C 10

```
EEEEEEEEEEEEEEE  DDDDDDDDDDDD      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  DDDDDDDDDDDD            TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD            TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD            TTT
```

```
EX

Mo
--
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
SY
LB
LI
```

EDT
V04

```
RRRRRRRR      AAAAAA    NN      NN  RRRRRRRR    PPPPPPPP      OOOOOO      SSSSSSSS
RRRRRRRR      AAAAAA    NN      NN  RRRRRRRR    PPPPPPPP      OOOOOO      SSSSSSSS
RR      RR  AA      AA  NN      NN  RR      RR  PP      PP  OO      OO  SS
RR      RR  AA      AA  NN      NN  RR      RR  PP      PP  OO      OO  SS
RR      RR  AA      AA  NNNN    NN  RR      RR  PP      PP  OO      OO  SS
RR      RR  AA      AA  NNNN    NN  RR      RR  PP      PP  OO      OO  SS
RRRRRRRR    AA      AA  NN  NN  NN  RRRRRRRR    PPPPPPPP    OO      OO    SSSSSS
RRRRRRRR    AA      AA  NN  NN  NN  RRRRRRRR    PPPPPPPP    OO      OO    SSSSSS
RR  RR      AAAAAAAAAA  NN    NNNN  RR  RR      PP          OO      OO          SS
RR  RR      AAAAAAAAAA  NN    NNNN  RR  RR      PP          OO      OO          SS
RR    RR    AA      AA  NN      NN  RR    RR    PP          OO      OO          SS      ....
RR    RR    AA      AA  NN      NN  RR    RR    PP          OO      OO          SS      ....
RR      RR  AA      AA  NN      NN  RR      RR  PP            OOOOOO    SSSSSSSS        ....
RR      RR  AA      AA  NN      NN  RR      RR  PP            OOOOOO    SSSSSSSS        ....

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
```

```
0 %TITLE 'EDT$RANRPOS - position to the first line of a range'
0 MODULE EDT$RANRPOS (                          ! Position to the first line of a range
0               IDENT = 'V04-000'                       ! File: RANRPOS.BLI Edit: SMB1017
0               ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:     EDT -- The DEC Standard Editor
1 !
1 ! ABSTRACT:
1 !
1 !       Position to the first line of a range.
1 !
1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
1 !
1 ! AUTHOR: Bob Kushlis, CREATION DATE: February 3, 1978
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  DJS 19-FEB-1981.  This module was created by
1 !       extracting routine RPOS from module RANGE.
1 ! 1-002 - Regularize headers.  JBS 12-Mar-1981
1 ! 1-003 - Change to use the ASSERT macro.  JBS 01-Jun-1981
1 ! 1-004 - Use the new message codes.  JBS 04-Aug-1981
1 ! 1-005 - Use the new PREV_RANGE field for ALL.  JBS 02-Nov-1981
1 ! 1-006 - In THRU, if part of the range is not found the whole
1 !          range fails.  JBS 19-Nov-1981
1 ! 1-007 - Fix any size problems in arithmetic and compares.  SMB 25-Jan-1982
1 ! 1-008 - Remove original line numbers.  SMB 28-Jan-1982
1 ! 1-009 - Add error message and change display for original line numbers.  SMB 6-Feb-1982
1 ! 1-010 - Change the way buffer pos. is saved for AND ranges.  SMB 15-Feb-1982
1 ! 1-011 - Don't change buffer pos if we have /STAY. STS 21-Apr-1982
1 ! 1-012 - Worry about string truncation.  JBS 05-May-1982
```

```
:   58          0058  1 ! 1-013 - Add error checks for incomplete select ranges.  SMB 01-Jul-1982
:   59          0059  1 ! 1-014 - Make sure range ok on "No select range".  SMB 02-Jul-1982
:   60          0060  1 ! 1-015 - Put edt$$rng_retfrst in line. STS 11-Oct-1982
:   61          0061  1 ! 1-016 - Modify to use new compare macro. STS 20-Oct-1982
:   62          0062  1 ! 1-017 - Remove setting of G_TXT_ONSCR with select range.  SMB 13-Dec-1982
:   63          0063  1 !--
:   64          0064  1
```

EDT
V04

```
;   66      0065  1 %SBTTL 'Declarations'
;   67      0066  1 !
;   68      0067  1 ! TABLE OF CONTENTS:
;   69      0068  1 !
;   70      0069  1
;   71      0070  1 REQUIRE 'EDTSRC:TRAROUNAM';
;   72      0509  1
;   73      0510  1 FORWARD ROUTINE
;   74      0511  1     EDT$$RNG_REPOS;
;   75      0512  1
;   76      0513  1 !
;   77      0514  1 ! INCLUDE FILES:
;   78      0515  1 !
;   79      0516  1
;   80      0517  1 REQUIRE 'EDTSRC:EDTREQ';
;   81      0652  1
;   82      0653  1 !
;   83      0654  1 ! MACROS:
;   84      0655  1 !
;   85      0656  1 !       NONE
;   86      0657  1 !
;   87      0658  1 ! EQUATED SYMBOLS:
;   88      0659  1 !
;   89      0660  1 !       NONE
;   90      0661  1 !
;   91      0662  1 ! OWN STORAGE:
;   92      0663  1 !
;   93      0664  1 !       NONE
;   94      0665  1 !
;   95      0666  1 ! EXTERNAL REFERENCES:
;   96      0667  1 !
;   97      0668  1 !       In the routine
```

```
1 %SBTTL 'EDT$$RNG_REPOS  - position to the first line of a range'
1
1 GLOBAL ROUTINE EDT$$RNG_REPOS (                       ! Position to the first line of a range
1     RANGE                                             ! Range to position to
1     ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine positions to the first line of a range.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !  RANGE                The range node.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       EDT$$A_SEL_BUF
1 !       EDT$$L_SEL_LN
1 !       EDT$$A_CUR_BUF
1 !       EDT$$A_PRV_BUF
1 !       EDT$$T_LN_BUF
1 !       EDT$$Z_RNG_ORIGPOS
1 !       EDT$$A_SEL_POS
1 !       EDT$$A_WK_LN
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       EDT$$G_RNG_MORELN
1 !       EDT$$G_RNG_NOOFLN
1 !       EDT$$L_RNG_F[illegible]
1 !       EDT$$A_PRV_BUF
1 !       EDT$$A_CUR_BUF
1 !       EDT$$A_SEL_BUF
1 !       EDT$$Z_RNG_SAVPOS
1 !       EDT$$Z_RNG_CURRNG
1 !       EDT$$Z_RNG_ORIGPOS
1 !
1 ! ROUTINE VALUE:
1 !
1 !       0 = no such line
1 !       1 = positioned successfully
1 !
1 ! SIDE EFFECTS:
1 !
1 !       Current text buffer is re-positioned
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL ROUTINE
2         EDT$$FMT_MSG,
2         EDT$$RNG_REPOS,
2         EDT$$G_EXE_SBITS,
2         EDT$$FND_STR,
2         EDT$$FND_BUF,
                EDT$$SET_SEASTR,
                EDT$$WF_BOT,
                EDT$$RD_PRVLN,
                EDT$$RD_CURLN,
                EDT$$RD_NXTLN,
                EDT$$TOP_BUF,
                EDT$$LOC_LN;

            EXTERNAL
                EDT$$G_CUR_COL,                     ! Current cursor column
                EDT$$L_LNO5 : LN_BLOCK,             ! 10**5
                EDT$$A_SEL_BUF,                     ! Select buffer
                EDT$$L_SEL_LN,                      ! Select line
                EDT$$A_PRV_BUF,                     ! The previous TBCB for LAST range.
                EDT$$A_CUR_BUF : REF TBCB_BLOCK,    ! The current text buffer
                EDT$$T_LN_BUF,                      ! Line buffer
                EDT$$G_RNG_MORELN,                  ! Used by EDT$$NXT_LNRNG  to indicate more lines.
                EDT$$G_RNG_FRSTLN,                  ! Indicates first line in a range.
                EDT$$G_RNG_NOOFLN,                  ! Count of number of lines in a range.
                EDT$$Z_RNG_ORIGPOS : POS_BLOCK,     ! To save the position at start of command.
                EDT$$L_RNG_EOL : LN_BLOCK,          ! The line number at which this range ends
                EDT$$Z_RNG_CURRNG : REF NODE_BLOCK, ! The current range node
                EDT$$Z_RNG_SAVPOS : POS_BLOCK,      ! To save the beginning of range
                EDT$$A_SEL_POS,                     ! Select position
                EDT$$A_WK_LN : REF LIN_BLOCK;       ! The current line pointer.

            MESSAGES ((NOSELRAN, NOSUCHLIN, STRNOTFND, INSMEM, NOORIGNUM, INVSTP, I' SRAN));

            MAP
                RANGE : REF NODE_BLOCK;

        !+
        ! Make sure the parameter is a range node.
        !-
            ASSERT (.RANGE [NODE_TYPE] EQL RANGE_NODE);
        !+
        ! Initialize for first line of range.
        !-
            EDT$$G_RNG_MORELN = 1;
            EDT$$G_RNG_NOOFLN = 0;
        !+
        ! Case on range type.
        !-

            CASE .RANGE [RAN_TYPE] FROM 0 TO NUM_RAN OF
                SET
        !+
        ! If range is '.' or REST, the original position is the first line.
        !-

                [RAN_DOT, RAN_NULL, RAN_REST] :         ! Use the current position
                    BEGIN
                    EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_ORIGPOS, .EDT$$A_CUR_BUF);
                    EDT$$RD_CURLN ();
                    END;
        !+
        ! Line number range.  Find the line.
```

```
2 !-
2
2         [RAN_NUMBER] :
3             BEGIN
3             EDT$$G_RNG_MORELN = EDT$$LOC_LN (RANGE [RAN_VAL]);
2             END;
2 !+
2 ! Range is BEFORE.  Try going back a line to find out if there are any,
2 ! then save that number as the last line number in the range.
2 !-
2
2         [RAN_BEFORE] :
3             BEGIN
3             EDT$$G_RNG_MORELN = EDT$$RD_PRVLN ();
3             MOVELINE (EDT$$A_WK_LN [LIN_NUM], EDT$$L_RNG_EOL);
3             EDT$$TOP_BUF ();
2             END;
2 !+
2 ! Range is WHOLE or BEGIN.  Position to the first line in the buffer.
2 !-
2
2         [RAN_WHOLE, RAN_BEGIN] :
2             EDT$$TOP_BUF ();
2 !+
2 ! Range is SELECT.  Check to see if there is a select range in effect.
2 ! If it not in the current buffer, then switch to the buffer which has
2 ! the select.  Then position to the first line of the select range.
2 !-
2
2         [RAN_SELECT] :
3             BEGIN
3
3             LOCAL
3                 TMPRAN : LN_BLOCK;
3
4             IF (.EDT$$A_SEL_BUF EQLA 0)
3             THEN
4                 BEGIN
4                 EDT$$FMT_MSG (EDT$_NOSELRAN);
4                 EDT$$G_RNG_MORELN = 0;
4                 EDT$$Z_RNG_CURRNG = .RANGE;
4
4                 IF (.EDT$$G_RNG_FRSTLN) THEN EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_SAVPOS);
4
4                 RETURN (0);
4                 END
3             ELSE
4                 BEGIN
4
5                 IF (.EDT$$A_SEL_BUF NEQA .EDT$$A_CUR_BUF)
4                 THEN
5                     BEGIN
5                     EDT$$A_PRV_BUF = .EDT$$A_CUR_BUF;
5                     EDT$$A_CUR_BUF = .EDT$$A_SEL_BUF;
5                     EDT$$RD_CURLN ();
4                     END;
4
```

```
4                       SUBLINE (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_SEL_LN, TMPRAN);
4                       RANGE [RAN_VAL] = .TMPRAN;        !Only want to move a word
4  !+
4  ! For line mode commands the select range must be in whole lines
4  !-
4
5                       IF (.EDT$$G_CUR_COL NEQ 0) OR (.TMPRAN EQL 0)
4                       THEN
5                           BEGIN
5                           EDT$$FMT_MSG (EDT$_INVSRAN);
5                           EDT$$G_RNG_MORELN = 0;
5                           EDT$$Z_RNG_CURRNG = .RANGE;
5
5                           IF (.EDT$$G_RNG_FRSTLN) THEN EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_SAVPOS)
5
5                           RETURN (0);
5                           END
4                       ELSE
4
5                           IF (.RANGE [RAN_VAL] LSS 0)
4                           THEN
5                               BEGIN
5                               RANGE [RAN_VAL] = -.RANGE [RAN_VAL];
5
5                               DECR I FROM .RANGE [RAN_VAL] - 1 TO 0 DO
5                                   EDT$$RD_PRVLN ();
5
4                               END;
4
5                       IF CH$PTR_NEQ (.EDT$$A_SEL_POS, CH$PTR (EDT$$T_LN_BUF))
4                       THEN
4                           RANGE [RAN_VAL] = .RANGE [RAN_VAL] + 1;
4
4                       EDT$$A_SEL_BUF = 0;
3                       END;
3
2                   END;
2  !+
2  ! Range is END.  Position to the end of the buffer.
2  !-
2
2           [RAN_END] :
3               BEGIN
3               EDT$$WF_BOT ();
3               EDT$$G_RNG_MORELN = 0;
2               END;
2  !+
2  ! Range is ORIGINAL.  This is no longer a feature of EDT, so print a warning
2  ! and type the real line number corresponding to the line input
2  !-
2
2           [RAN_ORIG] :
3               BEGIN
3
3               LOCAL
3                   LINNO : LN_BLOCK;
3
```

```
3              EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_SAVPOS);
3              EDT$$FMT_MSG (EDT$_NOORIGNUM);
3              MULTLINE (EDT$$L_LNO5, RANGE [RAN_VAL], LINNO);
3
3              WHILE NOT (LINNOEQL (EDT$$A_WK_LN [LIN_NUM], LINNO)) DO
3
4                  IF ( NOT EDT$$RD_NXTLN ())
3                  THEN
4                      BEGIN
4                      EDT$$G_RNG_MORELN = 0;
4                      EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_SAVPOS, .EDT$$A_CUR_BUF);
4                      EDT$$RD_CURLN ();
4                      EDT$$FMT_MSG (EDT$_NOSUCHLIN);
4                      RETURN (0);
4                      END;
3
2              END;
2  !+
2  ! Range is MINUS.  Make a recursive call to position to the range then
2  ! move back the specified number of lines.
2  !-
2
2          [RAN_MINUS] :
3              BEGIN
3
3              IF EDT$$RNG_REPOS (.RANGE [SUB_RANGE])
3              THEN
4                  BEGIN
4                  EDT$$G_RNG_MORELN = 1;
4
4                  INCR I FROM 1 TO .RANGE [RAN_VAL] DO
4
5                      IF ( NOT EDT$$RD_PRVLN ())
4                      THEN
5                          BEGIN
5                          EDT$$G_RNG_MORELN = 0;
5                          EXITLOOP;
5                          END
5
4                  END
3              ELSE
3                  EDT$$G_RNG_MORELN = 0;
3
2              END;
2  !+
2  ! Range is Plus. Make a recursive call to position to the range then
2  ! move forward the specified number of lines.
2  !-
2
2          [RAN_PLUS] :
3              BEGIN
3
3              IF EDT$$RNG_REPOS (.RANGE [SUB_RANGE])
3              THEN
4                  BEGIN
4                  EDT$$G_RNG_MORELN = 1;
4
```

```
4                  INCR I FROM 1 TO .RANGE [RAN_VAL] DO
4
5                      IF ( NOT EDT$$RD_NXTLN ())
4                      THEN
5                          BEGIN
5                          EDT$$G_RNG_MORELN = 0;
5                          EXITLOOP;
5                          END
5
4                  END
3              ELSE
3                  EDT$$G_RNG_MORELN = 0;
3
2              END;
2  !+
2  ! Range is FOR or #.  Just position to the original range.
2  !-
2
2          [RAN_FOR] :
2              EDT$$G_RNG_MORELN = EDT$$RNG_REPOS (.RANGE [SUB_RANGE]);
2  !+
2  ! Range is a search string.  Save the current position, search for the
2  ! string in the specified direction.  If the string is not found, then
2  ! reposition and return failure.
2  !-
2
2          [RAN_STR, RAN_MINSTR] :
3              BEGIN
3
3              LOCAL
3                  FIND_STATUS;
3
3              EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_SAVPOS);
3              FIND_STATUS = EDT$$FND_STR (.RANGE [STR_PNT], .RANGE [RAN_VAL], (.RANGE [RAN_TYPE] EQL RAN_STR))
3
3              CASE .FIND_STATUS FROM 0 TO 2 OF
3                  SET
3
3                  [0] :                               ! String not found
4                      BEGIN
4                      EDT$$FMT_MSG (EDT$_STRNOTFND);
4                      EDT$$G_RNG_MORELN = 0;
4                      EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_SAVPOS, .EDT$$A_CUR_BUF);
4                      EDT$$RD_CURLN ();
4                      RETURN (0);
3                      END;
3
3                  [1] :                               ! String found
4                      BEGIN
4                      0
3                      END;
3
3                  [2] :                               ! String invalid
4                      BEGIN
4                      EDT$$FMT_MSG (EDT$_INVSTR);
4                      EDT$$G_RNG_MORELN = 0;
4                      EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_SAVPOS, .EDT$$A_CUR_BUF);
```

```
4                       EDT$$RD_CURLN ();
4                       RETURN (0);
3                       END;
3                   TES;
3
2               END;
2  !+
2  ! Range is THRU or : .  Position to the first range, then find the line
2  ! number of the end of the range.  Special case when the end range is a
2  ! line number or END.  Otherwise, save the current position and position to
2  ! the end to get the end range.  If either the first or last cannot be
2  ! found the whole range fails.
2  !-
2
2           [RAN_THRU] :
3               BEGIN
3
3               LOCAL
3                   EDT$$SAV_BUFPOS : POS_BLOCK;
3
3               BIND
3                   END_RANGE = .RANGE [RANGE2] : NODE_BLOCK;
3
3               BIND
3                   START_RANGE = .RANGE [RANGE1] : NODE_BLOCK;
3
3               IF ( NOT EDT$$RNG_REPOS (START_RANGE)) THEN RETURN (0);
3
4               IF (.END_RANGE [RAN_TYPE] EQL RAN_NUMBER)    !
3               THEN
3                   EDT$$CPY_MEM (LN_SIZE, END_RANGE [RAN_VAL], EDT$$L_RNG_EOL)
3               ELSE
3
4                   IF (.END_RANGE [RAN_TYPE] EQL RAN_END)
3                   THEN
3                       RANGE [RAN_TYPE] = RAN_REST
3                   ELSE
4                       BEGIN
4
4                       LOCAL
4                           END_FOUND;
4
4                       EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$SAV_BUFPOS);
4                       END_FOUND = EDT$$RNG_REPOS (END_RANGE);
4
4                       IF .END_FOUND THEN MOVELINE (EDT$$A_WK_LN [LIN_NUM], EDT$$L_RNG_EOL);
4
4                       EDT$$CPY_MEM (POS_SIZE, EDT$$SAV_BUFPOS, .EDT$$A_CUR_BUF);
4                       EDT$$RD_CURLN ();
4
4                       IF ( NOT .END_FOUND) THEN RETURN (0);
4
3                       END;
3
3               EDT$$G_RNG_MORELN = 1;
2               END;
2  !+
2 ! Range is ALL range.  Look at the range to which ALL applies.  If it is
2 ! null, assume WHOLE.  Position to that range, then set up the ALL string
2 ! as the current search string.  Note that EDT$$Z_RNG_CURRNG  will be not
2 ! that ALL range but its subordinate.  PREV_RANGE will point back to the
2 ! ALL range.
2 !-
2
2         [RAN_ALL] :
3             BEGIN
3
3             LOCAL
3                 SUB_RAN : REF NODE_BLOCK;
3
3             SUB_RAN = .RANGE [NEXT_RANGE];
3             ASSERT (.SUB_RAN [PREV_RANGE] EQLA .RANGE);
3
3             IF (.SUB_RAN [RAN_TYPE] EQL RAN_NULL) THEN SUB_RAN [RAN_TYPE] = RAN_WHOLE;
3
3             EDT$$G_RNG_FRSTLN = 1;
3             EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
3
3             IF ( NOT EDT$$RNG_REPOS (.RANGE [NEXT_RANGE])) THEN RETURN (0);
3
4             IF ( NOT EDT$$SET_SEASTR (.RANGE [STR_PNT], .RANGE [RAN_VAL]))
3             THEN
4                 BEGIN
4                 EDT$$FMT_MSG (EDT$_INVSTR);
4                 RETURN (0);
3                 END;
3
3             RETURN (1);
3             END;
2 !+
2 ! The range contains a buffer specification.  Switch to the new buffer, then
2 ! position to the range within the buffer.  If the range within the buffer
2 ! was null, assume WHOLE.
2 !-
2
2         [RAN_BUFFER] :
3             BEGIN
3
3             LOCAL
3                 SUB_RANGE : REF NODE_BLOCK;
3
3             IF EDT$$FND_BUF (.RANGE [BUF_NAME], .RANGE [BUF_LEN])
3             THEN
4                 BEGIN
4                 SUB_RANGE = .RANGE [RANGE1];
4
5                 IF ( NOT .EDT$$G_EXE_SBITS<OPB_STAY>)    !
4                 THEN
4                     EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
4
4                 EDT$$TOP_BUF ();
4
4                 IF (.SUB_RANGE [RAN_TYPE] EQL RAN_NULL) THEN SUB_RANGE [RAN_TYPE] = RAN_WHOLE;
4
```

```
;     4                    RETURN (EDT$$RNG_REPOS (.SUB_RANGE));
;     4                    END
;     3               ELSE
;     4                    BEGIN
;     4                    EDT$$FMT_MSG (EDT$_INSMEM);
;     4                    RETURN (0);
;     4                    END
;     4
;     2               END;
;     2    !+
;     2    ! Range is LAST.  Switch back to the buffer pointed to by EDT$$A_PRV_BUF
;     2    ! at it's current position.
;     2    !-
;     2
;     2          [RAN_LAST] :
;     3               BEGIN
;     3
;     3               LOCAL
;     3                    TEMP;
;     3
;     3               TEMP = .EDT$$A_CUR_BUF;
;     3               EDT$$A_CUR_BUF = .EDT$$A_PRV_BUF;
;     3               EDT$$A_PRV_BUF = .TEMP;
;     3               EDT$$RD_CURLN ();
;     2               END;
;     2
;     2          [INRANGE, OUTRANGE] :
;     2               ASSERT (0);
;     2          TES;
;     2
;     2    !+
;     2    ! Save the range node and the current position.
;     2    !-
;     2       EDT$$Z_RNG_CURRNG = .RANGE;
;     2
;     2       IF (.EDT$$G_RNG_FRSTLN) THEN EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_SAVPOS);
;     2
;     2       RETURN (1);
;     1       END;                                          ! of routine EDT$$RNG_REPOS


                                                            .TITLE  EDT$RANRPOS EDT$RANRPOS - position to the first
;                                                                               line of a r
                                                            .IDENT  \V04-000\

                                                            .EXTRN  EDT$$FMT_MSG, EDT$$RNG_REPOS
                                                            .EXTRN  EDT$$G_EXE_SBITS
                                                            .EXTRN  EDT$$FND_STR, EDT$$FND_BUF
                                                            .EXTRN  EDT$$SET_SEASTR
                                                            .EXTRN  EDT$$WF_BOT, EDT$$RD_PRVLN
                                                            .EXTRN  EDT$$RD_CURLN, EDT$$RD_NXTLN
                                                            .EXTRN  EDT$$TOP_BUF, EDT$$LOC_LN
                                                            .EXTRN  EDT$$G_COR_COL, EDT$$L_LNO5
                                                            .EXTRN  EDT$$A_SEL_BUF, EDT$$L_SEL_LN
                                                            .EXTRN  EDT$$A_PRV_BUF, EDT$$A_CUR_BUF
                                                            .EXTRN  EDT$$T_LN_BUF, EDT$$G_RNG_MORELN
                                                            .EXTRN  EDT$$G_RNG_FRSTLN
```

```
                                                 J  9
EDT$RANRPOS      EDT$RANRPOS - position to the first line of a r 16-Sep-1984 01:26:05     VAX-11 Bliss-32 V4.0-742            Page  13
V04-000          EDT$$RNG_REPOS  - position to the first line of 14-Sep-1984 12:24:19     DISK$VMSMASTER:[EDT.SRC]RANRPOS.BLI;1    (3)

                                                           .EXTRN  EDT$$G_RNG_NOOFLN
                                                           .EXTRN  EDT$$Z_RNG_ORIGPOS
                                                           .EXTRN  EDT$$L_RNG_EOL, EDT$$Z_RNG_CURRNG
                                                           .EXTRN  EDT$$Z_RNG_SAVPOS
                                                           .EXTRN  EDT$$A_SEL_POS, EDT$$A_WK_LN
                                                           .EXTRN  EDT$_NOSELRAN, EDT$_NOSUCHLIN
                                                           .EXTRN  EDT$_STRNOTFND, EDT$_INSMEM
                                                           .EXTRN  EDT$_NOORIGNUM, EDT$_INVSTR
                                                           .EXTRN  EDT$_INVSRAN, EDT$$INTER_ERR

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

                                          0FFC 00000       .ENTRY  EDT$$RNG_REPOS, Save R2,R3,R4,R5,R6,R7,R8,- ; 0671
                                                                   R9,R10,R11                                  ;
                           5B 00000000G  00   9E 00002     MOVAB   EDT$$Z_RNG_SAVPOS, R11                      ;
                           5A 00000000G  00   9E 00009     MOVAB   EDT$$RNG_REPOS, R10                         ;
                           59 00000000G  00   9E 00010     MOVAB   EDT$$G_RNG_MORELN, R9                       ;
                           58 00000000G  00   9E 00017     MOVAB   EDT$$A_CUR_BUF, R8                          ;
                           5E            18   C2 0001E     SUBL2   #24, SP                                     ;
                           56       04   AC   D0 00021     MOVL    RANGE, R6                                   ; 0760
                           02            66   91 00025     CMPB    (R6), #2                                    ;
                                         07   13 00028     BEQL    1$                                          ;
                 00000000G 00            00   FB 0002A     CALLS   #0, EDT$$INTER_ERR                          ;
                           69            01   D0 00031 1$: MOVL    #1, EDT$$G_RNG_MORELN                       ; 0764
                              00000000G  00   D4 00034     CLRL    EDT$$G_RNG_NOOFLN                           ; 0765
                  14       00       01   A6   8F 0003A     CASEB   1(R6), #0, #20                              ; 0770
      0203      0033      0041         0033      0003F 2$: .WORD   4$-2$,-                                     ;
      002A      0135      012B         0068      00047             5$-2$,-                                     ;
      0068      0033      004E         034F      0004F             4$-2$,-                                     ;
      01B1      01D2      0305         0072      00057             37$-2$,-                                    ;
      02B8      0203      0253         01F7      0005F             7$-2$,-                                     ;
                                       002A      00067             21$-2$,-                                    ;
                                                                   22$-2$,-                                    ;
                                                                   3$-2$,-                                     ;
                                                                   58$-2$,-                                    ;
                                                                   6$-2$,-                                     ;
                                                                   4$-2$,-                                     ;
                                                                   7$-2$,-                                     ;
                                                                   9$-2$,-                                     ;
                                                                   52$-2$,-                                    ;
                                                                   30$-2$,-                                    ;
                                                                   27$-2$,-                                    ;
                                                                   34$-2$,-                                    ;
                                                                   44$-2$,-                                    ;
                                                                   37$-2$,-                                    ;
                                                                   49$-2$,-                                    ;
                                                                   3$-2$                                       ;
                 00000000G 00            00   FB 00069 3$: CALLS   #0, EDT$$INTER_ERR                          ; 1152
                                         3C   11 00070     BRB     8$                                          ; 0770
                           50            68   D0 00072 4$: MOVL    EDT$$A_CUR_BUF, R0                          ; 0778
              60 00000000G 00            0E   28 00075     MOVC3   #14, EDT$$Z_RNG_ORIGPOS, (R0)               ;
                                       031F   31 0007D     BRW     59$                                         ; 0779
                                    04   A6   9F 00080 5$: PUSHAB  4(R6)                                       ; 0787
                 00000000G 00            01   FB 00083     CALLS   #1, EDT$$LOC_LN                             ;
                                       01AF   31 0008A     BRW     35$                                         ;
                 00000000G 00            00   FB 0008D 6$: CALLS   #0, EDT$$RD_PRVLN                           ; 0796
                           69            50   D0 00094     MOVL    R0, EDT$$G_RNG_MORELN                       ;
```

```
                                   50 00000000G  00  D0 00097          MOVL     EDT$$A_WK_LN, R0                                ; 0797
           00000000G  00      01   A0        06  28 0009E          MOVC3    #6, 1(R0), EDT$$L_RNG_EOL                       ;
                  00000000G   00             00  FB 000A7  7$:     CALLS    #0, EDT$$TOP_BUF                                ; 0805
                                           02F5  31 000AE  8$:     BRW      60$                                             ;
                                   52 00000000G  00  D0 000B1  9$:     MOVL     EDT$$A_SEL_BUF, R2                              ; 0818
                                             08  12 000B8          BNEQ     10$                                             ;
                                      00000000G  8F  DD 000BA          PUSHL    #EDT$_NOSELRAN                                  ; 0821
                                             56  11 000C0          BRB      13$                                             ;
                                   50        68  D0 000C2 10$:     MOVL     EDT$$A_CUR_BUF, R0                              ; 0832
                                   50        52  D1 000C5          CMPL     R2, R0                                          ;
                                             11  13 000C8          BEQL     11$                                             ;
                  00000000G   00             50  D0 000CA          MOVL     R0, EDT$$A_PRV_BUF                              ; 0835
                              68             52  D0 000D1          MOVL     R2, EDT$$A_CUR_BUF                              ; 0836
                  00000000G   00             00  FB 000D4          CALLS    #0, EDT$$RD_CURLN                               ; 0837
                              51       16    AE  B0 000DB 11$:     MOVW     UPPER_WORD, SAVE                                ; 0840
                              50             68  D0 000DF          MOVL     EDT$$A_CUR_BUF, R0                              ;
            10    AE 00000000G   00       06    A0  C3 000E2          SUBL3    6(R0), EDT$$L_SEL_LN, TMPRAN                    ;
                         14   AE 00000000G   00  D0 000EC          MOVL     EDT$$L_SEL_LN+4, TMPRAN                         ;
                         14   AE       0A    A0  D9 000F4          SBWC     10(R0), TMPRAN                                  ;
                         16   AE             51  B0 000F9          MOVW     SAVE, UPPER_WORD                                ;
                              57       04    A6  9E 000FD          MOVAB    4(R6), R7                                       ; 0841
                              67       10    AE  D0 00101          MOVL     TMPRAN, (R7)                                    ;
                                      00000000G  00  D5 00105          TSTL     EDT$$G_CUR_COL                                  ; 0846
                                             05  12 0010B          BNEQ     12$                                             ;
                                       10    AE  D5 0010D          TSTL     TMPRAN                                          ;
                                             27  12 00110          BNEQ     15$                                             ;
                                      00000000G  8F  DD 00112 12$:     PUSHL    #EDT$_INVSRAN                                   ; 0849
                  00000000G   00             01  FB 00118 13$:     CALLS    #1, EDT$$FMT_MSG                                ;
                                             69  D4 0011F          CLRL     EDT$$G_RNG_MORELN                               ; 0850
                  00000000G   00             56  D0 00121          MOVL     R6, EDT$$Z_RNG_CURRNG                           ; 0851
                              07 00000000G   00  E9 00128          BLBC     EDT$$G_RNG_FRSTLN, 14$                          ; 0853
                              50             68  D0 0012F          MOVL     EDT$$A_CUR_BUF, R0                              ;
                      6B      60             0E  28 00132          MOVC3    #14, (R0), EDT$$Z_RNG_SAVPOS                    ;
                                           0286  31 00136 14$:     BRW      62$                                             ; 0855
                                             67  D5 00139 15$:     TSTL     (R7)                                            ; 0859
                                             12  18 0013B          BGEQ     18$                                             ;
                              67             67  CE 0013D          MNEGL    (R7), (R7)                                      ; 0862
                              52             67  D0 00140          MOVL     (R7), I                                         ; 0864
                                             07  11 00143          BRB      17$                                             ;
                  00000000G   00             00  FB 00145 16$:     CALLS    #0, EDT$$RD_PRVLN                               ; 0865
                              F6             52  F4 0014C 17$:     SOBGEQ   I, 16$                                          ;
                              50 00000000G   00  9E 0014F 18$:     MOVAB    EDT$$T_LN_BUF, R0                               ; 0869
                              50 00000000G   00  D1 00156          CMPL     EDT$$A_SEL_POS, R0                              ;
                                             02  13 0015D          BEQL     19$                                             ;
                                             67  D6 0015F          INCL     (R7)                                            ; 0871
                                      00000000G  00  D4 00161 19$:     CLRL     EDT$$A_SEL_BUF                                  ; 0873
                                           023C  31 00167 20$:     BRW      60$                                             ; 0770
                  00000000G   00             00  FB 0016A 21$:     CALLS    #0, EDT$$WF_BOT                                 ; 0883
                                           00BE  31 00171          BRW      33$                                             ; 0884
                              50             68  D0 00174 22$:     MOVL     EDT$$A_CUR_BUF, R0                              ; 0897
                      6B      60             0E  28 00177          MOVC3    #14, (R0), EDT$$Z_RNG_SAVPOS                    ;
                                      00000000G  8F  DD 0017B          PUSHL    #EDT$_NOORIGNUM                                 ; 0898
                  00000000G   00             01  FB 00181          CALLS    #1, EDT$$FMT_MSG                                ;
                         08   AE       04    A6  D0 00188          MOVL     4(R6), M2                                       ; 0899
                         0C   AE       08    A6  3C 0018D          MOVZWL   8(R6), M2+4                                     ;
                                             6E  7C 00192          CLRQ     P                                               ;
                              50             10  D0 00194          MOVL     #16, I                                          ;
```

```
        6E              6E              01  79 00197 23$:      ASHQ    #1, P, P                              ;
        09 00000000G    00              50  E1 0019B           BBC     I, M1, 24$                            ;
                        6E       08     AE  C0 001A3           ADDL2   M2, P                                 ;
                04      AE       0C     AE  D8 001A7           ADWC    M2, P                                 ;
                        E8              50  F4 001AC 24$:      SOBGEQ  I, 23$                                ;
                10      AE              6E  D0 001AF           MOVL    P, LINNO                              ;
                14      AE       04     AE  B0 001B3           MOVW    P+4, LINNO+4                          ;
        50 00000000G    00              01  C1 001B8 25$:      ADDL3   #1, EDT$$A_WK_LN, R0                  ; 0901
                10      AE              60  D1 001C0           CMPL    (R0), LOW_2                           ;
                                        07  12 001C4           BNEQ    26$                                   ;
                14      AE       04     A0  B1 001C6           CMPW    4(R0), HIGH_2                         ;
                                        9A  13 001CB           BEQL    20$                                   ;
           00000000G    00              00  FB 001CD 26$:      CALLS   #0, EDT$$RD_NXTLN                     ; 0903
                        E1              50  E8 001D4           BLBS    R0, 25$                               ;
                                        69  D4 001D7           CLRL    EDT$$G_RNG_MORELN                     ; 0906
                        50              68  D0 001D9           MOVL    EDT$$A_CUR_BUF, R0                    ; 0907
        60              6B              0E  28 001DC           MOVC3   #14, EDT$$Z_RNG_SAVPOS, (R0)          ;
           00000000G    00              00  FB 001E0           CALLS   #0, EDT$$RD_CURLN                     ; 0908
                             00000000G  8F  DD 001E7           PUSHL   #EDT$_NOSUCHLIN                       ; 0909
                                     0195  31 001ED           BRW     56$                                   ;
                                 08     A6  DD 001F0 27$:      PUSHL   8(R6)                                 ; 0922
                        6A              01  FB 001F3           CALLS   #1, EDT$$RNG_REPOS                    ;
                        39              50  E9 001F6           BLBC    R0, 33$                               ;
                        69              01  D0 001F9           MOVL    #1, EDT$$G_RNG_MORELN                 ; 0925
                                        52  D4 001FC           CLRL    I                                     ; 0927
                                        0A  11 001FE           BRB     29$                                   ;
           00000000G    00              00  FB 00200 28$:      CALLS   #0, EDT$$RD_PRVLN                     ; 0929
                        28              50  E9 00207           BLBC    R0, 33$                               ;
        F1              52       04     A6  F3 0020A 29$:      AOBLEQ  4(R6), I, 28$                         ;
                                        2E  11 0020F           BRB     36$                                   ; 0927
                                 08     A6  DD 00211 30$:      PUSHL   8(R6)                                 ; 0949
                        6A              01  FB 00214           CALLS   #1, EDT$$RNG_REPOS                    ;
                        18              50  E9 00217           BLBC    R0, 33$                               ;
                        69              01  D0 0021A           MOVL    #1, EDT$$G_RNG_MORELN                 ; 0952
                                        52  D4 0021D           CLRL    I                                     ; 0954
                                        0A  11 0021F           BRB     32$                                   ;
           00000000G    00              00  FB 00221 31$:      CALLS   #0, EDT$$RD_NXTLN                     ; 0956
                        07              50  E9 00228           BLBC    R0, 33$                               ;
        F1              52       04     A6  F3 0022B 32$:      AOBLEQ  4(R6), I, 31$                         ;
                                        0D  11 00230           BRB     36$                                   ; 0954
                                        69  D4 00232 33$:      CLRL    EDT$$G_RNG_MORELN                     ; 0965
                                        09  11 00234           BRB     36$                                   ; 0770
                                 08     A6  DD 00236 34$:      PUSHL   8(R6)                                 ; 0973
                        6A              01  FB 00239           CALLS   #1, EDT$$RNG_REPOS                    ;
                        69              50  D0 0023C 35$:      MOVL    R0, EDT$$G_RNG_MORELN                 ;
                                     0164  31 0023F 36$:      BRW     60$                                   ;
                        50              68  D0 00242 37$:      MOVL    EDT$$A_CUR_BUF, R0                    ; 0986
        6B              60              0E  28 00245           MOVC3   #14, (R0), EDT$$Z_RNG_SAVPOS          ;
                                        7E  D4 00249           CLRL    -(SP)                                 ; 0987
                        03       01     A6  91 0024B           CMPB    1(R6), #3                             ;
                                        02  12 0024F           BNEQ    38$                                   ;
                                        6E  D6 00251           INCL    (SP)                                  ;
                                 04     A6  DD 00253 38$:      PUSHL   4(R6)                                 ;
                                 08     A6  DD 00256           PUSHL   8(R6)                                 ;
           00000000G    00              03  FB 00259           CALLS   #3, EDT$$FND_STR                      ;
        02              00              50  CF 00260           CASEL   FIND_STATUS, #0, #2                   ; 0989
      000E            0142            0006     00264 39$:      .WORD   40$-39$,-                             ;
```

```
                                                                 60$-39$,-                                 ;
                                                                 41$-39$                                   ;
                          00000000G  8F  DD 0026A 40$:  PUSHL   #EDT$_STRNOTFND                            ; 0994
                                     06  11 00270       BRB     42$                                        ;
                          00000000G  8F  DD 00272 41$:  PUSHL   #EDT$_INVSTR                               ; 1008
               00000000G  00         01  FB 00278 42$:  CALLS   #1, EDT$$FMT_MSG                           ;
                                     69  D4 0027F       CLRL    EDT$$G_RNG_MORELN                          ; 1009
                          50         68  D0 00281       MOVL    EDT$$A_CUR_BUF, R0                         ; 1010
                    60    6B         0E  28 00284       MOVC3   #14, EDT$$Z_RNG_SAVPOS, (R0)               ;
               00000000G  00         00  FB 00288       CALLS   #0, EDT$$RD_CURLN                          ; 1011
                                   012D  31 0028F 43$:  BRW     62$                                        ; 1012
                          57     08  A6  D0 00292 44$:  MOVL    8(R6), R7                                  ; 1032
                                 04  A6  DD 00296       PUSHL   4(R6)                                      ; 1037
                          6A         01  FB 00299       CALLS   #1, EDT$$RNG_REPOS                         ;
                          F0         50  E9 0029C       BLBC    R0, 43$                                    ;
                          01     01  A7  91 0029F       CMPB    1(R7), #1                                  ; 1039
                                     0B  12 002A3       BNEQ    45$                                        ;
      00000000G  00      04  A7          06  28 002A5       MOVC3   #6, 4(R7), EDT$$L_RNG_EOL                  ; 1041
                                     41  11 002AE       BRB     48$                                        ;
                          05     01  A7  91 002B0 45$:  CMPB    1(R7), #5                                  ; 1044
                                     06  12 002B4       BNEQ    46$                                        ;
                     01   A6         0A  90 002B6       MOVB    #10, 1(R6)                                 ; 1046
                                     35  11 002BA       BRB     48$                                        ;
                          50         68  D0 002BC 46$:  MOVL    EDT$$A_CUR_BUF, R0                         ; 1053
           08   AE        60         0E  28 002BF       MOVC3   #14, (R0), EDT$$SAV_BUFPOS                 ;
                                     57  DD 002C4       PUSHL   R7                                         ; 1054
                          6A         01  FB 002C6       CALLS   #1, EDT$$RNG_REPOS                         ;
                          57         50  D0 002C9       MOVL    R0, END_FOUND                              ;
                          10         57  E9 002CC       BLBC    END_FOUND, 47$                             ; 1056
                          50 00000000G  00  D0 002CF       MOVL    EDT$$A_WK_LN, R0                           ;
      00000000G  00      01  A0          06  28 002D6       MOVC3   #6, 1(R0), EDT$$L_RNG_EOL                  ;
                          50         68  D0 002DF 47$:  MOVL    EDT$$A_CUR_BUF, R0                         ; 1058
                60       08  AE         0E  28 002E2       MOVC3   #14, EDT$$SAV_BUFPOS, (R0)                 ;
               00000000G  00         00  FB 002E7       CALLS   #0, EDT$$RD_CURLN                          ; 1059
                          9E         57  E9 002EE       BLBC    END_FOUND, 43$                             ; 1061
                          69         01  D0 002F1 48$:  MOVL    #1, EDT$$G_RNG_MORELN                      ; 1065
                                   00AF  31 002F4       BRW     60$                                        ; 0770
                          52     10  A6  D0 002F7 49$:  MOVL    16(R6), SUB_RAN                            ; 1081
                          56     14  A2  D1 002FB       CMPL    20(SUB_RAN), R6                            ; 1082
                                     07  13 002FF       BEQL    50$                                        ;
               00000000G  00         00  FB 00301       CALLS   #0, EDT$$INTER_ERR                         ;
                                 01  A2  95 00308 50$:  TSTB    1(SUB_RAN)                                 ; 1084
                                     04  12 0030B       BNEQ    51$                                        ;
                     01   A2         0B  90 0030D       MOVB    #11, 1(SUB_RAN)                            ;
               00000000G  00         01  D0 00311 51$:  MOVL    #1, EDT$$G_RNG_FRSTLN                      ; 1086
                          50         68  D0 00318       MOVL    EDT$$A_CUR_BUF, R0                         ; 1087
      00000000G  00                  60  0E  28 0031B       MOVC3   #14, (R0), EDT$$Z_RNG_ORIGPOS              ;
                                 10  A6  DD 00323       PUSHL   16(R6)                                     ; 1089
                          6A         01  FB 00326       CALLS   #1, EDT$$RNG_REPOS                         ;
                          60         50  E9 00329       BLBC    R0, 57$                                    ;
                                 04  A6  DD 0032C       PUSHL   4(R6)                                      ; 1091
                                 08  A6  DD 0032F       PUSHL   8(R6)                                      ;
               00000000G  00         02  FB 00332       CALLS   #2, EDT$$SET_SEASTR                        ;
                          7F         50  E8 00339       BLBS    R0, 61$                                    ;
                          00000000G  8F  DD 0033C       PUSHL   #EDT$_INVSTR                               ; 1094
                                     41  11 00342       BRB     56$                                        ;
                          7E     08  A6  7D 00344 52$:  MOVQ    8(R6), -(SP)                               ; 1112
```

```
                        00000000G  00               02  FB 00348        CALLS   #2, EDT$$FND_BUF                             ;
                                   2D               50  E9 0034F        BLBC    R0, 55$                                      ;
                                   57          04   A6  D0 00352        MOVL    4(R6), SUB_RANGE                             ; 1115
                                        00000000G   00  95 00356        TSTB    EDT$$G_EXE_SBITS                             ; 1117
                                                    0B  19 0035C        BLSS    53$                                          ;
                                   50               68  D0 0035E        MOVL    EDT$$A_CUR_BUF, R0                           ; 1119
              00000000G  00        60               0E  28 00361        MOVC3   #14, (R0), EDT$$Z_RNG_ORIGPOS                ;
                        00000000G  00               00  FB 00369 53$:   CALLS   #0, EDT$$TOP_BUF                             ; 1121
                                               01   A7  95 00370        TSTB    1(SUB_RANGE)                                 ; 1123
                                                    04  12 00373        BNEQ    54$                                          ;
                              01   A7               0B  90 00375        MOVB    #11, 1(SUB_RANGE)                            ;
                                                    57  DD 00379 54$:   PUSHL   SUB_RANGE                                    ; 1125
                                   6A               01  FB 0037B        CALLS   #1, EDT$$RNG_REPOS                           ;
                                                        04 0037E        RET                                                  ; 1128
                                        00000000G   8F  DD 0037F 55$:   PUSHL   #EDT$_INSMEM                                 ; 1129
                        00000000G  00               01  FB 00385 56$:   CALLS   #1, EDT$$FMT_MSG                             ;
                                                    31  11 0038C 57$:   BRB     62$                                          ; 1130
                                   50               68  D0 0038E 58$:   MOVL    EDT$$A_CUR_BUF, TEMP                         ; 1145
                                   68   00000000G   00  D0 00391        MOVL    EDT$$A_PRV_BUF, EDT$$A_CUR_BUF               ; 1146
                        00000000G  00               50  D0 00398        MOVL    TEMP, EDT$$A_PRV_BUF                         ; 1147
                        00000000G  00               00  FB 0039F 59$:   CALLS   #0, EDT$$RD_CURLN                            ; 1148
                        00000000G  00               56  D0 003A6 60$:   MOVL    R6, EDT$$Z_RNG_CURRNG                        ; 1158
                                   07   00000000G   00  E9 003AD        BLBC    EDT$$G_RNG_FRSTLN, 61$                       ; 1160
                                   50               68  D0 003B4        MOVL    EDT$$A_CUR_BUF, R0                           ;
                         6B        60               0E  28 003B7        MOVC3   #14, (R0), EDT$$Z_RNG_SAVPOS                 ;
                                   50               01  D0 003BB 61$:   MOVL    #1, R0                                       ; 1162
                                                        04 003BE        RET                                                  ;
                                                    50  D4 003BF 62$:   CLRL    R0                                           ; 1163
                                                        04 003C1        RET                                                  ;

; Routine Size:  962 bytes,    Routine Base:  _EDT$CODE + 0000


;  594          1164  1
;  595          1165  1 !<BLF/PAGE>
;  597          1166  1 END                                                      ! of module EDT$RANRPOS
;  598          1167  1
;  599          1168  0 ELUDOM



;                                        PSECT SUMMARY

;       Name                       Bytes                         Attributes

;  _EDT$CODE                            962  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        125         33          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$:RANRPOS/OBJ=OBJ$:RANRPOS MSRC$:RANRPOS.BLI/UPDATE=(ENH$:RANRPOS
;       )

; Size:          962 code + 0 data bytes
; Run Time:         00:40.2
; Elapsed Time:     00:50.3
; Lines/CPU Min:     1744
; Lexemes/CPU-Min:  9933
; Memory Used:  295 pages
; Compilation Complete
```

0138 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

PRNUMRAN
LIS

REAJOUTEX
LIS

PRPARCOMN
LIS

SCRCHKREV
LIS

SCRDELETE
LIS

SCRESCR
LIS

SCRCURS
LIS

PSECTS
LIS

PRMACCAL
LIS

PRPUSH
LIS

SCRFIND
LIS

PRPARCOM
LIS

PRPARDRV
LIS

RANRPOS
LIS

SCRCOMCUR
LIS

RANNEXT
LIS

SCRBLOB
LIS

SCRELINE
LIS

SCRFCURS
LIS

PRSEMRTN
LIS

SAUDIT
LIS