

[illegible]

```
DDDDDDDD  EEEEEEEEE  AAAAAA  CCCCCCCC  CCCCCCCC  SSSSSSSS
DDDDDDDD  EEEEEEEEE  AAAAAA  CCCCCCCC  CCCCCCCC  SSSSSSSS
DD      DD  EE      AA      AA  CC      CC      SS
DD      DD  EE      AA      AA  CC      CC      SS
DD      DD  EE      AA      AA  CC      CC      SS
DD      DD  EE      AA      AA  CC      CC      SS
DD      DD  EEEEEEEE  AA      AA  CC      CC      SSSSSS
DD      DD  EEEEEEEE  AA      AA  CC      CC      SSSSSS
DD      DD  EE      AAAAAAAAAA  CC      CC      SS
DD      DD  EE      AAAAAAAAAA  CC      CC      SS
DD      DD  EE      AA      AA  CC      CC      SS
DD      DD  EE      AA      AA  CC      CC      SS
DDDDDDDD  EEEEEEEEE  AA      AA  CCCCCCCC  CCCCCCCC  SSSSSSSS
DDDDDDDD  EEEEEEEEE  AA      AA  CCCCCCCC  CCCCCCCC  SSSSSSSS
```

```
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
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```

D 7
16-Sep-1984 02:15:44
14-Sep-1984 12:46:38

VAX-11 Bliss-32 V4.0-742
[MTAACP.SRC]DEACCS.B32;1

Page 1
(1)

END
V04

```
0001 0
0002 0 MODULE DEACCS (LANGUAGE (BLISS32) ,
0003 0 IDENT = 'V04-000' ,
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: MTAACP
0033 1
0034 1 ABSTRACT:
0035 1 this modules executes the deaccess function
0036 1
0037 1
0038 1 ENVIRONMENT:
0039 1
0040 1 starlet operating system, including privileged system services
0041 1 and internal exec routines.
0042 1
0043 1 --
0044 1
0045 1
0046 1
0047 1 AUTHOR: D. H. GILLESPIE, CREATION DATE: 18-JUL-1977
0048 1
0049 1 MODIFIED BY:
0050 1
0051 1 V02-008 REFORMAT Maria del C. Nasr 30-Jun-1980
0052 1
0053 1 A0007 SPR27676 Maria del C. Nasr 14-Dec-1979 16:20
0054 1 Update transaction count in VCB for each file access and
0055 1 deaccess.
0056 1
0057 1 A0006 R. I. Hustvedt, 16-Oct-1979 12:30
```



```

58      0058 1  !
59      0059 1  !**
60      0060 1
61      0061 1  LIBRARY 'SYSS$LIBRARY:LIB.L32';
62      0062 1
63      0063 1  REQUIRE 'SRC$:MTADEF.B32';
64      0447 1
65      0448 1  FORWARD ROUTINE
66      0449 1      DO_DEACCESS      : COMMON_CALL NOVALUE,      ! kernel mode deaccess
67      0450 1      MTA_DEACCESS     : NOPRES_NOVALUE,          ! main deaccess function
68      0451 1      ZERO_CHANNEL     : COMMON_CALL NOVALUE;      ! zeros user's window pter
69      0452 1
70      0453 1  EXTERNAL ROUTINE
71      0454 1      CLOSE_FILE       : L$CLOSE_FILE,             ! close file accessed for write
72      0455 1      DEALLOCATE,      :                          ! return space to system pool
73      0456 1      GET_FIB          : COMMON_CALL,              ! get file information block
74      0457 1      REWIND_VOL_SET   : COMMON_CALL;              ! rewind volume set
75      0458 1
76      0459 1  EXTERNAL
77      0460 1      CURRENT_WCB      : REF BBLOCK,                ! address of current window control block
78      0461 1      IO_PACKET       : REF BBLOCK,                ! address of current io request packet
79      0462 1
80      0463 1      ! system pcb vector
81      0464 1
82      0465 1      SCH$GL_PCBVEC    : REF VECTOR ADDRESSING_MODE (ABSOLUTE),
83      0466 1      USER_STATUS     : VECTOR [2];
84      0467 1
```

```
86 0468 1 GLOBAL ROUTINE MTA_DEACCESS : NOPRES NOVALUE =
87 0469 1
88 0470 1 ++
89 0471 1
90 0472 1 FUNCTIONAL DESCRIPTION:
91 0473 1     this routine executes the deaccess function
92 0474 1
93 0475 1 CALLING SEQUENCE:
94 0476 1     mta_deaccess()
95 0477 1
96 0478 1 INPUT PARAMETERS:
97 0479 1     none
98 0480 1
99 0481 1 IMPLICIT INPUTS:
100 0482 1     io_packet - address of user's io request packet
101 0483 1     current_vcb - address of current volume control block
102 0484 1     current_wcb - address of current window
103 0485 1
104 0486 1 OUTPUT PARAMETERS:
105 0487 1     none
106 0488 1
107 0489 1 IMPLICIT OUTPUTS:
108 0490 1     user's window pter zeroed
109 0491 1     window returned to system space and disconnected from volume control block
110 0492 1
111 0493 1 ROUTINE VALUE:
112 0494 1     none
113 0495 1
114 0496 1 SIDE EFFECTS:
115 0497 1     none
116 0498 1
117 0499 1 USER ERRORS:
118 0500 1     ss$_badparam - irrelevant fib data
119 0501 1
120 0502 1 --
121 0503 1
122 0504 2 BEGIN
123 0505 2
124 0506 2 EXTERNAL REGISTER
125 0507 2     COMMON_REG;
126 0508 2
127 0509 2 LOCAL
128 0510 2     ABD      : REF BBLOCKVECTOR [, ABD$K_LENGTH],      ! buffer descriptors
129 0511 2     FIB      : REF BBLOCK,                                ! user's fib
130 0512 2
131 0513 2     ! write indicator taken from window control block
132 0514 2     !
133 0515 2     WRITE_IND : BITVECTOR [8];
134 0516 2
135 0517 2     ABD = .BBLOCK[.IO PACKET[IRPSL_SVAPTE], AIB$K_DESCRIPTOR];
136 0518 2     FIB = GET_FIB(.ABD);
137 0519 2     WRITE_IND = NOT .CURRENT_WCB[WCB$V_READ]
138 0520 2         AND
139 0521 2         NOT .CURRENT_VCB[VCB$V_NOWRITE];
140 0522 2
141 0523 2     ! deaccess file by returning window to non_paged system space and dis
142 0524 2     ! connecting it from the volume control block
```



```
143 0525 2 !
144 0526 2 !
145 0527 2 !
146 0528 2 !
147 0529 2 !
148 0530 2 !
149 0531 2 !
150 0532 2 !
151 0533 2 !
152 0534 2 !
153 0535 2 !
154 0536 2 !
155 0537 2 !
156 0538 2 !
157 0539 2 !
158 0540 2 !
159 0541 2 !
160 0542 2 !
161 0543 2 !
162 0544 2 !
163 0545 2 !
164 0546 2 !
165 0547 2 !
166 0548 2 !
167 0549 2 !
168 0550 2 !
169 0551 2 !
170 0552 2 !
171 0553 2 !
172 0554 2 !
173 0555 2 !
174 0556 1 !

!
KERNEL_CALL(DO_DEACCESS);
!
! now check if trailers should be written
!
IF .WRITE_IND
THEN
    CLOSE_FILE();
!
! now if rewind specified on deaccess, rewind volume set
!
IF .FIB[FIBSV_REWIND]
THEN
    REWIND_VOL_SET();
!
! report any block count difference, user_status[1] = difference where neg
! number means read more than should have.
!
IF NOT .WRITE_IND
AND
.BBLOCK[.CURRENT_VCB[VCBSL_VPFL], VVP$BLOCKDIF] NEQ 0
THEN
    BEGIN
        USER_STATUS[1] = .BBLOCK[.CURRENT_VCB[VCBSL_VPFL], VVP$BLOCKDIF];
        .BBLOCK[.CURRENT_VCB[VCBSL_VPFL], VVP$BLOCKDIF] = 0;
        ERR_EXIT(SS$BLOCKCNTERR);
    END;
END;

! end of routine
```

```
.TITLE DEACCS
.IDENT \V04-000\

.EXTRN CLOSE_FILE, DEALLOCATE
.EXTRN GET_FIB, REWIND_VOL_SET
.EXTRN CURRENT_WCB, IO_PACKET
.EXTRN SCH$GL_PCBVEC, USER_STATUS
.EXTRN SYSSCMRNL
```

```
.PSECT $CODE$,NOWRT,2
```

```
.ENTRY MTA_DEACCESS, Save nothing
MOVL IO_PACKET, R0
MOVL @44(R0), ABD
PUSHL ABD
CALLS #1, GET_FIB
MOVL R0, FIB
MOVL CURRENT_WCB, R0
EXTZV #0, #1, 11(R0), R0
EXTZV #7, #1, 11(CURRENT_VCB), R1
BISL2 R1, R0
MCOMB R0, WRITE_IND
CLRL -(SP)
```

```
50 0000G CF 0000 00000
50 2C B0 D0 00002
0000G CF 50 DD 0000B
53 01 FB 0000D
50 0000G CF 50 D0 00012
01 00 EF 0001A
01 07 EF 00020
50 51 C8 00026
52 50 92 00029
7E D4 0002C
```

```
: 0468
: 0517
: 0518
: 0519
: 0521
: 0520
: 0526
```

DEACCS
V04-000

H 7
16-Sep-1984 02:15:44
14-Sep-1984 12:46:38

VAX-11 Bliss-32 V4.0-742
[MTAACP.SRC]DEACCS.B32;1

Page 5
(2)

			SE	DD	0002E	PUSHL	SP		
		0000V	CF	9F	00030	PUSHAB	DO_DEACCESS		
	00000000G	9F	03	F6	00034	CALLS	#3, @#SYSS\$CMKRNL		
		03	52	E9	00038	BLBC	WRITE_IND, 1\$		0531
			0000G	30	0003E	BSBW	CLOSE_FILE		0533
05		63	03	E1	00041	BBC	#3, (FIB), 2\$		0538
	0000G	CF	00	FB	00045	CALLS	#0, REWIND_VOL_SET		0540
		1D	52	E8	0004A	BLBS	WRITE_IND, 3\$		0546
		50	3C	AB	D0	MOVL	60(CURRENT_VCB), R0		0548
			01AC	C0	D5	TSTL	428(R0)		
				13	13	BEQL	3\$		
		50	3C	AB	D0	MOVL	60(CURRENT_VCB), R0		0551
	0000G	CF	01AC	C0	D0	MOVL	428(R0), USER_STATUS+4		
			01AC	C0	D4	CLRL	428(R0)		0552
			0940	8F	BF	CHMU	#2368		0553
				04	0006A	RET			0556
					3\$:				

; Routine Size: 107 bytes, Routine Base: \$CODE\$ + 0000

; 175 0557 1

END
V04

```
177 0558 1 ROUTINE DO_DEACCESS : COMMON_CALL NOVALUE =
178 0559 1
179 0560 1 ++
180 0561 1
181 0562 1 FUNCTIONAL DESCRIPTION:
182 0563 1     this routine does the deaccess cleanup to the system data base
183 0564 1
184 0565 1 CALLING SEQUENCE:
185 0566 1     do_deaccess(), called in kernel mode
186 0567 1
187 0568 1 INPUT PARAMETERS:
188 0569 1     none
189 0570 1
190 0571 1 IMPLICIT INPUTS:
191 0572 1     current_vcb - address of current volume control block
192 0573 1     current_wcb - address of current window control block
193 0574 1
194 0575 1 OUTPUT PARAMETERS:
195 0576 1     none
196 0577 1
197 0578 1 IMPLICIT OUTPUTS:
198 0579 1     none
199 0580 1
200 0581 1 ROUTINE VALUE:
201 0582 1     none
202 0583 1
203 0584 1 SIDE EFFECTS:
204 0585 1     none
205 0586 1
206 0587 1 --
207 0588 1
208 0589 2 BEGIN
209 0590 2
210 0591 2 EXTERNAL REGISTER
211 0592 2     COMMON_REG;
212 0593 2
213 0594 2 LOCAL
214 0595 2     AST_BLOCK,      ! address of ast control block supplied by user
215 0596 2     PCB : REF BBLOCK; ! address of user's process control block
216 0597 2
217 0598 2 ! if user ast control block was supplied for user label processing but not
218 0599 2 ! used then return it to the system pool and credit the user's ast quota.
219 0600 2 ! get saved address of ast block
220 0601 2
221 0602 2 AST_BLOCK = .CURRENT_VCB[VCB$L_USRLBLAST];
222 0603 2
223 0604 2 IF .AST_BLOCK NEQ 0
224 0605 2 THEN
225 0606 2     BEGIN
226 0607 2         DEALLOCATE(.AST_BLOCK);           ! return to system pool
227 0608 2         CURRENT_VCB[VCB$L_USRLBLAST] = 0; ! note this fact in vcb
228 0609 2
229 0610 2         ! get user's pcb address
230 0611 2
231 0612 2         PCB = .SCH$GL_PCBVEC[(IO_PACKET[IRP$L_PID])<0, 16>];
232 0613 2
233 0614 2         ! credit user's ast quota
```



```
: 234      0615      3
: 235      0616      3
: 236      0617      3
: 237      0618      3
: 238      0619      3
: 239      0620      3
: 240      0621      3
: 241      0622      3
: 242      0623      3
: 243      0624      3
: 244      0625      3
: 245      0626      3
: 246      0627      3
: 247      0628      3
: 248      0629      1

!
PCB[PCBSW_ASTCNT] = .PCB[PCBSW_ASTCNT] + 1;
END;

ZERO_CHANNEL(.IO_PACKET);      ! return zeroed window pter to user
DEALLOCATE(.CURRENT_WCB);      ! return window space
CURRENT_WCB = 0;                ! no longer an outstanding access
CURRENT_VCB[VCBSL_WCB] = 0;     ! update vcb to reflect deaccess
CURRENT_VCB[VCBSV_CANCELIO] = 0; ! cancel no longer applies
CURRENT_VCB[VCBSV_NOWRITE] = 0; ! turn of no write indicator

! decrement transaction count
CURRENT_VCB[VCBSW_TRANS] = .CURRENT_VCB[VCBSW_TRANS] - 1;
END;                                ! end of routine
```

```
0000 00000 DO_DEACCESS:
50      44      AB D0 00002      .WORD      Save nothing      : 0558
23      13 00006      MOVL      68(CURRENT_VCB), AST_BLOCK      : 0602
50      DD 00008      BEQL      1$      : 0604
0000G CF      01 FB 0000A      PUSHL     AST_BLOCK      : 0607
44      AB D4 0000F      CALLS      #1, DEALLOCATE
51 00000000G 9F D0 00012      CLRL      68(CURRENT_VCB)      : 0608
50      CF D0 00019      MOVL      @#SCH$GL PCBVEC, R1      : 0612
50      OC C0 0001E      MOVL      IO_PACKET, R0
50      60 3C 00021      ADDL2      #12, R0
50      6140 D0 00024      MOVZWL     (R0), R0
38      A0 B6 00028      MOVL      (R1)(R0), PCB
0000G CF DD 0002B 1$:      INCW      56(PCB)      : 0616
0000V CF      01 FB 0002F      PUSHL     IO_PACKET      : 0619
0000G CF      01 FB 00034      CALLS      #1, ZERO_CHANNEL
0000G CF      01 FB 00038      PUSHL     CURRENT_WCB      : 0620
38      AB D4 0003D      CALLS      #1, DEALLOCATE
OB AB      A0 8F 8A 00044      CLRL      CURRENT_WCB      : 0621
OC AB      B7 00049      CLRL      56(CURRENT_VCB)      : 0622
04 0004C      DECW      #160, 11(CURRENT_VCB)      : 0624
RET      12(CURRENT_VCB)      : 0628
: 0629
```

; Routine Size: 77 bytes, Routine Base: \$CODE\$ + 006B

; 249 0630 1

```
251 0631 1 GLOBAL ROUTINE ZERO_CHANNEL (PACKET) : COMMON_CALL NOVALUE =
252 0632 1
253 0633 1 !++
254 0634 1
255 0635 1 FUNCTIONAL DESCRIPTION:
256 0636 1
257 0637 1 This routine zeroes out the window pointer being returned to
258 0638 1 the user for his channel control block. It also credits one to the
259 0639 1 user's open file quota.
260 0640 1 This routine must be executed in kernel mode.
261 0641 1
262 0642 1 CALLING SEQUENCE:
263 0643 1 zero_channel ()
264 0644 1
265 0645 1 INPUT PARAMETERS:
266 0646 1 none
267 0647 1
268 0648 1 IMPLICIT INPUTS:
269 0649 1 io_packet: i/o packet of request
270 0650 1
271 0651 1 OUTPUT PARAMETERS:
272 0652 1 none
273 0653 1
274 0654 1 IMPLICIT OUTPUTS:
275 0655 1 none
276 0656 1
277 0657 1 ROUTINE VALUE:
278 0658 1 none
279 0659 1
280 0660 1 SIDE EFFECTS:
281 0661 1 channel window pointer cleared, file quota bumped
282 0662 1
283 0663 1 --
284 0664 1
285 0665 2 BEGIN
286 0666 2
287 0667 2 EXTERNAL REGISTER
288 0668 2 COMMON_REG;
289 0669 2
290 0670 2 MAP
291 0671 2 PACKET : REF BBLOCK;
292 0672 2
293 0673 2 LOCAL
294 0674 2 ABD : REF BBLOCKVECTOR [, ABD$C_LENGTH], ! buffer descriptors
295 0675 2 JIB : REF BBLOCK, ! Job information block
296 0676 2 PCB : REF BBLOCK; ! address of user process control block
297 0677 2
298 0678 2 ABD = .BBLOCK[.PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPTOR];
299 0679 2 ABD[ABD$C_WINDOW, ABD$W_COUNT] = 4;
300 0680 2 .ABD[ABD$C_WINDOW, ABD$W_TEXT] + ABD[ABD$C_WINDOW, ABD$W_TEXT] + 1 = 0;
301 0681 2 PCB = .SCH$GL PCBVEC[(.PACKET[IRP$L_PID])<0, 16>];
302 0682 2 JIB = .PCB[PCB$L_JIB];
303 0683 2 JIB[JIB$W_FILCNT] = .JIB[JIB$W_FILCNT] + 1;
304 0684 1 END; ! end of routine ZERO_CHANNEL
```


				0004 00000	.ENTRY	ZERO CHANNEL, Save R2		0631
	52	04	AC	D0 00002	MOVL	PACKET, R2	:	0678
	51	2C	B2	D0 00006	MOVL	@44(R2), ABD	:	
02	A1		04	B0 0000A	MOVW	#4, 2(ABD)	:	0679
	50		61	3C 0000E	MOVZWL	(ABD), R0	:	0680
		01	A140	9F 00011	PUSHAB	1(ABD)[R0]	:	
			9E	D4 00015	CLRL	@(SP)+	:	
	51	00000000G	9F	D0 00017	MOVL	@#SCH\$GL PCBVEC, R1	:	0681
	50	0C	A2	3C 0001E	MOVZWL	12(R2), R0	:	
	50		6140	D0 00022	MOVL	(R1)[R0], PCB	:	
	50	0080	C0	D0 00026	MOVL	128(PCB), JIB	:	0682
		30	A0	B6 0002B	INCW	48(JIB)	:	0683
			04	0002E	RET		:	0684

; Routine Size: 47 bytes, Routine Base: \$CODE\$ + 00B8

:	305	0685	1
:	306	0686	1 END
:	307	0687	1
:	308	0688	0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	231	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPI,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	24	0	1000	00:01.9

COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:DEACCS/OBJ=OBJ\$:DEACCS MSRC\$:DEACCS/UPDATE=(ENH\$:DEACCS)

; Size: 231 code + 0 data bytes
; Run Time: 00:09.4

DEACCS
V04-000

M 7
16-Sep-1984 02:15:44

VAX-11 Bliss-32 V4.0-742

Page 10

; Elapsed Time: 00:38.2
; Lines/CPU Min: 4400
; Lexemes/CPU-Min: 21780
; Memory Used: 88 pages
; Compilation Complete

END
V04

; R

0254 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

