

LL	IIIIII	BBBBBBBB	SSSSSSSS	TTTTTTTT	AAAAAA	TTTTTTTT	VV	VV	MM	MM	
LL	IIIIII	BBBBBBBB	SSSSSSSS	TTTTTTTT	AAAAAA	TTTTTTTT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MMMM	MMMM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MMMM	MMMM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BBBBBBBB	SSSSSS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BBBBBBBB	SSSSSS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AAAAAAAA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AAAAAAAA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LL	II	BB	SS	TT	AA	TT	VV	VV	MM	MM	
LLLLLLLLLL	IIIIII	BBBBBBBB	SSSSSSSS	TT	AA	TT	VV	VV	MM	MM	
LLLLLLLLLL	IIIIII	BBBBBBBB	SSSSSSSS	TT	AA	TT	VV	VV	MM	MM	

```

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
LLLLLLLLLLLL IIIIIII SSSSSSSS
LLLLLLLLLLLL IIIIIII SSSSSSSS

```



```
1 0001 0 MODULE LIB$STAT_VM ( ! Gather statistics from LIB$VM
2 0002 0 IDENT = '1-003' ! File: LIB$STATVM.B32
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: General Library
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module returns LIB$GET_VM statistics to the user.
36 0036 1 These are the number of calls to GET and FREE, and the
37 0037 1 the number of bytes gotten but not yet freed.
38 0038 1
39 0039 1 ENVIRONMENT: User mode, AST reentrant.
40 0040 1
41 0041 1 AUTHOR: John Sauter, CREATION DATE: 26-OCT-1979
42 0042 1
43 0043 1 MODIFIED BY:
44 0044 1
45 0045 1 1-001 - Original. JBS 28-OCT-1979
46 0046 1 1-002 - Don't use LIB$SYS_FAO, since it isn't in the sharable library.
47 0047 1 JBS 29-OCT-1979
48 0048 1 1-003 - Correct a comment. JBS 30-NOV-1979
49 0049 1 --
50 0050 1
51 0051 1 !<BLF/PAGE>
```

```
53 0052 1 |
54 0053 1 | SWITCHES:
55 0054 1 |
56 0055 1 |
57 0056 1 | SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
58 0057 1 |
59 0058 1 |
60 0059 1 | LINKAGES:
61 0060 1 |
62 0061 1 |     NONE
63 0062 1 |
64 0063 1 | TABLE OF CONTENTS:
65 0064 1 |
66 0065 1 |
67 0066 1 | FORWARD ROUTINE
68 0067 1 |     LIB$SHOW_VM,           ! Show VM statistics as text
69 0068 1 |     LIB$STAT_VM;          ! Show VM statistics in binary
70 0069 1 |
71 0070 1 |
72 0071 1 | INCLUDE FILES:
73 0072 1 |
74 0073 1 |
75 0074 1 | REQUIRE 'RTLIN:RTLPSECT'; ! DECLARE_PSECTS macro
76 0169 1 |
77 0170 1 | LIBRARY 'RTLSTARLE';      ! System symbols
78 0171 1 |
79 0172 1 |
80 0173 1 | EQUATED SYMBOLS:
81 0174 1 |
82 0175 1 |
83 0176 1 | LITERAL
84 0177 1 |     K_GET = 1,             ! Number of calls to LIB$GET_VM
85 0178 1 |     K_FREE = 2,           ! Number of calls to LIB$FREE_VM
86 0179 1 |     K_IN_USE = 3,         ! Number of bytes in use
87 0180 1 |     MAX_CODE = 3,         ! Max codes for reporting
88 0181 1 |     K_MAX_STR_LEN = 255;  ! Max FAO string length
89 0182 1 |
90 0183 1 |
91 0184 1 | PSECTS
92 0185 1 |
93 0186 1 | DECLARE_PSECTS (LIB);     ! Define psecks
94 0187 1 |
95 0188 1 | OWN STORAGE:
96 0189 1 |
97 0190 1 |     NONE
98 0191 1 |
99 0192 1 | EXTERNAL REFERENCES:
100 0193 1 |
101 0194 1 |
102 0195 1 | EXTERNAL ROUTINE
103 0196 1 |     LIB$PUT_OUTPUT;       ! Sends string to SYSS$OUTPUT
104 0197 1 |
105 0198 1 |
106 0199 1 | !+ The following are the statistics cells maintained by LIB$GET_VM.
107 0200 1 | !-
108 0201 1 |
109 0202 1 | EXTERNAL
```



```
: 110      0203 1      LIB$GL_GETVM_C,      ! Number of calls to LIB$GET_VM
: 111      0204 1      LIB$GL_FREVM_C,      ! Number of calls to LIB$FREE_VM
: 112      0205 1      LIB$GL_VMINUSE;      ! Number of bytes allocated but not freed
: 113      0206 1
: 114      0207 1      !+
: 115      0208 1      !- The following are the error codes used in this module.
: 116      0209 1
: 117      0210 1
: 118      0211 1      EXTERNAL LITERAL
: 119      0212 1      LIB$_INVARG;          ! Invalid argument error condition
: 120      0213 1
```

```
122 0214 1 GLOBAL ROUTINE LIB$SHOW_VM (          | Show VM statistics
123 0215 1     CODE,                          | Code for what to show
124 0216 1     ACTION,                      | Action routine to print text
125 0217 1     USERARG                      | User argument for action routine
126 0218 1 ) =
127 0219 1
128 0220 1 ++
129 0221 1 FUNCTIONAL DESCRIPTION:
130 0222 1
131 0223 1     LIB$SHOW_VM is used to obtain the accumulated statistics
132 0224 1     from calls to LIB$GET_VM and LIB$SHOW_VM. In the default
133 0225 1     mode, with neither CODE nor ACTION specified in the call, the
134 0226 1     routine will output to SYS$OUTPUT a line giving the following
135 0227 1     three items of information:
136 0228 1
137 0229 1     999 calls to LIB$GET_VM, 999 calls to LIB$FREE_VM, 999 bytes still allocated
138 0230 1
139 0231 1     Optionally, only one of the three statistics can be output to
140 0232 1     SYS$OUTPUT and/or the line of information can be passed to a
141 0233 1     user-specified "action routine", for processing different from
142 0234 1     the default.
143 0235 1
144 0236 1 CALLING SEQUENCE:
145 0237 1
146 0238 1     status.wlc.v = LIB$SHOW_VM ([[[code.rl.r],
147 0239 1                          action.flc.rp], userarg.xx.x])
148 0240 1
149 0241 1 FORMAL PARAMETERS:
150 0242 1
151 0243 1     code - (Optional) Specifies the particular statistic desired.
152 0244 1           If given, it must be one of the following values:
153 0245 1             1 = Number of calls to LIB$GET_VM
154 0246 1             2 = Number of calls to LIB$FREE_VM
155 0247 1             3 = Number of bytes allocated by LIB$GET_VM but not
156 0248 1                 yet deallocated by LIB$FREE_VM.
157 0249 1           If omitted or zero, all three statistics will be returned
158 0250 1           on one line.
159 0251 1
160 0252 1     action
161 0253 1     userarg - (Optional) These two parameters allow the user to direct
162 0254 1           the output of LIB$SHOW_VM somewhere other than SYS$OUTPUT.
163 0255 1           If "action" is given, it is the address of a function
164 0256 1           procedure to call. The arguments to this function are
165 0257 1           described below. The function should return either
166 0258 1           a success or failure condition value, which will be
167 0259 1           returned as the value of LIB$SHOW_VM.
168 0260 1
169 0261 1 CALLING SEQUENCE OF ACTION ROUTINE:
170 0262 1
171 0263 1     status.wlc.v = (action) (out_string.rt.dx [, userarg.rl.v])
172 0264 1
173 0265 1     out_string - A descriptor of a string
174 0266 1                 containing the statistics desired. It
175 0267 1                 is formatted exactly as it would be if
176 0268 1                 output to SYS$OUTPUT. The leading character
177 0269 1                 is blank.
178 0270 1
```



```
179      userarg      - If "userarg" is passed to LIB$SHOW_VM,  
180                  it is passed directly on to the action  
181                  routine. The 32-bit value in LIB$SHOW_VM's  
182                  argument list is passed to the action  
183                  routine without interpretation.  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235
```

```
0271 1  
0272 1  
0273 1  
0274 1  
0275 1  
0276 1  
0277 1  
0278 1  
0279 1  
0280 1  
0281 1  
0282 1  
0283 1  
0284 1  
0285 1  
0286 1  
0287 1  
0288 1  
0289 1  
0290 1  
0291 1  
0292 1  
0293 1  
0294 1  
0295 1  
0296 1  
0297 1  
0298 1  
0299 2  
0300 2  
0301 2  
0302 2  
0303 2  
0304 2  
0305 2  
0306 2  
0307 2  
0308 2  
0309 2  
0310 2  
0311 2  
0312 2  
0313 2  
0314 2  
0315 2  
0316 2  
0317 2  
0318 2  
0319 2  
0320 2  
0321 2  
0322 2  
0323 2  
0324 2  
0325 2  
0326 2  
0327 2
```

```
IMPLICIT INPUTS:  
    The statistics cells maintained by LIB$VM.  
  
IMPLICIT OUTPUTS:  
    NONE  
  
COMPLETION CODES:  
    SS$ NORMAL      - Successful completion.  
    LIB$_INVARG     - Invalid arguments. This can be caused by an  
                    invalid value for "code".  
    Other codes as may be returned by LIB$PUT_OUTPUT or the user's  
    action routine.  
  
SIDE EFFECTS:  
    NONE  
  
--  
  
BEGIN  
BUILTIN  
    NULLPARAMETER;  
  
LOCAL  
    STATUS,          ! Returned condition values  
    TEMPCODE,        ! Local value of CODE  
    OUT_BUF : VECTOR [K_MAX_STR_LEN, BYTE], ! FAO result string  
    OUT_DESC : BLOCK [8, BYTE], ! Output string descriptor  
    FAO_DESC : BLOCK [8, BYTE], ! Input string for FAO  
    FAO_PRMLIST : VECTOR [3, LONG]; ! Input parameter list for FAO  
  
+ Set up out_string descriptor  
-  
    OUT_DESC [DSC$W_LENGTH] = K_MAX_STR_LEN;  
    OUT_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;  
    OUT_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;  
    OUT_DESC [DSC$A_POINTER] = OUT_BUF [0];  
  
+ Now "output" the result, depending on the value of CODE.  
-  
    FAO_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;  
    FAO_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;  
  
    IF (NULLPARAMETER (1)) THEN TEMPCODE = 0 ELSE TEMPCODE = ..CODE;  
  
    CASE .TEMPCODE FROM 0 TO MAX_CODE OF
```

```
236 0328 2
237 0329
238 0330
239 0331
240 0332
241 0333
242 L 0334
243 0335
244 0336
245 0337
246 0338
247 P 0339
248 0340
249 0341
250 0342
251 0343
252 0344
253 0345
254 0346
255 0347
256 P 0348
257 0349
258 0350
259 0351
260 0352
261 0353
262 0354
263 0355
264 0356
265 P 0357
266 0358
267 0359
268 0360
269 0361
270 0362
271 0363
272 0364
273 0365
274 P 0366
275 0367
276 0368
277 0369
278 0370
279 0371
280 0372
281 0373
282 0374
283 0375
284 0376
285 0377
286 0378
287 0379
288 0380
289 0381
290 0382
291 0383
292 0384 2

SET
[0] : ! Default case, all three values
BEGIN
FAO_DESC [DSC$A_POINTER] = UPLIT (BYTE (' !UL calls to LIB$GET_VM, ',
!UL calls to LIB$FREE_VM, !UL bytes still allocated'));
FAO_DESC [DSC$W_LENGTH] = %CHARCOUNT (' !UL calls to LIB$GET_VM, ',
!UL calls to LIB$FREE_VM, !UL bytes still allocated');
FAO_PRMLIST [0] = .LIB$SGL_GETVM_C;
FAO_PRMLIST [1] = .LIB$SGL_FREVM_C;
FAO_PRMLIST [2] = .LIB$SGL_VMINUSE;
STATUS = $FAOL (CTRSTR = FAO_DESC, OUTLEN = OUT_DESC [DSC$W_LENGTH], OUTBUF = OUT_DESC,
PRMLST = FAO_PRMLIST);
END;

[K_GET] : ! Calls to LIB$GET_VM only
BEGIN
FAO_DESC [DSC$A_POINTER] = UPLIT (BYTE (' !UL calls to LIB$GET_VM'));
FAO_DESC [DSC$W_LENGTH] = %CHARCOUNT (' !UL calls to LIB$GET_VM');
FAO_PRMLIST [0] = .LIB$SGL_GETVM_C;
STATUS = $FAOL (CTRSTR = FAO_DESC, OUTLEN = OUT_DESC [DSC$W_LENGTH], OUTBUF = OUT_DESC,
PRMLST = FAO_PRMLIST);
END;

[K_FREE] : ! Calls to LIB$FREE_VM only
BEGIN
FAO_DESC [DSC$A_POINTER] = UPLIT (BYTE (' !UL calls to LIB$FREE_VM'));
FAO_DESC [DSC$W_LENGTH] = %CHARCOUNT (' !UL calls to LIB$FREE_VM');
FAO_PRMLIST [0] = .LIB$SGL_FREVM_C;
STATUS = $FAOL (CTRSTR = FAO_DESC, OUTLEN = OUT_DESC [DSC$W_LENGTH], OUTBUF = OUT_DESC,
PRMLST = FAO_PRMLIST);
END;

[K_IN_USE] : ! Number of bytes in use, only
BEGIN
FAO_DESC [DSC$A_POINTER] = UPLIT (BYTE (' !UL bytes still allocated'));
FAO_DESC [DSC$W_LENGTH] = %CHARCOUNT (' !UL bytes still allocated');
FAO_PRMLIST [0] = .LIB$SGL_VMINUSE;
STATUS = $FAOL (CTRSTR = FAO_DESC, OUTLEN = OUT_DESC [DSC$W_LENGTH], OUTBUF = OUT_DESC,
PRMLST = FAO_PRMLIST);
END;

[OUTRANGE] :
STATUS = LIB$_INVARG;
TES;

IF ( NOT .STATUS) THEN RETURN (.STATUS);
IF (NULLPARAMETER (2)) ! "action" not given?
THEN STATUS = LIB$PUT_OUTPUT (OUT_DESC) ! Default to SYS$OUTPUT
ELSE
IF (NULLPARAMETER (3)) ! "userarg" not given?
THEN STATUS = (.ACTION) (OUT_DESC) ! Call "action" without "userarg"
ELSE
```



```
: 293      0385  2      STATUS = (.ACTION) (OUT_DESC, .USERARG);      ! Call "action" with "userarg"
: 294      0386  2
: 295      0387  2      RETURN (.STATUS);      ! Exit
: 296      0388  1      END;      ! End of routine LIB$SHOW_VM
```

```
.TITLE LIB$STAT_VM
.IDENT \1-003\

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

4C 20 6F 74 20 73 6C 6C 61 63 20 4C 55 21 20 00000 P.AAA: .ASCII \ !UL calls to LIB$GET_VM,\
4C 20 6F 74 20 73 6C 6C 61 63 20 4C 55 21 20 0000F
4C 55 21 20 2C 4D 56 5F 45 45 52 46 24 42 49 00019 .ASCII \ !UL calls to LIB$FREE_VM, !UL bytes sti\
69 74 73 20 73 65 74 79 62 20 00028
64 65 74 61 63 6F 6C 6C 61 20 6C 6C 00037
.        .ASCII \ll allocated\
.        .BLKB 3
4C 20 6F 74 20 73 6C 6C 61 63 20 4C 55 21 20 00041 P.AAB: .ASCII \ !UL calls to LIB$GET_VM\
4D 56 5F 54 45 47 24 42 49 0004D
4C 20 6F 74 20 73 6C 6C 61 63 20 4C 55 21 20 00050 P.AAC: .ASCII \ !UL calls to LIB$FREE_VM\
4D 56 5F 45 45 52 46 24 42 49 0005F
00068 P.AAD: .ASCII \ !UL bytes still allocated\
00077
00081
6C 69 74 73 20 73 65 74 79 62 20 4C 55 21 20 00084
64 65 74 61 63 6F 6C 6C 61 20 6C 6C 00093

.EXTRN LIB$PUT_OUTPUT, LIB$$GL_GETVM_C
.EXTRN LIB$$GL_FREVM_C
.EXTRN LIB$$GL_VMINUSE
.EXTRN LIB$_INVARG, SYS$FAOL

.ENTRY LIB$SHOW_VM, Save R2,R3,R4      : 0214
MOVAB LIB$$GL_VMINUSE, R4
MOVAB LIB$$GL_FREVM_C, R3
MOVAB LIB$$GL_GETVM_C, R2
MOVAB -284(SPT), SP
MOVL #17694975, OUT_DESC      : 0315
MOVAB OUT_BUF, OUT_DESC+4      : 0318
MOVW #270, FAO_DESC+2      : 0322
TSTB (AP)      : 0325
BEQL 1$
TSTL 4(AP)
BNEQ 2$
CLRL TEMPCODE
BRB 3$
MOVL @CODE, TEMPCODE
CASEL TEMPCODE, #0, #3      : 0327
.WORD 5$-4$,-
6$-4$,-
7$-4$,-
8$-4$
MOVL #LIB$_INVARG, STATUS      : 0371
BRB 10$
MOVAB P.AAA, FAO_DESC+4      : 0332
MOVZBW #77, FAO_DESC      : 0334
MOVL LIB$$GL_GETVM_C, FAO_PRMLIST      : 0336
MOVL LIB$$GL_FREVM_C, FAO_PRMLIST+4      : 0337

54 00000000G 00 9E 00002
53 00000000G 00 9E 00009
52 00000000G 00 9E 00010
5E FEE4 CE 9E 00017
14 AE 010E00FF 8F D0 0001C
18 AE 1C AE 9E 00024
OE AE 010E 8F B0 00029
6C 95 0002F
05 13 00031
04 AC D5 00033
04 12 00036
50 D4 00038 1$:
04 11 0003A
50 04 BC D0 0003C 2$:
00 50 CF 00040 3$:
0047 0038 0029 0011 00044 4$:
50 00000000G 8F D0 0004C
10 AE FF09 55 11 00053
OC AE 4D CF 9E 00055 5$:
6E 62 D0 0005B
04 AE 63 D0 00060
63 D0 00063
```

08	AE		64	D0	00067	MOVL	LIB\$\$GL_VMINUSE, FAO_PRMLIST+8	:	0338		
			2B	11	0006B	BRB	9\$	:	0340		
10	AE	FF41	CF	9E	0006D	6\$:	MOVAB	P.AAB, FAO_DESC+4	:	0345	
0C	AE		18	B0	00073		MOVW	#24, FAO_DESC	:	0346	
	6E		62	D0	00077		MOVL	LIB\$\$GL_GETVM_C, FAO_PRMLIST	:	0347	
			1C	11	0007A		BRB	9\$	:	0349	
10	AE	FF4A	CF	9E	0007C	7\$:	MOVAB	P.AAC, FAO_DESC+4	:	0354	
0C	AE		19	B0	00082		MOVW	#25, FAO_DESC	:	0355	
	6E		63	D0	00086		MOVL	LIB\$\$GL_FREVM_C, FAO_PRMLIST	:	0356	
			0D	11	00089		BRB	9\$	:	0358	
10	AE	FF57	CF	9E	0008B	8\$:	MOVAB	P.AAD, FAO_DESC+4	:	0363	
0C	AE		1A	B0	00091		MOVW	#26, FAO_DESC	:	0364	
	6E		64	D0	00095		MOVL	LIB\$\$GL_VMINUSE, FAO_PRMLIST	:	0365	
			5E	DD	00098	9\$:	PUSHL	SP	:	0367	
			18	AE	9F	0009A	PUSHAB	OUT_DESC	:		
			1C	AE	9F	0009D	PUSHAB	OUT_DESC	:		
			18	AE	9F	000A0	PUSHAB	FAO_DESC	:		
00000000G	00		04	FB	000A3		CALLS	#4, -SYSS\$FAOL	:		
	31		50	E9	000AA	10\$:	BLBC	STATUS, 15\$	:	0374	
	02		6C	91	000AD		CMPB	(AP), #2	:	0376	
			05	1F	000B0		BLSSU	11\$	:		
			08	AC	D5	000B2	TSTL	8(AP)	:		
			0B	12	000B5		BNEQ	12\$	:		
00000000G	00		14	AE	9F	000B7	11\$:	PUSHAB	OUT_DESC	:	0378
			01	FB	000BA		CALLS	#1, -LIB\$PUT_OUTPUT	:		
	03		04	00	00C1		RET		:		
			6C	91	000C2	12\$:	CMPB	(AP), #3	:	0381	
			05	1F	000C5		BLSSU	13\$	:		
			0C	AC	D5	000C7	TSTL	12(AP)	:		
			08	12	000CA		BNEQ	14\$	:		
			14	AE	9F	000CC	13\$:	PUSHAB	OUT_DESC	:	0383
08	BC		01	FB	000CF		CALLS	#1, -@ACTION	:		
			04	00	00D3		RET		:		
			0C	AC	DD	000D4	14\$:	PUSHL	USERARG	:	0385
			18	AE	9F	000D7	PUSHAB	OUT_DESC	:		
08	BC		02	FB	000DA		CALLS	#2, -@ACTION	:		
			04	00	00DE	15\$:	RET		:	0388	

; Routine Size: 223 bytes, Routine Base: _LIB\$CODE + 009E

; 297 0389 1


```
299 0390 1 GLOBAL ROUTINE LIB$STAT_VM (
300 0391 1     CODE
301 0392 1     VALUE
302 0393 1 ) =
303 0394 1
304 0395 1 ++
305 0396 1 FUNCTIONAL DESCRIPTION:
306 0397 1
307 0398 1     LIB$STAT_VM returns to its caller one of three available
308 0399 1     statistics. Unlike LIB$SHOW_VM which formats the values
309 0400 1     for output, LIB$STAT_VM returns the value to a location
310 0401 1     specified as a parameter.
311 0402 1
312 0403 1     Only one of the three statistics can be returned by one call
313 0404 1     to LIB$STAT_VM. "code" must be one of the three values
314 0405 1     described for LIB$SHOW_VM. A "code" of zero is invalid.
315 0406 1
316 0407 1 CALLING SEQUENCE:
317 0408 1
318 0409 1     status.wlc.v = LIB$STAT_VM (code.rl.r, value.wl.r)
319 0410 1
320 0411 1 FORMAL PARAMETERS:
321 0412 1
322 0413 1     code    - Specifies which statistic is to be returned. Allowed
323 0414 1              values are:
324 0415 1              1 = Number of calls to LIB$GET_VM
325 0416 1              2 = Number of calls to LIB$FREE_VM
326 0417 1              3 = Number of bytes allocated by LIB$GET_VM but
327 0418 1                  not yet freed by LIB$FREE_VM.
328 0419 1              Note that it is invalid to omit "code" or to give
329 0420 1                  a "code" of zero.
330 0421 1
331 0422 1     value   - The location to store the result. All values are
332 0423 1              longword integers.
333 0424 1
334 0425 1 IMPLICIT INPUTS:
335 0426 1
336 0427 1     The statistics cells maintained by LIB$VM.
337 0428 1
338 0429 1 IMPLICIT OUTPUTS:
339 0430 1
340 0431 1     NONE
341 0432 1
342 0433 1 COMPLETION CODES:
343 0434 1
344 0435 1     SSS$ NORMAL    - Successful completion
345 0436 1     LIB$_INVARG    - Invalid argument.
346 0437 1
347 0438 1 SIDE EFFECTS:
348 0439 1
349 0440 1     NONE
350 0441 1
351 0442 1 --
352 0443 1
353 0444 2 BEGIN
354 0445 2
355 0446 2 BUILTIN
```

```

: 356      0447 2      NULLPARAMETER;
: 357      0448 2
: 358      0449 2      LOCAL
: 359      0450 2      STATUS;
: 360      0451 2
: 361      0452 2      ! Returned status condition
: 362      0453 2
: 363      0454 2      !+
: 364      0455 2      Now return appropriate statistic determined by CODE
: 365      0456 2      !-
: 366      0457 2      CASE .CODE FROM 1 TO MAX_CODE OF
: 367      0458 2      SET
: 368      0459 2      [K_GET] :
: 369      0460 2      .VALUE = .LIB$$GL_GETVM_C;
: 370      0461 2
: 371      0462 2      [K_FREE] :
: 372      0463 2      .VALUE = .LIB$$GL_FREVM_C;
: 373      0464 2
: 374      0465 2      [K_IN_USE] :
: 375      0466 2      .VALUE = .LIB$$GL_VMINUSE;
: 376      0467 2
: 377      0468 2      [OUTRANGE] :
: 378      0469 2      RETURN (LIB$_INVARG);
: 379      0470 2      TES;
: 380      0471 2
: 381      0472 2      RETURN (SS$_NORMAL);
: 382      0473 1      END;

```

02	01	04	0000	00000	.ENTRY	LIB\$STAT_VM, Save nothing	: 0390
0022	0018	BC	CF	00002	CASEL	@CODE, #T, #2	: 0456
		000E		00007	.WORD	2\$-1\$,-	:
						3\$-1\$,-	:
						4\$-1\$	:
	50	00000000G	8F	D0 0000D	MOVL	#LIB\$_INVARG, R0	: 0469
				04 00014	RET		:
08	BC	00000000G	00	D0 00015	MOVL	LIB\$\$GL_GETVM_C, @VALUE	: 0460
			12	11 0001D	BRB	5\$	:
08	BC	00000000G	00	D0 0001F	MOVL	LIB\$\$GL_FREVM_C, @VALUE	: 0463
			08	11 00027	BRB	5\$	:
08	BC	00000000G	00	D0 00029	MOVL	LIB\$\$GL_VMINUSE, @VALUE	: 0466
	50		01	D0 00031	MOVL	#1, R0	: 0472
				04 00034	RET		: 0473

; Routine Size: 53 bytes, Routine Base: _LIB\$CODE + 017D

```

: 383      0474 1
: 384      0475 1 END
: 385      0476 1
: 386      0477 0 ELUDOM

```

!End of module LIB\$STAT_VM

PSECT SUMMARY

```
:
:      Name          Bytes          Attributes
:
:  _LIB$CODE        434  NOVEC,NOWRT, RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
```

Library Statistics

```
:
:      File          ----- Symbols ----- Pages Processing
:                   Total   Loaded   Percent   Mapped   Time
:
:  _$255$DUA28:[SYSLIB]STARLET.L32;1    9776      10      0      581      00:00.7
```

COMMAND QUALIFIERS

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

```
: Size:          276 code + 158 data bytes
: Run Time:      00:05.9
: Elapsed Time:  00:24.8
: Lines/CPU Min: 4826
: Lexemes/CPU-Min: 31902
: Memory Used:   96 pages
: Compilation Complete
```


0210 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

