


```

TTTTTTTTT1  IIIIIII  CCCCCCCCC  LL  RRRRRRRR  AAAAAA  UU  UU  DDDDDDDD
TTTTTTTTTT  IIIIIII  CCCCCCCCC  LL  RRRRRRRR  AAAAAA  UU  UU  DDDDDDDD
TT          II      CC          LL  RR      RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR      RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR      RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RRRRRRRR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RRRRRRRR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR  RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR  RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR  RR  AA  AA  UU  UU  DD  DD
TT          II      CC          LL  RR  RR  AA  AA  UU  UU  DD  DD
TT          IIIIIII  CCCCCCCCC  LL  RR  RR  AA  AA  UU  UU  DD  DD
TT          IIIIIII  CCCCCCCCC  LL  RR  RR  AA  AA  UUUUUUUUU  DDDDDDDD
TT          IIIIIII  CCCCCCCCC  LL  RR  RR  AA  AA  UUUUUUUUU  DDDDDDDD

```

```

LL          IIIIIII  SSSSSSSS
LL          IIIIIII  SSSSSSSS
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          IIIIIII  SSSSSSSS
LL          IIIIIII  SSSSSSSS

```


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57

```

0001 0 %TITLE 'EDT$TICLRAUD - flush journal file'
0002 0 MODULE EDT$TICLRAUD ( ! Flush journal file
0003 0 IDENT = 'V04-000' ! File: TICLRAUD.BLI Edit: JBS1012
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 Flush journal file.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: Bob Kushlis, CREATION DATE: June 9, 1979
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. DJS 18-FEB-1981. This module was created by
0045 1 extracting routine EDT$TI_FLUSHJOUFI from module TINPUT.
0046 1 1-002 - Regularize headers. JBS 11-Mar-1981
0047 1 1-003 - Add parameter to routine and flag to record. JBS 18-Jun-1981
0048 1 1-004 - Make record flag compatible with EDT V2. JBS 07-Jul-1981
0049 1 1-005 - Fix a bug in control C processing. JBS 17-Dec-1981
0050 1 1-006 - Continue to debug control C processing. JBS 24-Dec-1981
0051 1 1-007 - Revise control C data names. JBS 29-Dec-1981
0052 1 1-008 - Use two words for control C counters. JBS 30-Dec-1981
0053 1 1-009 - Decrease stack usage. JBS 27-Jan-1982
0054 1 1-010 - We must write 0-length journal records. JBS 01-Apr-1982
0055 1 1-011 - We must not write a record unless it has been marked valid. JBS 09-Apr-1982
0056 1 1-012 - Use symbols instead of magic numbers in control C journaling. JBS 24-May-1982
0057 1 --

```

EDT\$TICLRAUD
V04-000

EDT\$TICLRAUD - flush journal file

K 8
16-Sep-1984 01:54:40
14-Sep-1984 12:24:49

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

: 58

0058 1

```
.. 60 0059 1 %SBTTL 'Declarations'  
.. 61 0060 1 |  
.. 62 0061 1 | TABLE OF CONTENTS:  
.. 63 0062 1 |  
.. 64 0063 1 |  
.. 65 0064 1 REQUIRE 'EDTSRC:TRAROUNAM';  
.. 66 0503 1 |  
.. 67 0504 1 FORWARD ROUTINE  
.. 68 0505 1 EDT$STI_FLUSHJOUFI : NOVALUE;  
.. 69 0506 1 |  
.. 70 0507 1 |  
.. 71 0508 1 | INCLUDE FILES:  
.. 72 0509 1 |  
.. 73 0510 1 |  
.. 74 0511 1 REQUIRE 'EDTSRC:EDTREQ';  
.. 75 0646 1 |  
.. 76 0647 1 |  
.. 77 0648 1 | MACROS:  
.. 78 0649 1 |  
.. 79 0650 1 | NONE  
.. 80 0651 1 |  
.. 81 0652 1 | EQUATED SYMBOLS:  
.. 82 0653 1 |  
.. 83 0654 1 | NONE  
.. 84 0655 1 |  
.. 85 0656 1 | OWN STORAGE:  
.. 86 0657 1 |  
.. 87 0658 1 | NONE  
.. 88 0659 1 |  
.. 89 0660 1 | EXTERNAL REFERENCES:  
.. 90 0661 1 |  
.. 91 0662 1 | In the routine
```



```

: 93      0663 1 %SBTTL 'EDT$STI_FLUSHJOUFI - flush journal file'
: 94      0664 1
: 95      0665 1 GLOBAL ROUTINE EDT$STI_FLUSHJOUFI (           ! Flush journal file
: 96      0666 1     RECORD_TYPE                               ! text or control C
: 97      0667 1     ) : NOVALUE =
: 98      0668 1
: 99      0669 1 ++
100     0670 1 | FUNCTIONAL DESCRIPTION:
101     0671 1 |
102     0672 1 |     This routine writes either a text record or a control C record.
103     0673 1 |     A text record is taken from the journal buffer. A control C record
104     0674 1 |     consists of the count of the number of times we tested for control C
105     0675 1 |     and didn't find it. Upon writing either record the control c counter
106     0676 1 |     is cleared; thus it records the number of tests since the last journal
107     0677 1 |     record.
108     0678 1 |
109     0679 1 | FORMAL PARAMETERS:
110     0680 1 |
111     0681 1 |     RECORD_TYPE     ASCII 'T' for a text record, ASCII 'C' for a control C record.
112     0682 1 |
113     0683 1 | IMPLICIT INPUTS:
114     0684 1 |
115     0685 1 |     EDT$ST_TIN_OBUF
116     0686 1 |     EDT$SG_TIN_OBUFPOS
117     0687 1 |     EDT$SG_CC_CNT1_LO
118     0688 1 |     EDT$SC_CC_CNT1_HI
119     0689 1 |     EDT$SG_JOU_VALID
120     0690 1 |
121     0691 1 | IMPLICIT OUTPUTS:
122     0692 1 |
123     0693 1 |     EDT$SG_TIN_OBUFPOS
124     0694 1 |     EDT$SG_JOU_VALID
125     0695 1 |
126     0696 1 | ROUTINE VALUE:
127     0697 1 |
128     0698 1 |     NONE
129     0699 1 |
130     0700 1 | SIDE EFFECTS:
131     0701 1 |
132     0702 1 |     NONE
133     0703 1 |
134     0704 1 | --
135     0705 1 |
136     0706 2 | BEGIN
137     0707 2 |
138     0708 2 | EXTERNAL ROUTINE
139     0709 2 |     EDT$JOU_PUTREC;
140     0710 2 |
141     0711 2 | EXTERNAL
142     0712 2 |     EDT$ST_TIN_OBUF : VECTOR [256, BYTE], ! The journal output buffer
143     0713 2 |     EDT$SG_TIN_OBUFPOS, ! Position in journal output buffer
144     0714 2 |     EDT$SG_CC_CNT1_LO, ! Number of control C tests
145     0715 2 |     EDT$SG_CC_CNT1_HI, ! High half of the above
146     0716 2 |     EDT$SG_JOU_VALID; ! 1 = journal record is valid
147     0717 2 |
148     0718 2 | LOCAL
149     0719 2 |     JOURNAL_RECORD : VECTOR [CC_REC_SIZE, BYTE]; ! For building the control C record
```

```

150 0720 2
151 0721 2
152 0722 2
153 0723 2
154 0724 2
155 0725 2
156 0726 2
157 0727 2
158 0728 2
159 0729 2
160 0730 2
161 0731 2
162 0732 2
163 0733 2
164 0734 2
165 0735 2
166 0736 2
167 0737 2
168 0738 2
169 0739 2
170 0740 2
171 0741 2
172 0742 2
173 0743 2
174 0744 2
175 0745 2
176 0746 2
177 0747 2
178 0748 2
179 0749 2
180 0750 2
181 0751 2
182 0752 2
183 0753 2
184 0754 2
185 0755 2
186 0756 2
187 0757 2
188 0758 2
189 0759 2
190 0760 1

```

```

!+
!-
Make sure the control C counter is reasonable.
ASSERT (.EDT$$G_CC_CNT1_LO LEQ CC_CTR_MAX);
ASSERT (.EDT$$G_CC_CNT1_HI LEQ CC_CTR_MAX);

SELECTONE .RECORD_TYPE OF
SET
  [%C'T'] : ! Output a text record
  BEGIN
  ASSERT (.EDT$$G_TIN_OBUFPOS LEQ 256);
  ASSERT (.EDT$$G_TIN_OBUFPOS GEQ 0);

  IF (.EDT$$G_TIN_OBUFPOS GTR 0) THEN ASSERT (.EDT$$G_JOU_VALID);

  IF .EDT$$G_JOU_VALID
  THEN
  BEGIN
  EDT$$JOU_PUTREC (EDT$$T_TIN_OBUF [0], .EDT$$G_TIN_OBUFPOS);
  EDT$$G_TIN_OBUFPOS = 0;
  EDT$$G_JOU_VALID = 0;
  END;

  END;

  [%C'C'] : ! Output a control C record
  BEGIN
  JOURNAL_RECORD [0] = JOU_REC_ESC; ! Flag as non-text record
  JOURNAL_RECORD [1] = CC_REC_FLAG; ! Control C record
  EDT$$CPY_MEM (2, EDT$$G_CC_CNT1_LO, JOURNAL_RECORD [2]);
  EDT$$CPY_MEM (2, EDT$$G_CC_CNT1_HI, JOURNAL_RECORD [4]);
  EDT$$JOU_PUTREC (JOURNAL_RECORD [0], CC_REC_SIZE);
  END;

  [OTHERWISE] :
  ASSERT (0);
  TES;

END; ! of routine EDT$$TI_FLUSHJOUFI

```

```

.TITLE EDT$TICLRAUD EDT$TICLRAUD - flush journal file
.IDENT \V04-000\

.EXTRN EDT$$JOU_PUTREC
.EXTRN EDT$$T_TIN_OBUF
.EXTRN EDT$$G_TIN_OBUFPOS
.EXTRN EDT$$G_CC_CNT1_LO
.EXTRN EDT$$G_CC_CNT1_HI
.EXTRN EDT$$G_JOU_VALID
.EXTRN EDT$$INTER_ERR

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

.ENTRY EDT$$TI_FLUSHJOUFI, Save R2,R3,R4,R5,R6,R7 ; 0665

```

00FC 0000

	57	00000000G	00	9E	00002	MOVAB	EDTSSG_CC_CNT1_LO, R7	
	56	00000000G	00	9E	00009	MOVAB	EDTSSJOU_PUTREC, R6	
	55	00000000G	00	9E	00010	MOVAB	EDTSSG_CC_CNT1_HI, R5	
	54	00000000G	00	9E	00017	MOVAB	EDTSSG_JOU_VALID, R4	
	53	00000000G	00	9E	0001E	MOVAB	EDTSSG_TIN_OBUFPOS, R3	
	52	00000000G	00	9E	00025	MOVAB	EDTSSINTER_ERR, R2	
	5E		08	C2	0002C	SUBL2	#8, SP	
00007530	8F		67	D1	0002F	CMPL	EDTSSG_CC_CNT1_LO, #30000	0724
			03	15	00036	BLEQ	1\$	
00007530	62		00	FB	00038	CALLS	#0, EDTSSINTER_ERR	
	8F		65	D1	0003B	CMPL	EDTSSG_CC_CNT1_HI, #30000	0725
			03	15	00042	BLEQ	2\$	
	62		00	FB	00044	CALLS	#0, EDTSSINTER_ERR	
00000054	8F	04	AC	D0	00047	MOVL	RECORD_TYPE, R0	0727
			50	D1	0004B	CMPL	R0, #8\$	0730
			30	12	00052	BNEQ	7\$	
00000100	8F		63	D1	00054	CMPL	EDTSSG_TIN_OBUFPOS, #256	0732
			03	15	00056	BLEQ	3\$	
	62		00	FB	0005D	CALLS	#0, EDTSSINTER_ERR	
			63	D5	00060	TSTL	EDTSSG_TIN_OBUFPOS	0733
			03	1F	00062	BGEQ	4\$	
	62		00	FE	00064	CALLS	#0, EDTSSINTER_ERR	
			63	D5	00067	TSTL	EDTSSG_TIN_OBUFPOS	0735
			06	15	00069	BLEQ	5\$	
	06		64	E8	0006B	BLBS	EDTSSG_JOU_VALID, 6\$	
	62		00	FB	0006E	CALLS	#0, EDTSSINTER_ERR	
	32		64	E9	00071	BLBC	EDTSSG_JOU_VALID, 9\$	0737
			63	DD	00074	PUSHL	EDTSSG_TIN_OBUFPOS	0740
		00000000G	00	9F	00076	PUSHAB	EDTSSINTER_ERR	
	66		02	FB	0007C	CALLS	#2, EDTSSJOU_PUTREC	
			63	D4	0007F	CLRL	EDTSSG_TIN_OBUFPOS	0741
			64	D4	00081	CLRL	EDTSSG_JOU_VALID	0742
			04	00083	RET			0727
00000043	8F		50	D1	00084	CMPL	R0, #67	0747
			16	12	0008B	BNEQ	8\$	
	6E	01FF	8F	B0	0008D	MOVW	#511, JOURNAL_RECORD	0749
	02	AE	67	B0	00092	MOVW	EDTSSG_CC_CNT1_LO, JOURNAL_RECORD+2	0751
	04	AE	65	B0	00096	MOVW	EDTSSG_CC_CNT1_HI, JOURNAL_RECORD+4	0752
			06	DD	0009A	PUSHL	#6	0753
		04	AE	9F	0009C	PUSHAB	JOURNAL_RECORD	
	66		02	FB	0009F	CALLS	#2, EDTSSJOU_PUTREC	
			04	000A2	RET			0727
	62		00	FB	000A3	CALLS	#0, EDTSSINTER_ERR	0757
			04	000A6	9\$:	RET		0760

; Routine Size: 167 bytes, Routine Base: _EDT\$CODE + 0000

; 191 0761 1
; 192 0762 1 !<BLF/PAGE>

EDT\$TICLRAUD
V04-000

EDT\$TICLRAUD - flush journal file
EDT\$STI_FLUSHJOUFI - flush journal file

C 9
16-Sep-1984 01:54:40
14-Sep-1984 12:24:49

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

Page 7

: 194 0763 1 END
: 195 0764 1
: 196 0765 0 ELUDOM

! of module EDT\$TICLRAUD

PSECT SUMMARY

Name Bytes Attributes
:_EDT\$CODE 167 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	5	1	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\$:TICLRAUD/OBJ=OBJ\$:TICLRAUD MSRC\$:TICLRAUD.BLI/UPDATE=(ENH\$:TICLRAUD)

: Size: 167 code + 0 data bytes
: Run Time: 00:13.3
: Elapsed Time: 00:16.2
: Lines/CPU Min: 3458
: Lexemes/CPU-Min: 11751
: Memory Used: 86 pages
: Compilation Complete

