


```

CCCCCCCC HH   HH MM   MM EEEEEEEEE XX   XX CCCCCCCC 000000 MM   MM
CCCCCCCC HH   HH MM   MM EEEEEEEEE XX   XX CCCCCCCC 000000 MM   MM
CC        HH   HH MMMM MMMM EE        XX   XX CC        00   00 MMMM MMMM
CC        HH   HH MMMM MMMM EE        XX   XX CC        00   00 MMMM MMMM
CC        HH   HH MM   MM MM   EE        XX   XX CC        00   00 MM   MM
CC        HH   HH MM   MM MM   EE        XX   XX CC        00   00 MM   MM
CC        HHHHHHHHHH MM   MM EEEEEEEEE XX   XX CC        00   00 MM   MM
CC        HHHHHHHHHH MM   MM EEEEEEEEE XX   XX CC        00   00 MM   MM
CC        HH   HH MM   MM EEEEEEEEE XX   XX CC        00   00 MM   MM
CC        HH   HH MM   MM EEEEEEEEE XX   XX CC        00   00 MM   MM
CC        HH   HH MM   MM EEEEEEEEE XX   XX CC        00   00 MM   MM
CCCCCCCC HH   HH MM   MM EEEEEEEEE XX   XX CCCCCCCC 000000 MM   MM
CCCCCCCC HH   HH MM   MM EEEEEEEEE XX   XX CCCCCCCC 000000 MM   MM

```



```

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

```



```

0001 0 %TITLE 'EDT$CHMEXCOM - execute certain change-mode commands'
0002 0 MODULE EDT$CHMEXCOM ( ! Execute certain change-mode commands
0003 0 IDENT = 'V04-000' ! File: CHMEXCOM.BLI Edit: JBS1035
0004 0 ) =
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0011 1 * ALL RIGHTS RESERVED. *
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0018 1 * TRANSFERRED. *
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0022 1 * CORPORATION. *
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 ++
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 This module executes the change mode commands which
0037 1 do not take an entity.
0038 1
0039 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0040 1
0041 1 AUTHOR: Bob Kushlis, CREATION DATE: Unknown
0042 1
0043 1 MODIFIED BY:
0044 1
0045 1 1-001 - Original. DJS 04-Feb-1981. This module was created by
0046 1 extracting the routine EXECUTE_COM from module CHANGE.BLI.
0047 1 1-002 - Regularized headers. JBS 25-Feb-1981
0048 1 1-003 - Fix module name. JBS 02-Mar-1981
0049 1 1-004 - Change SPLIT LINE to EDT$SPLT LNINS. JBS 30-Mar-1981
0050 1 1-005 - Use the ASSERT macro. JBS 01-Jun-1981
0051 1 1-006 - Remove explicit journaling. JBS 18-Jun-1981
0052 1 1-007 - Use new message codes. JBS 04-Aug-1981
0053 1 1-008 - Add bell verb. STS 11-Aug-1981
0054 1 1-009 - Add the date verb. STS 31-Aug-1981
0055 1 1-010 - Add verbs to set up default verb. STS 21-Sep-1981
0056 1 1-011 - Add verbs for toggle select and delete select. STS 23-Sep-1981
0057 1 1-012 - Added command to set success to 0 if verb was select and select

```



```
: 58      0058 1 | range was already active. I needed this status for search and
: 59      0059 1 | select. STS 28-Sep-1981
: 60      0060 1 | 1-013 - Add a return value to indicate end of journal file. JBS 02-Oct-1981
: 61      0061 1 | 1-014 - Remove parameter from EDT$$SUB_CMD call. SMB 28-Oct-1981
: 62      0062 1 | 1-015 - Revise Tab Compute calculation when SHFL not zero. SMB 06-Nov-1981
: 63      0063 1 | 1-016 - Add range checking to ASC command. JBS 10-Feb-1982
: 64      0064 1 | 1-017 - Correct spelling of error code. JBS 12-Feb-1982
: 65      0065 1 | 1-018 - Add a flag for EXT command mode entered. SMB 26-Feb-1982
: 66      0066 1 | 1-019 - Rewrite word wrapping code. JBS 07-Apr-1982
: 67      0067 1 | 1-020 - Give messages on error returns from setting search strings. JBS 04-May-1982
: 68      0068 1 | 1-021 - Set a flag if control C actually aborts something. JBS 24-May-1982
: 69      0069 1 | 1-022 - Change setting of output format routine. SMB 30-Jun-1982
: 70      0070 1 | 1-023 - Set format output to TI_WRSTR for EXT output. SMB 02-Jul-1982
: 71      0071 1 | 1-024 - Make KS move the cursor even if PST CNT = 0. SMB 22-Jul-1982
: 72      0072 1 | 1-025 - Add the XLATE command. STS 13-Aug-1982
: 73      0073 1 | 1-026 - Flag screen changed for HELP, SHL and SHR. JBS 13-Sep-1982
: 74      0074 1 | 1-027 - Remove EDT$$G_LN_NO for new screen update logic. JBS 29-Sep-1982
: 75      0075 1 | 1-028 - Remove external declaration of EDT$$FMT_LIT, not used. JBS 05-Oct-1982
: 76      0076 1 | 1-029 - Remove call to SC_INIT, set a flag instead. SMB 06-Oct-1982
: 77      0077 1 | 1-030 - Change EDT$$G_SCR_CHGD to EDT$$G_SCR_REBUILD in a few places. JBS 09-Oct-1982
: 78      0078 1 | 1-031 - Rebuild the screen data base if selection is too complex. JBS 02-Dec-1982
: 79      0079 1 | 1-032 - Revise handling of EDT$$G_SHF. JBS 14-Dec-1982
: 80      0080 1 | 1-033 - Put WPS and VT220 support under a conditional. JBS 10-Feb-1983
: 81      0081 1 | 1-034 - Remove declarations of routines which aren't called. SMB 23-Feb-1983
: 82      0082 1 | 1-035 - Add new value for EDT$$G_SCR_CHGD. JBS 02-Mar-1983
: 83      0083 1 | --
: 84      0084 1 |
```



```
: 86      0085 1 %SBTTL 'Declarations'
: 87      0086 1
: 88      0087 1 TABLE OF CONTENTS:
: 89      0088 1
: 90      0089 1
: 91      0090 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 92      0529 1
: 93      0530 1 FORWARD ROUTINE
: 94      0531 1     EDT$SEXE_CHMCMD1;
: 95      0532 1
: 96      0533 1
: 97      0534 1 INCLUDE FILES:
: 98      0535 1
: 99      0536 1
: 100     0537 1 REQUIRE 'EDTSRC:EDTREQ';
: 101     0672 1
: 102     0673 1 LIBRARY 'EDTSRC:SUPPORTS';
: 103     0674 1
: 104     0675 1
: 105     0676 1 MACROS:
: 106     0677 1
: 107     0678 1     NONE
: 108     0679 1
: 109     0680 1 EQUATED SYMBOLS:
: 110     0681 1
: 111     0682 1     NONE
: 112     0683 1
: 113     0684 1 OWN STORAGE:
: 114     0685 1
: 115     0686 1     NONE
: 116     0687 1
: 117     0688 1 EXTERNAL REFERENCES:
: 118     0689 1
: 119     0690 1 In the routine
```

! Execute the verbs which do not take an entity specification


```
121 0691 1 %SBTTL 'EDT$SEXE_CHMCMD1 - execute certain change-mode commands'
122 0692 1
123 0693 1 GLOBAL ROUTINE EDT$SEXE_CHMCMD1 (
124 0694 1     VERB,
125 0695 1     COUNT,
126 0696 1     OPERAND,
127 0697 1     EXPLICIT
128 0698 1 ) =
129 0699 1
130 0700 1 ++
131 0701 1 FUNCTIONAL DESCRIPTION:
132 0702 1
133 0703 1     This routine executes a command which is not of the verb entity form.
134 0704 1
135 0705 1 FORMAL PARAMETERS:
136 0706 1
137 0707 1     VERB          command number
138 0708 1     COUNT        repeat count (char value for ASC)
139 0709 1     OPERAND      Pointer to start of operand for insert, insert_cc etc.
140 0710 1     EXPLICIT     1 = the count is explicit
141 0711 1
142 0712 1 IMPLICIT INPUTS:
143 0713 1
144 0714 1     EDT$G_TI_SCROLL
145 0715 1     EDT$G_SCR_LNS
146 0716 1     EDT$ST_DEL_CH
147 0717 1     EDT$G_DEL_CHLEN
148 0718 1     EDT$G_DIR
149 0719 1     EDT$ST_DEL_LN
150 0720 1     EDT$G_DEL_LNLEN
151 0721 1     EDT$G_DIR_MOD
152 0722 1     EDT$ST_DEL_WD
153 0723 1     EDT$G_DEL_WDLEN
154 0724 1     EDT$G_EXI
155 0725 1     EDT$G_PST_CNT
156 0726 1     EDT$SA_RPL_STR
157 0727 1     EDT$G_RPL_LEN
158 0728 1     EDT$SA_SEA_STR
159 0729 1     EDT$SA_SEL_BUF
160 0730 1     EDT$SA_OLD_SEL
161 0731 1     EDT$SL_SEL_LN
162 0732 1     EDT$SA_SEL_POS
163 0733 1     EDT$G_SHF
164 0734 1     EDT$G_TRUN
165 0735 1     EDT$G_SEA_LEN
166 0736 1     EDT$SL_TOP_LN
167 0737 1     EDT$G_EXITD
168 0738 1     EDT$G_TAB_SIZ
169 0739 1     EDT$SA_CUR_BUF
170 0740 1     EDT$G_TAB_LVL
171 0741 1     EDT$G_TI_TYP
172 0742 1     EDT$ST_FMT_BUF
173 0743 1     EDT$SA_FMT_CUR
174 0744 1     EDT$ST_LN_BUF
175 0745 1     EDT$SA_LN_PTR
176 0746 1
177 0747 1 IMPLICIT OUTPUTS:
```



```
178 0748 1 |
179 0749 1 |      EDT$SG_SHF
180 0750 1 |      EDT$SG_TAB_LVL
181 0751 1 |      EDT$SA_FMT_CUR
182 0752 1 |      EDT$SA_LN_PTR
183 0753 1 |      EDT$SG_VERT
184 0754 1 |      EDT$SG_DFLT_VERB
185 0755 1 |      EDT$SA_SEL_BUF
186 0756 1 |      EDT$SG_CC_DONE
187 0757 1 |      EDT$SG_SCR_CHGD
188 0758 1 |      EDT$SG_SCR_REBUILD
189 0759 1 |
190 0760 1 | ROUTINE VALUE:
191 0761 1 |
192 0762 1 |      0 = failure, 1 = success, 2 = end of journal file
193 0763 1 |
194 0764 1 | SIDE EFFECTS:
195 0765 1 |
196 0766 1 |      MANY
197 0767 1 |
198 0768 1 | --
199 0769 1 |
200 0770 2 | BEGIN
201 0771 2 |
202 0772 2 | EXTERNAL ROUTINE
203 0773 2 |      EDT$INS_STR,
204 0774 2 |      EDT$INS_CHS,
205 0775 2 |      EDT$MOV_TOCOL,
206 0776 2 |
207 L 0777 2 | %IF %BLISS (BLISS32)
208 0778 2 | %THEN
209 0779 2 |      EDT$GET_XLATE,
210 0780 2 | %FI
211 0781 2 |
212 0782 2 |      EDT$UNDL,
213 0783 2 |      EDT$MSG_BELL : NOVALUE,
214 0784 2 |      EDT$CHK_CC,
215 0785 2 |      EDT$LN_DEFK,
216 0786 2 |      EDT$PST_CMD,
217 0787 2 |      EDT$SUB_CMD,
218 0788 2 |      EDT$EXT_CMD,
219 0789 2 |      EDT$KPAD_HLP,
220 0790 2 |      EDT$TI_WRLN,
221 0791 2 |      EDT$TI_WRSTR,
222 0792 2 |      EDT$RPE_CHGDLN,
223 0793 2 |      EDT$GET_TXTLN,
224 0794 2 |      EDT$CS_LEFT,
225 0795 2 |      EDT$SC_CPUC$POS,
226 0796 2 |      EDT$WORD_WRAP,
227 0797 2 |      EDT$SC_POSCSIF,
228 0798 2 |      EDT$SC_NONREVID,
229 0799 2 |      EDT$SC_SETSCLLR$G,
230 0800 2 |      EDT$SC_FULLSCLL,
231 0801 2 |      EDT$STOP_WKINGMSG,
232 0802 2 |      EDT$SET_SEASUBSTR
233 0803 2 |
234 L 0804 2 | %IF SUPPORT_WPS
```

! Insert a string of characters at the current position
! Insert a string of characters which may include carriage returns
! Insert tabs and spaces

! call translation routine

! Insert the contents of an undelete buffer
! Output a message to the terminal with a warning bell
! Check to see if a CTRL/C has been typed
! Define a key for keypad editing
! Execute the paste command
! Execute the SUBSTITUTE command
! Extend command handler
! Keypad mode help processor
! Write to terminal
! Write to terminal unformatted
! Declare current line as changed
! Get current line in line buffer
! Move left a character
! Compute cursor position
! Try doing word wrapping
! Put cursor position in format buffer
! End reverse video
! Set the scrolling region
! Reset the scrolling region
! Terminate working AST
! Setup SUBSTITUTE strings


```
: 235      0805  2 %THEN
: 236      0806  2
: 237      0807  2      EDT$RING BELL : NOVALUE,      ! Ring the bell
: 238      0808  2      EDT$INS_DATIM                ! Insert date and time
: 239      0809  2 %FI
: 240      0810  2
: 241      0811  2      ;
: 242      0812  2
: 243      0813  2      EXTERNAL
: 244      0814  2      EDT$G_TI_SCROLL,              ! Scrolling terminal
: 245      0815  2      EDT$G_SCR_LNS,                ! Number of screen lines
: 246      0816  2      EDT$A_FMT_WRRUT,              ! Holds address of output format routine
: 247      0817  2      EDT$G_EXT_MOD,                ! 1=in EXT command mode
: 248      0818  2      EDT$ST_DEL_CH : BLOCK          ! Deleted character buffer.
: 249      0819  2      [CH$ALLOCATION (2)],
: 250      0820  2      EDT$G_DEL_CHLEN,              ! Length of deleted character buffer
: 251      0821  2      EDT$G_DIR,                    ! The current direction.
: 252      0822  2      EDT$ST_DEL_LN : BLOCK          ! Deleted line buffer.
: 253      0823  2      [CH$ALLOCATION (257)],
: 254      0824  2      EDT$G_DEL_LNLEN,              ! Deleted line length.
: 255      0825  2      EDT$G_DIR_MOD,                ! The directional mode.
: 256      0826  2      EDT$ST_DEL_WD : BLOCK          ! Deleted word buffer.
: 257      0827  2      [CH$ALLOCATION (81)],
: 258      0828  2      EDT$G_DEL_WDLEN,              ! Length of del word string.
: 259      0829  2      EDT$G_EXI,                    ! Change mode has been exited.
: 260      0830  2      EDT$G_PST_CNT,                ! No. of characters pasted.
: 261      0831  2      EDT$A_RPL_STR,                ! Address of replace string.
: 262      0832  2      EDT$G_RPL_LEN,                ! Length of replace string.
: 263      0833  2      EDT$G_SEA_STRLEN,              ! Length of search string
: 264      0834  2      EDT$A_SEA_STR,                ! Address of search string.
: 265      0835  2      EDT$A_SEL_BUF,                ! Pointer to select buffer.
: 266      0836  2      EDT$A_OLD_SEL,                ! Pointer to old select buffer
: 267      0837  2      EDT$SL_SEL_LN : LN_BLOCK,      ! Relative line number of select.
: 268      0838  2      EDT$A_SEL_POS,                ! select position.
: 269      0839  2      EDT$G_SHF,                    ! The number of columns shifted.
: 270      0840  2      EDT$G_TRUN,                    ! 0 = SET NOTRUNCATE
: 271      0841  2      EDT$G_SEA_LEN,                ! Length of search string.
: 272      0842  2      EDT$SL_TOP_LN : LN_BLOCK,      ! The forced to top line.
: 273      0843  2      EDT$G_VERT,                  ! Last entity was VERT flag.
: 274      0844  2      EDT$G_EXITD,                 ! Exit from EDT
: 275      0845  2      EDT$G_SCR_CHGD,               ! Set if screen must be initialized and/or repainted
: 276      0846  2      EDT$G_SCR_REBUILD,            ! Set if text part of screen must be rebuilt from work file
: 277      0847  2      EDT$G_TAB_SIZ,                ! Structured tab size
: 278      0848  2      EDT$A_CUR_BUF : REF TBCB_BLOCK, ! The current buffer tbc
: 279      0849  2      EDT$G_TAB_LVL,                ! Structured tab level.
: 280      0850  2      EDT$G_TI_TYP,                 ! Terminal type.
: 281      0851  2      EDT$ST_FMT_BUF,               ! Format buffer
: 282      0852  2      EDT$A_FMT_CUR,                ! Pointer into format buffer
: 283      0853  2      EDT$ST_LN_BUF,                 ! Current line buffer
: 284      0854  2      EDT$A_LN_PTR,                 ! Current character pointer
: 285      0855  2
: 286      L 0856  2 %IF SUPPORT_WPS
: 287      0857  2 %THEN
: 288      0858  2      EDT$G_DFLT_VERB,              ! Default verb
: 289      0859  2 %FI
: 290      0860  2
: 291      0861  2      EDT$G_CC_DONE;                ! Set to 1 if control C aborts something
```



```
292 0862 2
293 0863 2
294 0864 2
295 0865 2
296 0866 2
297 0867 2
298 0868 2
299 0869 2
300 0870 2
301 0871 2
302 0872 2
303 0873 2
304 0874 2
305 0875 2
306 0876 2
307 0877 2
308 0878 2
309 0879 2
310 0880 2
311 0881 2
312 0882 2
313 0883 2
314 0884 2
315 0885 2
316 0886 2
317 0887 2
318 0888 2
319 0889 2
320 0890 2
321 0891 2
322 0892 2
323 0893 2
324 0894 2
325 0895 2
326 0896 2
327 0897 2
328 0898 2
329 0899 2
330 0900 2
331 0901 2
332 0902 2
333 0903 2
334 0904 2
335 0905 2
336 0906 2
337 0907 2
338 0908 2
339 0909 2
340 0910 2
341 0911 2
342 0912 2
343 0913 2
344 0914 2
345 0915 2
346 0916 2
347 0917 2
348 0918 2

!+ Declare the message codes to be used.
!-
  MESSAGES ((SELALRACT, INVSUBCOM, CLDNOTALN, INVASCCHR, INVSTR));

  LOCAL
    SUCCEED,
    START_POS : POS_BLOCK,
    END_POS : POS_BLOCK,
    NUM_LINES;

!+ If verb is SUBSTITUTE, set up the search and substitute strings.
!-
  IF (.VERB EQL VERB_K_SUBS)
  THEN
    SUCCEED = EDT$$SET SEASUBSTR (.EDT$$A_SEA_STR, !
      .EDT$$G_SEA_LEN,
      .EDT$$A_RPL_STR,
      .EDT$$G_RPL_LEN);
  ELSE
    SUCCEED = 1;

  IF ( NOT .SUCCEED)
  THEN
    EDT$$MSG_BELL (EDT$ INVSTR)
  ELSE
    DO
      BEGIN
        CASE .VERB FROM VERB_K_SEL TO LAST_K_VERB OF
          SET
            [VERB_K_UNDC] :
              BEGIN
                SUCCEED = EDT$$UNDL (EDT$$T_DEL_CH, .EDT$$G_DEL_CHLEN);
              END;
            [VERB_K_UNDW] :
              BEGIN
                SUCCEED = EDT$$UNDL (EDT$$T_DEL_WD, .EDT$$G_DEL_WDLEN);
              END;
            [VERB_K_UNDL] :
              BEGIN
                SUCCEED = EDT$$UNDL (EDT$$T_DEL_LN, .EDT$$G_DEL_LNLEN);
              END;
            [VERB_K_INSERT] :
              BEGIN
                SUCCEED = EDT$$INS_CHS (.OPERAND, .EDT$$G_SEA_LEN);
                IF .SUCCEED THEN SUCCEED = EDT$$WORD_WRAP ();
              END;
        END
      END
```



```
349      0919 3      END;
350      0920 3
351      0921 3      [VERB_K_XLATE] :
352      0922 4      BEGIN
353      0923 4
354      L 0924 4 %IF %BLISS (BLISS32)
355      0925 4 %THEN
356      0926 4      SUCCEED = EDT$$GET_XLATE (.OPERAND, .EDT$$G_SEA_LEN);
357      U 0927 4 %ELSE
358      U 0928 4      EDT$$MSG_BELL (EDT$_INVSUBCOM);
359      U 0929 4      RETURN (0);
360      0930 4 %FI
361      0931 4
362      0932 3      END;
363      0933 3
364      0934 3      [VERB_K_CC] :
365      0935 4      BEGIN
366      0936 4      SUCCEED = EDT$$INS_CHS (%REF (CH$RCHAR (.OPERAND) - %C'a'), 1);
367      0937 3      END;
368      0938 3
369      0939 3      [VERB_K_BACK] :
370      0940 4      BEGIN
371      0941 4      EDT$$G_DIR_MOD = DIR_BACKWARD;
372      0942 4      EXITLOOP;
373      0943 3      END;
374      0944 3
375      0945 3      [VERB_K_ADV] :
376      0946 4      BEGIN
377      0947 4      EDT$$G_DIR_MOD = DIR_FORWARD;
378      0948 4      EXITLOOP;
379      0949 3      END;
380      0950 3
381      0951 3      [VERB_K_DLWC] :
382      0952 4      BEGIN
383      0953 4
384      L 0954 4 %IF SUPPORT_WPS
385      0955 4 %THEN
386      0956 4      EDT$$G_DFLT_VERB = VERB_K_CHGL;      ! set up default verb to change case lower
387      0957 4 %FI
388      0958 4
389      0959 4      EXITLOOP;
390      0960 3      END;
391      0961 3
392      0962 3      [VERB_K_DUPC] :
393      0963 4      BEGIN
394      0964 4
395      L 0965 4 %IF SUPPORT_WPS
396      0966 4 %THEN
397      0967 4      EDT$$G_DFLT_VERB = VERB_K_CHGU;      ! set up default verb to change case upper
398      0968 4 %FI
399      0969 4
400      0970 4      EXITLOOP;
401      0971 3      END;
402      0972 3
403      0973 3      [VERB_K_DMOV] :
404      0974 4      BEGIN
405      0975 4
```



```
: 406      L 0976 4 %IF SUPPORT_WPS
: 407      0977 4 %THEN
: 408      0978 4
: 409      0979 4 %FI
: 410      0980 4
: 411      0981 4
: 412      0982 3
: 413      0983 3
: 414      0984 3
: 415      0985 4
: 416      0986 4
: 417      0987 4
: 418      0988 4
: 419      0989 4
: 420      0990 4
: 421      0991 3
: 422      0992 3
: 423      0993 3
: 424      0994 4
: 425      0995 4
: 426      0996 3
: 427      0997 3
: 428      0998 3
: 429      0999 4
: 430      1000 4
: 431      1001 5
: 432      1002 4
: 433      1003 5
: 434      1004 5
: 435      1005 5
: 436      1006 5
: 437      1007 4
: 438      1008 5
: 439      1009 5
: 440      1010 5
: 441      1011 5
: 442      1012 5
: 443      1013 5
: 444      1014 5
: 445      1015 4
: 446      1016 4
: 447      1017 4
: 448      1018 3
: 449      1019 3
: 450      1020 3
: 451      1021 4
: 452      1022 4
: 453      1023 4
: 454      1024 3
: 455      1025 3
: 456      1026 3
: 457      1027 4
: 458      1028 4
: 459      L 1029 4 %IF SUPPORT_WPS
: 460      1030 4 %THEN
: 461      1031 4
: 462      1032 4 %FI

      EDT$$G_DFLT_VERB = VERB_K_MOVE;      ! set up default verb to move

      EXITLOOP;
      END;

[VERB_K_EXIT, VERB_K_QUIT] :
      BEGIN
      EDT$$G_EXI = 1;

      IF (.VERB EQL VERB_K_QUIT) THEN EDT$$G_EXITD = 1;

      RETURN (1);
      END;

[VERB_K_PASTE] :
      BEGIN
      SUCCEED = EDT$$PST_CMD ();
      END;

[VERB_K_SEL] :
      BEGIN
      IF (.EDT$$A_SEL_BUF NEQA 0)
      THEN
      BEGIN
      EDT$$MSG_BELL (EDT$_SELALRACT);
      SUCCEED = 0;
      END
      ELSE
      BEGIN
      MOVELINE (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_SEL_LN);
      EDT$$A_SEL_BUF = .EDT$$A_CUR_BUF;
      EDT$$A_SEL_POS = .EDT$$A_LN_PTR;

      IF (.EDT$$A_OLD_SEL NEQA 0) THEN EDT$$G_SCR_REBUILD = 1;

      END;
      EXITLOOP;
      END;

[VERB_K_REF] :
      BEGIN
      EDT$$G_SCR_CHGD = 2;      ! Initialize the terminal and repaint the screen
      EXITLOOP;
      END;

[VERB_K_BELL] :
      BEGIN
```



```

: 463      1033  4
: 464      1034  4      EXITLOOP;
: 465      1035  3      END;
: 466      1036  3
: 467      1037  3      [VERB_K_DATE] :
: 468      1038  4      BEGIN
: 469      1039  4
: 470      L 1040  4  %IF SUPPORT_WPS
: 471      1041  4  %THEN
: 472      1042  4      SUCCEED = EDT$$INS_DATIM ();
: 473      U 1043  4  %ELSE
: 474      U 1044  4      0
: 475      1045  4  %FI
: 476      1046  4
: 477      1047  4      ;
: 478      1048  3      END;
: 479      1049  3
: 480      1050  3      [VERB_K_DEFK] :
: 481      1051  4      BEGIN
: 482      1052  4      SUCCEED = EDT$$LN_DEFK ();
: 483      1053  4      EXITLOOP;
: 484      1054  3      END;
: 485      1055  3
: 486      1056  3      [VERB_K_TOP] :
: 487      1057  4      BEGIN
: 488      1058  4      MOVELINE (EDT$$A_CUR_BUF [TBCB_CUR_LIN], EDT$$L_TOP_LN);
: 489      1059  4      EXITLOOP;
: 490      1060  3      END;
: 491      1061  3
: 492      1062  3      [VERB_K_HELP] :
: 493      1063  4      BEGIN
: 494      1064  4
: 495      1065  4      LOCAL
: 496      1066  4      KPAD_STATUS;
: 497      1067  4
: 498      1068  4      EDT$$SC_NONREVID ();
: 499      1069  4      EDT$$STOP_WKINGMSG ();
: 500      1070  4      KPAD_STATUS = EDT$$KPAD_HLP ();
: 501      1071  4      EDT$$G_SCR_CHGD = 1;      ! Repaint the screen
: 502      1072  4
: 503      1073  4      IF (.KPAD_STATUS EQL 0) THEN SUCCEED = 2;
: 504      1074  4
: 505      1075  4      EXITLOOP;
: 506      1076  3      END;
: 507      1077  3
: 508      1078  3      [VERB_K_ASC] :
: 509      1079  4      BEGIN
: 510      1080  4
: 511      1081  5      IF ((.COUNT GTR 255) OR (.COUNT LSS 0))
: 512      1082  4      THEN
: 513      1083  5      BEGIN
: 514      1084  5      EDT$$MSG_BELL (EDT$_INVASCCHR);
: 515      1085  5      SUCCEED = 0;
: 516      1086  5      END
: 517      1087  4      ELSE
: 518      1088  5      BEGIN
: 519      1089  5      EDT$$INS_STR (%REF (.COUNT), 1);
```

```

520      1090      5      EXITLOOP;
521      1091      4      END;
522      1092      4
523      1093      3      END;
524      1094      3
525      1095      3      [VERB_K SUBS, VERB_K_SN] :
526      1096      4      BEGIN
527      1097      4      SUCCEED = EDT$$SUB_CMD ();
528      1098      3      END;
529      1099      3
530      1100      3      [VERB_K KS] :
531      1101      4      BEGIN
532      1102      4      ! Adjust for KED SUBSTITUTE.
533      1103      4      !+ The cursor should move left one even if G_PST_CNT is zero
534      1104      4      !-
535      1105      4
536      1106      5      IF (.EDT$$G_DIR EQL DIR_BACKWARD) AND (.EDT$$G_PST_CNT NEQ 0)
537      1107      4      THEN
538      1108      4
539      1109      4      DECR I FROM .EDT$$G_PST_CNT - 1 TO 0 DO
540      1110      4      EDT$$CS_LEFT ()
541      1111      4
542      1112      4      ELSE
543      1113      4      EDT$$CS_LEFT ();
544      1114      4
545      1115      3      END;
546      1116      3
547      1117      3      [VERB_K SHL] :
548      1118      4      BEGIN
549      1119      4      EDT$$G_SHF = .EDT$$G_SHF + 8;
550      1120      4
551      1121      4      IF ((.EDT$$G_SHF GEQ 32767) OR (.EDT$$G_SHF LSS 0)) THEN EDT$$G_SHF = 0;
552      1122      4
553      1123      4      EDT$$G_SCR_CHGD = 1;      ! repaint the screen
554      1124      4
555      1125      4      IF ( NOT .EDT$$G_TRUN) THEN EDT$$G_SCR_REBUILD = 1;
556      1126      4
557      1127      3      END;
558      1128      3
559      1129      3      [VERB_K SHR] :
560      1130      4      BEGIN
561      1131      4      EDT$$G_SHF = .EDT$$G_SHF - 8;
562      1132      4
563      1133      4      IF ((.EDT$$G_SHF GEQ 32767) OR (.EDT$$G_SHF LSS 0)) THEN EDT$$G_SHF = 0;
564      1134      4
565      1135      4      EDT$$G_SCR_CHGD = 1;      ! repaint the screen
566      1136      4
567      1137      4      IF ( NOT .EDT$$G_TRUN) THEN EDT$$G_SCR_REBUILD = 1;
568      1138      4
569      1139      3      END;
570      1140      3
571      1141      3      [VERB_K TAB] :
572      1142      4      BEGIN
573      1143      4
574      1144      4      LOCAL
575      1145      4      TAB_COUNT;
576      1146      4
```



```
577 1147 5 IF (CH$PTR_NEQ (.EDT$$A_LN_PTR, CH$PTR (EDT$$T_LN_BUF)) OR !
578 1148 5 (.EDT$$G_TAB_SIZ EQL 0)
579 1149 4 THEN
580 1150 4 TAB_COUNT = 8
581 1151 4 ELSE
582 1152 4 TAB_COUNT = .EDT$$G_TAB_LVL*.EDT$$G_TAB_SIZ;
583 1153 4
584 1154 4 SUCCEED = EDT$$MOV_TOCOL (.TAB_COUNT);
585 1155 3 END;
586 1156 3
587 1157 3 [VERB_K_TC] :
588 1158 4 BEGIN
589 1159 4
590 1160 4 LOCAL
591 1161 4 COL,
592 1162 4 LIN;
593 1163 4
594 1164 4 IF (.EDT$$G_TAB_SIZ EQL 0) THEN EXITLOOP;
595 1165 4
596 1166 4 EDT$$SC_CPU_CSPS (LIN, COL);
597 1167 4 COL = .COL + .EDT$$G_SHF;
598 1168 4
599 1169 5 IF ((.COL MOD .EDT$$G_TAB_SIZ) NEQ 0)
600 1170 4 THEN
601 1171 4 EDT$$MSG_BELL (EDT$_CLDNOTALN)
602 1172 4 ELSE
603 1173 4 EDT$$G_TAB_LVL = (MAX (0, .COL))/EDT$$G_TAB_SIZ;
604 1174 4
605 1175 4 EXITLOOP;
606 1176 3 END;
607 1177 3
608 1178 3 [VERB_K_TD] :
609 1179 4 BEGIN
610 1180 4 EDT$$G_TAB_LVL = MAX (0, .EDT$$G_TAB_LVL - 1);
611 1181 3 END;
612 1182 3
613 1183 3 [VERB_K_TI] :
614 1184 4 BEGIN
615 1185 4 EDT$$G_TAB_LVL = .EDT$$G_TAB_LVL + 1;
616 1186 3 END;
617 1187 3
618 1188 3 [VERB_K_EXT] :
619 1189 4 BEGIN
620 1190 4 EDT$$SC_FULLSCLL ();
621 1191 4
622 1192 5 IF ((.EDT$$G_TI_TYP EQL TERM_VT52) OR !
623 1193 5 (.EDT$$G_TI_TYP EQL TERM_VT100))
624 1194 4 THEN
625 1195 4 EDT$$A_FMT_WRRUT = EDT$$TI_WRSTR
626 1196 4 ELSE
627 1197 5 BEGIN
628 1198 5 EDT$$STOP_WKINGMSG ();
629 1199 5 EDT$$A_FMT_WRRUT = EDT$$TI_WRLN;
630 1200 4 END;
631 1201 4
632 1202 4 EDT$$RPL_CHGDLN ();
633 1203 4 EDT$$A_COR_BUF [TBCB_CHAR_POS] = CH$DIFF (.EDT$$A_LN_PTR, CH$PTR (EDT$$T_LN_BUF));
```

```

: 634      1204  4      EDT$G_EXT_MOD = 1;
: 635      1205  4      EDT$EXT_CMD ();
: 636      1206  4      EDT$G_EXT_MOD = 0;
: 637      1207  4      EDT$A_FMT_CUR = EDT$T_FMT_BUF;
: 638      1208  4      EDT$GET_TITLN ();
: 639      1209  4      EDT$A_LN_PTR = CH$PTR (EDT$T_LN_BUF, .EDT$A_CUR_BUF [TBCB_CHAR_POS]);
: 640      1210  4
: 641      1211  4      IF (.EDT$G_TI_SCROLL) THEN EDT$SC_SETSCLLREG (0, .EDT$G_SCR_LNS);
: 642      1212  4
: 643      1213  4      EXITLOOP;
: 644      1214  3      END;
: 645      1215  3
: 646      1216  3      [VERB_K_DESEL] :
: 647      1217  4      BEGIN
: 648      1218  4
: 649      L 1219  4      %IF SUPPORT_WPS
: 650      1220  4      %THEN
: 651      1221  4      EDT$A_SEL_BUF = 0;          ! No select range active
: 652      U 1222  4      %ELSE
: 653      1223  4      0
: 654      1224  4      %FI
: 655      1225  4
: 656      1226  3      END;
: 657      1227  3
: 658      1228  3      [VERB_K_TGSEL] :
: 659      1229  4      BEGIN
: 660      1230  4
: 661      L 1231  4      %IF SUPPORT_WPS
: 662      1232  4      %THEN
: 663      1233  4
: 664      1234  5      IF (.EDT$A_SEL_BUF EQLA 0)
: 665      1235  4      THEN
: 666      1236  5      BEGIN
: 667      1237  5      MOVELINE (EDT$A_CUR_BUF [TBCB_CUR_LIN], EDT$SL_SEL_LN);
: 668      1238  5      EDT$A_SEL_BUF = .EDT$A_CUR_BUF;
: 669      1239  5      EDT$A_SEL_POS = .EDT$A_LN_PTR;
: 670      1240  5
: 671      1241  5      IF (.EDT$A_OLD_SEL NEQA 0) THEN EDT$G_SCR_REBUILD = 1;
: 672      1242  5
: 673      1243  5      END
: 674      1244  4      ELSE
: 675      1245  4      EDT$A_SEL_BUF = 0;
: 676      1246  4
: 677      U 1247  4      %ELSE
: 678      1248  4      0
: 679      1249  4      %FI
: 680      1250  4
: 681      1251  3      END;
: 682      1252  3
: 683      1253  3      [VERB_K_CLSS] :
: 684      1254  4      BEGIN
: 685      1255  4
: 686      L 1256  4      %IF SUPPORT_WPS
: 687      1257  4      %THEN
: 688      1258  4      EDT$G_SEA_STRLN = 0;          ! reset search string
: 689      U 1259  4      %ELSE
: 690      1260  4      0
```



```

: 691      1261  4 %FI
: 692      1262  4
: 693      1263  3
: 694      1264  3      END;
: 695      1265  3      [OUTRANGE] :
: 696      1266  3      ASSERT (0);
: 697      1267  3      TES;
: 698      1268  3
: 699      1269  3      IF (.EXPLICIT NEQ 0) THEN COUNT = .COUNT - 1;
: 700      1270  3
: 701      1271  3      IF (.SUCCEED NEQ 1) THEN EXITLOOP;
: 702      1272  3
: 703      1273  3      IF EDT$$CHK_CC ( )
: 704      1274  3      THEN
: 705      1275  4      BEGIN
: 706      1276  4
: 707      1277  4      IF (.COUNT GTR 0) THEN EDT$$G_CC_DONE = 1;
: 708      1278  4
: 709      1279  4      EXITLOOP;
: 710      1280  3      END;
: 711      1281  3
: 712      1282  3      END
: 713      1283  2      UNTIL (.COUNT LEQ 0);
: 714      1284  2
: 715      1285  2      !+
: 716      1286  2      Unless the command was advance or backup, turn off the EDT$$G_VERT flag.
: 717      1287  2      !-
: 718      1288  2
: 719      1289  2      IF ((.VERB NEQ VERB_K_ADV) AND (.VERB NEQ VERB_K_BACK)) THEN EDT$$G_VERT = 0;
: 720      1290  2
: 721      1291  2      RETURN (.SUCCEED);
: 722      1292  1      END;

```

! of routine EDT\$\$EXE_CHMCMD1

```

:
: .TITLE  EDT$CHMEXCOM EDT$CHMEXCOM - execute certain cha
:         nge-mode comm
: .IDENT  \V04-000\
:
: .EXTRN  EDT$$INS_STR, EDT$$INS_CHS
: .EXTRN  EDT$$MOV_TOCOL, EDT$$GET_XLATE
: .EXTRN  EDT$$UNDC, EDT$$MSG_BELL
: .EXTRN  EDT$$CHK_CC, EDT$$LN_DEFK
: .EXTRN  EDT$$PST_CMD, EDT$$SOB_CMD
: .EXTRN  EDT$$EXT_CMD, EDT$$KPAD_HLP
: .EXTRN  EDT$$TI_QRLN, EDT$$TI_WRSTR
: .EXTRN  EDT$$RPL_CHGDLN
: .EXTRN  EDT$$GET_TXTLN, EDT$$CS_LEFT
: .EXTRN  EDT$$SC_CPUCSP0S
: .EXTRN  EDT$$WORD_WRAP, EDT$$SC_POSCSIF
: .EXTRN  EDT$$SC_NONREVID
: .EXTRN  EDT$$SC_SETSCLLREG
: .EXTRN  EDT$$SC_FULLSCLL
: .EXTRN  EDT$$STOP_WKINGMSG
: .EXTRN  EDT$$SET_SEASUBSTR
: .EXTRN  EDT$$RING_BELL, EDT$$INS_DATIM
: .EXTRN  EDT$$G_TI_SCROLL
: .EXTRN  EDT$$G_SCR_LNS, EDT$$A_FMT_WRRUT

```


.EXTRN EDT\$SG_EXT_MOD, EDT\$ST_DEL_CH
.EXTRN EDT\$SG_DEL_CHLEN
.EXTRN EDT\$SG_DIR, EDT\$ST_DEL_LN
.EXTRN EDT\$SG_DEL_LNLEN
.EXTRN EDT\$SG_DIR_MOD, EDT\$ST_DEL_WD
.EXTRN EDT\$SG_DEL_WDLEN
.EXTRN EDT\$SG_EXI, EDT\$SG_PST_CNT
.EXTRN EDT\$SA_RPL_STR, EDT\$SG_RPL_LEN
.EXTRN EDT\$SG_SEA_STRLEN
.EXTRN EDT\$SA_SEA_STR, EDT\$SA_SEL_BUF
.EXTRN EDT\$SA_OLD_SEL, EDT\$SL_SEL_LN
.EXTRN EDT\$SA_SEL_POS, EDT\$SG_SHF
.EXTRN EDT\$SG_TRUN, EDT\$SG_SEA_LEN
.EXTRN EDT\$SL_TOP_LN, EDT\$SG_VERT
.EXTRN EDT\$SG_EXITD, EDT\$SG_SCR_CHGD
.EXTRN EDT\$SG_SCR_REBUILD
.EXTRN EDT\$SG_TAB_SIZ, EDT\$SA_CUR_BUF
.EXTRN EDT\$SG_TAB_LVL, EDT\$SG_TI_TYP
.EXTRN EDT\$ST_FMT_BUF, EDT\$SA_FMT_CUR
.EXTRN EDT\$ST_LN_BUF, EDT\$SA_LN_PTR
.EXTRN EDT\$SG_DFCT_VERB
.EXTRN EDT\$SG_CC_DONE, EDT\$SELALRACT
.EXTRN EDT\$_INVSUBCOM, EDT\$_CLDNOTALN
.EXTRN EDT\$_INVASCCHR, EDT\$_INVSTR
.EXTRN EDT\$INTER_ERR

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

.ENTRY EDT\$SEXE_CHMCMD1, Save R2,R3,R4,R5,R6,R7,- : 0693
R8,R9,R10,R11
MOVAB EDT\$SA_CUR_BUF, R11
MOVAB EDT\$SA_SEL_BUF, R10
MOVAB EDT\$SG_SHF, R9
SUBL2 #44, SP
MOVL VERB, R8 : 0878
CML R8, #12
BNEQ 1\$
PUSHL EDT\$SG_RPL_LEN : 0883
PUSHL EDT\$SA_RPL_STR : 0882
PUSHL EDT\$SG_SEA_LEN : 0881
PUSHL EDT\$SA_SEA_STR : 0880
CALLS #4, EDT\$SET_SEASUBSTR
MOVL R0, SUCCEED
BRB 2\$
MOVL #1, SUCCEED : 0885
BLBS SUCCEED, 3\$: 0887
PUSHL #EDT\$_INVSTR : 0889
BRW 60\$
CASEL R8, #11, #34 : 0895
.WORD 23\$-4\$,-
42\$-4\$,-
21\$-4\$,-
10\$-4\$,-
11\$-4\$,-
12\$-4\$,-
19\$-4\$,-
42\$-4\$,-

OFFC 00000

5B 00000000G 00 9E 00002
5A 00000000G 00 9E 00009
59 00000000G 00 9E 00010
5E 2C C2 00017
58 04 AC D0 0001A
0C 58 D1 0001E
24 12 00021
00000000G 00 DD 00023
00000000G 00 DD 00029
00000000G 00 DD 0002F
00000000G 00 DD 00035
00000000G 00 04 FB 0003B
00 57 50 D0 00042
03 11 00045
57 01 D0 00047 1\$:
09 57 E8 0004A 2\$:
00000000G 8F DD 0004D
02C9 31 00053
58 CF 00056 3\$:
0118 0005A 4\$:
00A0 00062
0050 0006A
00CD 00072
01B2 0007A
0259 00082
02FF 0008A
016A 00092

0081 22
01E6 010F 01E6
00F8 00F8 00B2
00D5 006C 005E
018D 017F 0158
0233 021E 00F8
02F6 02E1 0290
0161 0173 01F0
00F0 00DE 00E7

03D7

03A4

03D3

0009A

6\$-4\$,-
7\$-4\$,-
8\$-4\$,-
15\$-4\$,-
14\$-4\$,-
27\$-4\$,-
34\$-4\$,-
36\$-4\$,-
38\$-4\$,-
19\$-4\$,-
49\$-4\$,-
50\$-4\$,-
53\$-4\$,-
59\$-4\$,-
64\$-4\$,-
66\$-4\$,-
68\$-4\$,-
44\$-4\$,-
33\$-4\$,-
29\$-4\$,-
31\$-4\$,-
17\$-4\$,-
16\$-4\$,-
18\$-4\$,-
74\$-4\$,-
72\$-4\$,-
75\$-4\$,-

00000000G	00	00	FB	000A0	CALLS	#0, EDT\$\$INTER_ERR	1266
		038D	31	000A7	BRW	76\$	0895
	00000000G	00	DD	000AA	PUSHL	EDT\$\$G_DEL_CHLEN	0900
	00000000G	00	9F	000B0	PUSHAB	EDT\$\$T_DEL_CH	
		1A	11	000B6	BRB	9\$	
	00000000G	00	DD	000B8	PUSHL	EDT\$\$G_DEL_WDLEN	0905
	00000000G	00	9F	000BE	PUSHAB	EDT\$\$T_DEL_WD	
		0C	11	000C4	BRB	9\$	
	00000000G	00	DD	000C6	PUSHL	EDT\$\$G_DEL_LNLEN	0910
	00000000G	00	9F	000CC	PUSHAB	EDT\$\$T_DEL_LN	
00000000G	00	02	FB	000D2	CALLS	#2, EDT\$\$UNDL	
		4A	11	000D9	BRB	13\$	
	00000000G	00	DD	000DB	PUSHL	EDT\$\$G_SEA_LEN	0915
		0C	AC	DD	PUSHL	OPERAND	
	00000000G	00	02	FB	CALLS	#2, EDT\$\$INS_CHS	
		57	50	DD	MOVL	R0, SUCCEED	
	57		57	E9	BLBC	SUCCEED, 5\$	0917
00000000G	00	00	FB	000F1	CALLS	#0, EDT\$\$WORD_WRAP	
		7\$	11	000F8	BRB	22\$	
	00000000G	00	DD	000FA	PUSHL	EDT\$\$G_SEA_LEN	0926
		0C	AC	DD	PUSHL	OPERAND	
00000000G	00	02	FB	00103	CALLS	#2, EDT\$\$GET_XLATE	
		64	11	0010A	BRB	22\$	
		01	DD	0010C	PUSHL	#1	0936
	04	AE	BC	9A	MOVZBL	@OPERAND, 4(SP)	
	04	AE	8F	C2	SUBL2	#64, 4(SP)	
		04	AE	9F	PUSHAB	4(SP)	
00000000G	00	02	FB	0011E	CALLS	#2, EDT\$\$INS_CHS	
		49	11	00125	BRB	22\$	
	00000000G	00	D4	00127	CLRL	EDT\$\$G_DIR_MOD	0941

			56	11	0012D	BRB	24\$		0940	
			01	D0	0012F	15\$:	MOVL	#1, EDT\$G_DIR_MOD	0947	
			78	11	00136	BRB	26\$		0946	
			05	D0	00138	16\$:	MOVL	#5, EDT\$G_DFLT_VERB	0956	
			78	11	0013F	BRB	28\$		0952	
			04	D0	00141	17\$:	MOVL	#4, EDT\$G_DFLT_VERB	0967	
			78	11	00148	BRB	30\$		0963	
		00000000G	00	D4	0014A	18\$:	CLRL	EDT\$G_DFLT_VERB	0978	
			70	11	00150	BRB	30\$		0974	
			01	D0	00152	19\$:	MOVL	#1, EDT\$G_EXI	0986	
			58	D1	00159	CMPL	R8, #28		0988	
			07	12	0015C	BNEQ	20\$			
			01	D0	0015E	MOVL	#1, EDT\$G_EXITD			
			01	D0	00165	20\$:	MOVL	#1, R0	0990	
			04	00168	RET					
			00	FB	00169	21\$:	CALLS	#0, EDT\$PST_CMD	0995	
			59	11	00170	22\$:	BRB	32\$		
			6A	D5	00172	23\$:	TSTL	EDT\$A_SEL_BUF	1001	
			11	13	00174	BEQL	25\$			
		00000000G	8F	DD	00176	PUSHL	#EDT\$ SELALRACT		1004	
			01	FB	0017C	CALLS	#1, EDT\$MSG_BELL			
			57	D4	00183	CLRL	SUCCEED		1005	
			5E	11	00185	24\$:	BRB	35\$	1001	
			6B	D0	00187	25\$:	MOVL	EDT\$A_CUR_BUF, R6	1009	
00000000G	00	06	A6	06	28	0018A	MOVC3	#6, 6(R6), -EDT\$SEL_SEL_LN		
			6A	56	D0	00193	MOVL	R6, EDT\$A_SEL_BUF	1010	
			00	D0	00196	MOVL	EDT\$A_LN_PTR, -EDT\$A_SEL_POS		1011	
		00000000G	00	D5	001A1	TSTL	EDT\$A_OLD_SEL		1013	
			61	13	001A7	BEQL	37\$			
			01	D0	001A9	MOVL	#1, EDT\$G_SCR_REBUILD			
			58	11	001B0	26\$:	BRB	37\$	0999	
			02	D0	001B2	27\$:	MOVL	#2, EDT\$G_SCR_CHGD	1022	
			4F	11	001B9	28\$:	BRB	37\$	1021	
			00	FB	001BB	29\$:	CALLS	#0, EDT\$RING_BELL	1031	
			79	11	001C2	30\$:	BRB	41\$	1027	
			00	FB	001C4	31\$:	CALLS	#0, EDT\$INS_DATIM	1042	
			7A	11	001CB	32\$:	BRB	43\$		
			00	FB	001CD	33\$:	CALLS	#0, EDT\$LN_DEFK	1052	
			57	50	D0	001D4	MOVL	R0, SUCCEED		
			64	11	001D7	BRB	41\$		1051	
			50	6B	D0	001D9	34\$:	MOVL	EDT\$A_CUR_BUF, R0	1058
00000000G	00	06	A0	06	28	001DC	MOVC3	#6, 6(R0), -EDT\$SEL_TOP_LN		
			56	11	001E5	35\$:	BRB	41\$	1057	
			00	FB	001E7	36\$:	CALLS	#0, EDT\$SC_NONREVID	1068	
			00	FB	001EE	CALLS	#0, EDT\$STOP_WKINGMSG		1069	
			00	FB	001F5	CALLS	#0, EDT\$KPAD_HLP		1070	
			01	D0	001FC	MOVL	#1, EDT\$G_SCR_CHGD		1071	
			50	D5	00203	TSTL	KPAD_STATUS		1073	
			36	12	00205	BNEQ	41\$			
			02	D0	00207	MOVL	#2, SUCCEED			
			31	11	0020A	37\$:	BRB	41\$	1063	
			08	AC	D1	0020C	38\$:	CMPL	COUNT, #255	1081
			05	14	00214	BGTR	39\$			
			08	AC	D5	00216	TSTL	COUNT		
			11	18	00219	BGEQ	40\$			
			8F	DD	0021B	39\$:	PUSHL	#EDT\$ INVASCCHR	1084	
			01	FB	00221	CALLS	#1, EDT\$MSG_BELL			

			57	D4	00228	CLRL	SUCCEED	1085		
			4A	11	0022A	BRB	48\$	1081		
			01	DD	0022C	40\$: PUSHL	#1	1089		
04	AE	08	AC	D0	0022E	MOVL	COUNT, 4(SP)			
		04	AE	9F	00233	PUSHAB	4(SP)			
00000000G	00		02	FB	00236	CALLS	#2, EDT\$\$INS_STR			
			0224	31	0023D	41\$: BRW	79\$	1088		
00000000G	00		00	FB	00240	42\$: CALLS	#0, EDT\$\$SUB_CMD	1097		
			009B	31	00247	43\$: BRW	57\$			
			00000000G	00	D5	0024A	44\$: TSTL	EDT\$\$G_DIR	1106	
				1D	12	00250	BNEQ	47\$		
			00000000G	00	D5	00252	TSTL	EDT\$\$G_PST_CNT		
				15	13	00258	BEQL	47\$		
			52	00000000G	00	D0	0025A	MOVL	EDT\$\$G_PST_CNT, I	1109
				07	11	00261	BRB	46\$		
00000000G	00		00	FB	00263	45\$: CALLS	#0, EDT\$\$CS_LEFT	1110		
	F6		52	F4	0026A	46\$: SOBGEQ	I, 45\$			
			79	11	0026D	BRB	58\$	1109		
00000000G	00		00	FB	0026F	47\$: CALLS	#0, EDT\$\$CS_LEFT	1113		
			70	11	00276	48\$: BRB	58\$	0895		
	69		08	C0	00278	49\$: ADDL2	#8, EDT\$\$G_SHF	1119		
	50		69	D0	0027B	MOVL	EDT\$\$G_SHF, R0	1121		
00007FFF	8F		50	D1	0027E	CMPL	R0, #32767			
			19	18	00285	BGEQ	51\$			
			50	D5	00287	TSTL	R0			
			15	19	00289	BLSS	51\$			
			15	11	0028B	BRB	52\$	1123		
	69		08	C2	0028D	50\$: SUBL2	#8, EDT\$\$G_SHF	1131		
	50		69	D0	00290	MOVL	EDT\$\$G_SHF, R0	1133		
00007FFF	8F		50	D1	00293	CMPL	R0, #32767			
			04	18	0029A	BGEQ	51\$			
			50	D5	0029C	TSTL	R0			
			02	18	0029E	BGEQ	52\$			
			69	D4	002A0	51\$: CLRL	EDT\$\$G_SHF			
00000000G	00		01	D0	002A2	52\$: MOVL	#1, EDT\$\$G_SCR_CHGD	1135		
	38	00000000G	00	E8	002A9	BLBS	EDT\$\$G_TRUN, 58\$	1137		
			0171	31	002B0	BRW	73\$			
	50	00000000G	00	9E	002B3	53\$: MOVAB	EDT\$\$T_LN_BUF, R0	1147		
	50	00000000G	00	D1	002BA	CMPL	EDT\$\$A_LN_PTR, R0			
			08	12	002C1	BNEQ	54\$			
			00000000G	00	D5	002C3	TSTL	EDT\$\$G_TAB_SIZ	1148	
				05	12	002C9	BNEQ	55\$		
	50		08	D0	002CB	54\$: MOVL	#8, TAB_COUNT	1150		
			0C	11	002CE	BRB	56\$			
50	00000000G	00	00000000G	00	C5	002D0	55\$: MULL3	EDT\$\$G_TAB_SIZ, EDT\$\$G_TAB_LVL, TAB_COUNT	1152	
				50	DD	002DC	56\$: PUSHL	TAB_COUNT	1154	
00000000G	00		01	FB	002DE	CALLS	#1, EDT\$\$MOV_TOCOL			
	57		50	D0	002E5	57\$: MOVL	R0, SUCCEED			
			6C	11	002E8	58\$: BRB	67\$	0895		
			00000000G	00	D5	002EA	59\$: TSTL	EDT\$\$G_TAB_SIZ	1164	
				46	13	002F0	BEQL	63\$		
			04	AE	9F	002F2	PUSHAB	COL	1166	
			0C	AE	9F	002F5	PUSHAB	LIN		
00000000G	00		02	FB	002F8	CALLS	#2, EDT\$\$SC_CPUCSPOS			
	04		69	C0	002FF	ADDL2	EDT\$\$G_SHF, COL	1167		
	52	00000000G	00	D0	00303	MOVL	EDT\$\$G_TAB_SIZ, R2	1169		
7E	00	04	AE	01	7A	0030A	EMUL	#1, COL, #0, -(SP)		

50	50	8E	52	7B	00310	EDIV	R2, (SP)+, R0, R0	:	
			50	D5	00315	TSTL	R0	:	
			0F	13	00317	BEQL	61\$	:	
		00000000G	8F	DD	00319	PUSHL	#EDT\$ CLDNOTALN	:	1171
			01	FB	0031F	CALLS	#1, EDT\$\$MSG_BELL	:	
			10	11	00326	BRB	63\$	:	
		50	04	AE	D0	00328	MOVL	COL, R0	1173
			02	18	0032C	BGEQ	62\$	:	
			50	D4	0032E	CLRL	R0	:	
00000000G	00	50	52	C7	00330	DIVL3	R2, R0, EDT\$\$G_TAB_LVL	:	
			0129	31	00338	BRW	79\$	:	1158
	50	00000000G	00	01	C3	0033B	SUBL3	#1, EDT\$\$G_TAB_LVL, R0	1180
			02	18	00343	BGEQ	65\$	:	
			50	D4	00345	CLRL	R0	:	
		00000000G	00	50	D0	00347	MOVL	R0, EDT\$\$G_TAB_LVL	
			06	11	0034E	BRB	67\$	:	0895
		00000000G	00	D6	00350	INCL	EDT\$\$G_TAB_LVL	:	1185
			00DE	31	00356	BRW	76\$	:	0895
		00000000G	00	00	FB	00359	CALLS	#0, EDT\$\$SC_FULLSCLL	1190
			50	D0	00360	MOVL	EDT\$\$G_TI_TYP, R0	:	1192
			01	50	D1	00367	CMPL	R0, #1	
			02	50	D1	0036C	BEQL	69\$	
			0D	12	0036F	CMPL	R0, #2	:	1193
		00000000G	00	00	9E	00371	BNEQ	70\$	
			12	11	0037C	MOVAB	EDT\$\$TI_WRSTR, EDT\$\$A_FMT_WRRUT	:	1195
			00	FB	0037E	BRB	71\$	:	
		00000000G	00	00	FB	0037E	CALLS	#0, EDT\$\$STOP_WKINGMSG	1198
		00000000G	00	00	9E	00385	MOVAB	EDT\$\$TI_WRLN, EDT\$\$A_FMT_WRRUT	1199
		00000000G	00	00	FB	00390	CALLS	#0, EDT\$\$RPL_CHGDLN	1202
			50	6B	D0	00397	MOVL	EDT\$\$A_CUR_BUF, R0	1203
			51	00	9E	0039A	MOVAB	EDT\$\$T_LN_BUF, R1	
OC	AO	00000000G	00	51	A3	003A1	SUBW3	R1, EDT\$\$A_LN_PTR, 12(R0)	
		00000000G	00	01	D0	003AA	MOVL	#1, EDT\$\$G_EXT_MOD	1204
		00000000G	00	00	FB	003B1	CALLS	#0, EDT\$\$EXT_CMD	1205
			00	D4	003B8	CLRL	EDT\$\$G_EXT_MOD	:	1206
		00000000G	00	00	9E	003BE	MOVAB	EDT\$\$T_FMT_BUF, EDT\$\$A_FMT_CUR	1207
		00000000G	00	00	FB	003C9	CALLS	#0, EDT\$\$GET_TXTLN	1208
			50	6B	D0	003D0	MOVL	EDT\$\$A_CUR_BUF, R0	1209
			51	00	9E	003D3	MOVAB	EDT\$\$T_LN_BUF, R1	
			52	OC	A0	003DA	MOVZWL	12(R0), R2	
00000000G	00		51	52	C1	003DE	ADDL3	R2, R1, EDT\$\$A_LN_PTR	
			77	00	E9	003E6	BLBC	EDT\$\$G_TI_SCROLL, -79\$	1211
		00000000G	00	00	DD	003ED	PUSHL	EDT\$\$G_SCR_LNS	
			7E	D4	003F3	CLRL	-(SP)	:	
		00000000G	00	02	FB	003F5	CALLS	#2, EDT\$\$SC_SETSCLLREG	
			66	11	003FC	BRB	79\$	:	1189
			6A	D5	003FE	TSTL	EDT\$\$A_SEL_BUF	:	1234
			2B	12	00400	BNEQ	74\$	:	
			6B	D0	00402	MOVL	EDT\$\$A_CUR_BUF, R6	:	1237
00000000G	00	06	A6	06	28	00405	MOVC3	#6, 6(R6), EDT\$\$L_SEL_LN	
			6A	56	D0	0040E	MOVL	R6, EDT\$\$A_SEL_BUF	1238
		00000000G	00	00	D0	00411	MOVL	EDT\$\$A_LN_PTR, EDT\$\$A_SEL_POS	1239
			00	00	D5	0041C	TSTL	EDT\$\$A_OLD_SEL	1241
			13	13	00422	BEQL	76\$	:	
		00000000G	00	01	D0	00424	MOVL	#1, EDT\$\$G_SCR_REBUILD	
			0A	11	0042B	BRB	76\$	:	1234
			6A	D4	0042D	CLRL	EDT\$\$A_SEL_BUF	:	1245

EDT\$CHMEXCOM
V04-000

H 4
EDT\$CHMEXCOM - execute certain change-mode comm 15-Sep-1984 23:52:48
EDT\$\$EXE_CHMCMD1 - execute certain change-mode 14-Sep-1984 12:22:29

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]CHMEXCOM.BLI;1 Page 20
(3)

		06	11	0042F	BRB	76\$		: 0895
	00000000G	00	D4	00431	CLRL	EDT\$\$G_SEA_STRLEN		: 1258
	10	AC	D5	00437	TSTL	EXPLICIT		: 1269
		03	13	0043A	BEQL	77\$		: 1271
	08	AC	D7	0043C	DECL	COUNT		: 1273
01		57	D1	0043F	CMPL	SUCCEED, #1		: 1277
		20	12	00442	BNEQ	79\$		: 1275
00000000G	00	00	FB	00444	CALLS	#0, EDT\$\$CHK_CC		: 1283
0E		50	E9	0044B	BLBC	R0, 78\$		: 1289
	08	AC	D5	0044E	TSTL	COUNT		: 1291
		11	15	00451	BLEQ	79\$		: 1292
00000000G	00	01	D0	00453	MOVL	#1, EDT\$\$G_CC_DONE		: 1291
		08	11	0045A	BRB	79\$		: 1292
	08	AC	D5	0045C	TSTL	COUNT		: 1291
		03	15	0045F	BLEQ	79\$		: 1292
		FBF2	31	00461	BRW	3\$		: 1289
16		58	D1	00464	CMPL	R8, #22		: 1291
		0B	13	00467	BEQL	80\$		: 1292
17		58	D1	00469	CMPL	R8, #23		: 1291
		06	13	0046C	BEQL	80\$		: 1292
	00000000G	00	D4	0046E	CLRL	EDT\$\$G_VERT		: 1291
50		57	D0	00474	MOVL	SUCCEED, R0		: 1292
		04	00477	RET				: 1292

; Routine Size: 1144 bytes, Routine Base: _EDT\$CODE + 0000

: 723 1293 1
: 724 1294 1 !<BLF/PAGE>

EDT\$CHMEXCOM
V04-000

EDT\$CHMEXCOM - execute certain change-mode comm 15-Sep-1984 23:52:48
EDT\$EXE_CHMCMD1 - execute certain change-mode 14-Sep-1984 12:22:29

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]CHMEXCOM.BLI;1 (4)

Page 21

: 726 1295 1 END
: 727 1296 1
: 728 1297 0 ELUDOM

! of module EDT\$CHMEXCOM

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	1144	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	86	22	40	00:00.2
_\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1
_\$255\$DUA28:[EDT.SRC]SUPPORTS.L32;1	2	1	50	5	00:00.1

COMMAND QUALIFIERS

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

: Size: 1144 code + 0 data bytes
: Run Time: 00:40.7
: Elapsed Time: 00:47.8
: Lines/CPU Min: 1911
: Lexemes/CPU-Min: 6328
: Memory Used: 277 pages
: Compilation Complete

0131 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY