```
MMM         MMM     CCCCCCCCCCCC  LLL              DDDDDDDDDDDD     RRRRRRRRRRRR
MMM         MMM     CCCCCCCCCCCC  LLL              DDDDDDDDDDDD     RRRRRRRRRRRR
MMM         MMM     CCCCCCCCCCCC  LLL              DDDDDDDDDDDD     RRRRRRRRRRRR
MMMMMM   MMMMMM  CCC              LLL              DDD         DDD  RRR         RRR
MMMMMM   MMMMMM  CCC              LLL              DDD         DDD  RRR         RRR
MMMMMM   MMMMMM  CCC              LLL              DDD         DDD  RRR         RRR
MMM   MMM   MMM  CCC              LLL              DDD         DDD  RRR         RRR
MMM   MMM   MMM  CCC              LLL              DDD         DDD  RRR         RRR
MMM   MMM   MMM  CCC              LLL              DDD         DDD  RRR         RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRRRRRRRRRRR
MMM         MMM  CCC              LLL              DDD         DDD  RRRRRRRRRRRR
MMM         MMM  CCC              LLL              DDD         DDD  RRRRRRRRRRRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR   RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR   RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR   RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR      RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR      RRR
MMM         MMM  CCC              LLL              DDD         DDD  RRR      RRR
MMM         MMM     CCCCCCCCCCCC  LLLLLLLLLLLLLLL  DDDDDDDDDDDD     RRR         RRR
MMM         MMM     CCCCCCCCCCCC  LLLLLLLLLLLLLLL  DDDDDDDDDDDD     RRR         RRR
MMM         MMM     CCCCCCCCCCCC  LLLLLLLLLLLLLLL  DDDDDDDDDDDD     RRR         RRR
```

```
_$2

Val
---
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
```

```
LL              AAAAAA    LL              OOOOOO      AAAAAA    DDDDDDDD    EEEEEEEEEE  RRRRRRRR
LL              AAAAAA    LL              OOOOOO      AAAAAA    DDDDDDDD    EEEEEEEEEE  RRRRRRRR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR      RR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR      RR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR      RR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR      RR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EEEEEEEE    RRRRRRRR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EEEEEEEE    RRRRRRRR
LL            AAAAAAAAAA  LL            OO      OO  AAAAAAAAAA  DD      DD  EE          RR  RR
LL            AAAAAAAAAA  LL            OO      OO  AAAAAAAAAA  DD      DD  EE          RR  RR
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR    RR      ....
LL            AA      AA  LL            OO      OO  AA      AA  DD      DD  EE          RR    RR      ....
LLLLLLLLLL    AA      AA  LLLLLLLLLL      OOOOOO    AA      AA  DDDDDDDD    EEEEEEEEEE  RR      RR    ....
LLLLLLLLLL    AA      AA  LLLLLLLLLL      OOOOOO    AA      AA  DDDDDDDD    EEEEEEEEEE  RR      RR    ....

LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
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  LALOADER
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 UTILITY PROGRAMS
0000    31 ;
0000    32 ; ABSTRACT:
0000    33 ;       THIS PROGRAM IS THE LPA-11 MICROCODE LOADER.  IT LOADS MICROCODE
0000    34 ;       WHEN REQUESTED BY A MAILBOX REQUEST AND RELOADS MICROCODE AFTER
0000    35 ;       A POWER RECOVERY.
0000    36 ;
0000    37 ; ENVIRONMENT:  USER MODE
0000    38 ;
0000    39 ; AUTHOR:  STEVE BECKHARDT,     CREATION DATE:  4-OCT-78
0000    40 ;
0000    41 ; MODIFIED BY:
0000    42 ;
0000    43 ;       V02-003 SRB0003         Steve Beckhardt         3-Sep-1980
0000    44 ;               Changed microcode version number in mode word of initialize
0000    45 ;               command from 4 to 5.
0000    46 ;
0000    47 ;--
```

DECLARATIONS

```
                     0000     49          .SBTTL  DECLARATIONS
                     0000     50 ;
                     0000     51 ; INCLUDE FILES:
                     0000     52 ;
                     0000     53          $DIBDEF                 ; DIB OFFSETS
                     0000     54          $LADEF                  ; LPA-11 DEFINITIONS
                     0000     55
                     0000     56 ;
                     0000     57 ; MACROS:
                     0000     58 ;
                     0000     59
                     0000     60 ;
                     0000     61 ; EQUATED SYMBOLS:
                     0000     62 ;
                     0000     63
00000001             0000     64 CLKMODE = 1                     ; SET CLOCK MODE WORD
00000141             0000     65 CLKSTATUS = ^X141               ; SET CLOCK STATUS WORD
00000001             0000     66 DEFCLKRATE = 1                  ; DEFAULT CLOCK RATE (1 MHZ)
FFFFFF9C             0000     67 DEFPRESET = -100                ; DEFAULT CLOCK PRESET
                     0000     68
000000D6             0000     69 DEVNFND = ^0326                 ; DEVICE NOT FOUND ERROR (INITIALIZE QIO)
                     0000     70
00001100             0000     71 PROTMASK = ^X1100               ; MAILBOX PROTECTION MASK
                     0000     72
00000008             0000     73 MINMSGSIZ = 8                   ; MINIMUM MESSAGE SIZE
                     0000     74
                     0000     75 ; OFFSETS INTO MAILBOX
                     0000     76
00000000             0000     77 MBX$L_TYPE = 0                  ; MESSAGE TYPE
00000004             0000     78 MBX$B_CTRLR = 4                 ; CONTROLLER LETTER
00000005             0000     79 MBX$B_MCTYPE = 5                ; MICROCODE TYPE
00000006             0000     80 MBX$W_RMBUNIT = 6               ; RETURN MAILBOX UNIT NUMBER
                     0000     81
                     0000     82 ;
                     0000     83 ; OWN STORAGE:
                     0000     84 ;
                     0000     85
                00000000     86          .PSECT  _LPA$DATA,LONG
                     0000     87
                     0000     88 CNTRLR_TBL:                     ; CONTROLLER TABLE
00000000             0000     89          .LONG   0
                     0004     90
0000000C             0004     91 LAIOSB: .BLKQ    1              ; LPA-11 I/O STATUS BLOCK
                     000C     92
                     000C     93 PMBIOSB:
00000014             000C     94          .BLKQ   1              ; PERM. MAILBOX I/O STATUS BLOCK
                     0014     95
                     0014     96 PMBCHAN:                        ; PERMANENT MAILBOX CHANNEL
0000                 0014     97          .WORD   0
                     0016     98 RMBCHAN:                        ; REPLY MAILBOX CHANNEL
0000                 0016     99          .WORD   0
                     0018    100 LACHAN:                         ; LPA-11 CHANNEL
0000                 0018    101          .WORD   0
                     001A    102
                     001A    103 INITTBL:                        ; TABLE FOR INITIALIZE QIO
0500                 001A    104          .WORD   ^X0500         ; MODE WORD (INCLUDES UCODE VERSION NUMBER)
00000030             001C    105          .BLKW   10             ; SPACE FOR TEN DEVICE ADDRESSES
```

```
00000130  0030   106 DMDT:    .BLKB   256             ; DEDICATED MODE DISPATCH TABLE
          0130   107
          0130   108 CHANBFR:                         ; BUFFER FOR CHANNEL CHARACTERISTICS
0000013C  0130   109          .BLKB   DIB$L_DEVDEPEND+4
0000000C  013C   110 CHANBFRSIZ = .-CHANBFR
          013C   111
0000015C  013C   112 MBXBFR: .BLKB   32              ; MAILBOX BUFFER
          015C   113
   41 4C  015C   114 LANAME: .ASCII  'LA'            ; LPA-11 DEVICE NAME
          015E   115 LACTRLR:
      20  015E   116          .ASCII  ' '           ; SPACE FOR CONTROLLER LETTER
      30  015F   117          .ASCII  '0'           ; UNIT ZERO
00000004  0160   118 LANAMESIZ = .-LANAME
          0160   119
```

```
                     00000000   121          .PSECT  _LPA$CODE,NOWRT,LONG
                           0000 122
                           0000 123 PMBNAMDSC:                          ; PERMANENT MAILBOX NAME DESCRIPTOR
               0000000A'   0000 124          .LONG   PMBNAMSIZ
               0000002C'   0004 125          .LONG   PMBNAM
                           0008 126
                           0008 127 LANAMEDSC:                          ; LPA-11 DEVICE NAME DESCRIPTOR
               00000004    0008 128          .LONG   LANAMESIZ
               0000015C'   000C 129          .LONG   LANAME
                           0010 130
                           0010 131 CHANBFRDSC:                         ; CHANNEL CHARACTERISTICS BUFFER DESCRIPTOR
               0000000C    0010 132          .LONG   CHANBFRSIZ
               00000130'   0014 133          .LONG   CHANBFR
                           0018 134
                           0018 135
                           0018 136 DEVTBL:                             ; TABLE OF DEVICE ADDRESS FOR INITIALIZE QIO
                     F104  0018 137          .WORD   ^O170404           ; CLOCK A
                     F11A  001A 138          .WORD   ^O170432           ; CLOCK B
                     F100  001C 139          .WORD   ^O170400           ; A/D #1
                     F120  001E 140          .WORD   ^O170440           ; A/D #2
                     F110  0020 141          .WORD   ^O170420           ; D/A
                     EFF8  0022 142          .WORD   ^O167770           ; DIGITAL I/O #1
                     EFF0  0024 143          .WORD   ^O167760           ; DIGITAL I/O #2
                     EFE8  0026 144          .WORD   ^O167750           ; DIGITAL I/O #3
                     EFE0  0028 145          .WORD   ^O167740           ; DIGITAL I/O #4
                     EFD8  002A 146          .WORD   ^O167730           ; DIGITAL I/O #5
                           002C 147
52 45 44 41 4F 4C 24 41 50 4C  002C 148 PMBNAM: .ASCII  'LPA$LOADER'     ; PERMANENT MAILBOX NAME
                     0000000A  0036 149 PMBNAMSIZ = .-PMBNAM
```

```
                                0036   151          .SBTTL  MAIN PROGRAM
                                0036   152 ;++
                                0036   153 ; FUNCTIONAL DESCRIPTION:
                                0036   154 ;
                                0036   155 ;       THIS IS THE MAIN PROGRAM FOR THE LPA-11 MICROCODE LOADER.
                                0036   156 ;       IT PERFORMS SOME INITIALIZATION, SPECIFIES A POWER RECOVERY AST,
                                0036   157 ;       CREATES A MAILBOX, SPECIFIES A MAILBOX AST, AND THEN HIBERNATES.
                                0036   158 ;       ALL SUBSEQUENT PROCESSING IS PERFORMED IN ONE OF THE TWO AST ROUTINES.
                                0036   159 ;
                                0036   160 ; CALLING SEQUENCE:
                                0036   161 ;
                                0036   162 ;       ENTERED WHEN PROGRAM IS STARTED
                                0036   163 ;
                                0036   164 ; INPUT PARAMETERS:
                                0036   165 ;
                                0036   166 ;       NONE
                                0036   167 ;
                                0036   168 ; OUTPUT PARAMETERS:
                                0036   169 ;
                                0036   170 ;       NONE
                                0036   171 ;
                                0036   172 ;--
                                0036   173
                         003C   0036   174          .ENTRY  START,^M<R2,R3,R4,R5>
                                0038   175
                                0038   176          ; PERFORM INITIALIZATION
              00000000'EF   D4  0038   177          CLRL    CNTRLR_TBL                      ; CLEAR CONTROLLER TABLE
  00000000'EF   0100 8F   28  003E   178          MOVC3   #256,LPA$$DMDT,DMDT             ; COPY DED. MODE DISPATCH TABLE
              00000030'EF       0047
                                004C   179
                                004C   180          $SETSFM_S       #1                      ; SET SYS. SERVICE FAILURE EXCP.
                                0055   181
                                0055   182          $SETPRA_S       PWRRECAST               ; SET POWER RECOVERY AST
                                0064   183
                                0064   184          ; CREATE A PERMANENT MAILBOX AND MARK IT FOR DELETION TO CLEANUP
                                0064   185          ; IF THIS PROCESS EXITS
                                0064   186          $CREMBX_S       PRMFLG = #1,-           ; PERMANENT
                                0064   187                          CHAN = PMBCHAN,-        ; CHANNEL
                                0064   188                          MAXMSG = #32,-          ; MAXIMUM MESSAGE SIZE
                                0064   189                          BUFQUO = #64,-          ; BUFFER QUOTA
                                0064   190                          PROMSK = #PROTMASK,-    ; PROTECTION MASK
                                0064   191                          LOGNAM = PMBNAMDSC      ; LOGICAL NAME
                                0086   192
                                0086   193          $DELMBX_S       PMBCHAN
                                0094   194
                                0094   195          ; SET UP MAILBOX AST
                                0094   196          $QIOW_S         FUNC = #IO$_SETMODE!IO$M_WRTATTN,-      ; FUNCTION
                                0094   197                          CHAN = PMBCHAN,-        ; CHANNEL
                                0094   198                          P1 = PMBAST             ; MAILBOX AST ADDRESS
                                00B9   199
                                00B9   200          ; NOW HIBERNATE
                                00B9   201          $HIBER_S
```

```
                                00C0   203          .SBTTL  PMBAST - MAILBOX AST ROUTINE
                                00C0   204 ;++
                                00C0   205 ; FUNCTIONAL DESCRIPTION:
                                00C0   206 ;
                                00C0   207 ;       THIS ROUTINE IS THE MAILBOX AST ROUTINE.  WAKING UP HERE MEANS
                                00C0   208 ;       A PROCESS HAS SENT US A MAILBOX REQUEST TO LOAD SOME MICROCODE.
                                00C0   209 ;       THE GENERAL FLOW IS TO READ THE MAILBOX, LOAD THE MICROCODE,
                                00C0   210 ;       AND OPTIONALLY SEND STATUS BACK.
                                00C0   211 ;
                                00C0   212 ; CALLING SEQUENCE:
                                00C0   213 ;
                                00C0   214 ;       CALLS/G FROM AST DISPATCHER WHEN A MAILBOX WRITE OCCURS
                                00C0   215 ;
                                00C0   216 ; INPUT PARAMETERS:
                                00C0   217 ;
                                00C0   218 ;       NONE
                                00C0   219 ;
                                00C0   220 ; OUTPUT PARAMETERS:
                                00C0   221 ;
                                00C0   222 ;       NONE
                                00C0   223 ;
                                00C0   224 ;--
                                00C0   225
                        003C    00C0   226          .ENTRY  PMBAST, ^M<R2,R3,R4,R5>
                                00C2   227
                                00C2   228 10$:     ; ENABLE SYSTEM SERVICE FAILURE EXCEPTIONS
                                00C2   229          $SETSFM_S       #1
                                00CB   230
                                00CB   231 15$:     ; READ MAILBOX
                                00CB   232          $QIOW_S         FUNC = #IO$_READVBLK!IO$M_NOW,- ; FUNCTION
                                00CB   233                          CHAN = PMBCHAN,-        ; CHANNEL
                                00CB   234                          IOSB = PMBIOSB,-        ; I/O STATUS BLOCK
                                00CB   235                          P1 = MBXBFR,-           ; BUFFER ADDRESS
                                00CB   236                          P2 = #32                ; BUFFER SIZE
                                00F4   237
00'  0000000C'EF    B1          00F4   238          CMPW    PMBIOSB,S^#SS$_NORMAL   ; WAS THERE A MESSAGE THERE?
                       23   13  00FB   239          BEQL    20$                     ; YES
                                00FD   240
                                00FD   241          ; MAILBOX IS EMPTY - REENABLE MAILBOX ASTS AND HIBERNATE AGAIN
                                00FD   242          $QIOW_S         FUNC = #IO$_SETMODE!IO$M_WRTATTN,-      ; FUNCTION
                                00FD   243                          CHAN = PMBCHAN,-        ; CHANNEL
                                00FD   244                          P1 = PMBAST             ; MAILBOX AST ADDRESS
                                011F   245
                             04 011F   246          RET
                                0120   247
                                0120   248 20$:     ; HAVE A MESSAGE - MAKE SURE IT'S AT LEAST MINIMUM SIZE
08   0000000E'EF    B1          0120   249          CMPW    PMBIOSB+2,#MINMSGSIZ
                       A2   1F  0127   250          BLSSU   15$                     ; TOO SMALL -IGNORE
                                0129   251
                                0129   252          ; MESSAGE SIZE IS OK - GET LPA CONTROLLER LETTER
52   00000140'EF    9A          0129   253          MOVZBL  MBXBFR+MBX$B_CTRLR,R2   ; GET CONTROLLER LETTER
0000015E'EF    52   90          0130   254          MOVB    R2,LACTRLR              ; STORE IN LANAME
               52   D7          0137   255          DECL    R2                      ; CONVERT TO BIT NUMBER
      52    E0 8F   8A          0139   256          BICB    #^XE0,R2                ; FROM 0 TO 31
                                013D   257
                                013D   258          ; DISABLE SYSTEM SERVICE FAILURE EXCEPTIONS
                                013D   259          $SETSFM_S
```

```
                     0146   260
                     0146   261          ; ASSIGN A CHANNEL TO SPECIFIED LPA-11
                     0146   262          $ASSIGN_S       DEVNAM = LANAMEDSC,-    ; DEVICE NAME
                     0146   263                          CHAN = LACHAN           ; CHANNEL
                34 50 E9  0159   264          BLBC    R0,50$                  ; ERROR
                     015C   265
                     015C   266          ; GET MICROCODE TYPE AND LOAD MICROCODE
     54 00000141'EF 9A    015C   267          MOVZBL  MBXBFR+MBX$B_MCTYPE,R4  ; MICROCODE TYPE
                0114 30    0163   268          BSBW    LOADMC                  ; LOAD IT
                13 50 E9   0166   269          BLBC    R0,40$                  ; ERROR
                     0169   270
                     0169   271          ; SET APPROPRIATE BIT IN CONTROLLER TABLE
00 00000000'EF    52 E2    0169   272          BBSS    R2,CNTRLR_TBL,30$
                     0171   273
                     0171   274 30$:     ; START CLOCK AT A DEFAULT RATE
                54 01 9A   0171   275          MOVZBL  #DEFCLKRATE,R4          ; DEFAULT CLOCK RATE
             55 FF9C 8F 3C 0174   276          MOVZWL  #DEFPRESET,R5           ; DEFAULT PRESET
                0200 30    0179   277          BSBW    SETCLOCK                ; SET CLOCK
                     017C   278
                     017C   279 40$:     ; DEASSIGN CHANNEL TO LPA
                7E 50 7D   017C   280          MOVQ    R0,-(SP)                ; SAVE STATUS IN R0, R1
                     017F   281          $DASSGN_S       LACHAN
                50 8E 7D   018D   282          MOVQ    (SP)+,R0                ; RESTORE STATUS
                     0190   283
                     0190   284 50$:     ; OPTIONALLY SEND STATUS IN R0 AND R1 BACK TO REQUESTOR
     53 00000142'EF 3C    0190   285          MOVZWL  MBXBFR+MBX$W_RMBUNIT,R3 ; GET UNIT # OF RETURN MAILBOX
                40 13      0197   286          BEQL    60$                     ; NO RETURN MAILBOX
                     0199   287
00000013C'EF    50 7D      0199   288          MOVQ    R0,MBXBFR               ; STORE STATUS IN BUFFER
                     01A0   289
                0215 30    01A0   290          BSBW    ASSIGNRMB               ; ASSIGN A CHANNEL TO RETURN MAILBOX
                33 50 E9   01A3   291          BLBC    R0,60$                  ; CAN'T ASSIGN CHANNEL
                     01A6   292
                     01A6   293          ; SEND REPLY
                     01A6   294          $QIOW_S         FUNC = #IO$_WRITEVBLK!IO$M_NOW,-        ; FUNCTION
                     01A6   295                          CHAN = RMBCHAN,-        ; CHANNEL
                     01A6   296                          P1 = MBXBFR,-           ; BUFFER ADDRESS
                     01A6   297                          P2 = #8                 ; SIZE OF MESSAGE
                     01CB   298
                     01CB   299          $DASSGN_S       RMBCHAN         ; DEASSIGN CHANNEL TO RETURN MAILBOX
          FEE6  31   01D9   300 60$:     BRW     10$                     ; DO NEXT REQUEST
```

```
                                01DC  302          .SBTTL  PWRRECAST - POWER RECOVERY AST ROUTINE
                                01DC  303 ;++
                                01DC  304 ; FUNCTIONAL DESCRIPTION:
                                01DC  305 ;
                                01DC  306 ;       THIS ROUTINE IS THE POWER RECOVERY AST ROUTINE.  FOR EACH LPA-11
                                01DC  307 ;       CONTROLLER THAT HAS BEEN LOADED BY THIS LOADER (SAVED IN CNTRLR_TBL),
                                01DC  308 ;       THIS ROUTINE GETS THE DEVICE CHARACTERISTICS, LOADS THE SAME MICROCODE
                                01DC  309 ;       AS WAS LAST LOADED, AND SETS THE CLOCK TO THE LAST CLOCK RATE SET.
                                01DC  310 ;
                                01DC  311 ; CALLING SEQUENCE:
                                01DC  312 ;
                                01DC  313 ;       CALLS/G FROM AST DISPATCHER WHEN POWER RECOVERY OCCURS
                                01DC  314 ;
                                01DC  315 ; INPUT PARAMETERS:
                                01DC  316 ;
                                01DC  317 ;       NONE
                                01DC  318 ;
                                01DC  319 ; OUTPUT PARAMETERS:
                                01DC  320 ;
                                01DC  321 ;       NONE
                                01DC  322 ;--
                                01DC  323
                         003C   01DC  324          .ENTRY  PWRRECAST, ^M<R2,R3,R4,R5>
                                01DE  325
                                01DE  326          ; REENABLE POWER RECOVERY ASTS
                                01DE  327          $SETPRA_S       PWRRECAST
                                01EA  328
                                01EA  329          ; DISABLE SYSTEM SERVICE FAILURE EXCEPTIONS
                                01EA  330          $SETSFM_S
                                01F3  331
                     52    D4   01F3  332          CLRL    R2                      ; LOOP COUNTER
                                01F5  333
                                01F5  334 10$:     ; WAS NEXT CONTROLLER LOADED?
 6F 00000000'EF      52    E1   01F5  335          BBC     R2,CNTRLR_TBL,80$       ; BR. IF NO
                                01FD  336
                                01FD  337          ; YES, CONVERT NUMBER TO CONTROLLER LETTER
      53    52    40 8F    89   01FD  338          BISB3   #^X40,R2,R3             ; OR IN ^X40
                       53    D6   0202  339          INCL    R3                      ; ADD 1
                                0204  340
  0000015E'EF         53    90   0204  341          MOVB    R3,LACTRLR              ; MOVE CONTROLLER LETTER INTO LANAME
                                020B  342
                                020B  343          ; ASSIGN A CHANNEL TO LPA-11
                                020B  344          $ASSIGN_S       DEVNAM = LANAMEDSC,-    ; DEVICE NAME
                                020B  345                          CHAN = LACHAN           ; CHANNEL
                     4B 50      E9   021E  346          BLBC    R0,80$                  ; UNABLE TO ASSIGN A CHANNEL
                                0221  347
                                0221  348          ; GET CHANNEL INFO.
                                0221  349          $GETCHN_S       PRIBUF = CHANBFRDSC,-   ; BUFFER
                                0221  350                          CHAN = LACHAN           ; CHANNEL
                     22 50      E9   0239  351          BLBC    R0,70$                  ; ERROR
                                023C  352
      53  00000138'EF         D0   023C  353          MOVL    CHANBFR+DIB$L_DEVDEPEND,R3      ; GET DEV. DEP. CHARACTERISTICS
                                0243  354
                                0243  355          ; LOAD MICROCODE
    54    53    02    01     EF   0243  356          EXTZV   #LA$V_MCTYPE,#LA$S_MCTYPE,R3,R4 ; GET MICROCODE TYPE IN R4
                         002F   30   0248  357          BSBW    LOADMC
                      10 50     E9   024B  358          BLBC    R0,70$                  ; ERROR
```

```
                     024E   359
                     024E   360          ; SET CLOCK RATE
54   53   03   0D   EF 024E   361          EXTZV   #LA$V_RATE,#LA$S_RATE,R3,R4     ; GET CLOCK RATE
55   53   10   10   EE 0253   362          EXTV    #LA$V_PRESET,#LA$S_PRESET,R3,R5 ; GET CLOCK PRESET
                0121   30 0258   363          BSBW    SETCLOCK                        ; SET CLOCK RATE
              00 50   E9 025B   364          BLBC    R0,70$                          ; ERROR
                     025E   365
                     025E   366 70$:       $DASSGN_S       LACHAN                  ; DEASSIGN CHANNEL
                     026C   367
           85 52   20   F2 026C   368 80$:       AOBLSS  #32,R2,10$                      ; REPEAT FOR 32 CONTROLLERS
                     0270   369
                     0270   370          ; REENABLE SYSTEM SERVICE FAILURE EXCEPTIONS
                     0270   371          $SETSFM_S       #1
                     0279   372
                  04 0279   373          RET
```

```
                                027A   375          .SBTTL  LOADMC - ROUTINE TO LOAD MICROCODE
                                027A   376 ;++
                                027A   377 ; FUNCTIONAL DESCRIPTION:
                                027A   378 ;
                                027A   379 ;       THIS ROUTINE IS THE ACTUAL MICROCODE LOADER.  IT DOES THE FOLLOWING:
                                027A   380 ;               1)  LOADS THE SPECIFIED MICROCODE
                                027A   381 ;               2)  STARTS THE MICROPROCESSOR
                                027A   382 ;               3)  INITIALIZES THE LPA-11.  THE INITIALIZE IS REPEATED
                                027A   383 ;                   UNTIL THE CORRECT DEVICE CONFIGURATION ON THE LPA'S
                                027A   384 ;                   BUS IS FOUND.
                                027A   385 ;
                                027A   386 ; CALLING SEQUENCE:
                                027A   387 ;
                                027A   388 ;       BSBW/B
                                027A   389 ;
                                027A   390 ; INPUT PARAMETERS:
                                027A   391 ;
                                027A   392 ;       R4      CONTAINS MICROCODE TYPE TO LOAD:
                                027A   393 ;               1 = MULTIREQUEST MODE
                                027A   394 ;               2 = DEDICATED A/D MODE
                                027A   395 ;               3 = DEDICATED D/A MODE
                                027A   396 ;
                                027A   397 ; IMPLICIT INPUTS:
                                027A   398 ;
                                027A   399 ;       LACHAN  CONTAINS CHANNEL NUMBER OF LPA-11
                                027A   400 ;
                                027A   401 ; OUTPUT PARAMETERS:
                                027A   402 ;
                                027A   403 ;       R0      CONTAINS COMPLETION CODE
                                027A   404 ;       R1      CONTAINS SECOND LONGWORD OF I/O STATUS BLOCK ON I/O ERRORS
                                027A   405 ;
                                027A   406 ; COMPLETION CODES:
                                027A   407 ;
                                027A   408 ;       VARIOUS SYSTEM STATUS RETURNS
                                027A   409 ;       SS$_BADPARAM IS RETURNED IF R4 DOES NOT CONTAIN A 1,2, OR 3
                                027A   410 ;--
                                027A   411
                                027A   412 LOADMC:
                          3C    BB    027A   413          PUSHR   #^M<R2,R3,R4,R5>
                                027C   414
                                027C   415          ; SEE WHICH TYPE OF MICROCODE IS SPECIFIED IN R4
50  00000000'EF               3E    027C   416          MOVAW   LPA$$MRMCODE,R0         ; ASSUME MULTIREQUEST MODE
                    01    54    D1    0283   417          CMPL    R4,#1                   ; IS IT?
                          1F    13    0286   418          BEQL    10$                     ; YES
                                0288   419
50  00000000'EF               3E    0288   420          MOVAW   LPA$$ADMCODE,R0         ; ASSUME DED. A/D MODE
                    02    54    D1    028F   421          CMPL    R4,#2                   ; IS IT?
                          13    13    0292   422          BEQL    10$                     ; YES
                                0294   423
50  00000000'EF               3E    0294   424          MOVAW   LPA$$DAMCODE,R0         ; ASSUME DED. D/A MODE
                    03    54    D1    029B   425          CMPL    R4,#3                   ; IS IT?
                          07    13    029E   426          BEQL    10$                     ; YES
                                02A0   427
                                02A0   428          ; ERROR - BAD VALUE
    50  0000'8F               3C    02A0   429          MOVZWL  #SS$_BADPARAM,R0        ; RETURN STATUS
                          58    11    02A5   430          BRB     15$
                                02A7   431
```

```
                        02A7    432 10$:    ; R0 POINTS TO MICROCODE IMAGE TO LOAD - DO IT!
                        02A7    433         $QIOW_S         FUNC = #IO$_LOADMCODE,- ; FUNCTION
                        02A7    434                         CHAN = LACHAN,-         ; CHANNEL
                        02A7    435                         IOSB = LAIOSB,-         ; I/O STATUS BLOCK
                        02A7    436                         P1 = (R0),-             ; ADDRESS OF MICROCODE
                        02A7    437                         P2 = #2048,-            ; SIZE OF MICROCODE IMAGE
                        02A7    438                         P3 = #0                 ; STARTING MICRO PC
                  2C 50 E9      02D0    439         BLBC    R0,15$                  ; ERROR
          28 00000004'EF E9     02D3    440         BLBC    LAIOSB,17$              ; ERROR
                        02DA    441
                        02DA    442         ; NOW START MICROPROCESSOR
                        02DA    443         $QIOW_S         FUNC = #IO$_STARTMPROC,-        ; FUNCTION
                        02DA    444                         CHAN = LACHAN,-                 ; CHANNEL
                        02DA    445                         IOSB = LAIOSB                   ; I/O STATUS BLOCK
                  77 50 E9      02FF    446 15$:    BLBC    R0,70$                  ; ERROR
          69 00000004'EF E9     0302    447 17$:    BLBC    LAIOSB,60$              ; ERROR
                        0309    448
                        0309    449         ; NOW INITIALIZE IT
0000001C'EF   FD0A CF   14  28  0309    450         MOVC3   #20,DEVTBL,INITTBL+2    ; START WITH ALL 10 DEVICE ADDRESSES
                        0313    451
                        0313    452 20$:    $QIOW_S         FUNC = #IO$_INITIALIZE,-        ; FUNCTION
                        0313    453                         CHAN = LACHAN,-                 ; CHANNEL
                        0313    454                         IOSB = LAIOSB,-                 ; I/O STATUS BLOCK
                        0313    455                         P1 = INITTBL,-                  ; ADDRESS OF INIT. TABLE
                        0313    456                         P2 = #278                       ; SIZE OF TABLE
                  36 50 E9      0340    457         BLBC    R0,70$                  ; ERROR
          28 00000004'EF E8     0343    458         BLBS    LAIOSB,60$              ; SUCCESS!
                        034A    459
                        034A    460         ; INITIALIZE FAILED - IS IT BECAUSE OF A BAD DEVICE ADDRESS?
        D6 8F   00000009'EF 91  034A    461         CMPB    LAIOSB+5,#DEVNFND       ; IS IT DEVICE NOT FOUND?
                     1E 12      0352    462         BNEQ    60$                     ; NO, ERROR
                        0354    463
                        0354    464         ; HAVE A DEVICE NOT FOUND ERROR - FIND CORRESPONDING ENTRY IN
                        0354    465         ; DEVICE TABLE AND SET IT EQUAL TO -1
                     52 D4      0354    466         CLRL    R2                      ; LOOP COUNTER
0000001C'EF42   0000000A'EF B1  0356    467 30$:    CMPW    LAIOSB+6,INITTBL+2[R2]  ; MATCH?
                     0A 12      0362    468         BNEQ    40$                     ; NO
          0000001C'EF42 01 AE   0364    469         MNEGW   #1,INITTBL+2[R2]        ; YES - SET TO -1
                     A5 11      036C    470         BRB     20$                     ; RETRY INITIALIZE
                  E4 52 0A F2   036E    471 40$:    AOBLSS  #10,R2,30$              ; TRY NEXT ENTRY
                        0372    472
                        0372    473 60$:    ; PICK UP I/O STATUS BLOCK BEFORE RETURNING
               50 00000004'EF 7D 0372    474         MOVQ    LAIOSB,R0
                        0379    475
                        0379    476 70$:    ; HAVE STATUS IN R0 (AND MAYBE R1 TOO)
                     3C BA      0379    477         POPR    #^M<R2,R3,R4,R5>
                     05         037B    478         RSB
```

```
                     037C    480          .SBTTL  SETCLOCK - ROUTINE TO SET LPA-11 CLOCK
                     037C    481 ;++
                     037C    482 ; FUNCTIONAL DESCRIPTION:
                     037C    483 ;
                     037C    484 ;       THIS ROUTINE SETS THE LPA-11 CLOCK TO THE SPECIFIED RATE AND PRESET.
                     037C    485 ;
                     037C    486 ; CALLING SEQUENCE:
                     037C    487 ;
                     037C    488 ;       BSBW/B
                     037C    489 ;
                     037C    490 ; INPUT PARAMETERS:
                     037C    491 ;
                     037C    492 ;       R4      CONTAINS CLOCK RATE IN LOW 3 BITS
                     037C    493 ;       R5      CONTAINS CLOCK PRESET
                     037C    494 ;
                     037C    495 ; IMPLICIT INPUTS:
                     037C    496 ;
                     037C    497 ;       LACHAN  CONTAINS CHANNEL NUMBER OF LPA-11
                     037C    498 ;
                     037C    499 ; OUTPUT PARAMETERS:
                     037C    500 ;
                     037C    501 ;       R0      CONTAINS COMPLETION CODE
                     037C    502 ;       R1      CONTAINS SECOND LONGWORD OF I/O STATUS BLOCK ON I/O ERRORS
                     037C    503 ;
                     037C    504 ; COMPLETION CODES:
                     037C    505 ;
                     037C    506 ;       VARIOUS SYSTEM STATUS RETURNS
                     037C    507 ;
                     037C    508 ;--
                     037C    509
                     037C    510 SETCLOCK:
     50    0141 8F   3C  037C  511          MOVZWL  #CLKSTATUS,R0           ; CLOCK STATUS
50   03    01   54   F0  0381  512          INSV    R4,#1,#3,R0             ; INSERT CLOCK RATE
                     0386  513
                     0386  514          ; DO IT!
                     0386  515          $QIOW_S         FUNC = #IO$_SETCLOCK,-  ; FUNCTION
                     0386  516                          CHAN = LACHAN,-         ; CHANNEL
                     0386  517                          IOSB = LAIOSB,-         ; I/O STATUS BLOCK
                     0386  518                          P2 = #CLKMODE,-         ; MODE WORD
                     0386  519                          P3 = R0,-               ; CLOCK STATUS
                     0386  520                          P4 = R5                 ; CLOCK PRESET
           07 50   E9  03AD  521          BLBC    R0,50$                  ; ERROR
                     03B0  522
50   00000004'EF   7D  03B0  523          MOVQ    LAIOSB,R0               ; PICK UP I/O STATUS BLOCK
                     03B7  524
                     05  03B7  525 50$:    RSB
```

```
                     03B8   527          .SBTTL  ASSIGNRMB - ASSIGN A CHANNEL TO RETURN MAILBOX
                     03B8   528 ;++
                     03B8   529 ; FUNCTIONAL DESCRIPTION:
                     03B8   530 ;
                     03B8   531 ;       THIS ROUTINE ASSIGNS A CHANNEL TO THE RETURN MAILBOX.
                     03B8   532 ;       IT ASSIGNS A CHANNEL TO THE NAME MBn WHERE N IS AN INPUT ARGUMENT.
                     03B8   533 ;
                     03B8   534 ; CALLING SEQUENCE:
                     03B8   535 ;
                     03B8   536 ;       BSBW/B
                     03B8   537 ;
                     03B8   538 ; INPUT PARAMETERS:
                     03B8   539 ;
                     03B8   540 ;       R3      CONTAINS THE MAILBOX NUMBER TO ASSIGN A CHANNEL TO
                     03B8   541 ;
                     03B8   542 ; OUTPUT PARAMETERS:
                     03B8   543 ;
                     03B8   544 ;       R0      CONTAINS A COMPLETION CODE
                     03B8   545 ;
                     03B8   546 ; IMPLICIT OUTPUTS:
                     03B8   547 ;
                     03B8   548 ;       RMBCHAN         RECEIVES THE CHANNEL NUMBER OF THE CHANNEL ASSIGNED
                     03B8   549 ;
                     03B8   550 ; COMPLETION CODES:
                     03B8   551 ;
                     03B8   552 ;       THE SAME ONES THAT ARE SUPPLIED BY THE $ASSIGN SYSTEM SERVICE
                     03B8   553 ;
                     03B8   554 ;--
                     03B8   555
                     03B8   556 ASSIGNRMB:
        3C   BB      03B8   557          PUSHR   #^M<R2,R3,R4,R5>        ; SAVE SOME REGISTERS
                     03BA   558
    55  5E   D0      03BA   559          MOVL    SP,R5                   ; SAVE STACK POINTER
                     03BD   560
                     03BD   561          ; CONVERT NUMBER TO ASCII STRING ON STACK
                     03BD   562
                     03BD   563 10$:     ; CONVERT NEXT DIGIT
            54   D4  03BD   564          CLRL    R4                      ; HIGH BITS OF DIVIDEND
    54  53  53   0A  7B  03BF   565          EDIV    #10,R3,R3,R4            ; QUO.-> R3   REM. -> R4
        7E  54   30  89  03C4   566          BISB3   #^X30,R4,-(SP)          ; CONVERT TO ASCII AND PUSH ON STACK
                 53  D5  03C8   567          TSTL    R3                      ; REPEAT?
                 F1  12  03CA   568          BNEQ    10$                     ; BR. IF YES
                     03CC   569
                     03CC   570          ; NOW PUSH 'MB' ONTO STACK
        7E   424D    8F  B0  03CC   571          MOVW    #^A'MB',-(SP)
                     03D1   572
                     03D1   573          ; NOW BUILD A STRING DESCRIPTOR ON STACK
        53   55      5E  C3  03D1   574          SUBL3   SP,R5,R3                ; OVERALL LENGTH OF STRING
                     6E  9F  03D5   575          PUSHAB  (SP)                    ; PUSH ADDRESS OF STRING
                     53  DD  03D7   576          PUSHL   R3                      ; PUSH LENGTH
             54      5E  D0  03D9   577          MOVL    SP,R4                   ; R4 POINTS TO STRING DESCRIPTOR
                     03DC   578
                     03DC   579          ; NOW ASSIGN THE CHANNEL
                     03DC   580          $ASSIGN_S       DEVNAM = (R4),-         ; DEVICE NAME
                     03DC   581                          CHAN = RMBCHAN          ; CHANNEL
                     03ED   582
             5E      55  D0  03ED   583          MOVL    R5,SP                   ; RESTORE STACK POINTER
```

```
3C  BA  03F0   584          POPR    #^M<R2,R3,R4,R5>        ; RESTORE REGISTERS
    05  03F2   585          RSB                             ; RETURN CODE IN R0 FROM $ASSIGN
        03F3   586
        03F3   587
        03F3   588
        03F3   589          .END    START
```

```
$$T1              =  00000001
ASSIGNRMB            000003B8  R      03
CHANBFR              00000130  R      02
CHANBFRDSC           00000010  R      03
CHANBFRSIZ        =  0000000C
CLKMODE           =  00000001
CLKSTATUS         =  00000141
CNTRLR_TBL           00000000  R      02
DEFCLKRATE        =  00000001
DEFPRESET         =  FFFFFF9C
DEVNFND           =  000000D6
DEVTBL               00000018  R      03
DIB$L_DEVDEPEND=  00000008
DMDT                 00000030  R      02
INITTBL              0000001A  R      02
IO$M_NOW             ********      X  03
IO$M_WRTATTN         ********      X  03
IO$_INITIALIZE       ********      X  03
IO$_LOADMCODE        ********      X  03
IO$_READVBLK         ********      X  03
IO$_SETCLOCK         ********      X  03
IO$_SETMODE          ********      X  03
IO$_STARTMPROC       ********      X  03
IO$_WRITEVBLK        ********      X  03
LA$S_MCTYPE       =  00000002
LA$S_PRESET       =  00000010
LA$S_RATE         =  00000003
LA$V_MCTYPE       =  00000001
LA$V_PRESET       =  00000010
LA$V_RATE         =  0000000D
LACHAN               00000018  R      02
LACTRLR              0000015E  R      02
LAIOSB               00000004  R      02
LANAME               0000015C  R      02
LANAMEDSC            00000008  R      03
LANAMESIZ         =  00000004
LOADMC               0000027A  R      03
LPA$$ADMCODE         ********      X  03
LPA$$DAMCODE         ********      X  03
LPA$$DMDT            ********      X  03
LPA$$MRMCODE         ********      X  03
MBX$B_CTRLR       =  00000004
MBX$B_MCTYPE      =  00000005
MBX$L_TYPE        =  00000000
MBX$W_RMBUNIT     =  00000006
MBXBFR               0000013C  R      02
MINMSGSIZ         =  00000008
PMBAST               000000C0  RG     03
PMBCHAN              00000014  R      02
PMBIOSB              0000000C  R      02
PMBNAM               0000002C  R      03
PMBNAMDSC            00000000  R      03
PMBNAMSIZ         =  0000000A
PROTMASK          =  00001100
PWRRECAST            000001DC  RG     03
RMBCHAN              00000016  R      02
SETCLOCK             0000037C  R      03
SS$_BADPARAM         ********      X  03
SS$_NORMAL           ********      X  03
START                00000036  RG     03
SYS$ASSIGN           ********     GX  03
SYS$CREMBX           ********     GX  03
SYS$DASSGN           ********     GX  03
SYS$DELMBX           ********     GX  03
SYS$GETCHN           ********     GX  03
SYS$HIBER            ********     GX  03
SYS$QIOW             ********     GX  03
SYS$SETPRA           ********     GX  03
SYS$SETSFM           ********     GX  03
```

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
_LPA$DATA	00000160 (352.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
_LPA$CODE	000003F3 (1011.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	37	00:00:00.10	00:00:00.45
Command processing	123	00:00:00.44	00:00:01.30
Pass 1	170	00:00:03.78	00:00:10.09
Symbol table sort	0	00:00:00.13	00:00:00.32
Pass 2	109	00:00:01.41	00:00:02.61
Symbol table output	7	00:00:00.06	00:00:00.06
Psect synopsis output	1	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	449	00:00:05.95	00:00:14.87

The working set limit was 1200 pages.
20688 bytes (41 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 121 non-local and 20 local symbols.
589 source lines were read in Pass 1, producing 26 object records in Pass 2.
22 pages of virtual memory were used to define 21 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	18

235 GETS were required to define 18 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:LALOADER/OBJ=OBJ$:LALOADER MSRC$:LALOADER/UPDATE=(ENH$:LALOADER)

0233 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

MELDR

MDL32

LALOAD
MAP

MDL32
MAP

XFLOADER
MAP

LALOADER
LIS

XFLOADER
LIS

LADAMCODE
LIS

LALOAD
LIS

MARBLI

MARBLI
MAP

LALOADER
MAP

LAAMRMCODE
LIS

LAADMCODE
LIS

MARBLI
LIS

LADMDT
LIS