```
EEEEEEEEEEEEEEE   DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE  DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE  DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEEEEEEEEEEE      DDD        DDD          TTT
EEEEEEEEEEEE      DDD        DDD          TTT
EEEEEEEEEEEE      DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEE               DDD        DDD          TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDDD             TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDDD             TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDDD             TTT
```

```
MM      MM   CCCCCCC  LL         EEEEEEEEEE  FFFFFFFFFF  TTTTTTTTTT
MM      MM   CCCCCCC  LL         EEEEEEEEEE  FFFFFFFFFF  TTTTTTTTTT
MMMM  MMMM  CC        LL         EE          FF              TT
MMMM  MMMM  CC        LL         EE          FF              TT
MM  MM  MM  CC        LL         EE          FF              TT
MM      MM  CC        LL         EEEEEEE     FFFFFFF         TT
MM      MM  CC        LL         EEEEEEE     FFFFFFF         TT
MM      MM  CC        LL         EE          FF              TT
MM      MM  CC        LL         EE          FF              TT
MM      MM  CC        LL         EE          FF              TT        ....
MM      MM   CCCCCCC  LLLLLLLLL  EEEEEEEEEE  FF              TT        ::::
MM      MM   CCCCCCC  LLLLLLLLL  EEEEEEEEEE  FF              TT        ....


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
LLLLLLLLL      IIIIII     SSSSSSSS
LLLLLLLLL      IIIIII     SSSSSSSS
```

```
 1    0001   0  %TITLE 'EDT$MCLEFT - move cursor left'
 2    0002   0  MODULE EDT$MCLEFT (                              ! Move cursor left
 3    0003   0                    IDENT = 'V04-U00'                      ! File: MCLEFT.BLI Edit: JBS1003
 4    0004   0                    ) =
 5    0005   1  BEGIN
 6    0006   1
 7    0007   1  !*****************************************************************
 8    0008   1  !*                                                              *
 9    0009   1  !*    COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                   *
10    0010   1  !*    DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.    *
11    0011   1  !*    ALL RIGHTS RESERVED.                                      *
12    0012   1  !*                                                              *
13    0013   1  !*    THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
14    0014   1  !*    ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
15    0015   1  !*    INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
16    0016   1  !*    COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
17    0017   1  !*    OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
18    0018   1  !*    TRANSFERRED.                                              *
19    0019   1  !*                                                              *
20    0020   1  !*    THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
21    0021   1  !*    AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
22    0022   1  !*    CORPORATION.                                              *
23    0023   1  !*                                                              *
24    0024   1  !*    DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
25    0025   1  !*    SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.   *
26    0026   1  !*                                                              *
27    0027   1  !*                                                              *
28    0028   1  !*****************************************************************
29    0029   1  !
30    0030   1
31    0031   1  !++
32    0032   1  ! FACILITY:     EDT -- The DEC Standard Editor
33    0033   1  !
34    0034   1  ! ABSTRACT:
35    0035   1  !
36    0036   1  !       Move cursor left.
37    0037   1  !
38    0038   1  ! ENVIRONMENT:  Runs at any access mode - AST reentrant
39    0039   1  !
40    0040   1  ! AUTHOR: Bob Kushlis, CREATION DATE: September 9, 1979
41    0041   1  !
42    0042   1  ! MODIFIED BY:
43    0043   1  !
44    0044   1  ! 1-001 - Original.  DJS 19-FEB-1981.  This module was created by
45    0045   1  !         extracting routine EDT$$CS_LEFT  from module MOVCUR.
46    0046   1  ! 1-002 - Regularize headers.  JBS 11-Mar-1981
47    0047   1  ! 1-003 - Improve the appearance of the listing.  JBS 17-Jun-1983
48    0048   1  !--
49    0049   1
```

EDT$MCLEFT   EDT$MCLEFT - move cursor left   16-Sep-1984 01:11:02   VAX-11 Bliss-32 V4.0-742   Page 2
V04-000   Declarations   14-Sep-1984 12:23:55   DISK$VMSMASTER:[EDT.SRC]MCLEFT.BLI;1   (2)

K 9

```
51      0050  1 %SBTTL 'Declarations'
52      0051  1 !
53      0052  1 !  TABLE OF CONTENTS:
54      0053  1 !
55      0054  1
56      0055  1 REQUIRE 'EDTSRC:TRAROUNAM';
57      0494  1
58      0495  1 FORWARD ROUTINE
59      0496  1     EDT$$CS_LEFT;
60      0497  1
61      0498  1 !
62      0499  1 !  INCLUDE FILES:
63      0500  1 !
64      0501  1
65      0502  1 REQUIRE 'EDTSRC:EDTREQ';
66      0637  1
67      0638  1 !
68      0639  1 !  MACROS:
69      0640  1 !
70      0641  1 !       NONE
71      0642  1 !
72      0643  1 !  EQUATED SYMBOLS:
73      0644  1 !
74      0645  1 !       NONE
75      0646  1 !
76      0647  1 !  OWN STORAGE:
77      0648  1 !
78      0649  1 !       NONE
79      0650  1 !
80      0651  1 !  EXTERNAL REFERENCES:
81      0652  1 !
82      0653  1 !       In the routine
```

```
    84    0654  1  %SBTTL 'EDT$$CS_LEFT  - move cursor left'
    85    0655  1
    86    0656  1  GLOBAL ROUTINE EDT$$CS_LEFT                            ! Move cursor left
    87    0657  1      =
    88    0658  1
    89    0659  1  !++
    90    0660  1  ! FUNCTIONAL DESCRIPTION:
    91    0661  1  !
    92    0662  1  !       This routine moves the cursor one character to the left.  If the cur or
    93    0663  1  !       is on the first character of the line, we move to the previous line
    94    0664  1  !       (positioning on the line terminator).
    95    0665  1  !
    96    0666  1  ! FORMAL PARAMETERS:
    97    0667  1  !
    98    0668  1  !       NONE
    99    0669  1  !
   100    0670  1  ! IMPLICIT INPUTS:
   101    0671  1  !
   102    0672  1  !       EDT$$T_LN_BUF
   103    0673  1  !       EDT$$A_LN_END
   104    0674  1  !
   105    0675  1  ! IMPLICIT OUTPUTS:
   106    0676  1  !
   107    0677  1  !       EDT$$A_LN_PTR
   108    0678  1  !
   109    0679  1  ! ROUTINE VALUE:
   110    0680  1  !
   111    0681  1  !       1 = move successful
   112    0682  1  !       0 = could not move
   113    0683  1  !
   114    0684  1  ! SIDE EFFECTS:
   115    0685  1  !
   116    0686  1  !       NONE
   117    0687  1  !
   118    0688  1  !--
   119    0689  1
   120    0690  2      BEGIN
   121    0691  2
   122    0692  2      EXTERNAL ROUTINE
   123    0693  2          EDT$$CS_UP;
   124    0694  2
   125    0695  2      EXTERNAL
   126    0696  2          EDT$$T_LN_BUF,                            ! Current line buffer.
   127    0697  2          EDT$$A_LN_PTR,                            ! Current character pointer.
   128    0698  2          EDT$$A_LN_END;                            ! End of current line pointer.
   129    0699  2
   130    0700  2  !+
   131    0701  2  ! Check to see if we are on the first character.
   132    0702  2  !-
   133    0703  2
   134    0704  3      IF CH$PTR_EQL (.EDT$$A_LN_PTR, CH$PTR (EDT$$T_LN_BUF))
   135    0705  2      THEN
   136    0706  3          BEGIN
   137    0707  3  !+
   138    0708  3  ! On the first character.  Try to go up a line and position to the line
   139    0709  3  ! terminator.  If the move up fails, return failure.
   140    0710  3  !-
```

```
:  141        0711  3
:  142        0712  3              IF (EDT$$CS_UP () EQL 0) THEN RETURN (0);
:  143        0713  3
:  144        0714  3              EDT$$A_LN_PTR = .EDT$$A_LN_END;
:  145        0715  3              END
:  146        0716  2          ELSE
:  147        0717  2      !+
:  148        0718  2      ! Not on the first character, simply move back one character and return
:  149        0719  2      ! success.
:  150        0720  2      !-
:  151        0721  2              EDT$$A_LN_PTR = CH$PLUS (.EDT$$A_LN_PTR, -1);
:  152        0722  2
:  153        0723  2          RETURN (1);
:  154        0724  1          END;                                        ! of routine EDT$$CS_LEFT


                                                   .TITLE   EDT$MCLEFT EDT$MCLEFT - move cursor left
                                                   .IDENT   \V04-000\

                                                   .EXTRN   EDT$$CS_UP, EDT$$T_LN_BUF
                                                   .EXTRN   EDT$$A_LN_PTR, EDT$$A_LN_END

                                                   .PSECT   _EDT$CODE,NOWRT,  SHR,  PIC,2

                                 0004 00000        .ENTRY   EDT$$CS_LEFT, Save R2              :  0656
             52 00000000G  00  9E 00002            MOVAB    EDT$$A_LN_PTR, R2
             50 00000000G  00  9E 00009            MOVAB    EDT$$T_LN_BUF, R0                  :  0704
             50            62  D1 00010            CMPL     EDT$$A_LN_PTR, R0
                           14  12 00013            BNEQ     1$
   00000000G 00            00  FB 00015            CALLS    #0, EDT$$CS_UP                     :  0712
                           50  D5 0001C            TSTL     R0
                           0F  13 0001E            BEQL     3$
             62 00000000G  00  D0 00020            MOVL     EDT$$A_LN_END, EDT$$A_LN_PTR       :  0714
                           02  11 00027            BRB      2$                                :  0704
                           62  D7 00029 1$:        DECL     EDT$$A_LN_PTR                      :  0721
             50            01  D0 0002B 2$:        MOVL     #1, R0                            :  0723
                           04 0002E               RET
                           50  D4 0002F 3$:        CLRL     R0                                :  0724
                           04 00031               RET

; Routine Size:  50 bytes,    Routine Base:  _EDT$CODE + 0000


:  155        0725  1
:  156        0726  1 !<BLF/PAGE>
```

EDTSMCLEFT   EDTSMCLEFT - move cursor left   N 9
                                       16-Sep-1984 01:11:02   VAX-11 Bliss-32 V4.0-742   Page 5
V04-000   EDTSSCS_LEFT - move cursor left   14-Sep-1984 12:23:55   DISKSVMSMASTER:[EDT.SRC]MCLEFT.BLI;1 (4)

```
;  158        0727 1 END                              ! of module EDTSMCLEFT
;  159        0728 1
;  160        0729 0 ELUDOM
```

PSECT SUMMARY

Name	Bytes	Attributes
_EDTSCODE	50	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	-------- Symbols -------- Total	Loaded	Percent	Pages Mapped	Processing Time
_255DUA28:[EDT.SRC]EDT.L32;1	377	1	0	40	00:00.2
_255DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1

COMMAND QUALIFIERS

    BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:MCLEFT/OBJ=OBJ$:MCLEFT MSRC$:MCLEFT.BLI/UPDATE=(ENH$:MCLEFT)

```
; Size:         50 code + 0 data bytes
; Run Time:         00:09.5
; Elapsed Time:     00:12.0
; Lines/CPU Min:     4599
; Lexemes/CPU-Min: 12277
; Memory Used:  55 pages
; Compilation Complete
```

LXCOMNOO
LIS

MCGETLIN
LIS

MCRIGHT
LIS

MCCHANGE
LIS

MACCAL
LIS

NOOPEN
LIS

PRAPPNUM
LIS

MSGTXT
LIS

PRGETTOK
LIS

PRISTOK
LIS

LXPRINT
LIS

MCTOP
LIS

MCDOWN
LIS

MCLEFT
LIS

MCBOTTOM
LIS

PAUDIT
LIS

PRGETCHR
LIS

LXCOM
LIS

MAIN
LIS

MCUP
LIS