

EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTTTTTTTTTTTTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEEEEEEEEEEEEE	DDD	TTT
EEEEEEEEEEEEEE	DDD	TTT
EEEEEEEEEEEEEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEE	DDD	TTT
EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTT
EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTT
EEEEEEEEEEEEEEEE	DDDDDDDDDDDD	TTT

EXE

MOC

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

ED

```

MM      MM      CCCCCCCC  GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT  LL      IIIIII  NN      NN
MM      MM      CCCCCCCC  GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT  LL      IIIIII  NN      NN
MMMM    MMMM    CC        GG        EE        TT        LL      II       NN      NN
MMMM    MMMM    CC        GG        EE        TT        LL      II       NN      NN
MM      MM      CC        GG        EE        TT        LL      II       NNNN   NN
MM      MM      CC        GG        EE        TT        LL      II       NNNN   NN
MM      MM      CC        GG        EEEEEEEE  TT        LL      II       NN      NN
MM      MM      CC        GG        EEEEEEEE  TT        LL      II       NN      NN
MM      MM      CC        GG      GGGGGG    TT        LL      II       NN      NN
MM      MM      CC        GG      GGGGGG    TT        LL      II       NN      NN
MM      MM      CC        GG      GG        TT        LL      II       NN      NN
MM      MM      CC        GG      GG        TT        LL      II       NN      NN
MM      MM      CC        GG      GG        TT        LL      II       NN      NN
MM      MM      CCCCCCCC  GGGGGG    EEEEEEEEEE  TT        LLLLLLLLLL  IIIIII  NN      NN
MM      MM      CCCCCCCC  GGGGGG    EEEEEEEEEE  TT        LLLLLLLLLL  IIIIII  NN      NN

```

```

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

```

```

....
....
....
....

```

```

1 0001 0 %TITLE 'EDT$MCGETLIN - get a text line'
2 0002 0 MODULE EDT$MCGETLIN ( ! Get a text line
3 0003 0 IDENT = 'V04-000' ! File: MCGETLIN.BLI Edit: JBS1004
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 Get a text line.
37 0037 1
38 0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Bob Kushlis, CREATION DATE: September 9, 1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. DJS 19-FEB-1981. This module was created by
45 0045 1 extracting routine EDT$$GET TXTLN from module MOVCUR.
46 0046 1 1-002 - Regularize headers. JBS T1-Mar-1981
47 0047 1 1-003 - Put edt$$tst_eob in line. STS 22-Sep-1982
48 0048 1 1-004 - Improve the appearance of the listing. JBS 17-Jun-1983
49 0049 1 --
50 0050 1

```

```
52 0051 1 %SBTTL 'Declarations'  
53 0052 1  
54 0053 1 : TABLE OF CONTENTS:  
55 0054 1 :  
56 0055 1  
57 0056 1 REQUIRE 'EDT$SRC:TRAROUNAM';  
58 0495 1  
59 0496 1 FORWARD ROUTINE  
60 0497 1 EDT$$GET_TXTLN : NOVALUE;  
61 0498 1  
62 0499 1 :  
63 0500 1 : INCLUDE FILES:  
64 0501 1 :  
65 0502 1  
66 0503 1 REQUIRE 'EDT$SRC:EDTREQ';  
67 0638 1  
68 0639 1 :  
69 0640 1 : MACROS:  
70 0641 1  
71 0642 1 : NONE  
72 0643 1 :  
73 0644 1 : EQUATED SYMBOLS:  
74 0645 1 :  
75 0646 1 : NONE  
76 0647 1 :  
77 0648 1 : OWN STORAGE:  
78 0649 1 :  
79 0650 1 : NONE  
80 0651 1 :  
81 0652 1 : EXTERNAL REFERENCES:  
82 0653 1 :  
83 0654 1 : In the routine
```

```
85 0655 1 %SBTTL 'EDTSSGET_TXTLN - get a text line'
86 0656 1
87 0657 1 GLOBAL ROUTINE EDTSSGET_TXTLN          ! Get a text line
88 0658 1   : NOVALUE =
89 0659 1
90 0660 1 !++
91 0661 1 ! FUNCTIONAL DESCRIPTION:
92 0662 1
93 0663 1     This routine fetches the current line from the work file into the
94 0664 1     current line buffer. The pointer EDTSSA_LN_PTR is updated to
95 0665 1     point to the cursor position and EDTSSA_LN_END and EDTSSG_LN_LEN
96 0666 1     are initialized. A carriage return character is placed after the
97 0667 1     last character in the buffer to facilitate search operations.
98 0668 1
99 0669 1 FORMAL PARAMETERS:
100 0670 1
101 0671 1     NONE
102 0672 1
103 0673 1 IMPLICIT INPUTS:
104 0674 1
105 0675 1     EDTSSA_CUR_BUF
106 0676 1     EDTSST_LN_BUF
107 0677 1     EDTSSA_WK_LN
108 0678 1
109 0679 1 IMPLICIT OUTPUTS:
110 0680 1
111 0681 1     EDTSSG_LN_CHGD
112 0682 1     EDTSSA_LN_PTR
113 0683 1     EDTSSA_LN_END
114 0684 1     EDTSSG_LN_LEN
115 0685 1
116 0686 1 ROUTINE VALUE:
117 0687 1
118 0688 1     NONE
119 0689 1
120 0690 1 SIDE EFFECTS:
121 0691 1
122 0692 1     NONE
123 0693 1
124 0694 1 --
125 0695 1
126 0696 2 BEGIN
127 0697 2
128 0698 2 EXTERNAL ROUTINE
129 0699 2     ED*SSRD_CURLN;
130 0700 2
131 0701 2 EXTERNAL
132 0702 2     EDTSSG_LN_CHGD,          ! Indicates current line has changed.
133 0703 2     EDTSSA_CUR_BUF : REF TBCB_BLOCK, ! Current text buffer
134 0704 2     EDTSST_LN_BUF,          ! Current line buffer.
135 0705 2     EDTSSA_LN_PTR,          ! Current character pointer.
136 0706 2     EDTSSA_LN_END,          ! End of current line pointer.
137 0707 2     EDTSSG_LN_LEN,          ! Length of current line.
138 0708 2     EDTSSZ_EOB_LN,
139 0709 2     EDTSSA_WK_LN : REF LIN_BLOCK; ! Current line pointer.
140 0710 2
141 0711 2 !+
```

```

142 0712 2 | Make sure the current line is accessible.
143 0713 2 |
144 0714 2 |   EDTSSRD_CURLN ();
145 0715 2 |
146 0716 2 | Check for end of buffer.
147 0717 2 |
148 0718 2 |
149 0719 2 |   IF (.EDTSSA_WK_LN EOLA EDTSSZ_EOB_LN)
150 0720 2 |   THEN
151 0721 2 |     EDTSSG_LN_LEN = 0
152 0722 2 |   ELSE
153 0723 2 |     BEGIN
154 0724 2 |
155 0725 2 |     Get the length and move that many characters.
156 0726 2 |
157 0727 2 |     EDTSSG_LN_LEN = .EDTSSA_WK_LN [LIN_LENGTH];
158 0728 2 |     EDTSSCPY_MEM (.EDTSSG_LN_LEN, EDTSSA_WK_LN [LIN_TEXT], EDTSST_LN_BUF);
159 0729 2 |     END;
160 0730 2 |
161 0731 2 |
162 0732 2 | Set up the end of line pointer.
163 0733 2 |
164 0734 2 |   EDTSSA_LN_END = CHSPTR (EDTSST_LN_BUF, .EDTSSG_LN_LEN);
165 0735 2 |
166 0736 2 | Place a CR at the end of the line.
167 0737 2 |
168 0738 2 |   CHSWCHAR (ASC_K_CR, .EDTSSA_LN_END);
169 0739 2 |
170 0740 2 | Set up the cursor pointer to point to the current character.
171 0741 2 |
172 0742 2 |   EDTSSA_LN_PTR = CHSPTR (EDTSST_LN_BUF, .EDTSSA_CUR_BUF [TBCB_CHAR_POS]);
173 0743 2 |
174 0744 2 | And mark the current line as unchanged.
175 0745 2 |
176 0746 2 |   EDTSSG_LN_CHGD = 0;
177 0747 1 |   END;

```

! of routine EDTSSGET\_TXTLN

.TITLE EDTSMCGETLIN EDTSMCGETLIN - get a text line  
.IDENT \V04-000\

.EXTRN EDTSSRD CURLN, EDTSSG\_LN\_CHGD  
.EXTRN EDTSSA\_CUR\_BUF, EDTSST\_LN\_BUF  
.EXTRN EDTSSA\_LN\_PTR, EDTSSA\_LN\_END  
.EXTRN EDTSSG\_LN\_LEN, EDTSSZ\_EOB\_LN  
.EXTRN EDTSSA\_WK\_LN

.PSECT \_EDTSCODE, NOWRT, SHR, PIC, 2

```

01FC 00000
58 00000000G 00 9E 00002
57 00000000G 00 9E 00009
56 00000000G 00 9E 00010
00000000G 00 FB 00017
51 00000000G 00 D0 0001E
50 00000000G 00 9E 00025
50 51 D1 0002C

```

```

.ENTRY EDTSSGET_TXTLN, Save R2,R3,R4,R5,R6,R7,R8 : 0657
MOVAB EDTSSA_LN_END, R8 :
MOVAB EDTSST_LN_BUF, R7 :
MOVAB EDTSSG_LN_LEN, R6 :
CALLS #0, EDTSSRD CURLN : 0714
MOVL EDTSSA_WK_LN, R1 : 0719
MOVAB EDTSSZ_EOB_LN, R0 :
CMPL R1, R0 :

```

EDTSMCGETLIN  
V04-000

EDTSMCGETLIN - get a text line  
EDTSSGET\_TXTLN - get a text line

G 9  
16-Sep-1984 01:10:30  
14-Sep-1984 12:23:55

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[EDT.SRC]MCGETLIN.BLI;1 Page 5 (3)

ED  
VO

			04	12	0002F	BNEQ	1\$		
			65	D4	00031	CLRL	EDTSSG_LN_LEN	:	0721
			08	11	00033	BRB	2\$	:	
		66	61	9A	00035	1\$:	MOVZBL	(R1), EDTSSG_LN_LEN	0727
67	07	A1	66	28	00038	MOVCL	EDTSSG_LN_LEN, 7(R1), EDTSST_LN_BUF	:	0728
		50	67	9E	0003D	2\$:	MOVAB	EDTSSST_LN_BUF, R0	0734
68		50	66	C1	00040	ADDL3	EDTSSG_LN_LEN, R0, EDTSSA_LN_END	:	
		50	68	D0	00044	MOVL	EDTSSA_LN_END, R0	:	0738
		60	0D	90	00047	MOVB	#13, (R0)	:	
		50	00	D0	0004A	MOVL	EDTSSA_CUR_BUF, R0	:	0742
		51	67	9E	00051	MOVAB	EDTSSST_LN_BUF, R1	:	
		52	0C	A0	00054	MOVZWL	12(R0), R2	:	
0000000G	00	51	52	C1	00058	ADDL3	R2, R1, EDTSSA_LN_PTR	:	
			00	D4	00060	CLRL	EDTSSG_LN_CHGD	:	0746
			04	00	00066	RET		:	0749

: Routine Size: 103 bytes, Routine Base: \_EDTSCODE + 0000

: 178 0748 1  
: 179 0749 1 !<BLF/PAGE>

EDT\$MCGETLIN  
V04-000

EDT\$MCGETLIN - get a text line  
EDT\$\$GET\_TXTLN - get a text line

H 9  
16-Sep-1984 01:10:30  
14-Sep-1984 12:23:55

VAX-11 Bliss-32 V4.0-742  
DISK\$VMMASTER:[EDT.SRC]MCGETLIN.BLI;1 (4) Page 6

: 181 0750 1 END  
: 182 0751 1  
: 183 0752 0 ELUDOM

! of module EDT\$MCGETLIN

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	103	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	33	8	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\$:MCGETLIN/OBJ=OBJ\$:MCGETLIN MSRC\$:MCGETLIN.BLI/UPDATE=(ENH\$:MCGETLIN)

: Size: 103 code + 0 data bytes  
: Run Time: 00:11.0  
: Elapsed Time: 00:13.9  
: Lines/CPU Min: 4116  
: Lexemes/CPU-Min: 11709  
: Memory Used: 69 pages  
: Compilation Complete



This block contains a grid of 137 small terminal window screenshots, each displaying a different LIS (Library Information System) program. The programs are arranged in a grid that is approximately 13 columns wide and 11 rows high. Each window shows a command prompt, headers, and data lists. The programs include:

- LXCOMM0 LIS
- MCGETLIN LIS
- MCRIGHT LIS
- MCCHANGE LIS
- MCCAL LIS
- NOOPEN LIS
- PRAPPNUM LIS
- MSGTXT LIS
- PRGETTOK LIS
- PRISTOK LIS
- LXPRINT LIS
- MCTOP LIS
- MCDOWN LIS
- MCLEFT LIS
- MCBOTTOM LIS
- PAUDIT LIS
- PRGETCHR LIS
- LXCOM LIS
- MAIN LIS
- MCUP LIS