

Sources for MIDISLAVE V1.1

This program was coded by King Fisher/TRIAD.

The original sources are in TASS format. A zipped d64 with these sources is available here: [midisl11.zip](#). This archive also contains some binary files to include, if you plan to actually assemble this program, rather than just learn from it.

The program is split into several sources, and each source file is given in its own section below in no special order. The memory locations for the program files goes like this:

```

10 -----
11 MEMORY ALLOCATIONS FOR THE MIDISLAVE
12 -----
13 $0801-$0C00 MAIN MENU CODE
14 $0C00-$1000 MIDIMODE SCREEN
15 $1000-$1180 NOTES IN TEXT
16 $1180-$1480 MIDI BUFFER
17 $1480-$1E00 FREE!
18 $1E00-$2D00 MIDISLAVE MAIN CODE
19 $2D00-$4000 SOUND EDITOR CODE
20 $4000-$C000 SOUND PRESETS
21 $C000-      MORE MAIN ROUTINES
22 $CF00-$D000 DUMMY ZEROPAGE
23 $D000-$E000 FREE!
24 $E000-$F000 SINUS WAVES
25 $F000-$F400 SOUND EDITOR SCREEN
26 $F400-$F800 MAIN MENU SCREEN
27 $F800-$FFFA FREE!
28 -----

```

"MAIN39.S"

```

;-----
;
;          *= $0801
;-----;
;          BASIC KICKS! LAMEX RULAR!          ;
;-----;
;
;          .byte $0b,$08,$0a,$00,$9e,$32
;          .byte $30,$36,$31,$00,$00,$00
;          jmp start
;          jmp retfromidi
;          jmp retfromsound
;-----

```

```

i2      jmp bang
nmi     pha
        lda $dd0d
        and #1
        beq i2          ;jump if it wasn't a "Timer A Interrupt (RS232)"
mod4    lda $de06      ;Read from MIDI interface.. - Status Register Address
        lsr a          ;Shift down..
        bcc i2         ;bail out if lsb wasn't set.. (Other source of NMI
such as restore or so..)
        txa           ;If we're here we got: "Bit 0      Receive DATA
register Full (RDRF)"
        pha
;-----;
;$f7 = Reset if not 0      ;
;$f8 = Indicate activity  ;
;$f9 = 00 don't buffer 01 buffer ;
;$fa = Buffer pointer      ;
;$fd = # of data bytes for command ; <- Seems to count downwards..
;$fe = old #              ;
;$ff = Channel            ;
;$fb,$fc = data short buffer ; <- $FB is the command
;-----;
mod5    ldx $de07      ;!!! read from Receive Data Register Address
        txa
        bpl databyte_received ;Was data!
        cpx #$f0
        bcs system
        and #$0f        ;lsb 4 bits is chan number..
        cmp $ff         ;Is this on the currently active channel?
        bne nullify_shortbuff ;Ignore??????
        txa
        and #$f0        ;
        sta $fb         ;data short buffer (1/2)
        and #$e0        ;
        cmp #$c0        ;Prg change - ONLY ONE DATABYTE FOR THIS ONE AND
AFTERTOUCHE..
        beq onebytedata ;or chanpress
twobytedata:
        lda #2          ;TWO DATABYTES FOR ALL COMMANDS APART FROM THE PRG
CHANGE COMMAND..
        sta $fd         ;# of data bytes for command
        sta $fe         ;old #
        jmp activity_indicate_and_quit
onebytedata:
        lda #1          ;# of data bytes for command
        sta $fd         ;# of data bytes for command
        sta $fe         ;old #
        jmp activity_indicate_and_quit
databyte_received:
        ldx $fd         ;# of data bytes for command

```

```

    bne g3
    ldx $fe          ;old #
    beq activincateandquit
    cpx #1
    beq g6
    dex
    stx $fd          ;set 0 # of data bytes for command
    sta $fc          ;STORE DATABYTE..
activincateandquit:
    jmp activity_indicate_and_quit          ;Not valid

g3    cpx #1
    bne fd_is_not_0_or_1
    ;If we're here $fd was equal to 1.
g6    ldx $f9          ;00 don't buffer 01 buffer
    beq activity_indicate_and_quit
    ;-- PUSH NEW COMMAND + DATA ON THE THREE BYTE DEEP BUFFER STACK..
    ldx $fa          ;BUFFER POINTER
    sta $1380,x      ;Data2/1!
    lda $fb
    sta $1180,x
    lda $fc
    sta $1280,x
    lda #0
    sta $fd          ;# of data bytes for command
    inx
    beq databuffer_overrun
    stx $fa          ;BUFFER POINTER
    jmp activity_indicate_and_quit
databuffer_overrun:
    lda #$b0          ;Buffer overrun
    sta $1180,x
    lda #$7b
    sta $1280,x
    inx
    stx $fa
    jmp nullify_shortbuff
fd_is_not_0_or_1:
    sta $fc          ;STORE SECOND BYTE
    dec $fd          ;# of data bytes for command
    jmp activity_indicate_and_quit
system:
    cpx #$f4
    bcc nullify_shortbuff
    cpx #$ff          ;RESET COMMAND...
    bne activity_indicate_and_quit
    inc $f7
    jmp activity_indicate_and_quit
nullify_shortbuff:
    lda #0
    sta $fd

```

```
        sta $fe

activity_indicate_and_quit
        lda #100      ;Delay value...
        sta $f8      ;Indicate activity..
        pla
        tax
bang    dec $f8      ;Indicate activity
        bpl i0
        lda #0
        sta $f8
i0      pla
        rti

midiinit
mod6    lda #3       ;Master Chip Reset Command
mod1    sta $de04    ;Control Register Address
mod7    lda #$16    ;Enable Xmit/Rcv Command - 1:64 (MIDI)
mod2    sta $de04    ;Control Register Address
        lda #0
mod3    sta $de05    ;Transmit Data Register Address
        ldx #$18
mid0    lda $de06    ;Status Register Address
        dex
        bne mid0
        txa
mid1    sta $f7,x    ;Clear the temporary ZP variables..
        inx
        cpx #9
        bne mid1
        ldx #<nmi
        ldy #>nmi
        stx $fffa
        sty $fffb
        stx $0318
        sty $0319
        lda #$81
        ldx #$f0
        ldy #$00
        stx $dd04
        sty $dd05
        sta $dd0d
        bit $dd0d
        sta $dd0e
        bit $dd0e
        rts

;-----
; MAIN MENU
;-----
```

```

start   sei
        lda #$37
        sta 1
        jsr $ff84           ;Init I/O Devices, Ports & Timers
        jsr $ff8a           ;Restore Vectors
        jsr readpres
xstart  jsr setpres
        jsr midiinit
        jsr detectkey
        jsr setkey
st2     sei
        lda #$35
        sta 1
        lda #$1b
        sta $d011
        lda #$15
        sta $d018
        lda #1
        sta $0289
        lda #$80
        sta $028a
        jsr clrscr
        jsr plotscr
        lda #$37
        sta 1
        ldx #0
        stx xplode
        jsr mark
read    jsr rkey
        lda down
        beq j1
        lda #0
        sta down
        ldx xplode
        jsr unmark
        ldx xplode
        inx
        cpx #5
        bne j0
        dex
j0      stx xplode
        jsr mark
j1      lda up
        beq j3
        lda #0
        sta up
        ldx xplode
        jsr unmark
        ldx xplode
        dex
        cpx #$ff

```

```
                bne j2
                inx
j2             stx xplode
                jsr mark
j3             lda home
                beq j4
                ldx xplode
                jsr unmark
                ldx #0
                stx home
                stx xplode
                jsr mark
j4             lda return
                beq j5
                jmp commit
j5             jmp read

;-----
; JUMPING TO/FROM SUBPROGRAMS
;-----

sound         .byte 0
zeropage = $cf00

gomidimode

gm0           ldx #2
             lda $00,x
             sta zeropage,x
             inx
             cpx #$f7
             bne gm0
             lda keybtype
             ldx sound
             jmp $1e00

retfromidi

             stx sound
             pha
             lda #$37
             sta 1
             ldx #2
gm1           lda zeropage,x
             sta $00,x
             inx
             cpx #$f7
             bne gm1
             jsr restax
             jsr setkey
             pla
```

```
                bne gosounded
                jmp st2

gosounded
                ldx sound
                jmp $2d00

retfromsound
                pha
                stx sound
                lda #$37
                sta $01
                jsr setkey
                pla
                cmp #1
                beq gomidimode
                jmp st2

commit
                lda #0
                sta return
                lda xplode
                bne k0
                jmp gomidimode
k0      cmp #1
                bne k2
                jmp gosounded
k2      cmp #4
                bne k3
                jmp edpres
k3      cmp #2
                bne k4
                jsr hexread
                jmp read
k4      cmp #3
                bne k5
                jsr hexwrite
k5      jmp read

;-----
; EDIT PRESETS
;-----

edpres
                jsr i08
                jsr plotpre
ed0     jsr plotyp
                jsr plotad
                jsr setkey
                jsr rkey
                lda return
```

```

                                bne edxt
                                lda plus
                                beq ed1
                                ldx xtyp
                                inx
                                txa
                                and #3
                                sta xtyp
                                jsr setyp
                                jmp ed0
ed1      lda minus
                                beq ed0
                                ldx xtyp
                                dex
                                txa
                                and #3
                                sta xtyp
                                jsr setyp
                                jmp ed0
edxt     jsr setkey
                                jsr writpres
                                jmp xstart

plotad
                                ldx #<ctrl
                                ldy #>ctrl
                                stx $20
                                sty $21
                                ldx #$4f
                                ldy #$05
                                jsr plotword
                                ldx #<xmit
                                ldy #>xmit
                                stx $20
                                sty $21
                                ldx #$77
                                ldy #$05
                                jsr plotword
                                ldx #<stat
                                ldy #>stat
                                stx $20
                                sty $21
                                ldx #$9f
                                ldy #$05
                                jsr plotword
                                ldx #<recv
                                ldy #>recv
                                stx $20
                                sty $21
                                ldx #$c7
```



```

        ldy #$05
        jsr plotword
        rts

plotword
        stx $22
        sty $23
        ldy #0
pw0     lda ($20),y
        and #$0f
        tax
        lda fig,x
        pha
        lda ($20),y
        lsr a
        lsr a
        lsr a
        lsr a
        tax
        lda fig,x
        pha
        iny
        cpy #2
        bne pw0
        ldy #0
pw1     pla
        sta ($22),y
        iny
        cpy #4
        bne pw1
        rts

fig     .byte $30,$31,$32,$33,$34,$35
        .byte $36,$37,$38,$39,$01,$02
        .byte $03,$04,$05,$06

plotyp
        lda xtyp
        cmp #3
        bcc plo0
        lda #3
plo0    asl a
        asl a
        asl a
        tax
        ldy #0
plo1    lda typfc,x
        ora #$80
        sta $0526,y
        lda #1
        sta $d926,y

```

```
        inx
        iny
        cpy #$10
        bne plol
        rts

typfc   .byte $04,$01,$14,$05,$0c,$20
        .byte $2f,$20,$13,$09,$05,$0c
        .byte $2b,$0a,$0d,$13
        .byte $10,$01,$13,$13,$10,$0f
        .byte $12,$14,$20,$20,$20,$20
        .byte $20,$20,$20,$20
        .byte $13,$05,$11,$15,$05,$0e
        .byte $14,$09,$01,$0c,$20,$20
        .byte $20,$20,$20,$20
        .byte $15,$13,$05,$12,$20,$04
        .byte $05,$06,$09,$0e,$05,$04
        .byte $20,$20,$20,$20

setyp

        ldx xtyp
        ldy #$de
        lda tctrl,x
        sta ctrl
        sty ctrl+1
        lda txmit,x
        sta xmit
        sty xmit+1
        lda tstat,x
        sta stat
        sty stat+1
        lda trecv,x
        sta recv
        sty recv+1
        lda trset,x
        sta rset
        lda tenab,x
        sta enab
        rts

tctrl   .byte $04,$08,$00,$00
txmit   .byte $05,$09,$01,$01
tstat   .byte $06,$08,$02,$02
trecv   .byte $07,$09,$03,$03
trset   .byte $03,$03,$03,$03
tenab   .byte $16,$15,$15,$15

        *= $c000
```

;- - - - -

```
; OBJECT Tangentbord(Country)
```

```
;-----
```

```
keybtype .byte 0
```

```
detectkey
```

```
    lda $eba9  
    cmp #$2d  
    beq swe  
    lda #0  
    sta keybtype  
    rts
```

```
swe    lda #1  
        sta keybtype  
        rts
```

```
rkey   cli  
r0     lda $c6  
        beq r0  
        sei  
        lda #0  
        sta $c6  
        lda $0277  
        cmp #17  
        bne r2  
        inc down  
        rts
```

```
r2     cmp #145  
        bne r3  
        inc up  
        rts
```

```
r3     cmp #19  
        bne r4  
        inc home  
        rts
```

```
r4     cmp #13  
        bne r5  
        inc return  
        rts
```

```
r5     cmp #43  
        bne r6  
        inc plus  
        rts
```

```
r6     cmp #45  
        bne r7  
        inc minus
```

```
r7     rts
```

```
xplode .byte 0
```

```
up     .byte 0
```

```
down   .byte 0
```

```
home    .byte 0
return  .byte 0
plus    .byte 0
minus   .byte 0

setkey  ldx #6
        lda #0
sk      sta xplode,x
        dex
        bpl sk
        cli
sk0     lda $91
        cmp #$ff
        bne sk0
        ldx #0
        ldy #$60
sk1     dex
        bne sk1
        dey
        bne sk1
        lda $91
        cmp #$ff
        bne sk0
        lda #$00
        sta $c6
        sei
        rts

;-----

restax  ldx #$1f
re0     lda $fd30,x
        cpx #4
        beq re1
        cpx #5
        beq re1
        sta $0314,x
re1     dex
        bpl re0
        lda #0
        ldx #$1b
re2     sta $d400,x
        dex
        bpl re2
        lda #$7f
        sta $dc0d
        sta $dc00
        lda #8
        sta $dc0e
        sta $dc0f
```

```
        ldx #0
        stx $dc03
        dex
        stx $dc02
        lda #$25
        sta $dc04
        lda #$40
        sta $dc05
        lda #$81
        sta $dc0d
        lda $dc0e
        and #$80
        ora #$11
        sta $dc0e
        lda $dd00
        ora #$10
        sta $dd00
        rts

;-----
; SCREEN ROUTINES
;-----

clrscr
        ldx #0
        stx $d020
        stx $d021
i01      lda #$20
        sta $0400,x
        sta $0500,x
        sta $0600,x
        sta $0700,x
        inx
        bne i01
i08      ldx #0
i07      lda #$0c
        sta $d800,x
        sta $d900,x
        sta $da00,x
        sta $db00,x
        inx
        bne i07
        rts

plotscr
        ldx #<screenx
        ldy #>screenx
        jmp plot

plotpre
        ldx #<pscreen
        ldy #>pscreen

plot
```

```

        stx $20
        sty $21
        lda #13
        ldx #$a7
        ldy #$04
        sta $24
        stx $22
        sty $23
i05     ldy #$18
i02     lda ($20),y
        sta ($22),y
        dey
        bpl i02
        clc
        lda $20
        adc #$19
        sta $20
        bcc i03
        inc $21
i03     clc
        lda $22
        adc #$28
        sta $22
        bcc i04
        inc $23
i04     dec $24
        bpl i05
        rts
mark    jsr intz
        ldy #0
        lda #62
        sta ($22),y
        lda #1
        sta ($24),y
        iny
m0      lda ($20),y
        sta ($22),y
        lda #1
        sta ($24),y
        iny
        cpy #$16
        bne m0
        lda #60
        sta ($22),y
        lda #1
        sta ($24),y
        lda #$37
        sta 1
        rts
```

```

unmark   jsr intz
          ldy #0
m1       lda ($20),y
          sta ($22),y
          lda #$0c
          sta ($24),y
          iny
          cpy #$17
          bne m1
          lda #$37
          sta 1
          rts

intz     lda addl,x
          sta $22
          sta $24
          lda addl2,x
          sta $20
          lda addh,x
          sta $21
          lda #5
          sta $23
          lda #$d9
          sta $25
          lda #$35
          sta 1
          rts

addl     .byte $20,$48,$70,$98,$c0
addl2    .byte <rx1,<rx2,<rx3,<rx4,<rx5
addh     .byte >rx1,>rx2,>rx3,>rx4,>rx5

screenx
          .byte $55,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $49
          .byte $42,$14,$12,$09,$01,$04
          .byte $20,$0d,$09,$04,$09,$13
          .byte $0c,$01,$16,$05,$20,$0d
          .byte $01,$0e,$01,$07,$05,$12
          .byte $42
          .byte $6b,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $43,$43,$43,$43,$43,$43
          .byte $73,$42
rx1      .byte $20,$20,$20,$20,$07
          .byte $0f,$20,$14,$0f,$20,$0d
          .byte $09,$04,$09,$20,$0d,$0f

```

```
      .byte $04,$05,$20,$20,$20,$20
      .byte $42,$42
rx2   .byte $20,$20,$20,$07,$0f
      .byte $20,$14,$0f,$20,$13,$0f
      .byte $15,$0e,$04,$20,$05,$04
      .byte $09,$14,$0f,$12,$20,$20
      .byte $42,$42
rx3   .byte $20,$20,$20,$20,$0c
      .byte $0f,$01,$04,$20,$13,$0f
      .byte $15,$0e,$04,$20,$06,$09
      .byte $0c,$05,$20,$20,$20,$20
      .byte $42,$42
rx4   .byte $20,$20,$20,$20,$13
      .byte $01,$16,$05,$20,$13,$0f
      .byte $15,$0e,$04,$20,$06,$09
      .byte $0c,$05,$20,$20,$20,$20
      .byte $42,$42
rx5   .byte $20,$20,$20,$20,$20
      .byte $20,$20,$20,$10,$12,$05
      .byte $13,$05,$14,$13,$20,$20
      .byte $20,$20,$20,$20,$20,$20
      .byte $42
      .byte $6b,$43,$43,$43,$43,$43
      .byte $43,$43,$43,$43,$43,$43
      .byte $43,$43,$43,$43,$43,$43
      .byte $43,$43,$43,$43,$43,$43
      .byte $73
      .byte $42,$20,$20,$20,$20,$14
      .byte $08,$09,$13,$20,$10,$12
      .byte $0f,$07,$12,$01,$0d,$20
      .byte $09,$13,$20,$20,$20,$20
      .byte $42
      .byte $42,$20,$20,$20,$20,$20
      .byte $20,$20,$20,$06,$12,$05
      .byte $05,$17,$01,$12,$05,$20
      .byte $20,$20,$20,$20,$20,$20
      .byte $42
      .byte $42,$20,$20,$20,$20,$20
      .byte $20,$20,$20,$20,$20,$2d
      .byte $2d,$2d,$20,$20,$20,$20
      .byte $20,$20,$20,$20,$20,$20
      .byte $42
      .byte $42,$20,$20,$20,$20,$20
      .byte $17,$08,$05,$12,$05,$20
      .byte $09,$13,$20,$13,$08,$05
      .byte $3f,$20,$20,$20,$20,$20
      .byte $42
      .byte $4a,$43,$43,$43,$43,$43
      .byte $43,$43,$43,$43,$43,$43
      .byte $43,$43,$43,$43,$43,$43
```



```
.byte $43,$43,$43,$43,$43,$43
.byte $4b
pscreen
.byte $55,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $49
.byte $42,$20,$03,$08,$0f,$0f
.byte $13,$05,$20,$19,$0f,$15
.byte $12,$20,$09,$0e,$14,$05
.byte $12,$06,$01,$03,$05,$20
.byte $42
.byte $6b,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $73
.byte $42,$14,$19,$10,$05,$3a
.byte $20,$04,$01,$14,$05,$0c
.byte $20,$2f,$20,$13,$09,$05
.byte $0c,$2b,$0a,$0d,$13,$20
.byte $42
.byte $42,$03,$14,$12,$0c,$3a
.byte $20,$24,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $42
.byte $42,$18,$0d,$09,$14,$3a
.byte $20,$24,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $42
.byte $42,$13,$14,$01,$14,$3a
.byte $20,$24,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $42
.byte $42,$12,$05,$03,$16,$3a
.byte $20,$24,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $42
.byte $6b,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $73
.byte $42,$2b,$2f,$2d,$20,$14
.byte $0f,$20,$03,$08,$01,$0e
.byte $07,$05,$20,$09,$0e,$14
```

```
.byte $05,$12,$06,$01,$03,$05
.byte $42
.byte $42,$20,$10,$12,$05,$13
.byte $13,$20,$12,$05,$14,$15
.byte $12,$0e,$20,$14,$0f,$20
.byte $05,$18,$09,$14,$21,$20
.byte $42
.byte $42,$20,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$2d
.byte $2d,$2d,$20,$20,$20,$20
.byte $20,$20,$20,$20,$20,$20
.byte $42
.byte $42,$20,$20,$20,$20,$20
.byte $17,$08,$05,$12,$05,$20
.byte $09,$13,$20,$13,$08,$05
.byte $3f,$20,$20,$20,$20,$20
.byte $42
.byte $4a,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $43,$43,$43,$43,$43,$43
.byte $4b

;-----
; PRESET SECTION
;-----

setpres
    ldx ctrl
    ldy ctrl+1
    stx mod1+1
    sty mod1+2
    stx mod2+1
    sty mod2+2
    ldx xmit
    ldy xmit+1
    stx mod3+1
    sty mod3+2
    ldx stat
    ldy stat+1
    stx mod4+1
    sty mod4+2
    stx mid0+1
    sty mid0+2
    ldx recv
    ldy recv+1
    stx mod5+1
    sty mod5+2
    lda rset
    sta mod6+1
    lda enab
```

```

                sta mod7+1
                rts

preset
                .text "type:"
xtyp          .byte 0
;
;0= DATEL + SIEL/JMS
;1= PASSPORT
;2= SEQUENTIAL
;
                .text "6850 ctrl:"
ctrl          .byte $04,$de
                .text "6850 xmit:"
xmit          .byte $05,$de
                .text "6850 stat:"
stat          .byte $06,$de
                .text "6850 recv:"
recv          .byte $07,$de
                .text "6850 rset:"
rset          .byte $03
                .text "6850 enab:"
enab          .byte $16

;-----;
; Load and replacesave for TRIAD      ;
;   MIDISLAVE MANAGER   V1.1          ;
;-----;
pxname        .text "s:"
pname         .text "-program  setup-"
fxname        .text "s:"
fname         .text "-sound  presets-"
;-----;

readpres
                lda #$10
                ldx #<pname
                ldy #>pname
                jsr $ffbd
                jsr ropen
                lda $90
                bvs rp1
                ldy #0
rp0           jsr $ffe4
                sta preset,y
                iny
                cpy #$4c           ;Length
                beq rp1
                lda $90
                bvc rp0
rp1           jsr rclose

```

```
                rts

writpres
                jsr printsave
                jsr nmioff
                jsr prescr
                ldy #0
sp0            lda pxname,y
                jsr $ffa8
                iny
                cpy #$12
                bne sp0
                jsr $ffae
                lda #$10
                ldx #<pname
                ldy #>pname
                jsr $ffbd
                jsr presav
                lda #$00
                jsr $ffd2
                lda #$10
                jsr $ffd2
                ldy #0
sp2            sei
                lda preset,y
                jsr $ffd2
                dec $d020
                inc $d020
                iny
                cpy #$4c            ;Length
                bne sp2
                jsr postsav
                jsr blank
                jsr nmion
                rts

;-----

hexread jsr printload
                jsr nmioff
                ldx #$00            ;Set load addy!
                ldy #$40
                stx $20
                sty $21
                lda #$10
                ldx #<fname
                ldy #>fname
                jsr $ffbd
                jsr ropen
                lda $90
```

```

                                bvs l2
l0      jsr $ffe4
                                dec $d020
                                inc $d020
                                ldy #0
                                sta ($20),y
                                inc $20
                                bne l1
                                inc $21
l1      lda $90
                                bvs l2
                                lda $21
                                cmp #$c0
                                bne l0
l2      jsr rclose
                                jsr blank
                                jsr nmion
                                rts

ropen   lda #1
                                ldx $ba
                                ldy #$60
                                jsr $ffba
                                lda #0
                                sta $9d
                                jsr $ffc0
                                ldx #1
                                jsr $ffc6
                                jsr $ffe4
                                jmp $ffe4

rclose  jsr $ffcc
                                lda #1
                                jmp $ffc3

;-----
hexwrite
                                jsr printsave
                                jsr nmioff
                                jsr prescr
                                ldy #0
s0      lda fxname,y
                                jsr $ffa8
                                iny
                                cpy #$12
                                bne s0
                                jsr $ffae

                                lda #$10
                                ldx #<fname
                                ldy #>fname

```

```
        jsr $ffbd
        jsr presav
        ldx #$00
        ldy #$40
        stx $20
        sty $21
        lda $20
        jsr $ffd2
        lda $21
        jsr $ffd2

l5      ldy #0
        sei
        ldx #$34
        stx $01
        lda ($20),y
        ldx #$37
        stx $01
        jsr $ffd2
        dec $d020
        inc $d020
        inc $20
        bne l6
        inc $21

l6      lda $21
        cmp #$c0
        bne l5
        jsr postsav
        jsr blank
        jsr nmion
        rts

prescr
        lda #0
        sta $9d
        lda $ba
        jsr $ffb1
        lda #$6f
        jmp $ff93

presav
        lda #1
        ldx $ba
        ldy #1
        jsr $ffba
        jsr $ffc0
        ldx #1
        jmp $ffc9

postsav
        jsr $ffcc
        lda #1
```

```
                jmp $ffc3

;-----
printload
                ldx #$16
pl0            lda loadtext,x
                sta $0700,x
                lda loadtext+$17,x
                sta $0728,x
                lda loadtext+$2e,x
                sta $0750,x
                lda loadtext+$45,x
                sta $0778,x
                lda loadtext+$5c,x
                sta $07a0,x
                lda #$01
                sta $db00,x
                sta $db28,x
                sta $db50,x
                sta $db78,x
                sta $dba0,x
                dex
                bpl pl0
                rts

;-----
printsave
                jsr printload
                ldx #3
ps0            lda saveplus,x
                sta $0729,x
                dex
                bpl ps0
                rts

;-----
blank
                ldx #$16
                lda #$20
bt0            sta $0700,x
                sta $0728,x
                sta $0750,x
                sta $0778,x
                sta $07a0,x
                lda #0
                sta $db00,x
                sta $db28,x
                sta $db50,x
                sta $db78,x
                sta $dba0,x
                dex
                bpl bt0
                rts
```

```
;-----  
loadtext  
    .byte $70,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$6e  
    .byte $42,$0c,$0f,$01,$04,$09  
    .byte $0e,$07,$20,$2d,$20,$10  
    .byte $0c,$05,$01,$13,$05,$20  
    .byte $17,$01,$09,$14,$42  
    .byte $42,$20,$04,$0f,$0e,$27  
    .byte $14,$20,$13,$05,$0e,$04  
    .byte $20,$03,$0f,$0d,$0d,$01  
    .byte $0e,$04,$13,$20,$42  
    .byte $42,$04,$15,$12,$09,$0e  
    .byte $07,$20,$04,$09,$13,$0b  
    .byte $0f,$10,$05,$12,$01,$14  
    .byte $09,$0f,$0e,$13,$42  
    .byte $6d,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$43,$43  
    .byte $43,$43,$43,$43,$7d  
;-----  
saveplus  
    .byte $20,$13,$01,$16  
;-----  
nmioff  
    lda #$7f  
    sta $dd0d  
    bit $dd0d  
    ldx $0318  
    ldy $0319  
    stx x18+1  
    sty x19+1  
    ldx #$47  
    ldy #$fe  
    stx $0318  
    sty $0319  
    rts  
  
nmion  
x18    ldx #0  
x19    ldy #0  
    stx $0318  
    sty $0319  
    lda #$81  
    ldx #$f0  
    ldy #$00  
    stx $dd04  
    sty $dd05  
    sta $dd0d
```



```

        bit $dd0d
        sta $dd0e
        bit $dd0e
        rts

```

```

;-----

```

"MIDIPROG82.S"

```

;-----;
;      IRQ initialization with NMI      ;
; #      ----- # ;
; # # * Prevents restore damage # # ;
; ##### * With spacer. initialize #####;
;-----;
                *= $1e00
;-----
; NextZpage = $3d
;-----
                sta scan+1
                stx instr
                cli
                lda #$7f
                sta $dc0d
meffo      sei
                lda #$35
                sta 1
                ldx #$ff
                txs
                stx $dc02
                ldx #<irq
                ldy #>irq
                stx $fffe
                sty $ffff
                lda #$7f
                sta $dc0d
                lda #$00
                sta $d01a
                lda #$1b
                sta $d011
                lda #$00
                sta $dc03
                sta $d020
                sta $d021
                jsr reset
                jsr showmain
                jsr redat
                lda #$81
                sta $dc0d
                bit $dc0d

```

```
        sta $dc0e
        bit $dc0e
        sta $f9
        cli

hit     jsr midind
        jsr update
        lda $fa
        beq hi1
hi2     lda #0
        sta $f9
hi0     ldy #0
        jsr midiplay
        inc hi0+1
        inc $f9
        ldy hi0+1
        cpy $fa
        bne hi2
        lda #0
        sta hi0+1
        sta $fa
hi1     lda $f7
        beq hi3
        lda #0
        sta $f7
        sta $f9
        lda #$7f
        sta $dc0d
        jmp meffo
hi3     lda #$7f
        sta $dc00
        lda $dc01
        cmp #$7f           ;RUN/STOP
        beq sluss
        cmp #$fb
        beq sluz
        jmp hit

sluz    lda #1
        bne sl0
sluss   lda #0
sl0     sta retval+1
        sei
        lda #0
        sta $f9
        sta $fa
        lda #$7f
        sta $dc0d
        bit $dc0d
        lda #$37
```

```

retval    lda #0
          sta 1
          ldx instr
          jmp $0810 ;Block here

;-----
irq       pha
          lda $dc0d
          and #1
          beq bang2
          txa
          pha
          tya
          pha
          jsr macro
          pla
          tay
          pla
          tax

bang2    pla
          rti

macx     .byte 0
chan1    .byte 0,0,0,0,0,0,0,0
chan2    .byte 0,0,0,0,0,0,0,0
chan3    .byte 0,0,0,0,0,0,0,0
temp     .byte 0

macro    ldx #0
ma5      lda chan1,x
          bne ma8
          jmp ma4
ma8      lda chan1+2,x
          sta $28
          lda chan1+3,x
          sta $29
          ldy #0
          lda ($28),y
          cmp #$ff
          bne ma0
          lda #$28
          sta chan1+2,x
          sta $28
          lda ($28),y
          cmp #$ff
          beq ma1
ma0      cmp #$fe
          bne ma2
ma1      lda #0
          sta chan1,x
          beq ma4

```

```
ma2      ldy chan1+1,x
          lda d4mirror+4,y
          and #1
          sta temp
          ldy #0
          lda ($28),y
          pha
          lda temp
          bne fux0
          pla
          and #$fe
          jmp fux1

fux0     pla
fux1     ldy chan1+1,x
          sta $d404,y
          sta d4mirror+4,y
          ldy #1
          lda ($28),y
          bpl ex0
          and #$7f
          sta ex1+1
          lda chan1+4,x
          sec

ex1      sbc #0
          cmp #$60
          bcc ma7
          lda #$00
          beq ma7

ex0      clc
          adc chan1+4,x
          cmp #$60
          bcc ma7
          lda #$5f

ma7      tay
          lda notehi,y
          pha
          lda notelow,y
          ldy chan1+1,x
          sta $d400,y
          sta d4mirror,y
          pla
          sta $d401,y
          sta d4mirror+1,y
          ldy #2
          lda ($28),y
          beq ma3
          sta $d416
          sta d4mirror+$16

ma3      lda $28
          clc
```

```

                adc #3
                sta chan1+2,x
ma4            lda chan1+5,x
                beq ma9
                bpl mab
maa            and #$1f
                tay
                lda wheels,y
                jmp mad0
mab            pha
                lda chan1+7,x ;Vibrato delay
                beq mao
                dec chan1+7,x
                pla
                jmp ma9
mao            pla
mad0           pha
;-----
                lsr a      ;Begin of vibratoh!
                lsr a      ;a=depth 00..7F
                lsr a
                sta $3a
                pla
                asl a
                asl a
                asl a
                asl a
                sta $39
                clc
                lda $3a
                adc #$e0
                sta $3a
mae            ldy mam
                lda ($39),y
                bmi mac
                ldy chan1+1,x
                clc
                adc d4mirror,y
                sta $d400,y
                bcc ma9
                lda d4mirror+1,y
                adc #0
                sta $d401,y
                jmp ma9
mac            ldy chan1+1,x
                and #$7f
                sta mad+1
                lda d4mirror,y
                sec
mad            sbc #0

```

```

        sta $d400,y
        bcs ma9
        lda d4mirror+1,y
        sbc #0
        sta $d401,y
ma9     lda chan1+6,x
        bne mak
        jmp d
mak     bmi pulsqd
        jmp mag
pulsqd  and #$1f
        tay
        lda wheels,y
        pha
        ldy chan1+1,x
        lsr a
        lsr a
        lsr a
        lsr a
        ora #8
        sta $d403,y
        pla
        asl a
        asl a
        asl a
        asl a
        ora #$0f
        sta $d402,y
        jmp d
mag     pha
        lsr a
        lsr a
        lsr a
        sta $3c
        pla
        asl a
        asl a
        asl a
        asl a
        asl a
        sta $3b
        clc
        lda $3c
        adc #$e0
        sta $3c
mah     ldy mam+1
        lda ($3b),y
        bmi mai
        ldy chan1+1,x
        clc
```

```

        adc d4mirror+2,y
        sta $d402,y
        bcc d
        lda d4mirror+3,y
        adc #0
        sta $d403,y
        jmp d
mai     ldy chan1+1,x
        and #$7f
        sta maj+1
        lda d4mirror+2,y
        sec
maj     sbc #0
        sta $d402,y
        bcs d
        lda d4mirror+3,y
        sbc #0
        sta $d403,y
d       txa
        clc
        adc #8
        tax
        cpx #$18
        beq ma6
        jmp ma5
ma6     dec maf
        bne snurr
        ldx maf+1
        stx maf
        ldx mam
        inx
        txa
        and #$1f
        sta mam
snurr   dec vibspd
        bne man
        ldx vibspd+1
        stx vibspd
        ldx mam+1
        inx
        txa
        and #$1f
        sta mam+1
man     rts
admacro txa
        pha
        lda macx
        asl a
        asl a
        asl a

```

```
tax          ;Blockera macro!  
lda #1  
sei  
sta chan1,x ;+0 Macuppd.  
inx  
lda channel  
sta chan1,x ;+1 k 0 7 E  
inx  
lda #$28  
sta chan1,x ;+2 Startmacropos  
inx  
lda $27  
sta chan1,x ;+3 Instrument  
inx  
pla  
pha  
sta chan1,x ;+4 Ton  
inx  
ldy #0  
lda #$20  
sta $26  
lda ($26),y  
sta chan1,x ;+5 Vibrato  
inx  
lda #$1e  
sta $26  
lda ($26),y  
sta chan1,x ;+6 Pvib  
inx  
lda #$21  
sta $26  
lda ($26),y  
lsr a  
lsr a  
lsr a  
lsr a  
sta chan1,x ;+7 Vibdelay  
cli  
pla  
tax  
rts  
  
;-----  
chanloc .byte $80  
        .byte $80  
        .byte $80  
old     .byte 3  
  
addtone  
        lda old  
        cmp #1
```



```

        bne c6
        jsr loc2
        bcs c8
        lda #2
        sta old
        lda #$51
        sta $0558
        rts
c6      cmp #2
        bne c7
        jsr loc3
        bcs c8
        lda #3
        sta old
        lda #$51
        sta $0580
        rts
c7      cmp #3
        bne c8
        jsr loc1
        bcs c8
        lda #1
        sta old
        lda #$51
        sta $0530
        rts
c8      jsr loc1
        bcs c9
        lda #1
        sta old
        lda #$51
        sta $0530
        rts
c9      jsr loc2
        bcs ca
        lda #2
        sta old
        lda #$51
        sta $0558
        rts
ca      jsr loc3
        bcs cb
        lda #3
        sta old
        lda #$51
        sta $0580
cb      lda force
        bne cd
        ldx old
        cpx #3
        bne cc
```

```

                                ldx #0
cc      lda #$80
                                sta chanloc,x
                                jmp addtone
cd      rts

loc1    lda chanloc
                                bpl c0
                                lda note
                                sta chanloc
                                lda #0
                                sta channel
                                sta macx
                                jsr tonpa
                                clc
                                rts
c0      sec
                                rts

loc2    lda chanloc+1
                                bpl c1
                                lda note
                                sta chanloc+1
                                lda #7
                                sta channel
                                lda #1
                                sta macx
                                jsr tonpa
                                clc
                                rts
c1      sec
                                rts

loc3    lda chanloc+2
                                bpl c2
                                lda note
                                sta chanloc+2
                                lda #$0e
                                sta channel
                                lda #2
                                sta macx
                                jsr tonpa
                                clc
                                rts
c2      sec
                                rts

deltone
                                ldx #$80
```

```
        cmp chanloc
        bne c3
        stx chanloc
        lda #$20
        sta $0530
        lda #0
        sta channel
        jsr tonav
        clc
        rts
c3      cmp chanloc+1
        bne c4
        stx chanloc+1
        lda #$20
        sta $0558
        lda #$07
        sta channel
        jsr tonav
        clc
        rts
c4      cmp chanloc+2
        bne c5
        stx chanloc+2
        lda #$20
        sta $0580
        lda #$0e
        sta channel
        jsr tonav
        clc
        rts
c5      sec
        rts
alloff  lda #$80
        sta chanloc
        sta chanloc+1
        sta chanloc+2
        lda #0
        sta channel
        jsr tonav
        lda #7
        sta channel
        jsr tonav
        lda #$0e
        sta channel
        jsr tonav
        lda #$20
        sta $0530
        sta $0558
        sta $0580
        rts
```

```
-----  
;-----  
midiplay  
    lda comb1,y      ;What command?  
    cmp #$90         ;Note on?  
    bne h0  
    lda comb2,y  
    sta note  
    lda comb3,y  
    bne h5  
    lda note  
    jmp deltone  
h5    jmp addtone  
  
h0    cmp #$80         ;Off  
    bne h1  
    lda comb2,y  
    jmp deltone  
h1    cmp #$a0         ;Poly  
    bne h2  
    lda comb2,y  
    sta note  
    jmp addtone  
h2    cmp #$c0         ;Program  
    bne h3  
    lda comb3,y  
    sta instr  
    jsr alloff  
    jmp writeinst  
j6    rts  
  
-----  
;-----  
h3    cmp #$e0  
        beq j5  
        jmp h8  
j5    lda pitchen  
        beq j6  
        ldx old  
        dex  
        lda chandum,x  
        sta chand  
        lda chanloc,x  
        bmi j6  
        lda pitchen  
        cmp #1  
        bne j0  
        lda current  
        cmp #$5f  
        bcs j6  
        tax  
        jsr ini1  
        inx
```

```

                                inx
                                txa
                                jsr ini2
                                jmp j4
j0      cmp #2
                                bne j1
                                lda current
                                cmp #$03
                                bcc j6
                                cmp #$5e
                                bcs j6
                                pha
                                sec
                                sbc #1
                                jsr ini1
                                pla
                                clc
                                adc #3
                                jsr ini2
                                jmp j4
j1      cmp #3
                                bne j2
                                lda current
                                cmp #6
                                bcc h4
                                cmp #$5a
                                bcs h4
                                pha
                                sec
                                sbc #5
                                jsr ini1
                                pla
                                clc
                                adc #7
                                jsr ini2
                                jmp j4
j2      lda current
                                cmp #$0c
                                bcc h4
                                cmp #$54
                                bcs h4
                                pha
                                sec
                                sbc #$0b
                                jsr ini1
                                pla
                                clc
                                adc #$0d
                                jsr ini2
j4      lda comb2,y
                                asl a
```

```
        asl a
        lda comb3,y
        rol a
        sta mulx
        cmp #$80
        beq h7
        eor #$ff
        sta muly
        jsr mulifax
        ldy chand
        lda resulh
        sta $d400,y
        sta d4mirror,y
        lda resulx
        sta $d401,y
        sta d4mirror+1,y
h4      rts
h7      ldx current
        ldy chand
        lda notehi,x
        sta $d401,y
        sta d4mirror+1,y
        lda notelow,x
        sta $d400,y
        sta d4mirror,y
        jmp h4
h8      cmp #$b0
        bne hb
        lda comb2,y
        cmp #$20
        bcs h9
        tax
        lda comb3,y
        sta wheels,x
        rts
h9      cmp #$7b
        bne ha
        jmp alloff
ha      rts
hb      rts
;-----
pitchen .byte 0
chandum .byte 0,7,14
chand   .byte 0
not1h   .byte 0
not1l   .byte 0
not2h   .byte 0
not2l   .byte 0
mulx    .byte 0
muly    .byte 0
```

```
resultl .byte 0
resulth .byte 0
resultx .byte 0

z0      = $2a

ini1    tax
        lda notehi-1,x
        sta not2h
        lda notelow-1,x
        sta not2l
        rts

ini2    tax
        lda notehi-1,x
        sta not1h
        lda notelow-1,x
        sta not1l
        rts

mulifax lda not1l
        clc
        adc #2
        sta z0
        lda not1h
        adc #0
        sta z0+1
        lda mulx
        sta z0+3
        jsr mulu
        lda z0+4
        sta resultl
        lda z0+5
        sta resulth
        lda z0+6
        sta resultx

        lda not2l
        clc
        adc #2
        sta z0
        lda not2h
        adc #0
        sta z0+1
        lda muly
        sta z0+3
        jsr mulu
        clc
        lda z0+4
        adc resultl
        sta resultl
```

```
                lda z0+5
                adc resulh
                sta resulh
                lda z0+6
                adc resulx
                sta resulx
                lda resull
                bpl mul5
                inc resulh
                bne mul5
                inc resulx
mul5            inc resulh
                bne mul6
                inc resulx
mul6            rts
mulu
                lda #0
                sta z0+2
                sta z0+4
                sta z0+5
                sta z0+6
                ldx #7
mul0            lsr z0+3
                bcc mul1
                clc
                lda z0
                adc z0+4
                sta z0+4
                lda z0+1
                adc z0+5
                sta z0+5
                lda z0+2
                adc z0+6
                sta z0+6
mul1            asl z0
                rol z0+1
                rol z0+2
                dex
                bpl mul0
                rts
;-----
;-----
indicat        .byte 1
numdata        .byte 0
comb1          = $1180 ;MIDI buffer
comb2          = $1280
comb3          = $1380
force          .byte 0
wheels         .byte 0,0,0,0,0,0,0,0,0,0,0,0,0
```



```

                .byte 0,0,0,0,0,0,0,0,0,0,0,0,0
                .byte 0,0,0,0,0,0,0,0
;-----
channel  .byte 0          ;0=1,7=2,E=3
note     .byte 0
addon    .byte 0
instr    .byte 0
pulsewh  .byte $08
pulsewl  .byte $88
type     .byte $41
ad       .byte $0f
sr       .byte $c7
maf      .byte 1,1
vibspd   .byte 1,1
mam      .byte 0,0
;-----
d4mirror .byte 0        ;Freqlow channel 1
                .byte 0        ;Freqhi  channel 1
                .byte 0        ;Pulsewl  channel 1
                .byte 0        ;Pulsewh  channel 1
                .byte 0        ;Type     channel 1
                .byte 0        ;A/D     channel 1
                .byte 0        ;S/R     channel 1
                .byte 0        ;Freqlow channel 2
                .byte 0        ;Freqhi  channel 2
                .byte 0        ;Pulsewl  channel 2
                .byte 0        ;Pulsewh  channel 2
                .byte 0        ;Type     channel 2
                .byte 0        ;A/D     channel 2
                .byte 0        ;S/R     channel 2
                .byte 0        ;Freqlow channel 3
                .byte 0        ;Freqhi  channel 3
                .byte 0        ;Pulsewl  channel 3
                .byte 0        ;Pulsewh  channel 3
                .byte 0        ;Type     channel 3
                .byte 0        ;A/D     channel 3
                .byte 0        ;S/R     channel 3
                .byte 0        ;Filter low
                .byte 0        ;Filter high
                .byte 0        ;Filter control
                .byte 0        ;Volume control
                .byte 0        ;N/A
                .byte 0        ;N/A
                .byte 0        ;Oscillator 3
                .byte 0        ;Envelope  3
;-----
plusmode          ;Transposing modes
                lda #$18
                sta dum0
                lda #$6d
                sta dum1

```

```
        lda #$5f
        sta dum2+1
        rts

minusmode
        lda #$38
        sta dum0
        lda #$ed
        sta dum1
        lda #$00
        sta dum2+1
        rts

;-----
tonpa   lda instr
        clc
        adc #$40           ;Instr.base
        sta $27
        lda #$1c
        sta $26
        ldy #0
        lda ($26),y
        bpl k1
        lda note

dum0    clc
dum1    adc addon
        cmp #$60
        bcc k1

dum2    lda #$5f
k1      ldy #1

        pha
        lda ($26),y
        sta pitchen      ;Pitch enable
        ldy #3
        lda ($26),y
        and #$0f
        sta vibspd
        sta vibspd+1
        ldy #5
        lda ($26),y
        and #$0f
        sta maf
        sta maf+1
        pla
        ldy channel
        tax
        stx current
        lda notelow,x
        sta $d400,y
        sta d4mirror,y
        lda notehi,x
        sta $d401,y
```

```

                sta d4mirror+1,y
                lda #$11
                sta $26
                ldy #4
k0             lda ($26),y
                sta pulsewh,y
                dey
                bpl k0
                ldy channel
                lda pulsewl
                sta $d402,y
                sta d4mirror+2,y
                lda pulsewh
                sta $d403,y
                sta d4mirror+3,y
                lda type
                ora #1             ;Sure on!
                sta $d404,y
                sta d4mirror+4,y
                lda ad
                sta $d405,y
                sta d4mirror+5,y
                lda sr
                sta $d406,y
                sta d4mirror+6,y
                jsr admacro
                jsr plotcurr
noway         rts
tonav        ldy channel
                lda d4mirror+4,y
                and #$fe
                sta $d404,y
                sta d4mirror+4,y
                jsr delcurr
                rts
;-----
reset        ldx #$1b
                lda #0
r0           sta $d400,x
                sta d4mirror,x
                dex
                bpl r0
                lda #0
                tax
r1           sta $1180,x
                sta $1280,x
                sta $1380,x
                inx
                bne r1
                txa

```

```

        sta channel
        sta numdata
        sta comb1
        sta comb2
        sta comb3
        sta $fa
        sta note
        sta $37          ;Numdata
        sta $38
        lda #1
        sta maf
        sta maf+1
        sta vibspd
        sta vibspd+1
        lda #3
        sta old
ex3      clc
        bcs ex4
        lda #$38
        sta ex3
        lda #0
        sta addon
        sta instr
        sta force
ex4      rts
redat    lda addon
        bpl ex5
        lda #$2d
        bne ex6
ex5      lda #$2b
ex6      sta $04db
        jsr u5
        lda force
        bne ex7
        jmp plotyes
ex7      jmp plotno
;-----
showmain ldx #0
sh0      lda $0c00,x
        sta $0400,x
        lda $0d00,x
        sta $0500,x
        lda $0e00,x
        sta $0600,x
        lda $0ee8,x
        sta $06e8,x
        lda #$0c
        sta $d800,x
        sta $d900,x
```

```

        sta $da00,x
        sta $dae8,x
        inx
        bne sh0
        lda #3
        sta $dd00
        lda #15
        sta $d018
        ldx channel
        inx
        txa
        ldx #$b4
        ldy #$04
        jsr plothex
        lda addon
        ldx #$dc
        ldy #$04
        jsr plothex
        jsr writeinst
        jsr greyall
        lda #0
        sta choice
        ldx #$b3
        ldy #$d8
        jsr white
        jsr wr9
        rts
greyall ldx #$b3
        ldy #$d8
        jsr grey
        ldx #$db
        ldy #$d8
        jsr grey
        ldx #$03
        ldy #$d9
        jsr grey
        ldx #$7b
        ldy #$d9
        jsr grey
        rts
;-----
grey    stx $20
        sty $21
        ldy #0
        lda #$0c
        sta ($20),y
        iny
        sta ($20),y
        iny
        sta ($20),y
        rts

```

```
-----  
white      stx $20  
           sty $21  
           ldy #0  
           lda #1  
           sta ($20),y  
           iny  
           sta ($20),y  
           iny  
           sta ($20),y  
           rts  
-----  
plothex    stx $20  
           sty $21  
           ldy #0  
           pha  
           lsr a  
           lsr a  
           lsr a  
           lsr a  
           tax  
           lda pt0,x  
           sta ($20),y  
           iny  
           pla  
           and #$0f  
           tax  
           lda pt0,x  
           sta ($20),y  
           rts  
pt0        .byte $30,$31,$32,$33,$34  
           .byte $35,$36,$37,$38,$39  
           .byte 1,2,3,4,5,6  
-----  
writeinst  
           ldx instr  
           inx  
           txa  
           ldx #$04  
           ldy #$05  
           jsr plothex  
           ldx instr  
           inx  
           txa  
           cmp #100  
           bcc wr3  
           lda #$31  
           sta $052b  
           txa  
           sec
```

```

                sbc #100
                tax
                jmp wr5
wr3      lda #20
                sta $052b
wr5      lda #30
                sta $052c
                txa
                cmp #10
                bcs wr6
                lda #20
                sta $052c
                jmp wr7
wr6      sec
                inc $052c
                sbc #10
                cmp #10
                bcs wr6
                tax
wr7      lda #30
                sta $052d
                txa
                beq wr9
wr8      inc $052d
                dex
                bne wr8
wr9      lda instr
                clc
                adc #40      ;Instr.base
                sta $25
                lda #0
                sta $24
                ldy #0f
wr0      lda ($24),y      ;Name
                sta $05be,y
                dey
                bpl wr0
                ldy #1d
                lda ($24),y
                asl a
                asl a
                asl a
                tax
                ldy #0
wr4      lda modes,x
                sta $05ee,y
                inx
                iny
                cpy #8
                bne wr4
                ldy #1e

```

```
                ldx #7
                lda ($24),y
                bne wrb
wra             lda modes,x
                sta $063e,x
                dex
                bpl wra
                jmp wre
wrb             bmi wrd
wrc             lda auto,x
                sta $063e,x
                dex
                bpl wrc
                lda ($24),y
                ldx #$44
                ldy #$06
                jsr plothex
                jmp wre
wrd             lda wheel,x
                sta $063e,x
                dex
                bpl wrd
                lda ($24),y
                sec
                sbc #$7f
                ldx #$44
                ldy #$06
                jsr plothex
wre             ldy #$20
                ldx #7
                lda ($24),y
                bne wrg
wrf             lda modes,x
                sta $0616,x
                dex
                bpl wrf
                jmp wrj
wrg             bmi wri
wrh             lda auto,x
                sta $0616,x
                dex
                bpl wrh
                lda ($24),y
                ldx #$1c
                ldy #$06
                jsr plothex
                jmp wrj
wri             lda wheel,x
                sta $0616,x
                dex
```



```

        bpl wri
        lda ($24),y
        sec
        sbc #$7f
        ldx #$1c
        ldy #$06
        jsr plothex
wrj      ldy #$16
        lda ($24),y    ;Macro speed
        sta $dc04
        iny
        lda ($24),y
        sta $dc05
        iny
        ldx #0
wr2      lda ($24),y    ;Filters
        sta $d415,x
        sta d4mirror+$15,x
        iny
        inx
        cpx #4
        bne wr2
        rts

;-----
modes    .byte $0e,$2f,$01,$20,$20,$20
         .byte $20,$20
         .byte $31,$2f,$32,$20,$0e,$0f
         .byte $14,$05
         .byte $0e,$0f,$14,$05,$20,$20
         .byte $20,$20
         .byte $31,$2f,$32,$20,$0f,$03
         .byte $14,$2e
         .byte $0f,$03,$14,$01,$16,$05
         .byte $20,$20
auto     .byte $01,$15,$14,$0f,$20,$20
         .byte $20,$20
wheel    .byte $17,$08,$05,$05,$0c,$20
         .byte $20,$20

;-----
choice   .byte 0
int      .byte 1
keyrep   .byte 10,1
old00    .byte 0
old01    .byte 0
up       .byte 0
down     .byte 0
plus     .byte 0
minus    .byte 0

midind   lda $f8
         beq mi0

```

```
        lda #$51
        sta $04b8
        rts
mii0    lda #$20
        sta $04b8
        rts
update  jsr chkey
        lda choice
        bne v0
        lda plus
        beq u2
        ldx $ff
        inx
        cpx #$10
        bne u1
        ldx #0
u1      stx $ff
        jmp u5
u2      lda minus
        bne u3
        jmp dumex
u3      ldx $ff
        dex
        bpl u4
        ldx #$0f
u4      stx $ff
u5      ldx $ff
        inx
        txa
        ldx #$b4
        ldy #$04
        jsr plothex
        rts
v0      cmp #1
        bne w0
        lda plus
        beq v2
        lda $04db
        cmp #$2b
        bne v7
        ldx addon
        inx
        cpx #$20
        beq v1
v8      stx addon
v1      jmp v5
v7      ldx addon
        dex
        bne v8
```

```
        lda #$2b
        sta $04db
        jsr plusmode
        jmp v8
v2      lda minus
        bne v3
        jmp dumex
v3      lda $04db
        cmp #$2b
        bne v9
        ldx addon
        dex
        bpl va
        ldx #1
        lda #$2d
        sta $04db
        jsr minusmode
va      stx addon
        jmp v5
v9      ldx addon
        inx
        cpx #$20
        beq v5
        stx addon
v5      lda addon
        ldx #$dc
        ldy #$04
        jsr plothex
        rts
w0      cmp #2
        bne x0
        lda plus
        beq w2
        ldx instr
        inx
        cpx #$80
        bne w1
        ldx #0
w1      stx instr
        jmp w5
w2      lda minus
        bne w3
        jmp dumex
w3      ldx instr
        dex
        bpl w4
        ldx #$7f
w4      stx instr
w5      jsr alloff
        jsr writeinst
        rts
```

```
x0      cmp #3
        bne y0
        lda minus
        beq x1
        lda #1
        sta force
        jsr plotno
        rts
x1      lda plus
        bne x2
        jmp dumex
x2      lda #0
        sta force
        jsr plotyes
        rts
y0      jmp dumex
dumex  lda down
        beq uc
        ldx choice
        inx
        cpx #4           ;Last
        bne u7
        ldx #0
u7      jmp u8
uc      lda up
        beq u6
        ldx choice
        dex
        bpl u8
        ldx #3           ;Last
u8      stx choice
u9      jsr greyall
        lda choice
        bne ua
        ldx #$b3         ;Color 1
        ldy #$d8
        jsr white
        jmp u6
ua      cmp #1
        bne ub
        ldx #$db         ;Color 2
        ldy #$d8
        jsr white
        jmp u6
ub      cmp #2
        bne ud
        ldx #$03
        ldy #$d9
        jsr white
```

```

ud      cmp #3
        bne u6
        ldx #$7b
        ldy #$d9
        jsr white
u6      rts
plotyes lda #25
        sta $057b
        lda #5
        sta $057c
        lda #19
        sta $057d
        rts
plotno  lda #32
        sta $057b
        lda #14
        sta $057c
        lda #15
        sta $057d
        rts
;-----
chkey   lda #0
        sta up
        sta down
        sta plus
        sta minus
        lda old00
        sta $dc00
        lda old01
        cmp $dc01
        beq u11
u12     dec int
        beq scan
        rts
u11     dec keyrep
        bne u0
        dec keyrep+1
        bne u0
        jmp scan
u0      rts
scan    lda #0
        bne sca7
        lda #%11011111
        sta $dc00
        lda $dc01
        cmp #%11111110
        bne sca0
        inc plus
        jmp sca4

```

```
sca0    cmp  #%11110111
        bne sca1
        inc minus
        jmp sca4
sca1    lda  #%11111110
        sta $dc00
        lda $dc01
        cmp  #%01111111
        bne sca6
        lda  #%10111111
        sta $dc00
        lda $dc01
        cmp  #%11101111
        beq sca3
        lda  #%11111101
        sta $dc00
        lda $dc01
        cmp  #%01111111
        beq sca3
        inc down
        jmp sca4
sca3    inc up
        jmp sca4
sca6    lda  #0
        sta $dc00
        jmp sca4
sca7    lda  #%10111111
        sta $dc00
        lda $dc01
        and  #%00010000
        beq scaa
        lda  #%11111101
        sta $dc00
        lda $dc01
        and  #%10000000
        bne sca8
scaa    lda  #%10111111
        sta $dc00
        lda $dc01
        and  #%00100000
        bne sca1
        inc plus
        jmp sca4
sca8    lda  #%11011111
        sta $dc00
        lda $dc01
        cmp  #%11111110
        bne sca9
        inc minus
        jmp sca4
```

```

sca9      jmp sca1
sca4      lda $dc00
          sta old00
          lda $dc01
          sta old01
          lda #0
          sta keyrep
          lda #3
          sta keyrep+1
          lda #5
          sta int
          rts

;-----
current  .byte 0
plotcurr
          ldx #$00
          ldy #$10
          stx $22
          sty $23
          lda current
          asl a
          bcc p2
          inc $23
p2        asl a
          bcc p3
          inc $23
          clc
p3        adc $22
          sta $22
          lda $23
          adc #0
          sta $23
          ldy #2
p0        lda ($22),y
          sta $0553,y
          dey
          bpl p0
          rts
delcurr
          ldy #2
          lda #$2d
p1        sta $0553,y
          dey
          bpl p1
          rts

;-----
          .byte $01
notehi
          .byte $01,$01,$01,$01,$01
          .byte $01,$01,$01,$01,$01,$01
          .byte $02,$02,$02,$02,$02,$02

```

```

      .byte $02,$03,$03,$03,$03,$03
      .byte $04,$04,$04,$04,$05,$05
      .byte $05,$06,$06,$06,$07,$07
      .byte $08,$08,$09,$09,$0a,$0a
      .byte $0b,$0c,$0d,$0d,$0e,$0f
      .byte $10,$11,$12,$13,$14,$15
      .byte $17,$18,$1a,$1b,$1d,$1f
      .byte $20,$22,$24,$27,$29,$2b
      .byte $2e,$31,$34,$37,$3a,$3e
      .byte $41,$45,$49,$4e,$52,$57
      .byte $5c,$62,$68,$6e,$75,$7c
      .byte $83,$8b,$93,$9c,$a5,$af
      .byte $b9,$c4,$d0,$dd,$ea,$f8
      .byte $f8,$f8,$f8,$f8,$f8,$f8
      .byte $f8,$f8,$f8,$f8,$f8,$f8
;-----
notelow      .byte $06

      .byte $16,$27,$38,$4b,$5e
      .byte $73,$89,$a1,$ba,$d4,$f0
      .byte $0d,$2c,$4e,$71,$96,$bd
      .byte $e7,$13,$42,$74,$a8,$e0
      .byte $1b,$59,$9c,$e2,$2c,$7b
      .byte $ce,$27,$84,$e8,$51,$c0
      .byte $36,$b3,$38,$c4,$59,$f6
      .byte $9d,$4e,$09,$d0,$a2,$81
      .byte $6d,$67,$70,$88,$b2,$ed
      .byte $3a,$9c,$13,$a0,$44,$02
      .byte $da,$ce,$e0,$11,$64,$da
      .byte $75,$38,$26,$40,$89,$04
      .byte $b4,$9c,$c0,$22,$c8,$b4
      .byte $eb,$71,$4c,$80,$12,$08
      .byte $68,$38,$80,$45,$90,$68
      .byte $d6,$e3,$98,$00,$24,$10
      .byte $10,$10,$10,$10,$10,$10
      .byte $10,$10,$10,$10,$10,$10
;-----

```

"SOUNDED36.S"

```

;-----;
; 0I0IAAAE0I0 AEI0 O0EAA IEAE0IAO0 01.0 ;
;-----;
      *= $2D00
;-----
      SEI
      LDA #$36
      STA 1
      LDX #0      ;;;;;;;;;;??

```



```

      STX CURSND
      LDX #9
      LDA #0
ST0   STA PARAM,X
      DEX
      BPL ST0
      JSR INIT
      JSR REFRESH
      JSR READ
      LDX CURSND

      JMP $0800      ;;;;;;;;;;;

      JMP $0813
;-----
PARAM .BYTE 0
UP    .BYTE 0
DOWN  .BYTE 0
RUNSTOP .BYTE 0
CTRL  .BYTE 0
HOME  .BYTE 0
CLRHOME .BYTE 0
XCOR  .BYTE 0
YCOR  .BYTE 0
XPOS  .BYTE 0
;-----
READ  LDA RUNSTOP
      BEQ XU
      LDA #0
      RTS
XU    LDA CTRL
      BEQ X6
      LDA #1
      RTS
X6    LDA UP
      BEQ X7
      LDA #0
      STA UP
      LDX PARAM
      BEQ X7
      DEX
      STX PARAM
      JSR UNVERT
      JSR FIXCORD
X7    LDA DOWN
      BEQ X8
      LDA #0
      STA DOWN
      LDX PARAM
      INX
      CPX #15

```

```
X8      BEQ X8
        STX PARAM
        JSR UNVERT
        JSR FIXCORD
        LDA HOME
        BEQ XY
        LDA #0
        STA HOME
        LDA #0
        STA PARAM
        JSR UNVERT
        JSR FIXCORD
XY      JSR REFRESH
        LDA CLRHOME
        BEQ V2
        LDA #0
        STA CLRHOME
        LDY #$0F
        LDA #$20
V0      STA ($20),Y
        DEY
        BPL V0
        LDY #$10
        LDA #0
        STA ($20),Y
        INY
        LDA #$88
        STA ($20),Y
        INY
        LDA #8
        STA ($20),Y
        INY
        LDA #1
        STA ($20),Y
        INY
        LDA #0
        STA ($20),Y
        INY
        STA ($20),Y
        INY
        LDA #$C7
        STA ($20),Y
        INY
        LDA #$4C
        STA ($20),Y
        INY
        LDA #0
V1      STA ($20),Y
        INY
        BNE V1
```

```

LDA #1
LDY #$1F
STA ($20),Y
LDY #$21
STA ($20),Y
LDY #$28
LDA #$FE
STA ($20),Y
LDY #$1C
LDA #$80
STA ($20),Y
LDY #$1B
LDA #$1F
STA ($20),Y
LDY #$FD
LDA #$FE
STA ($20),Y
JSR REFRESH
INC HOME
JMP READ
V2 LDA PARAM
BEQ X4
JMP Y0
X4 LDX #$3A
LDY #$04
STX XCOR
STY YCOR
JSR INVERT
JSR READKEY
CMP #43
BNE X1
LDX CURSND
INX
TXA
AND #$7F
STA CURSND
JSR REFRESH
JMP READ
X1 CMP #45
BNE X2
LDX CURSND
DEX
TXA
AND #$7F
STA CURSND
JSR REFRESH
JMP READ
X2 JSR NUMRANGE
BCS X5
LDY X4+1
CPY #$3A

```

```

      BNE X3
      INY
      STY X4+1
      TXA
      AND #8
      BEQ XX
      LDA #0
      STA CURSND
XX     TXA
      ASL A
      ASL A
      ASL A
      ASL A
      STA $22
      LDA CURSND
      CLC
      ADC #1
      AND #$0F
      ORA $22
      SEC
      SBC #1
      STA CURSND
      LDA CURSND
      AND #$7F
      STA CURSND
      JSR REFRESH
      JSR UNVERT
      JMP READ
X3     DEY
      STY X4+1
      TXA
      AND #$0F
      STA $22
      LDA CURSND
      CLC
      ADC #1
      AND #$F0
      ORA $22
      SEC
      SBC #1
      AND #$7F
      BNE XZ
XZ     STA CURSND
      JSR REFRESH
      JSR UNVERT
X5     JMP READ
Y0     CMP #1
      BEQ Y1
      JMP Z0
Y1     LDA #$61
```

```
LDY #$04
CLC
ADC XPOS
TAX
STX XCOR
STY YCOR
JSR INVERT
JSR READKEY
JSR CHKLETTER
BCS Y4
LDX XCOR
LDY YCOR
STX $22
STY $23
LDY #0
STA ($22),Y
LDY XPOS
STA ($20),Y
LDX XCOR
LDY YCOR
Y5 JSR UNVERT
LDX XPOS
INX
CPX #$10
BEQ Y3
STX XPOS
Y3 JMP READ
Y4 CMP #29
BEQ Y5
CMP #157
BNE Y6
JSR UNVERT
LDX XPOS
DEX
BMI Y3
STX XPOS
JMP READ
Y6 CMP #20
BNE Y3
LDY XPOS
BEQ Y3
Y7 LDA ($20),Y
DEY
STA ($20),Y
INY
INY
CPY #$10
BNE Y7
LDA #$20
DEY
STA ($20),Y
```

```
DEC XPOS
JSR UNVERT
JSR WRITENAME
Z0 JMP READ
CMP #2
BEQ Z1
Z1 JMP B0
LDX #B2
LDY #04
STX XCOR
STY YCOR
JSR INVERT
JSR READKEY
CMP #43
BNE PTX4
LDY #11
LDA ($20),Y
CLC
ADC #1
STA ($20),Y
INY
LDA ($20),Y
ADC #0
AND #0F
STA ($20),Y
JSR WRITEPW
PTX4 JMP READ
CMP #45
BNE PTX5
LDY #11
LDA ($20),Y
SEC
SBC #1
STA ($20),Y
INY
LDA ($20),Y
SBC #0
AND #0F
STA ($20),Y
JSR WRITEPW
PTX5 JMP READ
JSR NUMRANGE
BCS Z6
LDA Z1+1
CMP #B2
BNE Z2
TXA
AND #0F
LDY #12
STA ($20),Y
```

```
Z2      JMP Z4
        CMP #$B3
        BNE Z3
        TXA
        ASL A
        ASL A
        ASL A
        ASL A
        STA $22
        LDY #$11
        LDA ($20),Y
        AND #$0F
        ORA $22
        STA ($20),Y
        JMP Z4
Z3      TXA
        AND #$0F
        STA $22
        LDY #$11
        LDA ($20),Y
        AND #$F0
        ORA $22
        STA ($20),Y
Z4      JSR UNVERT
        LDX Z1+1
        INX
        CPX #$B5
        BNE Z5
        LDX #$B2
Z5      STX Z1+1
        JSR WRITEPW
Z6      JMP READ
B0      CMP #3
        BEQ B4
        JMP E0
B4      LDX #$DA
        LDY #$04
        STX XCOR
        STY YCOR
        JSR INVERT
        JSR READKEY
        LDY #$13
        CMP #43
        BNE B1
        LDA ($20),Y
        CLC
        ADC #1
        ORA #1
        STA ($20),Y
        JSR WRITCTRL
        JMP READ
```

```
B1      CMP #45
        BNE B2
        LDA ($20),Y
        SEC
        SBC #2
        ORA #1
        STA ($20),Y
        JSR WRITECTRL
        JMP READ
B2      JSR NUMRANGE
        BCS B5
        LDY B4+1
        CPY #$DA
        BNE B3
        INY
        STY B4+1
        TXA
        ASL A
        ASL A
        ASL A
        ASL A
        STA $22
        LDY #$13
        LDA ($20),Y
        AND #$0F
        ORA $22
        ORA #1
        STA ($20),Y
        JSR WRITECTRL
        JSR UNVERT
        JMP READ
B3      DEY
        STY B4+1
        TXA
        AND #$0F
        STA $22
        LDY #$13
        LDA ($20),Y
        AND #$F0
        ORA $22
        ORA #1
        STA ($20),Y
        JSR WRITECTRL
        JSR UNVERT
B5      JMP READ
E0      CMP #4
        BEQ E1
        JMP F0
E1      LDX #$02
        LDY #$05
```



```

      STX XCOR
      STY YCOR
      JSR INVERT
      JSR READKEY
      JSR NUMRANGE
      BCS E2
      TXA
E3    ASL A
      ASL A
      ASL A
      ASL A
      STA $22
      LDY #$14
      LDA ($20),Y
      AND #$0F
      ORA $22
      STA ($20),Y
      JSR WRITEADSR
E2    PHA
      LDY #$14
      LDA ($20),Y
      LSR A
      LSR A
      LSR A
      LSR A
      TAX
      PLA
      JSR PLUSMIN
      BCS E3
      JMP READ
F0    CMP #5
      BEQ F1
      JMP G0
F1    LDX #$2A
      LDY #$05
      STX XCOR
      STY YCOR
      JSR INVERT
      JSR READKEY
      JSR NUMRANGE
      BCS F2
      TXA
F3    AND #$0F
      STA $22
      LDY #$14
      LDA ($20),Y
      AND #$F0
      ORA $22
      STA ($20),Y
      JSR WRITEADSR
F2    PHA

```

```
LDY #$14
LDA ($20),Y
AND #$0F
TAX
PLA
JSR PLUSMIN
BCS F3
JMP READ
G0  CMP #6
    BEQ G1
    JMP J0
G1  LDX #$52
    LDY #$05
    STX XCOR
    STY YCOR
    JSR INVERT
    JSR READKEY
    JSR NUMRANGE
    BCS G2
    TXA
G3  ASL A
    ASL A
    ASL A
    ASL A
    STA $22
    LDY #$15
    LDA ($20),Y
    AND #$0F
    ORA $22
    STA ($20),Y
    JSR WRITEADSR
G2  PHA
    LDY #$15
    LDA ($20),Y
    LSR A
    LSR A
    LSR A
    LSR A
    TAX
    PLA
    JSR PLUSMIN
    BCS G3
    JMP READ
J0  CMP #7
    BEQ J1
    JMP K0
J1  LDX #$7A
    LDY #$05
    STX XCOR
    STY YCOR
```

```

        JSR INVERT
        JSR READKEY
        JSR NUMRANGE
        BCS J2
        TXA
J3      AND #$0F
        STA $22
        LDY #$15
        LDA ($20),Y
        AND #$F0
        ORA $22
        STA ($20),Y
        JSR WRITEADSR
J2      PHA
        LDY #$15
        LDA ($20),Y
        AND #$0F
        TAX
        PLA
        JSR PLUSMIN
        BCS J3
        JMP READ
K0      CMP #8
        BEQ K1
        JMP M0
K1      LDX #$0A
        LDA #1
K2      STA $D9A1,X
        DEX
        BPL K2
        JSR READKEY
        CMP #43
        BNE K6
        LDY #$16
        LDA ($20),Y
        LDX #0
K3      CMP MACLOW,X
        BEQ K4
        INX
        CPX #5
        BNE K3
        BEQ K9
K4      INX
        CPX #5
        BNE K5
        LDX #0
K5      LDA MACLOW,X
        STA ($20),Y
        INY
        LDA MACHI,X
        STA ($20),Y

```

```
JSR WRITESPD
JMP READ
K6    CMP #45
      BNE K9
      LDY #16
      LDA ($20),Y
      LDX #0
K7    CMP MACLOW,X
      BEQ K8
      INX
      CPX #5
      BNE K7
      BEQ K9
K8    DEX
      BPL K10
      LDX #4
K10   JMP K5
K9    CMP #20
      BNE K11
      LDY #16
      LDX #1
      JMP K5
K11   JMP READ
M0    CMP #9
      BEQ M1
      JMP 00
M1    LDX #3
      LDA #1
M2    STA $D9C9,X
      DEX
      BPL M2
      JSR READKEY
      CMP #43
      BNE M5
      LDY #1C
      LDA ($20),Y
      BPL M3
      LDA #FF
M3    TAX
      INX
      CPX #60
      BNE M4
      LDX #80
M4    TXA
      STA ($20),Y
      JSR WRITEFIX
      JMP READ
M5    CMP #45
      BNE M9
      LDY #1C
```

```
M6      LDA ($20),Y
        BPL M6
        LDA #$60
        TAX
        DEX
        BPL M7
        LDX #$80
M7      TXA
        STA ($20),Y
        JSR WRITEFIX
M8      JMP READ
M9      CMP #20
        BEQ M10
        CMP #32
        BEQ M10
        JMP READ
M10     LDY #$1C
        LDX #$80
        JMP M4
00      CMP #10
        BEQ 01
        JMP P0
01      LDX #$0F
        LDA #1
02      STA $D9F1,X
        DEX
        BPL 02
        JSR READKEY
        CMP #43
        BNE 04
        LDY #$1D
        LDA ($20),Y
        TAX
        INX
        CPX #5
        BNE 03
        LDX #0
03      TXA
        STA ($20),Y
        JSR WRITEPITCH
        JMP READ
04      CMP #45
        BNE 07
        LDY #$1D
        LDA ($20),Y
        TAX
        DEX
        BPL 05
        LDX #4
05      TXA
        STA ($20),Y
```

```
JSR WRITEPITCH
06    JMP READ
07    CMP #20
      BEQ 08
      CMP #32
      BEQ 08
      JMP READ
08    LDY #1D
      LDX #0
      JMP 05
P0    CMP #11
      BEQ P1
      JMP Q0
P1    LDY #20
      LDA ($20),Y
      BNE P8
      LDA #1
      LDX #02
P2    STA $DA19,X
      DEX
      BPL P2
      JSR READKEY
      CMP #43
      BNE P4
P6    LDA #1
P3    LDY #20
      STA ($20),Y
      INY
      LDA #1
      STA ($20),Y
P7    JSR UNVERT
      JSR WRITEVIB
      JMP READ
P4    CMP #87
      BNE P5
      LDA #80
      BNE P3
P5    CMP #65
      BEQ P6
      JMP P7
P8    BPL PH
      JMP PI
PH    LDX #1E
      LDY #06
      STX XCOR
      STY YCOR
      JSR INVERT
      JSR READKEY
      CMP #20
      BNE P9
```

```
P9      LDA #0
        JMP P3
        CMP #32
        BNE PG
        LDA #0
        JMP P3
PG      CMP #87
        BNE PA
        LDA #$80
        JMP P3
PA      CMP #29
        BNE PA3
        LDX PH+1
        CPX #$26
        BNE PA0
        LDX #$2E
        BNE PA2
PA0     CPX #$2E
        BNE PA1
        LDX #$1E
        BNE PA2
PA1     LDX #$26
PA2     STX PH+1
        JSR UNVERT
        JMP READ
PA3     CMP #157
        BNE PAX
        LDX PH+1
        CPX #$26
        BNE PA4
        LDX #$1E
        BNE PA6
PA4     CPX #$2E
        BNE PA5
        LDX #$26
        BNE PA6
PA5     LDX #$2E
PA6     STX PH+1
        JSR UNVERT
        JMP READ
PAX     JSR NUMRANGE
        BCC PE
        JMP READ
PE      TXA
        LDX PH+1
        CPX #$1E
        BNE PB
        INX
        STX PH+1
        ASL A
        ASL A
```

```
ASL A
ASL A
CMP #$80
BCC PF
LDA #$70
PF STA $22
LDY #$20
LDA ($20),Y
AND #$0F
ORA $22
STA ($20),Y
JMP P7
PB CPX #$1F
BNE PC
LDX #$26
STX PH+1
AND #$0F
STA $22
LDY #$20
LDA ($20),Y
AND #$F0
ORA $22
STA ($20),Y
JMP P7
PC CPX #$26
BNE PD
LDX #$2E
STX PH+1
AND #$0F
BNE PW
LDA #1
PW STA $22
LDY #$21
LDA ($20),Y
AND #$F0
ORA $22
STA ($20),Y
JMP P7
PD LDX #$1E
STX PH+1
ASL A
ASL A
ASL A
ASL A
STA $22
LDY #$21
LDA ($20),Y
AND #$0F
ORA $22
STA ($20),Y
```



```

    JMP P7
PI   LDX #$20
    LDY #$06
    STX XCOR
    STY YCOR
    JSR INVERT
    JSR READKEY
    LDY #$20
    CMP #20
    BNE PQ
    LDA #0
PP   LDY #$20
    STA ($20),Y
    JSR UNVERT
    JSR WRITEVIB
    JMP READ
PQ   CMP #32
    BNE PR
    LDA #0
    JMP PP
PR   CMP #65
    BNE P0
    LDA #1
    JMP PP
P0   CMP #43
    BNE PJ
    LDY #$20
    LDA ($20),Y
    CLC
    ADC #1
    AND #$1F
    ORA #$80
    STA ($20),Y
    JSR WRITEVIB
    JMP READ
PJ   CMP #45
    BNE PK
    LDY #$20
    LDA ($20),Y
    SEC
    SBC #1
    AND #$1F
    ORA #$80
    STA ($20),Y
    JSR WRITEVIB
    JMP READ
PK   JSR NUMRANGE
    BCS PM
    LDY PI+1
    CPY #$20
    BNE PL
```

```
PN      INY
        STY PI+1
        TXA
        CMP #3
        BCC PN
        LDA #2
        ASL A
        ASL A
        ASL A
        ASL A
        STA $22
        LDY #20
        LDA ($20),Y
        AND #1F
        CLC
        ADC #1
        AND #0F
        ORA $22
        CMP #0
        BEQ PTXX
        SEC
        SBC #1
PTXX    AND #1F
        ORA #80
PTX0    STA ($20),Y
        JSR UNVERT
        JSR WRITEVIB
PM      JMP READ
PL      LDY #20
        STY PI+1
        TXA
        AND #0F
        STA $22
        LDY #20
        LDA ($20),Y
        AND #1F
        CLC
        ADC #1
        AND #F0
        ORA $22
        CMP #0
        BEQ TUF0
        SEC
        SBC #1
TUF0    ORA #80
        CMP #A0
        BCC PTX2
        LDA #9F
PTX2    STA ($20),Y
        JSR UNVERT
```

```
      JSR WRITEVIB
      JMP READ
Q0     CMP #12
      BEQ Q1
      JMP U0
Q1     LDY #1E
      LDA ($20),Y
      BNE Q6
      LDA #1
      LDX #02
Q2     STA $DA41,X
      DEX
      BPL Q2
      JSR READKEY
      CMP #43
      BEQ Q3
      CMP #65
      BEQ Q3
      CMP #87
      BNE Q5
      LDA #80
      JMP Q4
Q3     LDA #1
Q4     LDY #1E
      STA ($20),Y
      INY
      LDA #1
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
Q5     JMP READ
Q6     BMI Q7
      JMP QC
Q7     LDX #48
      LDY #06
      STX XCOR
      STY YCOR
      JSR INVERT
      JSR READKEY
      CMP #20
      BNE QF
      LDA #0
QE     LDY #1E
      STA ($20),Y
      INY
      LDA #1
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
      JMP READ
QF     CMP #32
```

```
      BNE QG
      LDA #0
      BEQ QE
QG     CMP #65
      BNE QH
      LDA #1
      BNE QE
QH     CMP #43
      BNE Q8
      LDY #$1E
      LDA ($20),Y
      CLC
      ADC #1
      AND #$1F
      ORA #$80
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
      JMP READ
Q8     CMP #45
      BNE Q9
      LDY #$1E
      LDA ($20),Y
      SEC
      SBC #1
      AND #$1F
      ORA #$80
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
      JMP READ
QD     JSR NUMRANGE
Q9     BCS QD
      TXA
      LDX Q7+1
      CPX #$48
      BNE QB
      INX
      STX Q7+1
      CMP #3
      BCC QA
      LDA #2
QA     ASL A
      ASL A
      ASL A
      ASL A
      STA $22
      LDY #$1E
      LDA ($20),Y
      AND #$1F
```

```

CLC
ADC #1
AND #$0F
ORA $22
CMP #0
BEQ PTYY
SEC
SBC #1
PTYY AND #$1F
ORA #$80
PTX1 STA ($20),Y
JSR UNVERT
JSR WRITEPVIB
JMP READ
QB DEX
STX Q7+1
AND #$0F
STA $22
LDY #$1E
LDA ($20),Y
AND #$1F
CLC
ADC #1
AND #$F0
ORA $22
CMP #0
BEQ TUF1
SEC
SBC #1
TUF1 ORA #$80
CMP #$A0
BCC PTX3
LDA #$9F
PTX3 STA ($20),Y
JSR UNVERT
JSR WRITEPVIB
JMP READ
QC LDX #$46
LDY #$06
STX XCOR
STY YCOR
JSR INVERT
JSR READKEY
CMP #29
BNE QC2
QC0 LDX #$46
CPX QC+1
BNE QC1
LDX #$4E
QC1 STX QC+1
JSR UNVERT

```

```
      JMP READ
QC2   CMP #157
      BEQ QC0
      CMP #20
      BNE QJ
      LDA #0
      LDY #$1E
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
      JMP READ
      CMP #32
      BNE QK
      LDA #0
      BEQ QI
      CMP #87
      BNE QL
      LDA #$80
      BNE QI
      JSR NUMRANGE
      BCC QP
      JMP READ
      LDY QC+1
      CPY #$46
      BNE QN
      INY
      STY QC+1
      CPX #8
      BCC QM
      LDX #7
      TXA
      ASL A
      ASL A
      ASL A
      ASL A
      STA $22
      LDY #$1E
      LDA ($20),Y
      AND #$0F
      ORA $22
      STA ($20),Y
      JSR UNVERT
      JSR WRITEPVIB
      JMP READ
      CPY #$47
      BNE Q0
      LDY #$4E
      STY QC+1
      TXA
      AND #$0F
```

```

        STA $22
        LDY #$1E
        LDA ($20),Y
        AND #$F0
        ORA $22
        STA ($20),Y
        JSR UNVERT
        JSR WRITEPVIB
        JMP READ
Q0      LDY #$46
        STY QC+1
        TXA
        AND #$0F
        BNE QR
        LDA #1
QR       LDY #$1F
        STA ($20),Y
        JSR UNVERT
        JSR WRITEPVIB
        JMP READ
U0      CMP #13
        BEQ U1
        JMP AA0
U1      LDA #0
        BEQ U2
        JMP U8
U2      LDX #$01
        TXA
U3      STA $DA6D,X
        DEX
        BPL U3
        JSR READKEY
        CMP #43
        BNE U5
        LDY #$1B
        LDA ($20),Y
        LSR A
        LSR A
        LSR A
        LSR A
        AND #$07
        TAX
U4      LDA NEXT,X
        ASL A
        ASL A
        ASL A
        ASL A
        ORA #$0F
        STA ($20),Y
        JSR WRITEFILT
        JMP READ
```

```
U5      CMP #45
        BNE U6
        LDY #1B
        LDA ($20),Y
        LSR A
        LSR A
        LSR A
        LSR A
        AND #07
        TAX
        LDA NEXT2,X
        JMP U4
U6      CMP #29
        BNE U7
        INC U1+1
        JSR UNVERT
U7      JMP READ
U8      CMP #1
        BEQ U9
U9      LDX #75
        LDY #06
        STX XCOR
        STY YCOR
        JSR INVERT
        JSR READKEY
        CMP #29
        BNE UA
        LDA #7E
        STA U9+1
        JSR UNVERT
        JMP READ
UA      CMP #157
        BNE UB
        LDA U9+1
        CMP #7E
        BEQ UX
        DEC U1+1
UX      LDA #75
        STA U9+1
        JSR UNVERT
        JMP READ
UB      JSR NUMRANGE
        BCC UC
        JMP READ
UC      LDY U9+1
        CPY #75
        BNE UD
        INY
        STY U9+1
        TXA
```



```

ASL A
ASL A
ASL A
ASL A
STA $22
LDY #$19
LDA ($20),Y
AND #$0F
ORA $22
STA ($20),Y
JSR UNVERT
JSR WRITEFILT
JMP READ
UD
CPY #$76
BNE UE
INY
STY U9+1
TXA
AND #$0F
STA $22
LDY #$19
LDA ($20),Y
AND #$F0
ORA $22
STA ($20),Y
JSR UNVERT
JSR WRITEFILT
JMP READ
UE
CPY #$77
BNE UF
LDY #$7E
STY U9+1
TXA
AND #$0F
STA $22
LDY #$18
STA ($20),Y
JSR UNVERT
JSR WRITEFILT
JMP READ
UF
TXA
ASL A
ASL A
ASL A
ASL A
ORA #$07
LDY #$1A
STA ($20),Y
LDY #$18
LDA ($20),Y
INY

```

```
HAMA      ORA ($20),Y
           BNE HAMA
           INY
           STA ($20),Y
           LDA #$75
           STA U9+1
           LDA #0
           STA U1+1
           JSR UNVERT
           JSR WRITEFILT
           JMP READ
```

```
;-----
; ÔÈÉÓ ÉÓ ÍÁÃÒÏ ÆÄÉÔ!
;-----
```

```
XX0      .BYTE $D8
YY0      .BYTE $06
LX0      .BYTE 0
RAD      .BYTE 0
MEM0     .BYTE $28
EDMEM0   .BYTE $28
KEY      .BYTE 0
```

```
AA0      LDA XX0
           CLC
           ADC LX0
           TAX
           LDA YY0
           ADC #0
           TAY
           STX XCOR
           STY YCOR
           JSR INVERT
           JSR READKEY
           STA KEY
           LDA DOWN
           BEQ AA2
           LDA MEM0
           CLC
           ADC #3
           CMP #4
           BCC AA2
           STA MEM0
           CLC
           LDA RAD
           CMP #5
           BEQ AA1
           INC RAD
           LDA XX0
           CLC
           ADC #28
```

```

        STA XX0
        LDA YY0
        ADC #0
        STA YY0
        JMP AA7
AA1     LDA EDMEM0
        CLC
        ADC #3
        STA EDMEM0
        JMP AA7
AA2     LDA UP
        BEQ AA8
        LDA RAD
        CMP #0
        BEQ AA4
        JMP AA5
AA4     LDA EDMEM0
        CMP #28
        BEQ AA3
AA5     LDA #0
        STA UP
        LDA MEM0
        SEC
        SBC #3
        STA MEM0
        LDA RAD
        BEQ AA6
        DEC RAD
        LDA XX0
        SEC
        SBC #28
        STA XX0
        LDA YY0
        SBC #0
        STA YY0
        JMP AA3
AA6     LDA EDMEM0
        SEC
        SBC #3
        STA EDMEM0
AA3     JMP AA7
AA8     LDA KEY
        CMP #29
        BNE AAA
        JSR AINC
        JMP AA7
AAA     CMP #157
        BNE AAB
        JSR ADEC
        JMP AA7
AAB     JSR NUMRANGE
```

```
BCC AA9
JMP AA7
AA9  LDY #0
    LDA LX0
AJ0  CMP ATAB,Y
    BEQ AJ1
    INY
    BNE AJ0
AJ1  LDA AX,Y
    CLC
    ADC MEM0
    TAY
    LDA LX0
    AND #1
    BNE AJ2
    TXA
    ASL A
    ASL A
    ASL A
    ASL A
    STA $22
    LDA ($20),Y
    AND #$0F
    ORA $22
    STA ($20),Y
    JSR AJ3
    JMP AA7
AJ2  TXA
    AND #$0F
    STA $22
    LDA ($20),Y
    AND #$F0
    ORA $22
    STA ($20),Y
AJ3  LDY #$FD
    LDA #$FE
    STA ($20),Y
    LDA #0
    INY
    STA ($20),Y
    INY
    STA ($20),Y
    JSR AINC
AA7  JSR UNVERT
    LDY EDMEM0
    JSR WRITEMAC
    JMP READ
AINC LDA LX0
    LDX #0
AI0  CMP ATAB,X
```

```

        BEQ AI1
        INX
AI1     BNE AI0
        INX
        LDA ATAB,X
        STA LX0
        RTS
ADEC    LDA LX0
        BEQ AI4
        LDX #5
AI2     CMP ATAB,X
        BEQ AI3
        DEX
AI3     BPL AI2
        DEX
        LDA ATAB,X
        STA LX0
AI4     RTS
ATAB    .BYTE 0,1,14,15,20,21,21
AX      .BYTE 0,0,1,1,2,2
NEXT    .BYTE $01,$02,$04,$04,$05
        .BYTE $00,$00,$00
NEXT2   .BYTE $05,$00,$01,$01,$02
        .BYTE $04,$04,$04
READKEY CLI
L0      LDA $C6
        BNE LX
        LDA $028D
        AND #4
        CMP #4
        BNE L0
        SEI
        LDA #0
        STA $028D
        INC CTRL
        RTS
LX      SEI
        LDA #0
        STA $C6
        LDA $0277
        CMP #3
        BNE L1
        INC RUNSTOP
        LDA #0
        BEQ L3
L1      CMP #17
        BNE L2
        INC DOWN
        LDA #0
        BEQ L4
L2      CMP #13

```

```

      BNE L3
      INC DOWN
      LDA #0
      BEQ L4
L3    CMP #145
      BNE L5
      INC UP
      LDA #0
      BEQ L4
L5    CMP #19
      BNE L6
      INC HOME
      LDA #0
L6    CMP #147
      BNE L4
      INC CLRHOME
      LDA #0
L4    SEI
      RTS
INVERT LDX XCOR
      LDY YCOR
      STX $22
      STY $23
      LDY #0
      LDA ($22),Y
      ORA #$80
      STA ($22),Y
      CLC
      LDA $23
      ADC #$D4
      STA $23
      LDA #1
      STA ($22),Y
      RTS
UNVERT LDX XCOR
      LDY YCOR
      STX $22
      STY $23
      LDY #0
      LDA ($22),Y
      AND #$7F
      STA ($22),Y
      CLC
      LDA $23
      ADC #$D4
      STA $23
      LDA #15
      STA ($22),Y
      LDX #$0F
UN0   STA $D9A1,X
```

```

        STA $D9F1,X
        STA $DA19,X
        STA $DA41,X
        DEX
        BPL UN0
        LDX #3
UN1     STA $D9C9,X
        STA $DA6D,X
        DEX
        BPL UN1
        RTS
FIXCORD LDA #$3A
        STA X4+1
        LDA #$B2
        STA Z1+1
        LDA #$DA
        STA B4+1
        LDA #0
        STA XPOS
        LDA #$1E
        STA PH+1
        LDA #$20
        STA PI+1
        LDA #$48
        STA Q7+1
        LDA #$46
        STA QC+1
        LDA #$75
        STA U9+1
        LDA #0
        STA U1+1
        LDA #$28
        STA MEM0
        STA EDMEM0
        LDA #$D8
        STA XX0
        LDA #6
        STA YY0
        LDA #0
        STA RAD
        LDA #0
        STA LX0
        RTS
PLUSMIN
        CMP #43
        BNE PL0
        INX
        TXA
        AND #$0F
        SEC
        RTS
```

```
PL0      CMP #45
         BNE PL1
         DEX
         TXA
         AND #0F
         SEC
         RTS
PL1      CLC
         RTS
NUMRANGE
         PHA
         LDX #0
N1       CMP N0,X
         BEQ N2
         INX
         CPX #10
         BNE N1
         PLA
         SEC
         RTS
N2       PLA
         CLC
         RTS
N0       .BYTE $30,$31,$32,$33,$34,$35
         .BYTE $36,$37,$38,$39,$41,$42
         .BYTE $43,$44,$45,$46
CHKLETTER
         CMP #32
         BCS CK0
         SEC
         RTS
CK0      CMP #64
         BCS CK1
         CLC
         RTS
CK1      CMP #96
         BCC CK2
         SEC
         RTS
CK2      SEC
         SBC #40
         CLC
         RTS
;-----
CURSND   .BYTE 0
INIT     LDX #0
         STX $D020
         STX $D021
         STX UP
```



```
I0      STX DOWN
        STX RUNSTOP
        STX PARAM
        JSR FIXCORD
        LDA #$35
        STA 1
        LDA #$0F
        STA $D800,X
        STA $D900,X
        STA $DA00,X
        STA $DB00,X
        LDA $F000,X
        STA $0400,X
        LDA $F100,X
        STA $0500,X
        LDA $F200,X
        STA $0600,X
        LDA $F2E8,X
        STA $06E8,X
        INX
        BNE I0
        LDA #$36
        STA 1
        LDA #$15
        STA $D018
        LDA #1
        STA $CC
        LDA #$80
        STA $0291
        STA $028A
        RTS
```

REFRESH

```
        JSR WRITESND
        LDA CURSND
        CLC
        ADC #$40
        STA $21
        LDA #0
        STA $20
        JSR WRITENAME
        JSR WRITETYPE
        JSR WRITEPW
        JSR WRITECTRL
        JSR WRITEADSR
        JSR WRITESPD
        JSR WRITEFIX
        JSR WRITEPITCH
        JSR WRITEVIB
        JSR WRITEPVIB
        JSR WRITEFILT
```

```
LDY #$28
JSR WRITEMAC
RTS

;-----
WRITENAME
LDY #$0F
R0 LDA ($20),Y
STA $0461,Y
DEY
BPL R0
RTS

;-----
WRITESND
LDX CURSND
INX
TXA
PHA
LDX #$3A
LDY #$04
JSR WRITEHEX
PLA
LDX #$3E
LDY #$04
JMP WRITEDEC

;-----
WRITEADSR
LDY #$14
LDA ($20),Y
PHA
LSR A
LSR A
LSR A
LSR A
PHA
LDX #$02
LDY #$05
JSR WRITEH1
PLA
LDX #$06
LDY #$05
JSR WRITED1
PLA
AND #$0F
PHA
LDX #$2A
LDY #$05
JSR WRITEH1
PLA
LDX #$2E
LDY #$05
```

```

        JSR WRITED1
        LDY #$15
        LDA ($20),Y
        PHA
        LSR A
        LSR A
        LSR A
        LSR A
        PHA
        LDX #$52
        LDY #$05
        JSR WRITEH1
        PLA
        LDX #$56
        LDY #$05
        JSR WRITED1
        PLA
        AND #$0F
        PHA
        LDX #$7A
        LDY #$05
        JSR WRITEH1
        PLA
        LDX #$7E
        LDY #$05
        JMP WRITED1
;-----
WRITECTRL
        LDY #$13
        LDA ($20),Y
        PHA
        LDX #$DA
        LDY #$04
        JSR WRITEHEX
        PLA
        STA $22
        LDX #0
C0      LDA C2,X
        ASL $22
        BCC C1
        ORA #$80
C1      STA $04DE,X
        INX
        CPX #8
        BNE C0
        RTS
C2      .BYTE $0E,$10,$13,$14,$04,$12
        .BYTE $13,$07
;-----
WRITEPW
        LDY #$12

```

```
LDA ($20),Y
LDX #$B2
LDY #$04
JSR WRITEH1
LDY #$11
LDA ($20),Y
LDX #$B3
LDY #$04
JMP WRITEHEX

;-----
WRITETYPE
LDY #$16
LDX #0
LDA ($20),Y
INY
CMP #1
BNE R2
R1 LDA MONO,X
STA $0489,X
INX
CPX #10
BNE R1
JMP R3
R2 LDA POLY,X
STA $0489,X
INX
CPX #10
BNE R2
R3 RTS
POLY .BYTE $10,$0F,$0C,$19,$10,$08
      .BYTE $0F,$0E,$09,$03
MONO  .BYTE $0D,$0F,$0E,$0F,$10,$08
      .BYTE $0F,$0E,$09,$03

;-----
WRITESPD
LDY #$16
LDA ($20),Y
LDX #0
SP0  CMP MACLOW,X
BEQ SP1
INX
CPX #5
BNE SP0
SP1  LDA LOWR,X
STA $22
LDA HIWR,X
STA $23
LDY #$0A
SP2  LDA ($22),Y
STA $05A1,Y
```

```

        DEY
        BPL SP2
        RTS
MACLOW  .BYTE $8E,$C7,$63,$31,$98
MACHI  .BYTE $99,$4C,$26,$13,$09
LOWR   .BYTE <T0,<T1,<T2,<T3,<T4,<T5
HIWR   .BYTE >T0,>T1,>T2,>T3,>T4,>T5
T0     .BYTE $08,$01,$0C,$06,$20,$06
       .BYTE $12,$01,$0D,$05,$20
T1     .BYTE $05,$16,$05,$12,$19,$20
       .BYTE $06,$12,$01,$0D,$05
T2     .BYTE $32,$20,$10,$05,$12,$20
       .BYTE $06,$12,$01,$0D,$05
T3     .BYTE $34,$20,$10,$05,$12,$20
       .BYTE $06,$12,$01,$0D,$05
T4     .BYTE $38,$20,$10,$05,$12,$20
       .BYTE $06,$12,$01,$0D,$05
T5     .BYTE $3F,$3F,$3F,$3F,$3F,$3F
       .BYTE $3F,$3F,$3F,$3F,$3F
;-----
WRITEFIX
        LDY #$1C
        LDA ($20),Y
        BMI FX1
        ASL A
        STA $22
        ASL $22
        LDA #0
        ADC #$10
        STA $23
        LDY #3
FX0     LDA ($22),Y
        STA $05C9,Y
        DEY
        BPL FX0
        RTS
FX1     LDY #3
FX2     LDA FX3,Y
        STA $05C9,Y
        DEY
        BPL FX2
        RTS
FX3     .BYTE $0E,$2F,$01,$20
;-----
WRITEPITCH
        LDY #$1D
        LDA ($20),Y
        ASL A
        ASL A
        ASL A
        ASL A

```

```
CLC
ADC #<PI1
STA $22
LDA #0
ADC #>PI1
STA $23
LDY #0F
PI0 LDA ($22),Y
STA $05F1,Y
DEY
BPL PI0
RTS
PI1 .BYTE $0E,$2F,$01,$20,$20,$20
.BYTE $20,$20,$20,$20,$20,$20
.BYTE $20,$20,$20,$20
.BYTE $08,$01,$0C,$06,$20,$0E
.BYTE $0F,$14,$05,$20,$20,$20
.BYTE $20,$20,$20,$20
.BYTE $06,$15,$0C,$0C,$20,$0E
.BYTE $0F,$14,$05,$20,$20,$20
.BYTE $20,$20,$20,$20
.BYTE $08,$01,$0C,$06,$20,$0F
.BYTE $03,$14,$01,$16,$05,$20
.BYTE $20,$20,$20,$20
.BYTE $06,$15,$0C,$0C,$20,$0F
.BYTE $03,$14,$01,$16,$05,$20
.BYTE $20,$20,$20,$20
;-----
WRITEVIB
LDY #020
LDX #015
LDA ($20),Y
BNE VI1
VI0 LDA VI7,X
STA $0619,X
DEX
BPL VI0
RTS
VI1 BPL VI3
AND #07F
PHA
VI2 LDA VI8,X
STA $0619,X
DEX
BPL VI2
PLA
CLC
ADC #1
LDX #020
LDY #06
```

```

        JSR WRITEHEX
        RTS
VI3    PHA
VI4    LDA VI9,X
        STA $0619,X
        DEX
        BPL VI4
        PLA
        LDX #$1E
        LDY #$06
        JSR WRITEHEX
        LDY #$21
        LDA ($20),Y
        PHA
        AND #$0F
        LDX #$26
        LDY #$06
        JSR WRITEH1
        PLA
        LSR A
        LSR A
        LSR A
        LSR A
        LDX #$2E
        LDY #$06
        JSR WRITEH1
        RTS

VI7    .BYTE $0E,$2F,$01,$20,$20,$20
        .BYTE $20,$20,$20,$20
        .BYTE $20,$20,$20,$20,$20
        .BYTE $20,$20,$20,$20,$20,$20
        .BYTE $20

VI8    .BYTE $17,$08,$05,$05,$0C,$20
        .BYTE $24,$20,$20,$20
        .BYTE $20,$20,$20,$20,$20
        .BYTE $20,$20,$20,$20,$20,$20
        .BYTE $20

VI9    .BYTE $01,$0D,$10,$3A,$24,$30
        .BYTE $30,$20,$13,$10
        .BYTE $04,$3A,$24,$30,$20
        .BYTE $04,$0C,$01,$19,$3A,$24
        .BYTE $30

WRITEPVIB
        LDY #$1E
        LDX #$0E
        LDA ($20),Y
        BNE PV1
PV0    LDA PI1,X

```

```

        STA $0641,X
        DEX
        BPL PV0
        RTS
PV1     BPL PV3
        AND #$7F
        PHA
PV2     LDA VI8,X
        STA $0641,X
        DEX
        BPL PV2
        PLA
        CLC
        ADC #1
        LDX #$48
        LDY #$06
        JSR WRITEHEX
        RTS
PV3     PHA
PV4     LDA VI9,X
        STA $0641,X
        DEX
        BPL PV4
        PLA
        LDX #$46
        LDY #$06
        JSR WRITEHEX
        LDY #$1F
        LDA ($20),Y
        AND #$0F
        LDX #$4E
        LDY #$06
        JSR WRITEH1
        RTS
;-----
```

WRITEFILT

```

        LDY #$1B
        LDA ($20),Y
        LSR A
        LSR A
        LSR A
        LSR A
        AND #7
        ASL A
        TAX
        LDY #0
FI0     LDA FI1,X
        STA $066D,Y
        INX
        INY
```



```

    CPY #2
    BNE FI0
    LDY #19
    LDA ($20),Y
    LDX #75
    LDY #06
    JSR WRITEHEX
    LDY #18
    LDA ($20),Y
    LDX #77
    LDY #06
    JSR WRITEH1
    LDY #1A
    LDA ($20),Y
    LSR A
    LSR A
    LSR A
    LSR A
    LDX #7E
    LDY #06
    JSR WRITEH1
    RTS
FI1  .BYTE $2D,$2D,$0C,$10,$02,$10
     .BYTE $3F,$3F,$08,$10,$0E,$03
     .BYTE $3F,$3F,$3F,$3F
;-----
WRITEMAC
    STY $24
    LDY $24      ;WHEREINSOUND
    LDX #0      ;RAD
    JSR WRITEROW
    CLC
    LDA $24
    ADC #3
    TAY
    LDX #1
    JSR WRITEROW
    CLC
    LDA $24
    ADC #6
    TAY
    LDX #2
    JSR WRITEROW
    CLC
    LDA $24
    ADC #9
    TAY
    LDX #3
    JSR WRITEROW
    CLC
    LDA $24

```

```
ADC #12  
TAY  
LDX #4  
JSR WRITEROW  
CLC  
LDA $24  
ADC #15  
TAY  
LDX #5  
JMP WRITEROW
```

;-----

```
WRITEROW LDA #0  
STA $28  
LDA WLO,X  
STA $26  
LDA WHI,X  
STA $27  
STY $25  
TYA  
SEC  
SBC #28  
BEQ WR1  
WR0 INC $28  
SEC  
SBC #3  
BNE WR0  
WR1 LDA $28  
LDX $26  
LDY $27  
JSR WRITEHEX  
CLC  
LDA $26  
ADC #6  
STA $26  
LDA $27  
ADC #0  
STA $27  
LDY $25  
LDA ($20),Y  
LDX $26  
LDY $27  
JSR WRITEHEX  
CLC  
LDA $26  
ADC #4  
STA $26  
LDA $27  
ADC #0  
STA $27  
LDY $25
```

```

LDA ($20),Y
STA $22
CMP #$FE
BNE WR5
LDX #0
BEQ WR6
WR5  CMP #$FF
      BNE WR4
      LDX #8
WR6  LDY #0
WR7  LDA WLX,X
      STA ($26),Y
      INX
      INY
      CPY #8
      BNE WR7
      JMP WR8
WR4  LDY #0
WR2  LDA C2,Y
      ASL $22
      BCC WR3
      ORA #$80
WR3  STA ($26),Y
      INY
      CPY #8
      BNE WR2
WR8  CLC
      LDA $26
      ADC #10
      STA $26
      LDA $27
      ADC #0
      STA $27
      LDY $25
      INY
      LDA ($20),Y
      LDX $26
      LDY $27
      JSR WRITEHEX
      CLC
      LDA $26
      ADC #6
      STA $26
      LDA $27
      ADC #0
      STA $27
      LDY $25
      INY
      INY
      LDA ($20),Y
      LDX $26

```

```
LDY $27
JMP WRITEHEX
WLO .BYTE $D2,$FA,$22,$4A,$72,$9A
WHI .BYTE $06,$06,$07,$07,$07,$07
WLX .BYTE $3C,$05,$0E,$04,$0D,$01
     .BYTE $03,$3E,$3C,$12,$05,$10
     .BYTE $05,$01,$14,$3E
;-----
WRITEHEX STX $22
        STY $23
        LDY #0
        PHA
        LSR A
        LSR A
        LSR A
        LSR A
        TAX
        LDA H0,X
        STA ($22),Y
        INY
        PLA
        AND #$0F
        TAX
        LDA H0,X
        STA ($22),Y
        RTS
WRITEH1 STX $22
        STY $23
        LDY #0
        AND #$0F
        TAX
        LDA H0,X
        STA ($22),Y
        RTS
H0 .BYTE $30,$31,$32,$33,$34
   .BYTE $35,$36,$37,$38,$39
   .BYTE $01,$02,$03,$04,$05
   .BYTE $06
WRITEDEC STX $22
        STY $23
        LDY #2
        PHA
        LDA #$30
D1 STA ($22),Y
   DEY
   BPL D1
   INY
   PLA
D2 CMP #100
   BCC D3
```

```

SEC
SBC #100
JSR INKA
JMP D2
D3  INY
D4  CMP #10
    BCC D5
    SEC
    SBC #10
    JSR INKA
    JMP D4
D5  INY
    TAX
D6  CPX #0
    BEQ D7
    DEX
    JSR INKA
    JMP D6
D7  RTS
INKA PHA
    LDA ($22),Y
    CLC
    ADC #1
    STA ($22),Y
    PLA
    RTS
D0  .BYTE 0
WRITED1 STX $22
    STY $23
    LDY #0
    CMP #10
    BCC D8
    PHA
    LDA #$31
    STA ($22),Y
    INY
    PLA
    SEC
    SBC #10
    CLC
    ADC #$30
    STA ($22),Y
    RTS
D8  CLC
    ADC #$30
    STA ($22),Y
    INY
    LDA #$20
    STA ($22),Y
    RTS

```

;-----

From:

<https://codebase64.org/> - **Codebase 64 wiki**

Permanent link:

https://codebase64.org/doku.php?id=base:triad_midislave_manager_v1.1

Last update: **2015-04-17 04:34**

