```
SSSSSSSSSSS   YYY       YYY   SSSSSSSSSSSS  LLL              00000000         AAAAAAAAA
SSSSSSSSSSS   YYY       YYY   SSSSSSSSSSSS  LLL             000000000        AAAAAAAAAA
SSSSSSSSSSS   YYY       YYY   SSSSSSSSSSSS  LLL             000000000        AAAAAAAAAA
SSS           YYY       YYY   SSS           LLL            000     000      AAA       AAA
SSS           YYY       YYY   SSS           LLL            000     000      AAA       AAA
SSS             YYY   YYY     SSS           LLL            000     000      AAA       AAA
SSS               YYY YYY     SSS           LLL            000     000      AAA       AAA
SSS               YYY YYY     SSS           LLL            000     000      AAA       AAA
   SSSSSSSSS        YYY          SSSSSSSSS   LLL            000     000      AAA       AAA
   SSSSSSSSS        YYY          SSSSSSSSS   LLL            000     000      AAA       AAA
   SSSSSSSSS        YYY          SSSSSSSSS   LLL            000     000      AAAAAAAAAAAAAAA
           SSS      YYY                 SSS  LLL            000     000      AAAAAAAAAAAAAAA
           SSS      YYY                 SSS  LLL            000     000      AAA       AAA
           SSS      YYY                 SSS  LLL            000     000      AAA       AAA
           SSS      YYY                 SSS  LLL            000     000      AAA       AAA
           SSS      YYY                 SSS  LLL            000     000      AAA       AAA
SSSSSSSSSSS         YYY       SSSSSSSSSSSS   LLLLLLLLLLLLLL  00000000        AAA       AAA
SSSSSSSSSSS         YYY       SSSSSSSSSSSS   LLLLLLLLLLLLLL  00000000        AAA       AAA
SSSSSSSSSSS         YYY       SSSSSSSSSSSS   LLLLLLLLLLLLLL  00000000        AAA       AAA
```

FILEID**UTILKEY

```
UU      UU  TTTTTTTTTT   IIIIII   LL          KK      KK  EEEEEEEEEE  YY      YY
UU      UU  TTTTTTTTTT   IIIIII   LL          KK      KK  EEEEEEEEEE  YY      YY
UU      UU      TT         II     LL          KK      KK  EE              YY  YY
UU      UU      TT         II     LL          KK      KK  EE              YY  YY
UU      UU      TT         II     LL          KK    KK    EE                YY YY
UU      UU      TT         II     LL          KK  KK      EE                YY YY
UU      UU      TT         II     LL          KKKKKK      EEEEEEE             YY
UU      UU      TT         II     LL          KKKKKK      EEEEEEE             YY
UU      UU      TT         II     LL          KK  KK      EE                  YY
UU      UU      TT         II     LL          KK    KK    EE                  YY
UU      UU      TT         II     LL          KK      KK  EE                  YY
UUUUUUUUUU      TT       IIIIII   LLLLLLLLLL  KK      KK  EEEEEEEEEE          YY    ::::
UUUUUUUUUU      TT       IIIIII   LLLLLLLLLL  KK      KK  EEEEEEEEEE          YY    ::::


LL          IIIIII   SSSSSSSS
LL          IIIIII   SSSSSSSS
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
```

_$25

Symbo

GETBU
GETFL
GEXIT
GOFSE
GONSE
GRPTE
HLDFL
HOMSE
HTSIZ
HTSIZ
INDIR
INITE
INPAR
INPNO
INPNT
INPSA
IO$B
IO$C
IO$M
IO$M
IO$M
IO$M
IO$M
IO$M
IO$V
IO$V
IO$V
IO$V
IO$_R
IO$_S
IO$_W
IOERR
IOERR
ITRCM
ITRSE
KILF
LBR$
LBR$
LBR$
LBR$
LBR$
LCLSI
LIB$
LIB$
LIB$
LIB$
LIB$
LIB$
LIB$
LIB$
LIB$
LIB$

```
0000        1              .TITLE  UTIL$RANDOM_KEY
0000        2              .IDENT  /V04-000/
0000        3      ;*******************************************************************
0000        4      ;*                                                                 *
0000        5      ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                        *
0000        6      ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.         *
0000        7      ;*  ALL RIGHTS RESERVED.                                           *
0000        8      ;*                                                                 *
0000        9      ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000       10      ;*  ONLY IN  ACCORDANCE WITH  THE   TERMS  OF  SUCH  LICENSE  AND WITH THE *
0000       11      ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
0000       12      ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
0000       13      ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY  *
0000       14      ;*  TRANSFERRED.                                                   *
0000       15      ;*                                                                 *
0000       16      ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
0000       17      ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
0000       18      ;*  CORPORATION.                                                   *
0000       19      ;*                                                                 *
0000       20      ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
0000       21      ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.         *
0000       22      ;*                                                                 *
0000       23      ;*                                                                 *
0000       24      ;*******************************************************************
0000       25
0000       26      ;++
0000       27
0000       28      ; FACILITY:     Random number generator.
0000       29
0000       30      ; ABSTRACT:     Produces a random longword.
0000       31
0000       32      ; AUTHOR:       Paul R. Beck
0000       33
0000       34      ; DATE:         19-APR-1983    Last Edit:  19-APR-1983 21:37:26
0000       35
0000       36      ; REVISION HISTORY:
0000       37
0000       38
0000       39      ;--
```

Symb

LINP
LIST
LSCH
M
MOFS
MONS
MPDL
MRKF
N
NACC
NFLG
NMRB
NOCT
NOPR
NP
NWAT
NWID
OFLG
OSCA
OUPA
OUPN
OUPN
OUTB
OUTC
OUTD
OUTS
P
PATC
PATS
PCNT
PDL
PDLS
PDLS
POST
PREC
PREL
PRIN
PRIN
PRIN
PRTL
PST
PUTB
QARR
QBAS
QCMN
QFLG
QLCM
QLEN
QMAX
QNMB
QPNT
QRST

```
                                 0000    41 ;
                                 0000    42 ; Local data
                                 0000    43 ;
              FFFFFFFF FFFE7960  0000    44 TENMS:  .LONG   -100*1000,-1          ; 10 ms timer
                                 0008    45 ;
                     0000000C    0008    46 KEY:    .BLKL   1                     ; build the key here
                     00000010    000C    47 DATA:   .BLKL   1                     ; Just increment this ad nauseum
                     00000014    0010    48 OFFSET: .BLKL   1                     ; Offset into KEY
                                 0014    49 ;++
                                 0014    50 ; UTIL$RANDOM_KEY
                                 0014    51 ;
                                 0014    52 ; FUNCTIONAL DESCRIPTION:
                                 0014    53 ;       Create a random, 32-bit key. This is done two bits at a time.
                                 0014    54 ; CALLING SEQUENCE:
                                 0014    55 ;       CALLS   #1,UTIL$RANDOM_KEY
                                 0014    56 ; INPUT PARAMETERS:
                                 0014    57 ;       None
                                 0014    58 ; OUTPUT PARAMETERS:
                                 0014    59 ;       P1      = address to receive key
                                 0014    60 ; COMPLETION CODES:
                                 0014    61 ;       SS$_NORMAL
                                 0014    62 ;--
                          0000   0014    63         .ENTRY  UTIL$RANDOM_KEY,^M<>
                                 0016    64
                     F3 AF  7C   0016    65         CLRQ    DATA                  ; set offset into key and counter
                                 0019    66         $SETIMR_S -
                                 0019    67                 daytim  = TENMS -     ; 10 ms timer
                                 0019    68                 astadr  = KEY_AST     ; address of timer AST
               19 50  E9         002D    69         BLBC    R0,20$                ; ?
               D9 AF  D6         0030    70 10$:    INCL    DATA                  ; *** LOOP ***
      FFFFFFFF 8F  DA AF  D1     0033    71         CMPL    OFFSET,#-1            ; done yet?
                    F3  12       003B    72         BNEQ    10$                   ; if NEQ, no.
        04 BC  C8 AF  D0         003D    73         MOVL    KEY,@4(AP)            ; return random key
     50 00000000'8F  D0          0042    74         MOVL    #SS$_NORMAL,R0        ; done
                    04           0049    75 20$:    RET
                                 004A    76
                                 004A    77 ; AST to collect the random key, two bits every 10 ms.
                                 004A    78 ; KEY contains address of the key being constructed
                                 004A    79 ; DATA contains the raw data (we just use the low two bits as random)
                                 004A    80 ; OFFSET contains the number of passes made *2 and offsets into the key.
                                 004A    81 ;
                          0000   004A    82         .ENTRY  KEY_AST,^M<>
                                 004C    83
            50  C1 AF  D0        004C    84         MOVL    OFFSET,R0
B1 AF  02  50  B9 AF  F0         0050    85         INSV    DATA,R0,#2,KEY        ; move next two bits into key
           B6 AF  D6             0057    86         INCL    OFFSET                ; adjust offset
      02 B2 AF  20  F2           005A    87         AOBLSS  #32,OFFSET,10$        ; ...and exit when we're done
                 12  11          005F    88         BRB     20$
                                 0061    89 10$:    $SETIMR_S -
                                 0061    90                 daytim  = TENMS -     ; 10 ms timer
                                 0061    91                 astadr  = KEY_AST     ; address of timer AST
                          04     0072    92         RET
   95 AF  FFFFFFFF 8F  D0        0073    93 20$:    MOVL    #-1,OFFSET            ; set flag and don't reissue AST
                          04     007B    94         RET
                                 007C    95 .END
```

```
UTIL$RANDOM_KEY                                16-SEP-1984 01:13:17  VAX/VMS Macro V04-00      Page   3
Symbol table                                    5-SEP-1984 04:11:31  [SYSLOA.SRC]UTILKEY.MAR;1            (2)

$$T1              = 00000000
DATA                0000000C R     01
KEY                 00000008 R     01
KEY_AST             0000004A RG    01
OFFSET              00000010 R     01
SS$_NORMAL          ******** X     01
SYS$SETIMR          ******** GX    01
TENMS               00000000 R     01
UTIL$RANDOM_KEY     00000014 RG    01
```

```
                                    +------------------+
                                    ! Psect synopsis !
                                    +------------------+


PSECT name                    Allocation          PSECT No.   Attributes
----------                    ----------          ---------   ----------
. ABS .                       00000000 (    0.)   00 (  0.)   NOPIC  USR   CON  ABS  LCL NOSHR NOEXE NORD  NOWRT NOVEC BYTE
. BLANK .                     0000007C (  124.)   01 (  1.)   NOPIC  USR   CON  REL  LCL NOSHR  EXE   RD    WRT  NOVEC BYTE
```

```
                            +----------------------------+
                            ! Performance indicators !
                            +----------------------------+


Phase                   Page faults   CPU Time       Elapsed Time
-----                   -----------   --------       ------------
Initialization                  35    00:00:00.04    00:00:01.44
Command processing             135    00:00:00.41    00:00:03.27
Pass 1                         107    00:00:00.47    00:00:05.35
Symbol table sort                0    00:00:00.00    00:00:00.00
Pass 2                          34    00:00:00.16    00:00:02.76
Symbol table output              2    00:00:00.02    00:00:00.01
Psect synopsis output            2    00:00:00.01    00:00:00.01
Cross-reference output           0    00:00:00.00    00:00:00.00
Assembler run totals           318    00:00:01.11    00:00:12.86
```

The working set limit was 1050 pages.
2146 bytes (5 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 9 non-local and 4 local symbols.
95 source lines were read in Pass 1, producing 16 object records in Pass 2.
4 pages of virtual memory were used to define 4 macros.

```
                            +----------------------------+
                            ! Macro library statistics !
                            +----------------------------+


Macro library name                      Macros defined
------------------                      --------------
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                0
_$255$DUA28:[SYSLIB]STARLET.MLB;2             4
TOTALS (all libraries)                        4
```

41 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:UTILKEY/OBJ=OBJ$:UTILKEY MSRC$:UTILKEY/UPDATE=(ENH$:UTILKEY)+EXECML$/LIB

UTILKEY
LIS

TECOLBR
LIS

TECOCMD
LIS

SCSLOA
LIS

TECO

TECO
MAP

UTILKEY
LIS

TECONAT
LIS