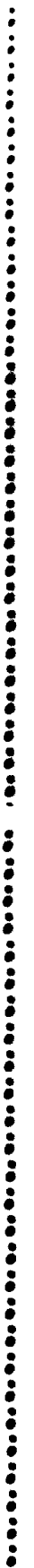



```
SSSSSSSS TTTTTTTTT RRRRRRRR MM MM 000000 VV VV QQQQQQ
SSSSSSSS TTTTTTTTT RRRRRRRR MM MM 000000 VV VV QQQQQQ
SS TT RR RR MMMM MMMM 00 00 VV VV QQ QQ
SS TT RR RR MMMM MMMM 00 00 VV VV QQ QQ
SS TT RR RR MM MM 00 00 VV VV QQ QQ
SSSSSS TT RRRRRRRR MM MM 00 00 VV VV QQ QQ
SSSSSS TT RRRRRRRR MM MM 00 00 VV VV QQ QQ
SS TT RR RR MM MM 00 00 VV VV QQ QQ
SS TT RR RR MM MM 00 00 VV VV QQ QQ
SS TT RR RR MM MM 00 00 VV VV QQ QQ
SSSSSS TT RR RR MM MM 000000 VV VV QQQQ QQ
SSSSSS TT RR RR MM MM 000000 VV VV QQQQ QQ
.....
.....
.....
.....
```

```
LL IIIIII SSSSSSS
LL IIIIII SSSSSSS
LL II SS
LL II SS
LL II SS
LL II SSSSSS
LL II SSSSSS
LL II SS
LL II SS
LL II SS
LLLLLLLLLL IIIIII SSSSSSS
LLLLLLLLLL IIIIII SSSSSSS
```

1
.....

(2)	54	DECLARATIONS
(3)	85	STR\$\$MOVQ_R1 -



```

0000 1      .TITLE STR$$MOVQ      Move a quadword monolithically
0000 2      .IDENT /1-003/      ; File: STRMOVQ.MAR
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: String Package
0000 31 :
0000 32 : ABSTRACT: This routine exists to allow STR$ routines written in BLISS
0000 33 :           to use the VAX instruction MOVQ so that the address and length
0000 34 :           in a descriptor can be updated simultaneously, so that at AST
0000 35 :           level the user will get either the entire new string or the
0000 36 :           entire old string.
0000 37 :
0000 38 :
0000 39 : ENVIRONMENT: User Mode, AST Reentrant
0000 40 :
0000 41 : --
0000 42 : AUTHOR: R. Will,      CREATION DATE: 9-Feb-79
0000 43 :
0000 44 : MODIFIED BY:
0000 45 :
0000 46 :           R. Will, 9-Feb-79: VERSION 01
0000 47 : 1-001 - Original
0000 48 : 1-002 - Minor edit to the TITLE and IDENT lines to conform to the
0000 49 :           RTL standard format. JBS 01-MAR-1979
0000 50 : 1-003 - Make this a double dollar sign module. JBS 25-JUL-1979
0000 51 :
0000 52 : --

```

```
0000 54      .SBTTL  DECLARATIONS
0000 55      :
0000 56      : INCLUDE FILES:
0000 57      :
0000 58      :
0000 59      :
0000 60      : EXTERNAL DECLARATIONS:
0000 61      :
0000 62      :      .DSABL  GBL                ; Prevent undeclared
0000 63      :                                          ; symbols from being
0000 64      :                                          ; automatically global.
0000 65      :
0000 66      :
0000 67      : MACROS:
0000 68      :
0000 69      :
0000 70      :
0000 71      : EQUATED SYMBOLS:
0000 72      :
0000 73      :
0000 74      :
0000 75      : OWN STORAGE:
0000 76      :
0000 77      :
0000 78      :
0000 79      : PSECT DECLARATIONS:
0000 80      :
00000000 81      .PSECT _STR$CODE PIC,USR,CON,REL,LCL,SHR,-
0000 82      EXE, RD, NOWRT, LONG
0000 83
```

```
0000 85 .SBTTL STR$$MOVQ_R1 -
0000 86 :++
0000 87 : FUNCTIONAL DESCRIPTION:
0000 88 :
0000 89 : This routine executes the VAX instruction MOVQ.
0000 90 :
0000 91 : CALLING SEQUENCE:
0000 92 :
0000 93 : JSB STR$$MOVQ_R1
0000 94 :
0000 95 : INPUT PARAMETERS:
0000 96 :
0000 97 : NONE
0000 98 :
0000 99 : IMPLICIT INPUTS:
0000 100 :
0000 101 : R0 contains the address of the new descriptor data
0000 102 : R1 contains the address of the descriptor to be modified
0000 103 :
0000 104 : OUTPUT PARAMETERS:
0000 105 :
0000 106 : NONE
0000 107 :
0000 108 : IMPLICIT OUTPUTS:
0000 109 :
0000 110 : NONE
0000 111 :
0000 112 : FUNCTION VALUE:
0000 113 : COMPLETION CODES:
0000 114 :
0000 115 : NONE
0000 116 :
0000 117 : SIDE EFFECTS:
0000 118 :
0000 119 : NONE
0000 120 :
0000 121 : --
0000 122 :
0000 123 STR$$MOVQ_R1:: : Entry point
0000 124 :
61 60 7D 0000 125 MOVQ (R0), (R1) : Do the MOVQ
05 0003 126 RSB
0004 127
0004 128 .END
```

STR\$\$MOVQ
Symbol table

Move a quadword monolithically

E 14

16-SEP-1984 00:35:30
6-SEP-1984 11:18:05

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

Page 4
(3)

STR\$\$MOVQ_R1 00000000 RG 01

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
_STR\$CODE	00000004 (4.)	01 (1.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.03	00:00:01.67
Command processing	122	00:00:00.34	00:00:01.37
Pass 1	66	00:00:00.25	00:00:01.44
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	37	00:00:00.19	00:00:01.85
Symbol table output	2	00:00:00.00	00:00:00.00
Psect synopsis output	2	00:00:00.02	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	260	00:00:00.83	00:00:06.35

The working set limit was 750 pages.
 1104 bytes (3 pages) of virtual memory were used to buffer the intermediate code.
 There were 10 pages of symbol table space allocated to hold 1 non-local and 0 local symbols.
 128 source lines were read in Pass 1, producing 8 object records in Pass 2.
 0 pages of virtual memory were used to define 0 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

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

