

```

EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD TTT

```

**EXE**

Mod

ED 1

ED

ED

ED

ED

ED1

ED

ED

ED  
EDED  
EDSYS  
LIB

L18

```
WW      WW  FFFFFFFF  SSSSSSSS  PPPPPPPP  LL      88888888  KK      KK  TTTTTTTTTT
WW      WW  FFFFFFFF  SSSSSSSS  PPPPPPPP  LL      88888888  KK      KK  TTTTTTTTTT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SSSSSS  PPPPPPPP  LL      88888888  KK      KK  TT
WW      WW  FFFFFFFF  SSSSSS  PPPPPPPP  LL      88888888  KKKKKK  TT
WW      WW  FFFFFFFF  SSSSSS  PPPPPPPP  LL      88888888  KKKKKK  TT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SS      PP      PP  LL      88      88  KK      KK  TT
WW      WW  FF      SSSSSSSS  PP      LL      88888888  KK      KK  TT
WW      WW  FF      SSSSSSSS  PP      LL      88888888  KK      KK  TT
LL      LL      LLLLLLLLLL  LLLLLLLLLL  88888888  KK      KK  TT
LL      LL      LLLLLLLLLL  LLLLLLLLLL  88888888  KK      KK  TT
```

```
LL      I111111  SSSSSSSS
LL      I111111  SSSSSSSS
LL      I1      SS
LL      I1      SS
LL      I1      SS
LL      I1      SS
LL      I1      SSSSSS
LL      I1      SSSSSS
LL      I1      SS
LL      I1      SS
LL      I1      SS
LL      I1      SS
LL      I1      SSSSSSSS
LLLLLLLLLLLL  I111111  SSSSSSSS
LLLLLLLLLLLL  I111111  SSSSSSSS
```

```
....
....
....
....
```

.....

:

.....

```
0001 0 %TITLE 'EDT$WFSPLBKT - split the current bucket'
0002 0 MODULE EDT$WFSPLBKT ( ! Split the current bucket
0003 0 IDENT = 'V04-000' ! File: WFSPLBKT.BLI Edit: JBS1010
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 Split the current bucket.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: Bob Kushlis, CREATION DATE: October 16, 1978
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. DJS 23-Feb-1981. This module was created by
0045 1 extracting routine SPLIT BUKT from module EDTWF.
0046 1 1-002 - Regularize headers. JBS 19-Mar-1981
0047 1 1-003 - Modify to use EDT$WORKIO. STS 15-Feb-1982
0048 1 1-004 - Copy entire old bucket before split. STS 18-Feb-1982
0049 1 1-005 - Don't copy on 11's since it uses too much stack. STS 01-Mar-1982
0050 1 1-006 - Change stack storage to heap storage. STS 05-Mar-1982
0051 1 1-007 - Add literals for callable EDT. STS 10-Mar-1982
0052 1 1-008 - Give an error return if heap storage is exhausted. JBS 09-Jun-1982
0053 1 1-009 - Put code for edt$$wf_nxt_buf in line. STS 11-oct-1982
0054 1 1-010 - Improve listing appearance. JBS 20-Jun-1983
0055 1
0056 1 --
```

EDT\$WFSPLBKT  
V04-000

EDT\$WFSPLBKT - split the current bucket  
Declarations

K 15  
16-Sep-1984 02:15:01  
14-Sep-1984 12:25:45

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

```

: 58 0057 1 %SBTTL 'Declarations'
: 59 0058 1
: 60 0059 1 : TABLE OF CONTENTS:
: 61 0060 1 :
: 62 0061 1
: 63 0062 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 64 0501 1
: 65 0502 1 FORWARD ROUTINE
: 66 0503 1 EDT$WF_SPLTBUK;
: 67 0504 1
: 68 0505 1
: 69 0506 1 : INCLUDE FILES:
: 70 0507 1 :
: 71 0508 1
: 72 0509 1 REQUIRE 'EDTSRC:EDTREQ';
: 73 0644 1
: 74 0645 1 :
: 75 0646 1 : MACROS:
: 76 0647 1
: 77 0648 1 : NONE
: 78 0649 1
: 79 0650 1 : EQUATED SYMBOLS:
: 80 0651 1
: 81 0652 1
: 82 0653 1 EXTERNAL LITERAL
: 83 0654 1 EDT$K_PUT;
: 84 0655 1
: 85 0656 1
: 86 0657 1 : OWN STORAGE:
: 87 0658 1
: 88 0659 1 : NONE
: 89 0660 1
: 90 0661 1 : EXTERNAL REFERENCES:
: 91 0662 1
: 92 0663 1 : In the routine
```

```

: 94      0664 1 %SBTTL 'EDT$$WF_SPLTBUK - split the current bucket'
: 95      0665 1
: 96      0666 1 GLOBAL ROUTINE EDT$$WF_SPLTBUK                ! Split the current bucket
: 97      0667 1 =
: 98      0668 1
: 99      0669 1 !++
100      0670 1 ! FUNCTIONAL DESCRIPTION:
101      0671 1
102      0672 1      This routine splits the current bucket at the current position into
103      0673 1      two buckets.  In the special case that we are at the end of the bucket
104      0674 1      this is done simply by appending a new bucket, otherwise, we must allocate
105      0675 1      a new bucket and copy all the text from the current line to the end of the
106      0676 1      bucket into the new bucket.
107      0677 1
108      0678 1 ! FORMAL PARAMETERS:
109      0679 1
110      0680 1      NONE
111      0681 1
112      0682 1 ! IMPLICIT INPUTS:
113      0683 1
114      0684 1      EDT$$A_CUR_BUF
115      0685 1      EDT$$Z_WF_DESC
116      0686 1      EDT$$A_WK_BUK
117      0687 1      EDT$$G_WK_CURBUK
118      0688 1
119      0689 1 ! IMPLICIT OUTPUTS:
120      0690 1
121      0691 1      EDT$$G_WK_MODFD
122      0692 1      EDT$$A_WK_BUK
123      0693 1
124      0694 1 ! ROUTINE VALUE:
125      0695 1
126      0696 1      1 = OK
127      0697 1      0 = heap storage exhausted
128      0698 1
129      0699 1 ! SIDE EFFECTS:
130      0700 1
131      0701 1      NONE
132      0702 1
133      0703 1 ! --
134      0704 1
135      0705 2 BEGIN
136      0706 2
137      0707 2 EXTERNAL ROUTINE
138      0708 2      EDT$$ALO_HEAP,                ! allocate heap storage
139      0709 2      EDT$$DEA_HEAP : NOVALUE,        ! deallocate heap storage
140      0710 2      EDT$$CALC_WIO,
141      0711 2      EDT$$WF_NEWBUK : NOVALUE,
142      0712 2      EDT$$WF_ALOBUF : NOVALUE,
143      0713 2      EDT$$WF_MAKECUR : NOVALUE;
144      0714 2
145      0715 2 EXTERNAL
146      0716 2      EDT$$Z_WF_DESC : BLOCK [, BYTE],    ! descriptor for the workfile record
147      0717 2      EDT$$G_WK_AVAIL,
148      0718 2      EDT$$G_WK_GRTSTBUK,
149      0719 2      EDT$$A_CUR_BUF : REF TBCB_BLOCK,    ! Current text buffer control block
150      0720 2      EDT$$A_WK_BUK :                  ! Pointer to current bucket
```

```
151 0721 2 REF BLOCK [WF_BUKT_SIZE, BYTE] FIELD (WFB_FIELDS),
152 0722 2 EDT$$G_WK_CURBUK, ! Number of the current bucket
153 0723 2 EDT$$G_WK_MODFD; ! Flag indicating bucket was modified
154 0724 2
155 0725 2 LOCAL
156 0726 2 OLD_NEXT,
157 0727 2 LEN,
158 0728 2 ORIG_BUKT;
159 0729 2
160 0730 2 +
161 0731 2 Remember the next bucket number, and the original one.
162 0732 2 -
163 0733 2 OLD_NEXT = .EDT$$A_WK_BUK [WFB_NEXT_BUKT];
164 0734 2 ORIG_BUKT = .EDT$$G_WK_CURBUK;
165 0735 2 +
166 0736 2 First check to see if we are at the end of a bucket.
167 0737 2 -
168 0738 2
169 0739 2 IF (.EDT$$A_CUR_BUF [TBCB_LINE_ADDR] EQL .EDT$$A_WK_BUK [WFB_END])
170 0740 2 THEN
171 0741 2 +
172 0742 2 We are at the end, just link a new bucket to this one
173 0743 2 -
174 0744 2 BEGIN
175 0745 2
176 0746 2 IF (.EDT$$G_WK_AVAIL NEQ 0)
177 0747 2 THEN
178 0748 2 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = .EDT$$G_WK_AVAIL
179 0749 2 ELSE
180 0750 2 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = .EDT$$G_WK_GRTSTBUK;
181 0751 2
182 0752 2 EDT$$G_WK_MODFD = 1;
183 0753 2 EDT$$WF_NEWBUK (.OLD_NEXT, .ORIG_BUKT)
184 0754 2 END
185 0755 2 ELSE
186 0756 2 BEGIN
187 0757 2 +
188 0758 2 We are not at the end of a bucket.
189 0759 2 Split the bucket into two, at the beginning of the current line.
190 0760 2 -
191 0761 2
192 0762 2 LOCAL
193 0763 2 OLD_BUKT,
194 0764 2 NEW_BUKT;
195 0765 2
196 0766 2 LEN = .EDT$$A_WK_BUK [WFB_END] - .EDT$$A_CUR_BUF [TBCB_LINE_ADDR];
197 0767 2 EDT$$A_WK_BUK [WFB_END] = .EDT$$A_CUR_BUF [TBCB_LINE_ADDR];
198 0768 2
199 0769 2 IF (.EDT$$G_WK_AVAIL NEQ 0)
200 0770 2 THEN
201 0771 2 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = NEW_BUKT = .EDT$$G_WK_AVAIL
202 0772 2 ELSE
203 0773 2 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = NEW_BUKT = .EDT$$G_WK_GRTSTBUK;
204 0774 2
205 0775 2 EDT$$CALLWIO (EDT$K_PUT, .EDT$$G_WK_CURBUK, EDT$$Z_WF_DESC);
206 0776 2 +
207 0777 2 Save the bucket contents so that later we can extract a portion.
```

EDT\$WFSPLBKT  
V04-000

EDT\$WFSPLBKT - split the current bucket  
EDT\$\$WF\_SPLTBUK - split the current bucket

N 15  
16-Sep-1984 02:15:01  
14-Sep-1984 12:25:45

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

```

208 0778 3 !-
209 0779
210 0780 IF ( NOT EDT$$ALO_HEAP (%REF (WF_BUKT_SIZE), OLD_BUKT)) THEN RETURN (0);
211 0781
212 0782 EDT$$CPY_MEM (WF_BUKT_SIZE, .EDT$$A_WK_BUK, .OLD_BUKT);
213 0783
214 0784 + Now get a fresh buffer.
215 0785
216 0786 EDT$$WF_ALOBUF ();
217 0787 EDT$$A_WK_BUK [WFB_NEXT_BUKT] = .OLD_NEXT;
218 0788 EDT$$A_WK_BUK [WFB_PREV_BUKT] = .ORIG_BUKT;
219 0789
220 0790 + Copy a portion of the old buffer into the new buffer.
221 0791
222 P 0792 EDT$$CPY_MEM (.LEN, CH$PTR (.OLD_BUKT, .EDT$$A_CUR_BUF [TBCB_LINE_ADDR]), !
223 0793 CH$PTR (.EDT$$A_WK_BUK, WFB_FIXED_SIZE));
224 0794
225 0795 + Discard the copy of the old bucket.
226 0796
227 0797 EDT$$DEA_HEAP (%REF (WF_BUKT_SIZE), OLD_BUKT);
228 0798 EDT$$A_WK_BUK [WFB_END] = .LEN + WFB_FIXED_SIZE;
229 0799 EDT$$G_WK_MODFD = T;
230 0800
231 0801 IF (.OLD_NEXT EQL 0)
232 0802 THEN
233 0803 BEGIN
234 0804
235 0805 IF (.ORIG_BUKT EQL .EDT$$A_CUR_BUF [TBCB_LAST_BUKT])
236 0806 THEN
237 0807 EDT$$A_CUR_BUF [TBCB_LAST_BUKT] = .NEW_BUKT;
238 0808
239 0809 END
240 0810 ELSE
241 0811 BEGIN
242 0812 EDT$$WF_MAKECUR (.OLD_NEXT);
243 0813 EDT$$A_WK_BUK [WFB_PREV_BUKT] = .NEW_BUKT;
244 0814 EDT$$G_WK_MODFD = T;
245 0815 END;
246 0816
247 0817 EDT$$WF_MAKECUR (.ORIG_BUKT)
248 0818 END;
249 0819
250 0820 RETURN (1);
251 0821 END;
```

! of routine EDT\$\$WF\_SPLTBUK

.TITLE EDT\$WFSPLBKT EDT\$WFSPLBKT - split the current bucket

.IDENT \V04-000\

.EXTRN EDT\$K\_PUT, EDT\$\$ALO\_HEAP  
.EXTRN EDT\$\$DEA\_HEAP, EDT\$\$CALLWIO  
.EXTRN EDT\$\$WF\_NEWBUK, EDT\$\$WF\_ALOBUF  
.EXTRN EDT\$\$WF\_MAKECUR  
.EXTRN EDT\$\$Z\_WF\_DESC, EDT\$\$G\_WK\_AVAIL  
.EXTRN EDT\$\$G\_WK\_GRTSTBUK  
.EXTRN EDT\$\$A\_CUR\_BUF, EDT\$\$A\_WK\_BUK

|  |    |           |           | OFFC 00000 |    |       |       |
|--|----|-----------|-----------|------------|----|-------|-------|
|  |    | 5B        | 00000000G | 00         | 9E | 00002 |       |
|  |    | 5A        | 00000000G | 00         | 9E | 00009 |       |
|  |    | 5E        |           | 08         | C2 | 00010 |       |
|  |    | 50        |           | 6A         | D0 | 00013 |       |
|  |    | 53        | 02        | A0         | 9E | 00016 |       |
|  |    | 59        |           | 63         | 3C | 0001A |       |
|  |    | 54        | 00000000G | 00         | D0 | 0001D |       |
|  |    | 58        |           | 54         | D0 | 00024 |       |
|  |    | 52        | 00000000G | 00         | D0 | 00027 |       |
|  |    | 51        |           | 6B         | D0 | 0002E |       |
|  | 04 | A0        |           | 61         | D1 | 00031 |       |
|  |    |           |           | 25         | 12 | 00035 |       |
|  |    |           |           | 52         | D5 | 00037 |       |
|  |    |           |           | 05         | 13 | 00039 |       |
|  |    | 63        |           | 52         | B0 | 0003B |       |
|  |    |           |           | 07         | 11 | 0003E |       |
|  |    | 63        | 00000000G | 00         | B0 | 00040 | 1\$:  |
|  |    | 00000000G | 00        | 01         | D0 | 00047 | 2\$:  |
|  |    |           |           | 58         | DD | 0004E |       |
|  |    |           |           | 59         | DD | 00050 |       |
|  |    | 00000000G | 00        | 02         | FB | 00052 |       |
|  |    |           |           | 00CB       | 31 | 00059 |       |
|  | 57 | 04        | A0        | 61         | C3 | 0005C | 3\$:  |
|  |    | 04        | A0        | 61         | D0 | 00061 |       |
|  |    |           |           | 52         | D5 | 00065 |       |
|  |    |           |           | 08         | 13 | 00067 |       |
|  |    | 56        |           | 52         | D0 | 00069 |       |
|  |    | 63        |           | 52         | B0 | 0006C |       |
|  |    |           |           | 0A         | 11 | 0006F |       |
|  |    | 56        | 00000000G | 00         | D0 | 00071 | 4\$:  |
|  |    | 63        |           | 56         | B0 | 00078 |       |
|  |    |           | 00000000G | 00         | 9F | 0007B | 5\$:  |
|  |    |           |           | 54         | DD | 00081 |       |
|  |    |           | 00000000G | 8F         | DD | 00083 |       |
|  |    | 00000000G | 00        | 03         | FB | 00089 |       |
|  |    |           | 04        | AE         | 9F | 00090 |       |
|  |    | 04        | AE        | 8F         | 3C | 00093 |       |
|  |    |           | 04        | AE         | 9F | 00099 |       |
|  |    | 00000000G | 00        | 02         | FB | 0009C |       |
|  |    |           | 03        | 50         | E8 | 000A3 |       |
|  |    |           |           | 00B2       | 31 | 000A6 |       |
|  |    | 50        |           | 6A         | D0 | 000A9 | 6\$:  |
|  | 04 | BE        | 00000000G | 60         | 8F | 28    | 000AC |
|  |    |           |           | 00         | FB | 000B3 |       |
|  |    |           |           | 50         | D0 | 000BA |       |
|  |    | 02        | A0        | 59         | B0 | 000BD |       |
|  |    |           |           | 60         | B0 | 000C1 |       |
|  |    | 51        |           | 6B         | D0 | 000C4 |       |
|  | 08 | 51        | 04        | 61         | C1 | 000C7 |       |
|  |    | A0        |           | 57         | 28 | 000CC |       |

| .EXTRN | EDTSSG_WK_CURBUK                              |      |
|--------|-----------------------------------------------|------|
| .EXTRN | EDTSSG_WK_MODFD                               |      |
| .PSECT | _EDTSCODE,NOWRT, SHR, PIC,2                   |      |
| .ENTRY | EDTSSWF_SPLTBUK, Save R2,R3,R4,R5,R6,R7,R8,-; | 0666 |
| MOVAB  | EDTSSA_CUR BUF, R11                           |      |
| MOVAB  | EDTSSA_WK_BUK, R10                            |      |
| SUBL2  | #8, SP                                        |      |
| MOVL   | EDTSSA_WK_BUK, R0                             | 0733 |
| MOVAB  | 2(R0), R3                                     |      |
| MOVZWL | (R3), OLD_NEXT                                |      |
| MOVL   | EDTSSG_WK_CURBUK, R4                          | 0734 |
| MOVL   | R4, ORIG_BUKT                                 |      |
| MOVL   | EDTSSG_WK_AVAIL, R2                           | 0746 |
| MOVL   | EDTSSA_CUR BUF, R1                            | 0739 |
| CMPL   | (R1), 4(R0)                                   |      |
| BNEQ   | 3\$                                           |      |
| TSTL   | R2                                            | 0746 |
| BEQL   | 1\$                                           |      |
| MOVW   | R2, (R3)                                      | 0748 |
| BRB    | 2\$                                           |      |
| MOVW   | EDTSSG_WK_GRTSTBUK, (R3)                      | 0750 |
| MOVL   | #1, EDTSSG_WK_MODFD                           | 0752 |
| PUSHL  | ORIG_BUKT                                     | 0753 |
| PUSHL  | OLD_NEXT                                      |      |
| CALLS  | #2, EDTSSWF_NEWBUK                            |      |
| BRW    | 9\$                                           |      |
| SUBL3  | (R1), 4(R0), LEN                              | 0766 |
| MOVL   | (R1), 4(R0)                                   | 0767 |
| TSTL   | R2                                            | 0769 |
| BEQL   | 4\$                                           |      |
| MOVL   | R2, NEW_BUKT                                  | 0771 |
| MOVW   | R2, (R3)                                      |      |
| BRB    | 5\$                                           |      |
| MOVL   | EDTSSG_WK_GRTSTBUK, NEW_BUKT                  | 0773 |
| MOVW   | NEW_BUKT, (R3)                                |      |
| PUSHAB | EDTSSZ_WF_DESC                                | 0775 |
| PUSHL  | R4                                            |      |
| PUSHL  | #EDTSK_PUT                                    |      |
| CALLS  | #3, EDTSSCALLWIO                              |      |
| PUSHAB | OLD_BUKT                                      | 0780 |
| MOVZWL | #512, 4(SP)                                   |      |
| PUSHAB | 4(SP)                                         |      |
| CALLS  | #2, EDTSSALO_HEAP                             |      |
| BLBS   | R0, 6\$                                       |      |
| BRW    | 10\$                                          |      |
| MOVL   | EDTSSA_WK_BUK, R0                             | 0782 |
| MOVW   | #512, (R0), @OLD_BUKT                         |      |
| CALLS  | #0, EDTSSWF_ALOBUF                            | 0786 |
| MOVL   | EDTSSA_WK_BUK, R0                             | 0787 |
| MOVW   | OLD_NEXT, 2(R0)                               |      |
| MOVW   | ORIG_BUKT, (R0)                               | 0788 |
| MOVL   | EDTSSA_CUR BUF, R1                            | 0793 |
| ADDL3  | (R1), OLD_BUKT, R1                            |      |
| MOVW   | LEN, (R1), 8(R0)                              |      |

EDT\$WFSPLBKT  
V04-000

EDT\$WFSPLBKT - split the current bucket  
EDT\$\$WF\_SPLTBUK - split the current bucket

C 16  
16-Sep-1984 02:15:01  
14-Sep-1984 12:25:45

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

|    |    |    |           |    |    |       |        |                            |   |      |
|----|----|----|-----------|----|----|-------|--------|----------------------------|---|------|
|    |    |    | 04        | AE | 9F | 000D1 | PUSHAB | OLD_BUKT                   | : | 0797 |
|    |    |    | 04        | AE | 9F | 000D4 | MOVZWL | #512, 4(SP)                | : |      |
|    |    |    | 04        | AE | 9F | 000DA | PUSHAB | 4(SP)                      | : |      |
|    |    |    | 00000000G | 00 | 02 | FB    | CALLS  | #2, EDT\$\$DEA_HEAP        | : |      |
|    |    |    |           | 50 | 6A | D0    | MOVL   | EDT\$\$A_WK_BUR, R0        | : | 0798 |
|    |    |    | 04        | A0 | A7 | 9E    | MOVAB  | 8(R7), -4(R0)              | : |      |
|    |    |    | 00000000G | 00 | 01 | D0    | MOVL   | #1, EDT\$\$G_WK_MODFD      | : | 0799 |
|    |    |    |           |    | 59 | D5    | TSTL   | OLD_NEXT                   | : | 0801 |
|    |    |    |           |    | 11 | 12    | BNEQ   | 7\$                        | : |      |
|    |    |    |           | 50 | 6B | D0    | MOVL   | EDT\$\$A_CUR_BUF, R0       | : | 0805 |
| 58 | 10 | A0 |           | 10 | 00 | ED    | CMPZV  | #0, #16, 16(R0), ORIG_BUKT | : |      |
|    |    |    |           |    | 1C | 12    | BNEQ   | 8\$                        | : |      |
|    |    |    | 10        | A0 | 56 | B0    | MOVW   | NEW_BUKT, 16(R0)           | : | 0807 |
|    |    |    |           |    | 16 | 11    | BRB    | 8\$                        | : | 0801 |
|    |    |    |           |    | 59 | DD    | PUSHL  | OLD_NEXT                   | : | 0812 |
|    |    |    | 00000000G | 00 | 01 | FB    | CALLS  | #1, EDT\$\$WF_MAKECUR      | : |      |
|    |    |    |           | 50 | 6A | D0    | MOVL   | EDT\$\$A_WK_BUK, R0        | : | 0813 |
|    |    |    |           | 60 | 56 | B0    | MOVW   | NEW_BUKT, (R0)             | : |      |
|    |    |    | 00000000G | 00 | 01 | D0    | MOVL   | #1, EDT\$\$G_WK_MODFD      | : | 0814 |
|    |    |    |           |    | 58 | DD    | PUSHL  | ORIG_BUKT                  | : | 0817 |
|    |    |    | 00000000G | 00 | 01 | FB    | CALLS  | #1, EDT\$\$WF_MAKECUR      | : |      |
|    |    |    |           | 50 | 01 | D0    | MOVL   | #1, R0                     | : | 0820 |
|    |    |    |           |    |    | 04    | RET    |                            | : |      |
|    |    |    |           |    | 50 | D4    | CLRL   | R0                         | : | 0821 |
|    |    |    |           |    | 04 | 0012D | RET    |                            | : |      |

; Routine Size: 302 bytes, Routine Base: \_EDT\$CODE + 0000

: 252 0822 1  
: 253 0823 1 !<BLF/PAGE>

EDT\$WFSPLBKT  
V04-000

EDT\$WFSPLBKT - split the current bucket  
EDT\$WF\_SPLTBUK - split the current bucket

D 16  
16-Sep-1984 02:15:01  
14-Sep-1984 12:25:45

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

Page 8  
(4)

: 255 0824 1 END  
: 256 0825 1  
: 257 0826 0 ELUDOM

! of module EDT\$WFSPLBKT

# PSECT SUMMARY

| Name       | Bytes | Attributes                                              |
|------------|-------|---------------------------------------------------------|
| _EDT\$CODE | 302   | NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2) |

# Library Statistics

| File                               | -----<br>Total | Symbols<br>Loaded | -----<br>Percent | Pages<br>Mapped | Processing<br>Time |
|------------------------------------|----------------|-------------------|------------------|-----------------|--------------------|
| \$255\$DUA28:[EDT.SRC]EDT.L32;1    | 377            | 31                | 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\$:WFSPLBKT/OBJ=OBJ\$:WFSPLBKT MSRC\$:WFSPLBKT.BLI/UPDATE=(ENH\$:WFSP  
LBKT)

: Size: 302 code + 0 data bytes  
: Run Time: 00:16.9  
: Elapsed Time: 00:20.8  
: Lines/CPU Min: 2939  
: Lexemes/CPU-Min: 10256  
: Memory Used: 120 pages  
: Compilation Complete

0141 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

