

(2) 47
(3) 77

DECLARATIONS
LIBSCALLG - Execute CALLG instruction

```
0000 1 .TITLE LIBSCALLG - Execute CALLG instruction
0000 2 .IDENT /1-002/ ; File: LIBSCALLG.MAR Edit: SBL1002
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 This module contains LIBSCALLG, which makes the VAX CALLG instruction
0000 35 available as a callable procedure.
0000 36
0000 37 ENVIRONMENT: Runs at any access mode, AST Reentrant
0000 38
0000 39 AUTHOR: Steven B. Lionel, CREATION DATE: 28-October-1981
0000 40
0000 41 MODIFIED BY:
0000 42
0000 43 1-001 - Original. SBL 28-October-1981
0000 44 1-002 - Inherit caller's FU, DV, IV PSW bits. SBL 9-Nov-1981
0000 45 --
```

```
0000 47      .SBTTL  DECLARATIONS
0000 48      :
0000 49      : LIBRARY MACRO CALLS:
0000 50      :
0000 51      $PSLDEF      ; Processor Status Longword definitions
0000 52      $SFDEF      ; Stack Frame definitions
0000 53      :
0000 54      : EXTERNAL DECLARATIONS:
0000 55      :
0000 56      .DSABL  GBL      ; Force all external symbols to be declared
0000 57      NONE
0000 58      :
0000 59      MACROS:
0000 60      :
0000 61      NONE
0000 62      :
0000 63      : EQUATED SYMBOLS:
0000 64      :
0000 65      NONE
0000 66      :
0000 67      : OWN STORAGE:
0000 68      :
0000 69      NONE
0000 70      :
0000 71      : PSECT DECLARATIONS:
0000 72      :
00000000 73      .PSECT  _LIB$CODE PIC,  USR,  CON,  REL,  LCL,  SHR,  -
0000 74      EXE,  RD,  NOWRT,  LONG
0000 75
```

```

0000 77 .SBTTL LIB$CALLG - Execute CALLG instruction
0000 78 :++
0000 79 : FUNCTIONAL DESCRIPTION:
0000 80 :
0000 81 : LIB$CALLG makes the VAX CALLG instruction available as
0000 82 : a callable procedure. It calls a procedure with an
0000 83 : argument list specified as an array.
0000 84 :
0000 85 : For more information, see the VAX-11 Architecture Handbook.
0000 86 :
0000 87 : CALLING SEQUENCE:
0000 88 :
0000 89 : status.wx.v = LIB$CALLG (arg_list.rl.ra, procedure.fzem.r)
0000 90 :
0000 91 : FORMAL PARAMETERS:
0000 92 :
00000004 0000 93 : arg_list = 4 ; An array of longwords containing the argument
0000 94 : list for the called procedure. Passed by
0000 95 : reference. The first byte of the first
0000 96 : longword must contain the count of longwords
0000 97 : to follow; the remaining three bytes of the
0000 98 : first longword must be zero.
00000008 0000 99 :
0000 100 : procedure = 8 ; The entry mask of the procedure to call.
0000 101 : Passed by reference.
0000 102 :
0000 103 :
0000 104 : IMPLICIT INPUTS:
0000 105 :
0000 106 : The caller's PSW bits FU, IV and DV are inherited for the call
0000 107 : to the specified procedure.
0000 108 :
0000 109 : IMPLICIT OUTPUTS:
0000 110 :
0000 111 : NONE
0000 112 :
0000 113 : FUNCTION VALUE:
0000 114 :
0000 115 : The contents of registers R0 and R1 returned by the called procedure
0000 116 : are returned unchanged to the caller of LIB$CALLG. Therefore, the
0000 117 : function value of LIB$CALLG is the same as that of the called function.
0000 118 :
0000 119 : SIDE EFFECTS:
0000 120 :
0000 121 : NONE
0000 122 :
0000 123 : --
0000 124 :
0000 125 : .ENTRY LIB$CALLG, ^M<> ; Entry point
50 04 AD FF1F 8F AB 0002 126
0002 127 BICW3 #^C<PSLSM_IV!PSLSM_DV!PSLSM_FU>, SFSW SAVE PSW(FP), -
0009 128 R0 ; Put in R0 the caller's PSW
0009 129 ; with only the IV, DV and FU
0009 130 ; bits copied.
0009 131 BISPSW R0 ; Set those bits in our PSW
08 BC 04 BC 50 B8 0009 132 CALLG @arg_list(AP), @procedure(AP) ; Call procedure
04 0010 133 RET ; return to caller

```


LIB\$CALLG
Symbol table

- Execute CALLG instruction

D 15

15-SEP-1984 23:48:59
6-SEP-1984 11:03:47

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

Page 5
(3)

ARG_LIST	=	00000004		
LIB\$CALLG	=	00000000	RG	02
PROCEDURE	=	00000008		
PSL\$M_DV	=	00000080		
PSL\$M_FU	=	00000040		
PSL\$M_IV	=	00000020		
SFSW_SAVE_PSW	=	00000004		

! 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	00000011 (17.)	02 (2.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.04	00:00:00.80
Command processing	117	00:00:00.33	00:00:02.85
Pass 1	122	00:00:00.84	00:00:05.76
Symbol table sort	0	00:00:00.03	00:00:00.19
Pass 2	41	00:00:00.26	00:00:01.56
Symbol table output	2	00:00:00.02	00:00:00.15
Psect synopsis output	3	00:00:00.03	00:00:00.31
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	317	00:00:01.56	00:00:11.63

The working set limit was 1050 pages.
5049 bytes (10 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 71 non-local and 0 local symbols.
135 source lines were read in Pass 1, producing 12 object records in Pass 2.
9 pages of virtual memory were used to define 8 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	5

131 GETS were required to define 5 macros.

There were no errors, warnings or information messages.

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

