


```

1 0001 0 MODULE RPG$EXT_INDS( %TITLE 'Read external RPGII indicators'
2 0002 0 IDENT = '1-003' ! file: RPGEXTIND.B32 EDIT:DG1003
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 ++
32 0032 1 FACILITY: RPGII SUPPORT
33 0033 1
34 0034 1 ABSTRACT
35 0035 1
36 0036 1 This routine reads external RPGII indicators.
37 0037 1
38 0038 1
39 0039 1 ENVIRONMENT: Vax-11 User Mode
40 0040 1
41 0041 1 AUTHOR: Debess Grabazs, CREATION DATE: 1-FEB-1983
42 0042 1
43 0043 1 MODIFIED BY:
44 0044 1
45 0045 1 1-001 - Original. DG 1-FEB-1983
46 0046 1 1-002 - Take out %ASCID - was making code unsharable. DG 11-Jul-1983
47 0047 1 1-003 - Change dynamic string to static. DG 15-May-1984
48 0048 1 --
49 0049 1
50 0050 1 !<BLF/PAGE>

```

```
52 0051 1 %SBTTL 'Declarations'  
53 0052 1 |  
54 0053 1 | +  
55 0054 1 | PROLOGUE FILE:  
56 0055 1 | -  
57 0056 1 REQUIRE 'RTLIN:RPGPROLOG';  
58 0121 1 |  
59 0122 1 |  
60 0123 1 | +  
61 0124 1 | LINKAGES  
62 0125 1 | NONE  
63 0126 1 | -  
64 0127 1 |  
65 0128 1 | +  
66 0129 1 | TABLE OF CONTENTS:  
67 0130 1 | -  
68 0131 1 |  
69 0132 1 FORWARD ROUTINE  
70 0133 1 RPGE$EXT_INDS;  
71 0134 1 |  
72 0135 1 | +  
73 0136 1 | INCLUDE FILES  
74 0137 1 | NONE  
75 0138 1 | -  
76 0139 1 |  
77 0140 1 | +  
78 0141 1 | MACROS  
79 0142 1 | NONE  
80 0143 1 | -  
81 0144 1 |  
82 0145 1 | +  
83 0146 1 | EQUATED SYMBOLS  
84 0147 1 | NONE  
85 0148 1 | -  
86 0149 1 |  
87 0150 1 | +  
88 0151 1 | EXTERNAL REFERENCES  
89 0152 1 | -  
90 0153 1 |  
91 0154 1 EXTERNAL ROUTINE  
92 0155 1 LIB$STOP,  
93 0156 1 OT$SCVT_L_TI;  
94 0157 1 |  
95 0158 1 EXTERNAL LITERAL  
96 0159 1 RPGE$_FATINTERR,  
97 0160 1 RPGE$_INVARG;  
98 0161 1 |
```

! Switches, PSECTs, macros,
! Linkages and LIBRARYs

! Stop execution via signalling
! Convert longword to text

! Fatal internal error
! Invalid argument

```
0162 1 %SBTTL 'RPG$EXT_INDS - Read external RPGII indicators'
0163 1 GLOBAL ROUTINE RPG$EXT_INDS(
0164 1     NUMBER                                ! Indicator number
0165 1     ) =
0166 1 ++
0167 1
0168 1 FUNCTIONAL DESCRIPTION:
0169 1
0170 1     The logical name RPG$EXT_INDS can be assigned to an
0171 1     indicator list of up to eight RPGII external indicators.
0172 1     An indicator is set ON by specifying it in the RPG$EXT_INDS
0173 1     indicator list. This routine accepts the number of a
0174 1     particular indicator as its input parameter. It then checks
0175 1     the indicator list for a match. If the indicator is present
0176 1     in the list, the routine returns a value of 1; otherwise it
0177 1     returns 0.
0178 1
0179 1 CALLING SEQUENCE
0180 1
0181 1     indicator_value.wlc.v = RPG$EXT_INDS (number.rl.v)
0182 1
0183 1 FORMAL PARAMETERS:
0184 1
0185 1     number            indicator number
0186 1
0187 1 IMPLICIT INPUTS:
0188 1
0189 1     RPG$EXT_INDS - logical name which may be assigned to
0190 1     an indicator list of up to eight RPGII
0191 1     external indicators
0192 1
0193 1 IMPLICIT OUTPUTS:
0194 1
0195 1     NONE
0196 1
0197 1 RETURN VALUE:
0198 1
0199 1     SWITCH_VALUE - 1 means indicator is ON
0200 1                   0 means indicator is OFF
0201 1
0202 1 SIDE EFFECTS:
0203 1
0204 1     NONE
0205 1
0206 1 --
0207 1
0208 1
0209 1 !<BLF/PAGE>
```

```

149 0210 1
150 0211 2 BEGIN
151 0212 3
152 0213 4 LITERAL
153 0214 5     MIN_INDICATOR = 1,
154 0215 6     MAX_INDICATOR = 8,
155 0216 7     ON = 1,
156 0217 8     OFF = 0;
157 0218 9
158 0219 10 LOCAL
159 0220 11     DEST_DESC: BLOCK[8,BYTE],           ! Local desc for call to $TRNLOG
160 0221 12     LOG_DESC: BLOCK[8,BYTE],         ! Local desc for call to $TRNLOG
161 0222 13     NAME_LENGTH,                       ! Length of logical name string
162 0223 14     NAME_TEXT: VECTOR [LNM$C_NAMLENGTH, BYTE]; ! Local string for logical name
163 0224 15
164 0225 16
165 0226 17
166 0227 18
167 0228 19     +
168 0229 20     | Make sure input parameter is in acceptable range.
169 0230 21     -
170 0231 22 IF .NUMBER LSS MIN_INDICATOR OR .NUMBER GTR MAX_INDICATOR
171 0232 23 THEN LIB$STOP (RPG$_INVARG);
172 0233 24
173 0234 25     +
174 0235 26     | Initialize descriptors for call to $TRNLOG.
175 0236 27     -
176 0237 28
177 0238 29 DEST_DESC[DSC$W_LENGTH] = LNM$C_NAMLENGTH;
178 0239 30 DEST_DESC[DSC$B_CLASS] = DSC$K_CLASS_S;
179 0240 31 DEST_DESC[DSC$B_DTYPE] = DSC$K_DTYPE_T;
180 0241 32 DEST_DESC[DSC$A_POINTER] = NAME_TEXT;
181 0242 33
182 0243 34
183 0244 35 LOG_DESC[DSC$W_LENGTH] = %CHARCOUNT ('RPGSEXT_INDS');
184 0245 36 LOG_DESC[DSC$B_CLASS] = DSC$K_CLASS_S;
185 0246 37 LOG_DESC[DSC$B_DTYPE] = DSC$K_DTYPE_T;
186 0247 38 LOG_DESC[DSC$A_POINTER] = UPLIT ('RPGSEXT_INDS');
187 0248 39
188 0249 40     +
189 0250 41     | Try to translate the logical name RPGSEXT_INDS.
190 0251 42     -
191 0252 43
192 0253 44
193 0254 45 P IF ($TRNLOG (LOGNAM = LOG_DESC, RSLLEN = NAME_LENGTH,
194 0255 46     RSLBUF = DEST_DESC)) EQL SS$_NORMAL
195 0256 47 THEN
196 0257 48     +
197 0258 49     | Logical name translated - see if requested indicator
198 0259 50     | is set or not
199 0260 51     -
200 0261 52
201 0262 53     BEGIN
202 0263 54
203 0264 55 LOCAL
204 0265 56     NUM_STR: BLOCK[8,BYTE],           ! Local desc for call to OTSS$CVT_L_TI
205 0266 57

```

```

: 206      0267      NUM_STR TEXT: INITIAL (%C'n');           ! Local string for converted number
: 207      0268      NUM_STR[DSC$W_LENGTH] = %CHARCOUNT ('n');
: 208      0269      NUM_STR[DSC$B_CLASS] = DSC$K_CLASS_S;           ! Initialize descriptor for
: 209      0270      NUM_STR[DSC$B_DTYPE] = DSC$K_DTYPE_T;           ! conversion
: 210      0271      NUM_STR[DSC$A_POINTER] = NUM_STR TEXT;
: 211      0272      IF NOT (OTSS$CVT_L_TI (NUMBER, NUM_STR)) ! Convert input number to string
: 212      0273      THEN
: 213      0274      LIB$STOP (RPG$ FATINTERR);
: 214      0275      IF CH$FAIL (CH$FIND_CH (.NAME_LENGTH, NAME_TEXT, .NUM_STR_TEXT))
: 215      0276      THEN
: 216      0277      RETURN OFF
: 217      0278      ELSE
: 218      0279      RETURN ON;
: 219      0280
: 220      0281      END
: 221      0282      ELSE
: 222      0283      +
: 223      0284      Logical name did not translate - return 0.
: 224      0285      -
: 225      0286      RETURN OFF;
: 226      0287
: 227      0288
: 228      0289
: 229      0290      END;

```

```

.TITLE  RPG$EXT_INDS Read external RPGII indicators
.IDENT  \1-003\

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

53 44 4E 49 5F 54 58 45 24 47 50 52 00000 P.AAA: .ASCII  \RPG$EXT_INDS\

.EXTRN  LIB$STOP, OTSS$CVT_L_TI
.EXTRN  RPG$ FATINTERR, RPG$ INVARG
.EXTRN  SYS$TRNLOG

          0004 00000
          52 00000000G 00 9E 00002
          5E      FEE0 CE 9E 00009
                   04 AC D5 0000E
                   06 15 00011
          08      04 AC D1 00013
                   09 15 00017
                   00000000G 8F DD 00019 1$:
          62      01 FB 0001F
          F8 AD 010E00FF 8F D0 00022 2$:
          FC AD      10 AE 9E 0002A
          F0 AD 010E000C 8F D0 0002F
          F4 AD      BA AF 9E 00037
                   7E 7C 0003C
                   7E D4 0003E
                   F8 AD 9F 00040
                   10 AE 9F 00043
                   F0 AD 9F 00046
          00000000G 00 06 FB 00049
                   01 50 D1 00050

          .ENTRY  RPG$EXT_INDS, Save R2
          MOVAB  LIB$STOP, R2
          MOVAB  -288(SP), SP
          TSTL   NUMBER
          BLEQ   1$
          CMPL  NUMBER, #8
          BLEQ   2$
          PUSHL #RPG$ INVARG
          CALLS #1, LIB$STOP
          MOVL  #17694975, DEST_DESC
          MOVAB NAME_TEXT, DEST_DESC+4
          MOVL  #17694732, LOG_DESC
          MOVAB P.AAA, LOG_DESC+4
          CLRQ  -(SP)
          CLRL  -(SP)
          PUSHAB DEST_DESC
          PUSHAB NAME_LENGTH
          PUSHAB LOG_DESC
          CALLS #6, SYS$TRNLOG
          CMPL  R0, #1

```

04	AE	6E	3D	12	00053	BNEQ	5\$		
08	AE	010E0001	8F	9A	00055	MOVZBL	#110, NUM_STR TEXT		0263
0C	AE	04	8F	D0	0005A	MOVL	#17694721, NUM_STR		0268
		08	AE	9E	00062	MOVAB	NUM_STR_TEXT, NUM_STR+4		0271
		04	AE	9F	00067	PUSHAB	NUM_STR		0272
		04	AC	9F	0006A	PUSHAB	NUMBER		
00000000G	00		02	FB	0006D	CALLS	#2, OTSSCVT_L_TI		
	09		50	E8	C0074	BLBS	R0, 3\$		
		00000000G	8F	DD	00077	PUSHL	#RPG\$ FATINTERR		0274
	62		01	FB	0007D	CALLS	#1, LIB\$STOP		
10 AE	6E	04	AE	3A	00080	LOCC	NUM_STR_TEXT, NAME_LENGTH, NAME_TEXT		0275
			02	12	00086	BNEQ	4\$		
			51	D4	00088	CLRL	R1		
			51	D5	0008A	TSTL	R1		
			04	13	0008C	BEQL	5\$		
	50		01	D0	0008E	MOVL	#1, R0		0279
			04	00	00091	RET			0288
			50	D4	00092	CLRL	R0		
			04	00	00094	RET			0290

; Routine Size: 149 bytes, Routine Base: _RPG\$CODE + 000C

: 230 0291 1
: 231 0292 0 END ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
_RPG\$CODE	161	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	11	0	581	00:01.0
_\$255\$DUA28:[RPGRTL.OBJ]RPGLIB.L32;1	54	0	0	9	00:00.1

COMMAND QUALIFIERS

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

RPG\$EXT_INDS Read external RPGII indicators
1-003 RPG\$EXT_INDS - Read external RPGII indicators

N 13
16-Sep-1984 02:14:55
14-Sep-1984 13:04:18

VAX-11 Bliss-32 V4.0-742
[RPGRTL.SRC]RPGEXTIND.B32;1

Page 7
(4)

:)
: Size: 149 code + 12 data bytes
: Run Time: 00:05.1
: Elapsed Time: 00:19.0
: Lines/CPU Min: 3448
: Lexemes/CPU-Min: 13263
: Memory Used: 62 pages
: Compilation Complete

