```
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  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        BBBBBBBBBBBB     RRRRRRRRRRRR           TTT        LLL
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           TTT        LLL
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           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
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
```

```
LL              IIIIII    BBBBBBBB      SSSSSSSS  PPPPPPPP      AAAAAA    NN      NN    CCCCCCCC
LL              IIIIII    BBBBBBBB      SSSSSSSS  PPPPPPPP      AAAAAA    NN      NN    CCCCCCCC
LL                II      BB      BB  SS          PP      PP  AA      AA  NN      NN  CC
LL                II      BB      BB  SS          PP      PP  AA      AA  NN      NN  CC
LL                II      BB      BB  SS          PP      PP  AA      AA  NNNN    NN  CC
LL                II      BB      BB  SS          PP      PP  AA      AA  NNNN    NN  CC
LL                II      BBBBBBBB      SSSSSS    PPPPPPPP    AA      AA  NN  NN  NN  CC
LL                II      BBBBBBBB      SSSSSS    PPPPPPPP    AA      AA  NN  NN  NN  CC
LL                II      BB      BB          SS  PP          AAAAAAAAAA  NN    NNNN  CC
LL                II      BB      BB          SS  PP          AAAAAAAAAA  NN    NNNN  CC
LL                II      BB      BB          SS  PP          AA      AA  NN      NN  CC
LL                II      BB      BB          SS  PP          AA      AA  NN      NN  CC
LLLLLLLLLL      IIIIII    BBBBBBBB    SSSSSSSS  PP          AA      AA  NN      NN    CCCCCCCC
LLLLLLLLLL      IIIIII    BBBBBBBB    SSSSSSSS  PP          AA      AA  NN      NN    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                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  LIB$SPANC - Scan for character
0000     2          .IDENT  /1-006/         ; File: LIBSPANC.MAR Edit: RKR1006
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 an index into the source string of the satisfying
0000    35 ;       character or zero
0000    36 ;
0000    37 ; ENVIRONMENT: User Mode, AST Reentrant
0000    38 ;
0000    39 ;--
0000    40 ; AUTHOR: Donald G. Petersen, CREATION DATE: 03-Jan-78
0000    41 ;
0000    42 ; MODIFIED BY:
0000    43 ;
0000    44 ;       DGP, 03-Jan-78 : VERSION 00
0000    45 ; 01    - Original
0000    46 ; 00-02 - DGP 05-Jan-78 - table.rbu.ra for table.rt.dx
0000    47 ; 00-03 - DGP 18-Jan-78 - remove one instruction
0000    48 ; 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
0000    49 ; 1-002 - Add "," to the PSECT directive.  JBS 21-DEC-78
0000    50 ; 1-003 - Fix entry mask so R2 and R3, which get clobbered, are saved.
0000    51 ;         SBL 02-Feb-79
0000    52 ; 1-004 - Enhance to recognize additional classes of string descriptors
0000    53 ;         by invoking LIB$ANLAYZE_SDESC_R3 to extract length and
0000    54 ;         address of 1st data byte of string.  RKR 26-MAY-1981.
0000    55 ; 1-005 - Add special-case code to process string descriptors that
0000    56 ;         "read" like fixed string descriptors.  RKR 7-OCT-1981
0000    57 ; 1-006 - Redirect jsb's from LIB$ANALYZE_SDESC_R3 to
```

```
0000     58 ;        LIB$ANALYZE_SDESC_R2.  RKR 18-NOV-1981.
```

```
                        .SBTTL  DECLARATIONS
                ;
                ; INCLUDE FILES: NONE
                ;

                ; EXTERNAL SYMBOLS
                        .DSABL  GBL                             ; Only explicit externals
                        .EXTRN  LIB$ANALYZE_SDESC_R2            ; Extract length and address of
                                                                ; 1st data byte of string
                ;
                ; MACROS:
                ;
                        $DSCDEF         ; fields within a descriptor

                ;
                ; EQUATED SYMBOLS: NONE
                ;

                ;
                ; OWN STORAGE: NONE
                ;

                ;
                ; PSECT DECLARATIONS:
                ;
00000000                .PSECT _LIB$CODE PIC, SHR, LONG, EXE, NOWRT
```

```
                        0000     88          .SBTTL  LIB$SPANC  - Span characters
                        0000     89 ;++
                        0000     90 ; FUNCTIONAL DESCRIPTION:
                        0000     91 ;
                        0000     92 ;       The bytes of the string specified by the source descriptor are
                        0000     93 ;       successively used to index into a the table.
                        0000     94 ;       The byte selected from the table is ANDed with the mask byte
                        0000     95 ;       specified by the mask descriptor.  The operation continues
                        0000     96 ;       until the result of the AND is zero.  The relative position of
                        0000     97 ;       the character in the source string which terminated the
                        0000     98 ;       operation is returned if one is found.  Otherwise, zero
                        0000     99 ;       is returned.  If the source string has a length of zero then a
                        0000    100 ;       zero is returned.
                        0000    101 ;
                        0000    102 ; CALLING SEQUENCE:
                        0000    103 ;
                        0000    104 ;       index.wlu.v = LIB$SPANC (src.rt.dx, table.rbu.ra, mask.rbu.r)
                        0000    105 ;
                        0000    106
                        0000    107 ; INPUT PARAMETERS:
                        0000    108 ;
00000004                0000    109         SOURCE = 4                      ; Adr of source string desc
00000008                0000    110         TABLE = 8                       ; Adr of base of table
0000000C                0000    111         MASK = 12                       ; Adr of mask byte
                        0000    112 ;
                        0000    113 ; IMPLICIT INPUTS:
                        0000    114 ;
                        0000    115 ;       NONE
                        0000    116 ;
                        0000    117 ; OUTPUT PARAMETERS:
                        0000    118 ;
                        0000    119 ;       NONE
                        0000    120 ;
                        0000    121 ; IMPLICIT OUTPUTS:
                        0000    122 ;
                        0000    123 ;       NONE
                        0000    124 ;
                        0000    125 ; FUNCTION VALUE:
                        0000    126 ;
                        0000    127 ;       index.wlu.v
                        0000    128 ;
                        0000    129 ; SIDE EFFECTS:
                        0000    130 ;
                        0000    131 ;       NONE
                        0000    132 ;
                        0000    133 ;--
                        0000    134
                001C    0000    135         .ENTRY LIB$SPANC , ^M<R2, R3, R4>; Entry point
 50    04 AC    D0      0002    136         MOVL    SOURCE(AP), R0          ; Address of source descriptor
 02    03 A0    91      0006    137         CMPB    DSC$B_CLASS(R0), #DSC$K_CLASS_D ; read like fixed ?
          06    1A      000A    138         BGTRU   1$                      ; no
 51    04 BC    7D      000C    139         MOVQ    @SOURCE(AP), R1         ; length->R1, address->R2
          06    11      0010    140         BRB     2$                      ; join common flow
                        0012    141
00000000'GF    16      0012    142 1$:     JSB     G^LIB$ANALYZE_SDESC_R2  ; Extract: length->R1, addr->R2
    54    51    3C      0018    143 2$:     MOVZWL  R1, R4                  ; save length of source string
                        001B    144
```

```
0C BC   08 BC   62   54   2B  001B          SPANC   R4, (R2), @TABLE(AP), @MASK(AP) ; spanc character
                               0022                                          ; State of regs after a SPANC instr.
                               0022                                          ; R0 = number of bytes remaining in
                               0022                                          ;      source string (including the byte
                               0022                                          ;      which produced the zero AND)
                               0022                                          ;      Is zero if no character found.
                               0022                                          ; R1 = If found, address of byte which
                               0022                                          ;      produced zero AND,
                               0022                                          ;      else address of 1 byte beyond
                               0022                                          ;      source string
                               0022                                          ; R2 = 0
                               0022                                          ; R3 = address of table
                               0022                                          ;
                               0022
                     06   13   0022          BEQL    10$                     ; branch if no character found
          50   54    50   A3   0024          SUBW3   R0, R4, R0              ; get position of character
                     50   D6   0028          INCL    R0                      ; one origin
                          04   002A  10$:    RET
                               002B          .END
```

```
DSC$B_CLASS                     = 00000003
DSC$K_CLASS_D                   = 00000002
LIB$ANALYZE_SDESC_R2              ********  X   00
LIB$SPANC                         00000000 RG    02
MASK                            = 0000000C
SOURCE                          = 00000004
TABLE                           = 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
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
_LIB$CODE	0000002B (43.)	02 (2.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.04	00:00:01.88
Command processing	109	00:00:00.31	00:00:02.59
Pass 1	132	00:00:01.13	00:00:05.40
Symbol table sort	0	00:00:00.10	00:00:00.94
Pass 2	44	00:00:00.32	00:00:02.46
Symbol table output	3	00:00:00.01	00:00:00.01
Psect synopsis output	2	00:00:00.01	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	321	00:00:01.92	00:00:13.30

```
The working set limit was 1050 pages.
7912 bytes (16 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 136 non-local and 3 local symbols.
163 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
_255DUA28:[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$:LIBSPANC/OBJ=OBJ$:LIBSPANC MSRC$:LIBSPANC/UPDATE=(ENH$:LIBSPANC)
```

LIBSPAWN
LIS

LIBSTATVM
LIS

LIBTRAA2E
LIS

LIBSPANC
LIS

LIBSYMBOL
LIS

LIBTRNLOG
LIS

LIBSKPC
LIS

LIBTIMER
LIS

LIBTPARSE
LIS

LIBTRIMFI
LIS

LIBSTRRET
LIS

LIBTRAE2A
LIS