


```

1 0001 0 MODULE GETTIM (
2 0002 0
3 0003 0     LANGUAGE (BLISS32),
4 0004 0     IDENT = 'V04-000'
5 0005 1 ) =
6 0006 1 BEGIN
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 *  ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 *  TRANSFERRED.
20 0020 1 *
21 0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 *  CORPORATION.
24 0024 1 *
25 0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 ++
32 0032 1
33 0033 1 FACILITY: F11ACP Structure Level 1
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1
37 0037 1     This routine builds a date/time string in the format for file headers
38 0038 1     in the indicated buffer.
39 0039 1
40 0040 1 ENVIRONMENT:
41 0041 1
42 0042 1     STARLET operating system, including privileged system services
43 0043 1     and internal exec routines.
44 0044 1
45 0045 1 --
46 0046 1
47 0047 1
48 0048 1 AUTHOR: Andrew C. Goldstein, CREATION DATE: 2-May-1977 16:07
49 0049 1
50 0050 1 MODIFIED BY:
51 0051 1
52 0052 1     V02-001 MLJ0029      Martin L. Jack, 11-Jul-1981 23:49
53 0053 1     Clean up quadword to file header format time conversion so that
54 0054 1     if a zero quadword is supplied, the result is binary zeros,
55 0055 1     which signifies an unspecified time, rather than "17NOV58".
56 0056 1
57 0057 1     A0100 ACG0001      Andrew C. Goldstein, 10-Oct-1978 20:01

```

GETTIM
V04-000

H 13
16-Sep-1984 01:07:07
14-Sep-1984 12:29:37

VAX-11 BLISS-32 V4.0-742
DISK\$VMSMASTER:[F11A.SRC]GETTIM.B32;1 Page (1)

Previous revision history moved to F11A.REV

:	58	0058	1	:	
:	59	0059	1	:	
:	60	0060	1	:	**
:	61	0061	1	:	
:	62	0062	1	:	
:	63	0063	1	:	LIBRARY 'SYSSLIBRARY:LIB.L32';
:	64	0064	1	:	REQUIRE 'SRC\$:FCPDEF.B32';

IN

.....

```

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_DESCRIPTOR : 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      : = DESCRIPTOR ('!2ZW!AD!4(2ZW)');
: 126      0439      2      :
: 127      0440      2      :
: 128      0441      2      : EXTERNAL ROUTINE
: 129      0442      2      : SYSSNUMTIM      : ADDRESSING_MODE (ABSOLUTE),
: 130      0443      2      :                  : system time in numeric format
: 131      0444      2      : SYSSFAO        : ADDRESSING_MODE (ABSOLUTE);
: 132      0445      2      :                  : formatted ASCII output
: 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 'DMMMYYHHMMSS'.
: 137      0450      2      : !
: 138      0451      2      :
: 139      0452      2      : BUF_DESCRIPTOR[0] = F11$$_CREDATE + F11$$_CRETIME;
: 140      0453      2      : BUF_DESCRIPTOR[1] = .BUFFER;
: 141      0454      2      :
: 142      0455      2      : TIMEADDR = 0;
: 143      0456      2      : IF ACTUALCOUNT GEQ 2
: 144      0457      2      : THEN
: 145      0458      2      : BEGIN
: 146      0459      2      :     TIMEADDR = .TIME;
: 147      0460      2      :
: 148      0461      2      :     ! If the input time is zero, return binary zeros in the output buffer,
: 149      0462      2      :     ! which is the ODS-1 representation for an unspecified time.
: 150      0463      2      :     !
: 151      0464      2      :     IF .TIMEADDR[0] EQL 0 AND .TIMEADDR[1] EQL 0
: 152      0465      2      :     THEN
: 153      0466      2      :     BEGIN
: 154      0467      2      :     CH$FILL(0, F11$$_CREDATE + F11$$_CRETIME, .BUFFER);
: 155      0468      2      :     RETURN;
: 156      0469      2      :     END;
: 157      0470      2      : END;
: 158      0471      2      :
: 159      0472      2      : SYSSNUMTIM (TIMBUF, .TIMEADDR);
: 160      0473      2      :
: 161      0474      2      : SYSSFAO (FORMAT, DUMMY, BUF_DESCRIPTOR,
: 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      2      : 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				
29	57	5A	32	28	34	21	44	41	21	57	5A	32	21	00028
														00030
														0003E
														00040
														00044

```

P.AAC: .ASCII \NOV DEC \
        .ASCII \!2ZW!AD!4(2ZW)\
        .BLKB 2
P.AAB: .LONG 14
        .ADDRESS P.AAC

```

```

MONTH TABLE= P.AAA
FORMAT= P.AAB
        .EXTRN SYSS$NUMTIM, SYSS$FAO

```

										007C	00000		
										1C	C2	00002	
										0D	D0	00005	
										AC	D0	00009	
										56	D4	0000E	
										5C	91	00010	
										15	1F	00013	
										AC	D0	00015	
										66	D5	00019	
										0D	12	0001B	
										A6	D5	0001D	
										08	12	00020	
										00	2C	00022	
										BC		00027	
										04		00029	
										56	DD	0002A	
										AE	9F	0002C	
										02	FB	0002F	
										AE	3C	00036	
										AE	3C	0003A	
										AE	3C	0003E	
										AE	3C	00042	
										01	7A	00046	
										8F	7B	0004B	
										1E	AE	3C	00054
										CF40	DF	00058	
										03	DD	0005D	
										AE	3C	0005F	
										AE	9F	00063	
										AE	9F	00066	
										AF	9F	00069	
										0A	FB	0006C	
										04		00073	

```

.ENTRY GET_TIME, Save R2,R3,R4,R5,R6
SUBL2 #28, SP
MOVL #13, BUF_DESCRIPTOR
MOVL BUFFER, BUF_DESCRIPTOR+4
CLRL TIMEADDR
CMPB (AP), #2
BLSSU 1$
MOVL TIME, TIMEADDR
TSTL (TIMEADDR)
BNEQ 1$
TSTL 4(TIMEADDR)
BNEQ 1$
MOVCS #0, (SP), #0, #13, @BUFFER
RET
PUSHL TIMEADDR
PUSHAB TIMBUF
CALLS #2, @SYSS$NUMTIM
MOVZWL TIMBUF+10, -(SP)
MOVZWL TIMBUF+8, -(SP)
MOVZWL TIMBUF+6, -(SP)
MOVZWL TIMBUF, R0
EMUL #1, R0, #0, -(SP)
EDIV #100, (SP)+, -(SP), (SP)
MOVZWL TIMBUF+2, R0
PUSHAL MONTH_TABLE-4[R0]
PUSHL #3
MOVZWL TIMBUF+4, -(SP)
PUSHAB BUF_DESCRIPTOR
PUSHAB DUMMY
PUSHAB FORMAT
CALLS #10, @SYSS$FAO
RET

```

; Routine Size: 116 bytes, Routine Base: \$CODES + 0048

```

: 171      0484 1
: 172      0485 1 END
: 173      0486 0 ELUDOM

```

PSECT SUMMARY

```

:
: Name                Bytes                Attributes
: $CODES              188 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
:

```

Library Statistics

```

:
: File                ----- Symbols ----- Pages Processing
:                   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

