


```

LL      IIIIII  BBBB8888  LL      000000  CCCCCCCC  CCCCCCCC
LL      IIIIII  BBBB8888  LL      000000  CCCCCCCC  CCCCCCCC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BBBB8888  LL      00      00  CC      CC
LL      II      BBBB8888  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      II      BB      BB  LL      00      00  CC      CC
LL      IIIIII  BBBB8888  LLLLLLLLLL 000000  CCCCCCCC  CCCCCCCC
LL      IIIIII  BBBB8888  LLLLLLLLLL 000000  CCCCCCCC  CCCCCCCC

```

```

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      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS

```



(2) 58
(3) 86

DECLARATIONS
LIB\$LOCC - locate a character

L1
1-
.....

```
0000 1 .TITLE LIB$LOCC - locate a character in a character string
0000 2 .IDENT 1-005/ ; File: LIBLOCC.MAR Edit:RKR1005
0000 3
0000 4
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 9 * ALL RIGHTS RESERVED. *
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 16 * TRANSFERRED. *
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 20 * CORPORATION. *
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 24 *
0000 25 *
0000 26 *****
0000 27
0000 28
0000 29 ++
0000 30 FACILITY: General Utility Library
0000 31
0000 32 ABSTRACT:
0000 33
0000 34 return the relative position of the matching character or zero.
0000 35
0000 36 ENVIRONMENT: User Mode, AST Reentrant
0000 37
0000 38 --
0000 39 AUTHOR: Donald G. Petersen, CREATION DATE: 03-Jan-78
0000 40
0000 41 MODIFIED BY:
0000 42
0000 43 DGP, 03-Jan-78 : VERSION 00
0000 44 01 - Original
0000 45 00-02 - DGP 06-Jan-78 - Relative position not found properly
0000 46 1-001 - Update version number and copyright notice. JBS 16-NOV-78
0000 47 1-002 - Add "" to the PSECT directive. JBS 21-DEC-78
0000 48 1-003 - Enhance to recognize additional classes of string descriptors
0000 49 by invoking LIB$ANALYZE_SDESC_R3 to extract length and
0000 50 address of 1st data byte of string. RKR 22-MAY-1981
0000 51 1-004 - Add special-case code to process string descriptors that
0000 52 "read" like fixed string descriptors. RKR 7-OCT-1981.
0000 53 1-005 - Redirect jsb's from LIB$ANALYZE_SDESC_R3 to
0000 54 LIB$ANALYZE_SDESC_R2. Reorganize register usage so that
0000 55 R5 is no longer needed and can be dropped from entry mask.
0000 56 RKR 18-NOV-1981
```

```
0000 58      .SBTTL  DECLARATIONS
0000 59      :
0000 60      : INCLUDE FILES: NONE
0000 61      :
0000 62      :
0000 63      : EXTERNAL SYMBOLS
0000 64      :   .DSABL  GBL           ; Only explicit externals
0000 65      :   .EXTRN  LIB$ANALYZE_SDESC_R2 ; Extract length and address of
0000 66      :                                           ; 1st byte of data
0000 67      :
0000 68      : MACROS:
0000 69      :
0000 70      :   $DSCDEF           ; fields in a descriptor
0000 71      :
0000 72      :
0000 73      : EQUATED SYMBOLS: NONE
0000 74      :
0000 75      :
0000 76      :
0000 77      : OWN STORAGE: NONE
0000 78      :
0000 79      :
0000 80      :
0000 81      : PSECT DECLARATIONS:
0000 82      :
00000000 83      :   .PSECT _LIB$CODE PIC, SHR, LONG, EXE, NOWRT
0000 84      :
```

```

0000 86      .SBTTL LIB$LOCC - locate a character
0000 87      :++
0000 88      : FUNCTIONAL DESCRIPTION:
0000 89      :
0000 90      : The character is compared with successive bytes of the string
0000 91      : specified by the string descriptor until an equality is found
0000 92      : or the string is exhausted.
0000 93      : The relative position of the unequal byte, or zero is returned.
0000 94      : If the string has a zero length then a zero is returned.
0000 95      :
0000 96      : CALLING SEQUENCE:
0000 97      :
0000 98      : index.wlu.v = LIB$LOCC (char.rt.dx, string.rt.dx)
0000 99      :
0000 100     :
0000 101     : INPUT PARAMETERS:
0000 102     :
0000 103     : CHAR = 4 ; Adr of character desc
00000004 0000 104     : STRING = 8 ; Adr of string desc
00000008 0000 105     :
0000 106     : IMPLICIT INPUTS:
0000 107     :
0000 108     : NONE
0000 109     :
0000 110     : OUTPUT PARAMETERS:
0000 111     :
0000 112     : NONE
0000 113     :
0000 114     : IMPLICIT OUTPUTS:
0000 115     :
0000 116     : NONE
0000 117     :
0000 118     : FUNCTION VALUE:
0000 119     :
0000 120     : index.wlu.v
0000 121     :
0000 122     : SIDE EFFECTS:
0000 123     :
0000 124     : NONE
0000 125     :
0000 126     :--
0000 127     :
0000 128     : .ENTRY LIB$LOCC , ^M<R2, R3, R4> ; Entry point
50 04 AC D0 0002 129     : MOVL CHAR(AP), R0 ; Address of char string descr.
02 03 A0 91 0006 130     : CMPB DSC$B_CLASS(R0), #DSC$K_CLASS_D ; read like fixed ?
0000 131     : BGTU 1$ ; no
53 04 A0 D0 000C 132     : MOVL DSC$A_POINTER(R0), R3 ; address of char
0000 133     : BRB 2$ ; join common flow
0000 134     :
00000000'GF 16 0012 135 1$ : JSB G^LIB$ANALYZE_SDESC_R2 ; Extract: length->R1, addr->R2
53 52 D0 0018 136     : MOVL R2, R3 ; save addr, length is assumed
0000 137     : ; to be one
0000 138     :
50 08 AC D0 001B 139 2$ : MOVL STRING(AP), R0 ; Address of string descriptor
02 03 A0 91 001F 140     : CMPB DSC$B_CLASS(R0), #DSC$K_CLASS_D ; read like fixed ?
0000 141     : BGTRU 3$ ; no
54 60 3C 0025 142     : MOVZWL DSC$W_LENGTH(R0), R4 ; length of string

```

```

52 04 A0 D0 0028 143      MOVL   DSC$A_POINTER(R0), R2      ; address of string
      09 11 002C 144      BRB    4$                          ; join common flow
00000000'GF 16 002E 145      JSB    G^LIB$ANALYZE_SDESC_R2      ; Extract: length->R1, addr->R2
      54 51 3C 0034 146 3$:  MOVZWL R1, R4                      ; save length of string
62 54 63 3A 0037 148      LOCC   (R3), R4, (R2)              ; locate a character
      003B 149 4$:
      003B 150              ; State of regs after a LOCC instr.
      003B 151              ; R0 = If found, number of bytes
      003B 152              ; remaining in string (including
      003B 153              ; located one),
      003B 154              ; else 0.
      003B 155              ; R1 = if found, address of byte found,
      003B 156              ; else address of 1 byte beyond
      003B 157              ; string.
      003B 158
      003B 159
50 54 06 13 003B 160      BEQL   10$                          ; branch if none found
      50 A3 003D 161      SUBW3  R0, R4, R0                    ; get relative position
      50 D6 0041 162      INCL   R0                          ; one origin
      04 0043 163 10$:  RET
      0044 164      .END

```

LIB\$LOCC
Symbol table

I 6
- locate a character in a character stri 16-SEP-1984 00:13:21 VAX/VMS Macro V04-00
6-SEP-1984 11:08:40 [LIBRTL.SRC]LIBLOCC.MAR;1

Page 5
(3)

CHAR = 00000004
DSCSA_POINTER = 00000004
DSCSB_CLASS = 00000003
DSCSK_CLASS_D = 00000002
DSCSW_LENGTH = 00000000
LIB\$ANALYZE_SDESC_R2 ***** X 00
LIB\$LOCC 00000000 RG 02
STRING = 00000008

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
_LIB\$CODE	00000044 (68.)	02 (2.)	PIC USR CGN REL LCL SHR EXE RD NOWRT NOVEC LONG

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	32	00:00:00.04	00:00:01.89
Command processing	112	00:00:00.30	00:00:01.91
Pass 1	132	00:00:01.07	00:00:05.07
Symbol table sort	0	00:00:00.11	00:00:00.43
Pass 2	44	00:00:00.31	00:00:02.70
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	3	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	327	00:00:01.86	00:00:12.04

The working set limit was 900 pages.
8086 bytes (16 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.
164 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\$:LIBLOCC/OBJ=OBJ\$:LIBLOCC MSRC\$:LIBLOCC/UPDATE=(ENH\$:LIBLOCC)

