


```

LL      TTTTTTTTTT  YY      YY  PPPPPPPP  EEEEEEEEEE
LL      TTTTTTTTTT  YY      YY  PPPPPPPP  EEEEEEEEEE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PPPPPPPP  EEEEEEEEEE
LL      TT          YY      YY  PPPPPPPP  EEEEEEEEEE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PP        PP  EE
LL      TT          YY      YY  PP        PP  EE
LLLLLLLLLLLL  TT          YY      YY  PP        PP  EEEEEEEEEE
LLLLLLLLLLLL  TT          YY      YY  PP        PP  EEEEEEEEEE

```

```

LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SSSSSS
LL      II          SSSSSS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SS
LLLLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLLLL  IIIIII  SSSSSSSS

```

```

1 0001 0 %TITLE 'EDT$LTYPE - TYPE and null line-mode commands'
2 0002 0 MODULE EDT$LTYPE ( 'TYPE and null line-mode commands
3 0003 0 IDENT = 'V04-000' ! File: LTYPE.BLI Edit: SMB1015
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
11 0011 1 * ALL RIGHTS RESERVED. *
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
18 0018 1 * TRANSFERRED. *
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
22 0022 1 * CORPORATION. *
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1
31 0031 1 **
32 0032 1 FACILITY: EDT -- The DEC Standard Editor
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 This module executes the line mode TYPE and null commands.
37 0037 1
38 0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Bob Kushlis, CREATION DATE: February 3, 1978
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. DJS 30-JAN-1981. This module was created by
45 0045 1 extracting the routines EDT$$TY_CMD and NULL_COM from the
46 0046 1 module EXEC.BLI.
47 0047 1 1-002 - Regularize headers. JBS 20-Mar-1981
48 0048 1 1-003 - Change NULL_COM to EDT$$NULL_CMD. JBS 30-Mar-1981
49 0049 1 1-005 - Switch back to first buffer if /STAY. STS 21-Apr-1982
50 0050 1 1-006 - Add a check for CNTRL/C. SMB 19-May-1982
51 0051 1 1-007 - Set a flag if control C actually aborts something. JBS 25-May-1982
52 0052 1 1-008 - Stop working message before printing to screen. SMB 22-Jun-1982
53 0053 1 1-009 - Stop processing on bad select range. SMB 01-Jul-1982
54 0054 1 1-010 - Undo yesterday's fix because we want to print current line. SMB 02-Jul-1982
55 0055 1 1-011 - Don't fool with the formatting routine. JBS 05-Jul-1982
56 0056 1 1-012 - Make the call to EDT$$RNG NUMLEN in line. STS 22-Sep-1982
57 0057 1 1-013 - Correct a typo in edit 1-012. JBS 30-Sep-1982

```

EDT\$LTYPE
V04-000

EDT\$LTYPE - TYPE and null line-mode commands

D 16
16-Sep-1984 01:03:05
14-Sep-1984 12:23:45

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

Page 2
(1)

: 58
: 59
: 60
: 61

0058 1 ! 1-014 - Put code for edt\$\$rng_retfst in line. STS 11-Oct-1982
0059 1 ! 1-015 - Remove setting of G_TXT_ONSCR. SMB 14-Dec-1982
0060 1 !--
0061 1

```
: 63      0062  1 %SBTTL 'Declarations'  
: 64      0063  1  
: 65      0064  1 : TABLE OF CONTENTS:  
: 66      0065  1 :  
: 67      0066  1  
: 68      0067  1 REQUIRE 'EDTSRC:TRAROUNAM';  
: 69      0506  1  
: 70      0507  1 FORWARD ROUTINE  
: 71      0508  1     EDT$TY_CMD : NOVALUE,           ! Process the TYPE command  
: 72      0509  1     EDT$NUCL_CMD : NOVALUE;       ! Process the null command  
: 73      0510  1  
: 74      0511  1  
: 75      0512  1 : INCLUDE FILES:  
: 76      0513  1 :  
: 77      0514  1  
: 78      0515  1 REQUIRE 'EDTSRC:EDTREQ';  
: 79      0650  1  
: 80      0651  1  
: 81      0652  1 : MACROS:  
: 82      0653  1 :  
: 83      0654  1 :     NONE  
: 84      0655  1 :  
: 85      0656  1 : EQUATED SYMBOLS:  
: 86      0657  1 :  
: 87      0658  1 :     NONE  
: 88      0659  1 :  
: 89      0660  1 : OWN STORAGE:  
: 90      0661  1 :  
: 91      0662  1 :     NONE  
: 92      0663  1 :  
: 93      0664  1 : EXTERNAL REFERENCES:  
: 94      0665  1 :  
: 95      0666  1 :     In the routines
```

```

97 0667 1 %SBTTL 'EDT$$TY_CMD - TYPE line-mode command'
98 0668 1
99 0669 1 GLOBAL ROUTINE EDT$$TY_CMD          ! TYPE line-mode command
100 0670 1      : NOVALUE =
101 0671 1
102 0672 1 |++
103 0673 1 | FUNCTIONAL DESCRIPTION:
104 0674 1 |
105 0675 1 |     Command processing routine for the TYPE command.  Position to
106 0676 1 |     the first line of the range, then loop through the range
107 0677 1 |     displaying each line.  If the /STAY qualifier was used,
108 0678 1 |     then reposition to the position at the start of the command,
109 0679 1 |     otherwise, reposition to the first line typed.
110 0680 1
111 0681 1 FORMAL PARAMETERS:
112 0682 1
113 0683 1     NONE
114 0684 1
115 0685 1 IMPLICIT INPUTS:
116 0686 1
117 0687 1     EDT$$A_CUR_BUF
118 0688 1     EDT$$Z_RNG_ORIGPOS
119 0689 1     EDT$$A_EXE_CURCMD
120 0690 1     EDT$$G_EXE_SBITS
121 0691 1     EDT$$G_RNG_NOOFLN
122 0692 1
123 0693 1 IMPLICIT OUTPUTS:
124 0694 1
125 0695 1     EDT$$A_CUR_BUF
126 0696 1     EDT$$G_CC_DONE
127 0697 1
128 0698 1 ROUTINE VALUE:
129 0699 1
130 0700 1     NONE
131 0701 1
132 0702 1 SIDE EFFECTS:
133 0703 1
134 0704 1     NONE
135 0705 1
136 0706 1 --
137 0707 1
138 0708 2 BEGIN
139 0709 2
140 0710 2 EXTERNAL ROUTINE
141 0711 2     EDT$$STOP_WKINGMSG,          ! Stop the working message
142 0712 2     EDT$$CHK_CC,                ! Check for CONTROL C typed
143 0713 2     EDT$$NXT_LNRNG,
144 0714 2     EDT$$RNG_REPOS,
145 0715 2     EDT$$RD_CURLN,
146 0716 2     EDT$$TY_CURLN;
147 0717 2
148 0718 2 EXTERNAL
149 0719 2     EDT$$G_EXT_MOD,                ! 1 = we are under the EXT command
150 0720 2     EDT$$G_RNG_FRSTLN,
151 0721 2     EDT$$A_CUR_BUF : REF TBCB_BLOCK,
152 0722 2     EDT$$Z_RNG_ORIGPOS : POS_BLOCK,
153 0723 2     EDT$$A_EXE_CURCMD : REF NODE_BLOCK, ! Pointer to the current command.

```

```

154 0724 2      EDT$$Z_RNG_SAVPOS : POS_BLOCK,
155 0725 2      EDT$$G_EXE_SBITS,           ! The options switches.
156 0726 2      EDT$$G_CC_DONE,           ! Set to 1 if control C actually aborts something
157 0727 2      EDT$$G_RNG_NOOFLN;       ! number of lines processed in a range
158 0728 2
159 0729 2      LOCAL
160 0730 2      SAV_BUF,
161 0731 2      STATUS;
162 0732 2
163 0733 2      SAV_BUF = .EDT$$A_CUR_BUF;
164 0734 2      EDT$$G_RNG_FRSTLN = 1;
165 0735 2      EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
166 0736 2      EDT$$RNG_REPOS (.EDT$$A_EXE_CURCMD [RANGE1]);
167 0737 2      STATUS = 1;
168 0738 2
169 0739 2      ! Turn off the 'working' message if it might be on, since the type
170 0740 2      ! command doesn't want competition with a flashing message.
171 0741 2      !-
172 0742 2
173 0743 2      IF (.EDT$$G_EXT_MOD) THEN EDT$$STOP_WKINGMSG ();
174 0744 2
175 0745 2      WHILE .STATUS DO
176 0746 2      BEGIN
177 0747 3      STATUS = EDT$$NXT_LNRNG (0);
178 0748 3
179 0749 3      IF .STATUS
180 0750 3      THEN
181 0751 4      BEGIN
182 0752 4      STATUS = ( NOT EDT$$CHK_CC ());
183 0753 4
184 0754 4      IF .STATUS THEN EDT$$TY_CURLN () ELSE EDT$$G_CC_DONE = 1;
185 0755 4
186 0756 3      END;
187 0757 3
188 0758 2      END;
189 0759 2
190 0760 2      IF (.EDT$$G_RNG_NOOFLN EQL 0) THEN EDT$$TY_CURLN ();
191 0761 2
192 0762 3      IF ( NOT .EDT$$G_EXE_SBITS<OPB_STAY>)
193 0763 3      THEN
194 0764 3      BEGIN
195 0765 3      EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_SAVPOS, .EDT$$A_CUR_BUF);
196 0766 3      EDT$$RD_CURLN ();
197 0767 3      END
198 0768 2      ELSE
199 0769 3      BEGIN
200 0770 3      EDT$$A_CUR_BUF = .SAV_BUF;
201 0771 3      EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_ORIGPOS, .EDT$$A_CUR_BUF);
202 0772 3      EDT$$RD_CURLN ();
203 0773 2      END;
204 0774 2
205 0775 1      END;

```

! of routine EDT\$TY_CMD

.TITLE EDT\$TYPE EDT\$TYPE - TYPE and null line-mode c
ommands
.IDENT \V04-000\

EDTSLTYPE
V04-000

EDTSLTYPE - TYPE and null line-mode commands
EDTSTY_CMD - TYPE line-mode command

I 16
16-Sep-1984 01:03:05
14-Sep-1984 12:23:45

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]LTYPE.BLI;1 Page 7 (3)

: 206

0776 1

```
208 0777 1 %SBTTL 'EDT$$NULL_CMD - null line-mode command'
209 0778 1
210 0779 1 GLOBAL ROUTINE EDT$$NULL_CMD ! Null line-mode command
211 0780 1 : NOVALUE =
212 0781 1
213 0782 1 ++
214 0783 1 FUNCTIONAL DESCRIPTION:
215 0784 1
216 0785 1 Command processing routine for the null command. If the range is '.'
217 0786 1 then move one line forward and display it. Otherwise, process it as
218 0787 1 a TYPE comamnd.
219 0788 1
220 0789 1 FORMAL PARAMETERS:
221 0790 1
222 0791 1 NONE
223 0792 1
224 0793 1 IMPLICIT INPUTS:
225 0794 1
226 0795 1 EDT$$A_EXE_CURCMD
227 0796 1
228 0797 1 IMPLICIT OUTPUTS:
229 0798 1
230 0799 1 NONE
231 0800 1
232 0801 1 ROUTINE VALUE:
233 0802 1
234 0803 1 NONE
235 0804 1
236 0805 1 SIDE EFFECTS:
237 0806 1
238 0807 1 NONE
239 0808 1
240 0809 1 --
241 0810 1
242 0811 2 BEGIN
243 0812 2
244 0813 2 EXTERNAL ROUTINE
245 0814 2 EDT$$RD_NXTLN,
246 0815 2 EDT$$TY_CURLN;
247 0816 2
248 0817 2 EXTERNAL
249 0818 2 EDT$$A_EXE_CURCMD : REF NODE_BLOCK; ! Pointer to the current command.
250 0819 2
251 0820 2 BIND
252 0821 2 RANGE = .EDT$$A_EXE_CURCMD [RANGE1] : NODE_BLOCK;
253 0822 2
254 0823 3 IF (.RANGE [RAN_TYPE] EQL RAN_NULL)
255 0824 2 THEN
256 0825 3 BEGIN
257 0826 3 EDT$$RD_NXTLN ();
258 0827 3 EDT$$TY_CURLN ();
259 0828 3 END
260 0829 2 ELSE
261 0830 2 EDT$$TY_CMD ();
262 0831 2
263 0832 1 END; ! of routine EDT$$NULL_CMD
```

EDT\$TYPE
V04-000

EDT\$TYPE - TYPE and null line-mode commands
EDT\$\$NULL_CMD - null line-mode command

K 16
16-Sep-1984 01:03:05
14-Sep-1984 12:23:45

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

Page 9
(4)

					.EXTRN EDT\$\$RD_NXTLN	
			0000 00000		.ENTRY EDT\$\$NULL_CMD, Save nothing	: 0779
	50	00000000G	00 D0 00002		MOVL EDT\$\$A_EXE_CURCMD, R0	: 0821
	50		04 A0 D0 00009		MOVL 4(R0), R0	:
			01 A0 95 0000D		TSTB 1(RC)	: 0823
			0F 12 00010		BNEQ !\$:
	00000000G	00	00 FB 00012		CALLS #0, EDT\$\$RD_NXTLN	: 0826
	00000000G	00	00 FB 00019		CALLS #0, EDT\$\$TY_CURLN	: 0827
			04 00020		RET	:
	FF31	CF	00 FB 00021 1\$:		CALLS #0, EDT\$\$TY_CMD	: 0830
			04 00026		RET	: 0832

: Routine Size: 39 bytes, Routine Base: _EDT\$CODE + 00A9

: 264 0833 1
: 265 0834 1 !<BLF/PAGE>

EDT\$LTYPE
V04-000

EDT\$LTYPE - TYPE and null line-mode commands
EDT\$\$NULL_CMD - null line-mode command

L 16
16-Sep-1984 01:03:05
14-Sep-1984 12:23:45

VAX-11 Bliss-32 V4.0-742
DISK\$VMMASTER:[EDT.SRC]LTYPE.BLI;1

Page 10
(5)

: 267 0835 1 END
: 268 0836 1
: 269 0837 0 ELUDOM

! of module EDT\$LTYPE

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	208	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	84	22	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\$:LTYPE/OBJ=OBJ\$:LTYPE MSRC\$:LTYPE.BLI/UPDATE=(ENH\$:LTYPE)

: Size: 208 code + 0 data bytes
: Run Time: 00:14.3
: Elapsed Time: 00:16.9
: Lines/CPU Min: 3504
: Lexemes/CPU-Min: 10325
: Memory Used: 96 pages
: Compilation Complete

