


```

LL      IIIIII  BBBB BBBB  GGGGGGGG  EEEEEEEEE  TTTTTTTTTT  DDDDDDDD  VV      VV      IIIIII
LL      IIIIII  BBBB BBBB  GGGGGGGG  EEEEEEEEE  TTTTTTTTTT  DDDDDDDD  VV      VV      IIIIII
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LL      II      BB      BB  GG      EE      TT      DD      DD  VV      VV      II
LLLLLLLL  IIIIII  BBBB BBBB  GGGGGG  EEEEEEEEE  TT      DD      DD  VV      VV      IIIIII
LLLLLLLL  IIIIII  BBBB BBBB  GGGGGG  EEEEEEEEE  TT      DD      DD  VV      VV      IIIIII

LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LL      II      SS      SS      SSSSSS
LLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLL  IIIIII  SSSSSSSS

```



```
1 0001 0 MODULE LIB$GETDVI ( %TITLE 'Get Device/Volume Information'
2 0002 0 IDENT = '1-003' ! File: LIB$GETDVI.B32 Edit: SBL1003
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 Utility Library
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 LIB$GETDVI obtains a specified item of Device/Volume
36 0036 1 and formats it in an appropriate manner.
37 0037 1
38 0038 1 ENVIRONMENT: User mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Steven B. Lionel, CREATION DATE: 11-Jan-1983
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. SBL 11-Jan-1983
45 0045 1 1-002 - Change format codes to LIB$K FMT. Change string length from
46 0046 1 NAM$C_MAXRSS (252) to LNM$C_NAMLENGTH (255). 11-Mar-1983
47 0047 1 1-003 - Change string length to 512. SBL 11-Mar-1983
48 0048 1 --
49 0049 1
```

```
: 51 0050 1 %SBTTL 'Declarations'
: 52 0051 1
: 53 0052 1 PROLOGUE FILE:
: 54 0053 1
: 55 0054 1
: 56 0055 1 REQUIRE 'RTLIN:LIBPROLOG'; ! Switches, PSECTS, macros
: 57 0126 1
: 58 0127 1
: 59 0128 1 LINKAGES:
: 60 0129 1
: 61 0130 1 NONE
: 62 0131 1
: 63 0132 1 TABLE OF CONTENTS:
: 64 0133 1
: 65 0134 1
: 66 0135 1 FORWARD ROUTINE
: 67 0136 1 LIB$GETDVI; ! Get Device/Volume Information
: 68 0137 1
: 69 0138 1
: 70 0139 1 MACROS:
: 71 0140 1
: 72 0141 1 NONE
: 73 0142 1
: 74 0143 1 EQUATED SYMBOLS:
: 75 0144 1
: 76 0145 1 NONE
: 77 0146 1
: 78 0147 1 FIELDS:
: 79 0148 1
: 80 0149 1 NONE
: 81 0150 1
: 82 0151 1 OWN STORAGE:
: 83 0152 1
: 84 0153 1 NONE
: 85 0154 1
: 86 0155 1 EXTERNALS:
: 87 0156 1
: 88 0157 1
: 89 0158 1 EXTERNAL ROUTINE
: 90 0159 1 LIB$GETDVI; ! Internal routine
: 91 0160 1 LIB$ANALYZE_SDESC_R2: LIB$ANALYZE_SDESC_R2$LINKAGE; ! Get length and pointer
: 92 0161 1 LIB$GET_EF; ! Allocate event flag number
: 93 0162 1 LIB$FREE_EF: NOVALUE; ! Free event flag number
: 94 0163 1 LIB$SCOPY_R_DX6: LIB$SCOPY_R_DX6$LINKAGE; ! Copy string by reference.
: 95 0164 1
: 96 0165 1 EXTERNAL LITERAL
: 97 0166 1 LIB$_INVARG; ! Invalid argument
: 98 0167 1 LIB$_STRTRU; ! String truncated
```



```
0168 1 XSBTTL 'LIB$GETDVI - Get Device/Volume Information'
0169 1 GLOBAL ROUTINE LIB$GETDVI (
0170 1     ITEM CODE: REF VECTOR [, WORD],      ! Code of desired item
0171 1     CHANNEL: REF VECTOR [, WORD],         ! Channel number
0172 1     DEVICE_NAME: REF BLOCK [, BYTE],      ! Device name
0173 1     OUT_VALUE: REF VECTOR [, LONG],       ! Output numeric value
0174 1     OUT_STRING: REF BLOCK [, BYTE],       ! Output string descriptor
0175 1     OUT_LEN: REF VECTOR [, WORD]          ! Output string length
0176 1 ) =
0177 1
0178 1 ++
0179 1 FUNCTIONAL DESCRIPTION:
0180 1
0181 1     LIB$GETDVI provides a simplified interface to the $GETDVI system
0182 1     service. It returns information about an I/O device. The
0183 1     calling process does not have to have a channel assigned to the
0184 1     device. Two categories of information can be returned:
0185 1
0186 1     o The primary device characteristics.
0187 1
0188 1     o The secondary device characteristics.
0189 1
0190 1     LIB$GETDVI provides the following features in addition to those
0191 1     provided by the $GETDVI system service:
0192 1
0193 1     o Instead of a list of item descriptors, which may be
0194 1     difficult to construct in high-level languages, the
0195 1     single item desired is specified as an integer code
0196 1     which is passed by reference. Results are written to
0197 1     separate arguments.
0198 1
0199 1     o For items which return numeric values, LIB$GETDVI can
0200 1     optionally provide a formatted string interpretation of
0201 1     the value. For example, if the device owner UIC is
0202 1     requested, LIB$GETDVI can return the UIC formatted as
0203 1     "[g,m]".
0204 1
0205 1     o For string arguments, all string classes supported by
0206 1     the Run-Time Library are understood.
0207 1
0208 1     o Calls to LIB$GETDVI are synchronous; LIB$GETDVI calls
0209 1     LIB$GET_EF to allocate a local event flag number for
0210 1     synchronization.
0211 1
0212 1     LIB$GETDVI does not provide the ability to obtain more than one
0213 1     item of information in a single call.
0214 1
0215 1 CALLING SEQUENCE:
0216 1
0217 1     ret-status.wlc.v = LIB$GETDVI (
0218 1         item-code.rw.r,
0219 1         [channel.rw.r],
0220 1         [device-name.rt.dx],
0221 1         [out-value.wz.r]
0222 1         [, [out-string.wt.dx]
0223 1         [, [out-len.wwu.r] ]])
0224 1
```

```
157 0225 1 FORMAL PARAMETERS:
158 0226 1
159 0227 1 item-code
160 0228 1 A longword integer item identifier code that specifies which item
161 0229 1 of information you are requesting. All valid $GETDVI item codes,
162 0230 1 whose names begin with DVI$, are accepted.
163 0231 1
164 0232 1 channel
165 0233 1 A longword integer specifying the VMS I/O channel assigned to
166 0234 1 the device for which information is to be returned. If not
167 0235 1 specified, device-name is used instead. You must specify
168 0236 1 either channel or device-name, but not both. If both or
169 0237 1 neither are specified, the error status $$$_IVDEVNAM is
170 0238 1 returned.
171 0239 1
172 0240 1 device-name
173 0241 1 A string specifying the name of the device for which
174 0242 1 information is to be returned. The string may be either a
175 0243 1 physical device name or a logical name. If the string
176 0244 1 contains a colon, the colon and the characters that follow it
177 0245 1 are ignored. If the first character in the string is an
178 0246 1 underscore character (_), the name is considered a physical
179 0247 1 device name. Otherwise, the name is considered a logical name
180 0248 1 and logical name translation is performed until either a
181 0249 1 physical device name is found or the system default number of
182 0250 1 translations has been performed.
183 0251 1
184 0252 1 If not specified, channel is used instead. You must specify
185 0253 1 either channel or device-name, but not both. If both or
186 0254 1 neither are specified, the error status $$$_IVDEVNAM is
187 0255 1 returned.
188 0256 1
189 0257 1 out-value
190 0258 1 A longword or quadword into which is placed the numeric
191 0259 1 value of the information requested. If an item only returns
192 0260 1 a string value, this parameter is ignored.
193 0261 1
194 0262 1 out-string
195 0263 1 A string into which is placed the string representation of
196 0264 1 the information requested. If out-string is not specified,
197 0265 1 and the value returned has only a string representation, the
198 0266 1 error status LIB$_INVARG is returned.
199 0267 1
200 0268 1 out-len
201 0269 1 A word integer into which is placed the number of significant
202 0270 1 characters written to out-string, not including blank padding
203 0271 1 or truncated characters.
204 0272 1
205 0273 1 IMPLICIT INPUTS:
206 0274 1
207 0275 1 NONE
208 0276 1
209 0277 1 IMPLICIT OUTPUTS:
210 0278 1
211 0279 1 NONE
212 0280 1
213 0281 1 COMPLETION STATUS:
```


LIB\$GETDVI
1-003

Get Device/Volume Information
LIB\$GETDVI - Get Device/Volume Information

L 6
16-Sep-1984 00:58:16
14-Sep-1984 12:38:56

VAX-11 Bliss-32 V4.0-742
[LIBRTL.SRC]LIBGETDVI.B32;1

Page 5
(3)

:	214	0282	1	:		
:	215	0283	1	:	SSS NORMAL	Normal successful completion
:	216	0284	1	:	LIB\$-STRTRU	String truncated. This is an alternate success status.
:	217	0285	1	:	LIB\$-INSEF	Insufficient event flags
:	218	0286	1	:	LIB\$-INVSTRDES	Invalid string descriptor
:	219	0287	1	:	LIB\$-WRONUMARG	Wrong number of arguments
:	220	0288	1	:	LIB\$-xxx	Any error status from LIB\$SCOPY_R_DX
:	221	0289	1	:	SSS-BADPARAM	The item code is not recognized as valid.
:	222	0290	1	:	SSS-xxx	Any error status from \$GETDVI
:	223	0291	1	:		
:	224	0292	1	:	SIDE EFFECTS:	
:	225	0293	1	:		
:	226	0294	1	:	NONE	
:	227	0295	1	:		
:	228	0296	1	:	--	

```
230 0297 2 BEGIN
231 0298
232 0299
233 0300 LOCAL
234 0301 RET_STRING: VECTOR [512, BYTE],      ! Local string for value
235 0302 RET_LENGTH: WORD,                  ! Length of RET_STRING
236 0303 RET_NUMBER: VECTOR [2, LONG],      ! Local quadword for value
237 0304 RET_TYPE,                          ! Returned type code
238 0305 LCL_DEVNAM_DSC: BLOCK [8, BYTE],   ! Local descriptor for DEVNAM
239 0306 DEVNAM_DSC_ADR,                    ! Address of DEVNAM descriptor
240 0307 EVENT_FLAG,                       ! Event flag number
241 0308 STR_STATUS, EF_STATUS, DVI_STATUS, COPY_STATUS; ! Return statuses
242 0309
243 0310 BUILTIN
244 0311 NULLPARAMETER;
245 0312
246 0313 !+ Validate argument count.
247 0314 !-
248 0315
249 0316 $LIB$VALIDATE_ARGCOUNT (4,6);
250 0317
251 0318 !+
252 0319 Build static descriptor for DEVICE_NAME, if any.
253 0320 !-
254 0321
255 0322 IF NULLPARAMETER (3)
256 0323 THEN
257 0324     DEVNAM_DSC_ADR = 0      ! Omitted
258 0325 ELSE
259 0326     BEGIN
260 0327         LCL_DEVNAM_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
261 0328         LCL_DEVNAM_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
262 0329         STR_STATUS = LIB$ANALYZE_SDESC R2 (DEVICE_NAME [0,0,0,0];
263 0330             LCL_DEVNAM_DSC [DSC$B_LENGTH], LCL_DEVNAM_DSC [DSC$A_POINTER]);
264 0331         IF NOT .STR_STATUS
265 0332         THEN
266 0333             RETURN .STR_STATUS;
267 0334         DEVNAM_DSC_ADR = LCL_DEVNAM_DSC;
268 0335         END;
269 0336
270 0337 !+
271 0338 Allocate an event flag number to use for the $GETDVI.
272 0339 !-
273 0340
274 0341 EF_STATUS = LIB$GET_EF (EVENT_FLAG);
275 0342 IF NOT .EF_STATUS
276 0343 THEN
277 0344     RETURN .EF_STATUS;
278 0345
279 0346 !+
280 0347 Call LIB$$GETDVI to do the work.
281 0348 !-
282 0349
283 0350 DVI_STATUS = LIB$$GETDVI (
284 0351     .ITEM_CODE [0],
285 0352     RET_STRING,
286 0353     RET_NUMBER,
```



```
287 0354 2 RET_LENGTH,  
288 0355 2 RET_TYPE,  
289 0356 2 .EVENT_FLAG,  
290 0357 2 (IF NULLPARAMETER (2) THEN 0 ELSE .CHANNEL [0]),  
291 0358 2 .DEVNAM_DSC_ADR);  
292 0359 2  
293 0360 2  
294 0361 2 !+ Free the event flag.  
295 0362 2 !-  
296 0363 2  
297 0364 2 LIB$FREE_EF (EVENT_FLAG);  
298 0365 2  
299 0366 2  
300 0367 2 !+ Check for errors.  
301 0368 2 !-  
302 0369 2  
303 0370 2 IF NOT .DVI_STATUS  
304 0371 2 THEN  
305 0372 2 RETURN .DVI_STATUS;  
306 0373 2  
307 0374 2 !+ Copy the numeric value, if desired.  
308 0375 2 !-  
309 0376 2  
310 0377 2  
311 0378 2 IF OUT_VALUE [0] NEQA 0  
312 0379 2 THEN  
313 0380 2 BEGIN  
314 0381 2 IF .RET_TYPE GTRU LIB$K_FMT_MAXSTRING ! Is it a number?  
315 0382 2 THEN  
316 0383 2 BEGIN  
317 0384 2 OUT_VALUE [0] = .RET_NUMBER [0];  
318 0385 2 IF .RET_TYPE EQL LIB$K_FMT_DATE OR  
319 0386 2 .RET_TYPE EQL LIB$K_FMT_PRIVILEGE  
320 0387 2 THEN  
321 0388 2 OUT_VALUE [1] = .RET_NUMBER [1]; ! Store second longword  
322 0389 2 END;  
323 0390 2 END;  
324 0391 2  
325 0392 2 !+ Store string value if desired.  
326 0393 2 !-  
327 0394 2  
328 0395 2  
329 0396 2 IF NOT NULLPARAMETER (5)  
330 0397 2 THEN  
331 0398 2 BEGIN  
332 0399 2 COPY_STATUS = LIB$SCOPY_R_DX6 (.RET_LENGTH, RET_STRING,  
333 0400 2 OUT_STRING [0,0,0,0]);  
334 0401 2 IF NOT NULLPARAMETER (6)  
335 0402 2 THEN  
336 0403 2 BEGIN  
337 0404 2 !+ Store result string length.  
338 0405 2 !-  
339 0406 2  
340 0407 2 OUT_LEN [0] = .RET_LENGTH;  
341 0408 2 IF .COPY_STATUS EQL LIB$STRTRU  
342 0409 2 THEN  
343 0410 2 LIB$ANALYZE_SDESC_R2 (OUT_STRING [0,0,0,0]; OUT_LEN [0]);
```

```

: 344      0411      3      END;
: 345      0412      3      RETURN .COPY_STATUS;
: 346      0413      3      END
: 347      0414      2      ELSE IF .RET_TYPE LEQU LIB$K_FMT_MAXSTRING
: 348      0415      2      THEN
: 349      0416      2      RETURN LIB$_INVARG;      ! Only string value, but nothing to return it in
: 350      0417      2
: 351      0418      2      RETURN SS$_NORMAL;      ! Success
: 352      0419      2
: 353      0420      1      END;

```

! End of routine LIB\$GETDVI

```

.TITLE LIB$GETDVI Get Device/Volume Information
.IDENT \1-003\

```

```

.EXTRN LIB$$GETDVI, LIB$ANALYZE_SDESC_R2
.EXTRN LIB$GET_EF, LIB$FREE_EF
.EXTRN LIB$SCOPY_R_DX6
.EXTRN LIB$_INVARG, LIB$_STRTRU
.EXTRN LIB$_WRONUMARG

```

```

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

```

```

.ENTRY LIB$GETDVI, Save R2, R3, R4, R5, R6, R7

```

```

MOVAB LIB$ANALYZE_SDESC_R2, R7      : 0169
MOVAB -540(SP), SP
SUBB3 #4, (AP), DIFF      : 0316
CMPB DIFF, #2
BLEQU 1$
MOVL #LIB$_WRONUMARG, R0
RET
CMPB (AP), #3      : 0322
BLSSU 2$
TSTL 12(AP)
BNEQ 3$
CLRL DEVNAM_DSC_ADR      : 0324
BRB 4$
MOVW #270, LCL_DEVNAM_DSC+2      : 0327
MOVL DEVICE_NAME, R0      : 0330
JSB LIB$ANALYZE_SDESC_R2
MOVW R1, LCL_DEVNAM_DSC
MOVL R2, LCL_DEVNAM_DSC+4
BLBC STR_STATUS, 5$      : 0331
MOVAB LCL_DEVNAM_DSC, DEVNAM_DSC_ADR      : 0334
PUSHAB EVENT_FLAG      : 0341
CALLS #1, LIB$GET_EF      : 0342
BLBS EF_STATUS, 8$
RET
PUSHL DEVNAM_DSC_ADR      : 0358
CMPB (AP), #2      : 0357
BLSSU 7$
TSTL 8(AP)
BNEQ 8$
CLRL -(SP)
BRB 9$
MOVZWL @CHANNEL, -(SP)
PUSHL EVENT_FLAG      : 0356

```


		0C	AE	9F	0006D	PUSHAB	RET_TYPE	:	0350	
		14	AE	9F	00070	PUSHAB	RET_LENGTH	:		
		28	AE	9F	00073	PUSHAB	RET_NUMBER	:		
		34	AE	9F	00076	PUSHAB	RET_STRING	:		
		04	BC	3C	00079	MOVZWL	@ITEM CODE, -(SP)	:	0351	
00000000G	7E		08	FB	0007D	CALLS	#8, LIB\$GETDVI	:		
	00		50	D0	00084	MOVL	R0, DVI_STATUS	:		
	52		08	AE	9F	00087	PUSHAB	EVENT_FLAG	:	0364
00000000G	00		01	FB	0008A	CALLS	#1, LIB\$FREE_EF	:		
	04		52	E8	00091	BLBS	DVI_STATUS, TOS	:	0370	
	50		52	D0	00094	MOVL	DVI_STATUS, R0	:	0372	
				04	00097	RET		:		
	50	10	AC	D0	00098	10\$: MOVL	OUT_VALUE, R0	:	0378	
			18	13	0009C	BEQL	12\$	:		
	03		6E	D1	0009E	CMPL	RET_TYPE, #3	:	0381	
			13	1B	000A1	BLEQU	12\$	:		
	60	14	AE	D0	000A3	MOVL	RET_NUMBER, (R0)	:	0384	
	0B		6E	D1	000A7	CMPL	RET_TYPE, #11	:	0385	
			05	13	000AA	BEQL	11\$	:		
	0C		6E	D1	000AC	CMPL	RET_TYPE, #12	:	0386	
			05	12	000AF	BNEQ	12\$	:		
04	A0	18	AE	D0	000B1	11\$: MOVL	RET_NUMBER+4, 4(R0)	:	0388	
	05		6C	91	000B6	12\$: CMPB	(AP), #5	:	0396	
			40	1F	000B9	BLSSU	14\$	:		
		14	AC	D5	000BB	TSTL	20(AP)	:		
			3B	13	000BE	BEQL	14\$	:		
	51	1C	AE	9E	000C0	MOVAB	RET_STRING, R1	:	0399	
	52	14	AC	D0	000C4	MOVL	OUT_STRING, R2	:	0400	
	50	04	AE	3C	000C8	MOVZWL	RET_LENGTH, R0	:		
		00000000G	00	16	000CC	JSB	LIB\$COPY_R DX6	:		
	53		50	D0	000D2	MOVL	R0, COPY_STATUS	:		
	06		6C	91	000D5	CMPB	(AP), #6	:	0401	
			1D	1F	000D8	BLSSU	13\$	:		
		18	AC	D5	000DA	TSTL	24(AP)	:		
			18	13	000DD	BEQL	13\$	:		
18	BC	04	AE	B0	000DF	MOVW	RET_LENGTH, @OUT_LEN	:	0407	
00000000G	8F		53	D1	000E4	CMPL	COPY_STATUS, #LIB\$_STRTRU	:	0408	
			0A	12	000EB	BNEQ	13\$	:		
	50	14	AC	D0	000ED	MOVL	OUT_STRING, R0	:	0410	
			67	16	000F1	JSB	LIB\$ANALYZE_SDESC_R2	:		
18	BC		51	B0	000F3	MOVW	R1, @OUT_LEN	:		
	50		53	D0	000F7	13\$: MOVL	COPY_STATUS, R0	:	0412	
				04	000FA	RET		:		
	03		6E	D1	000FB	14\$: CMPL	RET_TYPE, #3	:	0414	
			08	1A	000FE	BGTRU	15\$	:		
	50	00000000G	8F	D0	00100	MOVL	#LIB\$_INVARG, R0	:	0416	
				04	00107	RET		:		
	50		01	D0	00108	15\$: MOVL	#1, R0	:	0418	
				04	0010B	RET		:	0420	

; Routine Size: 268 bytes, Routine Base: _LIB\$CODE + 0000

: 355
 : 356
 : 357

0421 1 END
 0422 1
 0423 0 ELUDOM

! End of module LIB\$GETDVI

:
 :
 :
 :

PSECT SUMMARY

Name
 Bytes
 Attributes

:
 :

_LIB\$CODE
 268 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]STARLET.L32;1
 9776 7 0 581 00:00.7
 \$255\$DUA28:[LIBRTL.OBJ]RTLLIB.L32;1
 36 6 16 8 00:00.1

:
 :
 :

COMMAND QUALIFIERS

:
 :
 :

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

:
 :
 :
 :
 :
 :
 :

Size: 268 code + 0 data bytes
 Run Time: 00:05.1
 Elapsed Time: 00:27.7
 Lines/CPU Min: 4966
 Lexemes/CPU-Min: 17671
 Memory Used: 108 pages
 Compilation Complete

0207 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LIBFLTUND
LIS

LIBGETSYI
LIS

LIBICHAR
LIS

LIBINITIA
LIS

LIBFIXUPF
LIS

LIBGETFOR
LIS

LIBGETINP
LIS

LIBINISHR
LIS

LIBGETDVI
LIS

LIBGETOPC
LIS

LIBFNDING
LIS

LIBGETMSG
LIS

LIBINDEX
LIS

LIBINSQHI
LIS

LIBGETUP1
LIS

LIBGETTAB
LIS