

IIIIIIIIII	00000000	SSSSSSSSSS	UUU	UUU	PPPPPPPPPP
IIIIIIIIII	00000000	SSSSSSSSSS	UUU	UUU	PPPPPPPPPP
IIIIIIIIII	00000000	SSSSSSSSSS	UUU	UUU	PPPPPPPPPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
III	000	SSS	UUU	UUU	PPP
IIIIIIIIII	00000000	SSSSSSSSSS	UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	PPP
IIIIIIIIII	00000000	SSSSSSSSSS	UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	PPP
IIIIIIIIII	00000000	SSSSSSSSSS	UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	PPP

```

LL      AAAAAA  SSSSSSSS  NN      NN  DDDDDDDD  LL      DDDDDDDD  RRRRRRRR  QQQQQQ
LL      AAAAAA  SSSSSSSS  NN      NN  DDDDDDDD  LL      DDDDDDDD  RRRRRRRR  QQQQQQ
LL      AA      AA  SS      SS  NN      NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SS      SS  NN      NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SS      SS  NNNN     NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SS      SS  NNNN     NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SSSSSS  SS  NN      NN  NN  DD      DD  LL      DD      DD  RRRRRRRR  QQ      QQ
LL      AA      AA  SSSSSS  SS  NN      NN  NN  DD      DD  LL      DD      DD  RRRRRRRR  QQ      QQ
LL      AAAAAAAAAA  SS  NN      NN  NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AAAAAAAAAA  SS  NN      NN  NNNN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SS      SS  NN      NN  NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LL      AA      AA  SS      SS  NN      NN  NN  DD      DD  LL      DD      DD  RR      RR  QQ      QQ
LLLLLLLLLL  AA      AA  SSSSSSSS  NN      NN  DDDDDDDD  LLLLLLLLLL  DDDDDDDD  RR      RR  QQQQ  QQ
LLLLLLLLLL  AA      AA  SSSSSSSS  NN      NN  DDDDDDDD  LLLLLLLLLL  DDDDDDDD  RR      RR  QQQQ  QQ

```

```

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

```

LPA\$SNDLDRQ  
Table of contents

- SEND LOAD REQUEST

6 9

16-SEP-1984 01:45:00 VAX/VMS Macro V04-00

Page 0

LPI  
VOI

(2) 46  
(3) 92

DECLARATIONS  
LPA\$SNDLDRQ - SEND REQUEST TO LOADER PROCESS

```
0000 1 .TITLE LPA$SNDLDRQ - SEND LOAD REQUEST
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6 *****
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 9 * ALL RIGHTS RESERVED. *
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 16 * TRANSFERRED. *
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 20 * CORPORATION. *
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 24 *
0000 25 *
0000 26 *****
0000 27 *****
0000 28
0000 29 ++
0000 30 : FACILITY: LPA-11 PROCEDURE LIBRARY
0000 31 :
0000 32 : ABSTRACT:
0000 33 : THIS ROUTINE SENDS A LOAD REQUEST TO THE LPA-11 MICROCODE
0000 34 : LOADER PROCESS AND RECEIVES STATUS OVER A TEMPORARY MAILBOX
0000 35 :
0000 36 : ENVIRONMENT: USER MODE
0000 37 :
0000 38 : AUTHOR: STEVE BECKHARDT, CREATION DATE: 8-OCT-78
0000 39 :
0000 40 : MODIFIED BY:
0000 41 :
0000 42 : V03-001 SBL3001 Steven B. Lionel 30-Mar-1982
0000 43 : Change module name to LPA$SNDLDRQ.
0000 44 :--
```

```

0000 46      .SBTTL  DECLARATIONS
0000 47      :
0000 48      : INCLUDE FILES:
0000 49      :
0000 50      $DIBDEF      ; DIB OFFSETS
0000 51      $IODEF      ; I/O FUNCTION CODES
0000 52
0000 53      :
0000 54      : MACROS:
0000 55      :
0000 56
0000 57      :
0000 58      : EQUATED SYMBOLS:
0000 59      :
0000 60      : OFFSETS FROM STACK FRAME FOR TEMPORARY STORAGE
0000 61
00000070 0000 62 WRKSPACE = 112      ; AMOUNT OF WORKSPACE TO ALLOCATE
0000 63
FFFFFF90 0000 64 IOSB = -112      ; I/O STATUS BLOCK
FFFFFF98 0000 65 SMBCHAN = -104      ; SEND MAILBOX CHANNEL NUMBER
FFFFFF9A 0000 66 RMBCHAN = -102      ; RECEIVE MAILBOX CHANNEL NUMBER
FFFFFF9C 0000 67 MBXBFR = -100      ; MAILBOX BUFFER
FFFFFFBC 0000 68 CHANBFRDSC = -68      ; CHANNEL CHARACTERISTICS BUFFER DESCRIPTOR
FFFFFFC4 0000 69 CHANBFR = -60      ; CHANNEL CHARACTERISTICS BUFFER
0000 70
0000003C 0000 71 CHANBFRSIZ = 60      ; CHANNEL CHARACTERISTICS BUFFER SIZE
0000 72
0000 73      ; OFFSETS INTO MAILBOX MESSAGE
0000 74
00000000 0000 75 MBX$L_TYPE = 0      ; MESSAGE TYPE
00000004 0000 76 MBX$B_CTRLR = 4      ; CONTROLLER
00000005 0000 77 MBX$B_MCTYPE = 5      ; MICROCODE TYPE
00000006 0000 78 MBX$W_RMBUNIT = 6      ; RETURN MAILBOX UNIT
0000 79      :
0000 80      : OWN STORAGE:
0000 81      :
0000 82
00000000 0000 83      .PSECT  _LPASCODE,PIC,SHR,EXE,NOWRT,LONG
0000 84
0000 85
0000000A' 0000 86 SMBDSC: .LONG  SMBNAMSIZ      ; SEND MAILBOX NAME DESCRIPTOR
00000008' 0004 87      .LONG  SMBNAM
0000 88
52 45 44 41 4F 4C 24 41 50 4C 0008 89 SMBNAM: .ASCII  /LPASLOADER/      ; SEND MAILBOX NAME
0000000A 0012 90 SMBNAMSIZ = .-SMBNAM

```

```

0012 92 .SBTTL LPA$SNDLDRQ - SEND REQUEST TO LOADER PROCESS
0012 93 :++
0012 94 : FUNCTIONAL DESCRIPTION:
0012 95 :
0012 96 : THIS ROUTINE SENDS A LOAD MICROCODE REQUEST TO THE LOADER PROCESS
0012 97 : AND RECEIVES A RESPONSE OVER A TEMPORARY MAILBOX
0012 98 :
0012 99 : CALLING SEQUENCE:
0012 100 :
0012 101 : CALLS/G
0012 102 :
0012 103 : INPUT PARAMETERS:
0012 104 :
0012 105 : 4(AP) ADDRESS OF A WORD CONTAINING CHANNEL ASSIGNED
0012 106 : 8(AP) MICROCODE TYPE TO LOAD
0012 107 : 1 = MULTIREQUEST MODE
0012 108 : 2 = DEDICATED A/D MODE
0012 109 : 3 = DEDICATED D/A MODE
0012 110 :
0012 111 : IMPLICIT INPUTS:
0012 112 :
0012 113 : THIS ROUTINE ASSUMES THAT A CHANNEL HAS BEEN ASSIGNED TO AN LPA-11
0012 114 :
0012 115 : OUTPUT PARAMETERS:
0012 116 :
0012 117 : R0 CONTAINS COMPLETION CODE
0012 118 : R1 CONTAINS SECOND LONGWORD OF I/O STATUS BLOCK
0012 119 : IF R0 CONTAINS SS$_DEVREQERR, SS$_DEVCMDErr, OR
0012 120 : SS$_CTRLERR
0012 121 :
0012 122 : IMPLICIT OUTPUTS:
0012 123 :
0012 124 : THE CHANNEL IS DEASSIGNED
0012 125 :
0012 126 : COMPLETION CODES:
0012 127 :
0012 128 : VARIOUS COMPLETION CODES RETURNED BY THE SYSTEM
0012 129 :
0012 130 : --
0012 131 :
001C 0012 132 .ENTRY LPA$SNDLDRQ, ^M<R2,R3,R4>
0014 133
0014 134 : ALLOCATE WORK SPACE ON STACK
5E 90 AE 9E 0014 135 MOVAB -WRKSPACE(SP),SP
0018 136
A1 AD 9C AD D4 0018 137 CLRL MBXBFR+MBX$L_TYPE(FP) ; CLEAR MESSAGE TYPE
53 08 AC 90 0018 138 MOVB 8(AP),MBXBFR+MBX$B_MCTYPE(FP) ; STORE M.C. TYPE IN MAILBOX BFR
0020 139 MOVL 4(AP),R3 ; GET ADDRESS OF CHANNEL
0024 140
52 C4 AD 9E 0024 141 MOVAB CHANBFR(FP),R2 ; GET ADDRESS OF CHAN. INFO. BFR
BC AD 3C 9A 0028 142 MOVZBL #CHANBFRSIZ,CHANBFRDSC(FP) ; BUILD A DESCRIPTOR TO
CO AD 52 D0 002C 143 MOVL R2,CHANBFRDSC+4(FP) ; CHANNEL INFO. BUFFER
0030 144
0030 145 $GETCHN_S PRIBUF = CHANBFRDSC(FP),- ; GET CHANNEL INFO.
0030 146 CHAN = (R3) ; FOR DEVICE
50 DD 0043 147 PUSHL R0 ; SAVE STATUS
0045 148

```

```

      50 8ED0 0045 149      $DASSGN_S      CHAN = (R3)      : DEASSIGN CHANNEL TO DEVICE
    2E 50 E9 004F 150      POPL -R0      : RESTORE STATUS
      0052 151      BLBC RO,10$      : ERROR
      0053 152
      0055 153      ; GET DEVICE NAME AND VERIFY ITS AN LPA-11 BEFORE GETTING CONTROLLER
    51 D2 AD 3C 0055 154      MOVZWL B^CHANBFR+DIB$W_DEVNAMOFF(FP),R1 : GET OFFSET TO DEVICE NAME
    50 52 51 C0 0059 155      ADDL R1,R2      : ADD TO START ADDR. OF BUFFER
    03 0000 8F 3C 005C 156      MOVZWL #S$S IVDEVNAM,R0      : ASSUME ERROR
      03 82 91 0061 157      CMPB (R2)+,#3      : SHOULD HAVE AT LEAST 3 CHARS.
    414C 8F 1D 1F 0064 158      BLSSU 10$      : ERROR - LESS THAN 3 CHARS
      82 B1 0066 159      CMPW (R2)+,#^A'LA'      : MAKE SURE DEVICE NAME IS 'LA'
      16 12 006B 160      BNEQ 10$      : IT ISN'T - ERROR
    A0 AD 82 90 006D 161      MOVB (R2)+,MBXBFR+MBX$B_CTRLR(FP) : COPY CTRLR LETTER INTO MB BFR
      0071 162
      0071 163 :
      0071 164 :
      0071 165 :
      0071 166 :
    03 50 E8 0080 168      $ASSIGN_S      DEVNAM = SMBDSC,-      : DEVICE NAME
      00AD 31 0083 169 10$:      CHAN = SMBCHAN(FP)      : CHANNEL
      0086 170      BLBS RO,20$      : SUCCESS
      0086 171 20$:      BRW 90$      : ERROR
      0086 172      ; CREATE A TEMPORARY MAILBOX TO GET REPLY AND GET ITS UNIT NUMBER
      0086 173      $CREMBX_S      PRMFLG = #0,-      : TEMPORARY MAILBOX
      0086 174      CHAN = RMBCHAN(FP),-      : CHANNEL
      0086 175      MAXMSG = #32,-      : MAXIMUM MESSAGE SIZE
      03 50 E8 009A 176      BLBS RO,30$      : SUCCESS
      0082 31 009D 177      BRW 70$      : ERROR
      00A0 178
      00A0 179 30$:      $GETCHN_S      PRIBUF = CHANBFRDSC(FP),-      : PRIMARY BUFFER
      00A0 180      CHAN = RMBCHAN(FP)      : CHANNEL
    5B 50 E9 00B4 181      BLBC RO,60$      : ERROR
      00B7 182
    D0 AD B0 00B7 183      MOVW CHANBFR+DIB$W_UNIT(FP),-      : STORE UNIT # OF RETURN
    A2 AD 00BA 184      MBXBFR+MBX$W_RMBUNIT(FP)      : MAILBOX IN BUFFER
      00BC 185
      00BC 186      ; SEND REQUEST TO LOADER PROCESS
      00BC 187      $QIOW_S      FUNC = #IOS$ WRITEVBLK!IOSM_NOW,-      : FUNCTION
      00BC 188      CHAN = SMBCHAN(FP),-      : CHANNEL
      00BC 189      IOSB = IOSB(FP),-      : I/O STATUS
      00BC 190      P1 = MBXBFR(FP),-      : BUFFER
      00BC 191      P2 = #8      : SIZE
    50 33 50 E9 00DC 192      BLBC RO,60$      : ERROR
      90 AD 7D 00DF 193      MOVQ IOSB(FP),R0      : GET I/O STATUS
      2C 50 E9 00E3 194      BLBC RO,60$      : ERROR
      00E6 195
      00E6 196      ; GET REPLY FROM LOADER PROCESS
      00E6 197      $QIOW_S      FUNC = #IOS$ READVBLK,-      : FUNCTION
      00E6 198      CHAN = RMBCHAN(FP),-      : CHANNEL
      00E6 199      IOSB = IOSB(FP),-      : I/O STATUS BLOCK
      00E6 200      P1 = MBXBFR(FP),-      : BUFFER
      00E6 201      P2 = #32      : SIZE
    50 0B 50 E9 0104 202      BLBC RO,60$      : ERROR
      90 AD 7D 0107 203      MOVQ IOSB(FP),R0      : GET I/O STATUS
      04 50 E9 010B 204      BLBC RO,60$      : ERROR
      010E 205

```

```
50 9C AD 7D 010E 206 ; HAVE REPLY - PICK IT UP
010E 207 MOVQ MBXBFR(FP),R0
0112 208
0112 209 60$: ; DEASSIGN BOTH MAILBOX CHANNELS
7E 50 7D 0112 210 MOVQ R0,-(SP) ; SAVE STATUS
0115 211 $DASSGN_S RMBCHAN(FP)
03 11 0120 212 BRB -80$
0122 213
0122 214 70$: ; DEASSIGN SEND MAILBOX CHANNEL
7E 50 7D 0122 215 MOVQ R0,-(SP) ; SAVE STATUS
0125 216 80$: $DASSGN_S SMBCHAN(FP)
0130 217
50 8E 7D 0130 218 MOVQ (SP)+,R0 ; RESTORE STATUS
0133 219
04 0133 220 90$: RET
0134 221
0134 222
0134 223
0134 224 .END
```



```

SST1          = 00000001
CHANBFR       = FFFFFFFC4
CHANBFRDSC    = FFFFFFFBC
CHANBFRSIZ    = 0000003C
DIBSW_DEVNAMOFF = 0000000E
DIBSW_UNIT    = 0000000C
IOSM_ROW      = 00000040
IOS_READVBLK  = 00000031
IOS_WRITEVBLK = 00000030
IOSB          = FFFFFFF90
LPASSNDLDRQ   = 00000012 RG 02
MBXSB_CTRLR   = 00000004
MBXSB_MCTYPE  = 00000005
MBXSL_TYPE    = 00000000
MBXSW_RMBUNIT = 00000006
MBXBFR        = FFFFFFF9C
RMBCHAN       = FFFFFFF9A
SMBCHAN       = FFFFFFF98
SMBDSC        = 00000000 R 02
SMBNAM        = 00000008 R 02
SMBNAMSIZ     = 0000000A
SSS_IVDEVNAM  = ***** X 02
SYSSASSIGN    = ***** GX 02
SYSSCREMBX    = ***** GX 02
SYSSDASSGN    = ***** GX 02
SYSSGETCHN    = ***** GX 02
SYSSQIOW      = ***** GX 02
WRKSPACE      = 00000070
    
```

-----  
! 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
_LPASCODE	00000134 ( 308.)	02 ( 2.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	33	00:00:00.07	00:00:00.62
Command processing	115	00:00:00.47	00:00:01.40
Pass 1	189	00:00:04.45	00:00:08.90
Symbol table sort	0	00:00:00.60	00:00:01.21
Pass 2	53	00:00:00.93	00:00:02.45
Symbol table output	4	00:00:00.04	00:00:00.04
Psect synopsis output	1	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	397	00:00:06.59	00:00:14.65

The working set limit was 1050 pages.  
24453 bytes (48 pages) of virtual memory were used to buffer the intermediate code.

There were 30 pages of symbol table space allocated to hold 413 non-local and 7 local symbols.  
224 source lines were read in Pass 1, producing 13 object records in Pass 2.  
18 pages of virtual memory were used to define 17 macros.

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

Macro library name	Macros defined
-----	-----
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	14

547 GETS were required to define 14 macros.

There were no errors, warnings or information messages.

MACRO/DISABLE=TRACE/LIS=LIS\$:LASNDLDRQ/OBJ=OBJ\$:LASNDLDRQ MSRC\$:LASNDLDRQ/UPDATE=(ENH\$:LASNDLDRQ)



