

>txt AS -ls EPTTEST.MAC
 met ed alle FF (form feeds) verwijderd
 rlijn van txt is:

SI condensed
 ex M elik
 ex S set superscript
 ex A set linepitch at 5/60"
 ex L set left margin at 808
 ex G set double strike

+ blad 1 cm meer na - links in feeder gestoken

```

*****
;
; * EPTTEST
; *
; * A test program for the NdV EPROMgrammer.
; *
; * Enables the user to test and check his
; * EPROMgrammer card.
; *
*****
; Name: EPTTEST.MAC
;
; Written by: Nico de Vries
;            Mari Andriessenrade 49
;            2907 MA Capelle aan den IJssel
;            The Netherlands
;
; History:
;
; 13-04-1987 Start of work
; 28-05-1987 S command added
; 29-05-1987 MONE5 automatically loaded on start up
; 31-05-1987 Work on T command started
; 15-06-1987 Data bus and Vcc parts of T command working
; 23-06-1987 Program finished: release version 1.00
;
; Memory layout:
; 0000-00FF Zero page
; 0100-01FF Stack area
; 0200-0FFF This program
; 1000-8FFF RAM, unused
; 9000-9FFF Room for MONE5
; A000-BFFF Room for a DOS utility
; C000-DFFF D0865
; E000-E3FF I/O area
; E400-E7FF Scratchpad RAM for I065 and EXETIME
; E800-EFFF Video RAM
; F000-FFFF Boot-ROM
;
; Description of VIA and PIA use:
;
; VIA:
; port A: databus from/to EPROM
; Port B: PB0-PB2 EPROM type select
;         PB3 Switches Vcc from 5 to 6 Volts (pin 28 only)
;         PB4 Vcc on/off switch
;         PB5 Vpp on/off switch
;         PB6 OE drive
;         PB7 Program pulse, generated through timer 1
;         CB2 Vpp A/B select
;
; PIA:
; Port A: addresslines A0 through A7 to EPROM
; Port B: addresslines A8 through A15 to EPROM
;
; ** External Labels
;
3156 rev equ "1V" ;note: backwards!
3130 release equ "10" ;note: backwards!
E7B7 keypnt equ $e7b7 ;keyboard buffer pointer
F291 speed equ $f291 ;speed constant in I065
;
*****
; * EP.LIB
; *
; * External labels for EP.MAC
; *
*****
; Name: EP.LIB
;
; Written by: Nico de Vries
;            Mari Andriessenrade 49
;            2907 MA Capelle aan den IJssel
;            The Netherlands
;
E180 ebase equ $e180 ;set to base address of card
E180 via equ ebase
E180 vorb equ via ;VIA output register B (control lines)
E181 vorb equ via+1 ;VIA output register A (data bus to EPROM)
E182 vddrb equ via+2 ;VIA data direction register port B
E183 vddra equ via+3 ;VIA data direction register port A
E184 vt1cl equ via+4 ;VIA timer 1 latch/counter, lo
E185 vt1ch equ via+5 ;VIA timer 1 counter, hi
E186 vt1ll equ via+6 ;VIA timer 1 latch, lo
E187 vt1lh equ via+7 ;VIA timer 1 latch, hi
E188 vt2cl equ via+8 ;VIA timer 2 latch/counter, lo
E189 vt2ch equ via+9 ;VIA timer 2 counter, hi
E18A vsr equ via+10 ;VIA shift register
E18B vacr equ via+11 ;VIA auxilliary control register
E18C vpcr equ via+12 ;VIA peripheral control register
E18D vifr equ via+13 ;VIA interrupt flag register
E18E vier equ via+14 ;VIA interrupt enable register
E18F voranh equ via+15 ;VIA port A, no handshake
;
E190 pia equ ebase+16 ;base address of PIA
E190 pora equ pia ;PIA port A/data direction register A
E191 pcra equ pia+1 ;PIA control register A
E192 porb equ pia+2 ;PIA port B/data direction register B
E193 perb equ pia+3 ;PIA control register B
;
; ** DOS labels
;
3000 mon equ $9000 ;start address MONE5
3000 ramsize equ mon ;upper RAM limit
C005 doscom equ $c006 ;execute dos command pointed to by AY
C020 input equ $c020 ;get input character in A
C023 output equ $c023 ;print the character in A
C029 bufin equ $c029 ;get command into buffer
C02C getbuf equ $c02c ;read command from buffer
C02F orlf equ $c02f ;output CRLF
C032 spa equ $c032 ;output a space
C035 hnout equ $c035 ;print A as one or two hex nibbles
C038 prbyte equ $c038 ;print A as two hex nibbles
C03B prtext equ $c03b ;print string until zero byte
C041 loupch equ $c041 ;convert lower to upper case
;

```

```

;*** IO65 labels
;
E701 statog equ    $e701      ;inhibit status line update flag
F01B statoff equ  $f01b     ;switch status line off
F021 staton equ   $f021     ;switch status line on
F024 crsadr equ   $f024     ;move cursor to X,Y
F027 crsback equ  $f027     ;restore previous cursor position
;
;*** Page zero variables
;
0000          org    $80
;
0000          rcurr  res    2      ;current RAM address
;
;
0200          org    $200
;
;*** Initialise VIA and PIA, then set all pins
;*** of textool socket at zero volts
;
0200 A0 FF      eptest ldy    $fff      ;get value for DDR registers
0202 A9 00      lda      $000      ;and value for DDR access
0204 8D 91E1    sta      pora      ;set DDRA accessible
0207 8D 93E1    sta      porb      ;set DDRB accessible
020A 8C 90E1    sty      pora      ;set port A to outputs
020D 8C 92E1    sty      porb      ;set port B to outputs
0210 A9 04      lda      $200000100 ;get value for port access
0212 8D 91E1    sta      pora      ;set port A accessible
0215 8D 93E1    sta      porb      ;set port B accessible
0218 C8         iny              ;Y=0
0219 8C 90E1    sty      pora      ;set all addresslines
021C 8C 92E1    sty      porb      ;to zero
021F 98         tya              ;A=0
0220 88         dey              ;Y=$FF
0221 8C 03E1    sty      vddra      ;set VIA port A to outputs
0224 8D 8FE1    sta      voranh      ;set port A (databus) to zero volts
0227 8C 82E1    sty      vddrb      ;set VIA port B to outputs
022A 8D 80E1    sta      vorb      ;and set all controllines to zero
022D A9 80      lda      $210000000 ;set auxiliary control to
022F 8D 88E1    sta      vacr      ;Timer1 one shot mode with PB7 pulsing
0232 8D 8EE1    sta      vier      ;disable all interrupts
0235 A9 CC      lda      $211001100 ;set CB2, CA2 to output, lo
0237 8D 8CE1    sta      vpcr      ;CA1, CB1 negative edge
023A A9 00      lda      $0         ;get default type
023C 8D 700F    sta      eptyp      ;into current type
023F 20 250F    jsr      settyp      ;reflect on VIA
0242 20 3BC0    jsr      prtext      ;print sign-on message
0245 0C20455054 fcc      $0c, ' EPTTEST '
024A 45535420    fdb      rev
024E 5631        fdb      ' '
0250 2E          fcc      ' '
0251 3031        fdb      release
0253 1B690D204E fcc      $1b, 'i\r NdV Firmware ', $1b, '\n\r', 0
0258 6456204659
025D 726D776172
0262 6520186E0D
0267 00
0268 A9 0F      lda      $monstr>>8 ;get address
026A A0 63      ldy      $monstr&255 ;of dos command
026C 20 06C0    jsr      doscom      ;load monitor
026F 20 1B06    jsr      qtextpr     ;print command list
0272 20 3BC0    jsr      prtext      ;print
0275 0D36353232 fcc      '\r6522 VIA ', 0
027A 2056494120
027F 00
0280 AD 8FE1    lda      voranh      ;get VIA port A
0283 C9 E1      cmp      $voranh>>8 ;if bus in hiZ
0285 F0 35      beq      novia      ;report no VIA
0287 C9 00      cmp      $0         ;if not zero
0289 D0 42      bne      malvia      ;report defective VIA
028B A9 FF      lda      $fff      ;get
028D 8D 86E1    sta      vtill      ;value
0290 8D 85E1    sta      vtich      ;and start imer1
0293 AD 84E1    lda      vticl      ;get timer1 lo
0296 EA         nop              ;wait
0297 EA         nop              ;a
0298 EA         nop              ;while
0299 CD 84E1    cmp      vticl      ;if timer not running
029C F0 2F      beq      malvia      ;report faulty VIA
029E 20 3BC0    jsr      prtext      ;else
02A1 7072657365 fcc      'present and responding.', 0
02A6 6E7420616E
02AB 6420726573
02B0 706F6E6469
02B5 6E672E00
02B9 4C F502    jmp      piatst      ;and check PIA
02BC 20 3BC0    novia jsr      prtext ;print
02BF 6E6F742066 fcc      'not found.', 0
02C4 6F756E642E
02C9 00
02CA 4C F502    jmp      piatst      ;and test PIA
02CD 20 3BC0    malvia jsr      prtext ;print
02D0 666F756E64 fcc      'found, but it responds not properly.', 0
02D5 2C20627574
02DA 2063742072
02DF 6573706F6E
02E4 6473206E6F
02E9 742070726F
02EE 7065726C79
02F3 2E00
02F5 20 3BC0    piatst jsr      prtext ;print
02F8 0D36303231 fcc      '\r6821 PIA ', 0
02FD 2050494120
0302 00
0303 AD 90E1    lda      pora      ;get PIA port A
0306 C9 E1      cmp      $pora>>8 ;if bus in hiZ
0308 F0 2E      beq      nopia      ;report no PIA
030A C9 00      cmp      $0         ;if port not zero
030C D0 3C      bne      malpia      ;report bad PIA
030E AD 92E1    lda      porb      ;get PIA port A
0311 C9 E1      cmp      $porb>>8 ;if bus in hiZ
0313 F0 23      beq      nopia      ;report no PIA
0315 C9 00      cmp      $0         ;if port not zero
0317 D0 31      bne      malpia      ;report bad PIA
0319 20 3BC0    jsr      prtext      ;print
031C 7072657365 fcc      'present and responding.\r', 0
0321 6E7420616E
0326 6420726573
032B 706F6E6469
0330 6E672E0000

```

```

0335 4C 7303      jmp      nextcmd      ;and go get input
0338 20 38C0      nopia   jsr      prtext      ;print
0339 6E6F742066   fcc      'not found.\r',0
0340 6F756E642E
0345 0D00
0347 4C 7303      jmp      nextcmd      ;and get input
034A 20 38C0      molpia  jsr      prtext      ;print
034D 6E6F756E64   fcc      'found, but it responds not properly.\r',0
0352 2C20627574
0357 2063742072
035C 6573706F6E
0361 6473206E6F
0366 742070726F
036B 7065726C79
0370 2E0D00

;
;*** Command entry loop
;
0373 20 2FC0      i
nextcmd jsr      crlf      ;start on new line
0376 20 38C0      1      jsr      prtext      ;print prompt
0379 436F6D6D61   fcc      'Command: ',0
037E 6E643A2000
0383 20 20C0      2      jsr      input      ;get command character
0386 C9 0D        cmp      #$0d      ;if return only
0388 F0 F5        beq      2.b      ;ignore
038A 20 41C0      jsr      loupch      ;else convert to upper case
038D A2 07        ldx      #ncmds-1      ;set command counter
038F DD 4B0F      3      cmp      cmd,x      ;check if valid command
0392 F0 06        beq      4.f      ;if so, execute it
0394 CA          dex      ;else adapt counter
0395 10 F8        bpl      3.b      ;loop
0397 4C 8303      jmp      2.b      ;and get new input

;
039A 20 23C0      4      jsr      output      ;echo input
039D 20 20C0      jsr      input      ;get next character
03A0 C9 0D        cmp      #$0d      ;if return
03A2 F0 0E        beq      5.f      ;go execute command
03A4 C9 7F        cmp      #$7f      ;if not DEL
03A6 D0 F2        bne      4.b      ;or return, go wait for these
03A8 20 38C0      jsr      prtext      ;if DEL
03AB 08200800     fcc      $08,$20,$08,0 ;remove character
03AF 4C 8303      jmp      2.b      ;and go get new command
03B2 8A          txa      ;get index in A
03B3 0A          asla      ;multiply by two
03B4 AA          tax      ;get new index in X
03B5 A3 03        lda      *(nextcmd-1)>>8 ;get high byte of return
03B7 48          pha      ;address
03B8 A3 72        lda      *(nextcmd-1)&255 ;get lo too
03BA 48          pha      ;and complete command return address
03BB BD 540F      lda      cmdadr+1,x ;get high byte of command address
03BE 48          pha      ;on stack
03BF BD 530F      lda      cmdadr,x   ;get lo too
03C2 48          pha
03C3 60          rts      ;and execute command

;
;*** Command ?: show help text
;
03C4 4C 1B06      qstcmd jmp      qtextpr      ;print help text and exit

;
;*** Command M: call MON55 monitor
;
03C7 4C 0090      mcmd   jmp      mon      ;call monitor

;
;*** Command Q: exit to DOS
;
03CA 68          qcmd   pla      ;remove
03CB 68          pla      ;return address
03CC A9 0C        lda      #$0c      ;clear screen
03CE 4C 23C0      jmp      output      ;and exit to DOS

;
;*** Command S: report system speed
;
03D1 20 38C0      scmd   jsr      prtext      ;print text
03D4 0D446F6573   fcc      '\rDoes the system clock run on time? (Y/N)\r',0
03D9 2074686520
03DE 7379737465
03E3 6D20636C6F
03E8 636B207275
03ED 6E206F6E20
03F2 74696D653F
03F7 2020592F4E
03FC 230D00
03FF 20 20C0      jsr      input      ;get answer
0402 20 41C0      jsr      loupch      ;convert to upper case
0405 C9 4E        cmp      #'N'      ;if no,
0407 D0 31        bne      1.f      ;report
0409 20 38C0      jsr      prtext      ;message
040C 0D49206361   fcc      '\rI cannot tell the system clock speed then.\r',0
0411 6E6E6F7420
0416 7468656C20
041B 74686552073
0420 797374656D
0425 20636C6F63
042A 6B20737065
042F 6564207468
0434 656E2E0D00
0439 60          rts      ;and exit
043A 20 38C0      1      jsr      prtext      ;else print
043D 0D54686520   fcc      '\rThe system clock speed is ',0
0442 7379737465
0447 6D20636C6F
044C 636B207370
0451 6565642069
0456 732000
0459 AD 91F2      lda      speed      ;get clock constant from EPROM
045C C9 14        cmp      $20      ;if 1 MHz
045E D0 04        bne      2.f      ;
0460 A9 31        lda      $'1'      ;get a one
0462 D0 2B        bne      5.f      ;and print answer
0464 C9 28        cmp      $40      ;if 2 MHz
0466 D0 04        bne      3.f      ;get
0468 A9 32        lda      $'2'      ;a two
046A D0 23        bne      5.f      ;and go print it
046C C9 3C        cmp      $50      ;if not 3 MHz
046E D0 04        bne      4.f      ;report error
0470 A9 33        lda      $'3'      ;else get a three
0472 D0 1B        bne      6.f      ;and print result
0474 20 38C0      4      jsr      prtext      ;print message
0477 6E6F742061   fcc      'not a standard value.\r',0
047C 207374616E

```

```

0481 6461726420
0486 76616C7565
048B 2E0D00
048E 60
048F 20 23C0 6 rts ;and exit command
0492 20 38C0 jsr ;print speed
0495 204D487A2E jsr prtext ;and finish sentence
049A 0D00 fcc 'MHz.\r',0
049C 60 rts ;and exit command

;
;*** Command V: switch Vcc on or off.
;
049D 20 E20E vcmd jsr vcc tog ;toggle Vcc
04A0 20 050F jsr vcc 5 ;set it to 5 Volt
04A3 A9 00 lda $0 ;select
04A5 8D 700F sta epty ;type zero
04A8 AD 80E1 lda vorb ;get VIA control port
04AB 29 10 and $200010000 ;get Vcc bit
04AD F0 53 beq vex ;if no Vcc, exit
04AF 20 38C0 jsr prtext ;else print text
04B2 0D50696E73 fcc '\rPins 26 and 28 should be at 5 Volt, +/- 0.25 Volt.'
04B7 2032362061
04BC 6E64203238
04C1 2073686F75
04C6 6C64206265
04C8 2061742035
04D0 20566F6C74
04D5 2C202B2F2D
04DA 20302E3235
04DF 20566F6C74
04E4 2E
04E5 0D54686520 fcc '\rThe Vcc LED should be on.\r',0
04EA 566363204C
04EF 4544207368
04F4 6F756C6420
04F9 6265206F6E
04FE 2E0D00
0501 60 rts ;get next command
0502 20 38C0 vex jsr prtext ;print
0505 0D50696E73 fcc '\rPins 26 and 28 should carry a logic zero.'
050A 2032362061
050F 6E64203238
0514 2073686F75
0519 6C64206361
051E 7272792061
0523 206C6F6769
0528 63207A6572
052D 6F2E
052F 0D54686520 fcc '\rThe Vcc LED should be off.\r',0
0534 566363204C
0539 4544207368
053E 6F756C6420
0543 6265206F6E
0548 6E2E0D00
054C 60 rts ;and exit

;
;*** Command P: switch Vpp on or off.
;
054D 20 E50E pcmd jsr vpp tog ;toggle Vpp
0550 20 38C0 jsr prtext ;else print text
0553 0D54686520 fcc '\rThe Vpp LED should be ',0
0558 567070204C
055D 4544207368
0562 6F756C6420
0567 62652000
056B AD 80E1 lda vorb ;get VIA control port
056E 29 20 and $200100000 ;get Vpp bit
0570 F0 09 beq pex ;if no Vpp, exit
0572 20 38C0 jsr prtext ;else print text
0575 6F6E2E0D00 fcc 'on.\r',0
057A 60 rts ;get next command
057B 20 38C0 pex jsr prtext ;print message
057E 6F66662E0D fcc 'off.\r',0
0583 00
0584 60 rts ;and exit

;
;*** Command Z: zero volts on all textool pins.
;
0585 20 1607 zcmd jsr zeratxt ;zero volts on all pins
0588 20 38C0 jsr prtext ;else print text
058B 0D43686563 fcc '\rCheck if all Textool socket pins carry a logic zero or'
0590 6B20696620
0595 616C6C2054
059A 6578746F6F
059F 6C20736F63
05A4 6B65742070
05A9 696E732063
05AE 6172727920
05B3 61206C6F67
05B8 6963207A65
05BD 726F206F72
05C2 0D61726520 fcc '\rare at zero Volts. All LEDs should be off, except for the'
05C7 6174207A65
05CC 726F20566F
05D1 6C74732E20
05D6 416C6C204C
05DB 4544732073
05E0 686F756C64
05E5 206265206F
05EA 66662C2065
05EF 7863657074
05F4 20666F7220
05F9 746865
05FC 0D626F7474 fcc '\rbottom pin 1 indicator LED.\r',0
0601 6F6D207069
0606 6E20312069
060B 6E64696361
0610 746F72204C
0615 45442E0D00
061A 60 rts ;get next command

;
;*** Print helptext
;
061B 20 38C0 qtextpr jsr prtext ;print help text
061E 0D00546865 fcc '\rThe following commands are available:'
0623 20666F6C6C
0628 6F77696E67
062D 20636F6D60
0632 616E647320
0637 6172652061

```

```

0630 7661696C51
0641 626C653A
0645 0D
0646 0040202D20 fcc '\r'
064B 43616C6C20 fcc '\rM - Call MON65 monitor.'
0650 404F4E3635
0655 20606F6E69
065A 746F722E
065E 0051202D20 fcc '\rP - Toggle Vpp on/off.'
0663 546F67676C
0668 6520567070
066D 206F6E2F6F
0672 66662E
0675 0051202D20 fcc '\rQ - Quit program, exit to DOS.'
067A 5175697420
067F 70726F6772
0684 616D2C2065
0689 7069742074
068E 6F20444F53
0693 2E
0694 0053202D20 fcc '\rS - Show system speed.'
0699 53686F7720
069E 7379737465
06A3 6D20737065
06A9 65642E
06AB 0054202D20 fcc '\rT - Test individual pins.'
06B0 546F737420
06B5 696E646376
06BA 696475616C
06BF 2070696E73
06C4 2E
06C5 0056202D20 fcc '\rV - Toggle Vcc on/off.'
06CA 546F67676C
06CF 6520566363
06D4 206F6E2F6F
06D9 66662E
06DC 005A202D20 fcc '\rZ - Zero Volt on all textool pins.'
06E1 5A65726F20
06E6 566F6C7420
06EB 6F6E6C616C
06F0 6C20746578
06F5 746F6D6D20
06FA 70696E732E
06FF 003F202D20 fcc '\r? - Show this list.'
0704 53686F7720
0709 7468697320
070E 6C6973742E
0713 0000 fcc '\r',0
0715 60 rts ;and exit

;*** Zero volt on all textool pins
0716 A0 00
0718 8C 90E1 zerotxt ldy #0 ;set all
071B 8C 92E1 sty pora ;addresslines
071E 8C 8FE1 sty porb ;to zero
0721 8C 700F sty voranh ;zero databus on switch to outputs
0724 20 000F jsr eptyp ;select type zero
0727 20 F70E jsr vppoff ;switch Vpp off
072A 20 420F jsr nooe ;set EPROM in tri-state
072D 20 080F jsr dataout ;set databus to outputs
0730 60 jsr vccoff ;switch off Vcc
;and exit

;*** Command T: Test individual pins
0731 20 F10E tcmd jsr vccon ;switch Vcc on
0734 20 3BC0 jsr prtext ;print prompt
0737 00456E7465 fcc '\rEnter pin number (0=abort command): ',0
073C 722070696E
0741 206E756D62
0746 6572202830
074B 3D61626F72
0750 7420636F6D
0755 6D616E6429
075A 3A2000
075D 20 20C0 1 jsr input ;get input
0760 C9 0D cmp #0D ;if return only
0762 F0 F3 beq 1.b ;wait for input
0764 C9 30 cmp #'0' ;if below zero
0766 90 F5 bcc 1.b ;ignore input
0768 C9 3A cmp #'9'+1 ;if beyond 9,
076A B0 F1 bcs 1.b ;ignore too
076C 20 23C0 jsr output ;else echo input
076F 23 0F and #20001111 ;strip ASCII
0771 8D 710F sta temp ;save value
0774 20 20C0 2 jsr input ;get next character
0777 C9 7F cmp #7f ;if DEL
0779 D0 0A bne 3.f ;
077B 20 3BC0 jsr prtext ;clear input
077E 08200800 fcc B,$20,B,0
0782 4C 5D07 jmp 1.b ;and get new input
0785 C9 0D cmp #0D ;if return,
0787 F0 3B beq undert ;go get pin number below ten
0789 C9 30 cmp #'0' ;if below 0
078B 90 E7 bcc 2.b ;get new input
078D C9 3A cmp #'9'+1 ;if beyond 9
078F B0 E3 bcs 2.b ;get new input
0791 20 23C0 jsr output ;else echo input
0794 23 0F and #20001111 ;strip ASCII
0796 4B pha ;save input
0797 20 20C0 4 jsr input ;get input
079A C9 0D cmp #0D ;if return
079C F0 0F beq 5.f ;go calculate result
079E C9 7F cmp #7f ;if not DEL
07A0 D0 F5 bne 4.b ;get new input
07A2 68 pla ;if DEL, restore stack
07A3 20 3BC0 jsr prtext ;erase input
07A6 08200800 fcc B,$20,B,0
07AA 4C 7407 jmp 2.b ;and get new input
07AD AD 710F 5 lda temp ;get old result
07B0 0A aslo ;multiply by two
07B1 8D 710F sta temp ;store double
07B4 0A aslo ;get
07B5 0A aslo ;eight fold
07B6 6D 710F adc temp ;add two fold
07B9 8D 710F sta temp ;and store result
07BC 68 pla ;get units back
07BD 1B clc ;prepare for add
07BE 6D 710F adc temp ;add to result
07C1 8D 710F sta temp ;and store pin number

```

```

07C4 AD 710F   undert   lda      temp          ;get pin number
07C7 F0 2D     beq      tq          ;if zero, quit command
07C9 20 2FC0   jsr      crlf         ;start on new line
07CC AD 710F   lda      temp          ;else get number
07CF C9 1D     cmp      #28+1      ;if pin number is valid
07D1 90 29     bcc      pval         ;go print
07D3 20 38C0   jsr      prtext        ;else output error message
07D6 0D457226F fcc      '\rError: invalid pin number.\r',0
07D8 723A20696E
07E0 76616C6964
07E5 2070696E20
07EA 6E756D6265
07EF 722E0D00
07F3 4C 3107   jmp      tcmd          ;and get new input
07F6 20 2FC0   jsr      crlf         ;start on new line
07F9 4C 1607   jmp      zerotxt       ;quit with all pins at zero
07FC C9 0E     cmp      #14          ;if GND pin
07FE D0 31     bne      nognd        ;print message
0800 20 38C0   jsr      prtext        ;print message
0803 0D446F2079 fcc      '\rDo you really want to test the GND-pin??\r',0
0808 6F75207265
080D 616C6C7920
0812 77616E7420
0817 746F207465
081C 7374207468
0821 6520474E44
0826 2D70696E3F
082B 3F0D00
082E 4C 3107   jmp      tcmd          ;and get new input
0831 0A         nognd    asla      ;multiply pin number by two
0832 AA         tax          ;get offset
0833 A9 07     lda      $tcmd-1>>8    ;get hi of return
0835 48         pha          ;address
0836 A9 30     lda      $tcmd-1&255 ;and lo too
0838 48         pha          ;on stack
0839 BD A60E   lda      tpintab+1,x    ;get address
083C 48         pha          ;of
083D BD A50E   lda      tpintab,x      ;routine
0840 48         pha          ;on stack
0841 60         rts         ;and execute pin routine

0842 20 520E   ; tvpp    jsr      ppinnum      ;print pin number
0845 20 38C0   jsr      prtext        ;print
0848 2069732061 fcc      ' is a Vpp-pin.\r'
084D 205670702D
0852 70696E2E0D
0857 0D53686F72 fcc      '\rShort circuiting the pin to ground '
085C 7420636972
0861 6375697469
0866 6E67207468
086B 652070696E
0870 20746F2067
0875 726F756E64
087A 20
087B 73686F756C fcc      'should yield a current of 50 mA, +/- 10 mA.\r'
0880 6420796965
0885 6C64206120
088A 6375727265
088F 6E74206F66
0894 203530206D
0899 412C20202F
089E 2D20313020
08A3 6D412E0D
08A7 0D56707020 fcc      '\rVpp 1 is 12.5 Volt, +/- 0.5 Volt.'
08AC 3120697320
08B1 31322E3520
08B6 566F6C742C
08BB 20202F2D20
08C0 302E352056
08C5 6F6C742E
08C9 0D56707020 fcc      '\rVpp 2 is 21.0 Volt, +/- 0.5 Volt.'
08CE 3220697320
08D3 32312E3020
08D8 566F6C742C
08DD 20202F2D20
08E2 302E352056
08E7 6F6C742E
08EB 0D56707020 fcc      '\rVpp 3 is 25.0 Volt, +/- 0.75 Volt.\r'
08F0 3220697320
08F5 32352E3020
08FA 566F6C742C
08FF 20202F2D20
0904 302E373520
0909 566F6C742E
090E 0D
090F 00
0910 20 38C0   1 fcc      0
0913 0D456E7465 jsr      prtext
0918 7220737562 fcc      '\rEnter subcommand (Q, H, L, 1, 2, 3): ',0
091D 636F6D6D61
0922 6E64202051
0927 2C20482C20
092C 4C2C20312C
0931 20322C2033
0936 293A2000
093A 20 20C0   2 jsr      input          ;get input
093D 20 41C0   jsr      loupch        ;convert to upper case
0940 48         pha          ;save command
0941 AD 710F   lda      temp          ;get pin number
0944 C9 01     cmp      #1          ;if pin 1, do that
0946 D0 03     bne      ;
0948 4C FF09   jmp      vpp1          ;
094B C9 17     cmp      #23         ;if pin 23
094D F0 03     beq      vpp23        ;go do that
094F 4C 650A   jmp      vpp22        ;else do pin 22
0952 68         plo          ;get command back
0953 C9 51     cmp      #'Q'         ;if quit
0955 F0 54     beq      tvpp23q    ;exit
0957 C9 48     cmp      #'H'         ;if H
0959 F0 16     beq      tvpp23h    ;set pin high
095B C9 4C     cmp      #'L'         ;if L
095D F0 20     beq      tvpp231    ;set pin to zero
095F C9 31     cmp      #'1'         ;if 1
0961 F0 4E     beq      tvpp231    ;go set Vpp 12.5
0963 C9 32     cmp      #'2'         ;if 2
0965 D0 03     bne      6.f
0967 4C F609   jmp      tvpp232        ;go set Vpp 21
096A C9 33     cmp      #'3'         ;if 3
096C F0 2C     beq      tvpp233        ;go set Vpp 25
096E 4C 3A09   jmp      2.b           ;else get next command

```

```

0971 20 23C0 tvpp23h jsr output ;echo output
0974 20 000F jsr vppoff ;switch Vpp off
0977 A9 02 lda #2 ;select 2764
0979 20 140F jsr settype ;and set pin 1 H
097C A9 08 lda #00001000 ;get mask
097E 00 92E1 ora porb ;set A11 hi
0981 80 92E1 sta porb ;on socket
0984 4C 1009 jmp 1.b ;and get new command
0987 20 23C0 tvpp23l jsr output ;echo output
098A A9 02 lda #2 ;select 2764
098C 20 140F jsr settype ;and set pin 1 L
098F A9 F7 lda #211110111 ;get mask
0991 20 92E1 and porb ;set A11 lo
0994 80 92E1 sta porb ;on socket
0997 4C 1009 jmp 1.b ;and get new command
099A 20 23C0 tvpp233 jsr output ;echo input
099D 20 390F jsr vppa ;get Vpp a
09A0 20 F40E tvp232a jsr vppon ;switch Vpp on
09A3 A9 00 lda #0 ;select 2716
09A5 20 140F jsr settype ;and set Vpp 25
09A8 4C 1009 jmp 1.b ;and get new command
09AB 20 1697 tvpp23q jsr zerotxt ;exit with default
09AE 4C 2FC0 jmp crlf ;and on new line
09B1 20 23C0 tvpp23l jsr output ;echo input
09B4 20 3BC0 jsr prtxt ;print message
09B7 000D545865 fcc 'r\rThe current version cannot generate 12.5 Volt '
09BC 2063757272
09C1 656E742076
09C6 657273696F
09CB 6E2063616E
09D0 6E6F742067
09D5 656E657261
09DA 7465203132
09DF 2E3520566F
09E4 6C7420
09E7 6F6E207069 fcc 'on pin 23.\r',0
09EC 6E2032332E
09F1 0D00
09F3 4C 1009 jmp 1.b ;and get new command
09F6 20 23C0 tvpp232 jsr output ;echo input
09F9 20 320F jsr vppb ;get Vpp B
09FC 4C A009 jmp tvp232a ;and set Vpp 21

;
09FF 68 vpp1 pla ;get command back
0A00 C9 51 cmp #'Q' ;if quit
0A02 F0 58 beq tvpp1q ;exit
0A04 C9 48 cmp #'H' ;if H
0A06 F0 13 beq tvpp1h ;set pin high
0A08 C9 4C cmp #'L' ;if L
0A0A F0 1D beq tvpp1l ;set pin to zero
0A0C C9 31 cmp #'1' ;if 1
0A0E F0 24 beq tvpp11 ;go set Vpp 12.5
0A10 C9 32 cmp #'2' ;if 2
0A12 F0 31 beq tvpp12 ;go set Vpp 21
0A14 C9 33 cmp #'3' ;if 3
0A16 F0 36 beq tvpp13 ;go set Vpp 25
0A18 4C 3A09 jmp 2.b ;else get next command
0A1B 20 23C0 tvpp1h jsr output ;echo output
0A1E 20 000F jsr vppoff ;switch Vpp off
0A21 A9 02 lda #2 ;select 2764
0A23 20 140F jsr settype ;and set pin 1 H
0A26 4C 1009 jmp 1.b ;and get new command
0A29 20 23C0 tvpp1l jsr output ;echo output
0A2C A9 00 lda #0 ;select
0A2E 20 140F jsr settype ;2716 and set pin 1 L
0A31 4C 1009 jmp 1.b ;and get new command
0A34 20 23C0 tvpp11 jsr output ;echo input
0A37 20 320F jsr vppb ;select Vpp B
0A3A 20 F40E 4 jsr vppon ;switch Vpp on
0A3D A9 02 lda #2 ;select 2764
0A3F 20 140F jsr settype ;and set 12.5
0A42 4C 1009 jmp 1.b ;and get new command
0A45 20 23C0 tvpp12 jsr output ;echo input
0A48 20 390F jsr vppa ;select Vpp A
0A4B 4C 3A0A jmp 4.b ;and set Vpp 21
0A4E 20 23C0 tvpp13 jsr output ;echo input
0A51 20 F40E jsr vppon ;switch Vpp on
0A54 20 390F jsr vppa ;select Vpp A
0A57 A9 07 lda #7 ;select 2564
0A59 20 140F jsr settype ;and set Vpp 25
0A5C 4C 1009 jmp 1.b ;and get new command
0A5F 20 1697 tvpp1q jsr zerotxt ;exit with default
0A62 4C 2FC0 jmp crlf ;and on new line

;
0A65 68 vpp22 pla ;get command back
0A66 C9 51 cmp #'Q' ;if quit
0A68 F0 54 beq tvpp22q ;exit
0A6A C9 48 cmp #'H' ;if H
0A6C F0 13 beq tvpp22h ;set pin high
0A6E C9 4C cmp #'L' ;if L
0A70 F0 1D beq tvpp22l ;set pin to zero
0A72 C9 31 cmp #'1' ;if 1
0A74 F0 2A beq tvpp221 ;go set Vpp 12.5
0A76 C9 32 cmp #'2' ;if 2
0A78 F0 3A beq tvpp222 ;go set Vpp 21
0A7A C9 33 cmp #'3' ;if 3
0A7C F0 47 beq tvpp223 ;go set Vpp 25
0A7E 4C 3A09 jmp 2.b ;else get next command
0A81 20 23C0 tvpp22h jsr output ;echo output
0A84 20 F40E jsr vppon ;switch Vpp on
0A87 A9 00 lda #0 ;select 2716
0A89 20 140F jsr settype ;and set pin 1 H
0A8C 4C 1009 jmp 1.b ;and get new command
0A8F 20 23C0 tvpp22l jsr output ;echo output
0A92 20 000F jsr vppoff ;switch Vpp off
0A95 20 0B0F jsr oeon ;apply OE
0A98 A9 00 lda #0 ;select
0A9A 20 140F jsr settype ;2716 and set pin 1 L
0A9D 4C 1009 jmp 1.b ;and get new command
0AA0 20 23C0 tvpp221 jsr output ;echo input
0AA3 20 F40E jsr vppon ;switch Vpp on
0AA6 20 390F jsr vppa ;select Vpp A
0AA9 20 F70E jsr nooe ;remove OE
0AAC A9 05 lda #5 ;select 27512
0AAE 20 140F jsr settype ;and set 12.5
0AB1 4C 1009 jmp 1.b ;and get new command
0AB4 20 23C0 tvpp222 jsr output ;echo input
0AB7 20 320F jsr vppb ;select Vpp B
0ABA 20 F40E 5 jsr vppon ;switch Vpp on
0ABD A9 01 lda #1 ;select 2732

```

```

0ABF 20 140F      jsr      settype      ;and set Vpp 21
0AC2 4C 1009      jmp      1.b          ;and get new command
0AC5 20 23C0      tvpp223 jsr      output      ;echo input
0AC8 20 390F      jsr      vppa        ;get Vpp A
0ACB 4C BA0A      jmp      5.b          ;and set Vpp 25
0ACE 20 1507      tvpp22q jsr      zerotxt     ;exit with default
0AD1 4C 2FC0      jmp      crlf         ;and on new line

0AD4 20 520E      i          jsr      ppinnu      ;print pin number
0AD7 20 3BC0      tad          prtxt       ;print
0ADA 2069732061   fcc          ' is an address line pin.\r',0
0ADF 6E20615464
0AE4 7265737320
0AE9 6C696E6520
0AEE 70696E2E00
0AF3 00
0AF4 AD 00E1      lda          verb        ;get control lines
0AF7 29 F8        and          %%11111000    ;clear device select bits
0AF9 03 05        ora          %200000101    ;select 27512
0AFB 8D 00E1      sta          verb        ;and restore register
0AFE AD 710F      lda          temp        ;get pin number
0B01 C9 03        cmp          #3          ;if pin 3
0B03 D0 3C        bne          1.f        ;print
0B05 20 38C0      jsr      prtxt         ;text
0B08 0054686520   fcc          '\rThe activity LED should indicate '
0B0D 61E3746976
0B12 697479204C
0B17 4544207360
0B1C 6F756C6420
0B21 696E646963
0B26 61746520
0B2A 7468652073   fcc          'the state of the pin.\r',0
0B2F 7461746520
0B34 6F66207468
0B39 652070696E
0B3E 2E0000
0B41 20 38C0      1          jsr      prtxt
0B44 0D456E7465   fcc          '\rEnter subcommand (Q, H, L, T): ',0
0B49 7220737562
0B4E 636F6D6D61
0B53 6E64202051
0B58 2C20402C20
0B5D 4C2C205429
0B62 3A2000
0B65 20 20C0      2          jsr      input        ;get input
0B68 20 41C0      jsr      loupch       ;convert to upper case
0B6B 4B          pha          ;and save on stack
0B6C A0 00        ldy          #0          ;get index
0B6E AD 710F      lda          temp        ;get pin number
0B71 AA          tax          ;as index
0B72 BD 5E0E      lda          admtab,x    ;get mask
0B75 D0 05        bne          load        ;if mask is zero
0B77 A0 02        ldy          #2          ;get index to high
0B79 BD BA0E      lda          admtabh,x   ;and get new index
0B7C AA          tax          ;in X
0B7D 58          pla          ;get command letter back
0B7E C3 51        cmp          #'Q'        ;if quit
0B80 F0 2E        beq          tadq        ;exit
0B82 C3 54        cmp          #'T'        ;if T
0B84 F0 20        beq          tadl        ;go toggle pin
0B86 C3 4C        cmp          #'L'        ;if L
0B88 F0 0D        beq          tadl        ;set pin to zero
0B8A C3 48        cmp          #'H'        ;if not H
0B8C D0 D7        bne          2.b        ;get next command
0B8E 20 23C0      tadh          jsr      output      ;else echo input
0B91 8A          txa          ;get mask
0B92 19 90E1      ora          para,y      ;set data bit
0B95 D0 03        bne          3.f        ;then store and get new command
0B97 20 23C0      tadl          jsr      output      ;echo input
0B9A 8A          txa          ;get mask
0B9B 49 FF        eor          #$ff        ;invert it
0B9D 39 50E1      and          para,y      ;reset data bit
0BA0 39 50E1      3          sta          para,y    ;and store it
0BA3 4C 410B      jmp          1.b          ;and get new command
0BA6 20 23C0      tadl          jsr      output      ;echo input
0BA9 8A          txa          ;get mask
0BAA 59 90E1      eor          para,y      ;toggle data bit
0BAD 4C A00B      jmp          3.b          ;and store it
0BB0 4C 010C      tadq          jmp

0BB3 20 520E      i          jsr      ppinnu      ;print pin number
0BB6 20 3BC0      jsr      prtxt         ;print
0BB9 2069732061   fcc          ' is a data bus pin.\r',0
0BBE 2064617461
0BC3 2062757320
0BC8 70696E2E00
0BCD 00
0BCE 20 3BC0      1          jsr      prtxt
0BD1 0D456E7465   fcc          '\rEnter subcommand (Q, H, L, T, I): ',0
0BD6 7220737562
0BDB 636F6D6D61
0BE0 6E64202051
0BE5 2C20402C20
0BEA 4C2C20542C
0BEF 2049293A20
0BF4 00
0BF5 20 20C0      2          jsr      input        ;get input
0BF8 20 41C0      jsr      loupch       ;convert to upper case
0BFB 4B          pha          ;save command
0BFC AD 710F      lda          temp        ;get pinnumber
0BFF 38          sec          ;prepare for subtract
0C00 E9 0B        sbc          #11         ;convert to index
0C03 AA          tax          ;in X
0C06 BD 790E      lda          damtab,x    ;get mask
0C09 AA          tax          ;in X
0C0B 58          pla          ;get comand back
0C0D C9 51        cmp          #'Q'        ;if quit
0C0F F0 75        beq          tdaq        ;exit
0C11 C9 49        cmp          #'I'        ;if I
0C13 F0 2E        beq          tda1        ;test input
0C15 C9 4C        cmp          #'L'        ;if L
0C17 F0 11        beq          tda1        ;set pin to zero
0C19 C9 54        cmp          #'T'        ;if T
0C1B F0 1C        beq          tdat        ;go toggle pin
0C1D C9 48        cmp          #'H'        ;if not H
0C1F D0 D9        bne          2.b        ;get next command
0C21 20 23C0      tdah          jsr      output      ;else echo input
0C24 8A          txa          ;get mask
0C27 00 09        ora          varanh      ;set data bit
0C29 00 09        bne          3.f        ;then store and get new command

```

```

0C25 20 23C0   tda1   jsr    output      ;echo input
0C28 8A        txa      ;get mask
0C29 49 FF      eor      ;invert it
0C2B 2D 8FE1    and      voranh    ;reset data bit
0C2E 8D 8FE1    sta      voranh    ;and store it
0C31 4C CE0B    jmp      1.b        ;and get new command
0C34 20 23C0   tdat    jsr    output      ;echo input
0C37 8A        txa      ;get mask
0C38 4D 8FE1    eor      voranh    ;toggle data bit
0C3B 4C 2E0C    jmp      3.b        ;and store it
0C3E 20 23C0   tda1   jsr    output      ;echo input
0C41 8A        txa      ;get mask
0C42 8D 720F    sta      temp+1    ;save it
0C45 49 FF      eor      $fff      ;invert it
0C47 2D 83E1    and      vddra     ;set data bit as input
0C4A 8D 83E1    sta      vddra     ;and set register
0C4D 20 2FC0    jsr      crlf      ;start on next line
0C50 4D B7E7    lda      keypnt    ;get keyboard buffer pointer
0C53 48        pha      ;on stack
0C54 68        pla      ;get old pointer
0C55 48        pha      ;restore stack
0C56 CD 87E7    cmp      keypnt    ;if a key
0C59 D0 18      bne      tdaiq     ;exit
0C5B A3 0D      lda      $f0d     ;else get a CR
0C5D 20 23C0    jsr      output    ;start at beginning of line
0C60 4D 8FE1    lda      voranh    ;get data bus
0C63 2D 720F    and      temp+1    ;throw unnecessary bits away
0C66 F0 03      beq      prl        ;if zero, go print L
0C68 A9 48      lda      $'H'      ;get a H
0C6A 2C        fcc      $2c      ;skip next instruction
0C6B A9 4C      lda      $'L'      ;get an L
0C6D 20 23C0    jsr      output    ;print it
0C70 4C 540C    jmp      4.b        ;else loop
0C73 69        pla      ;clean stack
0C74 20 20C0    jsr      input      ;read key
0C77 20 420F    jsr      dataout    ;data bus to outputs
0C7A C8        iny      ;and
0C7B 8C 8FE1    sty      voranh    ;to zero volt
0C7E 4C CE0B    jmp      1.b        ;and get new command
0C81 20 420F    jsr      dataout    ;set data bus to outputs
0C84 88        dey      ;and
0C85 8C 8FE1    sty      voranh    ;to zero volt
0C88 4C 2FC0    jmp      crlf      ;and exit on new line

0C8B 20 520E    ;       jsr      ppinnum    ;print pin number
0C8E 20 38C0    ;       jsr      prtext     ;print
0C91 2069732061 ;       fcc      ' is a control pin.\r',0
0C96 20636F6E74
0C9B 726F6C2070
0CA0 636E2E0D00
0CA5 AD 80E1    lda      vorb      ;get control lines
0CA8 29 F8      and      $Z11111000    ;clear device select bits
0CAA 09 06      ora      $Z00000110 ;select 2532
0CAC 8D 80E1    sta      vorb      ;and restore register
0CAF 20 38C0    jsr      prtext     '\rEnter subcommand (Q, H, L, T): ',0
0CB2 0D456E7465
0CB7 7220737562
0CBC 636F6D6D61
0CC1 6E64202051
0CC6 2C20482C20
0CCB 4C2C205429
0CD0 3A2000    ;
0CD3 20 20C0    2       jsr      input      ;get input
0CD6 20 41C0    jsr      loupch     ;convert to upper case
0CD9 C3 51      cmp      $'Q'        ;if quit
0CDB F0 A4      beq      tdaq      ;exit
0CDD C3 54      cmp      $'T'        ;if T
0CDF F0 20      beq      tcont     ;go toggle pin
0CE1 C9 4C      cmp      $'L'        ;if L
0CE3 F0 0E      beq      tconl     ;set pin to zero
0CE5 C9 48      cmp      $'H'        ;if not H
0CE7 D0 EA      bne      2.b        ;get next command
0CE9 20 23C0    tconh    jsr      output    ;else echo input
0CEC A9 08      lda      $Z00001000 ;get mask
0CEE 0D 32E1    ora      porb      ;set data bit
0CF1 D0 08      bne      3.f        ;then store and get new command
0CF3 20 23C0    tconl    jsr      output    ;echo input
0CF6 A9 F7      lda      $Z11110111 ;get mask
0CF8 2D 32E1    and      porb      ;reset data bit
0CFB 8D 32E1    3       sta      porb      ;and store it
0CFE 4C AF0C    jmp      1.b        ;and get new command
0D01 20 23C0    tcont    jsr      output    ;echo input
0D04 A9 08      lda      $Z00001000 ;get mask
0D06 4D 32E1    eor      porb      ;toggle data bit
0D09 4C FB0C    jmp      3.b        ;and store it

0D0C 20 520E    ;       jsr      ppinnum    ;print pin number
0D0F 20 38C0    ;       jsr      prtext     ;print
0D12 2069732061 ;       fcc      ' is a supply pin.\r'
0D17 2073757070
0D1C 6C79207069
0D21 6E2E0D
0D24 0D53606F72
0D29 7420636972
0D2E 6375637469
0D33 6E67207468
0D38 652070696E
0D3D 20746F2067
0D42 726F756E64
0D47 20
0D48 73686F756C
0D4D 6420736965
0D52 6C64206120
0D57 6375727265
0D5C 6E74206F66
0D61 203230302B
0D66 6D412C202B
0D6B 2F2D203230
0D70 206D412E0D
0D75 00
0D76 AD 710F    lda      temp      ;get pin number
0D79 C9 1C      cmp      $28        ;if pin 28
0D7B F0 29      beq      tsup6      ;go test 6 volt
0D7D 20 38C0    jsr      prtext     ;else print
0D80 0D54657374 ;       fcc      '\rTest this pin using the V command.\r',0
0D85 2074686973
0D8A 2070696E20
0D8F 7573696E67
0D94 2074686E520
0D99 5620636F6D

```

```

009E 6D616E642E
0DA3 0D00
0DA5 60
0DA6 20 3BC0      tsup6      rts          ;and exit
0DA9 0D496E7075    jar          ;print
0DAE 7420737061    fcc          '\rInput spaces to toggle between 5 and 6 Volt.\r'
0DB3 6365732074
0DB8 6F20746F67
0DBD 676C652062
0DC2 6574776565
0DC7 6E20352061
0DCC 6E64203620
0DD1 566F6C742E
0DD6 00
0DD7 3620566F6C    fcc          '6 Volt means 6.0 Volt, +/- 0.25 Volt.\r'
0DDC 74206D6561
0DE1 6E7320362E
0DE6 3020566F6C
0DEB 742C20202F
0DF0 2D20302E32
0DF5 3520566F6C
0DFA 742E00
0DFD 4F74686572    fcc          'Other character than space gets you back '
0E02 2063686172
0E07 6163746572
0E0C 207468616E
0E11 2073706163
0E16 6520676574
0E1B 7320756F75
0E20 206261636B
0E25 20
0E26 746F207069    fcc          'to pin number input.\r',0
0E2B 6E206E756D
0E30 6265722063
0E35 6E7075742E
0E3A 0D00
0E3C 20 20C0      2          jsr          input          ;get input
0E3F C9 20          cmp          #$20          ;if not space
0E41 00 06          bne          tsupq          ;exit
0E43 20 DF0E        jsr          inttog          ;else toggle Vcc
0E46 4C 3C0E        jmp          2.b            ;and loop
0E49 A9 F7          tsupq      lda          $211110111    ;quit, get mask
0E4B 2D 00E1        and          verb          ;set 5 volt Vcc
0E4E 8D 00E1        sta          verb          ;store value
0E51 60            rts          ;and exit

;*** Print 'Pin ' plus pin number in decimal
0E52 20 3BC0
0E55 0D50696E20    ppinnum jsr          prtext          ;print
0E5A 00            fcc          '\rPin ',0          ;and:

;*** Print temp in decimal
0E5B AD 710F        ptempd  lda          temp          ;get temp variable, and:

;*** Print A in decimal
0E5E C9 0A          decpr    cmp          #10          ;if below 10
0E60 90 09          bcc          printd          ;go print
0E62 C9 14          cmp          #20          ;if below 20
0E64 90 02          bcc          tentw          ;add 6
0E66 63 05          adc          #$05          ;else add 12
0E68 18            clc          ;else prepare for add
0E69 63 06          adc          #$06          ;convert to decimal
0E6B 4C 35C0        printd  jmp          hnout          ;print number

;*** Table of masks for address and data lines
;(zero means look in second table)
0E6E 00            admtab  fcb          0            ;dummy, not used
0E6F 00            fcb          2000000000        ;pin 1, A15
0E70 00            fcb          2000000000        ;pin 2, A12
0E71 00            fcb          2100000000        ;pin 3, A7
0E72 40            fcb          2010000000        ;pin 4, A6
0E73 20            fcb          2001000000        ;pin 5, A5
0E74 10            fcb          2000100000        ;pin 6, A4
0E75 08            fcb          2000010000        ;pin 7, A3
0E76 04            fcb          2000001000        ;pin 8, A2
0E77 02            fcb          2000000100        ;pin 9, A1
0E78 01            fcb          2000000001        ;pin 10, A0
0E79 01            damtab  fcb          2000000001        ;pin 11, D0
0E7A 02            fcb          2000000010        ;pin 12, D1
0E7B 04            fcb          2000000100        ;pin 13, D2
0E7C 00            fcb          2000000000        ;dummy entry for pin 14
0E7D 08            fcb          2000010000        ;pin 15, D3
0E7E 10            fcb          2000100000        ;pin 16, D4
0E7F 20            fcb          2001000000        ;pin 17, D5
0E80 40            fcb          2010000000        ;pin 18, D6
0E81 80            fcb          2100000000        ;pin 19, D7
0E82 00            fcb          2000000000        ;pin 20, CE
0E83 00            fcb          2000000000        ;pin 21, A10
0E84 00            fcb          2000000000        ;pin 22, Vpp/0E
0E85 00            fcb          2000000000        ;pin 23, Vpp/A11
0E86 00            fcb          2000000000        ;pin 24, A9
0E87 00            fcb          2000000000        ;pin 25, A8
0E88 00            fcb          2000000000        ;pin 26, supply/A13
0E89 00            fcb          2000000000        ;pin 27, A14
0E8A 00            admtabh fcb          2000000000        ;pin 28, supply

;*** Table of masks for addresslines, hi
0E8B 00            fcb          2100000000        ;pin 1, A15
0E8C 10            fcb          2000100000        ;pin 2, A12
0E8D 00            fcb          2000000000        ;pin 3, A7
0E8E 00            fcb          2000000000        ;pin 4, A6
0E8F 00            fcb          2000000000        ;pin 5, A5
0E90 00            fcb          2000000000        ;pin 6, A4
0E91 00            fcb          2000000000        ;pin 7, A3
0E92 00            fcb          2000000000        ;pin 8, A2
0E93 00            fcb          2000000000        ;pin 9, A1
0E94 00            fcb          2000000000        ;pin 10, A0
0E95 00            fcb          2000000000        ;pin 11, D0
0E96 00            fcb          2000000000        ;pin 12, D1
0E97 00            fcb          2000000000        ;pin 13, D2
0E98 00            fcb          2000000000        ;dummy entry for pin 14
0E99 00            fcb          2000000000        ;pin 15, D3
0E9A 00            fcb          2000000000        ;pin 16, D4
0E9B 00            fcb          2000000000        ;pin 17, D5
0E9C 00            fcb          2000000000        ;pin 18, D6
0E9D 00            fcb          2000000000        ;pin 19, D7

```

```

0E9E 00      fcb      200000000      ;pin 20, CE
0E9F 04      fcb      200000100      ;pin 21, A10
0EA0 00      fcb      200000000      ;pin 22, Vpp/OE
0EA1 08      fcb      200001000      ;pin 23, Vpp/A11
0EA2 02      fcb      200000010      ;pin 24, A9
0EA3 01      fcb      200000001      ;pin 25, A8
0EA4 20      fcb      200100000      ;pin 26, supply/A13
0EA5 40      tpintab fcb      201000000      ;pin 27, A14
0EA6 00      fcb      200000000      ;pin 28, supply
;
;*** Table of addresses for pin test
;
0EA7 4108      fdb      tvpp-1      ;pin 01
0EA9 D30A      fdb      tad-1      ;pin 02, A12
0EAB D30A      fdb      tad-1      ;pin 03, A7
0EAD D30A      fdb      tad-1      ;pin 04, A6
0EAF D30A      fdb      tad-1      ;pin 05, A5
0EB1 D30A      fdb      tad-1      ;pin 06, A4
0EB3 D30A      fdb      tad-1      ;pin 07, A3
0EB5 D30A      fdb      tad-1      ;pin 08, A2
0EB7 D30A      fdb      tad-1      ;pin 09, A1
0EB9 D30A      fdb      tad-1      ;pin 10, A0
0EBB B20B      fdb      tda-1      ;pin 11, D0
0EBD B20B      fdb      tda-1      ;pin 12, D1
0EBF B20B      fdb      tda-1      ;pin 13, D2
0EC1 0000      fdb      0      ;dummy entry for pin 14
0EC3 B20B      fdb      tda-1      ;pin 15, D3
0EC5 B20B      fdb      tda-1      ;pin 16, D4
0EC7 B20B      fdb      tda-1      ;pin 17, D5
0EC9 B20B      fdb      tda-1      ;pin 18, D6
0ECB B20B      fdb      tda-1      ;pin 19, D7
0ECD 8A0C      fdb      tcon-1      ;pin 20, CE
0ECF D30A      fdb      tad-1      ;pin 21, A10
0ED1 4108      fdb      tvpp-1      ;pin 22
0ED3 4108      fdb      tvpp-1      ;pin 23
0ED5 D30A      fdb      tad-1      ;pin 24, A9
0ED7 D30A      fdb      tad-1      ;pin 25, A8
0ED9 0B0D      fdb      tsup-1      ;pin 26, supply/A13
0EDB D30A      fdb      tad-1      ;pin 27, A14
0EDD 0B0D      fdb      tsup-1      ;pin 28, supply
;
;*** Toggle Vcc between 5 and 5 volt
0EDF A9 08      inttog  lda      $200001000      ;get mask
0EE1 2C          fcc      $2c      ;skip next instruction
;
;*** Toggle Vcc on/off
0EE2 A9 10      vactog  lda      $200010000      ;get mask
0EE4 2C          fcc      $2c      ;skip next instruction
;
;*** Toggle Vpp on/off
0EE5 A9 20      vpptog  lda      $200100000      ;get mask
0EE7 4D 80E1      eor      vorb      ;toggle bit
0EEA 8D 80E1      sta      vorb      ;in VIA
0EED 60          rts      ;then exit
;
;*** Raise Vcc to 5 Volts
0EEE A9 08      vcc6    lda      $200001000      ;get mask
0EEF 2C          fcc      $2c      ;skip next instruction
;
;*** Switch Vcc on
0EF1 A9 10      vaccon  lda      $200010000      ;get mask
0EF3 2C          fcc      $2c      ;skip next instruction
;
;*** Switch Vpp on
0EF4 A9 20      vppon   lda      $200100000      ;get mask
0EF6 2C          fcc      $2c      ;skip next instruction
;
;*** Set OE high
0EF7 A9 40      nooe    lda      $201000000      ;get mask
0EF9 0D 80E1      ora      vorb      ;toggle bit
0EFC 8D 80E1      sta      vorb      ;in VIA
0EFF 60          rts      ;then exit
;
;*** Switch Vpp off
0F00 A9 DF      vppoff  lda      $211011111      ;get mask
0F02 20 0D0F      jsr      rsbit      ;switch Vpp off, then:
;
;*** Set Vcc to 5 Volts
0F05 A9 F7      vcc5    lda      $211110111      ;get mask
0F07 2C          fcc      $2c      ;skip next instruction
;
;*** Switch Vcc off
0F08 A9 EF      vccoff  lda      $211101111      ;get mask
0F0A 2C          fcc      $2c      ;skip next instruction
;
;*** Set OE low
0F0B A9 BF      oeon    lda      $210111111      ;get mask
0F0D 2D 80E1      rsbit   and      vorb      ;reset bit
0F10 8D 80E1      sta      vorb      ;in VIA
0F13 60          rts      ;then exit
;
;*** Settype: move A to type bits in VIA
0F14 48          settype pha      ;save mask
0F15 AD 80E1      lda      vorb      ;get lines
0F18 29 F8      and      $211111000      ;remove type
0F1A 8D 80E1      sta      vorb      ;restore register
0F1D 68          pla      ;get mask
0F1E 0D 80E1      ora      vorb      ;add to register
0F21 8D 80E1      sta      vorb      ;set type
0F24 60          rts      ;and exit
;
;*** Set EPROM type in A and corresponding Vpp
0F25 AD 700F      settype lda      eptyp      ;get type
0F28 29 07      and      $200000111      ;remove unneeded bits
0F2A 20 140F      jsr      settype      ;set device type in VIA
0F2D 2C 700F      bit      eptyp      ;get Vpp flag in N
0F30 10 07      bpl      vppa      ;and:

```

```

;*** Set Vpp B
;
0F32 AD BCE1 vppb lda vpcr ;get CB2 control
0F35 09 20 ora $200100000 ;set CB2
0F37 D0 05 bne 1.f ;to one and exit

;*** Set Vpp A
;
0F39 AD BCE1 vppa lda vpcr ;get CB2 control
0F3C 29 DF and $11011111 ;reset CB2
0F3E 8D BCE1 1 sta vpcr ;to zero
0F41 60 rts ;and exit

;*** Set databus to outputs
;
0F42 A0 FF dataout ldy $$ff ;get value
0F44 2C fcc $2c ;skip next instruction

;*** Set databus to inputs
;
0F45 A0 00 datain ldy $0 ;get value
0F47 8C 83E1 sty vddra ;set port
0F4A 60 rts ;and exit

;*** Table of available commands
;
0F4B 4D cmds fcc 'M' ;call monitor
0F4C 50 fcc 'P' ;toggle Vpp
0F4D 51 fcc 'Q' ;quit, exit to DOS
0F4E 53 fcc 'S' ;report system speed
0F4F 54 fcc 'T' ;test pins
0F50 56 fcc 'V' ;toggle Vcc
0F51 5A fcc 'Z' ;zero volt on all pins
0F52 3F fcc '?' ;show help file
0000 ncmds equ $-cmds ;number of commands

;*** Table of addresses of commands
;
0F53 C603 cmdadr fdb mcmd-1
0F55 4C05 fdb pcmd-1
0F57 C903 fdb qcmd-1
0F59 D003 fdb scmd-1
0F5B 3007 fdb tcmd-1
0F5D 9C04 fdb vcmd-1
0F5F 8405 fdb zcmd-1
0F61 C303 fdb qstcmd-1

0F63 4C4F0533A monstr fcc 'LO S:MONITOR',0
0F6B 4D4F4E4954
0F6D 4F5200

0F70 eptyp res 1
0F71 temp res 2

0200 end eptest

```

global labels

adatab	0EE6	admtabh	0E8A	bufin	C029	cmdadr	0F53	cmds	0F4B
orlf	C02F	crsadr	F024	crsback	F027	dantab	0E79	datain	0F45
dataout	0F42	decpr	0E5E	doscom	C006	epbase	E180	eptest	0200
eptyp	0F70	getbuf	C02C	hnout	C035	input	C020	inttog	0EDF
keypnt	E7B7	load	0B7C	loupch	C041	malpia	034A	malvia	02CD
mcmd	03C7	mon	9000	monstr	0F63	ncmds	0000	nextcmd	0373
nognd	0B31	nooe	0EF7	nopia	0338	novia	02BC	oeon	0F0B
output	C023	pcmd	054D	pcra	E191	pcrb	E193	pex	0578
pia	E190	piatst	02F5	pora	E190	porb	E192	ppinnum	0E52
prbyte	C038	printd	0EE8	pri	0CEB	prtext	C038	ptempd	0E58
pval	07FC	qcmd	03CA	qstcmd	03C4	qtextpr	061B	ramsize	9000
rcurr	0080	release	3130	rev	3156	rsbit	0F0D	scmd	03D1
settyp	0F25	settype	0F14	spa	C032	speed	F291	statoff	F01B
staton	F021	stattog	E701	tad	0AD4	tadh	0B8E	tadl	0B97
tadq	0B80	tadt	0BA6	tcmd	0731	tcon	0C88	tconh	0CE9
tconl	0CF3	tcont	0D01	tda	0BB3	tdah	0C1C	tdai	0C3E
tdaiq	0C73	tdal	0C25	tdaq	0CB1	tdat	0C34	temp	0F71
tenbw	0E68	tpintab	0EA5	tq	07F6	tsup	0D0C	tsup6	0DA6
tsupq	0E49	tvpp232a	09A0	tvpp	0842	tvpp11	0A34	tvpp12	0A45
tvpp13	0A4E	tvpp1h	0A18	tvpp11	0A29	tvpp1q	0A5F	tvpp221	0AA0
tvpp222	0A8A	tvpp223	0ACS	tvpp22h	0A81	tvpp221	0A8F	tvpp22q	0ACE
tvpp231	09B1	tvpp232	09F6	tvpp233	099A	tvpp23h	0971	tvpp231	09B7
tvpp23q	09AB	undert	07C4	vacr	E18B	vcc5	0F05	vcc6	0EEE
vccoff	0F08	vcon	0EF1	vcctog	0EE2	vcmd	0490	vddra	E1B3
vddrb	E182	vex	0502	via	E180	vier	E18E	vifr	E1B0
vora	E181	voranh	E18F	vorb	E180	vpcr	E18C	vpp1	09FF
vpp22	0A65	vpp23	0952	vppa	0F39	vppb	0F32	vppoff	0F00
vppon	0EF4	vpptog	0EE5	vsr	E18A	vt1ch	E185	vt1cl	E1B4
vt1lh	E187	vt1ll	E186	vt2ch	E189	vt2cl	E188	zcmd	0585
zerotxt	0716								

Errors detected: 0