeric format from the system, and then calling FAO to build the ASCII
;  136          0449  2 ! date/time string in the buffer. The string is of the form 'DDMMMYYHHMMSS'.
;  137          0450  2 !
;  138          0451  2
;  139          0452  2 BUF_DESCRIPT[0] = FI1$S_CREDATE + FI1$S_CRETIME;
;  140          0453  2 BUF_DESCRIPT[1] = .BUFFER;
;  141          0454  2
;  142          0455  2 TIMEADDR = 0;
;  143          0456  2 IF ACTUALCOUNT GEQ 2
;  144          0457  2 THEN
;  145          0458  3     BEGIN
;  146          0459  3     TIMEADDR = .TIME;
;  147          0460  3
;  148          0461  3     ! If the input time is zero, return binary zeros in the output buffer,
;  149          0462  3     ! which is the ODS-1 representation for an unspecified time.
;  150          0463  3     !
;  151          0464  3     IF .TIMEADDR[0] EQL 0 AND .TIMEADDR[1] EQL 0
;  152          0465  3     THEN
;  153          0466  4         BEGIN
;  154          0467  4         CH$FILL(0, FI1$S_CREDATE + FI1$S_CRETIME, .BUFFER);
;  155          0468  4         RETURN;
;  156          0469  3         END;
;  157          0470  2     END;
;  158          0471  2
;  159          0472  2 SYS$NUMTIM (TIMBUF, .TIMEADDR);
;  160          0473  2
;  161          0474  2 SYS$FAO (FORMAT, DUMMY, BUF_DESCRIPT,
;  162          0475  2         .TIMBUF[DAY],
;  163          0476  2         3, MONTH_TABLE[.TIMBUF[MONTH]-1],
;  164          0477  2         .TIMBUF[YEAR] MOD 100,
;  165          0478  2         .TIMBUF[HOUR],
;  166          0479  2         .TIMBUF[MINUTE],
;  167          0480  2         .TIMBUF[SECOND]
;  168          0481  2         );
;  169          0482  2
;  170          0483  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 \                                  ;
    29  57  5A  32  28  34  21  44  41  21  57  5A  32  21  00030 P.AAC:  .ASCII  \!2ZW!AD!4(2ZW)\                            ;
                                                            0003E         .BLKB   2
                                                  0000000E  00040 P.AAB:  .LONG   14                                          ;
                                                  00000000' 00044         .ADDRESS P.AAC                                      ;

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

                                                       007C 00000         .ENTRY  GET_TIME, Save R2,R3,R4,R5,R6               ; 0379
                                       5E            1C  C2 00002         SUBL2   #28, SP                                     ;
                                  04   AE            0D  D0 00005         MOVL    #13, BUF_DESCRIPT                           ; 0452
                                  08   AE       04   AC  D0 00009         MOVL    BUFFER, BUF_DESCRIPT+4                      ; 0453
                                                     56  D4 0000E         CLRL    TIMEADDR                                    ; 0455
                                       02            0C  91 00010         CMPB    (AP), #2                                    ; 0456
                                                     15  1F 00013         BLSSU   1$                                          ;
                                       56       08   AC  D0 00015         MOVL    TIME, TIMEADDR                              ; 0459
                                                     66  D5 00019         TSTL    (TIMEADDR)                                  ; 0464
                                                     0D  12 0001B         BNEQ    1$                                          ;
                                                04   A6  D5 0001D         TSTL    4(TIMEADDR)                                 ;
                                                     08  12 00020         BNEQ    1$                                          ;
           0D            00            6E            00  2C 00022         MOVC5   #0, (SP), #0, #13, @BUFFER                  ; 0467
                                                04   BC     00027                                                             ;
                                                         04 00029         RET                                                 ; 0466
                                                     56  DD 0002A 1$:     PUSHL   TIMEADDR                                    ; 0472
                                                10   AE  9F 0002C         PUSHAB  TIMBUF                                      ;
                            00000000G  9F            02  FB 0002F         CALLS   #2, @#SYS$NUMTIM                            ;
                                       7E       16   AE  3C 00036         MOVZWL  TIMBUF+10, -(SP)                            ; 0480
                                       7E       18   AE  3C 0003A         MOVZWL  TIMBUF+8, -(SP)                             ; 0479
                                       7E       1A   AE  3C 0003E         MOVZWL  TIMBUF+6, -(SP)                             ; 0478
                                       50       18   AE  3C 00042         MOVZWL  TIMBUF, R0                                  ; 0477
           7E            00            50            01  7A 00046         EMUL    #1, R0, #0, -(SP)                           ;
           6E            7E            8E 00000064   8F  7B 0004B         EDIV    #100, (SP)+, -(SP), (SP)                    ;
                                       50       1E   AE  3C 00054         MOVZWL  TIMBUF+2, R0                                ; 0476
                                              FF57 CF40  DF 00058         PUSHAL  MONTH_TABLE-4[R0]                           ;
                                                     03  DD 0005D         PUSHL   #3                                          ;
                                       7E       28   AE  3C 0005F         MOVZWL  TIMBUF+4, -(SP)                             ;
                                                20   AE  9F 00063         PUSHAB  BUF_DESCRIPT                                ; 0474
                                                20   AE  9F 00066         PUSHAB  DUMMY                                       ;
                                                8C   AF  9F 00069         PUSHAB  FORMAT                                      ;
                            00000000G  9F            0A  FB 0006C         CALLS   #10, @#SYS$FAO                              ; 0476
                                                         04 00073         RET                                                 ; 0483

; Routine Size:  116 bytes,    Routine Base:  $CODE$ + 0048


;  171         0484  1
;  172         0485  1 END
;  173         0486  0 ELUDOM
;                                        PSECT SUMMARY
;
;       Name                         Bytes                          Attributes
;
;   $CODE$                                188  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          3         0       1000           00:01.9




;                                        COMMAND QUALIFIERS

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

; Size:          116 code + 72 data bytes
; Run Time:         00:06.4
; Elapsed Time:     00:20.4
; Lines/CPU Min:     4527
; Lexemes/CPU-Min: 13798
; Memory Used:  80 pages
; Compilation Complete
```

0165 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



EXTFCB
LIS

DELFIL
LIS

DIRGET
LIS

EXTIDX
LIS

IODONE
LIS

LOCKDN
LIS

ENTER
LIS

GETREQ
LIS

GETTIM
LIS

DISPAT
LIS

INIFCP
LIS

DIRFCB
LIS

EXTHDR
LIS

DIRSCN
LIS

LOGDEL
LIS

LOCKDB
LIS

FIND
LIS

GETFIB
LIS

DIRACC
LIS

INIFCB
LIS

EXTDIR
LIS

EXTEND
LIS