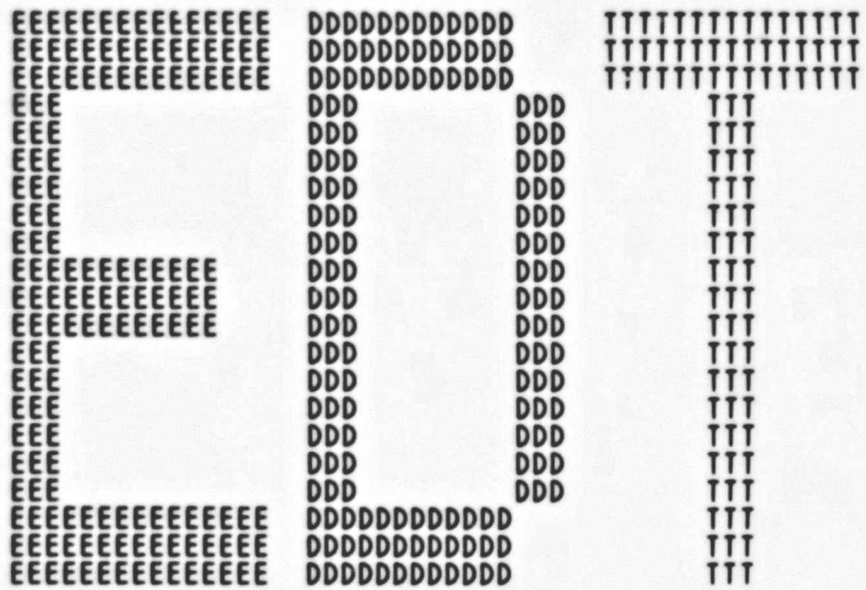
EEEEEEEEEEEEEEE DDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD 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

```
  SSSSSSSS    CCCCCCCC  RRRRRRRR    RRRRRRRR    LL          IIIIII    NN      NN
  SSSSSSSS    CCCCCCCC  RRRRRRRR    RRRRRRRR    LL          IIIIII    NN      NN
SS            CC        RR      RR  RR      RR  LL            II      NN      NN
SS            CC        RR      RR  RR      RR  LL            II      NN      NN
SS            CC        RR      RR  RR      RR  LL            II      NNNN    NN
SS            CC        RR      RR  RR      RR  LL            II      NNNN    NN
  SSSSSS      CC        RRRRRRRR    RRRRRRRR    LL            II      NN  NN  NN
  SSSSSS      CC        RRRRRRRR    RRRRRRRR    LL            II      NN  NN  NN
        SS    CC        RR  RR      RR  RR      LL            II      NN    NNNN
        SS    CC        RR  RR      RR  RR      LL            II      NN    NNNN
        SS    CC        RR    RR    RR    RR    LL            II      NN      NN    ....
        SS    CC        RR    RR    RR    RR    LL            II      NN      NN    ....
SSSSSSSS      CCCCCCCC  RR      RR  RR      RR  LLLLLLLLLL  IIIIII    NN      NN    ....
SSSSSSSS      CCCCCCCC  RR      RR  RR      RR  LLLLLLLLLL  IIIIII    NN      NN    ....

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
```

```
;       1     0001  0 %TITLE 'EDT$SCRRLIN - refresh a screen line'
;       2     0002  0 MODULE EDT$SCRRLIN (                          ! Refresh a screen line
;       3     0003  0                     IDENT = 'V04-000'                 ! File: SCRRLIN.BLI Edit: REM1034
;       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 refreshes a single line on the screen.
;      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_RFRELN  from module SCREEN.
;      46     0046  1 ! 1-002 - Regularize headers.  JBS 13-Mar-1981
;      47     0047  1 ! 1-003 - Change [EOB] to user defined string STS 06-Oct-1981
;      48     0048  1 ! 1-004 - Do an absolute cursor position before writing the blob at
;      49     0049  1 !           end of line, to avoid running off the edge of the screen.
;      50     0050  1 !           Also, show the blob only if the text exceeds the screen
;      51     0051  1 !           width.  JBS 02-Apr-1982
;      52     0052  1 ! 1-005 - Show characters all the way to end edge of the screen.  JBS 06-Apr-1982
;      53     0053  1 ! 1-006 - Worry about wide characters at the edge of the screen.  JBS 15-Apr-1982
;      54     0054  1 ! 1-007 - Continue work on edit 1-006.  JBS 16-Apr-1982
;      55     0055  1 ! 1-008 - Always show [EOB] (or whatever text it has been set to) in non-reverse
;      56     0056  1 !           video.  JBS 16-Apr-1982
;      57     0057  1 ! 1-009 - Make the edge of the screen logic work on a VT100, which clears its
```

```
;    58          0058  1 !                 wrap flag only when a character is printed.  JBS 19-Apr-1982
;    59          0059  1 ! 1-010 - Don't erase the message lines if an error occurs during select.
;    60          0060  1 !                 SMB 01-Jul-1982
;    61          0061  1 ! 1-011 - Fix bug introduced by edit 1-010.  SMB 20-Jul-1982
;    62          0062  1 ! 1-012 - Add check for message flag to erasure of screen.  SMB 23-Jul-1982
;    63          0063  1 ! 1-013 - Change the flag checked in edit 1-012.  SMB 28-Jul-1982
;    64          0064  1 ! 1-014 - Go back to edit 1-012.  SMB 17-Aug-1982
;    65          0065  1 ! 1-015 - Modify fo the new screen updater.  SMB 24-Sep-1982
;    66          0066  1 ! 1-016 - Simplify for the new screen update logic.  This version always repaints
;    67          0067  1 !                 any changed line.  JBS 30-Sep-1982
;    68          0068  1 ! 1-017 - Remove unused external declaration of EDT$$FMT_LIT.  JBS 05-Oct-1982
;    69          0069  1 ! 1-018 - Fix painting of select range.  JBS 08-Oct-1982
;    70          0070  1 ! 1-019 - Put call to fsetcol in line. STS 11-Oct-1982
;    71          0071  1 ! 1-020 - Start work on NOTRUNCATE mode.  JBS 11-Oct-1982
;    72          0072  1 ! 1-021 - Debug NOTRUNCATE mode.  JBS 12-Oct-1982
;    73          0073  1 ! 1-022 - Fix the call to EDT$$FMT_CHWID.  JBS 13-Oct-1982
;    74          0074  1 ! 1-023 - Add the second argument.  JBS 23-Oct-1982
;    75          0075  1 ! 1-024 - Use SCR_EDIT_MINPOS.  JBS 28-Oct-1982
;    76          0076  1 ! 1-025 - Be sure to print at least one character before the last character
;    77          0077  1 !                 of a line, so we won't be hit by the VT100's autowrap.  JBS 10-Nov-1982
;    78          0078  1 ! 1-026 - Set the final MINPOS to CHR_TO, so CHMEINPUT's text won't have to be rewritten.  JBS 02-Dec-1982
;    79          0079  1 ! 1-027 - Change the handling of EDT$$G_SHF.  JBS 14-Dec-1982
;    80          0080  1 ! 1-028 - Maintain and use SCR_EDIT_MAXPOS.  JBS 27-Dec-1982
;    81          0081  1 ! 1-029 - Don't erase to end of line if we do not repaint the whole line.  JBS 27-Dec-1982
;    82          0082  1 ! 1-030 - Put the most common cases of character formatting in-line, to improve speed.  JBS 04-Jan-1983
;    83          0083  1 ! 1-031 - Be sure the blob is painted with correct vodeo attributes.  JBS 21-Mar-1983
;    84          0084  1 ! 1-032 - Make sure we are in replace mode.  JBS 01-Apr-1983
;    85          0085  1 ! 1-033 - Adjust the width of a tab if it is at the front of a continued line.  JBS 03-May-1983
;    86          0086  1 ! 1-034 - Fix bug where if the EOB marker displays in the last column of the
;    87          0087  1 !                 screen, it was deleted when we attempted to delete to end of line.
;    88          0088  1 !                 The bug happened only if advancing to that line without clearing the
;    89          0089  1 !                 screen first.  REM 12-Dec-1983
;    90          0090  1 !--
;    91          0091  1
```

```
;    93          0092  1 %SBTTL 'Declarations'
;    94          0093  1 !
;    95          0094  1 ! TABLE OF CONTENTS:
;    96          0095  1 !
;    97          0096  1
;    98          0097  1 REQUIRE 'EDTSRC:TRAROUNAM';
;    99          0536  1
;   100          0537  1 FORWARD ROUTINE
;   101          0538  1     EDT$$SC_RFRELN : NOVALUE;
;   102          0539  1
;   103          0540  1 !
;   104          0541  1 ! INCLUDE FILES:
;   105          0542  1 !
;   106          0543  1
;   107          0544  1 REQUIRE 'EDTSRC:EDTREQ';
;   108          0679  1
;   109          0680  1 !
;   110          0681  1 ! MACROS:
;   111          0682  1 !
;   112          0683  1 !       NONE
;   113          0684  1 !
;   114          0685  1 ! EQUATED SYMBOLS:
;   115          0686  1 !
;   116          0687  1 !       NONE
;   117          0688  1 !
;   118          0689  1 ! OWN STORAGE:
;   119          0690  1 !
;   120          0691  1 !       NONE
;   121          0692  1 !
;   122          0693  1 ! EXTERNAL REFERENCES:
;   123          0694  1 !
;   124          0695  1 !       In the routine
```

```
;  126     0696  1 %SBTTL 'EDT$$SC_RFRELN  - refresh a line on the screen'
;  127     0697  1
;  128     0698  1 GLOBAL ROUTINE EDT$$SC_RFRELN (                        ! Refresh a line on the screen
;  129     0699  1     SCRPTR,                                            ! address of line info being refreshed
;  130     0700  1     ERASED                                             ! 1 = line has been erased
;  131     0701  1     ) : NOVALUE =
;  132     0702  1
;  133     0703  1 !++
;  134     0704  1 ! FUNCTIONAL DESCRIPTION:
;  135     0705  1 !
;  136     0706  1 !       This routine refreshes a single line on the screen.  It expects EDT$$G_CS_LNO
;  137     0707  1 !       to be the screen line number to be refreshed.  This routine operates only on
;  138     0708  1 !       the specified line; it does not clear the screen after an [EOB], for example.
;  139     0709  1 !
;  140     0710  1 ! FORMAL PARAMETERS:
;  141     0711  1 !
;  142     0712  1 !  SCRPTR               Pointer to the screen block for the line being refreshed
;  143     0713  1 !
;  144     0714  1 !  ERASED               1 = the line has already been erased
;  145     0715  1 !
;  146     0716  1 ! IMPLICIT INPUTS:
;  147     0717  1 !
;  148     0718  1 !       EDT$$G_CS_LNO
;  149     0719  1 !       EDT$$A_SEL_BUF
;  150     0720  1 !       EDT$$G_SHF
;  151     0721  1 !       EDT$$G_TI_WID
;  152     0722  1 !       EDT$$A_WK_LN
;  153     0723  1 !       EDT$$G_FMT_LNPOS
;  154     0724  1 !       EDT$$A_CUR_TBCB
;  155     0725  1 !       EDT$$A_EOB_SCRPTR
;  156     0726  1 !       EDT$$A_FMT_CUR
;  157     0727  1 !       EDT$$G_PRV_COL
;  158     0728  1 !       EDT$$T_FMT_BUF
;  159     0729  1 !       EDT$$G_INSERT_MODE
;  160     0730  1 !
;  161     0731  1 ! IMPLICIT OUTPUTS:
;  162     0732  1 !
;  163     0733  1 !       EDT$$A_FMT_CUR
;  164     0734  1 !       EDT$$G_PRV_COL
;  165     0735  1 !
;  166     0736  1 ! ROUTINE VALUE:
;  167     0737  1 !
;  168     0738  1 !       NONE
;  169     0739  1 !
;  170     0740  1 ! SIDE EFFECTS:
;  171     0741  1 !
;  172     0742  1 !       Writes on the screen.
;  173     0743  1 !
;  174     0744  1 !--
;  175     0745  1
;  176     0746  2     BEGIN
;  177     0747  2
;  178     0748  2     EXTERNAL ROUTINE
;  179     0749  2         EDT$$FMT_CH : NOVALUE,                         ! Output a character
;  180     0750  2         EDT$$FMT_CHWID,                                ! Compute the width of a character
;  181     0751  2         EDT$$SC_SHWBLOB : NOVALUE,                     ! Output a blob
;  182     0752  2         EDT$$SC_REVIDCHK : NOVALUE,                    ! Check for reverse video based on select region
```

```
;  183     0753  2          EDT$$SC_NONREVID : NOVALUE,                  ! Go to normal video
;  184     0754  2          EDT$$SC_POSCSIF : NOVALUE,                   ! Position the cursor
;  185     0755  2          EDT$$SC_ERATOEOL : NOVALUE,                  ! Erase to end of line
;  186     0756  2          EDT$$SC_ERAALL : NOVALUE,                    ! Erase to end of screen
;  187     0757  2          EDT$$FMT_TEXT : NOVALUE,                     ! Print [EOB]
;  188     0758  2          EDT$$OUT_FMTBUF,                             ! Output the format buffer
;  189     0759  2          EDT$$SC_REP_MODE : NOVALUE;                  ! Put the screen in replace mode
;  190     0760  2
;  191     0761  2      EXTERNAL
;  192     0762  2          EDT$$A_EOB_SCRPTR : REF SCREEN_LINE,         ! Special line block for [EOB]
;  193     0763  2          EDT$$G_CS_LNO,                               ! current screen line
;  194     0764  2          EDT$$A_SEL_BUF,                              ! select buffer.
;  195     0765  2          EDT$$G_SHF,                                  ! The number of columns shifted.
;  196     0766  2          EDT$$G_TI_WID,                               ! Width of terminal line.
;  197     0767  2          EDT$$A_WK_LN : REF LIN_BLOCK,                ! Current line pointer.
;  198     0768  2          EDT$$G_FMT_LNPOS,                            ! Current column number
;  199     0769  2          EDT$$A_CUR_BUF : REF TBCB_BLOCK,             ! Pointer to current text control block
;  200     0770  2          EDT$$A_FMT_CUR,                              ! Pointer to next char in output buffer
;  201     0771  2          EDT$$T_FMT_BUF : BLOCK [CH$ALLOCATION (EDT$$K_FMT_BUFLEN)],     ! Output buffer
;  202     0772  2          EDT$$G_PRV_COL,                              ! The cursor column number
;  203     0773  2          EDT$$G_INSERT_MODE;                          ! 1 = screen is in insert mode
;  204     0774  2
;  205     0775  2      MAP
;  206     0776  2          SCRPTR : REF SCREEN_LINE;
;  207     0777  2
;  208     0778  2      LOCAL
;  209     0779  2          TXTPTR,
;  210     0780  2          ORIG_TXTPTR,
;  211     0781  2          LEN,
;  212     0782  2          CHAR,
;  213     0783  2          CHAR_WIDTH,
;  214     0784  2          LEFT,
;  215     0785  2          FIRST_CHAR,
;  216     0786  2          WIDTH,
;  217     0787  2          SIMPLE_CHAR,
;  218     0788  2          MAXPOS;
;  219     0789  2
;  220     0790  2      !+
;  221     0791  2      ! Make sure we are in replace mode.
;  222     0792  2      !-
;  223     0793  2
;  224     0794  2          IF (.EDT$$G_INSERT_MODE NEQ 0) THEN EDT$$SC_REP_MODE ();
;  225     0795  2
;  226     0796  2      !+
;  227     0797  2      ! Check for EOB.
;  228     0798  2      !-
;  229     0799  2
;  230     0800  3          IF (.SCRPTR EQLA .EDT$$A_EOB_SCRPTR)
;  231     0801  2          THEN
;  232     0802  3              BEGIN
;  233     0803  3              EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, 0);
;  234     0804  3              EDT$$SC_NONREVID ();
;  235     0805  3              EDT$$FMT_TEXT (0);
;  236     0806  3
;  237     0807  3
;  238     0808  4              IF (( NOT .ERASED) AND (.SCRPTR [SCR_EDIT_MAXPOS] EQL 255))
;  239     0809  3              THEN            ! If not erased and not at end of the line,
```

```
;  240     0810  4                BEGIN
;  241     0811  4 !!              EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, MAX (0, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF));
;  242     0812  4                 EDT$$SC_ERATOEOL ()             !  erase any extra characters that may
;  243     0813  4                                                 !  have been left on the screen's line.
;  244     0814  3                 END;
;  245     0815  3
;  246     0816  3             !+
;  247     0817  3             ! Mark the line as finished with its edit.
;  248     0818  3             !-
;  249     0819  3                 SCRPTR [SCR_EDIT_MINPOS] = 255;
;  250     0820  3                 SCRPTR [SCR_EDIT_MAXPOS] = 0;
;  251     0821  3                 SCRPTR [SCR_EDIT_FLAGS] = .SCRPTR [SCR_EDIT_FLAGS] AND ( NOT (SCR_EDIT_MODIFY OR SCR_EDIT_INSLN));
;  252     0822  3                 RETURN;
;  253     0823  3                 END;
;  254     0824  2
;  255     0825  2             !+
;  256     0826  2             ! Not EOB.  Position to the first character to be updated in the line,
;  257     0827  2             ! keeping track of the screen column which it will occupy.
;  258     0828  2             !-
;  259     0829  2                 WIDTH = .EDT$$G_TI_WID + .EDT$$G_SHF;
;  260     0830  2                 LEFT = .SCRPTR [SCR_CHR_FROM];
;  261     0831  2                 LEN = MIN (.SCRPTR [SCR_CHR_TO] + 1, .EDT$$A_WK_LN [LIN_LENGTH]) - .LEFT;
;  262     0832  2                 TXTPTR = CH$PLUS (EDT$$A_WK_LN [LIN_TEXT], .LEFT);
;  263     0833  2                 ORIG_TXTPTR = .TXTPTR;
;  264     0834  2                 EDT$$G_FMT_LNPOS = 0;
;  265     0835  2                 CHAR = CH$RCHAR_A (TXTPTR);
;  266     0836  2
;  267     0837  3                 IF ((.CHAR GEQ %X'20') AND (.CHAR LEQ %X'7E'))
;  268     0838  2                 THEN
;  269     0839  3                     BEGIN
;  270     0840  3                     CHAR_WIDTH = 1;
;  271     0841  3                     SIMPLE_CHAR = 1;
;  272     0842  3                     END
;  273     0843  2                 ELSE
;  274     0844  3                     BEGIN
;  275     0845  3                     CHAR_WIDTH = EDT$$FMT_CHWID (.CHAR, .EDT$$G_FMT_LNPOS);
;  276     0846  3                     SIMPLE_CHAR = 0;
;  277     0847  2                     END;
;  278     0848  2
;  279     0849  2             !+
;  280     0850  2             ! Skip over unmodified characters on this line.
;  281     0851  2             !-
;  282     0852  2
;  283     0853  3                 WHILE (((.TXTPTR - .ORIG_TXTPTR) LEQ .SCRPTR [SCR_EDIT_MINPOS]) AND          !
;  284     0854  3                     (.LEN GTR 0) AND                                                         !
;  285     0855  2                     (.EDT$$G_FMT_LNPOS LSS (.WIDTH - .CHAR_WIDTH - 1))) DO
;  286     0856  3                     BEGIN
;  287     0857  3             !+
;  288     0858  3             ! Account for the blob at the front of continued lines.
;  289     0859  3             !-
;  290     0860  3
;  291     0861  4                     IF ((.EDT$$G_FMT_LNPOS EQL 0) AND (.SCRPTR [SCR_LINE_IDX] NEQ 0))
;  292     0862  3                     THEN
;  293     0863  4                         BEGIN
;  294     0864  4             !+
;  295     0865  4             ! Adjust for the blob at the front of a continued line.  This code requires
;  296     0866  4             ! that the shift amount always be a multiple of 8, so that shifting doesn't
```

```
;  297     0867  4 ! change tab stops.
;  298     0868  4 !-
;  299     0869  4               EDT$$G_FMT_LNPOS = .EDT$$G_SHF + 2;
;  300     0870  4
;  301     0871  5               IF (.CHAR EQL ASC_K_TAB)
;  302     0872  4               THEN
;  303     0873  5                   BEGIN
;  304     0874  5                   CHAR_WIDTH = .CHAR_WIDTH - 2;
;  305     0875  5                   ASSERT (.CHAR_WIDTH EQL 6);
;  306     0876  4                   END;
;  307     0877  4
;  308     0878  3               END;
;  309     0879  3
;  310     0880  3           EDT$$G_FMT_LNPOS = .EDT$$G_FMT_LNPOS + .CHAR_WIDTH;
;  311     0881  3           LEN = .LEN - 1;
;  312     0882  3           CHAR = CH$RCHAR_A (TXTPTR);
;  313     0883  3
;  314     0884  4           IF ((.CHAR GEQ %X'20') AND (.CHAR LEQ %X'7E'))
;  315     0885  3           THEN
;  316     0886  4               BEGIN
;  317     0887  4               CHAR_WIDTH = 1;
;  318     0888  4               SIMPLE_CHAR = 1;
;  319     0889  4               END
;  320     0890  3           ELSE
;  321     0891  4               BEGIN
;  322     0892  4               CHAR_WIDTH = EDT$$FMT_CHWID (.CHAR, .EDT$$G_FMT_LNPOS);
;  323     0893  4               SIMPLE_CHAR = 0;
;  324     0894  3               END;
;  325     0895  3
;  326     0896  2           END;
;  327     0897  2
;  328     0898  2 !+
;  329     0899  2 ! Put the characters into the format buffer.
;  330     0900  2 !-
;  331     0901  2     FIRST_CHAR = 1;
;  332     0902  2 !+
;  333     0903  2 ! If this is a continued line, indicate this at the front of the line.
;  334     0904  2 !-
;  335     0905  2
;  336     0906  3     IF ((.SCRPTR [SCR_LINE_IDX] NEQ 0) AND (.EDT$$G_FMT_LNPOS EQL 0))
;  337     0907  2     THEN
;  338     0908  3         BEGIN
;  339     0909  3         EDT$$G_FMT_LNPOS = .EDT$$G_SHF;
;  340     0910  3         EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF);
;  341     0911  3         FIRST_CHAR = 0;
;  342     0912  3
;  343     0913  4         IF (.EDT$$A_SEL_BUF EQL .EDT$$A_CUR_BUF)        !
;  344     0914  3         THEN
;  345     0915  3             EDT$$SC_REVIDCHK (CH$DIFF (.TXTPTR, CH$PTR (EDT$$A_WK_LN [LIN_TEXT])) - 1);
;  346     0916  3
;  347     0917  3         EDT$$SC_SHWBLOB ();
;  348     0918  3         EDT$$FMT_CH (%C' ');
;  349     0919  3
;  350     0920  4         IF (.CHAR EQL ASC_K_TAB)
;  351     0921  3         THEN
;  352     0922  4             BEGIN
;  353     0923  4             CHAR_WIDTH = .CHAR_WIDTH - 2;
```

```
;  354          0924  4              ASSERT (.CHAR_WIDTH EQL 6);
;  355          0925  3              END;
;  356          0926  3
;  357          0927  2          END;
;  358          0928  2
;  359          0929  2      MAXPOS = .SCRPTR [SCR_EDIT_MAXPOS];
;  360          0930  2  !+
;  361          0931  2  ! This is the loop that actually puts characters into the format buffer for output to the screen.
;  362          0932  2  ! The time around this loop is critical to EDT's performance in screen mode.
;  363          0933  2  !-
;  364          0934  2
;  365          0935  3      WHILE ((.LEN GTR 0) AND (.EDT$$G_FMT_LNPOS LSS (.WIDTH - .CHAR_WIDTH)) AND  !
;  366          0936  2          ((.TXTPTR - .ORIG_TXTPTR - 1) LEQ .MAXPOS)) DO
;  367          0937  3          BEGIN
;  368          0938  3
;  369          0939  4          IF (.EDT$$A_SEL_BUF EQL .EDT$$A_CUR_BUF)        !
;  370          0940  3          THEN
;  371          0941  3              EDT$$SC_REVIDCHK (CH$DIFF (.TXTPTR, CH$PTR (EDT$$A_WK_LN [LIN_TEXT])) - 1);
;  372          0942  3
;  373          0943  4          IF (.EDT$$G_FMT_LNPOS GEQ .EDT$$G_SHF)
;  374          0944  3          THEN
;  375          0945  4              BEGIN
;  376          0946  4
;  377          0947  4              IF .FIRST_CHAR
;  378          0948  4              THEN
;  379          0949  5                  BEGIN
;  380          0950  5                  EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF);
;  381          0951  5                  FIRST_CHAR = 0;
;  382          0952  4                  END;
;  383          0953  4
;  384          0954  4  !+
;  385          0955  4  ! Put the character in the format buffer.
;  386          0956  4  ! Do simple characters in-line; call EDT$$FMT_CH for complex characters.
;  387          0957  4  !-
;  388          0958  4
;  389          0959  4              IF .SIMPLE_CHAR
;  390          0960  4              THEN
;  391          0961  5                  BEGIN
;  392          0962  5                  EDT$$G_FMT_LNPOS = .EDT$$G_FMT_LNPOS + 1;
;  393          0963  5
;  394          0964  6                  IF (.EDT$$A_FMT_CUR EQLA CH$PTR (EDT$$T_FMT_BUF, EDT$$K_FMT_BUFLEN))
;  395          0965  5                  THEN
;  396          0966  6                      BEGIN
;  397          0967  6  !+
;  398          0968  6  ! We have reached the end of the buffer; empty it.
;  399          0969  6  !-
;  400          0970  6
;  401          0971  6                      LOCAL
;  402          0972  6                          SAV_LNPOS;
;  403          0973  6
;  404          0974  6                      SAV_LNPOS = .EDT$$G_FMT_LNPOS;
;  405          0975  6                      EDT$$OUT_FMTBUF ();
;  406          0976  6                      EDT$$G_FMT_LNPOS = .SAV_LNPOS;
;  407          0977  5                      END;
;  408          0978  5
;  409          0979  5                  CH$WCHAR_A (.CHAR, EDT$$A_FMT_CUR);
;  410          0980  5
```

```
;  411     0981  5                 IF (.EDT$$G_PRV_COL NEQ (.EDT$$G_TI_WID - 1)) THEN EDT$$G_PRV_COL = .EDT$$G_PRV_COL + 1;
;  412     0982  5
;  413     0983  5                 END
;  414     0984  4             ELSE
;  415     0985  4                 EDT$$FMT_CH (.CHAR);
;  416     0986  4
;  417     0987  4             END
;  418     0988  3         ELSE
;  419     0989  3             EDT$$G_FMT_LNPOS = .EDT$$G_FMT_LNPOS + .CHAR_WIDTH;
;  420     0990  3
;  421     0991  3         LEN = .LEN - 1;
;  422     0992  3         CHAR = CH$RCHAR_A (TXTPTR);
;  423     0993  3
;  424     0994  4         IF ((.CHAR GEQ %X'20') AND (.CHAR LEQ %X'7E'))
;  425     0995  3         THEN
;  426     0996  4             BEGIN
;  427     0997  4             CHAR_WIDTH = 1;
;  428     0998  4             SIMPLE_CHAR = 1;
;  429     0999  4             END
;  430     1000  3         ELSE
;  431     1001  4             BEGIN
;  432     1002  4             CHAR_WIDTH = EDT$$FMT_CHWID (.CHAR, .EDT$$G_FMT_LNPOS);
;  433     1003  4             SIMPLE_CHAR = 0;
;  434     1004  3             END;
;  435     1005  3
;  436     1006  2         END;
;  437     1007  2
;  438     1008  2 !+
;  439     1009  2 ! If we have not finished the line, it may be because the line won't fit on the screen.
;  440     1010  2 ! Since the loop above stops one column short of the right edge of the screen, there
;  441     1011  2 ! may be just room for one more character; if so, put it out.  If not, put a blob in the
;  442     1012  2 ! last column.
;  443     1013  2 !-
;  444     1014  2
;  445     1015  3     IF ((.LEN GTR 0) AND ((.TXTPTR - .ORIG_TXTPTR - 1) LEQ .MAXPOS))
;  446     1016  2     THEN
;  447     1017  3         BEGIN
;  448     1018  3
;  449     1019  4         IF ((.LEN EQL 1) AND (.EDT$$G_FMT_LNPOS EQL (.WIDTH - .CHAR_WIDTH)) AND          !
;  450     1020  4             (.EDT$$G_FMT_LNPOS GEQ .EDT$$G_SHF))
;  451     1021  3         THEN
;  452     1022  4             BEGIN
;  453     1023  4
;  454     1024  5             IF (.EDT$$A_SEL_BUF EQL .EDT$$A_CUR_BUF)     !
;  455     1025  4             THEN
;  456     1026  4                 EDT$$SC_REVIDCHK (CH$DIFF (.TXTPTR, CH$PTR (EDT$$A_WK_LN [LIN_TEXT])) - 1);
;  457     1027  4
;  458     1028  4             IF .FIRST_CHAR
;  459     1029  4             THEN
;  460     1030  5                 BEGIN
;  461     1031  5                 EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF);
;  462     1032  5                 FIRST_CHAR = 0;
;  463     1033  4                 END;
;  464     1034  4
;  465     1035  4             EDT$$FMT_CH (.CHAR);
;  466     1036  4             LEN = .LEN - 1;
;  467     1037  4             END
```

```
;  468          1038  3          ELSE
;  469          1039  4              BEGIN
;  470          1040  4
;  471          1041  5              IF (( NOT .ERASED) AND (.SCRPTR [SCR_EDIT_MAXPOS] EQL 255))
;  472          1042  4              THEN
;  473          1043  5                  BEGIN
;  474          1044  5                  EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, MAX (0, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF));
;  475          1045  5                  EDT$$SC_ERATOEOL ();
;  476          1046  4                  END;
;  477          1047  4
;  478          1048  4 !+
;  479          1049  4 ! If there is room left on the line, it may be that we have printed no characters.
;  480          1050  4 ! Therefore, print a space to be sure that the VT100's autowrap flag is not set.
;  481          1051  4 !-
;  482          1052  4
;  483          1053  4              IF (.EDT$$G_FMT_LNPOS LSS (.EDT$$G_TI_WID - 1)) THEN EDT$$FMT_CH (%C' ');
;  484          1054  4
;  485          1055  4              EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, .EDT$$G_TI_WID - 1);
;  486          1056  4              EDT$$SC_SHWBLOB ();
;  487          1057  3              END;
;  488          1058  3
;  489          1059  3          END
;  490          1060  3 !+
;  491          1061  3 ! Throw in an erase to end of line sequence if we have painted as close as we can to the right margin.
;  492          1062  3 ! Suppress the sequence if we have just put a character at the right margin or if the line is already erased
;  493          1063  3 !-
;  494          1064  2      ELSE
;  495          1065  2
;  496          1066  2          IF (( NOT .ERASED) AND (.SCRPTR [SCR_EDIT_MAXPOS] EQL 255))
;  497          1067  2          THEN
;  498          1068  3              BEGIN
;  499          1069  3
;  500          1070  3              IF .FIRST_CHAR THEN EDT$$SC_POSCSIF (.EDT$$G_CS_LNO, MAX (0, .EDT$$G_FMT_LNPOS - .EDT$$G_SHF));
;  501          1071  3
;  502          1072  3              EDT$$SC_ERATOEOL ();
;  503          1073  2              END;
;  504          1074  2
;  505          1075  2 !+
;  506          1076  2 ! Mark the line as finished with its edit.
;  507          1077  2 !-
;  508          1078  2      SCRPTR [SCR_EDIT_MINPOS] = MIN (.SCRPTR [SCR_CHR_TO] - .SCRPTR [SCR_CHR_FROM] + 1, 255);
;  509          1079  2      SCRPTR [SCR_EDIT_MAXPOS] = 0;
;  510          1080  2      SCRPTR [SCR_EDIT_FLAGS] = .SCRPTR [SCR_EDIT_FLAGS] AND ( NOT (SCR_EDIT_MODIFY OR SCR_EDIT_INSLN));
;  511          1081  1      END;                                                    ! of routine EDT$$SC_RFRELN


                                                                .TITLE  EDT$SCRRLIN EDT$SCRRLIN - refresh a screen line
                                                                .IDENT  \V04-000\

                                                                .EXTRN  EDT$$FMT_CH, EDT$$FMT_CHWID
                                                                .EXTRN  EDT$$SC_SHWBLOB
                                                                .EXTRN  EDT$$SC_REVIDCHK
                                                                .EXTRN  EDT$$SC_NONREVID
                                                                .EXTRN  EDT$$SC_POSCSIF
                                                                .EXTRN  EDT$$SC_ERATOEOL
                                                                .EXTRN  EDT$$SC_ERAALL, EDT$$FMT_TEXT
                                                                .EXTRN  EDT$$OUT_FMTBUF
```

```
                                                          .EXTRN  EDT$$SC_REP_MODE
                                                          .EXTRN  EDT$$A_EOB_SCRPTR
                                                          .EXTRN  EDT$$G_CS_LNO, EDT$$A_SEL_BUF
                                                          .EXTRN  EDT$$G_SHF, EDT$$G_TI_WID
                                                          .EXTRN  EDT$$A_WK_LN, EDT$$G_FMT_LNPOS
                                                          .EXTRN  EDT$$A_CUR_BUF, EDT$$A_FMT_CUR
                                                          .EXTRN  EDT$$T_FMT_BUF, EDT$$G_PRV_COL
                                                          .EXTRN  EDT$$G_INSERT_MODE
                                                          .EXTRN  EDT$$INTER_ERR

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

                                        0FFC 00000        .ENTRY  EDT$$SC_RFRELN, Save R2,R3,R4,R5,R6,R7,R8,- ; 0698
                                                                  R9,R10,R11                                   ;
                    5E            04    C2 00002          SUBL2   #4, SP                                       ;
                       00000000G  00    D5 00005          TSTL    EDT$$G_INSERT_MODE                           ; 0794
                                  07    13 0000B          BEQL    1$                                           ;
          00000000G 00            00    FB 0000D          CALLS   #0, EDT$$SC_REP_MODE                         ;
                    54        04  AC    D0 00014 1$:      MOVL    SCRPTR, R4                                   ; 0800
          00000000G 00            54    D1 00018          CMPL    R4, EDT$$A_EOB_SCRPTR                        ;
                                  38    12 0001F          BNEQ    3$                                           ;
                                  7E    D4 00021          CLRL    -(SP)                                        ; 0803
                       00000000G  00    DD 00023          PUSHL   EDT$$G_CS_LNO                                ;
          00000000G 00            02    FB 00029          CALLS   #2, EDT$$SC_POSCSIF                          ;
          00000000G 00            00    FB 00030          CALLS   #0, EDT$$SC_NONREVID                         ; 0804
                                  7E    D4 00037          CLRL    -(SP)                                        ; 0805
          00000000G 00            01    FB 00039          CALLS   #1, EDT$$FMT_TEXT                            ;
                    0E        08  AC    E8 00040          BLBS    ERASED, 2$                                   ; 0808
                 FF 8F        0C  A4    91 00044          CMPB    12(R4), #255                                 ;
                                  07    12 00049          BNEQ    2$                                           ;
          00000000G 00            00    FB 0004B          CALLS   #0, EDT$$SC_ERATOEOL                         ; 0812
                 0B A4            01    8E 00052 2$:      MNEGB   #1, 11(R4)                                   ; 0819
                                036F    31 00056          BRW     38$                                          ; 0820
       53 00000000G 00 00000000G  00    C1 00059 3$:      ADDL3   EDT$$G_SHF, EDT$$G_TI_WID, WIDTH             ; 0829
                    51        09  A4    9A 00065          MOVZBL  9(R4), LEFT                                  ; 0830
                    52        0A  A4    9A 00069          MOVZBL  10(R4), R2                                   ; 0831
                                  52    D6 0006D          INCL    R2                                           ;
                 50 00000000G     00    D0 0006F          MOVL    EDT$$A_WK_LN, R0                             ;
  52       60       08            00    ED 00076          CMPZV   #0, #8, (R0), R2                             ;
                                  03    18 0007B          BGEQ    4$                                           ;
                    52            60    9A 0007D          MOVZBL  (R0), R2                                     ;
           5B       52            51    C3 00080 4$:      SUBL3   LEFT, R2, LEN                                ;
                    52       07 A140    9E 00084          MOVAB   7(LEFT)[R0], TXTPTR                          ; 0832
                    6E            52    D0 00089          MOVL    TXTPTR, ORIG_TXTPTR                          ; 0833
                       00000000G  00    D4 0008C          CLRL    EDT$$G_FMT_LNPOS                             ; 0834
                                  69    11 00092          BRB     8$                                           ; 0835
                       00000000G  00    DD 00094 5$:      PUSHL   EDT$$G_FMT_LNPOS                             ; 0845
                                  59    DD 0009A          PUSHL   CHAR                                         ;
          00000000G 00            02    FB 0009C          CALLS   #2, EDT$$FMT_CHWID                           ;
                    55            50    D0 000A3          MOVL    R0, CHAR_WIDTH                               ;
                                  57    D4 000A6          CLRL    SIMPLE_CHAR                                  ; 0846
           50       52            6E    C3 000A8 6$:      SUBL3   ORIG_TXTPTR, TXTPTR, R0                      ; 0853
  50       0B  A4   08            00    ED 000AC          CMPZV   #0, #8, 11(R4), R0                           ;
                                  62    19 000B2          BLSS    9$                                           ;
                                  5B    D5 000B4          TSTL    LEN                                          ; 0854
                                  5E    15 000B6          BLEQ    9$                                           ;
           50       53            55    C3 000B8          SUBL3   CHAR_WIDTH, WIDTH, R0                        ; 0855
```

```
                                    50 D7 000BC          DECL    R0                                          ;
                      50 00000000G 00 D1 000BE          CMPL    EDT$$G_FMT_LNPOS, R0                        ;
                                    4F 18 000C5          BGEQ    9$                                          ;
                         00000000G 00 D5 000C7          TSTL    EDT$$G_FMT_LNPOS                            ; 0861
                                    25 12 000CD          BNEQ    7$                                          ;
                               08 A4 95 000CF          TSTB    8(R4)                                       ;
                                    20 13 000D2          BEQL    7$                                          ;
00000000G 00 00000000G 00           02 C1 000D4          ADDL3   #2, EDT$$G_SHF, EDT$$G_FMT_LNPOS            ; 0869
                            09     59 D1 000E0          CMPL    CHAR, #9                                    ; 0871
                                    0F 12 000E3          BNEQ    7$                                          ;
                            55     02 C2 000E5          SUBL2   #2, CHAR_WIDTH                              ; 0874
                            06     55 D1 000E8          CMPL    CHAR_WIDTH, #6                              ; 0875
                                    07 13 000EB          BEQL    7$                                          ;
                  00000000G 00     00 FB 000ED          CALLS   #0, EDT$$INTER_ERR                          ;
                  00000000G 00     55 C0 000F4 7$:      ADDL2   CHAR_WIDTH, EDT$$G_FMT_LNPOS                ; 0880
                                    5B D7 000FB          DECL    LEN                                         ; 0881
                            59     82 9A 000FD 8$:      MOVZBL  (TXTPTR)+, CHAR                             ; 0882
                            20     59 D1 00100          CMPL    CHAR, #32                                   ; 0884
                                    8F 19 00103          BLSS    5$                                          ;
                  0000007E 8F      59 D1 00105          CMPL    CHAR, #126                                  ;
                                    86 14 0010C          BGTR    5$                                          ;
                            55     01 D0 0010E          MOVL    #1, CHAR_WIDTH                              ; 0887
                            57     01 D0 00111          MOVL    #1, SIMPLE_CHAR                             ; 0888
                                    92 11 00114          BRB     6$                                          ; 0884
                            58     01 D0 00116 9$:      MOVL    #1, FIRST_CHAR                              ; 0901
                               08 A4 95 00119          TSTB    8(R4)                                       ; 0906
                                    70 13 0011C          BEQL    11$                                         ;
                         00000000G 00 D5 0011E          TSTL    EDT$$G_FMT_LNPOS                            ;
                                    68 12 00124          BNEQ    11$                                         ;
                      50 00000000G 00 D0 00126          MOVL    EDT$$G_SHF, R0                              ; 0909
                  00000000G 00     50 D0 0012D          MOVL    R0, EDT$$G_FMT_LNPOS                        ;
               7E 00000000G 00     50 C3 00134          SUBL3   R0, EDT$$G_FMT_LNPOS, -(SP)                 ; 0910
                         00000000G 00 DD 0013C          PUSHL   EDT$$G_CS_LNO                               ;
                  00000000G 00     02 FB 00142          CALLS   #2, EDT$$SC_POSCSIF                         ;
                                    58 D4 00149          CLRL    FIRST_CHAR                                  ; 0911
                  00000000G 00 00000000G 00 D1 0014B    CMPL    EDT$$A_SEL_BUF, EDT$$A_CUR_BUF              ; 0913
                                    12 12 00156          BNEQ    10$                                         ;
            50          52 00000000G 00 C3 00158        SUBL3   EDT$$A_WK_LN, TXTPTR, R0                    ; 0915
                               F8 A0 9F 00160          PUSHAB  -8(R0)                                      ;
                  00000000G 00     01 FB 00163          CALLS   #1, EDT$$SC_REVIDCHK                        ;
                  00000000G 00     00 FB 0016A 10$:     CALLS   #0, EDT$$SC_SHWBLOB                         ; 0917
                                    20 DD 00171          PUSHL   #32                                         ; 0918
                  00000000G 00     01 FB 00173          CALLS   #1, EDT$$FMT_CH                             ;
                            09     59 D1 0017A          CMPL    CHAR, #9                                    ; 0920
                                    0F 12 0017D          BNEQ    11$                                         ;
                            55     02 C2 0017F          SUBL2   #2, CHAR_WIDTH                              ; 0923
                            06     55 D1 00182          CMPL    CHAR_WIDTH, #6                              ; 0924
                                    07 13 00185          BEQL    11$                                         ;
                  00000000G 00     00 FB 00187          CALLS   #0, EDT$$INTER_ERR                          ;
                            5A  0C A4 9A 0018E 11$:     MOVZBL  12(R4), MAXPOS                              ; 0929
                                    5B D5 00192 12$:     TSTL    LEN                                         ; 0935
                                    03 14 00194          BGTR    14$                                         ;
                                  00FB 31 00196 13$:     BRW     23$                                         ;
            50              53     55 C3 00199 14$:     SUBL3   CHAR_WIDTH, WIDTH, R0                       ;
                            50 00000000G 00 D1 0019D    CMPL    EDT$$G_FMT_LNPOS, R0                        ;
                                    F0 18 001A4          BGEQ    13$                                         ;
            50              52     6E C3 001A6          SUBL3   ORIG_TXTPTR, TXTPTR, R0                     ; 0936
```

```
                                50   D7 001AA          DECL    R0                                                      ;
                             5A     50   D1 001AC          CMPL    R0, MAXPOS                                              ;
                                    E5   14 001AF          BGTR    13$                                                     ;
                00000000G    00 00000000G   00   D1 001B1          CMPL    EDT$$A_SEL_BUF, EDT$$A_CUR_BUF                  ; 0939
                                    12   12 001BC          BNEQ    15$                                                     ;
                50           52 00000000G   00   C3 001BE          SUBL3   EDT$$A_WK_LN, TXTPTR, R0                        ; 0941
                                         F8   A0   9F 001C6          PUSHAB  -8(R0)                                          ;
                00000000G    00             01   FB 001C9          CALLS   #1, EDT$$SC_REVIDCHK                            ;
                             51 00000000G   00   D0 001D0 15$:       MOVL    EDT$$G_FMT_LNPOS, R1                            ; 0943
                             50 00000000G   00   D0 001D7          MOVL    EDT$$G_SHF, R0                                  ;
                             50             51   D1 001DE          CMPL    R1, R0                                          ;
                                            78   19 001E1          BLSS    19$                                             ;
                             13             58   E9 001E3          BLBC    FIRST_CHAR, 16$                                 ; 0947
                7E           51             50   C3 001E6          SUBL3   R0, RT, -(SP)                                   ; 0950
                                 00000000G  00   DD 001EA          PUSHL   EDT$$G_CS_LNO                                   ;
                00000000G    00             02   FB 001F0          CALLS   #2, EDT$$SC_POSCSIF                             ;
                                            58   D4 001F7          CLRL    FIRST_CHAR                                      ; 0951
                             54             57   E9 001F9 16$:       BLBC    SIMPLE_CHAR, 18$                                ; 0959
                                 00000000G  00   D6 001FC          INCL    EDT$$G_FMT_LNPOS                                ; 0962
                             50 00000000G   00   9E 00202          MOVAB   EDT$$T_FMT_BUF+512, R0                          ; 0964
                             50 00000000G   00   D1 00209          CMPL    EDT$$A_FMT_CUR, R0                              ;
                                            15   12 00210          BNEQ    17$                                             ;
                             56 00000000G   00   D0 00212          MOVL    EDT$$G_FMT_LNPOS, SAV_LNPOS                     ; 0974
                00000000G    00             00   FB 00219          CALLS   #0, EDT$$OUT_FMTBUF                             ; 0975
                00000000G    00             56   D0 00220          MOVL    SAV_LNPOS, EDT$$G_FMT_LNPOS                     ; 0976
                             50 00000000G   00   D0 00227 17$:       MOVL    EDT$$A_FMT_CUR, R0                              ; 0979
                             60             59   90 0022E          MOVB    CHAR, (R0)                                      ;
                                 00000000G  00   D6 00231          INCL    EDT$$A_FMT_CUR                                  ;
                50 00000000G   00             01   C3 00237          SUBL3   #1, EDT$$G_TI_WID, R0                           ; 0981
                             50 00000000G   00   D1 0023F          CMPL    EDT$$G_PRV_COL, R0                              ;
                                            1A   13 00246          BEQL    20$                                             ;
                                 00000000G  00   D6 00248          INCL    EDT$$G_PRV_COL                                  ;
                                            12   11 0024E          BRB     20$                                             ; 0959
                                            59   DD 00250 18$:       PUSHL   CHAR                                            ; 0985
                00000000G    00             01   FB 00252          CALLS   #1, EDT$$FMT_CH                                 ;
                                            07   11 00259          BRB     20$                                             ; 0943
                00000000G    00             55   C0 0025B 19$:       ADDL2   CHAR_WIDTH, EDT$$G_FMT_LNPOS                    ; 0989
                                            5B   D7 00262 20$:       DECL    LEN                                             ; 0991
                             59             82   9A 00264          MOVZBL  (TXTPTR)+, CHAR                                 ; 0992
                             20             59   D1 00267          CMPL    CHAR, #32                                       ; 0994
                                            11   19 0026A          BLSS    21$                                             ;
                0000007E     8F             59   D1 0026C          CMPL    CHAR, #126                                      ;
                                            08   14 00273          BGTR    21$                                             ;
                             55             01   D0 00275          MOVL    #1, CHAR_WIDTH                                  ; 0997
                             57             01   D0 00278          MOVL    #1, SIMPLE_CHAR                                 ; 0998
                                            14   11 0027B          BRB     22$                                             ; 0994
                                 00000000G  00   DD 0027D 21$:       PUSHL   EDT$$G_FMT_LNPOS                                ; 1002
                                            59   DD 00283          PUSHL   CHAR                                            ;
                00000000G    00             02   FB 00285          CALLS   #2, EDT$$FMT_CHWID                              ;
                             55             50   D0 0028C          MOVL    R0, CHAR_WIDTH                                  ;
                                            57   D4 0028F          CLRL    SIMPLE_CHAR                                     ; 1003
                                       FEFE 31   00291 22$:       BRW     12$                                             ; 0935
                                            5B   D5 00294 23$:       TSTL    LEN                                             ; 1015
                                            03   14 00296          BGTR    25$                                             ;
                                       00DB 31   00298 24$:       BRW     33$                                             ;
                50           52             6E   C3 0029B 25$:       SUBL3   ORIG_TXTPTR, TXTPTR, R0                         ;
                                            50   D7 0029F          DECL    R0                                              ;
```

```
                                5A              50  D1 002A1          CMPL    R0, MAXPOS                                      ;
                                                F2  14 002A4          BGTR    24$                                             ;
                                01              5B  D1 002A6          CMPL    LEN, #1                                         ; 1019
                                                62  12 002A9          BNEQ    28$                                             ;
                                53              55  C2 002AB          SUBL2   CHAR_WIDTH, R3                                  ;
                                53 00000000G    00  D1 002AE          CMPL    EDT$$G_FMT_LNPOS, R3                            ;
                                                56  12 002B5          BNEQ    28$                                             ;
                     00000000G  00 00000000G    00  D1 002B7          CMPL    EDT$$G_FMT_LNPOS, EDT$$G_SHF                    ; 1020
                                                49  19 002C2          BLSS    28$                                             ;
                     00000000G  00 00000000G    00  D1 002C4          CMPL    EDT$$A_SEL_BUF, EDT$$A_CUR_BUF                  ; 1024
                                                11  12 002CF          BNEQ    26$                                             ;
                                52 00000000G    00  C2 002D1          SUBL2   EDT$$A_WK_LN, R2                                ; 1026
                                         F8     A2  9F 002D8          PUSHAB  -8(R2)                                          ;
                     00000000G  00              01  FB 002DB          CALLS   #1, EDT$$SC_REVIDCHK                            ;
                                1B              58  E9 002E2 26$:     BLBC    FIRST_CHAR, 27$                                 ; 1028
                  7E 00000000G  00 00000000G    00  C3 002E5          SUBL3   EDT$$G_SHF, EDT$$G_FMT_LNPOS, -(SP)             ; 1031
                                   00000000G    00  DD 002F1          PUSHL   EDT$$G_CS_LNO                                   ;
                     00000000G  00              02  FB 002F7          CALLS   #2, EDT$$SC_POSCSIF                             ;
                                                58  D4 002FE          CLRL    FIRST_CHAR                                      ; 1032
                                                59  DD 00300 27$:     PUSHL   CHAR                                            ; 1035
                     00000000G  00              01  FB 00302          CALLS   #1, EDT$$FMT_CH                                 ;
                                                5B  D7 00309          DECL    LEN                                             ; 1036
                                                67  11 0030B          BRB     32$                                             ; 1019
                                2D         08   AC  E8 0030D 28$:     BLBS    ERASED, 30$                                     ; 1041
                              FF   8F      0C   A4  91 00311          CMPB    12(R4), #255                                    ;
                                                26  12 00316          BNEQ    30$                                             ;
                  50 00000000G  00 00000000G    00  C3 00318          SUBL3   EDT$$G_SHF, EDT$$G_FMT_LNPOS, R0                ; 1044
                                                50  DD 00324          PUSHL   R0                                              ;
                                                02  18 00326          BGEQ    29$                                             ;
                                                6E  D4 00328          CLRL    (SP)                                            ;
                                   00000000G    00  DD 0032A 29$:     PUSHL   EDT$$G_CS_LNO                                   ;
                     00000000G  00              02  FB 00330          CALLS   #2, EDT$$SC_POSCSIF                             ;
                     00000000G  00              00  FB 00337          CALLS   #0, EDT$$SC_ERATOEOL                            ; 1045
                  50 00000000G  00              01  C3 0033E 30$:     SUBL3   #1, EDT$$G_TI_WID, R0                           ; 1053
                                50 00000000G    00  D1 00346          CMPL    EDT$$G_FMT_LNPOS, R0                            ;
                                                09  18 0034D          BGEQ    31$                                             ;
                                                20  DD 0034F          PUSHL   #32                                             ;
                     00000000G  00              01  FB 00351          CALLS   #1, EDT$$FMT_CH                                 ;
                  7E 00000000G  00              01  C3 00358 31$:     SUBL3   #1, EDT$$G_TI_WID, -(SP)                        ; 1055
                                   00000000G    00  DD 00360          PUSHL   EDT$$G_CS_LNO                                   ;
                     00000000G  00              02  FB 00366          CALLS   #2, EDT$$SC_POSCSIF                             ;
                     00000000G  00              00  FB 0036D          CALLS   #0, EDT$$SC_SHWBLOB                             ; 1056
                                                34  11 00374 32$:     BRB     36$                                             ; 1015
                                30         08   AC  E8 00376 33$:     BLBS    ERASED, 36$                                     ; 1066
                              FF   8F      0C   A4  91 0037A          CMPB    12(R4), #255                                    ;
                                                29  12 0037F          BNEQ    36$                                             ;
                                1F              58  E9 00381          BLBC    FIRST_CHAR, 35$                                 ; 1070
                  50 00000000G  00 00000000G    00  C3 00384          SUBL3   EDT$$G_SHF, EDT$$G_FMT_LNPOS, R0                ;
                                                50  DD 00390          PUSHL   R0                                              ;
                                                02  18 00392          BGEQ    34$                                             ;
                                                6E  D4 00394          CLRL    (SP)                                            ;
                                   00000000G    00  DD 00396 34$:     PUSHL   EDT$$G_CS_LNO                                   ;
                     00000000G  00              02  FB 0039C          CALLS   #2, EDT$$SC_POSCSIF                             ;
                     00000000G  00              00  FB 003A3 35$:     CALLS   #0, EDT$$SC_ERATOEOL                            ; 1072
                                50         0A   A4  9A 003AA 36$:     MOVZBL  10(R4), R0                                      ; 1078
                                51         09   A4  9A 003AE          MOVZBL  9(R4), R1                                       ;
                                50              51  C2 003B2          SUBL2   R1, R0                                          ;
```

```
                                                50  D6 003B5          INCL    R0                                          ;
                                    000000FF    8F                    50  D1 003B7          CMPL    R0, #255                    ;
                                                                      04  15 003BE          BLEQ    37$                         ;
                                                50          FF        8F  9A 003C0          MOVZBL  #255, R0                    ;
                                          0B    A4                    50  90 003C4 37$:     MOVB    R0, 11(R4)              ;
                                          0C    A4          03FF      8F  AA 003C8 38$:     BICW2   #1023, 12(R4)           ; 1080
                                                                          04 003CE          RET                             ; 1081

; Routine Size:  975 bytes,    Routine Base:  _EDT$CODE + 0000

;  512          1082  1
;  513          1083  1 !<BLF/PAGE>
```

```
;  515          1084  1 END                                   ! of module EDT$SCRRLIN
;  516          1085  1
;  517          1086  0 ELUDOM
```

```
;                                PSECT SUMMARY
;
;      Name                      Bytes                         Attributes
;
;  _EDT$CODE                        975  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        48        12          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$:SCRRLIN/OBJ=OBJ$:SCRRLIN MSRC$:SCRRLIN.BLI/UPDATE=(ENH$:SCRRLIN
;       )

; Size:         975 code + 0 data bytes
; Run Time:         00:36.5
; Elapsed Time:     00:43.0
; Lines/CPU Min:     1787
; Lexemes/CPU-Min:   7088
; Memory Used:  246 pages
; Compilation Complete
```

SCRRESET LIS

SCRRLIN LIS

SCRRELPOS LIS

SCRINSERT LIS

SCRNOSCR LIS

SCRNCOL LIS

SCRMCOL LIS

SCRUPDATE LIS

SCRINIT LIS

SCRNEWINS LIS

SCRREGION LIS

SCRREPMOD LIS

SCRREV LIS

SCRNEWDEL LIS

SCRNOREV LIS

SCRMOVETO LIS

SCRINSMOD LIS