

[illegible]


```
1 0001 0 MODULE COB$$RESTVA(
2 0002 0 IDENT = '1-005'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1
7 0007 1
8 0008 1
9 0009 1
10 0010 1
11 0011 1
12 0012 1
13 0013 1
14 0014 1
15 0015 1
16 0016 1
17 0017 1
18 0018 1
19 0019 1
20 0020 1
21 0021 1
22 0022 1
23 0023 1
24 0024 1
25 0025 1
26 0026 1
27 0027 1
28 0028 1
29 0029 1
30 0030 1
31 0031 1
32 0032 1
33 0033 1
34 0034 1
35 0035 1
36 0036 1
37 0037 1
38 0038 1
39 0039 1
40 0040 1
41 0041 1
42 0042 1
43 0043 1
44 0044 1
45 0045 1
46 0046 1
47 0047 1
48 0048 1
49 0049 1
50 0050 1
51 0051 1
52 0052 1
53 0053 1
54 0054 1

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

++
FACILITY: COBOL SUPPORT

ABSTRACT

ENVIRONMENT: Vax-11 User Mode

AUTHOR: MLJ , CREATION DATE: 05-MAY-1979

MODIFIED BY:
1-001 - Original. MLJ 05-MAY-1979
1-002 - Added boilerplate and comments. RKR 24-AUG-1979
1-003 - Use RTLLIB for LIBRARY and define psect via macros. RKR 18-SEPT-79
1-004 - Rename to COBRESTVA from former SYSRESTVA RKR 24-SEPT-79
(Entry point becomes COB$$RESTVA)
1-005 - Make module name match entry point, cosmetic changes.
RKR 18-OCT-79

--

!<BLF/PAGE>
```

```
56 0055 1 |
57 0056 1 | SWITCHES
58 0057 1 |
59 0058 1 |
60 0059 1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
61 0060 1 |
62 0061 1 |
63 0062 1 | LINKAGES
64 0063 1 |
65 0064 1 | NONE
66 0065 1 |
67 0066 1 | TABLE OF CONTENTS:
68 0067 1 |
69 0068 1 FORWARD ROUTINE
70 0069 1 |
71 0070 1 COB$$RESTVA ; ! Restore virtual address from image file
72 0071 1 |
73 0072 1 | INCLUDE FILES
74 0073 1 |
75 0074 1 REQUIRE 'RTLIN:RTLPSECT'; ! Macros for psect declarations
76 0169 1 LIBRARY 'RTLLIB'; ! Image header structures and sizes
77 0170 1 |
78 0171 1 |
79 0172 1 | MACROS
80 0173 1 |
81 0174 1 | NONE
82 0175 1 |
83 0176 1 | EQUATED SYMBOLS
84 0177 1 |
85 0178 1 | NONE
86 0179 1 |
87 0180 1 | PSECT DECLARATIONS:
88 0181 1 |
89 0182 1 DECLARE_PSECTS (COB) ; ! Psects for COB$ facility
```



```

91 0183 1 GLOBAL ROUTINE COB$$RESTVA (
92 0184 1
93 0185 1             INADR,           ! Input addr.'s to be restored
94 0186 1             CHAN,           ! Channel number
95 0187 1             EFN            ! Event flag number
96 0188 1
97 0189 1             )=
98 0190 1
99 0191 1 ++      FUNCTIONAL DESCRIPTION
100 0192 1
101 0193 1
102 0194 1 FORMAL PARAMETERS:
103 0195 1
104 0196 1
105 0197 1 INADR.ra.r
106 0198 1
107 0199 1         address of a 2-longword array containing the starting and ending
108 0200 1         virtual addresses in the process's virtual address space of the
109 0201 1         area to be restored from the image file.
110 0202 1
111 0203 1 CHAN.rl.v
112 0204 1
113 0205 1         number of the channel on which the image file has been accessed.
114 0206 1         The file must have been accessed with an RMS $OPEN macro; the
115 0207 1         file options parameter (FOP) in the FAB must indicate a user
116 0208 1         file open (UFO) keyword). The access mode at which the channel
117 0209 1         was opened must be the same or less privileged than the access
118 0210 1         mode of the caller.
119 0211 1
120 0212 1 EFN.rl.v
121 0213 1
122 0214 1         Event flag number to use for I/O on CHAN.
123 0215 1
124 0216 1
125 0217 1 IMPLICIT INPUTS:
126 0218 1
127 0219 1         NONE
128 0220 1
129 0221 1 IMPLICIT OUTPUTS:
130 0222 1
131 0223 1         NONE
132 0224 1
133 0225 1 ROUTINE VALUE:
134 0226 1 COMPLETION CODES:
135 0227 1
136 0228 1         $$$_BADIMGHDR      ! Bad image header read
137 0229 1         $$$_INSFARG       ! Insufficient arguments to this routine
138 0230 1         $$$_NORMAL        ! Normal successful completion
139 0231 1         $$$_PAGOWNVIO     ! Page ownership violation
140 0232 1
141 0233 1         If read error occurs, RMS error status is returned.
142 0234 1
143 0235 1
144 0236 1
145 0237 1 SIDE EFFECTS:
146 0238 1
147 0239 1         NONE
```

```
148 0240 1
149 0241 1
150 0242 1
151 0243 1
152 0244 1
153 0245 1
154 0246 1
155 0247 1
156 0248 1
157 0249 1
158 0250 1
159 0251 1
160 0252 1
161 0253 1
162 0254 1
163 0255 1
164 0256 1
165 0257 1
166 0258 1
167 0259 1
168 0260 1
169 0261 1
170 0262 1
171 0263 1
172 0264 1
173 0265 1
174 0266 1
175 0267 1
176 0268 1
177 0269 1
178 0270 1
179 0271 1
180 0272 1
181 0273 1
182 0274 1
183 0275 1
184 0276 1
185 0277 1
186 0278 1
187 0279 1
188 0280 1
189 0281 1
190 0282 1
191 0283 1
192 0284 1
193 0285 1
194 0286 1
195 0287 1
196 0288 1
197 0289 1
198 0290 1
199 0291 1
200 0292 1
201 0293 1
202 0294 1
203 0295 1
204 0296 1

--
BEGIN
MAP
    INADR:          REF VECTOR;

LOCAL
    STATUS:          BLOCK[512, BYTE]
    IHDBUF:          VECTOR[512, BYTE],
    IMGBUF:          VECTOR[2],
    IOSB:            REF BLOCK[, BYTE],
    ISD:             Pointer to image section descriptor
    IHDVBN,          Current VBN of image header
    MAXIHDVBN:       Maximum VBN of image header
BUILTIN
    ACTUALCOUNT:

!+
! Make sure enough arguments are present.
!-
IF ACTUALCOUNT() LSS 3 THEN RETURN SS$_INSFARG;

!+
! Read the first (usually, only) block of the image header.
!-
IHDVBN = 1;
STATUS = $QIOW(
    EFN=.EFN,
    CHAN=.CHAN,
    FUNC=IOS_READVBLK,
    IOSB=IOSB,
    P1=IHDBUF,
    P2=512,
    P3=1);
IF NOT .STATUS THEN RETURN .STATUS;
IF NOT .IOSB THEN RETURN .IOSB<0,16>;

!+
! Save the number of blocks in the image header, make sure that the
! SIZE field is reasonable, and point to the first ISD.
!-
MAXIHDVBN = .IHDBUF[IHDSB_HDRBLKCNT];
IF .IHDBUF[IHDSW_SIZE] GTRU 510 THEN RETURN SS$_BADIMGHDR;
ISD = IHDBUF + .IHDBUF[IHDSW_SIZE];

!+
! Loop over the requested virtual pages.
```



```

        ISD = IHDBUF;
    END;
END;

!+
! If the current ISD is for a virtual page that follows the
! requested one, this indicates that no ISD describes the
! requested one, since the preceding ISD described pages that
! precede the requested one.
-
IF .VPN LSS .ISD[ISD$V_VPN]
THEN
    RETURN SSS_PAGOWNVIO;

!+
! Make sure that the ISD does not describe a global page. We
! do not support them.
-
IF .ISD[ISD$V_GBL]
THEN
    RETURN SSS_PAGOWNVIO;

!+
! Compute the virtual address and length of the area within the
! current page that will be restored.
-
SVA = MAXA(.VPN*9, .INADR[0]);           ! Find base address
LEN = MINA(.VPN*9+511, .INADR[1]) - .SVA + 1; ! Find length

!+
! Do the restoration. If the page is not writable, we will not
! restore it, since it should not have changed (unless by
! debugger). If it is demand zero, just fill the region with
! zero. Otherwise, read the appropriate block from the image,
! and copy the region over.
-
IF .ISD[ISD$V_WRT]
THEN
    IF .ISD[ISD$V_DZRO]
    THEN
        CH$FILL(0, .LEN, .SVA)
    ELSE
        BEGIN
            STATUS = $QIOW(
                EFN=.EFN,
                CHAN=.CHAN,
                FUNC=IOS_READVBLK,
                IOSB=IOSB,
                P1=IMGBUF,
                P2=512,
                P3=.ISD[ISD$L_VBN]+.VPN-.ISD[ISD$V_VPN]);
            IF NOT .STATUS THEN RETURN .STATUS;
            IF NOT .IOSB THEN RETURN .IOSB<0,16>;
        END
    END
END

```


COB\$\$RESTVA
1-005

K 2
16-Sep-1984 00:14:08
14-Sep-1984 12:10:55

VAX-11 Bliss-32 V4.0-742
[COBRTL.SRC]COBRESTVA.B32;1

Page 7
(3)

```

: 319      0411 4
: 320      0412 4
: 321      0413 4
: 322      0414 4
: 323      0415 4
: 324      0416 4
: 325      0417 1

END:

SS$ NORMAL
END:
```

```
CH$MOVE(.LEN, IMGBUF[.SVA-(.VPN^9)], .SVA);
END;
```

				OFFC	00000		
		5E	FBE8	CE	9E	00002	
		03		6C	91	00007	
				06	1E	0000A	
		50	0114	8F	3C	0000C	
					04	00011	
	08	AE		01	D0	00012	1\$:
				7E	7C	00016	
		7E		01	7D	00018	
		7E	0200	8F	3C	0001B	
			FE00	CD	9F	00020	
				7E	7C	00024	
			30	AE	9F	00026	
				31	DD	00029	
			08	AC	DD	0002B	
			0C	AC	DD	0002E	
		00000000G	00	0C	FB	00031	
			6E	50	D0	00038	
			03	6E	E8	0003B	
				012B	31	0003E	
		03	10	AE	E8	00041	2\$:
				012C	31	00045	
	0C	AE	FE10	CD	9A	00048	3\$:
	01FE	8F	FE00	CD	B1	0004E	
				53	1A	00055	
		50	FE00	CD	9E	00057	
		56	FE00	CD	3C	0005C	
		56		50	C0	00061	
		5A	04	AC	D0	00064	
		17		01	EF	00068	
		17		09	EF	0006F	
				010E	31	00074	
		15		00	EF	00077	4\$:
		51	02	A6	3C	0007D	
		50		51	C0	00081	
		50		58	D1	00084	
				64	19	00087	
		50		66	3C	00089	
		56		50	C0	0008C	
		50	FE	AD	9E	0008F	

.TITLE	COB\$\$RESTVA	
.IDENT	\1-005\	
.EXTRN	SYSSQIOW	
.PSECT	_COB\$CODE,NOWRT, SHR, PIC,2	
.ENTRY	COB\$\$RESTVA, Save R2,R3,R4,R5,R6,R7,R8,R9,-	0183
MOVAB	R10,R11	
CMPB	-1048(SP), SP	
BGEQU	(AP), #3	0264
MOVZWL	1\$	
RET	#276, R0	
MOVL	#1, IHDVBN	0271
CLRQ	-(SP)	0279
MOVQ	#1, -(SP)	
MOVZWL	#512, -(SP)	
PUSHAB	IHDBUF	
CLRQ	-(SP)	
PUSHAB	IOSB	
PUSHL	#49	
PUSHL	CHAN	
PUSHL	EFN	
CALLS	#12, SYSSQIOW	
MOVL	R0, STATUS	
BLBS	STATUS, 2\$	0280
BRW	15\$	
BLBS	IOSB, 3\$	0281
BRW	17\$	
MOVZBL	IHDBUF+16, MAXIHDVBN	0289
CMPW	IHDBUF, #510	0290
BGTRU	5\$	
MOVAB	IHDBUF, R0	0291
MOVZWL	IHDBUF, ISD	
ADDL2	R0, ISD	
MOVL	INADR, R10	0298
EXTZV	#1, #23, 5(R10), 4(SP)	
EXTZV	#9, #23, (R10), VPN	
BRW	20\$	
EXTZV	#0, #21, 4(ISD), R0	0313
MOVZWL	2(ISD), R1	
ADDL2	R1, R0	
CMPB	VPN, R0	
BLSS	9\$	
MOVZWL	(ISD), R0	0320
ADDL2	R0, ISD	
MOVAB	IHDBUF+510, R0	0321

		50		56	D1	00093	CMPL	ISD, R0		
				12	1A	00096	BGTRU	5\$		
				66	B5	00098	TSTW	(ISD)		0329
				5D	13	0009A	BEQL	10\$		
	FFFF	8F		66	B1	0009C	CMPL	(ISD), #65535		0339
				D4	12	000A1	BNEQ	4\$		
	OC	AE	08	AE	D1	000A3	CMPL	IHDVBN, MAXIHDVBN		0342
				05	12	000A8	BNEQ	6\$		
		50	44	8F	9A	000AA	MOVZBL	#68, R0		
					04	000AE	RET			
			08	AE	D6	000AF	INCL	IHDVBN		0343
				7E	7C	000B2	CLRQ	-(SP)		0351
				7E	D4	000B4	CLRL	-(SP)		
			14	AE	DD	000B6	PUSHL	IHDVBN		
		7E	0200	8F	3C	000B9	MOVZWL	#512, -(SP)		
			FE00	CD	9F	000BE	PUSHAB	IHDBUF		
				7E	7C	000C2	CLRQ	-(SP)		
			30	AE	9F	000C4	PUSHAB	IOSB		
				31	DD	000C7	PUSHL	#49		
			08	AC	DD	000C9	PUSHL	CHAN		
			OC	AC	DD	000CC	PUSHL	EFN		
	00000000G	00		OC	FB	000CF	CALLS	#12, SYSSQIOW		
		6E		50	D0	000D6	MOVL	R0, STATUS		
		03		6E	E8	000D9	BLBS	STATUS, 7\$		0352
				008D	31	000DC	BRW	15\$		
		03	10	AE	E8	000DF	BLBS	IOSB, 8\$		0353
				008E	31	000E3	BRW	17\$		
		56	FE00	CD	9E	000E6	MOVAB	IHDBUF, ISD		0354
				8A	11	000EB	BRB	4\$		0313
58	04	A6	15	00	ED	000ED	CMPZV	#0, #21, 4(ISD), VPN		0365
				04	14	000F3	BGTR	10\$		
		06	08	A6	E9	000F5	BLBC	8(ISD), 11\$		0374
		50	01EC	8F	3C	000F9	MOVZWL	#492, R0		0376
					04	000FE	RET			
		58		09	78	000FF	ASHL	#9, VPN, R11		0383
		59		5B	D0	00103	MOVL	R11, R9		
		6A		59	D1	00106	CMPL	R9, (R10)		
				03	1E	00109	BGEQU	12\$		
		59		6A	D0	0010B	MOVL	(R10), R9		
		57		59	D0	0010E	MOVL	R9, SVA		
		59	01FF	CB	9E	00111	MOVAB	511(R11), R9		0384
	04	AA		59	D1	00116	CMPL	R9, 4(R10)		
				04	1B	0011A	BLEQU	13\$		
		59	04	AA	D0	0011C	MOVL	4(R10), R9		
		59		57	C2	00120	SUBL2	SVA, R9		
				59	D6	00123	INCL	LEN		
		59		03	E1	00125	BBC	#3, 8(ISD), 19\$		0394
		08	08	02	E1	0012A	BBC	#2, 8(ISD), 14\$		0396
59		00		00	2C	0012F	MOVCS	#0, (SP), #0, LEN, (SVA)		0398
				67		00134				
				4C	11	00135	BRB	19\$		
				7E	7C	00137	CLRQ	-(SP)		0408
				7E	D4	00139	CLRL	-(SP)		
		58	OC	A6	C1	0013B	ADDL3	12(ISD), VPN, R0		
51	04	50		00	EF	00140	EXTZV	#0, #21, 4(ISD), R1		
		A6		51	C3	00146	SUBL3	R1, R0, -(SP)		
		7E		8F	3C	0014A	MOVZWL	#512, -(SP)		

		2C	AE	9F	0014F	PUSHAB	IMGBUF		
			7E	7C	00152	CLRQ	-(SP)		
		30	AE	9F	00154	PUSHAB	IOSB		
			31	DD	00157	PUSHL	#49		
		08	AC	DD	00159	PUSHL	CHAN		
		0C	AC	DD	0015C	PUSHL	EFN		
	00000000G		0C	FB	0015F	CALLS	#12, SYSSQIOW		
			50	D0	00166	MOVL	R0, STATUS		
			6E	E8	00169	BLBS	STATUS, 16\$		0409
			6E	D0	0016C	MOVL	STATUS, R0		
				04	0016F	RET			
		10	AE	E8	00170	BLBS	IOSB, 18\$		0410
		10	AE	3C	00174	MOVZWL	IOSB, R0		
				04	00178	RET			
50			5B	C3	00179	SUBL3	R11, SVA, R0		0411
67			59	28	0017D	MOVC3	LEN, IMGBUF[R0], (SVA)		
	18	AE40	58	D6	00183	INCL	VPN		0298
			58	D1	00185	CMPL	VPN, 4(SP)		
	04	AE	03	1A	00189	BGTRU	21\$		
			FEE9	31	0018B	BRW	4\$		
			01	D0	0018E	MOVL	#1, R0		0417
	50		04	00191	RET				

; Routine Size: 402 bytes, Routine Base: _COB\$CODE + 0000

COB\$\$\$RESTVA
1-005

N 2
16-Sep-1984 00:14:08
14-Sep-1984 12:10:55

VAX-11 Bliss-32 V4.0-742
[COBRTL.SRC]COBRESTVA.B32;1

Page 10
(4)

: 327
: 328
0418 1 END
0419 0 ELUDOM

PSECT SUMMARY

Name Bytes Attributes
_COB\$CODE 402 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	17	0	1000	00:01.4

COMMAND QUALIFIERS

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

: Size: 402 code + 0 data bytes
: Run Time: 00:07.8
: Elapsed Time: 00:34.4
: Lines/CPU Min: 3231
: Lexemes/CPU-Min: 22465
: Memory Used: 148 pages
: Compilation Complete

0064 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY