





```

1 0001 0 %TITLE 'EDT$FCRLF - put a crlf in the format buffer'
2 0002 0 MODULE EDT$FCRLF ( ! Put a crlf in the format buffer
3 0003 0 IDENT = 'V04-000' ! File: FCRLF.BLI Edit: JBS1006
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 Put a crlf in the format buffer.
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: March 18, 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$FMT_CRLF from module FORMAT.
46 0046 1 1-002 - Regularize headers. JBS 05-Mar-1981
47 0047 1 1-003 - Use the new formatting flags to control display at the bottom
48 0048 1 of the screen. JBS 05-Jul-1982
49 0049 1 1-004 - Stop the working message before printing the line. JBS 07-Jul-1982
50 0050 1 1-005 - Maintain the cursor position variables. JBS 05-Oct-1982
51 0051 1 1-006 - Be a little moe honest when maintaining the cursor position variables. JBS 29-Oct-1982
52 0052 1 --
53 0053 1

```

EDT\$FCRLF  
V04-000

EDT\$FCRLF - put a crlf in the format buffer  
Declarations

I 10  
16-Sep-1984 00:20:06  
14-Sep-1984 12:23:04

VAX-11 Bliss-32 V4.0-742  
DISK\$VMMASTER:[EDT.SRC]FCRLF.BLI;1 Page 2  
(2)

```

55      0054 1 %SBTTL 'Declarations'
56      0055 1
57      0056 1 : TABLE OF CONTENTS:
58      0057 1 :
59      0058 1
60      0059 1 REQUIRE 'EDTSRC:TRAROUNAM';
61      0498 1
62      0499 1 FORWARD ROUTINE
63      0500 1     EDT$$FMT_CRLF;
64      0501 1
65      0502 1 :
66      0503 1 : INCLUDE FILES:
67      0504 1 :
68      0505 1
69      0506 1 REQUIRE 'EDTSRC:EDTREQ';
70      0641 1
71      0642 1 :
72      0643 1 : MACROS:
73      0644 1 :
74      0645 1 :     NONE
75      0646 1 :
76      0647 1 : EQUATED SYMBOLS:
77      0648 1 :
78      0649 1 :     NONE
79      0650 1 :
80      0651 1 : OWN STORAGE:
81      0652 1 :
82      0653 1 :     NONE
83      0654 1 :
84      0655 1 : EXTERNAL REFERENCES:
85      0656 1 :
86      0657 1 :     In the routine
```

```
88 0658 1 %SBTTL 'EDT$$FMT_CRLF - put a crlf in the format buffer'
89 0659 1
90 0660 1 GLOBAL ROUTINE EDT$$FMT_CRLF          ! Put a crlf in the format buffer
91 0661 1 =
92 0662 1
93 0663 1 ++
94 0664 1 | FUNCTIONAL DESCRIPTION:
95 0665 1 |
96 0666 1 |     Terminate the current output line.  If we are in line mode, we can
97 0667 1 |     just call EDT$$OUT_FMTBUF.  Otherwise there are two cases: normal
98 0668 1 |     and displaying text at the bottom of the screen.  If things are
99 0669 1 |     normal put a CR and LF into the format buffer.  Otherwise (we are in EXT)
100 0670 1 |     just call EDT$$OUT_FMTBUF for the first line, but precede subsequent lines
101 0671 1 |     with CRLF.  If there is more than one line, flag that we need
102 0672 1 |     "press return to continue" and a screen refresh when the operation
103 0673 1 |     is complete.
104 0674 1 |
105 0675 1 | FORMAL PARAMETERS:
106 0676 1 |
107 0677 1 |     NONE
108 0678 1 |
109 0679 1 | IMPLICIT INPUTS:
110 0680 1 |
111 0681 1 |     EDT$$A_FMT_WRRUT
112 0682 1 |     EDT$$G_FMT_BOT
113 0683 1 |     EDT$$G_FMT_LCNT
114 0684 1 |     EDT$$A_FMT_CUR
115 0685 1 |     EDT$$G_SCR_LNS
116 0686 1 |
117 0687 1 | IMPLICIT OUTPUTS:
118 0688 1 |
119 0689 1 |     EDT$$G_FMT_LNPOS
120 0690 1 |     EDT$$G_FMT_LCNT
121 0691 1 |     EDT$$G_MSGFLG
122 0692 1 |     EDT$$G_TXT_ONSCR
123 0693 1 |     EDT$$A_FMT_CUR
124 0694 1 |     EDT$$G_PRV_LN
125 0695 1 |     EDT$$G_PRV_COL
126 0696 1 |
127 0697 1 | ROUTINE VALUE:
128 0698 1 |
129 0699 1 |     Same as EDT$$OUT_FMTBUF, or 1.
130 0700 1 |
131 0701 1 | SIDE EFFECTS:
132 0702 1 |
133 0703 1 |     NONE
134 0704 1 |
135 0705 1 | --
136 0706 1 |
137 0707 2 | BEGIN
138 0708 2 |
139 0709 2 | EXTERNAL ROUTINE
140 0710 2 |     EDT$$OUT_FMTBUF,          ! Output the format buffer to the screen
141 0711 2 |     EDT$$STORE_FMTCH : NOVALUE, ! Store a character into the format buffer
142 0712 2 |     EDT$$TI_WRSTR,          ! String output routine
143 0713 2 |     EDT$$STOP_WKINGMSG : NOVALUE; ! Stop the "working" message
144 0714 2 |
```

```

145 0715 2  EXTERNAL
146 0716 2  EDT$$G_FMT_LNPOS,      ! The current column number
147 0717 2  EDT$$A_FMT_WRRUT,   ! Routine to call to write the buffer
148 0718 2  EDT$$G_FMT_BOT,   ! 1 = we are printing at the bottom of the screen
149 0719 2  EDT$$G_FMT_LCNT, ! Number of lines printed at the bottom of the screen
150 0720 2  EDT$$T_FMT_BUF : VECTOR [EDT$$K_FMT_BUFLN, BYTE], ! Format buffer
151 0721 2  EDT$$A_FMT_CUR,   ! Current location in the format buffer
152 0722 2  EDT$$G_MSGFLG,   ! 1 = erase the message line on the next keystroke
153 0723 2  EDT$$G_TXT_ONSCR, ! Text on screen: use 'Press return to continue'
154 0724 2  EDT$$G_PRV_COL,  ! Cursor column
155 0725 2  EDT$$G_PRV_LN,   ! Cursor line
156 0726 2  EDT$$G_SCR_LNS;  ! Number of lines in the scrolling area
157 0727 2
158 0728 2  LOCAL
159 0729 2  STATUS;
160 0730 2
161 0731 2  EDT$$STOP_WKINGMSG ();
162 0732 2
163 0733 2  IF (.EDT$$A_FMT_WRRUT NEQA EDT$$TI_WRSTR)
164 0734 2  THEN
165 0735 2  BEGIN
166 0736 2  !+
167 0737 2  ! Since we are effectively outputting a CR and LF, the column is set to
168 0738 2  ! zero and the line increases by one, but not below the bottom of the screen.
169 0739 2  -
170 0740 2  EDT$$G_PRV_COL = 0;
171 0741 2
172 0742 2  IF (.EDT$$G_PRV_LN LSS .EDT$$G_SCR_LNS) !
173 0743 2  THEN
174 0744 2  EDT$$G_PRV_LN = MIN (.EDT$$G_SCR_LNS - 1, .EDT$$G_PRV_LN + 1)
175 0745 2  ELSE
176 0746 2  EDT$$G_PRV_LN = .EDT$$G_PRV_LN + 1;
177 0747 2
178 0748 2  STATUS = EDT$$OUT_FMTBUF ();
179 0749 2  END
180 0750 2  ELSE
181 0751 2  BEGIN
182 0752 2
183 0753 2  IF .EDT$$G_FMT_BOT
184 0754 2  THEN
185 0755 2  BEGIN
186 0756 2
187 0757 2  IF (.EDT$$G_FMT_LCNT EQL 0)
188 0758 2  THEN
189 0759 2  BEGIN
190 0760 2  !+
191 0761 2  ! This is the first line printed in the message area at the bottom
192 0762 2  ! of the screen.
193 0763 2  -
194 0764 2  STATUS = EDT$$OUT_FMTBUF ();
195 0765 2  EDT$$G_FMT_LCNT = 1;
196 0766 2  EDT$$G_MSGFLG = 1;
197 0767 2  END
198 0768 2  ELSE
199 0769 2  BEGIN
200 0770 2  !+
201 0771 2  ! This is the second line to go into the message area. Preceed the text with a CRLF,

```

```

202 0772 5 | and adjust the column and line counters, assuming that there are no cursor
203 0773 5 | positioning sequences or control characters in the format buffer.
204 0774 5 |
205 0775 5 |
206 0776 6 IF (.EDT$$G_PRV_LN LSS .EDT$$G_SCR_LNS) !
207 0777 5 THEN
208 0778 5 EDT$$G_PRV_LN = MIN (.EDT$$G_SCR_LNS - 1, .EDT$$G_PRV_LN + 1)
209 0779 5 ELSE
210 0780 5 EDT$$G_PRV_LN = .EDT$$G_PRV_LN + 1;
211 0781 5
212 0782 5 EDT$$G_PRV_COL = .EDT$$A_FMT_CUR - EDT$$T_FMT_BUF;
213 0783 5 EDT$$COPY_MEM (EDT$$K_FMT_BUF[EN - 2, EDT$$T_FMT_BUF, EDT$$T_FMT_BUF + 2);
214 0784 5 EDT$$T_FMT_BUF [0] = ASC_K_CR;
215 0785 5 EDT$$T_FMT_BUF [1] = ASC_K_LF;
216 0786 5 EDT$$A_FMT_CUR = .EDT$$A_FMT_CUR + 2;
217 0787 5 STATUS = EDT$$OUT_FMTBUF ();
218 0788 5 EDT$$G_FMT_LCNT = .EDT$$G_FMT_LCNT + 1;
219 0789 5 EDT$$G_TXT_ONSCR = 1;
220 0790 4 END;
221 0791 4
222 0792 4 END
223 0793 3 ELSE
224 0794 4 BEGIN
225 0795 4 | +
226 0796 4 | This line is not being placed at the bottom of the buffer.
227 0797 4 | -
228 0798 4 EDT$$G_PRV_COL = 0;
229 0799 4
230 0800 5 IF (.EDT$$G_PRV_LN LSS .EDT$$G_SCR_LNS) !
231 0801 4 THEN
232 0802 4 EDT$$G_PRV_LN = MIN (.EDT$$G_SCR_LNS - 1, .EDT$$G_PRV_LN + 1)
233 0803 4 ELSE
234 0804 4 EDT$$G_PRV_LN = .EDT$$G_PRV_LN + 1;
235 0805 4
236 0806 4 EDT$$STORE_FMTCH (ASC_K_CR);
237 0807 4 EDT$$STORE_FMTCH (ASC_K_LF);
238 0808 4 STATUS = 1;
239 0809 3 END;
240 0810 3
241 0811 3 EDT$$G_FMT_LNPOS = 0;
242 0812 2 END;
243 0813 2
244 0814 2 RETURN (.STATUS);
245 0815 1 END;

```

! of routine EDT\$\$\$FMT\_CRLF

```

.TITLE EDT$FCRLF EDT$FCRLF - put a crlf in the format
      buffer
.IDENT \V04-000\

.EXTRN EDT$$OUT_FMTBUF
.EXTRN EDT$$STORE_FMTCH
.EXTRN EDT$$TI_WSTR, EDT$$STOP_WKINGMSG
.EXTRN EDT$$G_FMT_LNPOS
.EXTRN EDT$$A_FMT_WRRUT
.EXTRN EDT$$G_FMT_BOT, EDT$$G_FMT_LCNT
.EXTRN EDT$$T_FMT_BUF, EDT$$A_FMT_CUR

```

					.EXTRN	EDT\$\$G_MSGFLG, EDT\$\$G_TXT_ONSCR	
					.EXTRN	EDT\$\$G_PRV_COL, EDT\$\$G_PRV_LN	
					.EXTRN	EDT\$\$G_SCR_LNS	
					.PSECT	_EDT\$CODE, NOWRT, SHR, PIC, 2	
					.ENTRY	EDT\$\$FMT_CRLF, Save R2,R3,R4,R5,R6,R7,R8,-	0660
						R9,R10,R11	
					MOVAB	EDT\$\$OUT_FMTBUF, R11	
					MOVAB	EDT\$\$G_SCR_LNS, R10	
					MOVAB	EDT\$\$G_PRV_COL, R9	
					MOVAB	EDT\$\$T_FMT_BUF, R8	
					MOVAB	EDT\$\$G_PRV_LN, R7	
	00000000G	00	00	FB	CALLS	#0, EDT\$\$STOP_WKINGMSG	0731
		50	00	9E	MOVAB	EDT\$\$TI_WRSTR, R0	0733
		50	00	D1	CMPL	EDT\$\$A_FMT_WRRUT, R0	
			29	13	BEQL	4\$	
			69	D4	CLRL	EDT\$\$G_PRV_COL	0740
		50	67	D0	MOVL	EDT\$\$G_PRV_LN, R0	0742
		51	6A	D0	MOVL	EDT\$\$G_SCR_LNS, R1	
		51	50	D1	CMPL	R0, R1	
			11	18	BGEQ	2\$	
			51	D7	DECL	R1	0744
			50	D6	INCL	R0	
		50	51	D1	CMPL	R1, R0	
			03	15	BLEQ	1\$	
		51	50	D0	MOVL	R0, R1	
		67	51	D0	MOVL	R1, EDT\$\$G_PRV_LN	
			02	11	BRB	3\$	
			67	D6	INCL	EDT\$\$G_PRV_LN	0746
		68	00	FB	CALLS	#0, EDT\$\$OUT_FMTBUF	0748
		56	50	D0	MOVL	R0, STATUS	
			00B1	31	BRW	15\$	0733
		6F	00	E9	BLBC	EDT\$\$G_FMT_BOT, 10\$	0753
			00	D5	TSTL	EDT\$\$G_FMT_LCNT	0757
			16	12	BNEQ	5\$	
		68	00	FB	CALLS	#0, EDT\$\$OUT_FMTBUF	0764
		56	50	D0	MOVL	R0, STATUS	
	00000000G	00	01	D0	MOVL	#1, EDT\$\$G_FMT_LCNT	0765
	00000000G	00	01	D0	MOVL	#1, EDT\$\$G_MSGFLG	0766
			4F	11	BRB	9\$	0757
		50	67	D0	MOVL	EDT\$\$G_PRV_LN, R0	0776
		51	6A	D0	MOVL	EDT\$\$G_SCR_LNS, R1	
		51	50	D1	CMPL	R0, R1	
			11	18	BGEQ	7\$	
			51	D7	DECL	R1	0778
			50	D6	INCL	R0	
		50	51	D1	CMPL	R1, R0	
			03	15	BLEQ	6\$	
		51	50	D0	MOVL	R0, R1	
		67	51	D0	MOVL	R1, EDT\$\$G_PRV_LN	
			02	11	BRB	8\$	
			67	D6	INCL	EDT\$\$G_PRV_LN	0780
		50	68	9E	MOVAB	EDT\$\$T_FMT_BUF, R0	0782
	69 00000000G	00	50	C3	SUBL3	R0, EDT\$\$A_FMT_CUR, EDT\$\$G_PRV_COL	
02	A8	68	01FE	8F	MOVC3	#510, EDT\$\$T_FMT_BUF, EDT\$\$T_FMT_BUF+2	0783
		68	0A0D	8F	MOVW	#2573, EDT\$\$T_FMT_BUF	0784



EDT\$FCRLF  
V04-000

EDT\$FCRLF - put a crlf in the format buffer  
EDT\$\$FMT\_CRLF - put a crlf in the format buffe

N 10  
16-Sep-1984 00:20:06  
14-Sep-1984 12:23:04

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

Page 7  
(3)

ED  
VO

00000000G	00	02	C0	000BF	ADDL2	#2, EDT\$\$A_FMT_CUR	:	0786
	6B	00	FB	000C6	CALLS	#0, FDT\$\$OOT_FMTBUF	:	0787
	56	50	D0	000C9	MOVL	R0, STATUS	:	
00000000G	00	00	D6	000CC	INCL	EDT\$\$G_FMT_LCNT	:	0788
		01	D0	000D2	MOVL	#1, EDT\$\$G_TXT_ONSCR	:	0789
		35	11	000D9	BRB	14\$	:	0753
		69	D4	000DB	CLRL	EDT\$\$G_PRIV_COL	:	0798
	50	67	D0	000DD	MOVL	EDT\$\$G_PRIV_LN, R0	:	0800
	51	6A	D0	000E0	MOVL	EDT\$\$G_SCR_LNS, R1	:	
	51	50	D1	000E3	CMPL	R0, R1	:	
		11	18	000E6	BGEQ	12\$	:	
		51	D7	000E8	DECL	R1	:	0802
		50	D6	000EA	INCL	R0	:	
	50	51	D1	000EC	CMPL	R1, R0	:	
		03	15	000EF	BLEQ	11\$	:	
	51	50	D0	000F1	MOVL	R0, R1	:	
	67	51	D0	000F4	MOVL	R1, EDT\$\$G_PRIV_LN	:	
		02	11	000F7	BRB	13\$	:	
		67	D6	000F9	INCL	EDT\$\$G_PRIV_LN	:	0804
		0D	DD	000FB	PUSHL	#13	:	0806
00000000G	00	01	FB	000FD	CALLS	#1, EDT\$\$STORE_FMTCH	:	
		0A	DD	00104	PUSHL	#10	:	0807
00000000G	00	01	FB	00106	CALLS	#1, EDT\$\$STORE_FMTCH	:	
	56	01	D0	0010D	MOVL	#1, STATUS	:	0808
		00	D4	00110	CLRL	EDT\$\$G_FMT_LNPOS	:	0811
	50	56	D0	00116	MOVL	STATUS, R0	:	0814
		04	00	00119	RET		:	0815

: Routine Size: 282 bytes, Routine Base: \_EDT\$CODE + 0000

: 246 0816 1  
: 247 0817 1 !<BLF/PAGE>

EDT\$FCRLF  
V04-000

EDT\$FCRLF - put a crlf in the format buffer  
EDT\$\$FMT\_CRLF - put a crlf in the format buffe

B 11  
16-Sep-1984 00:20:06  
14-Sep-1984 12:23:04

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

Page 8  
(4)

: 249 0818 1 END  
: 250 0819 1  
: 251 0820 0 ELUDOM

! of module EDT\$FCRLF

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	282	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	3	0	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\$:FCRLF/OBJ=OBJ\$:FCRLF MSRCS\$:FCRLF.BLI/UPDATE=(ENHS:FCRLF)

: Size: 282 code + 0 data bytes  
: Run Time: 00:15.5  
: Elapsed Time: 00:18.4  
: Lines/CPU Min: 3184  
: Lexemes/CPU-Min: 8920  
: Memory Used: 94 pages  
: Compilation Complete

ED  
VO

EXTEND  
LIS

FDEC  
LIS

FILL  
LIS

FINDPARA  
LIS

FCRLF  
LIS

EDT  
LIS

EXEC  
LIS

EXECNOO  
LIS

FILEIO  
LIS

EDTVECTOR  
LIS

FINDKEY  
LIS

FCOLINC  
LIS

FINAL  
LIS

FINDHDR  
LIS

DEFKEY  
LIS

ERRMSG  
LIS

FCHAR  
LIS