

BBBBBBBBBBBB		AAAAAAA		DDDDDDDDDD		BBBBBBBBBBBB		LLL		KKK		KKK
BBBBBBBBBBBB		AAAAAAA		DDDDDDDDDD		BBBBBBBBBBBB		LLL		KKK		KKK
BBBBBBBBBBBB		AAAAAAA		DDDDDDDDDD		BBBBBBBBBBBB		LLL		KKK		KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK		KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK		KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK		KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBBBBBBBBBBB		AAA	AAA	DDD	DDD	BBBBBBBBBBBB		LLL		KKK	KKK	KKK
BBBBBBBBBBBB		AAA	AAA	DDD	DDD	BBBBBBBBBBBB		LLL		KKKKKKKK		
BBBBBBBBBBBB		AAA	AAA	DDD	DDD	BBBBBBBBBBBB		LLL		KKKKKKKK		
BBB	BBB	AAAAAAAAAAAA		DDD	DDD	BBB	BBB	LLL		KKK	KKK	
BBB	BBB	AAAAAAAAAAAA		DDD	DDD	BBB	BBB	LLL		KKK	KKK	
BBB	BBB	AAAAAAAAAAAA		DDD	DDD	BBB	BBB	LLL		KKK	KKK	
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBB	BBB	AAA	AAA	DDD	DDD	BBB	BBB	LLL		KKK	KKK	KKK
BBBBBBBBBBBB		AAA	AAA	DDDDDDDDDD		BBBBBBBBBBBB		LLLLLLLLLLLL		KKK	KKK	KKK
BBBBBBBBBBBB		AAA	AAA	DDDDDDDDDD		BBBBBBBBBBBB		LLLLLLLLLLLL		KKK	KKK	KKK
BBBBBBBBBBBB		AAA	AAA	DDDDDDDDDD		BBBBBBBBBBBB		LLLLLLLLLLLL		KKK	KKK	KKK

GGGGGGGG	EEEEEEEEEE	TTTTTTTTTT	RRRRRRRR	EEEEEEEEEE	QQQQQQ	
GGGGGGGG	EEEEEEEEEE	TTTTTTTTTT	RRRRRRRR	EEEEEEEEEE	QQQQQQ	
GG	EE	TT	RR	EE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EEEEEEEE	TT	RRRRRRRR	EEEEEEEE	QQ	
GG	EEEEEEEE	TT	RRRRRRRR	EEEEEEEE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EE	TT	RR	EE	QQ	
GG	EEEEEEEEEE	TT	RR	EEEEEEEEEE	QQ	
GGGGGG	EEEEEEEEEE	TT	RR	EEEEEEEEEE	QQQQ	
GGGGGG					QQQQ	

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

```
0001 0 MODULE GETREQ (MAIN=MAIN_BAD,  
0002 0     LANGUAGE=(BLISS32),  
0003 0     IDENT = 'V04-000'  
0004 0 ) =  
0005 1 BEGIN  
0006 1  
0007 1 |  
0008 1 |*****  
0009 1 |*  
0010 1 |*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0011 1 |*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0012 1 |*  ALL RIGHTS RESERVED.  
0013 1 |*  
0014 1 |*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0015 1 |*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0016 1 |*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0017 1 |*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0018 1 |*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0019 1 |*  TRANSFERRED.  
0020 1 |*  
0021 1 |*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0022 1 |*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0023 1 |*  CORPORATION.  
0024 1 |*  
0025 1 |*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0026 1 |*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0027 1 |*  
0028 1 |*  
0029 1 |*****  
0030 1 |  
0031 1 |++  
0032 1 |  
0033 1 |FACILITY:  DYNAMIC BADBLOCK UTILITY  
0034 1 |  
0035 1 |ABSTRACT:  
0036 1 |  
0037 1 |THESE ROUTINES PERFORM THE INTERACTIONS WITH THE FILE SYSTEM  
0038 1 |ACP FOR THE BAD BLOCK SCAN UTILITY.SPECIFIC FUNCTIONS ARE  
0039 1 |  
0040 1 |1. READ THE MAILBOX INTERFACING THE UTILITY AND THE ACP  
0041 1 |2. ACCESS THE SUSPECT FILE BY PATCHING ITS UCB INTO  
0042 1 |   A PREVIOUSLY OPENED CCB  
0043 1 |  
0044 1 |ENVIRONMENT:  
0045 1 |  
0046 1 |VAX/VMS OPERATING SYSTEM, VERSION 1.0  
0047 1 |  
0048 1 |--  
0049 1 |  
0050 1 |  
0051 1 |AUTHOR:  THOMAS G. DOPIRAK,    CREATION DATE:  7-JUNE-1978  
0052 1 |  
0053 1 |REVISION HISTORY:  
0054 1 |  
0055 1 |V03-001 LMP0035      L. Mark Pilant,      28-Jun-1982  13:15  
0056 1 |      Correct problems that resulted in informational messages.  
0057 1 |
```

GETREQ
V04-000

N 13
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 2
(1)

```
: 58      0058 1  !***
: 59      0059 1
: 60      0060 1
: 61      0061 1  LIBRARY 'SYSS$LIBRARY:LIB.L32';
: 62      0062 1
: 63      0063 1
: 64      0064 1  FORWARD ROUTINE
: 65      0065 1      MAIN BAD:NOVALUE,
: 66      0066 1      RESTORE_CHANNEL:NOVALUE,
: 67      0067 1      SET_UCB:NOVALUE,
: 68      0068 1      OPEN_CHANNEL,
: 69      0069 1      GET_BBSMSG,
: 70      0070 1      GET_CCB;
: 71      0071 1  EXTERNAL ROUTINE
: 72      0072 1      DATA_INIT:NOVALUE,
: 73      0073 1      SCAN;
```

```
!MAIN PROGRAM
!PLACE SYSS$DISK UCB BACK INTO CCB
!SET UCB ADDRESS IN TO CCB
!OPEN CHANNELS TO F11ACP AND SYSS$DISK
!READ MAIL FROM F11ACP
! get address of CCB of channel
```

GETREQ
V04-000

B 14
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 3
(2)

```

: 75      0074 1  |
: 76      0075 1  |OWN STORAGE
: 77      0076 1  |
: 78      0077 1  |
: 79      0078 1  |GLOBAL
: 80      0079 1  |CHANNEL:WORD,      !CHANNEL NUMBER OF SYSSDISK
: 81      0080 1  |MBX CHANNEL:WORD,  !CHANNEL NUMBER OF F11ACP MAILBOX
: 82      0081 1  |CCBUCB,            !ADDRESS OF UCB ENTRY IN CCB
: 83      0082 1  |OLD UCB,           !UCB ADDRESS FOR OPEN CHANNEL
: 84      0083 1  |SYSSDISK_UCB,      !SAVED ADDRESS OF UCB FOR SYSSDISK
: 85      0084 1  |ACP_MAIL:BLOCK[BBSSC_LENGTH,BYTE]; !MESSAGE FROM F11ACP
: 86      0085 1  |
: 87      0086 1  |READ ONLY STORAGE
: 88      0087 1  |
: 89      0088 1  |
: 90      0089 1  |BIND
: 91      0090 1  |
: 92      0091 1  |DISK_DESC=UPLIT(%CHARCOUNT('SYSSDISK'),
: 93      0092 1  |                        UPLIT BYTE('SYSSDISK'));
: 94      0093 1  |
: 95      0094 1  |LITERAL
: 96      0095 1  |TRUE=1;
```

```

98 0096 1 ROUTINE MAIN_BAD: NOVALUE=
99 0097 1
100 0098 1 ++
101 0099 1 FUNCTIONAL DESCRIPTION:
102 0100 1
103 0101 1 MAIN PROGRAM ON BADBLOCK SCAN UTILITY. THIS ROUTINE RECEIVES
104 0102 1 MAIL FROM THE F11ACP INDICATING FILES THAT HAVE BEEN DELETED
105 0103 1 AND THAT HAVE ENCOUNTERED HARD DEVICE ERRORS DURING IO.
106 0104 1 MAIN_BAD USES THE UCB ADDRESS(INDICATING WHICH DISK) AND
107 0105 1 FILE=ID(INDICATING THE FILE) TO SEPARATE THE SUSPECT FILE
108 0106 1 INTO ITS 'BAD' AND GOOD PORTIONS. THE BAD PORTIONS ARE
109 0107 1 ADDED ONTO BADBLK.SYS(SYSTEM BADBLOCK FILE) AND THE GOOD
110 0108 1 PORTIONS ARE RECYCLED FOR GENERAL USE.
111 0109 1 SINCE THE DISK INVOLVED MAY BE ALLOCATED, BADBLOCK SCAN
112 0110 1 SLIDES AROUND SYSTEM PROTECTION BY OPENING A CHANNEL TO
113 0111 1 SYSSDISK AND THEN REPLACING THE UCB ADDRESS IN THAT CCB
114 0112 1 BY THAT OF THE SUSPECT DISK.
115 0113 1
116 0114 1 FORMAL PARAMETERS:
117 0115 1
118 0116 1 NONE
119 0117 1
120 0118 1 IMPLICIT INPUTS:
121 0119 1
122 0120 1 NONE
123 0121 1
124 0122 1 IMPLICIT OUTPUTS:
125 0123 1
126 0124 1 NONE
127 0125 1
128 0126 1 ROUTINE VALUE:
129 0127 1 COMPLETION CODES:
130 0128 1
131 0129 1 NONE
132 0130 1
133 0131 1 SIDE EFFECTS:
134 0132 1
135 0133 1 NONE
136 0134 1
137 0135 1 --
138 0136 1
139 0137 2 BEGIN
140 0138 2
141 0139 2 ++
142 0140 2 OPEN CHANNELS TO SYSSDISK AND THE F11ACP MAILBOX
143 0141 2
144 0142 2 IF
145 0143 2 NOT OPEN_CHANNEL()
146 0144 2 THEN
147 0145 2 RETURN;
148 0146 2
149 0147 2 ++
150 0148 2 INITIALIZE TEST DATA
151 0149 2 --
152 0150 2
153 0151 2 DATA_INIT();
154 0152 2
```

```
155 0153 2 !+
156 0154 2 !GET THE ADDRESS OF THE CCB FOR OPEN CHANNEL
157 0155 2
158 0156 3 IF NOT $CMKRNL(ROUTIN=GET_CCB)
159 0157 3 THEN
160 0158 3 RETURN;
161 0159 3
162 0160 2 !+
163 0161 2 !CYCLE THROUGH UNTIL NOTHING IN MAILBOX TO DO
164 0162 2
165 0163 2
166 0164 2 WHILE TRUE DO
167 0165 3 BEGIN
168 0166 3 IF
169 0167 3 NOT GET_BBSMSG()
170 0168 3 THEN
171 0169 4 BEGIN
172 0170 4 RESTORE_CHANNEL();
173 0171 4 RETURN
174 0172 3 END;
175 0173 3
176 0174 2 !+
177 0175 2 !SET UCB FOR SUSPECT FILE INTO OPEN CCB
178 0176 2
179 0177 2 $CMKRNL(ROUTIN=SET_UCB);
180 0178 2
181 0179 2 !+
182 0180 2 !PROCESS THE FILE
183 0181 2
184 0182 2 SCAN();
185 0183 2 END;
186 0184 1 END;
```

```
.TITLE GETREQ
.IDENT \V04-000\
.PSECT $PLITS$,NOWRT,NOEXE,2
4B 53 49 44 24 53 59 53 00000 P.AAB: .ASCII \SYSSDISK\
00000008 00008 P.AAA: .LONG 8
00000000 0000C .ADDRESS P.AAB
.PSECT $GLOBAL$,NOEXE,2
00000 CHANNEL::
00002 .BLKB 2
00002 MBX_CHANNEL::
00004 .BLKB 2
00004 CCBUCB:: .BLKB 4
00008 OLD_UCB::
0000C .BLKB 4
0000C SYSSDISK_UCB::
00010 .BLKB 4
00010 ACP_MAIL::
00010 .BLKB 18
```

DISK_DESC= P.AAA
.EXTRN DATA_INIT, SCAN
.EXTRN SYSSCMKRNL
.PSECT \$CODE\$,NOWRT,2

```
0004 00000 MAIN_BAD:
0000V 52 00000000G 00 9E 00002 .WORD Save R2          : 0096
CF 00 FB 00009 MOVAB SYSSCMKRNL, R2
2F 50 E9 0000E CALLS #0, OPEN_CHANNEL : 0143
0000G CF 00 FB 00011 BLBC R0, 3$
7E D4 00016 CALLS #0, DATA_INIT : 0151
CF 9F 00018 CLRL -(SP) : 0156
0000V 62 0000V 02 FB 0001C PUSHAB GET_CCB
1E 50 E9 0001F CALLS #2, SYSSCMKRNL
CF 00 FB 00022 1$: BLBC R0, 3$
06 50 E8 00027 CALLS #0, GET_BBSMSG : 0167
0000V CF 00 FB 0002A BLBS R0, 2$
04 0002F CALLS #0, RESTORE_CHANNEL : 0170
0000V 7E D4 00030 2$: RET : 0169
CF 9F 00032 CLRL -(SP) : 0177
62 0000V 02 FB 00036 PUSHAB SET_UCB
CF 00 FB 00039 CALLS #2, SYSSCMKRNL
0000G 00 FB 00039 CALLS #0, SCAN : 0182
E2 11 0003E BRB 1$ : 0164
04 00040 3$: RET : 0184
```

; Routine Size: 65 bytes, Routine Base: \$CODE\$ + 0000

```
188 0185 1 GLOBAL ROUTINE GET_CCB =
189 0186 1
190 0187 1 !++
191 0188 1
192 0189 1 FUNCTIONAL DESCRIPTION:
193 0190 1
194 0191 1 THIS ROUTINE LOCATES THE CCB ASSOCIATED WITH A SPECIFIED
195 0192 1 CHANNEL NUMBER. THE ADDRESS OF THE UCB LONGWORD IN THE CCB
196 0193 1 IS SAVED IN CCBUCB AND THE UCB ADDRESS ITSELF IS SAVED IN
197 0194 1 OLD_UCB
198 0195 1
199 0196 1
200 0197 1 INPUT PARAMETERS:
201 0198 1
202 0199 1 !
203 0200 1 NONE
204 0201 1
205 0202 1 IMPLICIT INPUTS:
206 0203 1
207 0204 1 CHANNEL: THE CHANNEL NUMBER WHOSE CCB IS BEING LOCATED
208 0205 1
209 0206 1 OUTPUT PARAMETERS:
210 0207 1
211 0208 1 OLD UCB: THE ADDRESS OF THE SYS$DISK UCB
212 0209 1 CCBUCB: ADDRESS OF THE UCB LONGWORD IN THE SYS$DISK CCB
213 0210 1
214 0211 1 IMPLICIT OUTPUTS:
215 0212 1 NONE
216 0213 1
217 0214 1 ROUTINE VALUE:
218 0215 1
219 0216 1 IF CCB CAN BE FOUND THE SUCCESS
220 0217 1
221 0218 1
222 0219 1 SIDE EFFECTS:
223 0220 1 NONE
224 0221 1
225 0222 1 --
226 0223 1
227 0224 1
228 0225 2 BEGIN
229 0226 2
230 0227 2 LINKAGE
231 0228 2 L_VERIFYCHAN = JSB (REGISTER = 0) :
232 0229 2 GLOBAL (CCB = 1)
233 0230 2 NOPRESERVE (2, 3, 4, 5);
234 0231 2
235 0232 2 GLOBAL REGISTER
236 0233 2 CCB = 1 ; ! CCB address returned
237 0234 2
238 0235 2 LOCAL
239 0236 2 STATUS; ! status of system call
240 0237 2
241 0238 2 EXTERNAL ROUTINE
242 0239 2 IOC$VERIFYCHAN : L_VERIFYCHAN ADDRESSING_MODE (ABSOLUTE);
243 0240 2 ! exec routine to find CCB
244 0241 2
```

GETREQ
V04-000

G 14
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 8
(4)

```
: 245      0242 2 !*
: 246      0243 2 !CALL EXEC ROUTINE TO FIND CCB ADDRESS GIVEN CHANNEL NUMBER
: 247      0244 2
: 248      0245 2     STATUS = IOC$VERIFYCHAN (.CHANNEL);
: 249      0246 2
: 250      0247 2 !*
: 251      0248 2 !SAVE OLD UCB ADDRESS AND ADDRESS OF UCB ADDRESS IN CCB
: 252      0249 2
: 253      0250 2     BEGIN
: 254      0251 2     MAP CCB:REF BLOCK;
: 255      0252 2     CCBUCB=CCB[CCB$L_UCB];
: 256      0253 2     OLD_UCB=.CCB[CCB$L_UCB];
: 257      0254 2     END;
: 258      0255 2
: 259      0256 2 RETURN .STATUS
: 260      0257 1 END;                                ! end of routine GET_CCB
```

				OFFC 00000	.EXTRN	IOC\$VERIFYCHAN	
					.ENTRY	GET_CCB, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-	: 0185
						R11	:
					MOVZWL	CHANNEL, R0	: 0245
					JSB	@#IOC\$VERIFYCHAN	:
					MOVL	CCB, CCBUCB	: 0252
					MOVL	(CCB), OLD_UCB	: 0253
					RET		: 0257

; Routine Size: 24 bytes, Routine Base: \$CODE\$ + 0041

; 261 0258 1

```
: 264      0259 1 GLOBAL ROUTINE SET_UCB:NOVALUE=
: 265      0260 1
: 266      0261 1 !++
: 267      0262 1 ! FUNCTIONAL DESCRIPTION:
: 268      0263 1
: 269      0264 1 ROUTINE PLACES THE UCB ADDRESS FOUND IN ITS MAIL FROM
: 270      0265 1 F11ACP INTO THE CCB OF THE OPEN CHANNEL, THEREBY
: 271      0266 1 HAVING ACCESS TO THE SUSPECT DISK
: 272      0267 1
: 273      0268 1 FORMAL PARAMETERS:
: 274      0269 1
: 275      0270 1 NONE
: 276      0271 1
: 277      0272 1 IMPLICIT INPUTS:
: 278      0273 1
: 279      0274 1 ACP_MAIL: BUFFER CONTAINING MAIL FROM LAST READ OF F11ACP
: 280      0275 1 MAILBOX
: 281      0276 1
: 282      0277 1 IMPLICIT OUTPUTS:
: 283      0278 1
: 284      0279 1 NONE
: 285      0280 1
: 286      0281 1 ROUTINE VALUE:
: 287      0282 1 COMPLETION CODES:
: 288      0283 1
: 289      0284 1 NONE
: 290      0285 1
: 291      0286 1 SIDE EFFECTS:
: 292      0287 1
: 293      0288 1 NONE
: 294      0289 1
: 295      0290 1 !--
: 296      0291 1
: 297      0292 2 BEGIN
: 298      0293 2
: 299      0294 2
: 300      0295 2 !*
: 301      0296 2 !REPLACE THE UCB ADDRESS
: 302      0297 2
: 303      0298 2 .CCBUCB=.ACP_MAIL[BBSSL_UCB];
: 304      0299 1 END;
```

```
0000' DF 0000' CF 0000 00000
DO 00002
04 00009
```

```
.ENTRY SET_UCB, Save nothing
MOVL ACP_MAIL+8, @CCBUCB
RET
```

```
: 0259
: 0298
: 0299
```

; Routine Size: 10 bytes, Routine Base: \$CODE\$ + 0059

```
307 0300 1 ROUTINE OPEN_CHANNEL=
308 0301 1
309 0302 1 !++
310 0303 1 FUNCTIONAL DESCRIPTION:
311 0304 1
312 0305 1 THE ROUTINE OPENS CHANNEL TO THE SYSSDISK AND
313 0306 1 ACP$BADBLOCK.MBX. THE SYSSDISK CHANNEL IS USED TO
314 0307 1 OBTAIN ACCESS TO WHAT MIGHT BE AN ALLOCATED DISK.
315 0308 1 THE MBX CHANNEL IS USED TO RECEIVE MAIL FROM THE
316 0309 1 F11ACP, THAT IS THE WORK ORDERS FOR FILES TO PROCESS
317 0310 1
318 0311 1 FORMAL PARAMETERS:
319 0312 1
320 0313 1 NONE
321 0314 1
322 0315 1 IMPLICIT INPUTS:
323 0316 1
324 0317 1 NONE
325 0318 1
326 0319 1 IMPLICIT OUTPUTS:
327 0320 1
328 0321 1 CHANNEL: CHANNEL TO SYSSDISK
329 0322 1 MBX_CHANNEL: CHANNEL TO F11ACP MAILBOX
330 0323 1
331 0324 1 ROUTINE VALUE:
332 0325 1 COMPLETION CODES:
333 0326 1
334 0327 1 SHOULD EITHER $ASSIGN FAIL THEN THE FAILURE CODE IS RETURNED
335 0328 1 OTHERWISE SUCCESS IS RETURNED
336 0329 1
337 0330 1 SIDE EFFECTS:
338 0331 1
339 0332 1 NONE
340 0333 1
341 0334 1 --
342 0335 1
343 0336 2 BEGIN
344 0337 2 LOCAL
345 0338 2 STATUS;
346 0339 2 BIND
347 0340 2 MBX_DESC=UPLIT (%CHARCOUNT('ACP$BADBLOCK.MBX'),
348 0341 2 UPLIT BYTE('ACP$BADBLOCK.MBX'));
349 0342 2
350 0343 2 !+
351 0344 2 !OPEN A CHANNEL THE THE CURRENT DISK
352 0345 2
353 0346 2 IF
354 0347 2 NOT (STATUS=$ASSIGN(CHAN=CHANNEL,DEVNAM=DISK_DESC))
355 0348 2 THEN
356 0349 2 RETURN .STATUS;
357 0350 2
358 0351 2 !+
359 0352 2 !OPEN A CHANNEL TO THE MAILBOX FROM F11ACP
360 0353 2
361 0354 2 IF
362 0355 2 NOT(STATUS=$ASSIGN(CHAN=MBX_CHANNEL,DEVNAM=MBX_DESC))
363 0356 2 THEN
```

GETREQ
V04-000

J 14
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 11
(8)

```
: 364      0357 3      BEGIN
: 365      0358      $DASSGN(CHAN=.CHANNEL);
: 366      0359      RETURN .STATUS
: 367      0360      END;
: 368      0361      2
: 369      0362 2 RETURN TRUE
: 370      0363 1 END;
```

```
.PSECT $PLITS$,NOWRT,NOEXE,2
42 4D 5F 4B 43 4F 4C 42 44 41 42 24 50 43 41 00010 P.AAD: .ASCII \ACPS$BADBLOCK_MBX\
: 58 0001F
: 00000010 00020 P.AAC: .LONG 16
: 00000000 00024 .ADDRESS P.AAD
:
MBX_DESC= P.AAC
.EXTRN SYSS$ASSIGN, SYSS$DASSGN
.PSECT $CODE$,NOWRT,2
```

```
000C 00000 OPEN_CHANNEL:
53 00000000G 00 9E 00002 .WORD Save R2,R3 : 0300
: 7E 7C 00009 MOVAB SYSS$ASSIGN, R3 : 0347
: 0000' CF 9F 0000B CLRQ -(SP)
: 0000' CF 9F 0000F PUSHAB CHANNEL
63 04 FB 00013 PUSHAB DISK_DESC
52 50 D0 00016 CALLS #4, SYSS$ASSIGN
1F 52 E9 00019 MOVL R0, STATUS
: 7E 7C 0001C BLBC STATUS, 1$
: 0000' CF 9F 0001E CLRQ -(SP) : 0355
: 0000' CF 9F 00022 PUSHAB MBX_CHANNEL
63 04 FB 00026 PUSHAB MBX_DESC
52 50 D0 00029 CALLS #4, SYSS$ASSIGN
10 52 E8 0002C MOVL R0, STATUS
7E 0000' CF 3C 0002F BLBS STATUS, 2$
00000000G 00 01 FB 00034 MOVZWL CHANNEL, -(SP) : 0358
50 52 D0 0003B 1$: CALLS #1, SYSS$DASSGN : 0359
: 04 0003E MOVL STATUS, R0
50 01 D0 0003F 2$: RET : 0362
: 04 00042 RET : 0363
```

; Routine Size: 67 bytes, Routine Base: \$CODE\$ + 0063

```
372 0364 1 ROUTINE GET_BBSMSG=
373 0365 1
374 0366 1 ++
375 0367 1 FUNCTIONAL DESCRIPTION:
376 0368 1
377 0369 1 THIS ROUTINE ATTEMPTS TO READ MAIL FROM THE
378 0370 1 MAILBOX LINKING THE F11ACP TO TH BADBLOCK UTILITY
379 0371 1 IF THE READ IS SUCCESSFUL THE MESSAGE TYPE AND LENGTH
380 0372 1 OF THE MAIL IS CHECKED, IF EITHER IS WRONG THE MESSAGE
381 0373 1 IS DISGARDED AND ANOTHER IS READ. IF A READ FAILS,
382 0374 1 THE FAILURE STATUS IS RETURNED BY THE ROUTINE.
383 0375 1
384 0376 1 FORMAL PARAMETERS:
385 0377 1
386 0378 1 NONE
387 0379 1
388 0380 1 IMPLICIT INPUTS:
389 0381 1
390 0382 1 NONE
391 0383 1
392 0384 1 IMPLICIT OUTPUTS:
393 0385 1
394 0386 1 ACP_MAIL: MAIL IS READ INTO THIS BLOCK
395 0387 1
396 0388 1 ROUTINE VALUE:
397 0389 1 COMPLETION CODES:
398 0390 1
399 0391 1 IF THE MAILBOX READ FAILS THE FAILURE IS RETURNED
400 0392 1 OTHERWISE TRUE.
401 0393 1
402 0394 1 SIDE EFFECTS:
403 0395 1
404 0396 1 NONE
405 0397 1
406 0398 1 --
407 0399 1
408 0400 2 BEGIN
409 0401 2 LOCAL
410 0402 2 STATUS;
411 0403 2
412 0404 2 OWN
413 0405 2 MBX_IOSB:VECTOR[4,WORD];
414 0406 2
415 0407 2 ++
416 0408 2 !LOOP UNTIL AN ERROR OF EOF
417 0409 2
418 0410 2 WHILE TRUE DO
419 0411 2
420 0412 2 BEGIN
421 0413 3 ++
422 0414 3 !READ FROM THE MAILBOX AND CHECK THE SYS SERV STATUS
423 0415 3
424 0416 3
425 P 0417 4 IF NOT (STATUS=$QIOW(CHAN=.MBX CHANNEL,
426 P 0418 4 FUNC=IOS_READVBLK+IOSM_NOW,
427 P 0419 4 IOSB=MBX_IOSB,
428 P 0420 4 P1=ACP_MAIL,
```

GETREQ
V04-000

L 14
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 13
(9)

```
: 429      0421  4
: 430      0422  4
: 431      0423  4
: 432      0424  4
: 433      0425  4
: 434      0426  4
: 435      0427  4
: 436      0428  4
: 437      0429  4
: 438      0430  4
: 439      0431  4
: 440      0432  4
: 441      0433  4
: 442      0434  4
: 443      0435  4
: 444      0436  4
: 445      0437  4
: 446      0438  4
: 447      0439  4
: 448      0440  4
: 449      0441  4
: 450      0442  4
: 451      0443  4
: 452      0444  4
: 453      0445  1 END;
```

```
                P2=BB$$C_LENGTH))
THEN
    RETURN .STATUS;
!++
!CHECK IO STATUS
!IF-
    NOT .MBX_IOSB[0]
THEN
    RETURN .MBX_IOSB[0];
!+
!CHECK FOR PROPER MESSAGE TYPE AND LENGTH
IF
    (.ACP_MAIL[BB$$B_MSGTYPE] EQL MSG$_SCANBAD)
    AND
    (.MBX_IOSB[1] EQL BB$$C_LENGTH)
THEN
    RETURN TRUE;
END
```

.PSECT \$OWN\$,NOEXE,2

00000 MBX_IOSB:

.BLKB 8

.EXTRN SY\$\$QIOW

.PSECT \$CODE\$,NOWRT,2

0004 00000 GET_BBSMSG:

52	0000'	CF 9E 00002	1\$:	WORD	Save R2	: 0364
		7E 7C 00007		MOVAB	MBX_IOSB, R2	: 0421
		7E 7C 00009		CLRQ	-(SP)	
		12 DD 0000B		CLRQ	-(SP)	
	0000'	CF 9F 0000D		PUSHL	#18	
		7E 7C 00011		PUSHAB	ACP_MAIL	
		52 DD 00013		CLRQ	-(SP)	
7E	71	8F 9A 00015		PUSHL	R2	
7E	0000'	CF 3C 00019		MOVZBL	#113, -(SP)	
		7E D4 0001E		MOVZWL	MBX_CHANNEL, -(SP)	
00000000G	00	0C FB 00020		CLRL	-(SP)	
	17	50 E9 00027		CALLS	#12, SY\$\$QIOW	
	04	62 E8 0002A		BLBC	STATUS, 3\$	: 0430
	50	62 3C 0002D		BLBS	MBX_IOSB, 2\$	: 0432
		04 00030		MOVZWL	MBX_IOSB, R0	
	28	CF 91 00031	2\$:	RET		
		CF 12 00036		CMPB	ACP_MAIL, #40	: 0438
	12	A2 B1 00038		BNEQ	1\$	
				CMPW	MBX_IOSB+2, #18	: 0440

GETREQ
V04-000

M 14
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 14
(9)

50

C9 12 0003C BNEQ 1\$
01 D0 0003E MOVL #1, R0
04 00041 3\$: RET

: 0442
: 0445

; Routine Size: 66 bytes, Routine Base: \$CODE\$ + 00A6

```

: 455      0446 1 GLOBAL ROUTINE RESTORE_CHANNEL:NOVALUE=
: 456      0447 1
: 457      0448 1 ++
: 458      0449 1 FUNCTIONAL DESCRIPTION:
: 459      0450 1
: 460      0451 1 ROUTINE RESTORES THE UCB ADDRESS FO THE SYSSDISK INTO
: 461      0452 1 THE ACCESSED CCB AND $DASSGN'S BOTH OPEN CHANNELS
: 462      0453 1
: 463      0454 1 FORMAL PARAMETERS:
: 464      0455 1
: 465      0456 1 NONE
: 466      0457 1
: 467      0458 1 IMPLICIT INPUTS:
: 468      0459 1
: 469      0460 1 CHANNEL: SYSSDISK CHANNEL
: 470      0461 1 MBX_CHANNEL: CHANNEL NUMBER TO F11ACP
: 471      0462 1
: 472      0463 1 IMPLICIT OUTPUTS:
: 473      0464 1
: 474      0465 1 NONE
: 475      0466 1
: 476      0467 1 ROUTINE VALUE:
: 477      0468 1 COMPLETION CODES:
: 478      0469 1
: 479      0470 1 NONE
: 480      0471 1
: 481      0472 1 SIDE EFFECTS:
: 482      0473 1
: 483      0474 1 NONE
: 484      0475 1
: 485      0476 1 --
: 486      0477 1
: 487      0478 2 BEGIN
: 488      0479 2 ++
: 489      0480 2 PUT OLD UCB ADDRESS BACK INTO CCB
: 490      0481 2 --
: 491      0482 2
: 492      0483 2 ACP_MAIL[BBSSL_UCB]=.OLD_UCB;
: 493      0484 2 $CMKRNL(ROUTIN=SET_UCB);
: 494      0485 2
: 495      0486 2 ++
: 496      0487 2 DE-ASSIGN BOTH ACTIVE CHANNELS
: 497      0488 2
: 498      0489 2 -- $DASSGN(CHAN=.CHANNEL);
: 499      0490 2 $DASSGN(CHAN=.MBX_CHANNEL); !CLEAR MAILBOX CHANNEL
: 500      0491 1 END;
```

```

0000' 52 00000000G 00 0004 00000
CF 0000' CF 9E 00002
7E D0 00009
CF 9F 00010
00000000G 00 FF5B CF 02 FB 00012
```

```

.ENTRY RESTORE_CHANNEL, Save R2
MOVAB SYSSDASSGN, R2
MOVL OLD_UCB, ACP_MAIL+8
CLRL -(SP)
PUSHAB SET_UCB
CALLS #2, SYSSCMKRNL
```

```

: 0446
: 0483
: 0484
:
```

B 15
15-Sep-1984 23:36:36 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 11:54:33 [BADBLK.SRC]GETREQ.B32;1

7E	0000'	CF	3C	0001D
62		01	FB	00022
7E	0000'	CF	3C	00025
62		01	FB	0002A
			04	0002D

```
MOVZWL CHANNEL, -(SP)
CALLS #1, SYS$DASSGN
MOVZWL MBX_CHANNEL, -(SP)
CALLS #1, SYS$DASSGN
RET
```

: 0489
: 0490
: 0491

; Routine Size: 46 bytes, Routine Base: \$CODE\$ + 00E8

GETREQ
V04-000

C 15
15-Sep-1984 23:36:36
14-Sep-1984 11:54:33

VAX-11 Bliss-32 V4.0-742
[BADBLK.SRC]GETREQ.B32;1

Page 17
(12)

: 503 0492 1 END
: 504 0493 0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
\$GLOBALS	34 NOVEC, WRT, RD	,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$PLITS	40 NOVEC,NOWRT, RD	,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$CODES	278 NOVEC,NOWRT, RD	,EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$OWNS	8 NOVEC, WRT, RD	,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	13	0	1000	00:01.9

COMMAND QUALIFIERS

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

: Size: 278 code + 82 data bytes
: Run Time: 00:09.2
: Elapsed Time: 00:15.5
: Lines/CPU Min: 3215
: Lexemes/CPU-Min: 14060
: Memory Used: 71 pages
: Compilation Complete

0018 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

