```
EEEEEEEEEEEEEEE  DDDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    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
EEEEEEEEEEEEEEE  DDDDDDDDDDD          TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD         TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD         TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD         TTT
```

FILEID**SCRUPDATE

```
  SSSSSSSS    CCCCCCCC  RRRRRRRR    UU      UU  PPPPPPPP    DDDDDDDD      AAAAAA    TTTTTTTTTT  EEEEEEEEEE
  SSSSSSSS    CCCCCCCC  RRRRRRRR    UU      UU  PPPPPPPP    DDDDDDDD      AAAAAA    TTTTTTTTTT  EEEEEEEEEE
SS          CC          RR      RR  UU      UU  PP      PP  DD      DD  AA      AA      TT      EE
SS          CC          RR      RR  UU      UU  PP      PP  DD      DD  AA      AA      TT      EE
SS          CC          RR      RR  UU      UU  PP      PP  DD      DD  AA      AA      TT      EE
SS          CC          RR      RR  UU      UU  PP      PP  DD      DD  AA      AA      TT      EE
  SSSSSS    CC          RRRRRRRR    UU      UU  PPPPPPPP    DD      DD  AA      AA      TT      EEEEEEEE
  SSSSSS    CC          RRRRRRRR    UU      UU  PPPPPPPP    DD      DD  AA      AA      TT      EEEEEEEE
        SS  CC          RR  RR      UU      UU  PP          DD      DD  AAAAAAAAAA      TT      EE
        SS  CC          RR  RR      UU      UU  PP          DD      DD  AAAAAAAAAA      TT      EE
        SS  CC          RR    RR    UU      UU  PP          DD      DD  AA      AA      TT      EE                  ....
        SS  CC          RR    RR    UU      UU  PP          DD      DD  AA      AA      TT      EE                  ....
SSSSSSSS      CCCCCCCC  RR      RR  UUUUUUUUUU  PP          DDDDDDDD    AA      AA      TT      EEEEEEEEEE          ....
SSSSSSSS      CCCCCCCC  RR      RR  UUUUUUUUUU  PP          DDDDDDDD    AA      AA      TT      EEEEEEEEEE          ....


LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
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
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

EDT
V04

```
;  1          0001  0 %TITLE 'EDT$SCRUPDATE - update the screen'
;  2          0002  0 MODULE EDT$SCRUPDATE (                          ! Update the screen
;  3          0003  0                 IDENT = 'V04-000'                       ! File: SCRUPDATE.BLI Edit: JBS1080
;  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 !       This module does a screen update.
; 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 8, 1979
; 41          0041  1 !
; 42          0042  1 ! MODIFIED BY:
; 43          0043  1 !
; 44          0044  1 ! 1-001 - Original.  DJS 12-Feb-1981.  This module was created by
; 45          0045  1 !       extracting the routine EDT$$SC_UPD  from module SCREEN.
; 46          0046  1 ! 1-002 - Regularize headers.  JBS 13-Mar-1981
; 47          0047  1 ! 1-003 - Make sure the [EOB] test is valid when scrolling backwards.
; 48          0048  1 !         JBS 17-Sep-1981
; 49          0049  1 ! 1-004 - Revise autorepeat subroutine call.  JBS 30-Jan-1982
; 50          0050  1 ! 1-005 - Correct some spelling errors in comments.  JBS 02-Apr-1982
; 51          0051  1 ! 1-006 - Use new flag for scrolling logic.  JBS 02-Sep-1982
; 52          0052  1 ! 1-007 - Use the new screen structure and logic.  SMB 21-Sep-1982
; 53          0053  1 ! 1-008 - Remove unused external declaration of EDT$$FMT_LIT.  JBS 05-Oct-1982
; 54          0054  1 ! 1-009 - More debugging of screen scrolling and select.  SMB 08-Oct-1982
; 55          0055  1 ! 1-010 - Debug NOTRUNCATE mode.  JBS 12-Oct-1982
; 56          0056  1 ! 1-011 - Add insert and delete scrolling.  SMB 13-Oct-1982
; 57          0057  1 ! 1-012 - Clear EDT$$G_RECS_INSERTED.  JBS 21-Oct-1982
```

```
;    58          0058  1 ! 1-013 - Scrolling debugging.  SMB 21-Oct-1982
;    59          0059  1 ! 1-014 - Move the code for marking select changes for repaint.  JBS 23-Oct-1982
;    60          0060  1 ! 1-015 - Simplify the logic that repaints the old screen.  JBS 24-Oct-1982
;    61          0061  1 ! 1-016 - Make sure all lines off the screen are marked for repaint, and
;    62          0062  1 !         support non-scrolling-region terminals.  JBS 24-Oct-1982
;    63          0063  1 ! 1-017 - Fix a performance problem with deselecting.  JBS 24-Oct-1982
;    64          0064  1 ! 1-018 - Watch out for deleted lines when updating the old screen   JBS 24-Oct-1982
;    65          0065  1 ! 1-019 - Create UPDATE_LINE, so we can add fancy screen stuff for inserted and
;    66          0066  1 !         deleted lines.  JBS 23-Oct-1982
;    67          0067  1 ! 1-021 - Add scrolling to inset and delete line code.  SMB 25-Oct-1982
;    68          0068  1 ! 1-022 - Fix scrolling bug - add more notruncate code.  SMB 27-Oct-1982
;    69          0069  1 ! 1-023 - If we delete the top line, make the next line top.  JBS 01-Nov-1982
;    70          0070  1 ! 1-024 - Don't lose the line number if we must repaint but need not rebuild
;    71          0071  1 !         the screen data base.  JBS 01-Nov-1982
;    72          0072  1 ! 1-025 - Add the call to EDT$$FIX_NOTRUNC.  JBS 01-Nov-1982
;    73          0073  1 ! 1-026 - Fix a problem scrolling up on a small screen.  JBS 02-Nov-1982
;    74          0074  1 ! 1-027 - Speed up deselecting.  JBS 09-Nov-1982
;    75          0075  1 ! 1-028 - Rearrange select range processing.  JBS 10-Nov-1982
;    76          0076  1 ! 1-029 - Watch out for deleting the last line of the screen.  JBS 11-Nov-1982
;    77          0077  1 ! 1-030 - Recover from running out of memory.  JBS 15-Nov-1982
;    78          0078  1 ! 1-031 - Fix bug with cuts on noscroll terminal.  SMB 16-Nov-1982
;    79          0079  1 ! 1-032 - Fix notruncate bugs.  SMB 23-Nov-1982
;    80          0080  1 ! 1-033 - Worry about deleted lines.  JBS 25-Nov-1982
;    81          0081  1 ! 1-034 - Make a few efficiency improvements.  JBS 02-Dec-1982
;    82          0082  1 ! 1-035 - Add two paramaters to the SC_LNINS routine.  SMB 03-Dec-1982
;    83          0083  1 ! 1-036 - Change calculation of distance to select line. STS 07-Dec-1982
;    84          0084  1 ! 1-037 - When scrolling down, if we do not have scrolling regions
;    85          0085  1 !         erase the line that should have scrolled out of view.  JBS 14-Dec-1982
;    86          0086  1 ! 1-038 - Fix small bugs with boundary conditions.  SMB 20-Dec-1982
;    87          0087  1 ! 1-039 - Remove the edit buffer.  JBS 27-Dec-1982
;    88          0088  1 ! 1-040 - Do less repainting on select.  JBS 27-Dec-1982
;    89          0089  1 ! 1-041 - Add a missing dot in edit 1-040.  JBS 28-Dec-1982
;    90          0090  1 ! 1-042 - Collapse inserts and deletes together.  JBS 28-Dec-1982
;    91          0091  1 ! 1-043 - Add more TOP logic, to recover from rebuilds better.  JBS 29-Dec-1982
;    92          0092  1 ! 1-044 - Fix a bug that caused unnecessary rebuilding in NOTRUNCATE mode.  JBS 30-Dec-1982
;    93          0093  1 ! 1-045 - Modify setting of scrolling regions for multiple inserts.  SMB 30-Dec-1982
;    94          0094  1 ! 1-046 - Bug fixes on setting of top and more multiple insert work.  SMB 05-Jan-1983
;    95          0095  1 ! 1-047 - Fix bugs introduced in edit 046.  SMB 11-Jan-1983
;    96          0096  1 ! 1-048 - Bug fixes for "moving window" problems on deletes.  SMB 14-Jan-1983
;    97          0097  1 ! 1-049 - Worry about deleting the only line in the buffer.  JBS 18-Jan-1983
;    98          0098  1 ! 1-050 - Fix painting select regions on continuation lines.  JBS 19-Jan-1983
;    99          0099  1 ! 1-051 - Be more cautious about using the old cursor line after a rebuild.  JBS 20-Jan-1983
;   100          0100  1 ! 1-052 - Fix scrolling problems for NOSCROLL terminals.  SMB 25-Jan-1983
;   101          0101  1 ! 1-053 - Repair backwards scrolling bug introduced by edit 1-052.  SMB 26-Jan-1983
;   102          0102  1 ! 1-054 - We were updating the screen wrong if all of the following happened:
;   103          0103  1 !           1) we reset the screen, 2) we show the current position, and 3) we
;   104          0104  1 !           must jump to, rather than scroll to, the new position.  In showing
;   105          0105  1 !           the new position we should not assume that the screen is still erased.  JBS 28-Jan-1983
;   106          0106  1 ! 1-055 - Fix unreversing of backward select ranges.  JBS 28-Jan-1983
;   107          0107  1 ! 1-056 - Fix VT52 erase to end of screen bug with messages.  SMB 01-Feb-1983
;   108          0108  1 ! 1-057 - Avoid excess repainting after a CUT that crosses a line boundry.  JBS 25-Feb-1983
;   109          0109  1 ! 1-058 - Don't initialize the screen so often.  JBS 02-Mar-1983
;   110          0110  1 ! 1-059 - Mark the select region better on continued lines.  JBS 07-Mar-1983
;   111          0111  1 ! 1-060 - If we cut 20 lines and move forward 21 lines in a single keystroke we can
;   112          0112  1 !         scroll onto the screen lines inserted by the computation of BOTTOM.
;   113          0113  1 !         Don't fail when encountering such lines.  JBS 19-Mar-1983
;   114          0114  1 ! 1-061 - Don't update the screen based on starting or ending a selection if the
```

```
;  115          0115  1 !           screen has just been erased or if the (former) select buffer is not
;  116          0116  1 !           on the screen.  JBS 21-Mar-1983
;  117          0117  1 ! 1-062 - Try to improve performance by skipping over code not needed
;  118          0118  1 !           in simple cases.  JBS 05-Apr-1983
;  119          0119  1 ! 1-063 - We must mark all lines off the screen for repaint if any scrolling
;  120          0120  1 !           may be needed, because the motion may be done by jumping to the
;  121          0121  1 !           new area.  JBS 06-Apr-1983
;  122          0122  1 ! 1-064 - Add LOAD entry point so we can overlay the notruncate code against
;  123          0123  1 !           this module, and improve the scrolling heuristics.  JBS 18-Apr-1983
;  124          0124  1 ! 1-065 - Don't reference an undefined variable.  JBS 21-Apr-1983
;  125          0125  1 ! 1-066 - Fix deleting the top line when a previous line exists.  JBS 29-Apr-1983
;  126          0126  1 ! 1-067 - Fix message line handling on non-scrolling-region terminals.  JBS 02-May-1983
;  127          0127  1 ! 1-068 - Fix repainting an inserted line which is followed by a deleted line.  JBS 06-May-1983
;  128          0128  1 ! 1-069 - Handle the message line correctly on a non-scrolling terminal when deleting
;  129          0129  1 !           one text line.  JBS 06-May-1983
;  130          0130  1 ! 1-070 - Fix some inefficiencies when inserting and deleting lines on terminals that
;  131          0131  1 !           do not have scrolling regions.  JBS 09-May-1983
;  132          0132  1 ! 1-071 - More work on scrolling efficiency.  JBS 10-May-1983
;  133          0133  1 ! 1-072 - Don't set the scrolling region so often.  JBS 11-May-1983
;  134          0134  1 ! 1-073 - Paint the selected area correctly even if the cursor is moved after the select
;  135          0135  1 !           and before the screen is updated.  JBS 17-May-1983
;  136          0136  1 ! 1-074 - Improve performance when inserting a line in a buffer smaller than
;  137          0137  1 !           the screen size.  JBS 17-May-1983
;  138          0138  1 ! 1-075 - Fix a problem with lines moving into the message area on non-scrolling terminals.  JBS 18-May-1983
;  139          0139  1 ! 1-076 - Avoid a timing problem in VT52s by not doing reverse scrolls so fast.  JBS 18-May-1983
;  140          0140  1 ! 1-077 - Improve performance when deleting lines on noscrolling terminals.  JBS 19-May-1983
;  141          0141  1 ! 1-078 - Watch out for losing EDT$$A_CSR_SCRPTR.  JBS 20-May-1983
;  142          0142  1 ! 1-079 - Correct deleting a line when the top line is a continuation line.  JBS 27-May-1983
;  143          0143  1 ! 1-080 - Mark lines added to the screen data base as modified rather than inserted, since
;  144          0144  1 !           they do not come between two existing lines.  JBS 01-Jun-1983
;  145          0145  1 !--
;  146          0146  1
```

```
;  148          0147  1 %SBTTL 'Declarations'
;  149          0148  1 !
;  150          0149  1 ! TABLE OF CONTENTS:
;  151          0150  1 !
;  152          0151  1
;  153          0152  1 REQUIRE 'EDTSRC:TRAROUNAM';
;  154          0591  1
;  155          0592  1 FORWARD ROUTINE
;  156          0593  1     EDT$$SC_UPD : NOVALUE,
;  157          0594  1     DELETE_LINE,
;  158          0595  1     INSERT_LINE,
;  159          0596  1     EDT$$LOAD_SCRUPDATE : NOVALUE;
;  160          0597  1
;  161          0598  1 !
;  162          0599  1 ! INCLUDE FILES:
;  163          0600  1 !
;  164          0601  1
;  165          0602  1 REQUIRE 'EDTSRC:EDTREQ';
;  166          0737  1
;  167          0738  1 !
;  168          0739  1 ! MACROS:
;  169          0740  1 !
;  170          0741  1 !       NONE
;  171          0742  1 !
;  172          0743  1 ! EQUATED SYMBOLS:
;  173          0744  1 !
;  174          0745  1 !       NONE
;  175          0746  1 !
;  176          0747  1 ! OWN STORAGE:
;  177          0748  1 !
;  178          0749  1 !       NONE
;  179          0750  1 !
;  180          0751  1 ! EXTERNAL REFERENCES:
;  181          0752  1 !
;  182          0753  1 !       In the routine
```

```
;  184          0754  1 %SBTTL 'EDT$$SC_UPD  - update the screen'
;  185          0755  1
;  186          0756  1 GLOBAL ROUTINE EDT$$SC_UPD                      ! Update the screen
;  187          0757  1     : NOVALUE =
;  188          0758  1
;  189          0759  1 !++
;  190          0760  1 ! FUNCTIONAL DESCRIPTION:
;  191          0761  1 !
;  192          0762  1 !       This routine is called to do a screen update.  Most of the work done
;  193          0763  1 !       by this routine involves deciding on whether or not scrolling should
;  194          0764  1 !       be done.  Basically, it figures out which line should be on the top
;  195          0765  1 !       of the screen, then determines how far away from the current line it
;  196          0766  1 !       has moved.  The actual update is handled by the EDT$$SC_RFRELN routine.
;  197          0767  1 !
;  198          0768  1 ! FORMAL PARAMETERS:
;  199          0769  1 !
;  200          0770  1 !       NONE
;  201          0771  1 !
;  202          0772  1 ! IMPLICIT INPUTS:
;  203          0773  1 !
;  204          0774  1 !       EDT$$A_OLD_SEL
;  205          0775  1 !       EDT$$L_LNO_EMPTY
;  206          0776  1 !       EDT$$G_SCR_REBUILD
;  207          0777  1 !       EDT$$A_EOB_SCRPTR
;  208          0778  1 !       EDT$$G_SCR_CHGD
;  209          0779  1 !       EDT$$T_LN_BUF
;  210          0780  1 !       EDT$$A_LST_SCRPTR
;  211          0781  1 !       EDT$$A_CUR_SCRPTR
;  212          0782  1 !       EDT$$A_WK_LN
;  213          0783  1 !       EDT$$Z_EOB_LN
;  214          0784  1 !       EDT$$A_SCR_BUF
;  215          0785  1 !       EDT$$A_FST_SCRPTR
;  216          0786  1 !       EDT$$G_CS_LHNO
;  217          0787  1 !       EDT$$G_CS_OLDCHNO
;  218          0788  1 !       EDT$$L_CS_LN
;  219          0789  1 !       EDT$$G_CS_LNO
;  220          0790  1 !       EDT$$G_CUR_COL
;  221          0791  1 !       EDT$$G_LN_NO
;  222          0792  1 !       EDT$$A_SEL_BUF
;  223          0793  1 !       EDT$$L_TOP_LN
;  224          0794  1 !       EDT$$A_CUR_BUF
;  225          0795  1 !       EDT$$G_SCR_LNS
;  226          0796  1 !       EDT$$G_SCLL_TOP
;  227          0797  1 !       EDT$$G_SCLL_BOT
;  228          0798  1 !       EDT$$G_TI_TYP
;  229          0799  1 !       EDT$$A_LN_PTR
;  230          0800  1 !       EDT$$G_TI_SCROLL
;  231          0801  1 !       EDT$$A_CSR_SCRPTR
;  232          0802  1 !       EDT$$A_TOP_SCRPTR
;  233          0803  1 !       EDT$$A_BOT_SCRPTR
;  234          0804  1 !       EDT$$L_CUR_SCRLN
;  235          0805  1 !       EDT$$A_FST_AVLN
;  236          0806  1 !       EDT$$G_TRUN
;  237          0807  1 !       EDT$$G_RECS_INSERTED
;  238          0808  1 !       EDT$$G_BOT_LINE
;  239          0809  1 !       EDT$$G_TOP_RECNO
;  240          0810  1 !       EDT$$G_ANY_CHANGES
```

```
;  241          0811  1 !       EDT$$G_REVID
;  242          0812  1 !
;  243          0813  1 ! IMPLICIT OUTPUTS:
;  244          0814  1 !
;  245          0815  1 !       EDT$$A_OLD_SEL
;  246          0816  1 !       EDT$$G_SCR_REBUILD
;  247          0817  1 !       EDT$$A_CSR_SCRPTR
;  248          0818  1 !       EDT$$A_TOP_SCRPTR
;  249          0819  1 !       EDT$$L_CUR_SCRLN
;  250          0820  1 !       EDT$$A_SEL_BUF
;  251          0821  1 !       EDT$$G_CS_CHNO
;  252          0822  1 !       EDT$$G_CS_OLDCHNO
;  253          0823  1 !       EDT$$L_CS_LN
;  254          0824  1 !       EDT$$G_CS_LNO
;  255          0825  1 !       EDT$$G_LN_NO
;  256          0826  1 !       EDT$$L_TOP_LN
;  257          0827  1 !       EDT$$G_CUR_COL
;  258          0828  1 !       EDT$$G_RECS_INSERTED
;  259          0829  1 !       EDT$$A_FST_AVLN
;  260          0830  1 !       EDT$$A_BOT_SCRPTR
;  261          0831  1 !       EDT$$G_MEM_CNT
;  262          0832  1 !       EDT$$G_BOT_LINE
;  263          0833  1 !       EDT$$G_TOP_RECNO
;  264          0834  1 !       EDT$$G_ANY_CHANGES
;  265          0835  1 !       EDT$$G_MSGFLG
;  266          0836  1 !
;  267          0837  1 ! ROUTINE VALUE:
;  268          0838  1 !
;  269          0839  1 !       NONE
;  270          0840  1 !
;  271          0841  1 ! SIDE EFFECTS:
;  272          0842  1 !
;  273          0843  1 !       MANY
;  274          0844  1 !
;  275          0845  1 !--
;  276          0846  1
;  277          0847  2     BEGIN
;  278          0848  2
;  279          0849  2     EXTERNAL ROUTINE
;  280          0850  2         EDT$$SC_SETSCLLREG,                     ! Set the scrolling region
;  281          0851  2         EDT$$SC_LNINS,                          ! Insert a record into the screen data base
;  282          0852  2         EDT$$FMT_LIT,                           ! Output a literal string
;  283          0853  2         EDT$$SC_FNDREC,                         ! Find the current screen pointer
;  284          0854  2         EDT$$OUT_FMTBUF,                        ! Output the format buffer to the screen
;  285          0855  2         EDT$$RPL_CHGDLN,                        ! Replace a modified line in the work file
;  286          0856  2         EDT$$SC_INIT,                           ! Initialize the screen
;  287          0857  2         EDT$$SC_CPUCSPOS : NOVALUE,             ! Compute the cursor position
;  288          0858  2         EDT$$SC_POSCSIF : NOVALUE,              ! Position the cursor
;  289          0859  2         EDT$$SC_ERAALL : NOVALUE,               ! Erase the screen
;  290          0860  2         EDT$$SC_MOVTOLN,                        ! Move to a record in the work file relative to the current record
;  291          0861  2         EDT$$SC_RFRELN : NOVALUE,               ! Refresh a screen line
;  292          0862  2         EDT$$SC_NONREVID : NOVALUE,             ! Put the screen in normal video mode
;  293          0863  2         EDT$$SC_REPAINT : NOVALUE,              ! Mark some lines in the screen data base for repaint
;  294          0864  2         EDT$$TI_ENBLAUTREP : NOVALUE,           ! Enable or disable autorepeat
;  295          0865  2         EDT$$FIX_NOTRUNC_NOOVERLAY : NOVALUE;   ! Fix screen data base in NOTRUNCATE mode
;  296          0866  2
;  297          0867  2     EXTERNAL
```

```
;  298     0868  2          EDT$$A_BOT_SCRPTR : REF SCREEN_LINE,     ! Address of bottom screen line
;  299     0869  2          EDT$$A_OLD_SEL,                          ! Address of old select buffer
;  300     0870  2          EDT$$L_LNO_EMPTY : LN_BLOCK,             ! Code for empty line
;  301     0871  2          EDT$$G_SCR_REBUILD,                      ! Rebuild the screen
;  302     0872  2          EDT$$A_EOB_SCRPTR : REF SCREEN_LINE,     ! EOB screen pointer
;  303     0873  2          EDT$$G_SCR_CHGD,                         ! non-zero = the screen has been mangled
;  304     0874  2          EDT$$G_MESSAGE_LINE,                     ! Message line
;  305     0875  2          EDT$$T_LN_BUF,                           ! Start of line buffer
;  306     0876  2          EDT$$A_CSR_SCRPTR : REF SCREEN_LINE,     ! Current cursor line screen info
;  307     0877  2          EDT$$A_CUR_SCRPTR : REF SCREEN_LINE,     ! Current workfile line screen pointer
;  308     0878  2          EDT$$A_LST_SCRPTR : REF SCREEN_LINE,     ! Last data structure pointer
;  309     0879  2          EDT$$A_WK_LN : REF LIN_BLOCK,            ! Pointer to current line in work file
;  310     0880  2          EDT$$Z_EOB_LN,                           ! Special structure for [EOB] line
;  311     0881  2          EDT$$A_SCR_BUF : REF TBCB_BLOCK,         ! Current screen buffer
;  312     0882  2          EDT$$A_FST_SCRPTR : REF SCREEN_LINE,     ! First screen line info address
;  313     0883  2          EDT$$A_TOP_SCRPTR : REF SCREEN_LINE,     ! Top screen line info address
;  314     0884  2          EDT$$L_CUR_SCRLN : LN_BLOCK,             ! Current screen line record number
;  315     0885  2          EDT$$G_CS_CHNO,                          ! character position of cursor
;  316     0886  2          EDT$$G_CS_OLDCHNO,                       ! Previous character position of cursor
;  317     0887  2          EDT$$L_CS_LN : LN_BLOCK,                 ! record number of cursor line
;  318     0888  2          EDT$$G_CS_LNO,                           ! current cursor line
;  319     0889  2          EDT$$G_CUR_COL,                          ! current cursor column
;  320     0890  2          EDT$$G_LN_NO,                            ! current line number.
;  321     0891  2          EDT$$A_SEL_POS,                          ! select character position
;  322     0892  2          EDT$$L_SEL_LN : LN_BLOCK,                ! select record
;  323     0893  2          EDT$$A_SEL_BUF,                          ! select buffer.
;  324     0894  2          EDT$$L_TOP_LN : LN_BLOCK,                ! Line number of enforced top line.
;  325     0895  2          EDT$$A_CUR_BUF : REF TBCB_BLOCK,         ! The current buffer tbcb.
;  326     0896  2          EDT$$G_SCR_LNS,                          ! No of lines on screen
;  327     0897  2          EDT$$G_SCLL_TOP,                         ! Top line for scrolling up
;  328     0898  2          EDT$$G_SCLL_BOT,                         ! Bottom line for scrolling down
;  329     0899  2          EDT$$G_TI_TYP,                           ! Terminal type.
;  330     0900  2          EDT$$A_LN_PTR,                           ! Current character pointer.
;  331     0901  2          EDT$$G_TI_SCROLL,                        ! 1 = we have scrolling regions
;  332     0902  2          EDT$$G_RECS_INSERTED,                    ! Number of records inserted since the last screen update
;  333     0903  2          EDT$$A_FST_AVLN : REF SCREEN_LINE,       ! List of available screen line data blocks
;  334     0904  2          EDT$$G_TRUN,                             ! 0 = SET NOTRUNCATE
;  335     0905  2          EDT$$G_MEM_CNT,                          ! Number of lines in the screen data base
;  336     0906  2          EDT$$G_BOT_LINE,                         ! All lines below this one have been erased
;  337     0907  2          EDT$$G_TOP_RECNO,                        ! Record number of top line
;  338     0908  2          EDT$$G_ANY_CHANGES,                      ! 1 = an edit has been made
;  339     0909  2          EDT$$G_REVID,                            ! 1 = screen is in reverse video
;  340     0910  2          EDT$$G_MSGFLG;                           ! 1 = there is a message in the message area
;  341     0911  2
;  342     0912  2      LOCAL
;  343     0913  2          SCROLLING_NEEDED,                        ! 0 = no scrolling needed
;  344     0914  2          TOP_DIST,                                ! Displacement to top scrptr
;  345     0915  2          TEMP_LINE : LN_BLOCK,                    ! Temp line number
;  346     0916  2          TOP_DISP,                                ! Top line displacement from current
;  347     0917  2          DIR,                                     ! Direction of motion since last screen update
;  348     0918  2          SCLL_NUM,                                ! Scroll line limit
;  349     0919  2          TOP_SET,                                 ! Top record successfully set
;  350     0920  2          DISP,                                    ! Displacement from cursor screen line
;  351     0921  2          ABOVE,                                   ! Number of lines above current
;  352     0922  2          BUILD_SCR,                               ! Flag which says rebuild screen
;  353     0923  2          SCRPTR : REF SCREEN_LINE,                ! Address of a current screen line buffer
;  354     0924  2          CURSOR_POS,                              ! Column position of cursor
```

```
;  355     0925  2          CURSOR_LINE,                    ! Number of screen lines before cursor.
;  356     0926  2          BELOW,                          ! No. of screen lines below cursor line.
;  357     0927  2          REC_NO,                         ! Current relative reocrd number
;  358     0928  2          OLD_TOP_RECNO,                  ! Rel record number of old top record
;  359     0929  2          OLD_BOT_RECNO,                  ! Rel record number of old bottom record
;  360     0930  2          TOP_RECNO,                      ! Rel record number of new top record
;  361     0931  2          TOP_SCRPTR : REF SCREEN_LINE,   ! Address of the top line on the screen
;  362     0932  2          ERASE_ALL,                      ! 1 = we have erased the text part of the screen
;  363     0933  2          LNINS_VAL,                      ! Value returned by EDT$$SC_LNINS
;  364     0934  2          ANY_CHANGES;                    ! 1 = we must scan the screen data base for changes
;  365     0935  2
;  366     0936  2      !+
;  367     0937  2      ! Make sure we are in normal video if no select range.
;  368     0938  2      !-
;  369     0939  2
;  370     0940  2          IF ((.EDT$$G_REVID NEQ 0) AND (.EDT$$A_SEL_BUF NEQA .EDT$$A_CUR_BUF)) THEN EDT$$SC_NONREVID ();
;  371     0941  2
;  372     0942  2      !+
;  373     0943  2      ! Remember the original character position and relative line number in
;  374     0944  2      ! work file terms.
;  375     0945  2      !-
;  376     0946  2          EDT$$G_LN_NO = 0;
;  377     0947  2          EDT$$G_CS_CHNO = .EDT$$A_LN_PTR - EDT$$T_LN_BUF;
;  378     0948  2          MOVELINE (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_CS_LN);
;  379     0949  2          EDT$$RPL_CHGDLN ();
;  380     0950  2      !+
;  381     0951  2      ! If we are in NOTRUNCATE mode, make sure lines get adjusted due to carry from
;  382     0952  2      ! or borrow to earlier lines.
;  383     0953  2      !-
;  384     0954  2
;  385     0955  3          IF ( NOT .EDT$$G_TRUN)
;  386     0956  2          THEN
;  387     0957  3              BEGIN
;  388     0958  3
;  389     0959  3              IF .EDT$$G_ANY_CHANGES THEN EDT$$FIX_NOTRUNC_NOOVERLAY ();
;  390     0960  3
;  391     0961  2              END;
;  392     0962  2
;  393     0963  2          SCRPTR = 0;
;  394     0964  2      !+
;  395     0965  2      ! Compute the cursor position.  We will recompute if we must rebuild the screen data base.
;  396     0966  2      !-
;  397     0967  2          EDT$$SC_CPUCSPOS (CURSOR_LINE, CURSOR_POS);
;  398     0968  2          CURSOR_LINE = .EDT$$G_CS_LNO;
;  399     0969  2      !+
;  400     0970  2      ! If the screen has been mangled, or we have changed buffers or deleted or inserted a lot of lines,
;  401     0971  2      ! erase the screen and repaint all the lines.
;  402     0972  2      !-
;  403     0973  2
;  404     0974  3          IF ((.EDT$$G_SCR_CHGD NEQ 0) OR (.EDT$$A_SCR_BUF NEQA .EDT$$A_CUR_BUF) OR   !
;  405     0975  3              (.EDT$$G_RECS_INSERTED GTR (2*.EDT$$G_SCR_LNS)))
;  406     0976  2          THEN
;  407     0977  3              BEGIN
;  408     0978  3      !+
;  409     0979  3      ! Don't initialize the terminal unless it has been requested.
;  410     0980  3      !-
;  411     0981  3
```

```
;  412          0982  3          IF (.EDT$$G_SCR_CHGD EQL 2) THEN EDT$$SC_INIT ();
;  413          0983  3
;  414          0984  3      !+
;  415          0985  3      ! Erase the screen.
;  416          0986  3      !-
;  417          0987  3          EDT$$G_CS_LNO = 0;
;  418          0988  3          EDT$$SC_ERAALL ();
;  419          0989  3          EDT$$G_BOT_LINE = 0;
;  420          0990  3          ERASE_ALL = 1;
;  421          0991  3          END
;  422          0992  2      ELSE
;  423          0993  2          ERASE_ALL = 0;
;  424          0994  2
;  425          0995  2      !+
;  426          0996  2      ! Determine whether the screen structure has to be rebuilt.
;  427          0997  2      !-
;  428          0998  2      BUILD_SCR = .EDT$$G_SCR_REBUILD;
;  429          0999  2      !+
;  430          1000  2      ! If the current position is not in the screen data base, rebuild.
;  431          1001  2      !-
;  432          1002  2
;  433          1003  3      IF ( NOT .BUILD_SCR)
;  434          1004  2      THEN
;  435          1005  3          BEGIN
;  436          1006  3          EDT$$A_CUR_SCRPTR = EDT$$SC_FNDREC ((.EDT$$A_LN_PTR - EDT$$T_LN_BUF), DISP);
;  437          1007  3
;  438          1008  5          IF ((.EDT$$A_CUR_SCRPTR EQLA 0) OR (.EDT$$A_TOP_SCRPTR EQLA 0)  !
;  439          1009  4              OR (.EDT$$A_CSR_SCRPTR EQLA 0))
;  440          1010  3          THEN
;  441          1011  3              BUILD_SCR = 1;
;  442          1012  3
;  443          1013  2          END;
;  444          1014  2
;  445          1015  2      !+
;  446          1016  2      ! Compute whether or not we must scan the screen data base for any changes.
;  447          1017  2      !-
;  448          1018  2      ANY_CHANGES = .EDT$$G_ANY_CHANGES;
;  449          1019  2      !+
;  450          1020  2      ! Compute the direction of motion since the last screen update.
;  451          1021  2      ! If we have changed buffers, assume forward.
;  452          1022  2      !-
;  453          1023  2
;  454          1024  3      IF ((LINNOEQL (EDT$$L_LNO_EMPTY, EDT$$L_CUR_SCRLN)) OR (.EDT$$A_SCR_BUF NEQA .EDT$$A_CUR_BUF))
;  455          1025  2      THEN
;  456          1026  2          DIR = 1
;  457          1027  2      ELSE
;  458          1028  2          DIR = CMPLNO (EDT$$L_CS_LN, EDT$$L_CUR_SCRLN);
;  459          1029  2
;  460          1030  2      !+
;  461          1031  2      ! Compute whether or not any scrolling may be necessary.
;  462          1032  2      !-
;  463          1033  2
;  464          1034  4      IF ((.EDT$$G_RECS_INSERTED NEQ 0)            !
;  465          1035  4          OR (.EDT$$A_CSR_SCRPTR NEQA .EDT$$A_CUR_SCRPTR) !
;  466          1036  4          OR ( NOT LINNOEQL (EDT$$L_TOP_LN, EDT$$L_LNO_EMPTY))    !
;  467          1037  3          OR (.DIR NEQ 0))
;  468          1038  2      THEN
```

```
;  469          1039  2          SCROLLING_NEEDED = 1
;  470          1040  2      ELSE
;  471          1041  2          SCROLLING_NEEDED = 0;
;  472          1042  2
;  473          1043  3      IF ( NOT .BUILD_SCR)
;  474          1044  2      THEN
;  475          1045  3          BEGIN
;  476          1046  3
;  477          1047  4          IF ((.EDT$$A_SEL_BUF NEQA .EDT$$A_OLD_SEL) AND (.EDT$$G_TI_TYP EQL TERM_VT100) AND      !
;  478          1048  4              ((.EDT$$A_SCR_BUF EQLA .EDT$$A_OLD_SEL) OR (.EDT$$A_SCR_BUF EQLA .EDT$$A_SEL_BUF)) AND      !
;  479          1049  4              ( NOT .ERASE_ALL))
;  480          1050  3          THEN
;  481          1051  4              BEGIN
;  482          1052  4      !+
;  483          1053  4      ! We have started or ended a selection.  Repaint all selected or formerly selected lines.
;  484          1054  4      !-
;  485          1055  4
;  486          1056  4              LOCAL
;  487          1057  4                  SELDIR,
;  488          1058  4                  REC_OFFSET,
;  489          1059  4                  OUR_LINE : LN_BLOCK,
;  490          1060  4                  OUR_CHNO,
;  491          1061  4                  OUR_SCRPTR : REF SCREEN_LINE;
;  492          1062  4
;  493          1063  4      !+
;  494          1064  4      ! If this is a deselection we must repaint from the old line to the select line.
;  495          1065  4      ! If this is a selection we must repaint from the current line to the select line.
;  496          1066  4      !-
;  497          1067  4
;  498          1068  5              IF (.EDT$$A_SEL_BUF EQLA 0)
;  499          1069  4              THEN
;  500          1070  5                  BEGIN
;  501          1071  5                  MOVELINE (EDT$$L_CUR_SCRLN, OUR_LINE);
;  502          1072  5                  SUBLINE (EDT$$L_CUR_SCRLN, EDT$$L_CS_LN, TEMP_LINE);
;  503          1073  5                  REC_OFFSET = .(TEMP_LINE [LN_LO])<0, 16, 1>;
;  504          1074  5                  OUR_CHNO = .EDT$$G_CS_OLDCHNO;
;  505          1075  5                  OUR_SCRPTR = .EDT$$A_CSR_SCRPTR;
;  506          1076  5                  END
;  507          1077  4              ELSE
;  508          1078  5                  BEGIN
;  509          1079  5                  MOVELINE (EDT$$L_CS_LN, OUR_LINE);
;  510          1080  5                  REC_OFFSET = 0;
;  511          1081  5                  OUR_CHNO = .EDT$$G_CS_CHNO;
;  512          1082  5                  OUR_SCRPTR = .EDT$$A_CUR_SCRPTR;
;  513          1083  4                  END;
;  514          1084  4
;  515          1085  4              SUBLINE (OUR_LINE, EDT$$L_SEL_LN, TEMP_LINE);
;  516          1086  4
;  517          1087  5              IF ((.TEMP_LINE [LN_HI] AND %X'8000') NEQ 0)
;  518          1088  4              THEN
;  519          1089  4                  SELDIR = -1
;  520          1090  4              ELSE
;  521          1091  4
;  522          1092  4                  IF (.TEMP_LINE [LN_LO] NEQU 0) THEN SELDIR = 1 ELSE SELDIR = 0;
;  523          1093  4
;  524          1094  4              REC_NO = .(TEMP_LINE [LN_LO])<0, 16, 1> - .REC_OFFSET;
;  525          1095  4              EDT$$SC_MOVTOLN (.REC_NO);
```

```
;  526          1096  4              SCRPTR = EDT$$SC_FNDREC (.EDT$$A_SEL_POS - EDT$$T_LN_BUF, DISP);
;  527          1097  4
;  528          1098  5              IF (.SCRPTR EQLA 0)
;  529          1099  4              THEN
;  530          1100  4                  BUILD_SCR = 1
;  531          1101  4              ELSE
;  532          1102  5                  BEGIN
;  533          1103  5
;  534          1104  6                  IF (.SELDIR EQL 0)
;  535          1105  5                  THEN
;  536          1106  6                      BEGIN
;  537          1107  6
;  538          1108  7                      IF ((.EDT$$A_SEL_POS - EDT$$T_LN_BUF) LSS .OUR_CHNO)        !
;  539          1109  6                      THEN
;  540          1110  6                          EDT$$SC_REPAINT (.SCRPTR,       !
;  541          1111  6                              .OUR_SCRPTR,        !
;  542          1112  6                              .EDT$$A_SEL_POS - EDT$$T_LN_BUF - .SCRPTR [SCR_CHR_FROM],
;  543          1113  6                              .OUR_CHNO - .OUR_SCRPTR [SCR_CHR_FROM] - 1, 0)
;  544          1114  6                      ELSE
;  545          1115  6
;  546          1116  7                          IF ((.EDT$$A_SEL_POS - EDT$$T_LN_BUF) GTR .OUR_CHNO)    !
;  547          1117  6                          THEN
;  548          1118  6                              EDT$$SC_REPAINT (.OUR_SCRPTR,       !
;  549          1119  6                                  .SCRPTR,        !
;  550          1120  6                                  .OUR_CHNO - .OUR_SCRPTR [SCR_CHR_FROM],         !
;  551          1121  6                                  .EDT$$A_SEL_POS - EDT$$T_LN_BUF - .SCRPTR [SCR_CHR_FROM] - 1, 0);
;  552          1122  6
;  553          1123  6                      END
;  554          1124  5                  ELSE
;  555          1125  5
;  556          1126  6                      IF (.SELDIR GTR 0)
;  557          1127  5                      THEN
;  558          1128  5                          EDT$$SC_REPAINT (.OUR_SCRPTR,   !
;  559          1129  5                              .SCRPTR,        !
;  560          1130  5                              .OUR_CHNO - .OUR_SCRPTR [SCR_CHR_FROM],     !
;  561          1131  5                              .EDT$$A_SEL_POS - EDT$$T_LN_BUF - .SCRPTR [SCR_CHR_FROM] - 1, 0)
;  562          1132  5                      ELSE
;  563          1133  5
;  564          1134  6                          IF (.SELDIR LSS 0)
;  565          1135  5                          THEN
;  566          1136  5                              EDT$$SC_REPAINT (.SCRPTR,   !
;  567          1137  5                                  .OUR_SCRPTR,    !
;  568          1138  5                                  .EDT$$A_SEL_POS - EDT$$T_LN_BUF - .SCRPTR [SCR_CHR_FROM],       !
;  569          1139  5                                  .OUR_CHNO - .OUR_SCRPTR [SCR_CHR_FROM] - 1, 0)
;  570          1140  5                          ELSE
;  571          1141  5                              ASSERT (0);
;  572          1142  5
;  573          1143  4                      END;
;  574          1144  4
;  575          1145  4                  ANY_CHANGES = 1;
;  576          1146  3                  END;
;  577          1147  3
;  578          1148  2              END;
;  579          1149  2
;  580          1150  3          IF (( NOT .BUILD_SCR) AND ( NOT .ERASE_ALL))
;  581          1151  2          THEN
;  582          1152  3              BEGIN
```

```
;  583          1153  3
;  584          1154  4          IF ((.EDT$$A_SEL_BUF EQLA .EDT$$A_CUR_BUF) AND (.EDT$$G_TI_TYP EQL TERM_VT100))
;  585          1155  3          THEN
;  586          1156  4              BEGIN
;  587          1157  4      !+
;  588          1158  4      ! The select range is in the current buffer.  Repaint lines between the previous
;  589          1159  4      ! and the current cursor, to be sure they are properly reversed or not.
;  590          1160  4      !-
;  591          1161  4
;  592          1162  5              IF (.DIR LSS 0)
;  593          1163  4              THEN
;  594          1164  4                  EDT$$SC_REPAINT (.EDT$$A_CUR_SCRPTR, .EDT$$A_CSR_SCRPTR,
;  595          1165  4                      .EDT$$G_CS_CHNO - .EDT$$A_CUR_SCRPTR [SCR_CHR_FROM],
;  596          1166  4                      .EDT$$G_CS_OLDCHNO - .EDT$$A_CSR_SCRPTR [SCR_CHR_FROM] - 1, 0)
;  597          1167  4              ELSE
;  598          1168  4
;  599          1169  5                  IF (.DIR GTR 0)
;  600          1170  4                  THEN
;  601          1171  4                      EDT$$SC_REPAINT (.EDT$$A_CSR_SCRPTR, .EDT$$A_CUR_SCRPTR,
;  602          1172  4                          .EDT$$G_CS_OLDCHNO - .EDT$$A_CSR_SCRPTR [SCR_CHR_FROM],          !
;  603          1173  4                          .EDT$$G_CS_CHNO - .EDT$$A_CUR_SCRPTR [SCR_CHR_FROM] - 1, 0)
;  604          1174  4                  ELSE
;  605          1175  4
;  606          1176  5                      IF (.EDT$$G_CS_CHNO LSS .EDT$$G_CS_OLDCHNO) !
;  607          1177  4                      THEN
;  608          1178  4                          EDT$$SC_REPAINT (.EDT$$A_CUR_SCRPTR,    !
;  609          1179  4                              .EDT$$A_CSR_SCRPTR,          !
;  610          1180  4                              .EDT$$G_CS_CHNO - .EDT$$A_CUR_SCRPTR [SCR_CHR_FROM],
;  611          1181  4                              .EDT$$G_CS_OLDCHNO - .EDT$$A_CSR_SCRPTR [SCR_CHR_FROM] - 1, 0)
;  612          1182  4                      ELSE
;  613          1183  4
;  614          1184  5                          IF (.EDT$$G_CS_CHNO GTR .EDT$$G_CS_OLDCHNO)     !
;  615          1185  4                          THEN
;  616          1186  4                              EDT$$SC_REPAINT (.EDT$$A_CSR_SCRPTR,        !
;  617          1187  4                                  .EDT$$A_CUR_SCRPTR,     !
;  618          1188  4                                  .EDT$$G_CS_OLDCHNO - .EDT$$A_CSR_SCRPTR [SCR_CHR_FROM],          !
;  619          1189  4                                  .EDT$$G_CS_CHNO - .EDT$$A_CUR_SCRPTR [SCR_CHR_FROM] - 1, 0);
;  620          1190  4
;  621          1191  4              ANY_CHANGES = 1;
;  622          1192  3              END;
;  623          1193  3
;  624          1194  2          END;
;  625          1195  2
;  626          1196  2      !+
;  627          1197  2      ! Mark all lines off the screen for repaint.
;  628          1198  2      !-
;  629          1199  2
;  630          1200  3          IF ( NOT .BUILD_SCR)
;  631          1201  2          THEN
;  632          1202  3              BEGIN
;  633          1203  3      !+
;  634          1204  3      ! If the screen has been erased we must repaint everything, otherwise only lines
;  635          1205  3      ! off the screen will need to be repainted.  Marking the lines off the screen for
;  636          1206  3      ! repaint removes the deleted lines from the screen data base, to avoid confusing
;  637          1207  3      ! our count of the number of lines above and below the current line.
;  638          1208  3      !-
;  639          1209  3
```

```
;  640          1210  3          IF .ERASE_ALL
;  641          1211  3          THEN
;  642          1212  4              BEGIN
;  643          1213  4              EDT$$SC_REPAINT (.EDT$$A_FST_SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
;  644          1214  4              ANY_CHANGES = 1;
;  645          1215  4              END
;  646          1216  3          ELSE
;  647          1217  4              BEGIN
;  648          1218  4
;  649          1219  4              IF .SCROLLING_NEEDED
;  650          1220  4              THEN
;  651          1221  5                  BEGIN
;  652          1222  5                  SCRPTR = .EDT$$A_TOP_SCRPTR [SCR_PRV_LINE];
;  653          1223  5
;  654          1224  5                  IF (.SCRPTR NEQA 0) THEN EDT$$SC_REPAINT (.EDT$$A_FST_SCRPTR, .SCRPTR, 0, 255, 1);
;  655          1225  5
;  656          1226  6                  IF (.EDT$$A_BOT_SCRPTR NEQA 0)
;  657          1227  5                  THEN
;  658          1228  6                      BEGIN
;  659          1229  6                      SCRPTR = .EDT$$A_BOT_SCRPTR [SCR_NXT_LINE];
;  660          1230  6
;  661          1231  6                      IF (.SCRPTR NEQA 0) THEN EDT$$SC_REPAINT (.SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
;  662          1232  6
;  663          1233  5                      END;
;  664          1234  5
;  665          1235  4                  END;
;  666          1236  4
;  667          1237  3              END;
;  668          1238  3
;  669          1239  2          END;
;  670          1240  2
;  671          1241  2 !+
;  672          1242  2 ! If we have lost our record of the top of the screen we must rebuild.
;  673          1243  2 !-
;  674          1244  2
;  675          1245  2      IF (.EDT$$A_TOP_SCRPTR EQLA 0) THEN BUILD_SCR = 1;
;  676          1246  2
;  677          1247  3      IF ( NOT .BUILD_SCR)
;  678          1248  2      THEN
;  679          1249  3          BEGIN
;  680          1250  3 !+
;  681          1251  3 ! Find the relative record number of the old cursor line.
;  682          1252  3 ! We must be careful of deleted lines.  The convention is that a deleted line
;  683          1253  3 ! has the record number of the next lower line.  This prevents deleted
;  684          1254  3 ! lines before record zero from having negative absolute record numbers.
;  685          1255  3 !-
;  686          1256  3          SCRPTR = .EDT$$A_CUR_SCRPTR;
;  687          1257  3          REC_NO = 0;
;  688          1258  3
;  689          1259  3          CASE .DIR FROM -1 TO 1 OF
;  690          1260  3              SET
;  691          1261  3
;  692          1262  3              [1] :
;  693          1263  4                  BEGIN
;  694          1264  4 !+
;  695          1265  4 ! The new line is after the old.  We must move back in the work file.
;  696          1266  4 !-
```

```
;  697     1267  4
;  698     1268  4                    DO
;  699     1269  5                        BEGIN
;  700     1270  5
;  701     1271  6                        IF ((.SCRPTR [SCR_LINE_IDX] EQL 0) OR        !
;  702     1272  6                            ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0))
;  703     1273  5                        THEN
;  704     1274  6                            BEGIN
;  705     1275  6
;  706     1276  6                            LOCAL
;  707     1277  6                                PREV_SCRPTR : REF SCREEN_LINE;
;  708     1278  6
;  709     1279  6                            PREV_SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  710     1280  6
;  711     1281  7                            IF (.PREV_SCRPTR NEQA 0)
;  712     1282  6                            THEN
;  713     1283  7                                BEGIN
;  714     1284  7
;  715     1285  8                                IF ((.PREV_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0)
;  716     1286  7                                THEN
;  717     1287  7                                    REC_NO = .REC_NO - 1;
;  718     1288  7
;  719     1289  6                                END;
;  720     1290  6
;  721     1291  5                            END;
;  722     1292  5
;  723     1293  5                        SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  724     1294  5                        END
;  725     1295  4                    UNTIL ((.SCRPTR EQLA .EDT$$A_CSR_SCRPTR) OR (.SCRPTR EQLA 0));
;  726     1296  4
;  727     1297  3                    END;
;  728     1298  3
;  729     1299  3                [0] :
;  730     1300  4                    BEGIN
;  731     1301  4 !+
;  732     1302  4 ! We are positioned correctly in the work file.
;  733     1303  4 !-
;  734     1304  4                    SCRPTR = .EDT$$A_CSR_SCRPTR;
;  735     1305  3                    END;
;  736     1306  3
;  737     1307  3                [-1] :
;  738     1308  4                    BEGIN
;  739     1309  4 !+
;  740     1310  4 ! The new line is before the old.  We must move forward in the work file.
;  741     1311  4 !-
;  742     1312  4
;  743     1313  4                    DO
;  744     1314  5                        BEGIN
;  745     1315  5
;  746     1316  5                        LOCAL
;  747     1317  5                            NEXT_SCRPTR : REF SCREEN_LINE;
;  748     1318  5
;  749     1319  5                        NEXT_SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  750     1320  5
;  751     1321  6                        IF (.NEXT_SCRPTR NEQA 0)
;  752     1322  5                        THEN
;  753     1323  6                            BEGIN
```

```
;  754          1324  6                          IF (((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0) AND   !
;  755          1325  7                              ((.NEXT_SCRPTR [SCR_LINE_IDX] EQL 0) OR     !
;  756          1326  8                              ((.NEXT_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)))
;  757          1327  7                          THEN
;  758          1328  6                              REC_NO = .REC_NO + 1;
;  759          1329  6
;  760          1330  6                          END;
;  761          1331  5
;  762          1332  5                      SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  763          1333  5                      END
;  764          1334  5                  UNTIL ((.SCRPTR EQLA .EDT$$A_CSR_SCRPTR) OR (.SCRPTR EQLA 0));
;  765          1335  4
;  766          1336  4                  END;
;  767          1337  3
;  768          1338  3              [OUTRANGE] :
;  769          1339  3                  ASSERT (0);
;  770          1340  3              TES;
;  771          1341  3
;  772          1342  3
;  773          1343  3      !+
;  774          1344  3      ! If we couldn't find it, rebuild the screen.
;  775          1345  3      !-
;  776          1346  3
;  777          1347  3              IF ((.SCRPTR NEQA .EDT$$A_CSR_SCRPTR) OR (.SCRPTR EQLA 0)) THEN BUILD_SCR = 1;
;  778          1348  3
;  779          1349  2              END;
;  780          1350  2
;  781          1351  2      !+
;  782          1352  2      ! Now find the relative record number of the old top line.  We can use the value
;  783          1353  2      ! from last time if no scrolling will be needed.
;  784          1354  2      !-
;  785          1355  2
;  786          1356  3          IF ( NOT .BUILD_SCR)
;  787          1357  2          THEN
;  788          1358  3              BEGIN
;  789          1359  3
;  790          1360  4              IF ( NOT .SCROLLING_NEEDED)
;  791          1361  3              THEN
;  792          1362  3                  OLD_TOP_RECNO = .EDT$$G_TOP_RECNO
;  793          1363  3              ELSE
;  794          1364  4                  BEGIN
;  795          1365  4
;  796          1366  4                  WHILE ((.SCRPTR NEQA .EDT$$A_TOP_SCRPTR) AND (.SCRPTR NEQA 0)) DO
;  797          1367  5                      BEGIN
;  798          1368  5
;  799          1369  6                      IF ((.SCRPTR [SCR_LINE_IDX] EQL 0) OR    !
;  800          1370  6                          ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0))
;  801          1371  5                      THEN
;  802          1372  6                          BEGIN
;  803          1373  6
;  804          1374  6                          LOCAL
;  805          1375  6                              PREV_SCRPTR : REF SCREEN_LINE;
;  806          1376  6
;  807          1377  6                          PREV_SCRPTR = .SCRPTR [SCR_PREV_LINE];
;  808          1378  6
;  809          1379  7                          IF (.PREV_SCRPTR NEQA 0)
;  810          1380  6                          THEN
```

```
;   811     1381  7                         BEGIN
;   812     1382  7
;   813     1383  8                         IF ((.PREV_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0)
;   814     1384  7                         THEN
;   815     1385  7                             REC_NO = .REC_NO - 1;
;   816     1386  7
;   817     1387  6                         END;
;   818     1388  6
;   819     1389  5                     END;
;   820     1390  5
;   821     1391  5                 SCRPTR = .SCRPTR [SCR_PRV_LINE];
;   822     1392  4                 END;
;   823     1393  4
;   824     1394  4             OLD_TOP_RECNO = .REC_NO;
;   825     1395  4 !+
;   826     1396  4 ! If we didn't find it, rebuild the screen data base.
;   827     1397  4 !-
;   828     1398  4
;   829     1399  4             IF (.SCRPTR NEQA .EDT$$A_TOP_SCRPTR) THEN BUILD_SCR = 1;
;   830     1400  4
;   831     1401  3             END;
;   832     1402  3
;   833     1403  2         END;
;   834     1404  2
;   835     1405  3     IF ((.ANY_CHANGES OR .SCROLLING_NEEDED) AND ( NOT .BUILD_SCR))
;   836     1406  2     THEN
;   837     1407  2 !+
;   838     1408  2 ! Update the lines which are on the screen.  This is needed even if no lines have been changed if
;   839     1409  2 ! we must do scrolling, in order to compute the relative record number of the bottom line.
;   840     1410  2 !-
;   841     1411  3         BEGIN
;   842     1412  3
;   843     1413  3         LOCAL
;   844     1414  3             UPDATE_DONE,
;   845     1415  3             ANOTHER_PASS,
;   846     1416  3             BEG_SCRPTR : REF SCREEN_LINE,
;   847     1417  3             INS_COUNT,
;   848     1418  3             PREV_INS_COUNT;
;   849     1419  3
;   850     1420  3 !+
;   851     1421  3 ! Check for regions containing an equal number of inserted and deleted lines.
;   852     1422  3 ! Avoid double scrolling (and scrolling lines off the screen then back on) by
;   853     1423  3 ! changing all inserted lines in such regions into modified lines, and freeing
;   854     1424  3 ! the deleted lines.
;   855     1425  3 !-
;   856     1426  3
;   857     1427  4         IF (.EDT$$G_RECS_INSERTED NEQ 0)
;   858     1428  3         THEN
;   859     1429  4             BEGIN
;   860     1430  4
;   861     1431  4             DO
;   862     1432  5                 BEGIN
;   863     1433  5                 ANOTHER_PASS = 0;
;   864     1434  5                 INS_COUNT = 0;
;   865     1435  5                 SCRPTR = .EDT$$A_TOP_SCRPTR;
;   866     1436  5
;   867     1437  5                 DO
```

```
;   868          1438  6                       BEGIN
;   869          1439  6                       UPDATE_DONE = 0;
;   870          1440  6                       PREV_INS_COUNT = .INS_COUNT;
;   871          1441  6
;   872          1442  6                       IF ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_INSLN) NEQ 0) THEN INS_COUNT = .INS_COUNT + 1
;   873          1443  6
;   874          1444  6                       IF ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0) THEN INS_COUNT = .INS_COUNT - 1
;   875          1445  6
;   876          1446  6                       IF ((.INS_COUNT NEQ 0) AND (.PREV_INS_COUNT EQL 0)) THEN BEG_SCRPTR = .SCRPTR;
;   877          1447  6
;   878          1448  7                       IF ((.INS_COUNT EQL 0) AND (.PREV_INS_COUNT NEQ 0))
;   879          1449  6                       THEN
;   880          1450  7                           BEGIN
;   881          1451  7 !+
;   882          1452  7 ! Move the top line down, if it was deleted.
;   883          1453  7 !-
;   884          1454  7
;   885          1455  8                           IF (.BEG_SCRPTR EQLA .EDT$$A_TOP_SCRPTR)
;   886          1456  7                           THEN
;   887          1457  7
;   888          1458  7                               WHILE ((.EDT$$A_TOP_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0) DO
;   889          1459  7                                   EDT$$A_TOP_SCRPTR = .EDT$$A_TOP_SCRPTR [SCR_NXT_LINE];
;   890          1460  7
;   891          1461  7 !+
;   892          1462  7 ! Free deleted lines and mark all other lines to be repainted.
;   893          1463  7 !-
;   894          1464  7                           EDT$$SC_REPAINT (.BEG_SCRPTR, .SCRPTR, 0, 255, 1);
;   895          1465  7                           UPDATE_DONE = 1;
;   896          1466  7                           ANOTHER_PASS = 1;
;   897          1467  7                           END
;   898          1468  6                       ELSE
;   899          1469  7                           BEGIN
;   900          1470  7
;   901          1471  8                           IF (.SCRPTR EQLA .EDT$$A_BOT_SCRPTR) OR (.SCRPTR [SCR_NXT_LINE] EQLA 0)
;   902          1472  7                           THEN
;   903          1473  7                               UPDATE_DONE = 1
;   904          1474  7                           ELSE
;   905          1475  7                               SCRPTR = .SCRPTR [SCR_NXT_LINE];
;   906          1476  7
;   907          1477  6                           END;
;   908          1478  6
;   909          1479  6                       END
;   910          1480  5                   UNTIL .UPDATE_DONE;
;   911          1481  5
;   912          1482  5                   END
;   913          1483  4               UNTIL ( NOT .ANOTHER_PASS);
;   914          1484  4
;   915          1485  4 !+
;   916          1486  4 ! If more than 2/3 of the lines on the screen are to be deleted or inserted,
;   917          1487  4 ! just repaint the screen; repainting is likely to be faster.
;   918          1488  4 !-
;   919          1489  4
;   920          1490  4               IF (ABS (.INS_COUNT) GTR ((2*.EDT$$G_SCR_LNS)/3)) THEN BUILD_SCR = 1;
;   921          1491  4
;   922          1492  3               END;
;   923          1493  3
;   924          1494  2           END;
```

```
;  925          1495  2
;  926          1496  3      IF ((.ANY_CHANGES OR .SCROLLING_NEEDED) AND ( NOT .BUILD_SCR))
;  927          1497  2      THEN
;  928          1498  3          BEGIN
;  929          1499  3  !+
;  930          1500  3  ! Now repaint all the lines so marked, and do any residual inserts and deletes on the screen.
;  931          1501  3  !-
;  932          1502  3
;  933          1503  3          LOCAL
;  934          1504  3              STATUS,                             ! Insert or Delete status
;  935          1505  3              ANOTHER_PASS,                       ! 1 = make another pass over the data structure
;  936          1506  3              UPDATE_DONE,                        ! 1 = this pass is complete
;  937          1507  3              ANY_INS_DEL,                        ! 1 = some inserts or deletes done during this pass
;  938          1508  3              INS_DEL_DONE;                       ! 1 = all inserts and deletes are complete
;  939          1509  3
;  940          1510  3  !+
;  941          1511  3  ! If no records were inserted or deleted then no screen lines should need to be inserted or deleted.
;  942          1512  3  !-
;  943          1513  3          INS_DEL_DONE = (.EDT$$G_RECS_INSERTED EQL 0);
;  944          1514  3
;  945          1515  3          DO
;  946          1516  4              BEGIN
;  947          1517  4              ANOTHER_PASS = 0;
;  948          1518  4              ANY_INS_DEL = 0;
;  949          1519  4              REC_NO = .OLD_TOP_RECNO;
;  950          1520  4              SCRPTR = .EDT$$A_TOP_SCRPTR;
;  951          1521  4              EDT$$G_CS_LNO = 0;
;  952          1522  4
;  953          1523  4              DO
;  954          1524  5                  BEGIN
;  955          1525  5                  UPDATE_DONE = 0;
;  956          1526  5
;  957          1527  7                  IF ((.SCRPTR [SCR_EDIT_FLAGS] AND       !
;  958          1528  6                      (SCR_EDIT_MODIFY OR SCR_EDIT_INSLN OR SCR_EDIT_DELLN)) NEQ 0)
;  959          1529  5                  THEN
;  960          1530  6                      BEGIN
;  961          1531  6  !+
;  962          1532  6  ! Is this a deleted line?
;  963          1533  6  !-
;  964          1534  6
;  965          1535  7                      IF ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)
;  966          1536  6                      THEN
;  967          1537  7                          BEGIN
;  968          1538  7                          ASSERT ( NOT .INS_DEL_DONE);
;  969          1539  7                          STATUS = DELETE_LINE (.SCRPTR, OLD_TOP_RECNO);
;  970          1540  7                          ANY_INS_DEL = 1;
;  971          1541  7                          ASSERT ( NOT .STATUS);
;  972          1542  7                          END
;  973          1543  6                      ELSE
;  974          1544  6
;  975          1545  7                          IF ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_INSLN) NEQ 0)
;  976          1546  6                          THEN
;  977          1547  7                              BEGIN
;  978          1548  7                              ASSERT ( NOT .INS_DEL_DONE);
;  979          1549  7                              STATUS = INSERT_LINE (.SCRPTR, .REC_NO, OLD_TOP_RECNO);
;  980          1550  7                              ANY_INS_DEL = 1;
;  981          1551  7  !+
```

```
;  982     1552  7 ! If INSERT_LINE returns true, repaint the current line.  INSERT_LINE will have created a blank
;  983     1553  7 ! line for it.
;  984     1554  7 !-
;  985     1555  7
;  986     1556  7                                 IF .STATUS
;  987     1557  7                                 THEN
;  988     1558  8                                     BEGIN
;  989     1559  8                                     ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  990     1560  8                                     EDT$$SC_REPAINT (.SCRPTR, .SCRPTR, 0, 255, 1);
;  991     1561  8                                     EDT$$SC_RFRELN (.SCRPTR, 1);
;  992     1562  7                                     END;
;  993     1563  7
;  994     1564  7                                 END
;  995     1565  6                             ELSE
;  996     1566  7                                 BEGIN
;  997     1567  7 !+
;  998     1568  7 ! This line is marked as modified.  If all the processing from lines inserted and deleted is complete
;  999     1569  7 ! repaint this line.
; 1000     1570  7 !-
; 1001     1571  7
; 1002     1572  7                                 IF .INS_DEL_DONE
; 1003     1573  7                                 THEN
; 1004     1574  8                                     BEGIN
; 1005     1575  8                                     ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
; 1006     1576  8                                     EDT$$SC_RFRELN (.SCRPTR, .ERASE_ALL);
; 1007     1577  7                                     END;
; 1008     1578  7
; 1009     1579  7                                 STATUS = 1;
; 1010     1580  6                                 END;
; 1011     1581  6
; 1012     1582  7                         IF ( NOT .STATUS)
; 1013     1583  6                         THEN
; 1014     1584  7                             BEGIN
; 1015     1585  7                             UPDATE_DONE = 1;
; 1016     1586  7                             ANOTHER_PASS = 1;
; 1017     1587  6                             END;
; 1018     1588  6
; 1019     1589  5                         END;
; 1020     1590  5
; 1021     1591  6                     IF ( NOT .UPDATE_DONE)
; 1022     1592  5                     THEN
; 1023     1593  6                         BEGIN
; 1024     1594  6                         EDT$$G_CS_LNO = .EDT$$G_CS_LNO + 1;
; 1025     1595  6
; 1026     1596  7                         IF (.EDT$$G_CS_LNO EQL .EDT$$G_SCR_LNS)
; 1027     1597  6                         THEN
; 1028     1598  6                             UPDATE_DONE = 1
; 1029     1599  6                         ELSE
; 1030     1600  7                             BEGIN
; 1031     1601  7
; 1032     1602  8                             IF (.SCRPTR [SCR_NXT_LINE] EQLA 0)
; 1033     1603  7                             THEN
; 1034     1604  8                                 BEGIN
; 1035     1605  8 !+
; 1036     1606  8 ! We have run out of screen data base, but we have not yet filled the screen.  If we
; 1037     1607  8 ! are at [EOB] that is OK, otherwise extend the screen data base.
; 1038     1608  8 !-
```

```
; 1039          1609  8
; 1040          1610  9                                 IF (.SCRPTR EQLA .EDT$$A_EOB_SCRPTR)
; 1041          1611  8                                 THEN
; 1042          1612  9                                     BEGIN
; 1043          1613  9 !+
; 1044          1614  9 ! We have reached [EOB] before filling the screen.  This will be fixed by scrolling later,
; 1045          1615  9 ! if that is possible.  Erase the rest of the screen unless the whole screen has been
; 1046          1616  9 ! erased already.
; 1047          1617  9 !-
; 1048          1618  9
; 1049          1619  9                                     IF (.EDT$$G_CS_LNO LSS .EDT$$G_BOT_LINE) THEN EDT$$SC_ERAALL ();
; 1050          1620  9
; 1051          1621  9                                     UPDATE_DONE = 1
; 1052          1622  9                                     END
; 1053          1623  8                                 ELSE
; 1054          1624  9                                     BEGIN
; 1055          1625  9 !+
; 1056          1626  9 ! We are not at [EOB].  Add another record to the screen data base, and keep
; 1057          1627  9 ! painting the screen.
; 1058          1628  9 !-
; 1059          1629  9                                     REC_NO = .REC_NO + 1;
; 1060          1630  9                                     ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
; 1061          1631  9                                     LNINS_VAL = EDT$$SC_LNINS (0, EDT$$A_WK_LN [LIN_TEXT],
; 1062          1632  9                                         .EDT$$A_WK_LN [LIN_LENGTH]);
; 1063          1633  9
; 1064          1634 10                                     IF (.LNINS_VAL EQL 0)
; 1065          1635  9                                     THEN
; 1066          1636  9                                         UPDATE_DONE = 1
; 1067          1637  9                                     ELSE
; 1068          1638 10                                         BEGIN
; 1069          1639 10                                         SCRPTR = .SCRPTR;
; 1070          1640 10                                         ASSERT (.SCRPTR [SCR_NXT_LINE] NEQA 0);
; 1071          1641 10                                         SCRPTR = .SCRPTR [SCR_NXT_LINE];
; 1072          1642 10                                         ASSERT (.SCRPTR [SCR_LINE_IDX] EQL 0);
; 1073          1643 10                                         SCRPTR [SCR_EDIT_FLAGS] = SCR_EDIT_MODIFY;
; 1074          1644  9                                         END;
; 1075          1645  9
; 1076          1646  8                                     END;
; 1077          1647  8
; 1078          1648  8                             END
; 1079          1649  7                         ELSE
; 1080          1650  8                             BEGIN
; 1081          1651  8
; 1082          1652  8                             LOCAL
; 1083          1653  8                                 NEXT_SCRPTR : REF SCREEN_LINE;
; 1084          1654  8
; 1085          1655  8                             NEXT_SCRPTR = .SCRPTR [SCR_NXT_LINE];
; 1086          1656  8
; 1087          1657  9                             IF (.NEXT_SCRPTR NEQA 0)
; 1088          1658  8                             THEN
; 1089          1659  9                                 BEGIN
; 1090          1660  9
; 1091          1661 10                                 IF (((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0) AND    !
; 1092          1662 11                                     ((.NEXT_SCRPTR [SCR_LINE_IDX] EQL 0) OR      !
; 1093          1663 10                                     (.NEXT_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)))
; 1094          1664  9                                 THEN
; 1095          1665  9                                     REC_NO = .REC_NO + 1;
```

```
;  1096          1666  9
;  1097          1667  8                                    END;
;  1098          1668  8
;  1099          1669  8                                SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1100          1670  7                                END;
;  1101          1671  7
;  1102          1672  6                            END;
;  1103          1673  6
;  1104          1674  5                        END;
;  1105          1675  5
;  1106          1676  5                    END
;  1107          1677  4                UNTIL .UPDATE_DONE;
;  1108          1678  4
;  1109          1679  4 !+
;  1110          1680  4 ! If we have just been looking for inserts and deletes we must make another pass over the screen data base.
;  1111          1681  4 !-
;  1112          1682  4
;  1113          1683  4                IF ( NOT .INS_DEL_DONE) THEN ANOTHER_PASS = 1;
;  1114          1684  4
;  1115          1685  4 !+
;  1116          1686  4 ! If there were no inserts or deletes on this pass then the next pass can look for modified lines.
;  1117          1687  4 !-
;  1118          1688  4
;  1119          1689  5                IF ( NOT .ANY_INS_DEL)
;  1120          1690  4                THEN
;  1121          1691  5                    BEGIN
;  1122          1692  5                    INS_DEL_DONE = 1;
;  1123          1693  5 !+
;  1124          1694  5 ! Make sure the scrolling region is normal for the final update pass.
;  1125          1695  5 !-
;  1126          1696  5                    EDT$$SC_SETSCLLREG (0, .EDT$$G_SCR_LNS);
;  1127          1697  4                    END;
;  1128          1698  4
;  1129          1699  4                END
;  1130          1700  3            UNTIL ( NOT .ANOTHER_PASS);
;  1131          1701  3
;  1132          1702  3            EDT$$G_BOT_LINE = .EDT$$G_CS_LNO;
;  1133          1703  3            OLD_BOT_RECNO = .REC_NO;
;  1134          1704  3            EDT$$A_BOT_SCRPTR = .SCRPTR;
;  1135          1705  3 !+
;  1136          1706  3 ! The screen is no longer erased.
;  1137          1707  3 ! Painting subsequent lines must blank out the end of a completely painted line,
;  1138          1708  3 ! except in special cases such as scrolling a line onto the screen.
;  1139          1709  3 !-
;  1140          1710  3            ERASE_ALL = 0;
;  1141          1711  2            END;
;  1142          1712  2
;  1143          1713  2        IF .BUILD_SCR
;  1144          1714  2        THEN
;  1145          1715  3            BEGIN
;  1146          1716  3 !+
;  1147          1717  3 ! We must rebuild the screen data base.  Put all the screen line
;  1148          1718  3 ! blocks on the free list.
;  1149          1719  3 !-
;  1150          1720  3
;  1151          1721  4            IF (.EDT$$A_FST_SCRPTR NEQA 0)
;  1152          1722  3            THEN
```

```
;  1153          1723  4              BEGIN
;  1154          1724  4              EDT$$A_LST_SCRPTR [SCR_NXT_LINE] = .EDT$$A_FST_AVLN;
;  1155          1725  4              EDT$$A_FST_AVLN = .EDT$$A_FST_SCRPTR;
;  1156          1726  3              END;
;  1157          1727  3
;  1158          1728  3          EDT$$A_FST_SCRPTR = 0;
;  1159          1729  3          EDT$$A_LST_SCRPTR = 0;
;  1160          1730  3          EDT$$A_TOP_SCRPTR = 0;
;  1161          1731  3          EDT$$A_CUR_SCRPTR = 0;
;  1162          1732  3          EDT$$A_BOT_SCRPTR = 0;
;  1163          1733  3          EDT$$A_EOB_SCRPTR = 0;
;  1164          1734  3          EDT$$G_MEM_CNT = 0;
;  1165          1735  2          END;
;  1166          1736  2
;  1167          1737  2      !+
;  1168          1738  2      ! Align the cursor screen pointer with the current screen pointer.
;  1169          1739  2      !-
;  1170          1740  2          REC_NO = 0;
;  1171          1741  2          ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  1172          1742  2          EDT$$A_SCR_BUF = .EDT$$A_CUR_BUF;
;  1173          1743  2          MOVELINE (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_CUR_SCRLN);
;  1174          1744  2
;  1175          1745  2          IF .BUILD_SCR
;  1176          1746  2          THEN
;  1177          1747  3              BEGIN
;  1178          1748  3              EDT$$SC_LNINS (0, EDT$$A_WK_LN [LIN_TEXT], .EDT$$A_WK_LN [LIN_LENGTH]);
;  1179          1749  3              EDT$$A_CSR_SCRPTR = .EDT$$A_FST_SCRPTR;
;  1180          1750  3
;  1181          1751  4              IF (.EDT$$G_TRUN EQL 0)                    !
;  1182          1752  3              THEN
;  1183          1753  3                  EDT$$A_CUR_SCRPTR = EDT$$SC_FNDREC (.EDT$$G_CS_CHNO, DISP)
;  1184          1754  3              ELSE
;  1185          1755  3                  EDT$$A_CUR_SCRPTR = .EDT$$A_CSR_SCRPTR;
;  1186          1756  3
;  1187          1757  2              END;
;  1188          1758  2
;  1189          1759  2      !+
;  1190          1760  2      ! When we reach this point either the old screen has been updated, if necessary,
;  1191          1761  2      ! or we will be rebuilding the screen data base.
;  1192          1762  2      ! Determine which line should be at the top of the screen
;  1193          1763  2      !-
;  1194          1764  2
;  1195          1765  3          IF (.SCROLLING_NEEDED OR .BUILD_SCR)
;  1196          1766  2          THEN
;  1197          1767  3              BEGIN
;  1198          1768  3
;  1199          1769  3              LOCAL
;  1200          1770  3                  AT_BOTTOM;
;  1201          1771  3
;  1202          1772  3              SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1203          1773  3              REC_NO = 0;
;  1204          1774  3              BELOW = 0;
;  1205          1775  3              ABOVE = 0;
;  1206          1776  3      !+
;  1207          1777  3      ! Calculate the number of lines above and below the current line
;  1208          1778  3      ! because we might have to move the cursor.  This may cause the screen
;  1209          1779  3      ! data structure to be extended.
```

```
;  1210          1780  3 !-
;  1211          1781  3          AT_BOTTOM = 0;
;  1212          1782  3
;  1213          1783  3          WHILE ((.BELOW LSS (.EDT$$G_SCR_LNS*2)) AND ( NOT .AT_BOTTOM)) DO
;  1214          1784  4              BEGIN
;  1215          1785  4
;  1216          1786  5              IF (.SCRPTR EQLA 0)
;  1217          1787  4              THEN
;  1218          1788  5                  BEGIN
;  1219          1789  5                  ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  1220          1790  5                  LNINS_VAL = EDT$$SC_LNINS (0, EDT$$A_WK_LN [LIN_TEXT], .EDT$$A_WK_LN [LIN_LENGTH]);
;  1221          1791  5
;  1222          1792  6                  IF (.LNINS_VAL EQL 0)
;  1223          1793  5                  THEN
;  1224          1794  5                      AT_BOTTOM = 1
;  1225          1795  5                  ELSE
;  1226          1796  6                      BEGIN
;  1227          1797  6                      BELOW = .BELOW + .LNINS_VAL;
;  1228          1798  6                      REC_NO = .REC_NO + 1;
;  1229          1799  6
;  1230          1800  7                      IF (.EDT$$A_WK_LN EQLA EDT$$Z_EOB_LN)
;  1231          1801  6                      THEN
;  1232          1802  7                          BEGIN
;  1233          1803  7                          EDT$$A_EOB_SCRPTR = .EDT$$A_LST_SCRPTR;
;  1234          1804  7                          AT_BOTTOM = 1;
;  1235          1805  6                          END;
;  1236          1806  6
;  1237          1807  5                      END;
;  1238          1808  5
;  1239          1809  5                  END
;  1240          1810  4              ELSE
;  1241          1811  5                  BEGIN
;  1242          1812  5                  BELOW = .BELOW + 1;
;  1243          1813  5
;  1244          1814  6                  IF (.SCRPTR EQLA .EDT$$A_EOB_SCRPTR)
;  1245          1815  5                  THEN
;  1246          1816  5                      AT_BOTTOM = 1
;  1247          1817  5                  ELSE
;  1248          1818  6                      BEGIN
;  1249          1819  6                      SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1250          1820  6
;  1251          1821  7                      IF (.SCRPTR EQLA 0)
;  1252          1822  6                      THEN
;  1253          1823  6                          REC_NO = .REC_NO + 1
;  1254          1824  6                      ELSE
;  1255          1825  6
;  1256          1826  6                          IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO + 1;
;  1257          1827  6
;  1258          1828  5                      END;
;  1259          1829  5
;  1260          1830  4                  END;
;  1261          1831  4
;  1262          1832  3              END;
;  1263          1833  3
;  1264          1834  3 !+
;  1265          1835  3 ! Now see how many lines are available above the current line.
;  1266          1836  3 !-
```

```
;  1267          1837  3          SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1268          1838  3          REC_NO = 0;
;  1269          1839  4          BEGIN
;  1270          1840  4
;  1271          1841  4          LOCAL
;  1272          1842  4              AT_TOP;
;  1273          1843  4
;  1274          1844  4          AT_TOP = 0;
;  1275          1845  4
;  1276          1846  4          WHILE ((.ABOVE LSS (.EDT$$G_SCR_LNS*2)) AND ( NOT .AT_TOP)) DO
;  1277          1847  5              BEGIN
;  1278          1848  5
;  1279          1849  6              IF (.SCRPTR NEQA 0)
;  1280          1850  5              THEN
;  1281          1851  6                  BEGIN
;  1282          1852  6
;  1283          1853  6                  IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO - 1;
;  1284          1854  6
;  1285          1855  6                  SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1286          1856  5                  END;
;  1287          1857  5
;  1288          1858  5 !+
;  1289          1859  5 ! If the screen data structure ends, try to add new items to the front of it.
;  1290          1860  5 !-
;  1291          1861  5
;  1292          1862  6              IF (.SCRPTR EQLA 0)
;  1293          1863  5              THEN
;  1294          1864  6                  BEGIN
;  1295          1865  6
;  1296          1866  6                  IF EDT$$SC_MOVTOLN (.REC_NO)
;  1297          1867  6                  THEN
;  1298          1868  7                      BEGIN
;  1299          1869  7                      LNINS_VAL = EDT$$SC_LNINS (.EDT$$A_FST_SCRPTR, EDT$$A_WK_LN [LIN_TEXT],
;  1300          1870  7                          .EDT$$A_WK_LN [[IN_LENGTH]);
;  1301          1871  7
;  1302          1872  8                      IF (.LNINS_VAL EQL 0)
;  1303          1873  7                      THEN
;  1304          1874  7                          AT_TOP = 1
;  1305          1875  7                      ELSE
;  1306          1876  8                          BEGIN
;  1307          1877  8                          ABOVE = .ABOVE + .LNINS_VAL;
;  1308          1878  8                          SCRPTR = .EDT$$A_FST_SCRPTR;
;  1309          1879  7                          END;
;  131C          1880  7
;  1311          1881  7                      END
;  1312          1882  6                  ELSE
;  1313          1883  6                      AT_TOP = 1;
;  1314          1884  6
;  1315          1885  6                  END
;  1316          1886  5              ELSE
;  1317          1887  5                  ABOVE = .ABOVE + 1;
;  1318          1888  5
;  1319          1889  4              END;
;  1320          1890  4
;  1321          1891  3          END;
;  1322          1892  3          SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1323          1893  3 !+
```

```
;  1324     1894  3 ! Now compute the top line.  If there is an enforced top line, we try to use it.
;  1325     1895  3 ! If there is not, we try to use the old top line.  Otherwise we go up a number of
;  1326     1896  3 ! lines depending on the direction of the last move, to preserve as much context
;  1327     1897  3 ! as possible.
;  1328     1898  3 !-
;  1329     1899  3         TOP_SET = 0;
;  1330     1900  3
;  1331     1901  4         IF ( NOT LINNOEQL (EDT$$L_TOP_LN, EDT$$L_LNO_EMPTY))
;  1332     1902  3         THEN
;  1333     1903  4             BEGIN
;  1334     1904  4 !+
;  1335     1905  4 ! There is a request for a top line.  If it is below the current line, reject it.
;  1336     1906  4 !-
;  1337     1907  4
;  1338     1908  5             IF (CMPLNO (EDT$$L_TOP_LN, EDT$$L_CS_LN) GTR 0)
;  1339     1909  4             THEN
;  1340     1910  5                 MOVELINE (EDT$$L_LNO_EMPTY, EDT$$L_TOP_LN)
;  1341     1911  4             ELSE
;  1342     1912  5                 BEGIN
;  1343     1913  5 !+
;  1344     1914  5 ! The requested top line is above or on the current line.  If it is too far above, reject it.
;  1345     1915  5 !-
;  1346     1916  5                 TOP_DIST = 0;
;  1347     1917  5                 SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1348     1918  5                 MOVELINE (EDT$$L_CS_LN, TEMP_LINE);
;  1349     1919  5
;  1350     1920  6                 WHILE (( NOT LINNOEQL (TEMP_LINE, EDT$$L_TOP_LN)) AND   !
;  1351     1921  6                     (.TOP_DIST LSS .EDT$$G_SCLL_BOT) AND        !
;  1352     1922  5                     (.SCRPTR NEQA 0)) DO
;  1353     1923  6                     BEGIN
;  1354     1924  6
;  1355     1925  6                     IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN SUBLINE (NUMBER_ONE, TEMP_LINE);
;  1356     1926  6
;  1357     1927  6                     SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1358     1928  6
;  1359     1929  6                     IF (.SCRPTR NEQA 0) THEN TOP_DIST = .TOP_DIST + 1;
;  1360     1930  6
;  1361     1931  5                     END;
;  1362     1932  5
;  1363     1933  5 !+
;  1364     1934  5 ! If we found the line and it would not require [EOB] to be above the bottom
;  1365     1935  5 ! of the screen, accept it.
;  1366     1936  5 !-
;  1367     1937  5
;  1368     1938  6                 IF (LINNOEQL (TEMP_LINE, EDT$$L_TOP_LN) AND ((.BELOW + .TOP_DIST) GEQ .EDT$$G_SCR_LNS))
;  1369     1939  5                 THEN
;  1370     1940  5                     TOP_SET = 1
;  1371     1941  5                 ELSE
;  1372     1942  5                     MOVELINE (EDT$$L_LNO_EMPTY, EDT$$L_TOP_LN);
;  1373     1943  5
;  1374     1944  4                 END;
;  1375     1945  4
;  1376     1946  3             END;
;  1377     1947  3
;  1378     1948  3 !+
;  1379     1949  3 ! If we have no top determined yet, try to use the old top.
;  1380     1950  3 !-
```

```
;  1381          1951  3
;  1382          1952  4          IF (( NOT .TOP_SET) AND (.EDT$$A_TOP_SCRPTR NEQA 0))
;  1383          1953  3          THEN
;  1384          1954  4              BEGIN
;  1385          1955  4              SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1386          1956  4              REC_NO = 0;
;  1387          1957  4              TOP_DIST = 0;
;  1388          1958  4
;  1389          1959  5              WHILE ((.TOP_DIST LEQ .EDT$$G_SCLL_BOT) AND          !
;  1390          1960  5                  (.SCRPTR NEQA .EDT$$A_TOP_SCRPTR) AND    !
;  1391          1961  4                  (.SCRPTR NEQA 0)) DO
;  1392          1962  5                  BEGIN
;  1393          1963  5
;  1394          1964  5                  IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO - 1;
;  1395          1965  5
;  1396          1966  5                  SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1397          1967  5
;  1398          1968  5                  IF (.SCRPTR NEQA 0) THEN TOP_DIST = .TOP_DIST + 1;
;  1399          1969  5
;  1400          1970  4                  END;
;  1401          1971  4
;  1402          1972  4 !+
;  1403          1973  4 ! If we found the old top line and it will leave the cursor line in range
;  1404          1974  4 ! and not put the [EOB] above the bottom of the screen, use it.
;  1405          1975  4 !-
;  1406          1976  4
;  1407          1977  5              IF ((.TOP_DIST LEQ .EDT$$G_SCLL_BOT) AND     !
;  1408          1978  5                  (.TOP_DIST GEQ .EDT$$G_SCLL_TOP) AND     !
;  1409          1979  5                  ((.BELOW + .TOP_DIST) GEQ .EDT$$G_SCR_LNS) AND  !
;  1410          1980  5                  (.SCRPTR EQLA .EDT$$A_TOP_SCRPTR))
;  1411          1981  4              THEN
;  1412          1982  4                  TOP_SET = 1;
;  1413          1983  4
;  1414          1984  3              END;
;  1415          1985  3
;  1416          1986  3 !+
;  1417          1987  3 ! If top is still not set and there is a record of a previous cursor line
;  1418          1988  3 ! and we are rebuilding the screen data base, try to compute the top line
;  1419          1989  3 ! such that the cursor stays where it was.  This is useful in case the code
;  1420          1990  3 ! for fixing notruncated lines must force a rebuild of the screen data base.
;  1421          1991  3 ! Try to keep the cursor in proper boundries.
;  1422          1992  3 !-
;  1423          1993  3
;  1424          1994  4          IF (( NOT .TOP_SET) AND .EDT$$G_SCR_REBUILD)
;  1425          1995  3          THEN
;  1426          1996  4              BEGIN
;  1427          1997  4
;  1428          1998  4              LOCAL
;  1429          1999  4                  TARGET_LINE;
;  1430          2000  4
;  1431          2001  4              TARGET_LINE = MAX (MIN (.CURSOR_LINE, .EDT$$G_SCLL_BOT), .EDT$$G_SCLL_TOP);
;  1432          2002  4              SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1433          2003  4              TOP_DIST = -1;
;  1434          2004  4
;  1435          2005  4              WHILE ((.SCRPTR NEQA 0) AND (.TOP_DIST NEQ .TARGET_LINE)) DO
;  1436          2006  5                  BEGIN
;  1437          2007  5                  SCRPTR = .SCRPTR [SCR_PRV_LINE];
```

```
;  1438          2008  5                  TOP_DIST = .TOP_DIST + 1;
;  1439          2009  4                  END;
;  1440          2010  4
;  1441          2011  5              IF ((.TOP_DIST EQL .TARGET_LINE) AND ((.BELOW + .TARGET_LINE) GEQ .EDT$$G_SCR_LNS))
;  1442          2012  4              THEN
;  1443          2013  4                  TOP_SET = 1;
;  1444          2014  4
;  1445          2015  3              END;
;  1446          2016  3
;  1447          2017  3 !+
;  1448          2018  3 ! If top is still not set, try to find a new top line a suitable distance
;  1449          2019  3 ! above the current line.
;  1450          2020  3 !-
;  1451          2021  3
;  1452          2022  4          IF ( NOT .TOP_SET)
;  1453          2023  3          THEN
;  1454          2024  4              BEGIN
;  1455          2025  4
;  1456          2026  4              LOCAL
;  1457          2027  4                  TARGET_LINE;
;  1458          2028  4
;  1459          2029  4 !+
;  1460          2030  4 ! Work back until the beginning of the screen data structure or until TOP_DIST is
;  1461          2031  4 ! big enough for the direction we are moving.
;  1462          2032  4 ! The (.CURSOR_LINE + .EDT$$G_RECS_INSERTED) code is here to fix a problem on VT52's
;  1463          2033  4 ! with the screen scrolling too far up on a paste.
;  1464          2034  4 !-
;  1465          203?  4
;  1466          2036  7              IF ((.DIR GEQ 0) OR (((.CURSOR_LINE + .EDT$$G_RECS_INSERTED) GTR .EDT$$G_SCLL_BOT)  !
;  1467          2037  5                  AND (.EDT$$G_RECS_INSERTED GTR 0)))
;  1468          2038  4              THEN
;  1469          2039  4                  TARGET_LINE = .EDT$$G_SCLL_BOT
;  1470          2040  4              ELSE
;  1471          2041  4                  TARGET_LINE = .EDT$$G_SCLL_TOP;
;  1472          2042  4
;  1473          2043  4 !+
;  1474          2044  4 ! If necessary, work back further to avoid lifting the [EOB] above the last line
;  1475          2045  4 ! of the screen.
;  1476          2046  4 !-
;  1477          2047  4              TARGET_LINE = MAX (.TARGET_LINE, .EDT$$G_SCR_LNS - .BELOW);
;  1478          2048  4 !
;  1479          2049  4              REC_NO = 0;
;  1480          2050  4              SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1481          2051  4              TOP_DIST = -1;
;  1482          2052  4
;  1483          2053  4              WHILE ((.SCRPTR NEQA 0) AND (.TOP_DIST NEQ .TARGET_LINE)) DO
;  1484          2054  5                  BEGIN
;  1485          2055  5                  SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1486          2056  5                  TOP_DIST = .TOP_DIST + 1;
;  1487          2057  4                  END;
;  1488          2058  4
;  1489          2059  4 !+
;  1490          2060  4 ! If we found the line we were looking for, accept it.
;  1491          2061  4 !-
;  1492          2062  4
;  1493          2063  4              IF (.TOP_DIST EQL .TARGET_LINE) THEN TOP_SET = 1;
;  1494          2064  4
```

```
;  1495          2065  3              END;
;  1496          2066  3
;  1497          2067  3          !+
;  1498          2068  3          ! If no line is suitable, use the first line in the screen data base.  This can happen when we
;  1499          2069  3          ! have a buffer that fits on the screen.
;  1500          2070  3          !-
;  1501          2071  3
;  1502          2072  4              IF ( NOT .TOP_SET)
;  1503          2073  3              THEN
;  1504          2074  4                  BEGIN
;  1505          2075  4                  TOP_DIST = -1;
;  1506          2076  4                  SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1507          2077  4
;  1508          2078  4                  WHILE (.SCRPTR NEQA 0) DO
;  1509          2079  5                      BEGIN
;  1510          2080  5                      TOP_DIST = .TOP_DIST + 1;
;  1511          2081  5                      SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1512          2082  4                      END;
;  1513          2083  4
;  1514          2084  3                  END;
;  1515          2085  3
;  1516          2086  3          !+
;  1517          2087  3          ! Now that TOP_DIST is computed, find the new top screen pointer.
;  1518          2088  3          !-
;  1519          2089  3              SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1520          2090  3              REC_NO = 0;
;  1521          2091  3
;  1522          2092  3              INCR I FROM 1 TO .TOP_DIST DO
;  1523          2093  4                  BEGIN
;  1524          2094  4
;  1525          2095  4                  IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO - 1;
;  1526          2096  4
;  1527          2097  4                  SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1528          2098  3                  END;
;  1529          2099  3
;  1530          2100  3              TOP_RECNO = .REC_NO;
;  1531          2101  3              TOP_SCRPTR = .SCRPTR;
;  1532          2102  3          !+
;  1533          2103  3          ! Compute the number of lines between the old and new top lines,
;  1534          2104  3          ! so we can see how far to scroll, and in which direction.
;  1535          2105  3          !-
;  1536          2106  3
;  1537          2107  4              IF ((.EDT$$A_TOP_SCRPTR NEQA .TOP_SCRPTR) AND (.EDT$$A_TOP_SCRPTR NEQA 0))
;  1538          2108  3              THEN
;  1539          2109  4                  BEGIN
;  1540          2110  4
;  1541          2111  4                  LOCAL
;  1542          2112  4                      SEEN_OLD,
;  1543          2113  4                      SEEN_NEW;
;  1544          2114  4
;  1545          2115  4                  SEEN_OLD = 0;
;  1546          2116  4                  SEEN_NEW = 0;
;  1547          2117  4                  SCLL_NUM = 0;
;  1548          2118  4                  SCRPTR = .EDT$$A_FST_SCRPTR;
;  1549          2119  4
;  1550          2120  4                  WHILE ((.SCRPTR NEQA 0) AND ( NOT (.SEEN_OLD AND .SEEN_NEW))) DO
;  1551          2121  5                      BEGIN
```

```
;  1552          2122  5
;  1553          2123  5                    IF (.SEEN_OLD AND ( NOT .SEEN_NEW)) THEN SCLL_NUM = .SCLL_NUM + 1;
;  1554          2124  5
;  1555          2125  5                    IF (.SEEN_NEW AND ( NOT .SEEN_OLD)) THEN SCLL_NUM = .SCLL_NUM - 1;
;  1556          2126  5
;  1557          2127  5                    IF (.SCRPTR EQLA .TOP_SCRPTR) THEN SEEN_NEW = 1;
;  1558          2128  5
;  1559          2129  5                    IF (.SCRPTR EQLA .EDT$$A_TOP_SCRPTR) THEN SEEN_OLD = 1;
;  1560          2130  5
;  1561          2131  5                    SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1562          2132  4                    END;
;  1563          2133  4
;  1564          2134  4                ASSERT (.SEEN_NEW);
;  1565          2135  4 !+
;  1566          2136  4 ! If the old top line is not in the data base, it must be too far away
;  1567          2137  4 ! to scroll.
;  1568          2138  4 !-
;  1569          2139  4
;  1570          2140  4                IF ( NOT .SEEN_OLD) THEN SCLL_NUM = 0;
;  1571          2141  4
;  1572          2142  4 !+
;  1573          2143  4 ! If the amount to scroll is too large, don't do any scrolling.
;  1574          2144  4 !-
;  1575          2145  4
;  1576          2146  4                IF (ABS (.SCLL_NUM) GEQ .EDT$$G_SCR_LNS) THEN SCLL_NUM = 0;
;  1577          2147  4
;  1578          2148  4 !+
;  1579          2149  4 ! The sign of SCLL_NUM says which way to scroll, and the magnitude says
;  1580          2150  4 ! how much.  First position to the bottom or top of the old screen,
;  1581          2151  4 ! depending on which way we are scrolling.
;  1582          2152  4 !-
;  1583          2153  4
;  1584          2154  4                WHILE (.SCLL_NUM NEQ 0) DO
;  1585          2155  5                    BEGIN
;  1586          2156  5
;  1587          2157  6                    IF (.SCLL_NUM GTR 0)
;  1588          2158  5                    THEN
;  1589          2159  6                        BEGIN
;  1590          2160  6 !+
;  1591          2161  6 ! The cursor is moving down, so scroll the screen up.
;  1592          2162  6 !-
;  1593          2163  6                        SCRPTR = .EDT$$A_BOT_SCRPTR;
;  1594          2164  6                        REC_NO = .OLD_BOT_RECNO;
;  1595          2165  6                        SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1596          2166  6
;  1597          2167  6                        IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO + 1;
;  1598          2168  6
;  1599          2169  6                        OLD_BOT_RECNO = .REC_NO;
;  1600          2170  6
;  1601          2171  6                        IF .EDT$$G_TI_SCROLL
;  1602          2172  6                        THEN
;  1603          2173  7                            BEGIN
;  1604          2174  7                            EDT$$SC_POSCSIF (.EDT$$G_SCR_LNS - 1, 0);
;  1605          2175  7                            EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_LF)), 1);
;  1606          2176  7                            END
;  1607          2177  6                        ELSE
;  1608          2178  7                            BEGIN
```

```
;1609   2179  7                       EDT$$SC_POSCSIF (.EDT$$G_MESSAGE_LINE + 1, 0);
;1610   2180  7                       EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_LF)), 1);
;1611   2181  7                       EDT$$SC_POSCSIF (.EDT$$G_SCR_LNS - 1, 0);
;1612   2182  6                       END;
;1613   2183  6
;1614   2184  6 !+
;1615   2185  6 ! If this is a non-scrolling-region terminal, text may have moved down into the message
;1616   2186  6 ! region or a message may have moved up one line.  Erase the message region.
;1617   2187  6 !-
;1618   2188  6
;1619   2189  7                   IF ( NOT .EDT$$G_TI_SCROLL)
;1620   2190  6                   THEN
;1621   2191  7                       BEGIN
;1622   2192  7                       EDT$$G_MSGFLG = 0;      ! Any message is lost
;1623   2193  7                       EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;1624   2194  7                       EDT$$SC_ERAALL ();
;1625   2195  6                       END;
;1626   2196  6
;1627   2197  6                   EDT$$G_CS_LNO = .EDT$$G_SCR_LNS - 1;
;1628   2198  6                   ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;1629   2199  6                   EDT$$SC_RFRELN (.SCRPTR, .EDT$$G_TI_SCROLL);
;1630   2200  6                   EDT$$A_TOP_SCRPTR = .EDT$$A_TOP_SCRPTR [SCR_NXT_LINE];
;1631   2201  6                   EDT$$A_BOT_SCRPTR = .SCRPTR;
;1632   2202  6                   SCLL_NUM = .SCLL_NUM - 1;
;1633   2203  6                   END
;1634   2204  5               ELSE
;1635   2205  6                   BEGIN
;1636   2206  6 !+
;1637   2207  6 ! The cursor is moving up, so scroll the screen down.
;1638   2208  6 !-
;1639   2209  6                   SCRPTR = .EDT$$A_TOP_SCRPTR;
;1640   2210  6                   REC_NO = .OLD_TOP_RECNO;
;1641   2211  6
;1642   2212  6                   IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO - 1;
;1643   2213  6
;1644   2214  6                   OLD_TOP_RECNO = .REC_NO;
;1645   2215  6                   SCRPTR = .SCRPTR [SCR_PRV_LINE];
;1646   2216  6                   EDT$$SC_POSCSIF (0, 0);
;1647   2217  6
;1648   2218  7                   IF (.EDT$$G_TI_TYP EQL TERM_VT52)   !
;1649   2219  6                   THEN
;1650   2220  6                       EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'I')), 2)
;1651   2221  6                   ELSE
;1652   2222  6                       EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'M')), 2);
;1653   2223  6
;1654   2224  6 !+
;1655   2225  6 ! If this is a non-scrolling-region terminal, text may have moved down into the message
;1656   2226  6 ! region or a message may have moved up one line.  Erase the message region.  It is important
;1657   2227  6 ! to erase the message region before painting the new line at the top of the screen.  Because
;1658   2228  6 ! of what appears to be a problem within the VT52, if EDT paints the new line first and then
;1659   2229  6 ! erases the message area, some of the leading characters will sometimes be lost from the
;1660   2230  6 ! newly painted line.  To demonstrate the problem, SET LINES 10 and edit a file of 22 80-character
;1661   2231  6 ! lines on a VT52.  Go to the bottom of the buffer and type GOLD 8 up arrow.  A more extreme
;1662   2232  6 ! case is to SET LINES 5 and type GOLD 4 up arrow.
;1663   2233  6 !-
;1664   2234  6
;1665   2235  7                   IF ( NOT .EDT$$G_TI_SCROLL)
```

```
;  1666     2236  6                        THEN
;  1667     2237  7                            BEGIN
;  1668     2238  7                            EDT$$G_MSGFLG = 0;      ! Any message is lost
;  1669     2239  7                            EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;  1670     2240  7                            EDT$$SC_ERAALL ();
;  1671     2241  6                            END;
;  1672     2242  6
;  1673     2243  6                        EDT$$G_CS_LNO = 0;
;  1674     2244  6                        ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  1675     2245  6                        EDT$$SC_RFRELN (.SCRPTR, 1);
;  1676     2246  6                        EDT$$A_TOP_SCRPTR = .SCRPTR;
;  1677     2247  6                        EDT$$A_BOT_SCRPTR = .EDT$$A_BOT_SCRPTR [SCR_PRV_LINE];
;  1678     2248  6                        SCLL_NUM = .SCLL_NUM + 1;
;  1679     2249  5                        END;
;  1680     2250  5
;  1681     2251  4                    END;
;  1682     2252  4
;  1683     2253  3                END;
;  1684     2254  3
;  1685     2255  3            END                                     ! of .SCROLLING_NEEDED OR .BUILD_SCR
;  1686     2256  2        ELSE
;  1687     2257  3            BEGIN
;  1688     2258  3            TOP_SCRPTR = .EDT$$A_TOP_SCRPTR;
;  1689     2259  3            TOP_RECNO = .EDT$$G_TOP_RECNO;
;  1690     2260  2            END;
;  1691     2261  2
;  1692     2262  2 !+
;  1693     2263  2 ! Make a final update pass over the screen.  This will be needed if
;  1694     2264  2 ! no scrolling took place due to the new screen being too far from
;  1695     2265  2 ! the old screen, or if we erased the screen, or if we are rebuilding
;  1696     2266  2 ! the screen data base from scratch.  If scrolling was needed then
;  1697     2267  2 ! this pass is necessary to recompute CURSOR_LINE, even if no updates
;  1698     2268  2 ! need to be made.
;  1699     2269  2 !-
;  1700     2270  2
;  1701     2271  3        IF (.SCROLLING_NEEDED OR .ERASE_ALL OR .BUILD_SCR)
;  1702     2272  2        THEN
;  1703     2273  3            BEGIN
;  1704     2274  3            REC_NO = .TOP_RECNO;
;  1705     2275  3            SCRPTR = .TOP_SCRPTR;
;  1706     2276  3            CURSOR_LINE = -1;
;  1707     2277  3            EDT$$G_CS_LNO = 0;
;  1708     2278  3
;  1709     2279  3            WHILE ((.EDT$$G_CS_LNO LSS .EDT$$G_SCR_LNS) AND (.SCRPTR NEQA 0)) DO
;  1710     2280  4                BEGIN
;  1711     2281  4
;  1712     2282  5                IF ((.SCRPTR [SCR_EDIT_FLAGS] AND (SCR_EDIT_MODIFY OR SCR_EDIT_INSLN OR SCR_EDIT_DELLN)) NEQ 0)
;  1713     2283  4                THEN
;  1714     2284  5                    BEGIN
;  1715     2285  5                    ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  1716     2286  5                    EDT$$SC_RFRELN (.SCRPTR, .ERASE_ALL);
;  1717     2287  4                    END;
;  1718     2288  4
;  1719     2289  4                EDT$$A_BOT_SCRPTR = .SCRPTR;
;  1720     2290  4
;  1721     2291  4                IF (.SCRPTR EQLA .EDT$$A_CUR_SCRPTR) THEN CURSOR_LINE = .EDT$$G_CS_LNO;
;  1722     2292  4
```

```
;  1723          2293  4              SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1724          2294  4              EDT$$G_CS_LNO = .EDT$$G_CS_LNO + 1;
;  1725          2295  4
;  1726          2296  5              IF (.SCRPTR NEQA 0)
;  1727          2297  4              THEN
;  1728          2298  5                  BEGIN
;  1729          2299  5
;  1730          2300  5                  IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN REC_NO = .REC_NO + 1;
;  1731          2301  5
;  1732          2302  4                  END;
;  1733          2303  4
;  1734          2304  3              END;
;  1735          2305  3
;  1736          2306  3          !+
;  1737          2307  3          ! Be sure that we leave the screen updater positioned on the current record.
;  1738          2308  3          !-
;  1739          2309  3                  REC_NO = 0;
;  1740          2310  3                  ASSERT (EDT$$SC_MOVTOLN (.REC_NO));
;  1741          2311  3          !+
;  1742          2312  3          ! If there is more room on the screen, erase it if necessary.
;  1743          2313  3          !-
;  1744          2314  3
;  1745          2315  3                  IF (.EDT$$G_CS_LNO LSS .EDT$$G_BOT_LINE) THEN EDT$$SC_ERAALL ();
;  1746          2316  3
;  1747          2317  3                  EDT$$G_BOT_LINE = .EDT$$G_CS_LNO;
;  1748          2318  2                  END;
;  1749          2319  2
;  1750          2320  2          !+
;  1751          2321  2          ! Do the clean-up on the screen data pointers.
;  1752          2322  2          !-
;  1753          2323  2              EDT$$G_TOP_RECNO = .TOP_RECNO;
;  1754          2324  2              EDT$$A_TOP_SCRPTR = .TOP_SCRPTR;
;  1755          2325  2              EDT$$G_SCR_REBUILD = 0;
;  1756          2326  2              EDT$$G_ANY_CHANGES = 0;
;  1757          2327  2              EDT$$A_CSR_SCRPTR = .EDT$$A_CUR_SCRPTR;
;  1758          2328  2              EDT$$G_RECS_INSERTED = 0;
;  1759          2329  2              EDT$$A_OLD_SEL = .EDT$$A_SEL_BUF;
;  1760          2330  2              EDT$$G_CS_OLDCHNO = .EDT$$G_CS_CHNO;
;  1761          2331  2              EDT$$SC_POSCSIF (.CURSOR_LINE, .CURSOR_POS);
;  1762          2332  2              EDT$$G_CUR_COL = .CURSOR_POS;
;  1763          2333  2              EDT$$G_CS_LNO = .CURSOR_LINE;
;  1764          2334  2              EDT$$TI_ENBLAUTREP (1);
;  1765          2335  2              EDT$$OUT_FMTBUF ();
;  1766          2336  2              EDT$$A_LN_PTR = .EDT$$G_CS_CHNO + EDT$$T_LN_BUF;
;  1767          2337  1              END;                                    ! of routine EDT$$SC_UPD

                                                        .TITLE  EDT$SCRUPDATE EDT$SCRUPDATE - update the screen
                                                        .IDENT  \V04-000\

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

                                            0A  00000 P.AAA:    .BYTE   10                                      ;
                                                00001           .BLKB   3
                                            0A  00004 P.AAB:    .BYTE   10                                      ;
                                                00005           .BLKB   3
                                        49  1B  00008 P.AAC:    .BYTE   27, 73                                  ;
```

```
                                                 0000A           .BLKB   2
                                         4D  1B  0000C P.AAD:    .BYTE   27, 77                                      ;

                                                                 .EXTRN  EDT$$SC_SETSCLLREG
                                                                 .EXTRN  EDT$$SC_LNINS, EDT$$FMT_LIT
                                                                 .EXTRN  EDT$$SC_FNDREC, EDT$$OUT_FMTBUF
                                                                 .EXTRN  EDT$$RPL_CHGDLN
                                                                 .EXTRN  EDT$$SC_INIT, EDT$$SC_CPUCSPOS
                                                                 .EXTRN  EDT$$SC_POSCSIF
                                                                 .EXTRN  EDT$$SC_ERAALL, EDT$$SC_MOVTOLN
                                                                 .EXTRN  EDT$$SC_RFRELN, EDT$$SC_NONREVID
                                                                 .EXTRN  EDT$$SC_REPAINT
                                                                 .EXTRN  EDT$$TI_ENBLAUTREP
                                                                 .EXTRN  EDT$$FIX_NOTRUNC_NOOVERLAY
                                                                 .EXTRN  EDT$$A_BOT_SCRPTR
                                                                 .EXTRN  EDT$$A_OLD_SEL, EDT$$L_LNO_EMPTY
                                                                 .EXTRN  EDT$$G_SCR_REBUILD
                                                                 .EXTRN  EDT$$A_EOB_SCRPTR
                                                                 .EXTRN  EDT$$G_SCR_CHGD
                                                                 .EXTRN  EDT$$G_MESSAGE_LINE
                                                                 .EXTRN  EDT$$T_LN_BUF, EDT$$A_CSR_SCRPTR
                                                                 .EXTRN  EDT$$A_CUR_SCRPTR
                                                                 .EXTRN  EDT$$A_LST_SCRPTR
                                                                 .EXTRN  EDT$$A_WK_LN, EDT$$Z_EOB_LN
                                                                 .EXTRN  EDT$$A_SCR_BUF, EDT$$A_FST_SCRPTR
                                                                 .EXTRN  EDT$$A_TOP_SCRPTR
                                                                 .EXTRN  EDT$$L_CUR_SCRLN
                                                                 .EXTRN  EDT$$G_CS_CHNO, EDT$$G_CS_OLDCHNO
                                                                 .EXTRN  EDT$$L_CS_LN, EDT$$G_CS_LNO
                                                                 .EXTRN  EDT$$G_CUR_COL, EDT$$G_LN_NO
                                                                 .EXTRN  EDT$$A_SEL_POS, EDT$$L_SEL_LN
                                                                 .EXTRN  EDT$$A_SEL_BUF, EDT$$L_TOP_LN
                                                                 .EXTRN  EDT$$A_CUR_BUF, EDT$$G_SCR_LNS
                                                                 .EXTRN  EDT$$G_SCLL_TOP
                                                                 .EXTRN  EDT$$G_SCLL_BOT
                                                                 .EXTRN  EDT$$G_TI_TYP, EDT$$A_LN_PTR
                                                                 .EXTRN  EDT$$G_TI_SCROLL
                                                                 .EXTRN  EDT$$G_RECS_INSERTED
                                                                 .EXTRN  EDT$$A_FST_AVLN
                                                                 .EXTRN  EDT$$G_TRUN, EDT$$G_MEM_CNT
                                                                 .EXTRN  EDT$$G_BOT_LINE
                                                                 .EXTRN  EDT$$G_TOP_RECNO
                                                                 .EXTRN  EDT$$G_ANY_CHANGES
                                                                 .EXTRN  EDT$$G_REVID, EDT$$G_MSGFLG
                                                                 .EXTRN  EDT$$INTER_ERR

                                          0FFC  00000            .ENTRY  EDT$$SC_UPD, Save R2,R3,R4,R5,R6,R7,R8,R9,- ; 0756
                                                                         R10,R11                                     ;
                    5E                  3C  C2  00002            SUBL2   #60, SP                                     ;
                          00000000G     00  D5  00005            TSTL    EDT$$G_REVID                                ; 0940
                                        14  13  0000B            BEQL    1$                                          ;
           00000000G  00  00000000G     00  D1  0000D            CMPL    EDT$$A_SEL_BUF, EDT$$A_CUR_BUF              ;
                                        07  13  00018            BEQL    1$                                          ;
           00000000G  00                00  FB  0001A            CALLS   #0, EDT$$SC_NONREVID                        ;
                          00000000G     00  D4  00021 1$:        CLRL    EDT$$G_LN_NO                                ; 0946
                    50    00000000G     00  9E  00027            MOVAB   EDT$$T_LN_BUF, R0                           ; 0947
00000000G  00  00000000G  00            50  C3  0002E            SUBL3   R0, EDT$$A_LN_PTR, EDT$$G_CS_CHNO           ;
```

```
                              50 00000000G  00  D0 0003A        MOVL    EDT$$A_CUR_BUF, R0                      ; 0948
        00000000G  00     06  A0            06  28 00041        MOVC3   #6, 6(R0), EDT$$L_CS_LN                  ;
                   00000000G  00            00  FB 0004A        CALLS   #0, EDT$$RPL_CHGDLN                      ; 0949
                              0E 00000000G  00  E8 00051        BLBS    EDT$$G_TRUN, 2$                          ; 0955
                              07 00000000G  00  E9 00058        BLBC    EDT$$G_ANY_CHANGES, 2$                   ; 0959
                   00000000G  00            00  FB 0005F        CALLS   #0, EDT$$FIX_NOTRUNC_NOOVERLAY           ;
                                            57  D4 00066 2$:    CLRL    SCRPTR                                   ; 0963
                                     1C  AE  9F 00068           PUSHAB  CURSOR_POS                               ; 0967
                                     24  AE  9F 0006B           PUSHAB  CURSOR_LINE                              ;
                   00000000G  00            02  FB 0006E        CALLS   #2, EDT$$SC_CPUCSPOS                     ;
                          20  AE 00000000G  00  D0 00075        MOVL    EDT$$G_CS_LNO, CURSOR_LINE               ; 0968
                              51 00000000G  00  D0 0007D        MOVL    EDT$$G_SCR_CHGD, R1                      ; 0974
                                            1E  12 00084        BNEQ    3$                                       ;
                   00000000G  00 00000000G  00  D1 00086        CMPL    EDT$$A_SCR_BUF, EDT$$A_CUR_BUF           ;
                                            11  12 00091        BNEQ    3$                                       ;
                50 00000000G  00            01  78 00093        ASHL    #1, EDT$$G_SCR_LNS, R0                   ; 0975
                              50 00000000G  00  D1 0009B        CMPL    EDT$$G_RECS_INSERTED, R0                 ;
                                            25  15 000A2        BLEQ    5$                                       ;
                              02            51  D1 000A4 3$:    CMPL    R1, #2                                   ; 0982
                                            07  12 000A7        BNEQ    4$                                       ;
                   00000000G  00            00  FB 000A9        CALLS   #0, EDT$$SC_INIT                         ;
                                 00000000G  00  D4 000B0 4$:    CLRL    EDT$$G_CS_LNO                            ; 0987
                   00000000G  00            00  FB 000B6        CALLS   #0, EDT$$SC_ERAALL                       ; 0988
                                 00000000G  00  D4 000BD        CLRL    EDT$$G_BOT_LINE                          ; 0989
                          10  AE            01  D0 000C3        MOVL    #1, ERASE_ALL                            ; 0990
                                            03  11 000C7        BRB     6$                                       ; 0974
                                     10  AE  D4 000C9 5$:       CLRL    ERASE_ALL                                ; 0993
                          0C  AE 00000000G  00  D0 000CC 6$:    MOVL    EDT$$G_SCR_REBUILD, BUILD_SCR            ; 0998
                              36     0C  AE  E8 000D4           BLBS    BUILD_SCR, 8$                            ; 1003
                                     28  AE  9F 000D8           PUSHAB  DISP                                     ; 1006
                              50 00000000G  00  9E 000DB        MOVAB   EDT$$T_LN_BUF, R0                        ;
                7E 00000000G  00            50  C3 000E2        SUBL3   R0, EDT$$A_LN_PTR, -(SP)                 ;
                   00000000G  00            02  FB 000EA        CALLS   #2, EDT$$SC_FNDREC                       ;
                   00000000G  00            50  D0 000F1        MOVL    R0, EDT$$A_CUR_SCRPTR                    ;
                                            10  13 000F8        BEQL    7$                                       ; 1008
                                 00000000G  00  D5 000FA        TSTL    EDT$$A_TOP_SCRPTR                        ;
                                            08  13 00100        BEQL    7$                                       ;
                                 00000000G  00  D5 00102        TSTL    EDT$$A_CSR_SCRPTR                        ; 1009
                                            04  12 00108        BNEQ    8$                                       ;
                          0C  AE            01  D0 0010A 7$:    MOVL    #1, BUILD_SCR                            ; 1011
                              58 00000000G  00  D0 0010E 8$:    MOVL    EDT$$G_ANY_CHANGES, ANY_CHANGES          ; 1018
                              53 00000000G  00  D0 00115        MOVL    LOW_1, R3                                ; 1024
                              51 00000000G  00  D0 0011C        MOVL    LOW_2, R1                                ;
                              51            53  D1 00123        CMPL    R3, R1                                   ;
                                            0D  12 00126        BNEQ    9$                                       ;
                   00000000G  00 00000000G  00  B1 00128        CMPW    HIGH_1, HIGH_2                           ;
                                            3B  13 00133        BEQL    12$                                      ;
                   00000000G  00 00000000G  00  D1 00135 9$:    CMPL    EDT$$A_SCR_BUF, EDT$$A_CUR_BUF           ;
                                            2E  12 00140        BNEQ    12$                                      ;
                              52 00000000G  00  3C 00142        MOVZWL  HIGH_1, R2                               ; 1028
                              50 00000000G  00  3C 00149        MOVZWL  HIGH_2, R0                               ;
                              50            52  D1 00150        CMPL    R2, R0                                   ;
                                            0E  1F 00153        BLSSU   10$                                      ;
                                            19  12 00155        BNEQ    12$                                      ;
                              50 00000000G  00  D0 00157        MOVL    LOW_1, R0                                ;
                              51            50  D1 0015E        CMPL    R0, R1                                   ;
                                            06  1E 00161        BGEQU   11$                                      ;
```

```
             08 AE             01  CE 00163 10$:  MNEGL   #1, DIR                                     ;
                               0B  11 00167       BRB     13$                                         ;
                               05  12 00169 11$:  BNEQ    12$                                         ;
                          08   AE  D4 0016B       CLRL    DIR                                         ;
                               04  11 0016E       BRB     13$                                         ;
             08 AE             01  D0 00170 12$:  MOVL    #1, DIR                                     ;
                   00000000G   00  D5 00174 13$:  TSTL    EDT$$G_RECS_INSERTED                        ; 1034
                               28  12 0017A       BNEQ    14$                                         ;
      00000000G 00 00000000G   00  D1 0017C       CMPL    EDT$$A_CSR_SCRPTR, EDT$$A_CUR_SCRPTR        ; 1035
                               1B  12 00187       BNEQ    14$                                         ;
                53 00000000G   00  D1 00189       CMPL    LOW_1, R3                                   ; 1036
                               12  12 00190       BNEQ    14$                                         ;
      00000000G 00 00000000G   00  B1 00192       CMPW    HIGH_1, HIGH_2                              ;
                               05  12 0019D       BNEQ    14$                                         ;
                          08   AE  D5 0019F       TSTL    DIR                                         ; 1037
                               06  13 001A2       BEQL    15$                                         ;
             14 AE             01  D0 001A4 14$:  MOVL    #1, SCROLLING_NEEDED                        ; 1039
                               03  11 001A8       BRB     16$                                         ;
                          14   AE  D4 001AA 15$:  CLRL    SCROLLING_NEEDED                            ; 1041
                03        0C   AE  E9 001AD 16$:  BLBC    BUILD_SCR, 17$                              ; 1043
                             02C8  31 001B1       BRW     47$                                         ;
                51 00000000G   00  D0 001B4 17$:  MOVL    EDT$$A_SEL_BUF, R1                          ; 1047
                52 00000000G   00  D0 001BB       MOVL    EDT$$A_OLD_SEL, R2                          ;
                52             51  D1 001C2       CMPL    R1, R2                                      ;
                               03  12 001C5       BNEQ    19$                                         ;
                             0184  31 001C7 18$:  BRW     35$                                         ;
                02 00000000G   00  D1 001CA 19$:  CMPL    EDT$$G_TI_TYP, #2                           ;
                               F4  12 001D1       BNEQ    18$                                         ;
                50 00000000G   00  D0 001D3       MOVL    EDT$$A_SCR_BUF, R0                          ; 1048
                52             50  D1 001DA       CMPL    R0, R2                                      ;
                               05  13 001DD       BEQL    20$                                         ;
                51             50  D1 001DF       CMPL    R0, R1                                      ;
                               E3  12 001E2       BNEQ    18$                                         ;
                DF        10   AE  E8 001E4 20$:  BLBS    ERASE_ALL, 18$                              ; 1049
                               51  D5 001E8       TSTL    R1                                          ; 1068
                               42  12 001EA       BNEQ    21$                                         ;
2C AE 00000000G 00             06  28 001EC       MOVC3   #6, EDT$$L_CUR_SCRLN, OUR_LINE              ; 1071
                50        3A   AE  B0 001F5       MOVW    UPPER_WORD, SAVE                            ; 1072
34 AE 00000000G 00 00000000G   00  C3 001F9       SUBL3   EDT$$L_CUR_SCRLN, EDT$$L_CS_LN, TEMP_LINE   ;
             38 AE 00000000G   00  D0 00206       MOVL    EDT$$L_CS_LN+4, TEMP_LINE                   ;
             38 AE 00000000G   00  D9 0020E       SBWC    EDT$$L_CUR_SCRLN+4, TEMP_LINE               ;
             3A AE             50  B0 00216       MOVW    SAVE, UPPER_WORD                            ;
                50        34   AE  32 0021A       CVTWL   TEMP_LINE, REC_OFFSET                       ; 1073
                52 00000000G   00  D0 0021E       MOVL    EDT$$G_CS_OLDCHNO, OUR_CHNO                 ; 1074
                53 00000000G   00  D0 00225       MOVL    EDT$$A_CSR_SCRPTR, OUR_SCRPTR               ; 1075
                               19  11 0022C       BRB     22$                                         ; 1068
2C AE 00000000G 00             06  28 0022E 21$:  MOVC3   #6, EDT$$L_CS_LN, OUR_LINE                  ; 1079
                               50  D4 00237       CLRL    REC_OFFSET                                  ; 1080
                52 00000000G   00  D0 00239       MOVL    EDT$$G_CS_CHNO, OUR_CHNO                    ; 1081
                53 00000000G   00  D0 00240       MOVL    EDT$$A_CUR_SCRPTR, OUR_SCRPTR               ; 1082
                51        3A   AE  B0 00247 22$:  MOVW    UPPER_WORD, SAVE                            ; 1085
34 AE 00000000G 00        2C   AE  C3 0024B       SUBL3   OUR_LINE, EDT$$L_SEL_LN, TEMP_LINE          ;
             38 AE 00000000G   00  D0 00255       MOVL    EDT$$L_SEL_LN+4, TEMP_LINE                  ;
             38 AE        30   AE  D9 0025D       SBWC    OUR_LINE, TEMP_LINE                         ;
             3A AE             51  B0 00262       MOVW    SAVE, UPPER_WORD                            ;
                          38   AE  B5 00266       TSTW    TEMP_LINE+4                                 ; 1087
                               05  18 00269       BGEQ    23$                                         ;
```

```
                    54           01  CE 0026B          MNEGL   #1, SELDIR                           ; 1089
                                 0C  11 0026E          BRB     25$                                  ;
                          34     AE  B5 00270 23$:     TSTW    TEMP_LINE                            ; 1092
                                 05  13 00273          BEQL    24$                                  ;
                    54           01  D0 00275          MOVL    #1, SELDIR                           ;
                                 02  11 00278          BRB     25$                                  ;
                                 54  D4 0027A 24$:     CLRL    SELDIR                               ;
                    59    34     AE  32 0027C 25$:     CVTWL   TEMP_LINE, REC_NO                    ; 1094
                    59           50  C2 00280          SUBL2   REC_OFFSET, REC_NO                   ;
                                 59  DD 00283          PUSHL   REC_NO                               ; 1095
         00000000G  00           01  FB 00285          CALLS   #1, EDT$$SC_MOVTOLN                  ;
                          28     AE  9F 0028C          PUSHAB  DISP                                 ; 1096
                    50 00000000G 00  9E 0028F          MOVAB   EDT$$T_LN_BUF, R0                    ;
      7E 00000000G  00           50  C3 00296          SUBL3   R0, EDT$$A_SEL_POS, -(SP)            ;
         00000000G  00           02  FB 0029E          CALLS   #2, EDT$$SC_FNDREC                   ;
                    57           50  D0 002A5          MOVL    R0, SCRPTR                           ;
                                 07  12 002A8          BNEQ    26$                                  ; 1098
               0C   AE           01  D0 002AA          MOVL    #1, BUILD_SCR                        ; 1100
                               009A  31 002AE          BRW     34$                                  ;
                                 54  D5 002B1 26$:     TSTL    SELDIR                               ; 1104
                                 48  12 002B3          BNEQ    29$                                  ;
                    50 00000000G 00  9E 002B5          MOVAB   EDT$$T_LN_BUF, R0                    ; 1108
                    50 00000000G 00  C2 002BC          SUBL2   EDT$$A_SEL_POS, R0                   ;
                    51           50  CE 002C3          MNEGL   R0, R1                               ;
                    52           51  D1 002C6          CMPL    R1, OUR_CHNO                         ;
                                 0F  18 002C9          BGEQ    27$                                  ;
                                 7E  D4 002CB          CLRL    -(SP)                                ; 1110
                    51    09     A3  9A 002CD          MOVZBL  9(OUR_SCRPTR), R1                    ; 1113
      51            52           51  C3 002D1          SUBL3   R1, OUR_CHNO, R1                     ;
                          FF     A1  9F 002D5          PUSHAB  -1(R1)                               ;
                                 53  11 002D8          BRB     31$                                  ; 1112
                    51           50  CE 002DA 27$:     MNEGL   R0, R1                               ; 1116
                    52           51  D1 002DD          CMPL    R1, OUR_CHNO                         ;
                                 69  15 002E0          BLEQ    34$                                  ;
                                 7E  D4 002E2          CLRL    -(SP)                                ; 1118
                    51    09     A7  9A 002E4 28$:     MOVZBL  9(SCRPTR), R1                        ; 1121
                          01 A140    9F 002E8          PUSHAB  1(R1)[R0]                            ;
                    6E           6E  CE 002EC          MNEGL   (SP), (SP)                           ;
                    50    09     A3  9A 002EF          MOVZBL  9(OUR_SCRPTR), R0                    ; 1120
      7E            52           50  C3 002F3          SUBL3   R0, OUR_CHNO, -(SP)                  ;
                        0088     8F  BB 002F7          PUSHR   #^M<R3,R7>                           ; 1118
                                 3E  11 002FB          BRB     32$                                  ;
                                 12  15 002FD 29$:     BLEQ    30$                                  ; 1126
                                 7E  D4 002FF          CLRL    -(SP)                                ; 1128
                    50 00000000G 00  9E 00301          MOVAB   EDT$$T_LN_BUF, R0                    ; 1131
                    50 00000000G 00  C2 00308          SUBL2   EDT$$A_SEL_POS, R0                   ;
                                 D3  11 0030F          BRB     28$                                  ;
                                 31  18 00311 30$:     BGEQ    33$                                  ; 1134
                                 7E  D4 00313          CLRL    -(SP)                                ; 1136
                    50    09     A3  9A 00315          MOVZBL  9(OUR_SCRPTR), R0                    ; 1139
                    52           50  C2 00319          SUBL2   R0, R2                               ;
                          FF     A2  9F 0031C          PUSHAB  -1(R2)                               ;
                    50 00000000G 00  9E 0031F          MOVAB   EDT$$T_LN_BUF, R0                    ; 1138
                    50 00000000G 00  C2 00326          SUBL2   EDT$$A_SEL_POS, R0                   ;
                    51    09     A7  9A 0032D 31$:     MOVZBL  9(SCRPTR), R1                        ;
                               6140  9F 00331          PUSHAB  (R1)[R0]                             ;
                    6E           6E  CE 00334          MNEGL   (SP), (SP)                           ;
```

```
                                 53  DD 00337        PUSHL   OUR_SCRPTR                           ; 1137
                                 57  DD 00339        PUSHL   SCRPTR                               ; 1136
          00000000G   00         05  FB 0033B  32$:  CALLS   #5, EDT$$SC_REPAINT                  ;
                                 07  11 00342        BRB     34$                                  ;
          00000000G   00         00  FB 00344  33$:  CALLS   #0, EDT$$INTER_ERR                   ; 1141
                      58         01  D0 0034B  34$:  MOVL    #1, ANY_CHANGES                      ; 1145
                      03    0C   AE  E9 0034E  35$:  BLBC    BUILD_SCR, 36$                       ; 1150
                               0127  31 00352        BRW     47$                                  ;
                      03    10   AE  E9 00355  36$:  BLBC    ERASE_ALL, 38$                       ;
                               00AB  31 00359  37$:  BRW     44$                                  ;
          00000000G   00 00000000G   00  D1 0035C  38$:  CMPL    EDT$$A_SEL_BUF, EDT$$A_CUR_BUF   ; 1154
                                 F0  12 00367        BNEQ    37$                                  ;
                      02 00000000G   00  D1 00369        CMPL    EDT$$G_TI_TYP, #2                ;
                                 E7  12 00370        BNEQ    37$                                  ;
                      53 00000000G   00  D0 00372        MOVL    EDT$$G_CS_OLDCHNO, R3            ; 1166
                      52 00000000G   00  D0 00379        MOVL    EDT$$G_CS_CHNO, R2               ; 1165
                            08   AE  D5 00380        TSTL    DIR                                  ; 1162
                                 2C  19 00383        BLSS    40$                                  ;
                                 25  15 00385        BLEQ    39$                                  ; 1169
                                 7E  D4 00387        CLRL    -(SP)                                ; 1171
                      51 00000000G   00  D0 00389        MOVL    EDT$$A_CUR_SCRPTR, R1            ; 1173
                      50    09   A1  9A 00390        MOVZBL  9(R1), R0                            ;
            50        52         50  C3 00394        SUBL3   R0, R2, R0                           ;
                            FF   A0  9F 00398        PUSHAB  -1(R0)                               ;
                      50 00000000G   00  D0 0039B        MOVL    EDT$$A_CSR_SCRPTR, R0            ; 1172
                      54    09   A0  9A 003A2        MOVZBL  9(R0), R4                            ;
            7E        53         54  C3 003A6        SUBL3   R4, R3, -(SP)                        ;
                                 4F  11 003AA        BRB     42$                                  ; 1171
                      53         52  D1 003AC  39$:  CMPL    R2, R3                               ; 1176
                                 25  18 003AF        BGEQ    41$                                  ;
                                 7E  D4 003B1  40$:  CLRL    -(SP)                                ; 1178
                      51 00000000G   00  D0 003B3        MOVL    EDT$$A_CSR_SCRPTR, R1            ; 1181
                      50    09   A1  9A 003BA        MOVZBL  9(R1), R0                            ;
            50        53         50  C3 003BE        SUBL3   R0, R3, R0                           ;
                            FF   A0  9F 003C2        PUSHAB  -1(R0)                               ;
                      50 00000000G   00  D0 003C5        MOVL    EDT$$A_CUR_SCRPTR, R0            ; 1180
                      54    09   A0  9A 003CC        MOVZBL  9(R0), R4                            ;
            7E        52         54  C3 003D0        SUBL3   R4, R2, -(SP)                        ;
                                 25  11 003D4        BRB     42$                                  ; 1179
                                 2C  15 003D6  41$:  BLEQ    43$                                  ; 1184
                                 7E  D4 003D8        CLRL    -(SP)                                ; 1186
                      51 00000000G   00  D0 003DA        MOVL    EDT$$A_CUR_SCRPTR, R1            ; 1189
                      50    09   A1  9A 003E1        MOVZBL  9(R1), R0                            ;
            50        52         50  C3 003E5        SUBL3   R0, R2, R0                           ;
                            FF   A0  9F 003E9        PUSHAB  -1(R0)                               ;
                      50 00000000G   00  D0 003EC        MOVL    EDT$$A_CSR_SCRPTR, R0            ; 1188
                      52    09   A0  9A 003F3        MOVZBL  9(R0), R2                            ;
            7E        53         52  C3 003F7        SUBL3   R2, R3, -(SP)                        ;
                                 03  BB 003FB  42$:  PUSHR   #^M<R0,R1>                           ; 1186
          00000000G   00         05  FB 003FD        CALLS   #5, EDT$$SC_REPAINT                  ;
                      58         01  D0 00404  43$:  MOVL    #1, ANY_CHANGES                      ; 1191
                      71    0C   AE  E8 00407  44$:  BLBS    BUILD_SCR, 47$                       ; 1200
                      20    10   AE  E9 0040B        BLBC    ERASE_ALL, 45$                       ; 1210
                                 01  DD 0040F        PUSHL   #1                                   ; 1213
                      7E    FF   8F  9A 00411        MOVZBL  #255, -(SP)                          ;
                                 7E  D4 00415        CLRL    -(SP)                                ;
                         00000000G   00  DD 00417        PUSHL   EDT$$A_LST_SCRPTR                ;
```

```
                          00000000G  00  DD 0041D          PUSHL   EDT$$A_FST_SCRPTR                        ;
              00000000G  00          05  FB 00423          CALLS   #5, EDT$$SC_REPAINT                      ;
                         58          01  D0 0042A          MOVL    #1, ANY_CHANGES                          ; 1214
                                     4D  11 0042D          BRB     47$                                      ; 1210
                         49      14  AE  E9 0042F 45$:     BLBC    SCROLLING_NEEDED, 47$                    ; 1219
                         50 00000000G  00  D0 00433          MOVL    EDT$$A_TOP_SCRPTR, R0                    ; 1222
                         57          60  D0 0043A          MOVL    (R0), SCRPTR                             ;
                                     17  13 0043D          BEQL    46$                                      ; 1224
                                     01  DD 0043F          PUSHL   #1                                       ;
                         7E      FF  8F  9A 00441          MOVZBL  #255, -(SP)                              ;
                                     7E  D4 00445          CLRL    -(SP)                                    ;
                                     57  DD 00447          PUSHL   SCRPTR                                   ;
                          00000000G  00  DD 00449          PUSHL   EDT$$A_FST_SCRPTR                        ;
              00000000G  00          05  FB 0044F          CALLS   #5, EDT$$SC_REPAINT                      ;
                         50 00000000G  00  D0 00456 46$:     MOVL    EDT$$A_BOT_SCRPTR, R0                    ; 1226
                                     1D  13 0045D          BEQL    47$                                      ;
                         57      04  A0  D0 0045F          MOVL    4(R0), SCRPTR                            ; 1229
                                     17  13 00463          BEQL    47$                                      ; 1231
                                     01  DD 00465          PUSHL   #1                                       ;
                         7E      FF  8F  9A 00467          MOVZBL  #255, -(SP)                              ;
                                     7E  D4 0046B          CLRL    -(SP)                                    ;
                          00000000G  00  DD 0046D          PUSHL   EDT$$A_LST_SCRPTR                        ;
                                     57  DD 00473          PUSHL   SCRPTR                                   ;
              00000000G  00          05  FB 00475          CALLS   #5, EDT$$SC_REPAINT                      ;
                          00000000G  00  D5 0047C 47$:     TSTL    EDT$$A_TOP_SCRPTR                        ; 1245
                                     04  12 00482          BNEQ    48$                                      ;
                    0C   AE          01  D0 00484          MOVL    #1, BUILD_SCR                            ;
                         03      0C  AE  E9 00488 48$:     BLBC    BUILD_SCR, 49$                           ; 1247
                                   00D5  31 0048C          BRW     66$                                      ;
                         57 00000000G  00  D0 0048F 49$:     MOVL    EDT$$A_CUR_SCRPTR, SCRPTR                ; 1256
                                     59  D4 00496          CLRL    REC_NO                                   ; 1257
           02  FFFFFFFF  8F      08  AE  CF 00498          CASEL   DIR, #-1, #2                             ; 1259
       000F          0037      0040     004A1 50$:     .WORD   55$-50$,-                                ;
                                                               54$-50$,-                                ;
                                                               51$-50$                                  ;
              00000000G  00          00  FB 004A7          CALLS   #0, EDT$$INTER_ERR                       ; 1340
                                     59  11 004AE          BRB     58$                                      ; 1259
                             08      A7  95 004B0 51$:     TSTB    8(SCRPTR)                                ; 1271
                                     05  13 004B3          BEQL    52$                                      ;
           0C          0D    A7      02  E1 004B5          BBC     #2, 13(SCRPTR), 53$                      ; 1272
                         50          67  D0 004BA 52$:     MOVL    (SCRPTR), PREV_SCRPTR                    ; 1279
                                     07  13 004BD          BEQL    53$                                      ; 1281
           02          0D    A0      02  E0 004BF          BBS     #2, 13(PREV_SCRPTR), 53$                 ; 1285
                                     59  D7 004C4          DECL    REC_NO                                   ; 1287
                         57          67  D0 004C6 53$:     MOVL    (SCRPTR), SCRPTR                         ; 1293
              00000000G  00          57  D1 004C9          CMPL    SCRPTR, EDT$$A_CSR_SCRPTR                ; 1295
                                     37  13 004D0          BEQL    58$                                      ;
                                     57  D5 004D2          TSTL    SCRPTR                                   ;
                                     DA  12 004D4          BNEQ    51$                                      ;
                                     31  11 004D6          BRB     58$                                      ; 1259
                         57 00000000G  00  D0 004D8 54$:     MOVL    EDT$$A_CSR_SCRPTR, SCRPTR                ; 1304
                                     28  11 004DF          BRB     58$                                      ; 1259
                         50      04  A7  D0 004E1 55$:     MOVL    4(SCRPTR), NEXT_SCRPTR                   ; 1319
                                     11  13 004E5          BEQL    57$                                      ; 1321
           0C          0D    A7      02  E0 004E7          BBS     #2, 13(SCRPTR), 57$                      ; 1325
                             08      A0  95 004EC          TSTB    8(NEXT_SCRPTR)                           ; 1326
                                     05  13 004EF          BEQL    56$                                      ;
```

```
        02        0D   A0           02  E1 004F1        BBC     #2, 13(NEXT_SCRPTR), 57$                ; 1327
                                    59  D6 004F6 56$:   INCL    REC_NO                                  ; 1329
                  57          04    A7  D0 004F8 57$:   MOVL    4(SCRPTR), SCRPTR                       ; 1333
         00000000G 00               57  D1 004FC        CMPL    SCRPTR, EDT$$A_CSR_SCRPTR               ; 1335
                                    04  13 00503        BEQL    58$                                     ;
                                    57  D5 00505        TSTL    SCRPTR                                  ;
                                    D8  12 00507        BNEQ    55$                                     ;
         00000000G 00               57  D1 00509 58$:   CMPL    SCRPTR, EDT$$A_CSR_SCRPTR               ; 1347
                                    04  12 00510        BNEQ    59$                                     ;
                                    57  D5 00512        TSTL    SCRPTR                                  ;
                                    04  12 00514        BNEQ    60$                                     ;
                  0C   AE           01  D0 00516 59$:   MOVL    #1, BUILD_SCR                           ;
                  46          0C    AE  E8 0051A 60$:   BLBS    BUILD_SCR, 66$                          ; 1356
                  0A          14    AE  E8 0051E        BLBS    SCROLLING_NEEDED, 61$                   ; 1360
              24  AE 00000000G      00  D0 00522        MOVL    EDT$$G_TOP_RECNO, OLD_TOP_RECNO         ; 1362
                                    38  11 0052A        BRB     66$                                     ;
                  51 00000000G      00  D0 0052C 61$:   MOVL    EDT$$A_TOP_SCRPTR, R1                   ; 1366
                  51                57  D1 00533 62$:   CMPL    SCRPTR, R1                              ;
                                    1F  13 00536        BEQL    65$                                     ;
                                    57  D5 00538        TSTL    SCRPTR                                  ;
                                    1B  13 0053A        BEQL    65$                                     ;
                              08    A7  95 0053C        TSTB    8(SCRPTR)                               ; 1369
                                    05  13 0053F        BEQL    63$                                     ;
        0C        0D   A7           02  E1 00541        BBC     #2, 13(SCRPTR), 64$                     ; 1370
                  50                67  D0 00546 63$:   MOVL    (SCRPTR), PREV_SCRPTR                   ; 1377
                                    07  13 00549        BEQL    64$                                     ; 1379
        02        0D   A0           02  E0 0054B        BBS     #2, 13(PREV_SCRPTR), 64$                ; 1383
                                    59  D7 00550        DECL    REC_NO                                  ; 1385
                  57                67  D0 00552 64$:   MOVL    (SCRPTR), SCRPTR                        ; 1391
                                    DC  11 00555        BRB     62$                                     ; 1366
              24  AE                59  D0 00557 65$:   MOVL    REC_NO, OLD_TOP_RECNO                   ; 1394
                  51                57  D1 0055B        CMPL    SCRPTR, R1                              ; 1399
                                    04  13 0055E        BEQL    66$                                     ;
                  0C   AE           01  D0 00560        MOVL    #1, BUILD_SCR                           ;
                  07                58  E8 00564 66$:   BLBS    ANY_CHANGES, 68$                        ; 1405
                  03          14    AE  E8 00567        BLBS    SCROLLING_NEEDED, 68$                   ;
                                  00B0  31 0056B 67$:   BRW     82$                                     ;
                  F9          0C    AE  E8 0056E 68$:   BLBS    BUILD_SCR, 67$                          ;
                     00000000G      00  D5 00572        TSTL    EDT$$G_RECS_INSERTED                    ; 1427
                                    F1  13 00578        BEQL    67$                                     ;
                                    56  D4 0057A 69$:   CLRL    ANOTHER_PASS                            ; 1433
                                    52  D4 0057C        CLRL    INS_COUNT                               ; 1434
                  57 00000000G      00  D0 0057E        MOVL    EDT$$A_TOP_SCRPTR, SCRPTR               ; 1435
                                    53  D4 00585 70$:   CLRL    UPDATE_DONE                             ; 1439
                  54                52  D0 00587        MOVL    INS_COUNT, PREV_INS_COUNT               ; 1440
        02        0D   A7           01  E1 0058A        BBC     #1, 13(SCRPTR), 71$                     ; 1442
                                    52  D6 0058F        INCL    INS_COUNT                               ;
        02        0D   A7           02  E1 00591 71$:   BBC     #2, 13(SCRPTR), 72$                     ; 1444
                                    52  D7 00596        DECL    INS_COUNT                               ;
                                    07  13 00598 72$:   BEQL    73$                                     ; 1446
                                    54  D5 0059A        TSTL    PREV_INS_COUNT                          ;
                                    03  12 0059C        BNEQ    73$                                     ;
                  55                57  D0 0059E        MOVL    SCRPTR, BEG_SCRPTR                      ;
                                    52  D5 005A1 73$:   TSTL    INS_COUNT                               ; 1448
                                    3E  12 005A3        BNEQ    76$                                     ;
                                    54  D5 005A5        TSTL    PREV_INS_COUNT                          ;
                                    3A  13 005A7        BEQL    76$                                     ;
```

```
                          00000000G  00              55  D1 005A9          CMPL    BEG_SCRPTR, EDT$$A_TOP_SCRPTR          ; 1455
                                                    16  12 005B0          BNEQ    75$                                    ;
                                     50 00000000G   00  D0 005B2 74$:      MOVL    EDT$$A_TOP_SCRPTR, R0                  ; 1458
                                 0A        0D   A0   02  E1 005B9          BBC     #2, 13(R0), 75$                        ;
                              00000000G   00        04  A0 D0 005BE          MOVL    4(R0), EDT$$A_TOP_SCRPTR               ; 1459
                                                    EA  11 005C6          BRB     74$                                    ;
                                                    01  DD 005C8 75$:      PUSHL   #1                                     ; 1464
                                     7E         FF  8F  9A 005CA          MOVZBL  #255, -(SP)                            ;
                                                    7E  D4 005CE          CLRL    -(SP)                                  ;
                                              00A0  8F  BB 005D0          PUSHR   #^M<R5,R7>                             ;
                          00000000G  00             05  FB 005D4          CALLS   #5, EDT$$SC_REPAINT                    ;
                                     53             01  D0 005DB          MOVL    #1, UPDATE_DONE                        ; 1465
                                     56             01  D0 005DE          MOVL    #1, ANOTHER_PASS                       ; 1466
                                                    17  11 005E1          BRB     79$                                    ; 1448
                          00000000G  00             57  D1 005E3 76$:      CMPL    SCRPTR, EDT$$A_BOT_SCRPTR              ; 1471
                                                    05  13 005EA          BEQL    77$                                    ;
                                               04   A7  D5 005EC          TSTL    4(SCRPTR)                              ;
                                                    05  12 005EF          BNEQ    78$                                    ;
                                     53             01  D0 005F1 77$:      MOVL    #1, UPDATE_DONE                        ; 1473
                                                    04  11 005F4          BRB     79$                                    ;
                                     57         04  A7  D0 005F6 78$:      MOVL    4(SCRPTR), SCRPTR                      ; 1475
                                     88             53  E9 005FA 79$:      BLBC    UPDATE_DONE, 70$                       ; 1480
                                     03             56  E9 005FD          BLBC    ANOTHER_PASS, 80$                      ; 1483
                                                  FF77  31 00600          BRW     69$                                    ;
                                                    52  D5 00603 80$:      TSTL    R2                                     ; 1490
                                                    03  18 00605          BGEQ    81$                                    ;
                                     52             52  CE 00607          MNEGL   R2, R2                                 ;
                                50 00000000G        00  01  78 0060A 81$:      ASHL    #1, EDT$$G_SCR_LNS, R0                 ;
                                     50             03  C6 00612          DIVL2   #3, R0                                 ;
                                     50             52  D1 00615          CMPL    R2, R0                                 ;
                                                    04  15 00618          BLEQ    82$                                    ;
                                   0C   AE          01  D0 0061A          MOVL    #1, BUILD_SCR                          ;
                                     07             58  E8 0061E 82$:      BLBS    ANY_CHANGES, 83$                       ; 1496
                                     03         14  AE  E8 00621          BLBS    SCROLLING_NEEDED, 83$                  ;
                                                01D1  31 00625          BRW     115$                                   ;
                                     03         0C  AE  E9 00628 83$:      BLBC    BUILD_SCR, 84$                         ;
                                                01CE  31 0062C          BRW     116$                                   ;
                                                    50  D4 0062F 84$:      CLRL    R0                                     ; 1513
                                        00000000G   00  D5 00631          TSTL    EDT$$G_RECS_INSERTED                   ;
                                                    02  12 00637          BNEQ    85$                                    ;
                                                    50  D6 00639          INCL    R0                                     ;
                                     5A             50  D0 0063B 85$:      MOVL    R0, INS_DEL_DONE                       ;
                                                    54  7C 0063E 86$:      CLRQ    ANY_INS_DEL                            ; 1518
                                     59         24  AE  D0 00640          MOVL    OLD_TOP_RECNO, REC_NO                  ; 1519
                                     57 00000000G   00  D0 00644          MOVL    EDT$$A_TOP_SCRPTR, SCRPTR              ; 1520
                                        00000000G   00  D4 0064B          CLRL    EDT$$G_CS_CNO                          ; 1521
                                                    53  D4 00651 87$:      CLRL    UPDATE_DONE                            ; 1525
                                     07         0D  A7  93 00653          BITB    13(SCRPTR), #7                         ; 1528
                                                    03  12 00657          BNEQ    88$                                    ;
                                                00B0  31 00659          BRW     99$                                    ;
                                26        0D   A7  02  E1 0065C 88$:      BBC     #2, 13(SCRPTR), 90$                    ; 1535
                                     07             5A  E9 00661          BLBC    INS_DEL_DONE, 89$                      ; 1538
                             00000000G   00        00  FB 00664          CALLS   #0, EDT$$INTER_ERR                     ;
                                               24  AE  9F 0066B 89$:      PUSHAB  OLD_TOP_RECNO                          ; 1539
                                                    57  DD 0066E          PUSHL   SCRPTR                                 ;
                                   0000V   CF       02  FB 00670          CALLS   #2, DELETE_LINE                        ;
                                     56             50  D0 00675          MOVL    R0, STATUS                             ;
```

```
                   54          01 D0 00678          MOVL    #1, ANY_INS_DEL                 ; 1540
                   2A          56 E9 0067B          BLBC    STATUS, 92$                     ; 1541
         00000000G 00          00 FB 0067E          CALLS   #0, EDT$$INTER_ERR              ;
                               7C 11 00685          BRB     97$                             ; 1535
      52      0D   A7          01 E1 00687 90$:     BBC     #1, 13(SCRPTR), 94$             ; 1545
                   07          5A E9 0068C          BLBC    INS_DEL_DONE, 91$               ; 1548
         00000000G 00          00 FB 0068F          CALLS   #0, EDT$$INTER_ERR              ;
                          24   AE 9F 00696 91$:     PUSHAB  OLD_TOP_RECNO                   ; 1549
                        0280   8F BB 00699          PUSHR   #^M<R7,R9>                      ;
             0000V CF          03 FB 0069D          CALLS   #3, INSERT_LINE                 ;
                   56          50 D0 006A2          MOVL    R0, STATUS                      ;
                   54          01 D0 006A5          MOVL    #1, ANY_INS_DEL                 ; 1550
                   5B          56 E9 006A8 92$:     BLBC    STATUS, 98$                     ; 1556
                               59 DD 006AB          PUSHL   REC_NO                          ; 1559
         00000000G 00          01 FB 006AD          CALLS   #1, EDT$$SC_MOVTOLN             ;
                   07          50 E8 006B4          BLBS    R0, 93$                         ;
         00000000G 00          00 FB 006B7          CALLS   #0, EDT$$INTER_ERR              ;
                               01 DD 006BE 93$:     PUSHL   #1                              ; 1560
                   7E     FF   8F 9A 006C0          MOVZBL  #255, -(SP)                     ;
                               7E D4 006C4          CLRL    -(SP)                           ;
                               57 DD 006C6          PUSHL   SCRPTR                          ;
                               57 DD 006C8          PUSHL   SCRPTR                          ;
         00000000G 00          05 FB 006CA          CALLS   #5, EDT$$SC_REPAINT             ;
                               01 DD 006D1          PUSHL   #1                              ; 1561
                               57 DD 006D3          PUSHL   SCRPTR                          ;
         00000000G 00          02 FB 006D5          CALLS   #2, EDT$$SC_RFRELN              ;
                               25 11 006DC          BRB     97$                             ; 1545
                   1F          5A E9 006DE 94$:     BLBC    INS_DEL_DONE, 96$               ; 1572
                               59 DD 006E1          PUSHL   REC_NO                          ; 1575
         00000000G 00          01 FB 006E3          CALLS   #1, EDT$$SC_MOVTOLN             ;
                   07          50 E8 006EA          BLBS    R0, 95$                         ;
         00000000G 00          00 FB 006ED          CALLS   #0, EDT$$INTER_ERR              ;
                          10   AE DD 006F4 95$:     PUSHL   ERASE_ALL                       ; 1576
                               57 DD 006F7          PUSHL   SCRPTR                          ;
         00000000G 00          02 FB 006F9          CALLS   #2, EDT$$SC_RFRELN              ;
                   56          01 D0 00700 96$:     MOVL    #1, STATUS                      ; 1579
                   06          56 E8 00703 97$:     BLBS    STATUS, 99$                     ; 1582
                   53          01 D0 00706 98$:     MOVL    #1, UPDATE_DONE                 ; 1585
                   55          01 D0 00709          MOVL    #1, ANOTHER_PASS                ; 1586
                   03          53 E9 0070C 99$:     BLBC    UPDATE_DONE, 100$               ; 1591
                             00AD 31 0070F          BRW     111$                            ;
                   00000000G   00 D6 00712 100$:    INCL    EDT$$G_CS_LNO                   ; 1594
                50 00000000G   00 D0 00718          MOVL    EDT$$G_CS_LNO, R0               ; 1596
         00000000G 00          50 D1 0071F          CMPL    R0, EDT$$G_SCR_LNS              ;
                               51 13 00726          BEQL    103$                            ;
                   52     04   A7 D0 00728          MOVL    4(SCRPTR), R2                   ; 1602
                               72 12 0072C          BNEQ    107$                            ;
         00000000G 00          57 D1 0072E          CMPL    SCRPTR, EDT$$A_EOB_SCRPTR       ; 1610
                               12 12 00735          BNEQ    101$                            ;
         00000000G 00          50 D1 00737          CMPL    R0, EDT$$G_BOT_LINE             ; 1619
                               39 18 0073E          BGEQ    103$                            ;
         00000000G 00          00 FB 00740          CALLS   #0, EDT$$SC_ERAALL              ;
                               30 11 00747          BRB     103$                            ; 1621
                               59 D6 00749 101$:    INCL    REC_NO                          ; 1629
                               59 DD 0074B          PUSHL   REC_NO                          ; 1630
         00000000G 00          01 FB 0074D          CALLS   #1, EDT$$SC_MOVTOLN             ;
                   07          50 E8 00754          BLBS    R0, 102$                        ;
```

```
        00000000G  00                 00  FB 00757        CALLS   #0, EDT$$INTER_ERR                     ;
                   50 00000000G       00  D0 0075E 102$:  MOVL    EDT$$A_WK_LN, R0                       ; 1632
                   7E                 60  9A 00765        MOVZBL  (R0), -(SP)                            ;
                            07        A0  9F 00768        PUSHAB  7(R0)                                  ; 1631
                                      7E  D4 0076B        CLRL    -(SP)                                  ;
        00000000G  00                 03  FB 0076D        CALLS   #3, EDT$$SC_LNINS                      ;
                   58                 50  D0 00774        MOVL    R0, LNINS_VAL                          ;
                                      05  12 00777        BNEQ    104$                                   ; 1634
                   53                 01  D0 00779 103$:  MOVL    #1, UPDATE_DONE                        ; 1636
                                      3B  11 0077C        BRB     110$                                   ;
                            04        A7  D5 0077E 104$:  TSTL    4(SCRPTR)                              ; 1640
                                      07  12 00781        BNEQ    105$                                   ;
        00000000G  00                 00  FB 00783        CALLS   #0, EDT$$INTER_ERR                     ;
                   57       04        A7  D0 0078A 105$:  MOVL    4(SCRPTR), SCRPTR                      ; 1641
                            08        A7  95 0078E        TSTB    8(SCRPTR)                              ; 1642
                                      07  13 00791        BEQL    106$                                   ;
        00000000G  00                 00  FB 00793        CALLS   #0, EDT$$INTER_ERR                     ;
               0D  A7                 01  90 0079A 106$:  MOVB    #1, 13(SCRPTR)                         ; 1643
                                      19  11 0079E        BRB     110$                                   ; 1602
                   50                 52  D0 007A0 107$:  MOVL    R2, NEXT_SCRPTR                        ; 1655
                                      11  13 007A3        BEQL    109$                                   ; 1657
     0C        0D  A7                 02  E0 007A5        BBS     #2, 13(SCRPTR), 109$                   ; 1661
                            08        A0  95 007AA        TSTB    8(NEXT_SCRPTR)                         ; 1662
                                      05  13 007AD        BEQL    108$                                   ;
     02        0D  A0                 02  E1 007AF        BBC     #2, 13(NEXT_SCRPTR), 109$              ; 1663
                                      59  D6 007B4 108$:  INCL    REC_NO                                 ; 1665
                   57                 52  D0 007B6 109$:  MOVL    R2, SCRPTR                             ; 1669
                   03                 53  E8 007B9 110$:  BLBS    UPDATE_DONE, 111$                      ; 1677
                                    FE92  31 007BC        BRW     87$                                    ;
                   03                 5A  E8 007BF 111$:  BLBS    INS_DEL_DONE, 112$                     ; 1683
                   55                 01  D0 007C2        MOVL    #1, ANOTHER_PASS                       ;
                   12                 54  E8 007C5 112$:  BLBS    ANY_INS_DEL, 113$                      ; 1689
                   5A                 01  D0 007C8        MOVL    #1, INS_DEL_DONE                       ; 1692
                      00000000G       00  DD 007CB        PUSHL   EDT$$G_SCR_LNS                         ; 1696
                                      7E  D4 007D1        CLRL    -(SP)                                  ;
        00000000G  00                 02  FB 007D3        CALLS   #2, EDT$$SC_SETSCLLREG                 ;
                   03                 55  E9 007DA 113$:  BLBC    ANOTHER_PASS, 114$                     ; 1700
                                    FE5E  31 007DD        BRW     86$                                    ;
        00000000G  00 00000000G       00  D0 007E0 114$:  MOVL    EDT$$G_CS_LNO, EDT$$G_BOT_LINE         ; 1702
               18  AE                 59  D0 007EB        MOVL    REC_NO, OLD_BOT_RECNO                  ; 1703
        00000000G  00                 57  D0 007EF        MOVL    SCRPTR, EDT$$A_BOT_SCRPTR              ; 1704
                            10        AE  D4 007F6        CLRL    ERASE_ALL                              ; 1710
                   49       0C        AE  E9 007F9 115$:  BLBC    BUILD_SCR, 118$                        ; 1713
                   51 00000000G       00  D0 007FD 116$:  MOVL    EDT$$A_FST_SCRPTR, R1                  ; 1721
                                      16  13 00804        BEQL    117$                                   ;
                   50 00000000G       00  D0 00806        MOVL    EDT$$A_LST_SCRPTR, R0                  ; 1724
               04  A0 00000000G       00  D0 0080D        MOVL    EDT$$A_FST_AVLN, 4(R0)                 ;
        00000000G  00                 51  D0 00815        MOVL    R1, EDT$$A_FST_AVLN                    ; 1725
                      00000000G       00  D4 0081C 117$:  CLRL    EDT$$A_FST_SCRPTR                      ; 1728
                      00000000G       00  D4 00822        CLRL    EDT$$A_LST_SCRPTR                      ; 1729
                      00000000G       00  D4 00828        CLRL    EDT$$A_TOP_SCRPTR                      ; 1730
                      00000000G       00  D4 0082E        CLRL    EDT$$A_CUR_SCRPTR                      ; 1731
                      00000000G       00  D4 00834        CLRL    EDT$$A_BOT_SCRPTR                      ; 1732
                      00000000G       00  D4 0083A        CLRL    EDT$$A_EOB_SCRPTR                      ; 1733
                      00000000G       00  D4 00840        CLRL    EDT$$G_MEM_CNT                         ; 1734
                                      59  D4 00846 118$:  CLRL    REC_NO                                 ; 1740
                                      59  DD 00848        PUSHL   REC_NO                                 ; 1741
```

```
                    00000000G  00                  01  FB 0084A          CALLS   #1, EDT$$SC_MOVTOLN                       ;
                               07                  50  E8 00851          BLBS    R0, 119$                                  ;
                    00000000G  00                  00  FB 00854          CALLS   #0, EDT$$INTER_ERR                        ;
                               50 00000000G        00  D0 0085B 119$:    MOVL    EDT$$A_CUR_BUF, R0                        ; 1742
                    00000000G  00                  50  D0 00862          MOVL    R0, EDT$$A_SCR_BUF                        ;
     00000000G  00         06                      A0  06 28 00869       MOVC3   #6, 6(R0), EDT$$L_CUR_SCRLN               ; 1743
                               4D          0C      AE  E9 00872          BLBC    BUILD_SCR, 121$                           ; 1745
                               50 00000000G        00  D0 00876          MOVL    EDT$$A_WK_LN, R0                          ; 1748
                               7E                  60  9A 0087D          MOVZBL  (R0), -(SP)                               ;
                                           07      A0  9F 00880          PUSHAB  7(R0)                                     ;
                                                   7E  D4 00883          CLRL    -(SP)                                     ;
                    00000000G  00                  03  FB 00885          CALLS   #3, EDT$$SC_LNINS                         ;
                    00000000G  00 00000000G        00  D0 0088C          MOVL    EDT$$A_FST_SCRPTR, EDT$$A_CSR_SCRPTR      ; 1749
                                  00000000G        00  D5 00897          TSTL    EDT$$G_TRUN                               ; 1751
                                                   19  12 0089D          BNEQ    120$                                      ;
                                           28      AE  9F 0089F          PUSHAB  DISP                                      ; 1753
                                  00000000G        00  DD 008A2          PUSHL   EDT$$G_CS_CHNO                            ;
                    00000000G  00                  02  FB 008A8          CALLS   #2, EDT$$SC_FNDREC                        ;
                    00000000G  00                  50  D0 008AF          MOVL    R0, EDT$$A_CUR_SCRPTR                     ;
                                                   0B  11 008B6          BRB     121$                                      ;
                    00000000G  00 00000000G        00  D0 008B8 120$:    MOVL    EDT$$A_CSR_SCRPTR, EDT$$A_CUR_SCRPTR      ; 1755
                               07          14      AE  E8 008C3 121$:    BLBS    SCROLLING_NEEDED, 122$                    ; 1765
                               03          0C      AE  E8 008C7          BLBS    BUILD_SCR, 122$                           ;
                                              0505  31 008CB             BRW     194$                                      ;
                               57 00000000G        00  D0 008CE 122$:    MOVL    EDT$$A_CUR_SCRPTR, SCRPTR                 ; 1772
                                                   59  D4 008D5          CLRL    REC_NO                                    ; 1773
                                                   56  D4 008D7          CLRL    BELOW                                     ; 1774
                                                   52  7C 008D9          CLRQ    AT_BOTTOM                                 ; 1781
                 50 00000000G  00                  01  78 008DB 123$:    ASHL    #1, EDT$$G_SCR_LNS, R0                    ; 1783
                               50                  56  D1 008E3          CMPL    BELOW, R0                                 ;
                                                   76  18 008E6          BGEQ    130$                                      ;
                               73                  52  E8 008E8          BLBS    AT_BOTTOM, 130$                           ;
                                                   57  D5 008EB          TSTL    SCRPTR                                    ; 1786
                                                   50  12 008ED          BNEQ    125$                                      ;
                                                   59  DD 008EF          PUSHL   REC_NO                                    ; 1789
                    00000000G  00                  01  FB 008F1          CALLS   #1, EDT$$SC_MOVTOLN                       ;
                               07                  50  E8 008F8          BLBS    R0, 124$                                  ;
                    00000000G  00                  00  FB 008FB          CALLS   #0, EDT$$INTER_ERR                        ;
                               50 00000000G        00  D0 00902 124$:    MOVL    EDT$$A_WK_LN, R0                          ; 1790
                               7E                  60  9A 00909          MOVZBL  (R0), -(SP)                               ;
                                           07      A0  9F 0090C          PUSHAB  7(R0)                                     ;
                                                   7E  D4 0090F          CLRL    -(SP)                                     ;
                    00000000G  00                  03  FB 00911          CALLS   #3, EDT$$SC_LNINS                         ;
                               58                  50  [] 00918          MOVL    R0, LNINS_VAL                             ;
                                                   2D  13 0091B          BEQL    126$                                      ; 1792
                               56                  58  C0 0091D          ADDL2   LNINS_VAL, BELOW                          ; 1797
                                                   59  D6 00920          INCL    REC_NO                                    ; 1798
                               50 00000000G        00  9E 00922          MOVAB   EDT$$Z_EOB_LN, R0                         ; 1800
                               50 00000000G        00  D1 00929          CMPL    EDT$$A_WK_LN, R0                          ;
                                                   A9  12 00930          BNEQ    123$                                      ;
                    00000000G  00 00000000G        00  D0 00932          MOVL    EDT$$A_LST_SCRPTR, EDT$$A_EOB_SCRPTR      ; 1803
                                                   0B  11 0093D          BRB     126$                                      ; 1804
                                                   56  D6 C093F 125$:    INCL    BELOW                                     ; 1812
                    00000000G  00                  57  D1 00941          CMPL    SCRPTR, EDT$$A_EOB_SCRPTR                 ; 1814
                                                   05  12 00948          BNEQ    128$                                      ;
                               52                  01  D0 0094A 126$:    MOVL    #1, AT_BOTTOM                             ; 1816
                                                   8C  11 0094D 127$:    BRB     123$                                      ;
```

```
                                57        04    A7  D0 0094F 128$:   MOVL    4(SCRPTR), SCRPTR                    ; 1819
                                                05  13 00953         BEQL    129$                                 ; 1821
                                          08    A7  95 00955         TSTB    8(SCRPTR)                            ; 1826
                                                81  12 00958         BNEQ    123$                                 ;
                                                59  D6 0095A 129$:   INCL    REC_NO                               ;
                                                EF  11 0095C         BRB     127$                                 ; 1783
                                57 000000)0G    00  D0 0095E 130$:   MOVL    EDT$$A_CUR_SCRPTR, SCRPTR            ; 1837
                                                59  D4 00965         CLRL    REC_NO                               ; 1838
                                                52  D4 00967         CLRL    AT_TOP                               ; 1844
                          50 00000000G          00  01  78 00969 131$:   ASHL    #1, EDT$$G_SCR_LNS, R0               ; 1846
                                                50  53  D1 00971         CMPL    ABOVE, R0                            ;
                                                    53  18 00974         BGEQ    136$                                 ;
                                                50  52  E8 00976         BLBS    AT_TOP, 136$                         ;
                                                    57  D5 00979         TSTL    SCRPTR                               ; 1849
                                                    0C  13 0097B         BEQL    133$                                 ;
                                          08    A7  95 0097D         TSTB    8(SCRPTR)                            ; 1853
                                                02  12 00980         BNEQ    132$                                 ;
                                                59  D7 00982         DECL    REC_NO                               ;
                                57              67  D0 00984 132$:   MOVL    (SCRPTR), SCRPTR                     ; 1855
                                                3C  12 00987         BNEQ    135$                                 ; 1862
                                                59  DD 00989 133$:   PUSHL   REC_NO                               ; 1866
                          00000000G             00  01  FB 0098B         CALLS   #1, EDT$$SC_MOVTOLN                  ;
                                                2B  50  E9 00992         BLBC    R0, 134$                             ;
                                50 00000000G    00  D0 00995         MOVL    EDT$$A_WK_LN, R0                     ; 1870
                                7E              60  9A 0099C         MOVZBL  (R0), -(SP)                          ;
                                          07    A0  9F 0099F         PUSHAB  7(R0)                                ; 1869
                                00000000G       00  DD 009A2         PUSHL   EDT$$A_FST_SCRPTR                    ;
                          00000000G             00  03  FB 009A8         CALLS   #3, EDT$$SC_LNINS                    ;
                                58              50  D0 009AF         MOVL    R0, LNINS_VAL                        ;
                                                0C  13 009B2         BEQL    134$                                 ; 1872
                                53              58  C0 009B4         ADDL2   LNINS_VAL, ABOVE                     ; 1877
                                57 00000000G    00  D0 009B7         MOVL    EDT$$A_FST_SCRPTR, SCRPTR            ; 1878
                                                A9  11 009BE         BRB     131$                                 ; 1866
                                52              01  D0 009C0 134$:   MOVL    #1, AT_TOP                           ; 1883
                                                A4  11 009C3         BRB     131$                                 ; 1862
                                                53  D6 009C5 135$:   INCL    ABOVE                                ; 1887
                                                A0  11 009C7         BRB     131$                                 ; 1846
                                5B 00000000G    00  D0 009C9 136$:   MOVL    EDT$$A_CUR_SCRPTR, R11               ; 1892
                                57              5B  D0 009D0         MOVL    R11, SCRPTR                          ;
                                                5A  D4 009D3         CLRL    TOP_SET                              ; 1899
                                6E 00000000G    00  D0 009D5         MOVL    LOW_1, (SP)                          ; 1901
                    00000000G                   00  6E  D1 009DC         CMPL    (SP), LOW_2                          ;
                                                    10  12 009E3         BNEQ    137$                                 ;
                    00000000G                   00  00000000G   00  B1 009E5         CMPW    HIGH_1, HIGH_2                       ;
                                                    03  12 009F0         BNEQ    137$                                 ;
                                                00A4    31 009F2         BRW     147$                                 ;
                          04    AE 00000000G    00  3C 009F5 137$:   MOVZWL  HIGH_1, 4(SP)                       ; 1908
                                50 00000000G    00  3C 009FD         MOVZWL  HIGH_2, R0                           ;
                                50        04    AE  D1 00A04         CMPL    4(SP), R0                            ;
                                                    0E  1F 00A08         BLSSU   138$                                 ;
                                                    17  12 00A0A         BNEQ    140$                                 ;
                                50 00000000G    00  D0 00A0C         MOVL    LOW_2, R0                            ;
                                50              6E  D1 00A13         CMPL    (SP), R0                             ;
                                                    05  1E 00A16         BGEQU   139$                                 ;
                                50              01  CE 00A18 138$:   MNEGL   #1, R0                               ;
                                                    09  11 00A1B         BRB     141$                                 ;
                                                    04  12 00A1D 139$:   BNEQ    140$                                 ;
```

```
                                    50  D4 00A1F        CLRL    R0                                       ;
                                    03  11 00A21        BRB     141$                                     ;
                           50       01  D0 00A23 140$:  MOVL    #1, R0                                   ;
                                    65  14 00A26 141$:  BGTR    146$                                     ;
                                    58  D4 00A28        CLRL    TOP_DIST                                 ; 1916
                           57       5B  D0 00A2A        MOVL    R11, SCRPTR                              ; 1917
           34  AE 00000000G 00      06  28 00A2D        MOVC3   #6, EDT$$L_CS_LN, TEMP_LINE              ; 1918
                           6E   34  AE  D1 00A36 142$:  CMPL    LOW_1, (SP)                              ; 1920
                                    07  12 00A3A        BNEQ    143$                                     ;
                   04      AE   38  AE  B1 00A3C        CMPW    HIGH_1, 4(SP)                            ;
                                    2B  13 00A41        BEQL    145$                                     ;
              00000000G    00       58  D1 00A43 143$:  CMPL    TOP_DIST, EDT$$G_SCLL_BOT                ; 1921
                                    22  18 00A4A        BGEQ    145$                                     ;
                                    57  D5 00A4C        TSTL    SCRPTR                                   ; 1922
                                    1E  13 00A4E        BEQL    145$                                     ;
                                08  A7  95 00A50        TSTB    8(SCRPTR)                                ; 1925
                                    10  12 00A53        BNEQ    144$                                     ;
                           50   34  AE  D0 00A55        MOVL    LOW_1, SAVE                              ;
                                34  AE  D7 00A59        DECL    FIRST_WORD                               ;
                           50   34  AE  D1 00A5C        CMPL    FIRST_WORD, SAVE                         ;
                                    03  1B 00A60        BLEQU   144$                                     ;
                                38  AE  B7 00A62        DECW    NEXT_WORD                                ;
                           57       67  D0 00A65 144$:  MOVL    (SCRPTR), SCRPTR                         ; 1927
                                    CC  13 00A68        BEQL    142$                                     ; 1929
                                    58  D6 00A6A        INCL    TOP_DIST                                 ;
                                    C8  11 00A6C        BRB     142$                                     ; 1920
                           6E   34  AE  D1 00A6E 145$:  CMPL    LOW_1, (SP)                              ; 1938
                                    19  12 00A72        BNEQ    146$                                     ;
                   04      AE   38  AE  B1 00A74        CMPW    HIGH_1, 4(SP)                            ;
                                    12  12 00A79        BNEQ    146$                                     ;
           50              56       58  C1 00A7B        ADDL3   TOP_DIST, BELOW, R0                      ;
              00000000G    00       50  D1 00A7F        CMPL    R0, EDT$$G_SCR_LNS                       ;
                                    05  19 00A86        BLSS    146$                                     ;
                           5A       01  D0 00A88        MOVL    #1, TOP_SET                              ; 1940
                                    0C  11 00A8B        BRB     147$                                     ;
 00000000G  00 00000000G   00       06  28 00A8D 146$:  MOVC3   #6, EDT$$L_LNO_EMPTY, EDT$$L_TOP_LN      ; 1942
                           5D       5A  E8 00A99 147$:  BLBS    TOP_SET, 151$                            ; 1952
                      00000000G     00  D5 00A9C        TSTL    EDT$$A_TOP_SCRPTR                        ;
                                    55  13 00AA2        BEQL    151$                                     ;
                           57       5B  D0 00AA4        MOVL    R11, SCRPTR                              ; 1955
                                    58  7C 00AA7        CLRQ    TOP_DIST                                 ; 1957
                     50 00000000G   00  D0 00AA9        MOVL    EDT$$G_SCLL_BOT, R0                      ; 1959
                           50       58  D1 00AB0 148$:  CMPL    TOP_DIST, R0                             ;
                                    1D  14 00AB3        BGTR    150$                                     ;
              00000000G    00       57  D1 00AB5        CMPL    SCRPTR, EDT$$A_TOP_SCRPTR                ; 1960
                                    14  13 00ABC        BEQL    150$                                     ;
                                    57  D5 00ABE        TSTL    SCRPTR                                   ; 1961
                                    10  13 00AC0        BEQL    150$                                     ;
                                08  A7  95 00AC2        TSTB    8(SCRPTR)                                ; 1964
                                    02  12 00AC5        BNEQ    149$                                     ;
                                    59  D7 00AC7        DECL    REC_NO                                   ;
                           57       67  D0 00AC9 149$:  MOVL    (SCRPTR), SCRPTR                         ; 1966
                                    E2  13 00ACC        BEQL    148$                                     ; 1968
                                    58  D6 00ACE        INCL    TOP_DIST                                 ;
                                    DE  11 00AD0        BRB     148$                                     ; 1959
                           50       58  D1 00AD2 150$:  CMPL    TOP_DIST, R0                             ; 1977
                                    22  14 00AD5        BGTR    151$                                     ;
```

```
                     00000000G  00              58  D1 00AD7           CMPL    TOP_DIST, EDT$$G_SCLL_TOP                ; 1978
                                                19  19 00ADE           BLSS    151$
                          50                    58  C1 00AE0           ADDL3   TOP_DIST, BELOW, R0                      ; 1979
                     00000000G  00              50  D1 00AE4           CMPL    R0, EDT$$G_SCR_LNS
                                                0C  19 00AEB           BLSS    151$
                     00000000G  00              57  D1 00AED           CMPL    SCRPTR, EDT$$A_TOP_SCRPTR                ; 1980
                                                03  12 00AF4           BNEQ    151$
                                5A              01  D0 00AF6           MOVL    #1, TOP_SET                              ; 1982
                                55              5A  E8 00AF9 151$:      BLBS    TOP_SET, 156$                            ; 1994
                                4E 00000000G    00  E9 00AFC           BLBC    EDT$$G_SCR_REBUILD, 156$
                                50        20    AE  D0 00B03           MOVL    CURSOR_LINE, R0                          ; 2001
                     00000000G  00              50  D1 00B07           CMPL    R0, EDT$$G_SCLL_BOT
                                                07  15 00B0E           BLEQ    152$
                                50 00000000G    00  D0 00B10           MOVL    EDT$$G_SCLL_BOT, R0
                     00000000G  00              50  D1 00B17 152$:      CMPL    R0, EDT$$G_SCLL_TOP
                                                07  18 00B1E           BGEQ    153$
                                50 00000000G    00  D0 00B20           MOVL    EDT$$G_SCLL_TOP, R0
                                57              5B  D0 00B27 153$:      MOVL    R11, SCRPTR                              ; 2002
                                58              01  CE 00B2A           MNEGL   #1, TOP_DIST                             ; 2003
                                                57  D5 00B2D 154$:      TSTL    SCRPTR                                   ; 2005
                                                0C  13 00B2F           BEQL    155$
                                50              58  D1 00B31           CMPL    TOP_DIST, TARGET_LINE
                                                07  13 00B34           BEQL    155$
                                57              67  D0 00B36           MOVL    (SCRPTR), SCRPTR                         ; 2007
                                                58  D6 00B39           INCL    TOP_DIST                                 ; 2008
                                                F0  11 00B3B           BRB     154$                                     ; 2005
                                50              58  D1 00B3D 155$:      CMPL    TOP_DIST, TARGET_LINE                    ; 2011
                                                0F  12 00B40           BNEQ    156$
                                50              56  C0 00B42           ADDL2   BELOW, R0
                     00000000G  00              50  D1 00B45           CMPL    R0, EDT$$G_SCR_LNS
                                                03  19 00B4C           BLSS    156$
                                5A              01  D0 00B4E           MOVL    #1, TOP_SET                              ; 2013
                                76              5A  E8 00B51 156$:      BLBS    TOP_SET, 165$                            ; 2022
                                          08    AE  D5 00B54           TSTL    DIR                                      ; 2036
                                                19  18 00B57           BGEQ    157$
                                50 00000000G    00  D0 00B59           MOVL    EDT$$G_RECS_INSERTED, R0
                          51    50        20    AE  C1 00B60           ADDL3   CURSOR_LINE, R0, R1
                     00000000G  00              51  D1 00B65           CMPL    R1, EDT$$G_SCLL_BOT
                                                0D  15 00B6C           BLEQ    158$
                                                50  D5 00B6E           TSTL    R0                                       ; 2037
                                                09  15 00B70           BLEQ    158$
                                50 00000000G    00  D0 00B72 157$:      MOVL    EDT$$G_SCLL_BOT, TARGET_LINE             ; 2039
                                                07  11 00B79           BRB     159$
                                50 00000000G    00  D0 00B7B 158$:      MOVL    EDT$$G_SCLL_TOP, TARGET_LINE             ; 2041
                52 00000000G    00              56  C3 00B82 159$:      SUBL3   BELOW, EDT$$G_SCR_LNS, R2                ; 2047
                                51              50  D0 00B8A           MOVL    TARGET_LINE, R1
                                52              51  D1 00B8D           CMPL    R1, R2
                                                03  18 00B90           BGEQ    160$
                                51              52  D0 00B92           MOVL    R2, R1
                                50              51  D0 00B95 160$:      MOVL    R1, TARGET_LINE
                                                59  D4 00B98           CLRL    REC_NO                                   ; 2049
                                57              5B  D0 00B9A           MOVL    R11, SCRPTR                              ; 2050
                                58              01  CE 00B9D           MNEGL   #1, TOP_DIST                             ; 2051
                                                57  D5 00BA0 161$:      TSTL    SCRPTR                                   ; 2053
                                                0C  13 00BA2           BEQL    162$
                                50              58  D1 00BA4           CMPL    TOP_DIST, TARGET_LINE
                                                07  13 00BA7           BEQL    162$
```

```
                 57          67 D0 00BA9          MOVL    (SCRPTR), SCRPTR              ; 2055
                             58 D6 00BAC          INCL    TOP_DIST                      ; 2056
                             F0 11 00BAE          BRB     161$                          ; 2053
                 50          58 D1 00BB0 162$:    CMPL    TOP_DIST, TARGET_LINE         ; 2063
                             03 12 00BB3          BNEQ    163$                          ;
                 5A          01 D0 00BB5          MOVL    #1, TOP_SET                   ;
                 0F          5A E8 00BB8 163$:    BLBS    TOP_SET, 165$                 ; 2072
                 58          01 CE 00BBB          MNEGL   #1, TOP_DIST                  ; 2075
                 57          5B D0 00BBE          MOVL    R11, SCRPTR                   ; 2076
                             07 13 00BC1 164$:    BEQL    165$                          ; 2078
                             58 D6 00BC3          INCL    TOP_DIST                      ; 2080
                 57          67 D0 00BC5          MOVL    (SCRPTR), SCRPTR              ; 2081
                             F7 11 00BC8          BRB     164$                          ; 2078
                 57          5B D0 00BCA 165$:    MOVL    R11, SCRPTR                   ; 2089
                             59 D4 00BCD          CLRL    REC_NO                        ; 2090
                             50 D4 00BCF          CLRL    I                             ; 2092
                             0A 11 00BD1          BRB     168$                          ;
                        08   A7 95 00BD3 166$:    TSTB    8(SCRPTR)                     ; 2095
                             02 12 00BD6          BNEQ    167$                          ;
                             59 D7 00BD8          DECL    REC_NO                        ;
                 57          67 D0 00BDA 167$:    MOVL    (SCRPTR), SCRPTR              ; 2097
      F2         50          58 F3 00BDD 168$:    AOBLEQ  TOP_DIST, I, 166$             ; 2092
                 55          59 D0 00BE1          MOVL    REC_NO, TOP_RECNO             ; 2100
                 54          57 D0 00BE4          MOVL    SCRPTR, TOP_SCRPTR            ; 2101
                 51 00000000G 00 D0 00BE7         MOVL    EDT$$A_TOP_SCRPTR, R1         ; 2107
                 54          51 D1 00BEE          CMPL    R1, TOP_SCRPTR                ;
                             61 13 00BF1          BEQL    180$                          ;
                             51 D5 00BF3          TSTL    R1                            ;
                             5D 13 00BF5          BEQL    180$                          ;
                             50 D4 00BF7          CLRL    SEEN_NEW                      ; 2116
                             52 7C 00BF9          CLRQ    SCLL_NUM                      ; 2117
                 57 00000000G 00 D0 00BFB         MOVL    EDT$$A_FST_SCRPTR, SCRPTR     ; 2118
                             2C 13 00C02 169$:    BEQL    175$                          ; 2120
                 0B          53 E9 00C04          BLBC    SEEN_OLD, 170$                ;
                 30          50 E8 00C07          BLBS    SEEN_NEW, 176$                ;
                 05          53 E9 00C0A          BLBC    SEEN_OLD, 170$                ; 2123
                 05          50 E8 00C0D          BLBS    SEEN_NEW, 171$                ;
                             52 D6 00C10          INCL    SCLL_NUM                      ;
                 05          50 E9 00C12 170$:    BLBC    SEEN_NEW, 172$                ; 2125
                 02          53 E8 00C15 171$:    BLBS    SEEN_OLD, 172$                ;
                             52 D7 00C18          DECL    SCLL_NUM                      ;
                 54          57 D1 00C1A 172$:    CMPL    SCRPTR, TOP_SCRPTR            ; 2127
                             03 12 00C1D          BNEQ    173$                          ;
                 50          01 D0 00C1F          MOVL    #1, SEEN_NEW                  ;
                 51          57 D1 00C22 173$:    CMPL    SCRPTR, R1                    ; 2129
                             03 12 00C25          BNEQ    174$                          ;
                 53          01 D0 00C27          MOVL    #1, SEEN_OLD                  ;
                 57     04   A7 D0 00C2A 174$:    MOVL    4(SCRPTR), SCRPTR             ; 2131
                             D2 11 00C2E          BRB     169$                          ; 2120
                 07          50 E8 00C30 175$:    BLBS    SEEN_NEW, 176$                ; 2134
       00000000G 00          00 FB 00C33          CALLS   #0, EDT$$INTER_ERR            ;
                 02          53 E8 00C3A 176$:    BLBS    SEEN_OLD, 177$                ; 2140
                             52 D4 00C3D          CLRL    SCLL_NUM                      ;
                 50          52 D0 00C3F 177$:    MOVL    SCLL_NUM, R0                  ; 2146
                             03 18 00C42          BGEQ    178$                          ;
                 50          50 CE 00C44          MNEGL   R0, R0                        ;
       00000000G 00          50 D1 00C47 178$:    CMPL    R0, EDT$$G_SCR_LNS            ;
```

```
                                     02  19 00C4E       BLSS    179$                                   ;
                                     52  D4 00C50       CLRL    SCLL_NUM                               ;
                                     52  D5 00C52 179$: TSTL    SCLL_NUM                               ; 2154
                                     03  12 00C54 180$: BNEQ    181$                                   ;
                                   0188  31 00C56       BRW     195$                                   ;
                                     03  14 00C59 181$: BGTR    182$                                   ; 2157
                                   00D8  31 00C5B       BRW     188$                                   ;
                  57 00000000G 00    D0 00C5E 182$: MOVL    EDT$$A_BOT_SCRPTR, SCRPTR              ; 2163
                  59         18 AE   D0 00C65       MOVL    OLD_BOT_RECNO, REC_NO                  ; 2164
                  57         04 A7   D0 00C69       MOVL    4(SCRPTR), SCRPTR                      ; 2165
                             08 A7   95 00C6D       TSTB    8(SCRPTR)                              ; 2167
                                     02  12 00C70       BNEQ    183$                                   ;
                                     59  D6 00C72       INCL    REC_NO                                 ;
              18 AE                  59  D0 00C74 183$: MOVL    REC_NO, OLD_BOT_RECNO                  ; 2169
                     20 00000000G    00  E9 00C78       BLBC    EDT$$G_TI_SCROLL, 184$                 ; 2171
                                     7E  D4 00C7F       CLRL    -(SP)                                  ; 2174
        7E 00000000G 00              01  C3 00C81       SUBL3   #1, EDT$$G_SCR_LNS, -(SP)              ;
           00000000G 00              02  FB 00C89       CALLS   #2, EDT$$SC_POSCSIF                    ;
                                     01  DD 00C90       PUSHL   #1                                     ; 2175
                            F35C     CF  9F 00C92       PUSHAB  P.AAA                                  ;
           00000000G 00              02  FB 00C96       CALLS   #2, EDT$$FMT_LIT                       ;
                                     2F  11 00C9D       BRB     185$                                   ; 2171
                                     7E  D4 00C9F 184$: CLRL    -(SP)                                  ; 2179
        7E 00000000G 00              01  C1 00CA1       ADDL3   #1, EDT$$G_MESSAGE_LINE, -(SP)         ;
           00000000G 00              02  FB 00CA9       CALLS   #2, EDT$$SC_POSCSIF                    ;
                                     01  DD 00CB0       PUSHL   #1                                     ; 2180
                            F340     CF  9F 00CB2       PUSHAB  P.AAB                                  ;
           00000000G 00              02  FB 00CB6       CALLS   #2, EDT$$FMT_LIT                       ;
                                     7E  D4 00CBD       CLRL    -(SP)                                  ; 2181
        7E 00000000G 00              01  C3 00CBF       SUBL3   #1, EDT$$G_SCR_LNS, -(SP)              ;
           00000000G 00              02  FB 00CC7       CALLS   #2, EDT$$SC_POSCSIF                    ;
                  18 00000000G       00  E8 00CCE 185$: BLBS    EDT$$G_TI_SCROLL, 186$                 ; 2189
                     00000000G       00  D4 00CD5       CLRL    EDT$$G_MSGFLG                          ; 2192
           00000000G 00 00000000G    00  D0 00CDB       MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO          ; 2193
           00000000G 00              00  FB 00CE6       CALLS   #0, EDT$$SC_ERAALL                     ; 2194
00000000G 00 00000000G 00            01  C3 00CED 186$: SUBL3   #1, EDT$$G_SCR_LNS, EDT$$G_CS_LNO      ; 2197
                                     59  DD 00CF9       PUSHL   REC_NO                                 ; 2198
           00000000G 00              01  FB 00CFB       CALLS   #1, EDT$$SC_MOVTOLN                    ;
                     07              50  E8 00D02       BLBS    R0, 187$                               ;
           00000000G 00              00  FB 00D05       CALLS   #0, EDT$$INTER_ERR                     ;
                     00000000G       00  DD 00D0C 187$: PUSHL   EDT$$G_TI_SCROLL                       ; 2199
                                     57  DD 00D12       PUSHL   SCRPTR                                 ;
           00000000G 00              02  FB 00D14       CALLS   #2, EDT$$SC_RFRELN                     ;
                  50 00000000G       00  D0 00D1B       MOVL    EDT$$A_TOP_SCRPTR, R0                  ; 2200
           00000000G 00         04   A0  D0 00D22       MOVL    4(R0), EDT$$A_TOP_SCRPTR               ;
           00000000G 00              57  D0 00D2A       MOVL    SCRPTR, EDT$$A_BOT_SCRPTR              ; 2201
                                     52  D7 00D31       DECL    SCLL_NUM                               ; 2202
                                   FF1C  31 00D33       BRW     179$                                   ; 2157
                  57 00000000G 00    D0 00D36 188$: MOVL    EDT$$A_TOP_SCRPTR, SCRPTR              ; 2209
                  59         24 AE   D0 00D3D       MOVL    OLD_TOP_RECNO, REC_NO                  ; 2210
                             08 A7   95 00D41       TSTB    8(SCRPTR)                              ; 22'2
                                     02  12 00D44       BNEQ    189$                                   ;
                                     59  D7 00D46       DECL    REC_NO                                 ;
              24 AE                  59  D0 00D48 189$: MOVL    REC_NO, OLD_TOP_RECNO                  ; 2214
                     57              67  D0 00D4C       MOVL    (SCRPTR), SCRPTR                       ; 2215
                                     7E  7C 00D4F       CLRQ    -(SP)                                  ; 2216
           00000000G 00              02  FB 00D51       CALLS   #2, EDT$$SC_POSCSIF                    ;
```

```
                                01 00000000G   00  D1 00D58          CMPL    EDT$$G_TI_TYP, #1                          ; 2218
                                               08  12 00D5F          BNEQ    190$                                       ;
                                               02  DD 00D61          PUSHL   #2                                         ; 2220
                                     F293      CF  9F 00D63          PUSHAB  P.AAC                                      ;
                                               06  11 00D67          BRB     191$                                       ;
                                               02  DD 00D69 190$:     PUSHL   #2                                         ; 2222
                                     F28F      CF  9F 00D6B          PUSHAB  P.AAD                                      ;
                  00000000G   00               02  FB 00D6F 191$:     CALLS   #2, EDT$$FMT_LIT                           ;
                              18 00000000G   00  E8 00D76          BLBS    EDT$$G_TI_SCROLL, 192$                     ; 2235
                                 00000000G   00  D4 00D7D          CLRL    EDT$$G_MSGFLG                              ; 2238
                  00000000G   00 00000000G   00  D0 00D83          MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO              ; 2239
                  00000000G   00               00  FB 00D8E          CALLS   #0, EDT$$SC_ERAALL                         ; 2240
                                 00000000G   00  D4 00D95 192$:     CLRL    EDT$$G_CS_LNO                              ; 2243
                                               59  DD 00D9B          PUSHL   REC_NO                                     ; 2244
                  00000000G   00               01  FB 00D9D          CALLS   #1, EDT$$SC_MOVTOLN                        ;
                              07               50  E8 00DA4          BLBS    R0, 193$                                   ;
                  00000000G   00               00  FB 00DA7          CALLS   #0, EDT$$INTER_ERR                         ;
                                               01  DD 00DAE 193$:     PUSHL   #1                                         ; 2245
                                               57  DD 00DB0          PUSHL   SCRPTR                                     ;
                  00000000G   00               02  FB 00DB2          CALLS   #2, EDT$$SC_RFRELN                         ;
                  00000000G   00               57  D0 00DB9          MOVL    SCRPTR, EDT$$A_TOP_SCRPTR                  ; 2246
                              50 00000000G   00  D0 00DC0          MOVL    EDT$$A_BOT_SCRPTR, R0                      ; 2247
                  00000000G   00               60  D0 00DC7          MOVL    (R0), EDT$$A_BOT_SCRPTR                    ;
                                               52  D6 00DCE          INCL    SCLL_NUM                                   ; 2248
                                          FE7F 31  00DD0          BRW     179$                                       ; 2154
                              54 00000000G   00  D0 00DD3 194$:     MOVL    EDT$$A_TOP_SCRPTR, TOP_SCRPTR              ; 2258
                              55 00000000G   00  D0 00DDA          MOVL    EDT$$G_TOP_RECNO, TOP_RECNO                ; 2259
                              0B       14    AE  E8 00DE1 195$:     BLBS    SCROLLING_NEEDED, 196$                     ; 2271
                              07       10    AE  E8 00DE5          BLBS    ERASE_ALL, 196$                            ;
                              03       0C    AE  E8 00DE9          BLBS    BUILD_SCR, 196$                            ;
                                          00A9 31  00DED          BRW     204$                                       ;
                              59               55  D0 00DF0 196$:     MOVL    TOP_RECNO, REC_NO                          ; 2274
                              57               54  D0 00DF3          MOVL    TOP_SCRPTR, SCRPTR                         ; 2275
                        20    AE               01  CE 00DF6          MNEGL   #1, CURSOR_LINE                            ; 2276
                                 00000000G   00  D4 00DFA          CLRL    EDT$$G_CS_LNO                              ; 2277
                  00000000G   00 00000000G   00  D1 00E00 197$:     CMPL    EDT$$G_CS_LNO, EDT$$G_SCR_LNS              ; 2279
                                               58  18 00E0B          BGEQ    201$                                       ;
                                               57  D5 00E0D          TSTL    SCRPTR                                     ;
                                               54  13 00E0F          BEQL    201$                                       ;
                              07       0D    A7  93 00E11          BITB    13(SCRPTR), #7                             ; 2282
                                               1F  13 00E15          BEQL    199$                                       ;
                                               59  DD 00E17          PUSHL   REC_NO                                     ; 2285
                  00000000G   00               01  FB 00E19          CALLS   #1, EDT$$SC_MOVTOLN                        ;
                              07               50  E8 00E20          BLBS    R0, 198$                                   ;
                  00000000G   00               00  FB 00E23          CALLS   #0, EDT$$INTER_ERR                         ;
                                     10        AE  DD 00E2A 198$:     PUSHL   ERASE_ALL                                  ; 2286
                                               57  DD 00E2D          PUSHL   SCRPTR                                     ;
                  00000000G   00               02  FB 00E2F          CALLS   #2, EDT$$SC_RFRELN                         ;
                  00000000G   00               57  D0 00E36 199$:     MOVL    SCRPTR, EDT$$A_BOT_SCRPTR                  ; 2289
                  00000000G   00               57  D1 00E3D          CMPL    SCRPTR, EDT$$A_CUR_SCRPTR                  ; 2291
                                               08  12 00E44          BNEQ    200$                                       ;
                        20    AE 00000000G   00  D0 00E46          MOVL    EDT$$G_CS_LNO, CURSOR_LINE                 ;
                              57       04    A7  D0 00E4E 200$:     MOVL    4(SCRPTR), SCRPTR                          ; 2293
                                 00000000G   00  D6 00E52          INCL    EDT$$G_CS_LNO                              ; 2294
                                               57  D5 00E58          TSTL    SCRPTR                                     ; 2296
                                               A4  13 00E5A          BEQL    197$                                       ;
                                     08        A7  95 00E5C          TSTB    8(SCRPTR)                                  ; 2300
```

```
                                        9F   12 00E5F        BNEQ    197$                                    ;
                                        59   D6 00E61        INCL    REC_NO                                  ;
                                        9B   11 00E63        BRB     197$                                    ; 2279
                                        59   D4 00E65 201$:  CLRL    REC_NO                                  ; 2309
                                        59   DD 00E67        PUSHL   REC_NO                                  ; 2310
                    00000000G  00       01   FB 00E69        CALLS   #1, EDT$$SC_MOVTOLN                     ;
                               07       50   E8 00E70        BLBS    R0, 202$                                ;
                    00000000G  00       00   FB 00E73        CALLS   #0, EDT$$INTER_ERR                      ;
                    00000000G  00 00000000G  00   D1 00E7A 202$:  CMPL    EDT$$G_CS_LNO, EDT$$G_BOT_LINE     ; 2315
                                        07   18 00E85        BGEQ    203$                                    ;
                    00000000G  00       00   FB 00E87        CALLS   #0, EDT$$SC_ERAALL                      ;
                    00000000G  00 00000000G  00   D0 00E8E 203$:  MOVL    EDT$$G_CS_LNO, EDT$$G_BOT_LINE     ; 2317
                    00000000G  00       55   D0 00E99 204$:  MOVL    TOP_RECNO, EDT$$G_TOP_RECNO             ; 2323
                    00000000G  00       54   D0 00EA0        MOVL    TOP_SCRPTR, EDT$$A_TOP_SCRPTR           ; 2324
                               00000000G  00   D4 00EA7        CLRL    EDT$$G_SCR_REBUILD                    ; 2325
                               00000000G  00   D4 00EAD        CLRL    EDT$$G_ANY_CHANGES                    ; 2326
                    00000000G  00 00000000G  00   D0 00EB3        MOVL    EDT$$A_CUR_SCRPTR, EDT$$A_CSR_SCRPTR ; 2327
                               00000000G  00   D4 00EBE        CLRL    EDT$$G_RECS_INSERTED                  ; 2328
                    00000000G  00 00000000G  00   D0 00EC4        MOVL    EDT$$A_SEL_BUF, EDT$$A_OLD_SEL     ; 2329
                    00000000G  00 00000000G  00   D0 00ECF        MOVL    EDT$$G_CS_CHNO, EDT$$G_CS_OLDCHNO  ; 2330
                                   1C   AE   DD 00EDA        PUSHL   CURSOR_POS                              ; 2331
                                   24   AE   DD 00EDD        PUSHL   CURSOR_LINE                             ;
                    00000000G  00       02   FB 00EE0        CALLS   #2, EDT$$SC_POSCSIF                     ;
                    00000000G  00  1C   AE   D0 00EE7        MOVL    CURSOR_POS, EDT$$G_CUR_COL              ; 2332
                    00000000G  00  20   AE   D0 00EEF        MOVL    CURSOR_LINE, EDT$$G_CS_LNO              ; 2333
                                        01   DD 00EF7        PUSHL   #1                                      ; 2334
                    00000000G  00       01   FB 00EF9        CALLS   #1, EDT$$TI_ENBLAUTREP                  ;
                    00000000G  00       00   FB 00F00        CALLS   #0, EDT$$OUT_FMTBUF                     ; 2335
                               50 00000000G  00   9E 00F07        MOVAB   EDT$$T_LN_BUF, R0                  ; 2336
      00000000G  00            50 00000000G  00   C1 00F0E        ADDL3   EDT$$G_CS_CHNO, R0, EDT$$A_LN_PTR  ;
                                        04 00F1A             RET                                             ; 2337

; Routine Size:  3867 bytes,    Routine Base:  _EDT$CODE + 000E


; 1768         2338  1
```

```
;  1770          2339  1 %SBTTL 'DELETE_LINE - delete a line on the screen'
;  1771          2340  1 ROUTINE DELETE_LINE (                                  ! Delete a line on the screen
;  1772          2341  1     SCRPTR,                                            ! The line to delete
;  1773          2342  1     OLD_TOP_RECNO                                      ! Record number of the top line
;  1774          2343  1     ) =
;  1775          2344  1
;  1776          2345  1 !++
;  1777          2346  1 ! FUNCTIONAL DESCRIPTION:
;  1778          2347  1 !
;  1779          2348  1 !       Delete one screen line.
;  1780          2349  1 !
;  1781          2350  1 ! FORMAL PARAMETERS:
;  1782          2351  1 !
;  1783          2352  1 !  SCRPTR               The screen data block to delete
;  1784          2353  1 !
;  1785          2354  1 !  OLD_TOP_RECNO        Record number of the top line
;  1786          2355  1 !
;  1787          2356  1 ! IMPLICIT INPUTS:
;  1788          2357  1 !
;  1789          2358  1 !       EDT$$G_FST_SCRPTR
;  1790          2359  1 !       EDT$$G_LST_SCRPTR
;  1791          2360  1 !       EDT$$G_TI_SCROLL
;  1792          2361  1 !       EDT$$A_TOP_SCRPTR
;  1793          2362  1 !       EDT$$G_SCR_LNS
;  1794          2363  1 !       EDT$$G_SCLL_TOP
;  1795          2364  1 !       EDT$$G_SCLL_BOT
;  1796          2365  1 !       EDT$$G_BOT_SCRPTR
;  1797          2366  1 !       EDT$$G_CS_LNO
;  1798          2367  1 !       EDT$$A_LST_SCRPTR
;  1799          2368  1 !       EDT$$G_BOT_LINE
;  1800          2369  1 !
;  1801          2370  1 ! IMPLICIT OUTPUTS:
;  1802          2371  1 !
;  1803          2372  1 !       EDT$$A_TOP_SCRPTR
;  1804          2373  1 !       EDT$$A_BOT_SCRPTR
;  1805          2374  1 !       EDT$$G_BOT_LINE
;  1806          2375  1 !       EDT$$G_MSGFLG
;  1807          2376  1 !
;  1808          2377  1 ! ROUTINE VALUE:
;  1809          2378  1 !
;  1810          2379  1 !       0 = must start update over
;  1811          2380  1 !
;  1812          2381  1 ! SIDE EFFECTS:
;  1813          2382  1 !
;  1814          2383  1 !       Will store into the format buffer
;  1815          2384  1 !
;  1816          2385  1 !--
;  1817          2386  1
;  1818          2387  2     BEGIN
;  1819          2388  2
;  1820          2389  2     MAP
;  1821          2390  2         SCRPTR : REF SCREEN_LINE;
;  1822          2391  2
;  1823          2392  2     EXTERNAL ROUTINE
;  1824          2393  2         EDT$$SC_ERAALL,                                ! Erase to end of screen
;  1825          2394  2         EDT$$SC_MOVTOLN,                   ! Move to a record in the work file relative to the current record
;  1826          2395  2         EDT$$SC_RFRELN : NOVALUE,                      ! Refresh a screen line
```

```
;  1827          2396  2          EDT$$SC_LNDEL,                          ! Delete a line from data structure
;  1828          2397  2          EDT$$SC_SETSCLLREG,                     ! Set scrolling region
;  1829          2398  2          EDT$$FMT_LIT,                           ! Format a literal for output
;  1830          2399  2          EDT$$SC_POSCSIF : NOVALUE,              ! Position the cursor
;  1831          2400  2          EDT$$SC_REPAINT : NOVALUE;              ! Mark some lines in the screen data base for repaint
;  1832          2401  2
;  1833          2402  2      EXTERNAL
;  1834          2403  2          EDT$$G_MSGFLG,                          ! Is a message on the screen
;  1835          2404  2          EDT$$G_MESSAGE_LINE,                    ! Line number for prompts
;  1836          2405  2          EDT$$G_TI_TYP,                          ! Terminal type
;  1837          2406  2          EDT$$G_TI_EDIT,                         ! VT102 editing features
;  1838          2407  2          EDT$$G_SCR_LNS,                         ! Number of lines on screen
;  1839          2408  2          EDT$$G_SCLL_TOP,                        ! Top line the cursor is allowed to be on
;  1840          2409  2          EDT$$G_SCLL_BOT,                        ! Bottom line the cursor is allowed to be on
;  1841          2410  2          EDT$$A_FST_SCRPTR : REF SCREEN_LINE,    ! First data structure pointer
;  1842          2411  2          EDT$$A_BOT_SCRPTR : REF SCREEN_LINE,    ! Bottom screen pointer
;  1843          2412  2          EDT$$A_LST_SCRPTR : REF SCREEN_LINE,    ! Last data structure pointer
;  1844          2413  2          EDT$$A_TOP_SCRPTR : REF SCREEN_LINE,    ! Top screen line info address
;  1845          2414  2          EDT$$A_EOB_SCRPTR : REF SCREEN_LINE,    ! EOB screen pointer
;  1846          2415  2          EDT$$G_CS_LNO,                          ! current cursor line
;  1847          2416  2          EDT$$G_TI_SCROLL,                       ! 1 = we have scrolling regions
;  1848          2417  2          EDT$$G_BOT_LINE;                        ! All lines below this one have been erased
;  1849          2418  2
;  1850          2419  2      LOCAL
;  1851          2420  2          PRV_SCRPTR : REF SCREEN_LINE,           ! Previous screen pointer
;  1852          2421  2          NXT_SCRPTR : REF SCREEN_LINE,           ! Next screen pointer
;  1853          2422  2          TOP_SCRPTR : REF SCREEN_LINE,           ! Top screen pointer (at entry)
;  1854          2423  2          SCLL_CENTER;                            ! Number of center screen line
;  1855          2424  2
;  1856          2425  2      PRV_SCRPTR = .SCRPTR [SCR_PRV_LINE];
;  1857          2426  2      NXT_SCRPTR = .SCRPTR [SCR_NXT_LINE];
;  1858          2427  2      TOP_SCRPTR = .EDT$$A_TOP_SCRPTR;
;  1859          2428  2      SCLL_CENTER = (.EDT$$G_SCLL_TOP + .EDT$$G_SCLL_BOT)/2;
;  1860          2429  2  !+
;  1861          2430  2  ! If we are deleting the top line, make the following line the top line
;  1862          2431  2  ! unless there are lines preceding it that have been modified or inserted,
;  1863          2432  2  ! in which case we will paint over the deleted line instead of scrolling.
;  1864          2433  2  !-
;  1865          2434  2
;  1866          2435  3      IF (.SCRPTR EQLA .TOP_SCRPTR)
;  1867          2436  2      THEN
;  1868          2437  3          BEGIN
;  1869          2438  3
;  1870          2439  4          IF (.PRV_SCRPTR NEQA 0)
;  1871          2440  3          THEN
;  1872          2441  3  !+
;  1873          2442  3  ! Test for insertion or modification of previous lines.
;  1874          2443  3  !-
;  1875          2444  3
;  1876          2445  4              IF ((.PRV_SCRPTR [SCR_EDIT_FLAGS] AND (SCR_EDIT_INSLN OR SCR_EDIT_MODIFY)) NEQ 0)
;  1877          2446  3              THEN
;  1878          2447  4                  BEGIN
;  1879          2448  4  !+
;  1880          2449  4  ! Delete the current line and backup to the previous line for a
;  1881          2450  4  ! repaint.  The top screen pointer and record number offsets
;  1882          2451  4  ! must also be updated.  No further processing is needed on this pass.
;  1883          2452  4  ! The new top line will be repainted on the next pass.
```

```
;  1884          2453  4 !-
;  1885          2454  4
;  1886          2455  4                 IF (.SCRPTR [SCR_LINE_IDX] EQL 0) THEN .OLD_TOP_RECNO = ..OLD_TOP_RECNO - 1;
;  1887          2456  4
;  1888          2457  4                 EDT$$SC_LNDEL (.SCRPTR);
;  1889          2458  4                 EDT$$A_TOP_SCRPTR = .PRV_SCRPTR;
;  1890          2459  4                 RETURN (0);
;  1891          2460  3                 END;
;  1892          2461  3
;  1893          2462  3 !+
;  1894          2463  3 ! The previous line is non-existent or has not been modified.  Handle this in the usual way.
;  1895          2464  3 !-
;  1896          2465  2         END;
;  1897          2466  2
;  1898          2467  4     IF (((.EDT$$A_EOB_SCRPTR EQLA .EDT$$A_BOT_SCRPTR) OR (.EDT$$G_CS_LNO LEQ .SCLL_CENTER))     !
;  1899          2468  3         AND (.TOP_SCRPTR [SCR_PRV_LINE] NEQA 0))
;  1900          2469  2     THEN
;  1901          2470  3         BEGIN
;  1902          2471  3 !+
;  1903          2472  3 ! The buffer is long enough that there is a line before the top line that appears on the screen,
;  1904          2473  3 ! and either the [EOB] appears on the screen or the line being deleted is above the center of
;  1905          2474  3 ! the screen.  We will be scrolling down, bringing a new line onto the top screen line.
;  1906          2475  3 ! Move the top screen pointer up one line.  This must be done before deleting
;  1907          2476  3 ! the current line in case the top line is the current line.
;  1908          2477  3 !-
;  1909          2478  3         EDT$$A_TOP_SCRPTR = .TOP_SCRPTR [SCR_PRV_LINE];
;  1910          2479  3 !+
;  1911          2480  3 ! Adjust the record number of the top line.
;  1912          2481  3 !-
;  1913          2482  3
;  1914          2483  3         IF (.TOP_SCRPTR [SCR_LINE_IDX] EQL 0) THEN .OLD_TOP_RECNO = ..OLD_TOP_RECNO - 1;
;  1915          2484  3
;  1916          2485  3         IF .EDT$$G_TI_SCROLL
;  1917          2486  3         THEN
;  1918          2487  4             BEGIN
;  1919          2488  4
;  1920          2489  4             LOCAL
;  1921          2490  4                 TMP_SCRPTR : REF SCREEN_LINE,
;  1922          2491  4                 SCLL_LINE;
;  1923          2492  4
;  1924          2493  4 !+
;  1925          2494  4 ! Use the scrolling region or DL command to avoid having to repaint part of the screen.
;  1926          2495  4 !-
;  1927          2496  4             SCLL_LINE = .EDT$$G_CS_LNO;
;  1928          2497  4             TMP_SCRPTR = .SCRPTR;
;  1929          2498  4 !+
;  1930          2499  4 ! To speed up deletes, the scrolling line will be the last line
;  1931          2500  4 ! for which we will scroll down.
;  1932          2501  4 !-
;  1933          2502  4
;  1934          2503  4             DO
;  1935          2504  5                 BEGIN
;  1936          2505  5                 SCLL_LINE = .SCLL_LINE + 1;
;  1937          2506  5                 TMP_SCRPTR = .TMP_SCRPTR [SCR_NXT_LINE];
;  1938          2507  5                 END
;  1939          2508  5             UNTIL (((.TMP_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0) OR          !
;  1940          2509  4                 ((.SCLL_LINE GTR .SCLL_CENTER) AND (.EDT$$A_EOB_SCRPTR NEQA .EDT$$A_BOT_SCRPTR)));
```

```
;  1941          2510  4
;  1942          2511  4 !+
;  1943          2512  4 ! Scroll down
;  1944          2513  4 !-
;  1945          2514  4             EDT$$SC_SETSCLLREG (0, .SCLL_LINE);
;  1946          2515  4             EDT$$SC_POSCSIF (0, 0);
;  1947          2516  4             EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'M')), 2);
;  1948          2517  4             EDT$$SC_LNDEL (.SCRPTR);
;  1949          2518  4             END
;  1950          2519  3         ELSE
;  1951          2520  4             BEGIN
;  1952          2521  4 !+
;  1953          2522  4 ! We do not have scrolling regions.  If the line to be deleted is high
;  1954          2523  4 ! on the screen then repaint from the top line to the current line.  If
;  1955          2524  4 ! the line to be deleted is low on the screen, scroll the screen down
;  1956          2525  4 ! and repaint the lines below this one.
;  1957          2526  4 !-
;  1958          2527  4
;  1959          2528  5             IF (.EDT$$G_CS_LNO LEQ (.EDT$$G_BOT_LINE/2))
;  1960          2529  4             THEN
;  1961          2530  5                 BEGIN
;  1962          2531  5 !+
;  1963          2532  5 ! We have a high line, so we will repaint the top of the screen.
;  1964          2533  5 ! Ignore deleted lines right after this one, since there could be enough to
;  1965          2534  5 ! make us decide to also repaint the bottom half of the screen.
;  1966          2535  5 !-
;  1967          2536  5                     EDT$$SC_LNDEL (.SCRPTR);
;  1968          2537  5
;  1969          2538  6                     IF ((.SCRPTR NEQA .EDT$$A_BOT_SCRPTR) AND       !
;  1970          2539  6                         ((.NXT_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0))
;  1971          2540  5                     THEN
;  1972          2541  5                         DELETE_LINE (.NXT_SCRPTR, .OLD_TOP_RECNO);
;  1973          2542  5
;  1974          2543  5                     IF (.PRV_SCRPTR NEQA 0) THEN EDT$$SC_REPAINT (.EDT$$A_TOP_SCRPTR, .PRV_SCRPTR, 0, 255, 0);
;  1975          2544  5
;  1976          2545  5                     END
;  1977          2546  4                 ELSE
;  1978          2547  5                     BEGIN
;  1979          2548  5 !+
;  1980          2549  5 ! We have a low line.  Scroll the whole screen down and repaint the lines below the deleted line.
;  1981          2550  5 !-
;  1982          2551  5                     EDT$$SC_POSCSIF (0, 0);
;  1983          2552  5
;  1984          2553  6                     IF (.EDT$$G_TI_TYP EQL TERM_VT52)       !
;  1985          2554  5                     THEN
;  1986          2555  5                         EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'I')), 2)
;  1987          2556  5                     ELSE
;  1988          2557  5                         EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'M')), 2);
;  1989          2558  5
;  1990          2559  5 !+
;  1991          2560  5 ! Make sure the message area is blank, since we may have moved text into the message area.
;  1992          2561  5 !-
;  1993          2562  6                     BEGIN
;  1994          2563  6
;  1995          2564  6                     LOCAL
;  1996          2565  6                         SAV_CS_LN;
;  1997          2566  6
```

```
;  1998          2567  6                   EDT$$G_MSGFLG = 0;
;  1999          2568  6                   SAV_CS_LN = .EDT$$G_CS_LNO;
;  2000          2569  6                   EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;  2001          2570  6                   EDT$$SC_ERAALL ();
;  2002          2571  6                   EDT$$G_CS_LNO = .SAV_CS_LN;
;  2003          2572  5                   END;
;  2004          2573  5 !+
;  2005          2574  5 ! Now mark the bottom of the screen to be repainted.
;  2006          2575  5 !-
;  2007          2576  5                   EDT$$SC_REPAINT (.SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
;  2008          2577  4                   END;
;  2009          2578  4
;  2010          2579  3               END;
;  2011          2580  3
;  2012          2581  3           END
;  2013          2582  2       ELSE
;  2014          2583  3           BEGIN
;  2015          2584  3 !+
;  2016          2585  3 ! Either there is no line before the current top line or the cursor is below the
;  2017          2586  3 ! center of the screen.  Scroll up, which will bring a new line onto the
;  2018          2587  3 ! bottom line of the screen, unless the [EOB] is already on the screen.
;  2019          2588  3 ! If we are deleting the top line, make the next line the top line.  We don't need to
;  2020          2589  3 ! worry about the top line record number since deleted lines have the record number
;  2021          2590  3 ! of the next following non-deleted line.
;  2022          2591  3 !-
;  2023          2592  3
;  2024          2593  3           IF (.SCRPTR EQLA .TOP_SCRPTR) THEN EDT$$A_TOP_SCRPTR = .NXT_SCRPTR;
;  2025          2594  3
;  2026          2595  4           IF ((.EDT$$G_TI_SCROLL) AND (.EDT$$G_CS_LNO LSS (.EDT$$G_SCR_LNS - 1)))
;  2027          2596  3           THEN
;  2028          2597  4               BEGIN
;  2029          2598  4 !+
;  2030          2599  4 ! Set the scrolling region so we can update the screen without repainting text
;  2031          2600  4 ! that has moved up.
;  2032          2601  4 !-
;  2033          2602  4
;  2034          2603  5               IF (.EDT$$G_TI_EDIT)
;  2035          2604  4               THEN
;  2036          2605  4 !+
;  2037          2606  4 ! Use VT102 edit feature
;  2038          2607  4 !-
;  2039          2608  5                   BEGIN
;  2040          2609  5                   EDT$$SC_SETSCLLREG (0, .EDT$$G_SCR_LNS);
;  2041          2610  5                   EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
;  2042          2611  5                   EDT$$FMT_LIT (UPLIT (%STRING (%CHAR (ASC_K_ESC), 'EM')), 3);
;  2043          2612  5                   END
;  2044          2613  4               ELSE
;  2045          2614  5                   BEGIN
;  2046          2615  5 !+
;  2047          2616  5 ! Simulate edit feature.
;  2048          2617  5 !-
;  2049          2618  5                   EDT$$SC_SETSCLLREG (.EDT$$G_CS_LNO, .EDT$$G_SCR_LNS);
;  2050          2619  5                   EDT$$SC_POSCSIF (.EDT$$G_SCR_LNS - 1, 0);
;  2051          2620  5                   EDT$$FMT_LIT (UPLIT (%CHAR (ASC_K_LF)), 1);
;  2052          2621  4                   END;
;  2053          2622  4
;  2054          2623  4 !+
```

```
;  2055     2624  4 ! Free the deleted line.
;  2056     2625  4 !-
;  2057     2626  4             EDT$$SC_LNDEL (.SCRPTR);
;  2058     2627  4             END
;  2059     2628  3         ELSE
;  2060     2629  4             BEGIN
;  2061     2630  4 !+
;  2062     2631  4 ! If we cannot use the scrolling region, repaint the screen from the point
;  2063     2632  4 ! of the deletion to the bottom.  If we're deleting a high line, then
;  2064     2633  4 ! scroll up instead of repainting the whole bottom.
;  2065     2634  4 !-
;  2066     2635  4
;  2067     2636  5             IF (.EDT$$G_CS_LNO LEQ (.EDT$$G_BOT_LINE/2))
;  2068     2637  4             THEN
;  2069     2638  5                 BEGIN
;  2070     2639  5 !+
;  2071     2640  5 ! The line being deleted is high on the screen.  Scroll the screen up and repaint
;  2072     2641  5 ! all of the lines above the deleted line.
;  2073     2642  5 !-
;  2074     2643  5
;  2075     2644  5                 LOCAL
;  2076     2645  5                     SAV_CS_LN;
;  2077     2646  5
;  2078     2647  5                 EDT$$SC_REPAINT (.TOP_SCRPTR, .SCRPTR, 0, 255, 1);
;  2079     2648  5                 SAV_CS_LN = .EDT$$G_CS_LNO;
;  2080     2649  5                 EDT$$SC_POSCSIF (.EDT$$G_MESSAGE_LINE + 1, 0);
;  2081     2650  5                 EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_LF)), 1);
;  2082     2651  5 !+
;  2083     2652  5 ! If we've done a scroll then the message is no longer on the screen.
;  2084     2653  5 !-
;  2085     2654  5                 EDT$$G_MSGFLG = 0;
;  2086     2655  5                 EDT$$G_BOT_LINE = .EDT$$G_BOT_LINE + 1;
;  2087     2656  5                 EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;  2088     2657  5                 EDT$$SC_ERAALL ();
;  2089     2658  5                 EDT$$G_CS_LNO = .SAV_CS_LN;
;  2090     2659  5                 END
;  2091     2660  4             ELSE
;  2092     2661  5                 BEGIN
;  2093     2662  5 !+
;  2094     2663  5 ! The line being deleted is low on the screen.  Repaint it and all lines below it.
;  2095     2664  5 !-
;  2096     2665  5                 EDT$$SC_REPAINT (.SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
;  2097     2666  4                 END;
;  2098     2667  4
;  2099     2668  3             END;
;  2100     2669  3
;  2101     2670  3 !+
;  2102     2671  3 ! Adjust the bottom screen pointer if it is not the EOB.
;  2103     2672  3 !-
;  2104     2673  3
;  2105     2674  4         IF ((.EDT$$A_BOT_SCRPTR NEQA 0) AND (.EDT$$A_BOT_SCRPTR NEQA .EDT$$A_EOB_SCRPTR))
;  2106     2675  3         THEN
;  2107     2676  3             EDT$$A_BOT_SCRPTR = .EDT$$A_BOT_SCRPTR [SCR_NXT_LINE];
;  2108     2677  3
;  2109     2678  3         END;
;  2110     2679  3
;  2111     2680  2 !+
```

```
; 2112      2681  2 ! Make another pass over the screen data.
; 2113      2682  2 !-
; 2114      2683  2     RETURN (0);
; 2115      2684  1     END:                                  ! of routine DELETE_LINE


                              00F29           .BLKB   3
                      4D  1B  00F2C P.AAE:    .BYTE   27, 77                                   ;
                              00F2E           .BLKB   2
                      49  1B  00F30 P.AAF:    .BYTE   27, 73                                   ;
                              00F32           .BLKB   2
                      4D  1B  00F34 P.AAG:    .BYTE   27, 77                                   ;
                              00F36           .BLKB   2
              00  4D  5B  1B  00F38 P.AAH:    .ASCII  <27>\[M\<0>                              ;
              00  00  00  0A  00F3C P.AAI:    .ASCII  <10><0><0><0>                            ;
                          0A  00F40 P.AAJ:    .BYTE   10                                       ;

                                              .EXTRN  EDT$$SC_LNDEL, EDT$$G_TI_EDIT

                        0FFC 00000 DELETE_LINE:
                                              .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11     ; 2340
        5B 00000000G  00  9E 00002            MOVAB   EDT$$G_SCR_LNS, R11                      ;
        5A 00000000G  00  9E 00009            MOVAB   EDT$$SC_POSCSIF, R10                     ;
        59 00000000G  00  9E 00010            MOVAB   EDT$$G_CS_LNO, R9                        ;
        52         04 AC  D0 00017            MOVL    SCRPTR, R2                               ; 2425
        54            62  7D 0001B            MOVQ    (R2), PRV_SCRPTR                         ;
        53 00000000G  00  D0 0001E            MOVL    EDT$$A_TOP_SCRPTR, TOP_SCRPTR            ; 2427
50 00000000G  00 00000000G  00  C1 00025      ADDL3   EDT$$G_SCLL_BOT, EDT$$G_SCLL_TOP, R0     ; 2428
58            50            02  C7 00031      DIVL3   #2, R0, SCLL_CENTER                      ;
                            57  D4 00035      CLRL    R7                                       ; 2435
        53                  52  D1 00037      CMPL    R2, TOP_SCRPTR                           ;
                            26  12 0003A      BNEQ    2$                                       ;
                            57  D6 0003C      INCL    R7                                       ;
                            54  D5 0003E      TSTL    PRV_SCRPTR                               ; 2439
                            20  13 00040      BEQL    2$                                       ;
        03         0D       A4  93 00042      BITB    13(PRV_SCRPTR), #3                       ; 2445
                            1A  13 00046      BEQL    2$                                       ;
                   08       A2  95 00048      TSTB    8(R2)                                    ; 2455
                            03  12 0004B      BNEQ    1$                                       ;
                   08       BC  D7 0004D      DECL    @OLD_TOP_RECNO                           ;
                            52  DD 00050 1$:  PUSHL   R2                                       ; 2457
     00000000G  00          01  FB 00052      CALLS   #1, EDT$$SC_LNDEL                        ;
     00000000G  00          54  D0 00059      MOVL    PRV_SCRPTR, EDT$$A_TOP_SCRPTR            ; 2458
                            7C  11 00060      BRB     8$                                       ; 2459
        50 00000000G  00    D0 00062 2$:      MOVL    EDT$$G_TI_SCROLL, R0                     ; 2485
        56 00000000G  00    D0 00069          MOVL    EDT$$A_EOB_SCRPTR, R6                    ; 2467
        51 00000000G  00    D0 00070          MOVL    EDT$$A_BOT_SCRPTR, R1                    ;
        51                  56  D1 00077      CMPL    R6, R1                                   ;
                            08  13 0007A      BEQL    4$                                       ;
        58                  69  D1 0007C      CMPL    EDT$$G_CS_LNO, SCLL_CENTER               ;
                            03  15 0007F      BLEQ    4$                                       ;
                      00F4  31 00081 3$:      BRW     16$                                      ;
                            63  D5 00084 4$:  TSTL    (TOP_SCRPTR)                             ; 2468
                            F9  13 00086      BEQL    3$                                       ;
     00000000G  00          63  D0 00088      MOVL    (TOP_SCRPTR), EDT$$A_TOP_SCRPTR          ; 2478
                   08       A3  95 0008F      TSTB    8(TOP_SCRPTR)                            ; 2483
```

```
                                 03  12 00092        BNEQ    5$                                      :
                          08     BC  D7 00094        DECL    @OLD_TOP_RECNO                          :
                 57              69  D0 00097 5$:    MOVL    EDT$$G_CS_LNO, R7                       : 2496
                 44              50  E9 0009A        BLBC    R0, 9$                                  :
                 53              57  D0 0009D        MOVL    R7, SCLL_LINE                           :
                 50              52  D0 000A0        MOVL    R2, TMP_SCRPTR                          : 2497
                                 53  D6 000A3 6$:    INCL    SCLL_LINE                               : 2505
                 50       04     A0  D0 000A5        MOVL    4(TMP_SCRPTR), TMP_SCRPTR               : 2506
     0A        0D        A0      02  E1 000A9        BBC     #2, 13(TMP_SCRPTR), 7$                  : 2508
                 58              53  D1 000AE        CMPL    SCLL_LINE, SCLL_CENTER                  : 2509
                                 F0  15 000B1        BLEQ    6$                                      :
                 51              56  D1 000B3        CMPL    R6, R1                                  :
                                 EB  13 000B6        BEQL    6$                                      :
                                 53  DD 000B8 7$:    PUSHL   SCLL_LINE                               : 2514
                                 7E  D4 000BA        CLRL    -(SP)                                   :
          00000000G  00          02  FB 000BC        CALLS   #2, EDT$$SC_SETSCLLREG                  :
                                 7E  7C 000C3        CLRQ    -(SP)                                   : 2515
                 6A              02  FB 000C5        CALLS   #2, EDT$$SC_POSCSIF                     :
                                 02  DD 000C8        PUSHL   #2                                      : 2516
                        FF1D     CF  9F 000CA        PUSHAB  P.AAE                                   :
          00000000G  00          02  FB 000CE        CALLS   #2, EDT$$FMT_LIT                        :
                                 52  DD 000D5        PUSHL   R2                                      : 2517
          00000000G  00          01  FB 000D7        CALLS   #1, EDT$$SC_LNDEL                       :
                            0186     31 000DE 8$:    BRW     23$                                     : 2485
     51   00000000G  00          02  C7 000E1 9$:    DIVL3   #2, EDT$$G_BOT_LINE, R1                 : 2528
                 51              57  D1 000E9        CMPL    R7, R1                                  :
                                 37  14 000EC        BGTR    11$                                     :
                                 52  DD 000EE        PUSHL   R2                                      : 2536
          00000000G  00          01  FB 000F0        CALLS   #1, EDT$$SC_LNDEL                       :
          00000000G  00          52  D1 000F7        CMPL    R2, EDT$$A_BOT_SCRPTR                   : 2538
                                 0F  13 000FE        BEQL    10$                                     :
     0A        0D        A5      02  E1 00100        BBC     #2, 13(NXT_SCRPTR), 10$                 : 2539
                          08     AC  DD 00105        PUSHL   OLD_TOP_RECNO                           : 2541
                                 55  DD 00108        PUSHL   NXT_SCRPTR                              :
               FEF1  CF          02  FB 0010A        CALLS   #2, DELETE_LINE                         :
                                 54  D5 0010F 10$:   TSTL    PRV_SCRPTR                              : 2543
                                 62  13 00111        BEQL    15$                                     :
                                 7E  D4 00113        CLRL    -(SP)                                   :
                 7E       FF     8F  9A 00115        MOVZBL  #255, -(SP)                             :
                                 7E  D4 00119        CLRL    -(SP)                                   :
                                 54  DD 0011B        PUSHL   PRV_SCRPTR                              :
                     00000000G   00  DD 0011D        PUSHL   EDT$$A_TOP_SCRPTR                       :
                                 49  11 00123        BRB     14$                                     :
                                 7E  7C 00125 11$:   CLRQ    -(SP)                                   : 2551
                 6A              02  FB 00127        CALLS   #2, EDT$$SC_POSCSIF                     :
                 01  00000000G   00  D1 0012A        CMPL    EDT$$G_TI_TYP, #1                       : 2553
                                 08  12 00131        BNEQ    12$                                     :
                                 02  DD 00133        PUSHL   #2                                      : 2555
                        FEB6     CF  9F 00135        PUSHAB  P.AAF                                   :
                                 06  11 00139        BRB     13$                                     :
                                 02  DD 0013B 12$:   PUSHL   #2                                      : 2557
                        FEB2     CF  9F 0013D        PUSHAB  P.AAG                                   :
          00000000G  00          02  FB 00141 13$:   CALLS   #2, EDT$$FMT_LIT                        :
                     00000000G   00  D4 00148        CLRL    EDT$$G_MSGFLG                           : 2567
                 53              69  D0 0014E        MOVL    EDT$$G_CS_LNO, SAV_CS_LN                : 2568
                 69              6B  D0 00151        MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO           : 2569
          00000000G  00          00  FB 00154        CALLS   #0, EDT$$SC_ERAALL                      : 2570
```

```
                                69              53  D0 0015B          MOVL    SAV_CS_LN, EDT$$G_CS_LNO                ; 2571
                                                01  DD 0015E          PUSHL   #1                                      ; 2576
                                7E        FF    8F  9A 00160          MOVZBL  #255, -(SP)
                                                7E  D4 00164          CLRL    -(SP)
                                   00000000G    00  DD 00166          PUSHL   EDT$$A_LST_SCRPTR
                                                52  DD 0016C          PUSHL   R2
                          00000000G  00                 05  FB 0016E 14$:    CALLS   #5, EDT$$SC_REPAINT
                                                   00EF 31  00175 15$:    BRW     23$                                     ; 2467
                                07                 57  E9 00178 16$:    BLBC    R7, 17$                                 ; 2593
                          00000000G  00            55  D0 0017B          MOVL    NXT_SCRPTR, EDT$$A_TOP_SCRPTR
                                59                 50  E9 00182 17$:    BLBC    R0, 20$                                 ; 2595
                     50           6B              01  C3 00185          SUBL3   #1, EDT$$G_SCR_LNS, R0
                                50                 69  D1 00189          CMPL    EDT$$G_CS_LNO, R0
                                                   50  18 0018C          BGEQ    20$
                                50                 6B  D0 0018E          MOVL    EDT$$G_SCR_LNS, R0                      ; 2609
                                1A 00000000G       00  E9 00191          BLBC    EDT$$G_TI_EDIT, 18$                     ; 2603
                                                   50  DD 00198          PUSHL   R0                                      ; 2609
                                                   7E  D4 0019A          CLRL    -(SP)
                          00000000G  00            02  FB 0019C          CALLS   #2, EDT$$SC_SETSCLLREG
                                                   7E  D4 001A3          CLRL    -(SP)                                   ; 2610
                                                   69  DD 001A5          PUSHL   EDT$$G_CS_LNO
                                6A                 02  FB 001A7          CALLS   #2, EDT$$SC_POSCSIF
                                                   03  DD 001AA          PUSHL   #3                                      ; 2611
                                      FE47         CF  9F 001AC          PUSHAB  P.AAH
                                                   1A  11 001B0          BRB     19$
                                                   50  DD 001B2 18$:    PUSHL   R0                                      ; 2618
                                                   69  DD 001B4          PUSHL   EDT$$G_CS_LNO
                          00000000G  00            02  FB 001B6          CALLS   #2, EDT$$SC_SETSCLLREG
                                                   7E  D4 001BD          CLRL    -(SP)                                   ; 2619
                     7E           6B              01  C3 001BF          SUBL3   #1, EDT$$G_SCR_LNS, -(SP)
                                6A                 02  FB 001C3          CALLS   #2, EDT$$SC_POSCSIF
                                                   01  DD 001C6          PUSHL   #1                                      ; 2620
                                      FE2F         CF  9F 001C8          PUSHAB  P.AAI
                          00000000G  00            02  FB 001CC 19$:    CALLS   #2, EDT$$FMT_LIT
                                                   52  DD 001D3          PUSHL   R2                                      ; 2626
                          00000000G  00            01  FB 001D5          CALLS   #1, EDT$$SC_LNDEL
                                                   6F  11 001DC          BRB     22$                                     ; 2595
                     50 00000000G  00              02  C7 001DE 20$:    DIVL3   #2, EDT$$G_BOT_LINE, R0                 ; 2636
                                50                 69  D1 001E6          CMPL    EDT$$G_CS_LNO, R0
                                                   4B  14 001E9          BGTR    21$
                                                   01  DD 001EB          PUSHL   #1                                      ; 2647
                                7E        FF       8F  9A 001ED          MOVZBL  #255, -(SP)
                                                   7E  D4 001F1          CLRL    -(SP)
                                                   52  DD 001F3          PUSHL   R2
                                                   53  DD 001F5          PUSHL   TOP_SCRPTR
                          00000000G  00            05  FB 001F7          CALLS   #5, EDT$$SC_REPAINT
                                53                 69  D0 001FE          MOVL    EDT$$G_CS_LNO, SAV_CS_LN                ; 2648
                                                   7E  D4 00201          CLRL    -(SP)                                   ; 2649
                     7E 00000000G  00              01  C1 00203          ADDL3   #1, EDT$$G_MESSAGE_LINE, -(SP)
                                6A                 02  FB 0020B          CALLS   #2, EDT$$SC_POSCSIF
                                                   01  DD 0020E          PUSHL   #1                                      ; 2650
                                      FDEB         CF  9F 00210          PUSHAB  P.AAJ
                          00000000G  00            02  FB 00214          CALLS   #2, EDT$$FMT_LIT
                                   00000000G       00  D4 0021B          CLRL    EDT$$G_MSGFLG                           ; 2654
                                   00000000G       00  D6 00221          INCL    EDT$$G_BOT_LINE                         ; 2655
                                69                 6B  D0 00227          MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO           ; 2656
                          00000000G  00            00  FB 0022A          CALLS   #0, EDT$$SC_ERAALL                      ; 2657
```

```
                                69              53  D0 00231          MOVL    SAV_CS_LN, EDT$$G_CS_LNO                ; 2658
                                                17  11 00234          BRB     22$                                     ; 2636
                                                01  DD 00236 21$:     PUSHL   #1                                      ; 2665
                                7E      FF      8F  9A 00238          MOVZBL  #255, -(SP)                             ;
                                                7E  D4 0023C          CLRL    -(SP)                                   ;
                                    00000000G   00  DD 0023E          PUSHL   EDT$$A_LST_SCRPTR                       ;
                                                52  DD 00244          PUSHL   R2                                      ;
                        00000000G   00          05  FB 00246          CALLS   #5, EDT$$SC_REPAINT                     ;
                                50 00000000G    00  D0 0024D 22$:     MOVL    EDT$$A_BOT_SCRPTR, R0                   ; 2674
                                                11  13 00254          BEQL    23$                                     ;
                        00000000G   00          50  D1 00256          CMPL    R0, EDT$$A_EOB_SCRPTR                   ;
                                                08  13 0025D          BEQL    23$                                     ;
                        00000000G   00      04  A0  D0 0025F          MOVL    4(R0), EDT$$A_BOT_SCRPTR                ; 2676
                                                50  D4 00267 23$:     CLRL    R0                                      ; 2684
                                                    04 00269          RET                                             ;

; Routine Size:  618 bytes,    Routine Base:  _EDT$CODE + 0F41
```

```
;  2117          2685  1 %SBTTL 'INSERT_LINE - insert a line on the screen'
;  2118          2686  1 ROUTINE INSERT_LINE (                                  ! Insert a line on the screen
;  2119          2687  1     SCRPTR,                                            ! The line to insert
;  2120          2688  1     REC_NO,                                            ! It's relative record number
;  2121          2689  1     OLD_TOP_RECNO                                      ! Record number of the top line
;  2122          2690  1     ) =
;  2123          2691  1
;  2124          2692  1 !++
;  2125          2693  1 ! FUNCTIONAL DESCRIPTION:
;  2126          2694  1 !
;  2127          2695  1 !       Insert one screen line.  It may be ma  ed inserted or deleted.
;  2128          2696  1 !       This routine is not called unless it has some kind of edit.
;  2129          2697  1 !
;  2130          2698  1 ! FORMAL PARAMETERS:
;  2131          2699  1 !
;  2132          2700  1 !  SCRPTR               The screen data block to insert
;  2133          2701  1 !
;  2134          2702  1 !  REC_NO               The relative record number of that line
;  2135          2703  1 !
;  2136          2704  1 !  OLD_TOP_RECNO        Record number of the top line
;  2137          2705  1 !
;  2138          2706  1 ! IMPLICIT INPUTS:
;  2139          2707  1 !
;  2140          2708  1 !       EDT$$G_TI_SCROLL
;  2141          2709  1 !       EDT$$A_TOP_SCRPTR
;  2142          2710  1 !       EDT$$G_SCR_LNS
;  2143          2711  1 !       EDT$$G_SCLL_BOT
;  2144          2712  1 !       EDT$$G_SCLL_TOP
;  2145          2713  1 !       EDT$$G_BOT_SCRPTR
;  2146          2714  1 !       EDT$$G_CS_RNO
;  2147          2715  1 !       EDT$$A_LST_SCRPTR
;  2148          2716  1 !       EDT$$G_BOT_LINE
;  2149          2717  1 !       EDT$$G_MSGFLG
;  2150          2718  1 !       EDT$$G_TI_TYP
;  2151          2719  1 !
;  2152          2720  1 ! IMPLICIT OUTPUTS:
;  2153          2721  1 !
;  2154          2722  1 !       EDT$$A_TOP_SCRPTR
;  2155          2723  1 !       EDT$$A_BOT_SCRPTR
;  2156          2724  1 !       EDT$$G_BOT_LINE
;  2157          2725  1 !       EDT$$G_MSGFLG
;  2158          2726  1 !
;  2159          2727  1 ! ROUTINE VALUE:
;  2160          2728  1 !
;  2161          2729  1 !       1 = repaint this line and continue this pass, 0 = must start update over
;  2162          2730  1 !
;  2163          2731  1 ! SIDE EFFECTS:
;  2164          2732  1 !
;  2165          2733  1 !       Will store into the format buffer
;  2166          2734  1 !
;  2167          2735  1 !--
;  2168          2736  1
;  2169          2737  2     BEGIN
;  2170          2738  2
;  2171          2739  2     MAP
;  2172          2740  2         SCRPTR : REF SCREEN_LINE;
;  2173          2741  2
```

```
;2174    2742  2      EXTERNAL ROUTINE
;2175    2743  2          EDT$$SC_LNDEL,                          ! Free a deleted line
;2176    2744  2          EDT$$SC_SETSCLLREG,                     ! Set scrolling region
;2177    2745  2          EDT$$FMT_LIT,                           ! Format a literal for output
;2178    2746  2          EDT$$SC_POSCSIF : NOVALUE,              ! Position the cursor
;2179    2747  2          EDT$$SC_ERATOEOL : NOVALUE,             ! Erase to end of line
;2180    2748  2          EDT$$SC_MOVTOLN,                ! Move to a record in the work file relative to the current record
;2181    2749  2          EDT$$SC_ERAALL : NOVALUE,               ! Erase part of the screen
;2182    2750  2          EDT$$SC_REPAINT : NOVALUE;              ! Mark some lines in the screen data base for repaint
;2183    2751  2
;2184    2752  2      EXTERNAL
;2185    2753  2          EDT$$G_MESSAGE_LINE,                    ! Line number for error msgs
;2186    2754  2          EDT$$G_TI_EDIT,                         ! VT102 editing features
;2187    2755  2          EDT$$G_SCR_LNS,                         ! Number of lines on screen
;2188    2756  2          EDT$$G_SCLL_TOP,                        ! Top line the cursor is allowed to stay on
;2189    2757  2          EDT$$G_SCLL_BOT,                        ! Bottom line the cursor is allowed to stay on
;2190    2758  2          EDT$$A_BOT_SCRPTR : REF SCREEN_LINE,    ! Bottom screen pointer
;2191    2759  2          EDT$$A_LST_SCRPTR : REF SCREEN_LINE,    ! Last data structure pointer
;2192    2760  2          EDT$$A_TOP_SCRPTR : REF SCREEN_LINE,    ! Top screen line info address
;2193    2761  2          EDT$$A_EOB_SCRPTR : REF SCREEN_LINE,    ! EOB screen pointer
;2194    2762  2          EDT$$G_CS_LNO,                          ! current cursor line
;2195    2763  2          EDT$$G_TI_SCROLL,                       ! 1 = we have scrolling regions
;2196    2764  2          EDT$$G_TI_TYP,                          ! Terminal type
;2197    2765  2          EDT$$G_BOT_LINE,                        ! All lines below this one have been erased
;2198    2766  2          EDT$$G_MSGFLG;                          ! 1 = there is a message on the screen
;2199    2767  2
;2200    2768  2      LOCAL
;2201    2769  2          NXT_SCRPTR : REF SCREEN_LINE,           ! Next screen pointer
;2202    2770  2          SCLL_CENTER;                            ! The center line of the screen
;2203    2771  2
;2204    2772  2  !+
;2205    2773  2  ! There must always be a line after an inserted line, since the [EOB] is never inserted.
;2206    2774  2  !-
;2207    2775  2      NXT_SCRPTR = .SCRPTR [SCR_NXT_LINE];
;2208    2776  2      ASSERT (.NXT_SCRPTR NEQA 0);
;2209    2777  2
;2210    2778  3      IF ((.NXT_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)
;2211    2779  2      THEN
;2212    2780  3          BEGIN
;2213    2781  3  !+
;2214    2782  3  ! The next line is deleted.  Combine an inserted line and a deleted line by
;2215    2783  3  ! just repainting the inserted line and ignoring the deleted line.
;2216    2784  3  !-
;2217    2785  3          EDT$$SC_REPAINT (.SCRPTR, .NXT_SCRPTR, 0, 255, 1);
;2218    2786  3  !+
;2219    2787  3  ! Erase the deleted line so we can just paint the inserted line over it as though
;2220    2788  3  ! we had opened a line for the inserted text.
;2221    2789  3  !-
;2222    2790  3          EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
;2223    2791  3          EDT$$SC_ERATOEOL ();
;2224    2792  3          RETURN (1);
;2225    2793  2          END;
;2226    2794  2
;2227    2795  2  !+
;2228    2796  2  ! Decide whether we will be scrolling up or down.  If we scroll up then the lines above
;2229    2797  2  ! the inserted line are moved up to make room for the inserted line, and the top line
;2230    2798  2  ! is removed from the screen.  Lines below the inserted line are not changed.  If we
```

```
;  2231          2799  2 ! scroll down then the lines below the inserted line are moved down to make room for
;  2232          2800  2 ! the inserted line, and the bottom line, if there is one, is removed from the screen.
;  2233          2801  2 !-
;  2234          2802  2     SCLL_CENTER = (.EDT$$G_SLLL_TOP + .EDT$$G_SCLL_BOT)/2;
;  2235          2803  2
;  2236          2804  3     IF ((.EDT$$G_CS_LNO GEQ .SCLL_CENTER) AND (.EDT$$G_BOT_LINE GEQ .EDT$$G_SCR_LNS))
;  2237          2805  2     THEN
;  2238          2806  3         BEGIN
;  2239          2807  3
;  2240          2808  3         LOCAL
;  2241          2809  3             RET_VALUE;
;  2242          2810  3
;  2243          2811  3 !+
;  2244          2812  3 ! The inserted line is low on the screen and the screen is full.  Move a new line onto
;  2245          2813  3 ! the bottom of the screen.  This will require scrolling the top of the screen up one
;  2246          2814  3 ! and having a new top line.
;  2247          2815  3 !-
;  2248          2816  3
;  2249          2817  3         IF .EDT$$G_TI_SCROLL
;  2250          2818  3         THEN
;  2251          2819  4             BEGIN
;  2252          2820  4 !+
;  2253          2821  4 ! We have scrolling regions, so we can avoid repainting the bottom of the screen.
;  2254          2822  4 !-
;  2255          2823  4             EDT$$G_CS_LNO = .EDT$$G_CS_LNO - 1;
;  2256          2824  4             EDT$$SC_SETSCLLREG (0, .EDT$$G_CS_LNO + 1);
;  2257          2825  4             EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
;  2258          2826  4             EDT$$FMT_LIT (UPLIT (%STRING (%CHAR (ASC_K_LF))), 1);       ! Scroll up
;  2259          2827  4             RET_VALUE = 1;                              ! Paint this line and continue
;  2260          2828  4             END
;  2261          2829  3         ELSE
;  2262          2830  4             BEGIN
;  2263          2831  4 !+
;  2264          2832  4 ! We do not have scrolling regions, we must repaint the bottom of the screen.
;  2265          2833  4 ! Make sure the message area is blank, since otherwise we may scroll a message
;  2266          2834  4 ! into the text area.
;  2267          2835  4 !-
;  2268          2836  5             BEGIN
;  2269          2837  5
;  2270          2838  5             LOCAL
;  2271          2839  5                 SAV_CS_LN;
;  2272          2840  5
;  2273          2841  5             EDT$$G_MSGFLG = 0;
;  2274          2842  5             SAV_CS_LN = .EDT$$G_CS_LNO;
;  2275          2843  5             EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;  2276          2844  5             EDT$$SC_ERAALL ();
;  2277          2845  5             EDT$$G_CS_LNO = .SAV_CS_LN;
;  2278          2846  4             END;
;  2279          2847  4 !+
;  2280          2848  4 ! Now scroll the whole screen up by one.
;  2281          2849  4 !-
;  2282          2850  4             EDT$$SC_POSCSIF (.EDT$$G_MESSAGE_LINE + 1, 0);
;  2283          2851  4             EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_LF)), 1);
;  2284          2852  4 !+
;  2285          2853  4 ! Arrange to repaint all lines below the deleted line.
;  2286          2854  4 !-
;  2287          2855  4             EDT$$SC_REPAINT (.SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
```

```
; 2288          2856  4               RET_VALUE = 0;                          ! Start a new pass
; 2289          2857  3               END;
; 2290          2858  3
; 2291          2859  3           !+
; 2292          2860  3           ! Mark the top line for repaint since it is moving off the screen.
; 2293          2861  3           !-
; 2294          2862  3                   EDT$$SC_REPAINT (.EDT$$A_TOP_SCRPTR, .EDT$$A_TOP_SCRPTR, 0, 255, -1);
; 2295          2863  3                   EDT$$A_TOP_SCRPTR = .EDT$$A_TOP_SCRPTR [SCR_NXT_LINE];
; 2296          2864  3           !+
; 2297          2865  3           ! Adjust the record number of the top line.
; 2298          2866  3           !-
; 2299          2867  3
; 2300          2868  4                   IF ((.EDT$$A_TOP_SCRPTR [SCR_LINE_IDX] EQL 0) AND       !
; 2301          2869  4                       ((.EDT$$A_TOP_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0))
; 2302          2870  3                   THEN
; 2303          2871  3                       .OLD_TOP_RECNO = ..OLD_TOP_RECNO + 1;
; 2304          2872  3
; 2305          2873  3                   RETURN (.RET_VALUE);
; 2306          2874  3                   END
; 2307          2875  2               ELSE
; 2308          2876  3                   BEGIN
; 2309          2877  3
; 2310          2878  3                   LOCAL
; 2311          2879  3                       RET_VALUE;
; 2312          2880  3
; 2313          2881  3           !+
; 2314          2882  3           ! The inserted line is high on the screen, or the screen is not full.
; 2315          2883  3           ! Scroll the bottom of the screen down.
; 2316          2884  3           !-
; 2317          2885  3
; 2318          2886  3                   IF .EDT$$G_TI_SCROLL
; 2319          2887  3                   THEN
; 2320          2888  4                       BEGIN
; 2321          2889  4           !+
; 2322          2890  4           ! We have scrolling regions, so we do not have to repaint the top of the screen.
; 2323          2891  4           !-
; 2324          2892  4
; 2325          2893  5                       IF (.EDT$$G_TI_EDIT)
; 2326          2894  4                       THEN
; 2327          2895  4           !+
; 2328          2896  4           ! Use VT102 edit feature
; 2329          2897  4           !-
; 2330          2898  5                           BEGIN
; 2331          2899  5                           EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
; 2332          2900  5                           EDT$$FMT_LIT (UPLIT (%STRING (%CHAR (ASC_K_ESC), '[L')), 3);
; 2333          2901  5                           END
; 2334          2902  4                       ELSE
; 2335          2903  5                           BEGIN
; 2336          2904  5           !+
; 2337          2905  5           ! Simulate edit feature
; 2338          2906  5           !-
; 2339          2907  5                           EDT$$SC_SETSCLLREG (.EDT$$G_CS_LNO, .EDT$$G_SCR_LNS);
; 2340          2908  5                           EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
; 2341          2909  5                           EDT$$FMT_LIT (UPLIT (%STRING (%CHAR (ASC_K_ESC), 'M')), 2);     ! Scroll down
; 2342          2910  4                           END;
; 2343          2911  4
; 2344          2912  4                   RET_VALUE = 1;                      ! Repaint this line and continue
```

```
;  2345          2913  4              END
;  2346          2914  3          ELSE
;  2347          2915  4              BEGIN
;  2348          2916  4          !+
;  2349          2917  4          ! This is a terminal without scrolling regions.  If the inserted line is high on the screen
;  2350          2918  4          ! scroll down then repaint from the beginning of the screen to this point.  If the inserted
;  2351          2919  4          ! line is low on the screen just repaint from here to the bottom of the screen.
;  2352          2920  4          !-
;  2353          2921  4
;  2354          2922  5              IF (.EDT$$G_CS_LNO LSS (.EDT$$G_BOT_LINE/2))
;  2355          2923  4              THEN
;  2356          2924  5                  BEGIN
;  2357          2925  5          !+
;  2358          2926  5          ! The inserted line is high on the screen.  Scroll down and repaint the part of the screen
;  2359          2927  5          ! above the inserted line.
;  2360          2928  5          !-
;  2361          2929  5                  EDT$$SC_POSCSIF (0, 0);
;  2362          2930  5
;  2363          2931  6                  IF (.EDT$$G_TI_TYP EQL TERM_VT52)       !
;  2364          2932  5                  THEN
;  2365          2933  5                      EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'I')), 2)
;  2366          2934  5                  ELSE
;  2367          2935  5                      EDT$$FMT_LIT (UPLIT (BYTE (ASC_K_ESC, %C'M')), 2);
;  2368          2936  5
;  2369          2937  5          !+
;  2370          2938  5          ! Make sure the message area is blank,
;  2371          2939  5          ! since we may have moved text into the message area.
;  2372          2940  5          !-
;  2373          2941  6                  BEGIN
;  2374          2942  6
;  2375          2943  6                  LOCAL
;  2376          2944  6                      SAV_CS_LN;
;  2377          2945  6
;  2378          2946  6                  EDT$$G_MSGFLG = 0;
;  2379          2947  6                  SAV_CS_LN = .EDT$$G_CS_LNO;
;  2380          2948  6                  EDT$$G_CS_LNO = .EDT$$G_SCR_LNS;
;  2381          2949  6                  EDT$$SC_ERAALL ();
;  2382          2950  6                  EDT$$G_CS_LNO = .SAV_CS_LN;
;  2383          2951  5                  END;
;  2384          2952  5          !+
;  2385          2953  5          ! Now mark the top of the screen to be repainted.
;  2386          2954  5          !-
;  2387          2955  5                  EDT$$SC_REPAINT (.EDT$$A_TOP_SCRPTR, .SCRPTR, 0, 255, 1);
;  2388          2956  5                  RET_VALUE = 0;                  ! Start a new update pass
;  2389          2957  5                  END
;  2390          2958  4              ELSE
;  2391          2959  5                  BEGIN
;  2392          2960  5          !+
;  2393          2961  5          ! The inserted line is low on the screen.  Just repaint the inserted line and all lower lines.
;  2394          2962  5          !-
;  2395          2963  5                  EDT$$SC_REPAINT (.SCRPTR, .EDT$$A_LST_SCRPTR, 0, 255, 1);
;  2396          2964  5                  RET_VALUE = 0;                  ! Start a new pass
;  2397          2965  4                  END;
;  2398          2966  4
;  2399          2967  3              END;
;  2400          2968  3
;  2401          2969  3          !+
```

```
3 ! If the bottom line will move off the screen, arrange to repaint it if it should move back on.
3 !-
3
4         IF ((.EDT$$A_BOT_SCRPTR NEQA 0) AND (.EDT$$G_BOT_LINE GEQ (.EDT$$G_SCR_LNS - 1)))
3         THEN
4             BEGIN
4             EDT$$SC_REPAINT (.EDT$$A_BOT_SCRPTR, .EDT$$A_BOT_SCRPTR, 0, 255, 0);
4             EDT$$A_BOT_SCRPTR = .EDT$$A_BOT_SCRPTR [SCR_PRV_LINE];
3             END;
3
3 !+
3 ! The bottom line may be lower on the screen.  It doesn't matter much if EDT$$G_BOT_LINE is too large.
3 !-
3         EDT$$G_BOT_LINE = .EDT$$G_BOT_LINE + 1;
3         RETURN (.RET_VALUE);
2         END;
2
1     END;                                         ! of routine INSERT_LINE


                                   011AB          .BLKB    1
            00  00  00  0A  011AC P.AAK:   .ASCII   <10><0><0><0>                        ;
                        0A  011B0 P.AAL:   .BYTE    10                                   ;
                            011B1          .BLKB    3
            00  4C  5B  1B  011B4 P.AAM:   .ASCII   <27>\[L\<0>                          ;
            00  00  4D  1B  011B8 P.AAN:   .ASCII   <27>\M\<0><0>                        ;
                    49  1B  011BC P.AAO:   .BYTE    27, 73                               ;
                            011BE          .BLKB    2
                    4D  1B  011C0 P.AAP:   .BYTE    27, 77                               ;

                                           .EXTRN   EDT$$SC_ERATOEOL

                         0FFC 00000 INSERT_LINE:
                                           .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11     ; 2686
           5B 00000000G  00  9E 00002      MOVAB    EDT$$FMT_LIT, R11                    ;
           5A 00000000G  00  9E 00009      MOVAB    EDT$$G_BOT_LINE, R10                 ;
           59 00000000G  00  9E 00010      MOVAB    EDT$$A_TOP_SCRPTR, R9                ;
           58 00000000G  00  9E 00017      MOVAB    EDT$$G_SCR_LNS, R8                   ;
           57 00000000G  00  9E 0001E      MOVAB    EDT$$SC_REPAINT, R7                  ;
           56 00000000G  00  9E 00025      MOVAB    EDT$$SC_POSCSIF, R6                  ;
           55 00000000G  00  9E 0002C      MOVAB    EDT$$G_CS_LNO, R5                    ;
           54         04  AC  D0 00033      MOVL     SCRPTR, R4                           ; 2775
           52         04  A4  D0 00037      MOVL     4(R4), NXT_SCRPTR                    ;
                          07  12 0003B      BNEQ     1$                                   ; 2776
  00000000G  00               00  FB 0003D      CALLS    #0, EDT$$INTER_ERR                   ;
21        0D  A2               02  E1 00044 1$: BBC      #2, 13(NXT_SCRPTR), 2$               ; 2778
                          01  DD 00049      PUSHL    #1                                   ; 2785
           7E         FF  8F  9A 0004B      MOVZBL   #255, -(SP)                          ;
                          7E  D4 0004F      CLRL     -(SP)                                ;
                          52  DD 00051      PUSHL    NXT_SCRPTR                           ;
                          54  DD 00053      PUSHL    R4                                   ;
           67             05  FB 00055      CALLS    #5, EDT$$SC_REPAINT                  ;
                          7E  D4 00058      CLRL     -(SP)                                ; 2790
                          65  DD 0005A      PUSHL    EDT$$G_CS_LNO                        ;
           66             02  FB 0005C      CALLS    #2, EDT$$SC_POSCSIF                  ;
  00000000G  00               00  FB 0005F      CALLS    #0, EDT$$SC_ERATOEOL                 ; 2791
```

```
                    50              01 D0 00066         MOVL    #1, R0                                      ; 2792
                                    04 00069            RET                                                 ;
      50 00000000G  00 00000000G    00 C1 0006A  2$:    ADDL3   EDT$$G_SCLL_BOT, EDT$$G_SCLL_TOP, R0        ; 2802
                    50              02 C6 00076         DIVL2   #2, SCLL_CENTER                             ;
                    51 00000000G    00 D0 00079         MOVL    EDT$$G_TI_SCROLL, R1                        ; 2817
                    53              65 D0 00080         MOVL    EDT$$G_CS_LNO, R3                           ; 2804
                    50              53 D1 00083         CMPL    R3, SCLL_CENTER                             ;
                                    03 19 00086         BLSS    3$                                          ;
                    68              6A D1 00088         CMPL    EDT$$G_BOT_LINE, EDT$$G_SCR_LNS             ;
                                    03 18 0008B  3$:    BGEQ    4$                                          ;
                                  0095 31 0008D         BRW     8$                                          ;
                    24              51 E9 00090  4$:    BLBC    R1, 5$                                      ; 2817
                                    65 D7 00093         DECL    EDT$$G_CS_LNO                               ; 2823
      7E            65              01 C1 00095         ADDL3   #1, EDT$$G_CS_LNO, -(SP)                    ; 2824
                                    7E D4 00099         CLRL    -(SP)                                       ;
         00000000G  00              02 FB 0009B         CALLS   #2, EDT$$SC_SETSCLLREG                      ;
                                    7E D4 000A2         CLRL    -(SP)                                       ; 2825
                                    65 DD 000A4         PUSHL   EDT$$G_CS_LNO                               ;
                    66              02 FB 000A6         CALLS   #2, EDT$$SC_POSCSIF                         ;
                                    01 DD 000A9         PUSHL   #1                                          ; 2826
                          FF3B      CF 9F 000AB         PUSHAB  P.AAK                                       ;
                    6B              02 FB 000AF         CALLS   #2, EDT$$FMT_LIT                            ;
                    52              01 D0 000B2         MOVL    #1, RET_VALUE                               ; 2827
                                    41 11 000B5         BRB     6$                                          ; 2817
                       00000000G    00 D4 000B7  5$:    CLRL    EDT$$G_MSGFLG                               ; 2841
                    52              53 D0 000BD         MOVL    R3, SAV_CS_LN                               ; 2842
                    65              68 D0 000C0         MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO               ; 2843
         00000000G  00              00 FB 000C3         CALLS   #0, EDT$$SC_ERAALL                          ; 2844
                    65              52 D0 000CA         MOVL    SAV_CS_LN, EDT$$G_CS_LNO                    ; 2845
                                    7E D4 000CD         CLRL    -(SP)                                       ; 2850
      7E 00000000G  00              01 C1 000CF         ADDL3   #1, EDT$$G_MESSAGE_LINE, -(SP)              ;
                    66              02 FB 000D7         CALLS   #2, EDT$$SC_POSCSIF                         ;
                                    01 DD 000DA         PUSHL   #1                                          ; 2851
                          FF0E      CF 9F 000DC         PUSHAB  P.AAL                                       ;
                    6B              02 FB 000E0         CALLS   #2, EDT$$FMT_LIT                            ;
                                    01 DD 000E3         PUSHL   #1                                          ; 2855
                    7E          FF  8F 9A 000E5         MOVZBL  #255, -(SP)                                 ;
                                    7E D4 000E9         CLRL    -(SP)                                       ;
                       00000000G    00 DD 000EB         PUSHL   EDT$$A_LST_SCRPTR                           ;
                                    54 DD 000F1         PUSHL   R4                                          ;
                    67              05 FB 000F3         CALLS   #5, EDT$$SC_REPAINT                         ;
                                    52 D4 000F6         CLRL    RET_VALUE                                   ; 2856
                    7E              01 CE 000F8  6$:    MNEGL   #1, -(SP)                                   ; 2862
                    7E          FF  8F 9A 000FB         MOVZBL  #255, -(SP)                                 ;
                                    7E D4 000FF         CLRL    -(SP)                                       ;
                    50              69 D0 00101         MOVL    EDT$$A_TOP_SCRPTR, R0                       ;
                                    50 DD 00104         PUSHL   R0                                          ;
                                    50 DD 00106         PUSHL   R0                                          ;
                    67              05 FB 00108         CALLS   #5, EDT$$SC_REPAINT                         ;
                    50              69 D0 0010B         MOVL    EDT$$A_TOP_SCRPTR, R0                       ; 2863
                    69          04  A0 D0 0010E         MOVL    4(R0), EDT$$A_TOP_SCRPTR                    ;
                    50              69 D0 00112         MOVL    EDT$$A_TOP_SCRPTR, R0                       ; 2868
                                08  A0 95 00115         TSTB    8(R0)                                       ;
                                    08 12 00118         BNEQ    7$                                          ;
      03        0D  A0              02 E0 0011A         BBS     #2, 13(R0), 7$                              ; 2869
                                0C  BC D6 0011F         INCL    @OLD_TOP_RECNO                              ; 2871
                                  00CB 31 00122  7$:    BRW     18$                                         ; 2876
```

```
                  36            51  E9 00125 8$:   BLBC    R1, 11$                              ; 2886
                  0F 00000000G  00  E9 00128       BLBC    EDT$$G_TI_EDIT, 9$                   ; 2893
                                7E  D4 0012F       CLRL    -(SP)                                ; 2899
                                53  DD 00131       PUSHL   R3                                   ;
                  66            02  FB 00133       CALLS   #2, EDT$$SC_POSCSIF                  ;
                                03  DD 00136       PUSHL   #3                                   ; 2900
                         FEB6   CF  9F 00138       PUSHAB  P.AAM                                ;
                                18  11 0013C       BRB     10$                                  ;
                                68  DD 0013E 9$:   PUSHL   EDT$$G_SCR_LNS                       ; 2907
                                53  DD 00140       PUSHL   R3                                   ;
         00000000G  00          02  FB 00142       CALLS   #2, EDT$$SC_SETSCLLREG               ;
                                7E  D4 00149       CLRL    -(SP)                                ; 2908
                                65  DD 0014B       PUSHL   EDT$$G_CS_LNO                        ;
                  66            02  FB 0014D       CALLS   #2, EDT$$SC_POSCSIF                  ;
                                02  DD 00150       PUSHL   #2                                   ; 2909
                         FEA0   CF  9F 00152       PUSHAB  P.AAN                                ;
                  6B            02  FB 00156 10$:  CALLS   #2, EDT$$FMT_LIT                     ;
                  52            01  D0 00159       MOVL    #1, RET_VALUE                        ; 2912
                                61  11 0015C       BRB     16$                                  ; 2886
                                52  D4 0015E 11$:  CLRL    RET_VALUE                            ; 2956
         50       6A            02  C7 00160       DIVL3   #2, EDT$$G_BOT_LINE, R0              ; 2922
                  50            53  D1 00164       CMPL    R3, R0                               ;
                                43  18 00167       BGEQ    14$                                  ;
                                7E  7C 00169       CLRQ    -(SP)                                ; 2929
                  66            02  FB 0016B       CALLS   #2, EDT$$SC_POSCSIF                  ;
                  01 00000000G  00  D1 0016E       CMPL    EDT$$G_TI_TYP, #1                    ; 2931
                                08  12 00175       BNEQ    12$                                  ;
                                02  DD 00177       PUSHL   #2                                   ; 2933
                         FE7D   CF  9F 00179       PUSHAB  P.AAO                                ;
                                06  11 0017D       BRB     13$                                  ;
                                02  DD 0017F 12$:  PUSHL   #2                                   ; 2935
                         FE79   CF  9F 00181       PUSHAB  P.AAP                                ;
                  6B            02  FB 00185 13$:  CALLS   #2, EDT$$FMT_LIT                     ;
                     00000000G  00  D4 00188       CLRL    EDT$$G_MSGFLG                        ; 2946
                  53            65  D0 0018E       MOVL    EDT$$G_CS_LNO, SAV_CS_LN             ; 2947
                  65            68  D0 00191       MOVL    EDT$$G_SCR_LNS, EDT$$G_CS_LNO        ; 2948
         00000000G  00          00  FB 00194       CALLS   #0, EDT$$SC_ERAALL                   ; 2949
                  65            53  D0 0019B       MOVL    SAV_CS_LN, EDT$$G_CS_LNO             ; 2950
                                01  DD 0019E       PUSHL   #1                                   ; 2955
                  7E      FF    8F  9A 001A0       MOVZBL  #255, -(SP)                          ;
                                7E  D4 001A4       CLRL    -(SP)                                ;
                                54  DD 001A6       PUSHL   R4                                   ;
                                69  DD 001A8       PUSHL   EDT$$A_TOP_SCRPTR                    ;
                                10  11 001AA       BRB     15$                                  ;
                                01  DD 001AC 14$:  PUSHL   #1                                   ; 2963
                  7E      FF    8F  9A 001AE       MOVZBL  #255, -(SP)                          ;
                                7E  D4 001B2       CLRL    -(SP)                                ;
                     00000000G  00  DD 001B4       PUSHL   EDT$$A_LST_SCRPTR                    ;
                                54  DD 001BA       PUSHL   R4                                   ;
                  67            05  FB 001BC 15$:  CALLS   #5, EDT$$SC_REPAINT                  ;
                  51 00000000G  00  D0 001BF 16$:  MOVL    EDT$$A_BOT_SCRPTR, R1                ; 2973
                                26  13 001C6       BEQL    17$                                  ;
         50       68            01  C3 001C8       SUBL3   #1, EDT$$G_SCR_LNS, R0               ;
                  50            6A  D1 001CC       CMPL    EDT$$G_BOT_LINE, R0                  ;
                                1D  19 001CF       BLSS    17$                                  ;
                                7E  D4 001D1       CLRL    -(SP)                                ; 2976
                  7E      FF    8F  9A 001D3       MOVZBL  #255, -(SP)                          ;
```

```
                                      7E  D4 001D7          CLRL    -(SP)                          ;
                                      51  DD 001D9          PUSHL   R1                             ;
                                      51  DD 001DB          PUSHL   R1                             ;
                        67            05  FB 001DD          CALLS   #5, EDT$$SC_REPAINT            ;
                        50 00000000G  00  D0 001E0          MOVL    EDT$$A_BOT_SCRPTR, R0          ; 2977
             00000000G  00            60  D0 001E7          MOVL    (R0), EDT$$A_BOT_SCRPTR        ;
                                      6A  D6 001EE 17$:     INCL    EDT$$G_BOT_LINE                ; 2983
                        50            52  D0 001F0 18$:     MOVL    R2, R0                         ; 2876
                                          04 001F3          RET                                    ; 2987

; Routine Size:  500 bytes,    Routine Base:  _EDT$CODE + 11C2
```

```
;  2421          2988  1 %SBTTL 'EDT$$LOAD_SCRUPDATE - load this module into memory'
;  2422          2989  1
;  2423          2990  1 GLOBAL ROUTINE EDT$$LOAD_SCRUPDATE                ! Load this module into memory
;  2424          2991  1     : NOVALUE =
;  2425          2992  1
;  2426          2993  1 !++
;  2427          2994  1 ! FUNCTIONAL DESCRIPTION:
;  2428          2995  1 !
;  2429          2996  1 !       This routine has no function.  It exists as an entry point so that
;  2430          2997  1 !       EDT$$FIXNOTRUNC_NOOVERLAY can call this module back into memory before
;  2431          2998  1 !       returning to it.
;  2432          2999  1 !
;  2433          3000  1 ! FORMAL PARAMETERS:
;  2434          3001  1 !
;  2435          3002  1 !       NONE
;  2436          3003  1 !
;  2437          3004  1 ! IMPLICIT INPUTS:
;  2438          3005  1 !
;  2439          3006  1 !       NONE
;  2440          3007  1 !
;  2441          3008  1 ! IMPLICIT OUTPUTS:
;  2442          3009  1 !
;  2443          3010  1 !       NONE
;  2444          3011  1 !
;  2445          3012  1 ! ROUTINE VALUE:
;  2446          3013  1 !
;  2447          3014  1 !       NONE
;  2448          3015  1 !
;  2449          3016  1 ! SIDE EFFECTS:
;  2450          3017  1 !
;  2451          3018  1 !       NONE
;  2452          3019  1 !
;  2453          3020  1 !--
;  2454          3021  1
;  2455          3022  2     BEGIN
;  2456          3023  2     0
;  2457          3024  1     END;                                    ! of routine EDT$$LOAD_SCRUPDATE



                                                0000 00000          .ENTRY  EDT$$LOAD_SCRUPDATE, Save nothing       ; 2990
                                                  04 00002          RET                                             ; 3024

; Routine Size:  3 bytes,    Routine Base:  _EDT$CODE + 13B6

;  2458          3025  1
;  2459          3026  1 !<BLF/PAGE>
```

```
; 2461          3027  1 END                                           ! of module EDT$SCRUPDATE
; 2462          3028  1
; 2463          3029  0 ELUDOM



;                                          PSECT SUMMARY
;
;       Name                          Bytes                          Attributes
;
;   _EDT$CODE                             5049  NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)


;                               Library Statistics
;
;                                            -------- Symbols --------        Pages        Processing
;       File                                 Total    Loaded   Percent        Mapped       Time
;
;   _$255$DUA28:[EDT.SRC]EDT.L32;1             377        61        16           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$:SCRUPDATE/OBJ=OBJ$:SCRUPDATE MSRC$:SCRUPDATE.BLI/UPDATE=(ENH$:S
;       CRUPDATE)

; Size:          4988 code + 61 data bytes
; Run Time:         03:20.3
; Elapsed Time:     04:44.2
; Lines/CPU Min:      907
; Lexemes/CPU-Min:  5067
; Memory Used:  896 pages
; Compilation Complete
```

SCRESET LIS

SCRLIN 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