

```

LLL      IIIIIIII  BBBB BBBB RRRRRRRRRR TTTTTTTTTT LLL
LLL      IIIIIIII  BBBB BBBB RRRRRRRRRR TTTTTTTTTT LLL
LLL      IIIIIIII  BBBB BBBB RRRRRRRRRR TTTTTTTTTT LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLL      III     BBB     BBB RRR           RRR       TTT       LLL
LLLLLLLLLLLLLLLLLLLL  IIIIIIII  BBBB BBBB RRRRRRRRRR TTT       LLL
LLLLLLLLLLLLLLLLLLLL  IIIIIIII  BBBB BBBB RRRRRRRRRR TTT       LLL
LLLLLLLLLLLLLLLLLLLL  IIIIIIII  BBBB BBBB RRRRRRRRRR TTT       LLL

```

3-

Sy

LI

LI
LILI
LI
LILI
LI

LI

LI
LILI
LILI
LI

LI

LI
LI

LI

LI
LI

LI

LI
LI

LI

11

LI

LI
LI

LI

LI
LI

21

LI

LI
LI

22

11

LI
LI

LI

LI
LI

LI

LL	IIIIII	BBBBBBBB	IIIIII	NN	NN	DDDDDDDD	EEEEEEEEEE	XX	XX	
LL	IIIIII	BBBBBBBB	IIIIII	NN	NN	DDDDDDDD	EEEEEEEEEE	XX	XX	
LL	II	BB	II	NN	NN	DD	DD	XX	XX	
LL	II	BB	II	NN	NN	DD	DD	XX	XX	
LL	II	BB	II	NNNN	NN	DD	DD	XX	XX	
LL	II	BB	II	NNNN	NN	DD	DD	XX	XX	
LL	II	BBBBBBBB	II	NN	NN	DD	DD	EE	EE	
LL	II	BBBBBBBB	II	NN	NN	DD	DD	EE	EE	
LL	II	BB	II	NN	NN	DD	DD	EE	EE	
LL	II	BB	II	NN	NN	DD	DD	EE	EE	
LL	II	BB	II	NN	NN	DD	DD	EE	EE	
LL	II	BB	II	NN	NN	DD	DD	EE	EE	
LL	II	BB	II	NN	NN	DD	DD	EE	EE	
LLLLLLLLLL	IIIIII	BBBBBBBB	IIIIII	NN	NN	DDDDDDDD	EEEEEEEEEE	XX	XX	
LLLLLLLLLL	IIIIII	BBBBBBBB	IIIIII	NN	NN	DDDDDDDD	EEEEEEEEEE	XX	XX	

LL	IIIIII	SSSSSSSS
LL	IIIIII	SSSSSSSS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LLLLLLLLLL	IIIIII	SSSSSSSS
LLLLLLLLLL	IIIIII	SSSSSSSS

LIB\$INDEX
Table of contents

; Return position of substring K 14

16-SEP-1984 00:10:18 VAX/VMS Macro V04-00

Page 0

(2) 48
(3) 65
(4) 96

HISTORY ; Detailed Current Edit History
DECLARATIONS
LIB\$INDEX - return relative pos. of substring

```
0000 1      .TITLE LIB$INDEX      ; Return position of substring
0000 2      .IDENT /1-005/        ; File: LIBINDEX.MAR Edit:RKRT005
0000 3
0000 4
0000 5 :*****
0000 6 :*
0000 7 :*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 :*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 :*  ALL RIGHTS RESERVED.
0000 10 :*
0000 11 :*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 :*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 :*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 :*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 :*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 :*  TRANSFERRED.
0000 17 :*
0000 18 :*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 :*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 :*  CORPORATION.
0000 21 :*
0000 22 :*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 :*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 :*
0000 25 :*
0000 26 :*****
0000 27
0000 28
0000 29 : FACILITY: General Utility Library
0000 30 :++
0000 31 : ABSTRACT:
0000 32 :
0000 33 :      Return relative position of substring in string or 0
0000 34 :--
0000 35
0000 36 : VERSION: 0
0000 37
0000 38 : HISTORY:
0000 39
0000 40 : AUTHOR:
0000 41 :      Thomas N. Hastings, 6-Aug-77: Version 0
0000 42
0000 43 : MODIFIED BY:
0000 44
0000 45 :      04 - DGP, 03-Jan-78
0000 46 :
```

LIB\$INDEX
1-005

M 14
; Return position of substring
HISTORY ; Detailed Current Edit History

16-SEP-1984 00:10:18 VAX/VMS Macro V04-00
6-SEP-1984 11:08:05 [LIBRTL.SRC]LIBINDEX.MAR;1

Page 2
(2)

```
0000 48      .SBTTL HISTORY      ; Detailed Current Edit History
0000 49
0000 50
0000 51 : Edit History for Version 0
0000 52 :
0000 53 : 04 - Remove R4 and R5 from entry mask
0000 54 : 1-001 - Update version number and copyright notice. JBS 16-NOV-78
0000 55 : 1-002 - Add "" to PSECT directive. JBS 21-DEC-78
0000 56 : 1-003 - Enhance to recognize additional classes of descriptors by
0000 57 : using LIB$ANALYZE_SDESC R3 to extract length and address of
0000 58 : first data byte. RKR 22-MAY-1981
0000 59 : 1-004 - Add special case code to process string descriptors that
0000 60 : "read" like fixed string descriptors. RKR 7-OCT-1981.
0000 61 : 1-005 - Redirect jsb's from LIB$ANALYZE_SDESC R3 to
0000 62 : LIB$ANALYZE_SDESC R2. Reorganize register usage so that
0000 63 : R6 is no longer used. RKR 18-NOV-1981
```



```
0000 65      .SBTTL  DECLARATIONS
0000 66
0000 67 :
0000 68 : INCLUDE FILES: NONE
0000 69 :
0000 70 :
0000 71 :
0000 72 : EXTERNAL SYMBOLS:
0000 73 :
0000 74      .DSABL  GBL      ; only explicit externals
0000 75      .EXTRN  LIB$ANALYZE_SDESC_R2 ; Extract length and addr of
0000 76                                     ; 1st data byte from descriptor
0000 77
0000 78 :
0000 79 : MACROS:
0000 80 :
0000 81      $DSCDEF      ; fields within a descriptor
0000 82
0000 83 :
0000 84 : PSECT DECLARATIONS:
0000 85 :
00000000 86      .PSECT  _LIB$CODE PIC,SHR,LONG,EXE,NOWRT
0000 87 :
0000 88 : EQUATED SYMBOLS: NONE
0000 89 :
0000 90 :
0000 91 :
0000 92 : OWN STORAGE: NONE
0000 93 :
0000 94
```

```
0000 96 .SBTTL LIB$INDEX - return relative pos. of substring
0000 97
0000 98 :++
0000 99 : FUNCTIONAL DESCRIPTION:
0000 100 :
0000 101 : Return the relative position of the first character of
0000 102 : sub_string in src_string or 0.
0000 103 : The value is an unsigned integer longword.
0000 104 : The relative character positions are numbered 1, 2, ....
0000 105 : thus 0 is a unique number meaning sub_string was not found.
0000 106 : The value returned is the relative position of the first
0000 107 : character of sub_string in src_string (not the last+1).
0000 108 : If both strings have zero length or sub_string has zero length
0000 109 : a 1 is returned indicating a found string. If src_string
0000 110 : has zero length and sub_string has non-zero length, a 0 is
0000 111 : returned indicating that sub_string was not found.
0000 112 :
0000 113 : CALLING SEQUENCE:
0000 114 :
0000 115 : rel_position.wlu.v = LIB$INDEX (src_string.rt.dx,
0000 116 : sub_string.rt.dx)
0000 117 :
0000 118 :
0000 119 :
0000 120 : INPUT PARAMETERS:
0000 121 :
00000004 0000 122 : SRC_STRING = 4 ; Address of source string descriptor
0000 123 : ; to be searched.
00000008 0000 124 : SUB_STRING = 8 ; Address of sub string descriptor to
0000 125 : ; be searched for.
0000 126 :
0000 127 : IMPLICIT INPUTS:
0000 128 : NONE
0000 129 :
0000 130 : OUTPUT PARAMETERS:
0000 131 : NONE
0000 132 :
0000 133 : IMPLICIT OUTPUTS:
0000 134 : NONE
0000 135 :
0000 136 : ROUTINE VALUE
0000 137 : REL_POSITION.wlu.v The resulting position
0000 138 :
0000 139 : SIDE EFFECTS:
0000 140 : NONE
0000 141 :--
0000 142
```



```
LIB$INDEX - return relative pos. of subs
003C 0000 144 .ENTRY LIB$INDEX, ^M<R2,R3,R4,R5> ; entry point / entry mask
      0002 145 ; integer enable is off for
      0002 146 ; SUBW3 (unsigned)
      0002 147
50 08 AC D0 0002 148 MOVL SUB_STRING(AP), R0 ; Address of sub_string descr.
02 03 A0 91 0006 149 CMPB DSC$B_CLASS(R0), #DSC$K_CLASS_D ; read like fixed ?
      06 1A 000A 150 BGTRU 1$ ; no
53 08 BC 7D 000C 151 MOVQ @SUB_STRING(AP), R3 ; length->R3, address-> R4
      0C 11 0010 152 BRB 2$ ; join common flow
      0012 153
00000000'GF 16 0012 154 1$: JSB G^LIB$ANALYZE_SDESC_R2 ; Extract: length->R1, addr->R2
      53 51 3C 0018 155 MOVZWL R1, R3 ; save sub_string length
      54 52 D0 001B 156 MOVL R2, R4 ; save sub_string data address
      001E 157
50 04 AC D0 001E 158 2$: MOVL SRC_STRING(AP), R0 ; Address of src_string descr.
02 03 A0 91 0022 159 CMPB DSC$B_CLASS(R0), #DSC$K_CLASS_D ; read like fixed ?
      06 1A 0026 160 BGTRU 3$ ; no
51 04 BC 7D 0028 161 MOVQ @SRC_STRING(AP), R1 ; length->R1, address->R2
      06 11 002C 162 BRB 4$ ; join common flow
      002E 163
00000000'GF 16 002E 164 3$: JSB G^LIB$ANALYZE_SDESC_R2 ; Extract: length->R1, addr->R2
      0034 165
55 51 53 A3 0034 166 4$: SUBW3 R3, R1, R5 ; R5 = unsigned difference in
      0038 167 ; length of src minus sub
      0038 168
62 51 64 53 39 0038 169 MATCHC R3, (R4), R1, (R2) ; Find sub_string in src_string
      003D 170 ; State of regs after a MATCHC instr.
      003D 171 ; R0 = If match occurred, 0,
      003D 172 ; else number of bytes in object
      003D 173 ; string.
      003D 174 ; R1 = If match occurred, the address of
      003D 175 ; one byte beyond the object
      003D 176 ; string,
      003D 177 ; else address of the object string
      003D 178 ; R2 = If match occurred, the number of
      003D 179 ; bytes remaining at the source
      003D 180 ; string after the match,
      003D 181 ; else 0
      003D 182 ; R3 = If match occurred, the address of
      003D 183 ; one byte beyond the last byte
      003D 184 ; matched,
      003D 185 ; else address of one byte beyond
      003D 186 ; the source string.
      003D 187
      003D 188 BNEQ 10$ ; branch if no match found
50 09 12 003D 188 MOVZWL R5, R0 ; R0 = len src - len sub
50 55 3C 003F 189 SUBL R2, R0 ; R0 = len src - len sub - # char left
      52 C2 0042 190 ; to scan = rel. pos. of 1st char of sub
      0045 191 ; wants 1-origin, not 0-origin position.
50 D6 0045 192 INCL R0 ; R0 = relative position
      04 0047 193 RET
      0048 194
      0048 195
      0048 196 ;+
      0048 197 ; Here when sub_string not found
      0048 198 ;+
      0048 199
50 D4 0048 200 10$: CLRL R0 ; R0 = 0 to indicate not found
```


LIB\$INDEX
1-005

LIB\$INDEX - return position of substring
LIB\$INDEX - return relative pos. of subs

04	004A	201	RET	
	004B	202		
	004B	203	.END	

D 15 16-SEP-1984 00:10:18 VAX/VMS Macro V04-00
6-SEP-1984 11:08:05 [LIBRTL.SRC]LIBINDEX.MAR;1

; return
; End of module LIB\$INDEX

Page 6
(5)

LIB\$INDEX
Symbol table

; Return position of substring

E 15

16-SEP-1984 00:10:18
6-SEP-1984 11:08:05

VAX/VMS Macro V04-00
[LIBRTL.SRC]LIBINDEX.MAR;1

Page 7
(5)

DSC\$B_CLASS = 00000003
DSC\$K_CLASS_D = 00000002
LIB\$ANALYZE_SDESC_R2 ***** X 00
LIB\$INDEX 00000000 RG 02
SRC_STRING = 00000004
SUB_STRING = 00000009

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR
\$AB\$\$	00000000 (0.)	01 (1.)	NOPIC USR
_LIB\$CODE	0000004B (75.)	02 (2.)	PIC USR

CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
CON REL LCL SHR EXE RD NOWRT NOVEC LONG

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.03	00:00:03.14
Command processing	108	00:00:00.31	00:00:03.20
Pass 1	132	00:00:01.13	00:00:06.57
Symbol table sort	0	00:00:00.10	00:00:00.11
Pass 2	50	00:00:00.35	00:00:01.92
Symbol table output	2	00:00:00.01	00:00:00.41
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	325	00:00:01.96	00:00:15.37

The working set limit was 1050 pages.
8502 bytes (17 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 135 non-local and 5 local symbols.
203 source lines were read in Pass 1, producing 13 object records in Pass 2.
8 pages of virtual memory were used to define 7 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4

190 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:LIBINDEX/OBJ=OBJ\$:LIBINDEX MSRC\$:LIBINDEX/UPDATE=(ENH\$:LIBINDEX)

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

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

