```
EEEEEEEEEEEEEEEE  DDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE  DDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE  DDDDDDDDDD      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
EEEEEEEEEEEEEEEE  DDDDDDDDDD             TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDD             TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDD             TTT
```

**FILE**ID**SCRINSERT

; Ro

: 1
: 1

....
....
....
....

```
SSSSSSSS   CCCCCCC  RRRRRRR   IIIIII   NN     NN  SSSSSSSS  EEEEEEEEEE  RRRRRRR   TTTTTTTTTT
SSSSSSSS   CCCCCCC  RRRRRRR   IIIIII   NN     NN  SSSSSSSS  EEEEEEEEEE  RRRRRRR   TTTTTTTTTT
SS         CC       RR    RR    II     NN     NN  SS        EE          RR    RR      TT
SS         CC       RR    RR    II     NN     NN  SS        EE          RR    RR      TT
SS         CC       RR    RR    II     NNNN   NN  SS        EE          RR    RR      TT
SS         CC       RR    RR    II     NNNN   NN  SS        EE          RR    RR      TT
 SSSSSS    CC       RRRRRRR     II     NN NN  NN   SSSSSS   EEEEEEE     RRRRRRR       TT
 SSSSSS    CC       RRRRRRR     II     NN NN  NN   SSSSSS   EEEEEEE     RRRRRRR       TT
     SS    CC       RR  RR      II     NN   NNNN       SS   EE          RR  RR        TT
     SS    CC       RR  RR      II     NN   NNNN       SS   EE          RR  RR        TT
     SS    CC       RR   RR     II     NN     NN       SS   EE          RR   RR       TT
     SS    CC       RR    RR    II     NN     NN       SS   EE          RR    RR      TT
SSSSSSSS   CCCCCCC  RR    RR   IIIIII  NN     NN  SSSSSSSS  EEEEEEEEEE   RR    RR      TT
SSSSSSSS   CCCCCCC  RR    RR   IIIIII  NN     NN  SSSSSSSS  EEEEEEEEEE   RR    RR      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
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```

```
0001  0  %TITLE 'EDT$SCRINSERT - insert a line on the screen'
0002  0  MODULE EDT$SCRINSERT (                                    ! Insert a line on the screen
0003  0                  IDENT = 'V04-000'                         ! File: SCRINSERT.BLI Edit: JBS1022
0004  0                  ) =
0005  1  BEGIN
0006  1
0007  1  !******************************************************************
0008  1  !*                                                                *
0009  1  !*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                      *
0010  1  !*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.       *
0011  1  !*   ALL RIGHTS RESERVED.                                         *
0012  1  !*                                                                *
0013  1  !*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
0014  1  !*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH LICENSE  AND WITH THE   *
0015  1  !*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
0016  1  !*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
0017  1  !*   OTHER PERSON  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0018  1  !*   TRANSFERRED.                                                 *
0019  1  !*                                                                *
0020  1  !*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
0021  1  !*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
0022  1  !*   CORPORATION.                                                 *
0023  1  !*                                                                *
0024  1  !*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
0025  1  !*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.      *
0026  1  !*                                                                *
0027  1  !*                                                                *
0028  1  !******************************************************************
0029  1  !
0030  1
0031  1  !++
0032  1  ! FACILITY:      EDT -- The DEC Standard Editor
0033  1  !
0034  1  ! ABSTRACT:
0035  1  !
0036  1  !       This module updates the screen information vector to
0037  1  !       reflect the insertion of a line.
0038  1  !
0039  1  ! ENVIRONMENT:  Runs at any access mode - AST reentrant
0040  1  !
0041  1  ! AUTHOR: Bob Kushlis, CREATION DATE: September 8, 1979
0042  1  !
0043  1  ! MODIFIED BY:
0044  1  !
0045  1  ! 1-001 - Original.  DJS 12-Feb-1981.  This module was created by
0046  1  !         extracting the routine EDT$$SC_INSLN  from module SCREEN.
0047  1  ! 1-002 - Regularize headers.  JBS 13-Mar-1981
0048  1  ! 1-003 - Repaint when a line is inserted above the top of the screen,
0049  1  !         but not when a line is inserted at the top of the screen.
0050  1  !         JBS 24-Sep-1981
0051  1  ! 1-004 - Check a new flag for scrolling regions.  JBS 31-Aug-1982
0052  1  ! 1-005 - Modify for the new screen updater.  SMB 17-Sep-1982
0053  1  ! 1-006 - Modify to use new 48 bit macros. STS 01-Oct-1982
0054  1  ! 1-007 - Take out references to L_CUR_SCRLN.  SMB 04-Oct-1982
0055  1  ! 1-008 - Reduce parameters to SC_[NINS.  SMB 06-Oct-1982
0056  1  ! 1-009 - Change the way SC_FNDREC is used.  SMB 08-Oct-1982
0057  1  ! 1-010 - Change EDT$$G_SCR_CHGD to EDT$$G_SCR_REBUILD.  JBS 09-Oct-1982
```

```
:    58          0058  1 !  1-011 - Set the current screen pointer when a line is inserted.  SMB 10-Oct-1982
:    59          0059  1 !  1-012 - Add support for NOTRUNCATE mode.  JBS 11-Oct-1982
:    60          0060  1 !  1-013 - Update L_CUR_SCRLN on insert.  SMB 13-Oct-1982
:    61          0061  1 !  1-014 - Don't rebuild the screen after deleting a line.  JBS 23-Oct-1982
:    62          0062  1 !  1-015 - Don't increment the record number if our record was replaced
:    63          0063  1 !          rather than inserted before.  JBS 25-Oct-1982
:    64          0064  1 !  1-016 - Check for replacing a multi-line record.  JBS 26-Oct-1982
:    65          0065  1 !  1-017 - Fix notruncate bug on searching for deleted lines.  SMB 02-Dec-1982
:    66          0066  1 !  1-018 - Remove the edit buffer.  JBS 27-Dec-1982
:    67          0067  1 !  1-019 - Remove the unused reference to EDT$$G_CS_CHNO.  JBS 27-Dec-1982
:    68          0068  1 !  1-020 - Support EDT$$G_ANY_CHANGES.  JBS 05-Apr-1983
:    69          0069  1 !  1-021 - Worry about replacing the current line when the cursor is on
:    70          0070  1 !          a continuation line of it.  JBS 06-Jul-1983
:    71          0071  1 !  1-022 - Maintain EDT$$L_CUR_SCRLN even if the current line is outside of
:    72          0072  1 !          the screen data base.  JBS 25-Jul-1983
:    73          0073  1 !--
:    74          0074  1
```

```
  76    0075  1  %SBTTL 'Declarations'
  77    0076  1  !
  78    0077  1  ! TABLE OF CONTENTS:
  79    0078  1  !
  80    0079  1
  81    0080  1  REQUIRE 'EDTSRC:TRAROUNAM';
  82    0519  1
  83    0520  1  FORWARD ROUTINE
  84    0521  1      EDT$$SC_INSLN : NOVALUE;
  85    0522  1
  86    0523  1  !
  87    0524  1  ! INCLUDE FILES:
  88    0525  1  !
  89    0526  1
  90    0527  1  REQUIRE 'EDTSRC:EDTREQ';
  91    0662  1
  92    0663  1  !
  93    0664  1  ! MACROS:
  94    0665  1  !
  95    0666  1  !      NONE
  96    0667  1  !
  97    0668  1  ! OWN STORAGE:
  98    0669  1  !
  99    0670  1  !      NONE
 100    0671  1  !
 101    0672  1  ! EXTERNAL REFERENCES:
 102    0673  1  !
 103    0674  1  !      In the routine
```

```
105   0675  1  %SBTTL 'EDT$$SC_INSLN  - mark a line for insert'
106   0676  1
107   0677  1  GLOBAL ROUTINE EDT$$SC_INSLN (                      ! Mark a work file line for insert
108   0678  1      REC_ADDR,                                       ! Address of new line
109   0679  1      REC_LEN                                         ! Length of new line
110   0680  1      ) : NOVALUE =
111   0681  1
112   0682  1  !++
113   0683  1  ! FUNCTIONAL DESCRIPTION:
114   0684  1  !
115   0685  1  !      A line has been inserted.  Mark the line for insert in the screen
116   0686  1  !      information structure.
117   0687  1  !
118   0688  1  ! FORMAL PARAMETERS:
119   0689  1  !
120   0690  1  !  REC_ADDR              the address of the new line
121   0691  1  !
122   0692  1  !  REC_LEN               the length of the new line
123   0693  1  !
124   0694  1  ! IMPLICIT INPUTS:
125   0695  1  !
126   0696  1  !      EDT$$A_LN_PTR
127   0697  1  !      EDT$$T_LN_BUF
128   0698  1  !      EDT$$A_SCR_BUF
129   0699  1  !      EDT$$A_CUR_BUF
130   0700  1  !
131   0701  1  ! IMPLICIT OUTPUTS:
132   0702  1  !
133   0703  1  !      EDT$$L_CUR_SCRLN
134   0704  1  !      EDT$$A_CSR_SCRPTR
135   0705  1  !      EDT$$G_ANY_CHANGES
136   0706  1  !
137   0707  1  ! ROUTINE VALUE:
138   0708  1  !
139   0709  1  !      NONE
140   0710  1  !
141   0711  1  ! SIDE EFFECTS:
142   0712  1  !
143   0713  1  !      NONE
144   0714  1  !
145   0715  1  !--
146   0716  1
147   0717  2      BEGIN
148   0718  2
149   0719  2      EXTERNAL ROUTINE
150   0720  2          EDT$$SC_LNINS,                              ! Insert a line into sceen structure
151   0721  2          EDT$$SC_FNDREC;                             ! Find a record in the data structure
152   0722  2
153   0723  2      EXTERNAL
154   0724  2          EDT$$A_LN_PTR,
155   0725  2          EDT$$L_CUR_SCRLN : LN_BLOCK,                ! Current screen line
156   0726  2          EDT$$T_LN_BUF,                             ! Workfile line buffer
157   0727  2          EDT$$A_CSR_SCRPTR : REF SCREEN_LINE,       ! Pointer to current line screen info
158   0728  2          EDT$$A_SCR_BUF : REF TBCB_BLOCK,           ! Current screen block
159   0729  2          EDT$$A_CUR_BUF : REF TBCB_BLOCK,           ! Current text buffer control block'
160   0730  2          EDT$$G_ANY_CHANGES;                        ! 1 = a change has been mode to the screen data structure
161   0731  2
```

```
  162    0732   2         LOCAL
  163    0733   2             CURSOR_LINE,                                    ! Relative cursor line
  164    0734   2             CURSOR_POS,                                     ! Cursor position on the line
  165    0735   2             SCRPTR : REF SCREEN_LINE,                       ! Current line screen info
  166    0736   2             CURSOR_DELETED,                                 ! 1 = the cursor line had been deleted
  167    0737   2             DISP,                                           ! Displacement from cursor line
  168    0738   2             SCRLN_POS;                                      ! Our position relative to the screen line
  169    0739   2
  170    0740   2     !+
  171    0741   2     ! Only perform this routine if the current buffer is on the screen and the screen data structure exists.
  172    0742   2     !-
  173    0743   2
  174    0744   2         IF (.EDT$$A_SCR_BUF NEQA .EDT$$A_CUR_BUF) THEN RETURN;
  175    0745   2
  176    0746   2     !+
  177    0747   2     ! Compute our position relative to the screen line.
  178    0748   2     !-
  179    0749   2         SCRLN_POS = CMPLNO (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_CUR_SCRLN);
  180    0750   2     !+
  181    0751   2     ! Get the pointer which corresponds to the current record.  The insert will be just
  182    0752   2     ! before that record.
  183    0753   2     !-
  184    0754   2         SCRPTR = EDT$$SC_FNDREC (.EDT$$A_LN_PTR - EDT$$T_LN_BUF, DISP);
  185    0755   2     !+
  186    0756   2     ! We will do the insert above the current screen information block.
  187    0757   2     ! If there is a deleted line before the current screen information block
  188    0758   2     ! we will replace it.  If that line is the cursor line, don't increment
  189    0759   2     ! the record number of the cursor line.
  190    0760   2     !-
  191    0761   2
  192    0762   3         IF (.SCRPTR NEQA 0)
  193    0763   2         THEN
  194    0764   3             BEGIN
  195    0765   3     !+
  196    0766   3     ! Tell the screen updater to search the screen data structure for any changes.
  197    0767   3     !-
  198    0768   3             EDT$$G_ANY_CHANGES = 1;
  199    0769   3     !+
  200    0770   3     ! See if the cursor line has already been deleted.
  201    0771   3     !-
  202    0772   3
  203    0773   4             IF ((.EDT$$A_CSR_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)
  204    0774   3             THEN
  205    0775   3                 CURSOR_DELETED = 1
  206    0776   3             ELSE
  207    0777   3                 CURSOR_DELETED = 0;
  208    0778   3
  209    0779   3     !+
  210    0780   3     ! Now insert just before the current line.
  211    0781   3     !-
  212    0782   3             EDT$$SC_LNINS (.SCRPTR, .REC_ADDR, .REC_LEN);
  213    0783   3     !+
  214    0784   3     ! If we just replaced the current screen line, don't increment it's record number.
  215    0785   3     !-
  216    0786   3
  217    0787   4             IF ( NOT (.CURSOR_DELETED AND ((.EDT$$A_CSR_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0)))
  218    0788   3             THEN
```

```
 219    0789  4              BEGIN
 220    0790  4
 221    0791  4                  IF (.SCRLN_POS LEQ 0) THEN ADDLINE (NUMBER_ONE, EDT$$L_CUR_SCRLN);
 222    0792  4
 223    0793  3              END;
 224    0794  3
 225    0795  3          END
 226    0796  2      ELSE
 227    0797  2  !+
 228    0798  2  ! If the line to be inserted is in the buffer being displayed on the screen but outside
 229    0799  2  ! the screen data base we must still maintain EDT$$L_CUR_SCRLN.
 230    0800  2  !-
 231    0801  3          BEGIN
 232    0802  3
 233    0803  3              IF (.SCRLN_POS LEQ 0) THEN ADDLINE (NUMBER_ONE, EDT$$L_CUR_SCRLN);
 234    0804  3
 235    0805  2          END;
 236    0806  2
 237    0807  1      END;                                    ! of routine EDT$$SC_INSLN


                                               .TITLE   EDT$SCRINSERT EDT$SCRINSERT - insert a line on
                                                                 the screen
                                               .IDENT   \V04-000\

                                               .EXTRN   EDT$$SC_LNINS, EDT$$SC_FNDREC
                                               .EXTRN   EDT$$A_LN_PTR, EDT$$L_CUR_SCRLN
                                               .EXTRN   EDT$$T_LN_BUF, EDT$$A_CSR_SCRPTR
                                               .EXTRN   EDT$$A_SCR_BUF, EDT$$A_CUR_BUF
                                               .EXTRN   EDT$$G_ANY_CHANGES

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

                         003C 00000            .ENTRY   EDT$$SC_INSLN, Save R2,R3,R4,R5
         55 00000000G 00   9E 00002            MOVAB    EDT$$A_CSR_SCRPTR, R5
         54 00000000G 00   9E 00009            MOVAB    HIGH_2, R4
         5E          04   C2 00010             SUBL2    #4, SP
         50 00000000G 00   D0 00013            MOVL     EDT$$A_CUR_BUF, R0
         50 00000000G 00   D1 0001A            CMPL     EDT$$A_SCR_BUF, R0
                     7D   12 00021             BNEQ     10$
         51          64   3C 00023             MOVZWL   HIGH_2, R1
         51       0A A0   B1 00026             CMPW     10(R0), R1
                     0C   1F 0002A             BLSSU    1$
                     15   12 0002C             BNEQ     3$
         51       FC A4   D0 0002E             MOVL     LOW_2, R1
         51       06 A0   D1 00032             CMPL     6(R0), R1
                     05   1E 00036             BGEQU    2$
         52          01   CE 00038 1$:         MNEGL    #1, R2
                     09   11 0003B             BRB      4$
                     04   12 0003D 2$:         BNEQ     3$
                     52   D4 0003F             CLRL     R2
                     03   11 00041             BRB      4$
         52          01   D0 00043 3$:         MOVL     #1, R2
         5E          DD 00046 4$:              PUSHL    SP
         50 00000000G 00   9E 00048            MOVAB    EDT$$T_LN_BUF, R0
   7E 00000000G 00        50   C3 0004F        SUBL3    R0, EDT$$A_LN_PTR, -(SP)
      00000000G 00         02   FB 00057       CALLS    #2, EDT$$SC_FNDREC
```

EDT$

V04-

: 0677

: 0744

: 0749

: Ro

: 1
: 1

: 0754

```
                                                    L 2

                                        50 D5 0005E        TSTL    SCRPTR                              . 0762
                                        33 13 00060        BEQL    8$
                         00000000G 00   01 D0 00062        MOVL    #1, EDT$$G_ANY_CHANGES              . 0768
                                   51   65 D0 00069        MOVL    EDT$$A_CSR_SCRPTR, R1              . 0773
                    05        0D   A1   02 E1 0006C        BBC     #2, 13(R1), 5$
                                   53   01 D0 00071        MOVL    #1, CURSOR_DELETED                  . 0775
                                        02 11 00074        BRB     6$
                                        53 D4 00076 5$:    CLRL    CURSOR_DELETED                     . 0777
                              7E   04 AC 7D 00078 6$:      MOVQ    REC_ADDR, -(SP)                    . 0782
                                   50 DD 0007C             PUSHL   SCRPTR
                         00000000G 00   03 FB 0007E        CALLS   #3, EDT$$SC_LNINS
                                   08   53 E9 00085        BLBC    CURSOR_DELETED, 7$                 . 0787
                                   50   65 D0 00088        MOVL    EDT$$A_CSR_SCRPTR, R0
                    10        0D   A0   02 E1 0008B        BBC     #2, 13(R0), 10$
                                        52 D5 00090 7$:    TSTL    SCRLN_POS                          . 0791
                                        05 15 00092        BLEQ    9$
                                        04 00094           RET
                                        52 D5 00095 8$:    TSTL    SCRLN_POS                          . 0803
                                        07 14 00097        BGTR    10$
                              FC   A4 D6 00099 9$:         INCL    FIRST_LWORD
                                        02 12 0009C        BNEQ    10$
                                        64 B6 0009E        INCW    NEXT_WORD
                                        04 000A0 10$:      RET                                        . 0807

; Routine Size: 161 bytes,    Routine Base: _EDT$CODE + 0000


;  238          0808  1
;  239          0809  1 !<BLF/PAGE>
```

```
; 241   0810  1 END                                    ! of module EDT$SCRINSERT
; 242   0811  1
; 243   0812  0 ELUDOM
```

## PSECT SUMMARY

| Name | Bytes | Attributes |
|------|-------|------------|
| _EDT$CODE | 161 | NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2) |

## Library Statistics

| File | -------- Symbols -------- Total | Loaded | Percent | Pages Mapped | Processing Time |
|------|------|------|------|------|------|
| _$255$DUA28:[EDT.SRC]EDT.L32;1 | 377 | 44 | 11 | 40 | 00:00.2 |
| _$255$DUA28:[EDT.SRC]PSECTS.L32;1 | 2 | 1 | 50 | 7 | 00:00.1 |

## COMMAND QUALIFIERS

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

```
; Size:          161 code + 0 data bytes
; Run Time:        00:15.2
; Elapsed Time:    00:19.9
; Lines/CPU Min:     3205
; Lexemes/CPU-Min: 12635
; Memory Used:  114 pages
; Compilation Complete
```

SCRRESET
LIS

SCRRLIN
LIS

SCRRELPOS
LIS

SCRINSERT
LIS

SCRNOSCR
LIS

SCRNCOL
LIS

SCRMCOL
LIS

SCRUPDATE
LIS

SCRINIT
LIS

SCRNEWINS
LIS

SCRREGION
LIS

SCRREPMOD
LIS

SCRREV
LIS

SCRNEWDEL
LIS

SCRNOREV
LIS

SCRMOVETO
LIS

SCRINSMOD
LIS