

CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCCCCCCCCCCC	LLL	IIIIIIII	UUU	UUU	TTTTTTTTTTTTTTTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCC	LLL	III	UUU	UUU	TTT	LLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL
CCCCCCCCCCCC	LLLLLLLLLLLLLLLL	IIIIIIII	UUUUUUUUUUUUUU	UUUUUUUUUUUUUU	TTTT	LLLLLLLLLLLLLLLL

```

SSSSSSSS EEEEEEEEE TTTT TTTT I I I I MM MM EEEEEEEEE
SSSSSSSS EEEEEEEEE TTTT TTTT I I I I MM MM EEEEEEEEE
SS      EE          TT      TT      II      MMMM MMMM EE
SS      EE          TT      TT      II      MMMM MMMM EE
SS      EE          TT      TT      II      MM  MM  MM  EE
      SSSSSS      EEEEEEEE TT      TT      II      MM  MM  EE
      SSSSSS      EEEEEEEE TT      TT      II      MM  MM  EE
      SS          EE          TT      TT      II      MM  MM  EE
      SS          EE          TT      TT      II      MM  MM  EE
      SS          EE          TT      TT      II      MM  MM  EE
      SS          EE          TT      TT      II      MM  MM  EE
SSSSSSSS EEEEEEEEE TTTT TTTT I I I I MM MM EEEEEEEEE
SSSSSSSS EEEEEEEEE TTTT TTTT I I I I MM MM EEEEEEEEE

```

```

LL      I I I I SSSSSSS
LL      I I I I SSSSSSS
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
LLLLLLLLL I I I I SSSSSSS
LLLLLLLLL I I I I SSSSSSS

```

```

1 0001 0 MODULE settime ( IDENT = 'V04-000',
2 0002 0 ADDRESSING_MODE (EXTERNAL = GENERAL)) =
3 0003 1 BEGIN
4 0004 1
5 0005 1 |*****
6 0006 1 |*
7 0007 1 |* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
8 0008 1 |* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
9 0009 1 |* ALL RIGHTS RESERVED.
10 0010 1 |*
11 0011 1 |* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
12 0012 1 |* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
13 0013 1 |* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
14 0014 1 |* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
15 0015 1 |* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
16 0016 1 |* TRANSFERRED.
17 0017 1 |*
18 0018 1 |* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
19 0019 1 |* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
20 0020 1 |* CORPORATION.
21 0021 1 |*
22 0022 1 |* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
23 0023 1 |* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
24 0024 1 |*
25 0025 1 |*
26 0026 1 |*****
27 0027 1
28 0028 1 ++
29 0029 1 FACILITY: SETPRO Command
30 0030 1
31 0031 1 ABSTRACT:
32 0032 1
33 0033 1 This utility sets the system time.
34 0034 1
35 0035 1 ENVIRONMENT:
36 0036 1
37 0037 1 VAX/VMS operating system. Privileged user mode.
38 0038 1
39 0039 1 AUTHOR: Gerry Smith 6-Jan-1983
40 0040 1
41 0041 1 Modified by:
42 0042 1
43 0043 1 V03-001 GAS0112 29-Mar-1983
44 0044 1 Remove last traces of old command dispatcher.
45 0045 1
46 0046 1 --

```

SETTIME
V04-000

C 16
16-Sep-1984 01:01:27 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:09:21 [CLIUTL.SRC]SETTIME.B32;1

Page 2
(2)

: 48
: 49
: 50
: 51
0047 1 !
0048 1 ! Include files
0049 1 !
0050 1 LIBRARY 'SYS\$LIBRARY:STARLET';

! VAX/VMS common definitions

```
.. 53      0051  1  |  
.. 54      0052  1  | Table of contents  
.. 55      0053  1  |  
.. 56      0054  1  |  
.. 57      0055  1  | FORWARD ROUTINE  
.. 58      0056  1  |   set$time : NOVALUE;           . Routine to set the time  
.. 59      0057  1  |  
.. 60      0058  1  |  
.. 61      0059  1  | External routines  
.. 62      0060  1  |  
.. 63      0061  1  | EXTERNAL ROUTINE  
.. 64      0062  1  |   cli$get_value,               ! CLI routine to get the time  
.. 65      0063  1  |   lib$cvtime;                 ! Routine to convert to system time  
.. 66      0064  1  |  
.. 67      0065  1  |  
.. 68      0066  1  | External definitions  
.. 69      0067  1  |  
.. 70      0068  1  | EXTERNAL LITERAL  
.. 71      0069  1  |   set$_writeerr;              ! Error modifying...  
.. 72      0070  1  |  
.. 73      0071  1  |  
.. 74      0072  1  | Declare some shared messages  
.. 75      0073  1  |  
.. 76      P 0074  1  | $SHR_MSGDEF (SET,119,LOCAL,  
.. 77      P 0075  1  |              (invquaval, error),  
.. 78      0076  1  |              (valerr, error));  
.. 79      0077  1  |  
.. 80      0078  1  |
```

```

: 82 0079 1 GLOBAL ROUTINE set$time : NOVALUE =
: 83 0080 2 BEGIN
: 84 0081 2 +-
: 85 0082 2 | Functional description
: 86 0083 2 |
: 87 0084 2 |     This is the routine for the SET TIME command.
: 88 0085 2 |     It is called from the SET command processor.
: 89 0086 2 |
: 90 0087 2 | Inputs
: 91 0088 2 |     None
: 92 0089 2 |
: 93 0090 2 | Outputs
: 94 0091 2 |     None
: 95 0092 2 |
: 96 0093 2 | ----
: 97 0094 2 |
: 98 0095 2 LOCAL
: 99 0096 2     status,                | Status return
100 0097 2     time : VECTOR[2] INITIAL(0,0), | System time initially set to zero
101 0098 2     time_desc : $BLOCK[dsc$c_s_b[n]; | Descriptor for system time
102 0099 2
103 0100 2 |
104 0101 2 | First, get the time value by asking the CLI for it.
105 0102 2 |
106 0103 2 $init_dyndesc(time_desc);          | Make the time descriptor dynamic
107 0104 2 IF cli$get_value(%ASCII 'TIME',   | If there was a value given
108 0105 2     time_desc)                    | then convert it.
109 0106 2 THEN
110 0107 3     BEGIN
111 0108 4     IF NOT (status = lib$cvt_time(time_desc, time))
112 0109 3     THEN
113 0110 4         BEGIN
114 0111 4         SIGNAL(set$_writeerr, 1, %ASCII 'time', set$_valerr);
115 0112 4         RETURN;
116 0113 3         END;
117 0114 2     END;
118 0115 2 |
119 0116 2 | Set the time.
120 0117 2 |
121 0118 2 |
122 0119 3 IF NOT (status = $SETIME(TIMADR = time))
123 0120 2 THEN SIGNAL(set$_writeerr, 1, %ASCII 'time', .status);
124 0121 2 |
125 0122 2 RETURN;
126 0123 1 END;

```

```

.TITLE SETTIME
.IDENT \V04-000\

.PSECT $PLITS,NOWRT,NOEXE,2

```

```

45 4D 49 54 0000 P.AAB: .ASCII \TIME\
      010E0004 00004 P.AAA: .LONG 17694724
      00000000' 00008 .ADDRESS P.AAB
65 6D 69 74 0000 P.AAD: .ASCII \time\
      010E0004 00010 P.AAC: .LONG 17694724

```

```

00000000' 00014 .ADDRESS P.AAD
65 6D 69 74 00018 P.AAF: .ASCII \time\
010E0004 0001C P.AAE: .LONG 17694724
00000000' 00020 .ADDRESS P.AAF

.EXTRN CLISGET_VALUE, LIB$CVT_TIME
.EXTRN SET$WRITEERR, SYS$SETIME

.PSECT $CODE$,NOWRT,2

0004 0000 .ENTRY SET$TIME, Save R2
SE 04 OC C2 00002 SUBL2 #12, SP
020E0000 04 AE 7C 00005 CLRQ TIME
04 AE D4 0000E PUSHL #34471936
0000' SE DD 00011 CLRL TIME_DESC+4
0000000G 00 CF 9F 00013 PUSHL SP
1F 02 FB 00017 PUSHAB P.AAA
08 AE 9F 00021 CALLS #2, CLISGET_VALUE
04 AE 9F 00024 BLBC R0, 1$
0000000G 00 02 FB 00027 PUSHAB TIME
52 50 D0 0002E PUSHAB TIME_DESC
OC 52 E8 00031 CALLS #2, LIB$CVT_TIME
007711EA 8F DD 00034 MOVL R0, STATUS
0000' CF 9F 0003A BLBS STATUS, 1$
16 11 0003E PUSHL #7803370
08 AE 9F 00040 1$: PUSHAB P.AAC
0000000G 00 01 FB 00043 BRB 2$
52 50 D0 0004A PUSHAB TIME
15 52 E8 0004D CALLS #1, SYS$SETIME
0000' 52 DD 00050 MOVL R0, STATUS
01 DD 00052 2$: BLBS STATUS, 3$
0000000G 00 01 DD 00056 2$: PUSHL STATUS
8F DD 00058 2$: PUSHL P.AAE
04 FB 0005E 2$: PUSHL #1
04 00065 3$: PUSHL #SETS_WRITEERR
CALLS #4, LIB$SIGNAL
RET

```

; Routine Size: 102 bytes, Routine Base: \$CODE\$ + 0000

SETTIME
V04-000

G 16
16-Sep-1984 01:01:27
14-Sep-1984 12:09:21

VAX-11 Bliss-32 V4.0-742
[CLIUTL.SRC]SETTIME.B32;1

Page 6
(5)

: 128 0124 1 END
: 129 0125 0 ELUDOM

.EXTRN LIB\$SIGNAL

PSECT SUMMARY

Name	Bytes	Attributes
\$SPLITS	36	NOVEC,NOWRT, RD ,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$SCODES	102	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	----- Symbols -----		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	17 0	581	00:01.0

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS\$:SETTIME/OBJ=OBJ\$:SETTIME MSRCS\$:SETTIME/UPDATE=(ENHS\$:SETTIME)

: Size: 102 code + 36 data bytes
: Run Time: 00:03.6
: Elapsed Time: 00:13.6
: Lines/CPU Min: 2106
: Lexemes/CPU-Min: 12674
: Memory Used: 54 pages
: Compilation Complete

0054 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

