

Boxing

```
SPACE=START  
SHIFT=REDEFINE
```

```
P=LEFT    L=RIGHT    Q=UP        A=DOWN
```

```
033  33  1/4  FINAL  BOXING  03  000
```



For Boxing I wanted to display 4 or 5 graphics on a line. That seemed too much. With a 3 character display I was able to built it how it should be. Your goal is to reach the final where each next contestant will be better than the previous.

```
; Boxing
```

```
? * TORNADO *
```

```
ORG    #4009                ;#4009  
DUMP 49161
```

```
udglines EQU  init+39
```

```
basic    LD    B,5           ; preset for 48K bug  
          JR    init0
```

```
DEFB 236,212,28             ; The BASIC  
DEFB 126                     ; fully placed over sysvar  
DEFB 143,0,18                ; start to BASIC=#4009
```

```
eline    DEFW last           ; needed by loading  
chadd     DEFW last-1  
xptr      DEFW 0  
stkbot    DEFW last  
stkend     DEFW last  
berg      DEFB 0  
mem        DEFW 0  
          DEFB 128
```

```

init1      JP      init          ; init can be anywhere

; all above reusable AFTER loading

lastk      DEFB 255,255,255      ; used by ZX81
margin     DEFB 55               ; used by ZX81
nxtlin     DEFW basic           ; reusable after load

init0      XOR     A             ; delay intrupts by
          DEFB 254               ; CP n ; skip flagx
flagx      DEFB 0

          EX      AF,AF'        ; intruptcounter reset
          DEFB #3A              ; LD A,(nn) ; skip taddr

taddr      DEFW 3213             ; used by ZX81
          LD      E,L           ; low byte equal 48K bug
          DEFB #3A              ; LD A,(NN) ; skip frames

frames     DEFW 65535            ; used by ZX81
coords     JR      init1        ; useable
prcc       DEFB 188              ; used by ZX81
sposn      DEFB 33,24           ; used by ZX81
cdflag     DEFB 64               ; used by ZX81

head2      EQU     cdflag+4      ; lowbyte pointers
head2m     EQU     head2+8       ; for all graphics
head1      EQU     head2+16
head4      EQU     head1+7
head3m     EQU     head4+8
head3      EQU     head4+16
boxlu      EQU     head3+8
boxdl      EQU     boxlu+7
punch      EQU     boxdl+8
boxru      EQU     punch+8
boxrd      EQU     boxru+7

h1         EQU     head1*256/256
h2         EQU     head2*256/256
h4         EQU     head4*256/256
h3         EQU     head3*256/256
z0         EQU     cdflag*256/256
lu         EQU     boxlu*256/256
dl         EQU     boxdl*256/256
pu         EQU     punch*256/256
ru         EQU     boxru*256/256
rd         EQU     boxrd*256/256
ml         EQU     head2m*256/256
mr         EQU     head3m*256/256

          DEFB 0                ; space
          DEFW 0,0,0

          DEFB 0                ; head2
          DEFB 120,228
          DEFB 236,228

          DEFB 252,252          ; head2m
          DEFB 254,255
          DEFB 255,254
          DEFB 252,252

```

```

DEFB 228,236          ; head1
DEFB 228,120
DEFB 0,0,0

DEFB 0,0,0,0          ; head4
DEFB 30,39
DEFB 55,39

DEFB 63,63            ; head3m
DEFB 127,255
DEFB 255,127,63,63

DEFB 39,55            ; head 3
DEFB 39,30
DEFB 0,0,0,0

DEFB 92,222,223,223   ; boxlu
DEFB 227,255,254
DEFB 124,254,255,227   ; boxru
DEFB 223,223,222,92

DEFB 0,0,255,255      ; punch
DEFB 255,255,0,0

DEFB 58,123,251,251   ; boxru
DEFB 199,255,127

DEFB 62,127,255,199   ; boxrd
DEFB 251,251,123,58

; some lowres
hr      LD    HL,lowres+#8000   ; the lowres display
        LD    BC,#A01          ; 11 minimum needed
        LD    A,#1E
        LD    I,A
        LD    A,#FB
        CALL  #2B5

hr00    LD    B,03              ; sync hires display
        DJNZ  hr00
        LD    A,(HL)

        LD    IX,#80D           ; 13 positions 8 lines/pos
        LD    BC,udglines-3     ; pointer udg's
        LD    H,#40             ; udg1/256
        LD    A,H
        LD    I,A              ; set high pointer
        LD    D,A              ; line also same highbyte

cloop   PUSH  HL                ; timing
        INC  SP                 ; timing
        INC  SP                 ; timing

        INC  BC                 ; undo previous read
        INC  BC                 ; undo
        INC  BC                 ; undo

nline   XOR   A                 ; reset destination
        LD    E,A               ; in 8 tstates
        LD    A,(BC)            ; get udg
        DEFB  #DD
        ADD  A,H                ; add indexpointer

```

```

        LD    L,A                ; point to right udgline
        LDI                     ; copy udg

        LD    A,(BC)             ; get next udg
        DEFB #DD
        ADD   A,H
        LD    L,A
        LDI

        LD    A,(BC)             ; get third udg
        DEFB #DD
        ADD   A,H
        LD    L,A
        LDI

        XOR    A                 ; reset for R register
        DEFB #DD
        DEC    H                 ; decrease number of lines
        JP     lbuf+#8000        ; do display

lbuf    LD    R,A
        DEFB 0,0,0              ; only 3 bytes display
        JP     NZ,cloop          ; 48K bug
        JP     bloop            ; 48K bug

bloop   DEFB #DD                ; ld ixh,8
        LD    H,8               ; next 8 lines counter
        DEFB #DD                ; dec ixl
        DEC    L                ; test all lines done
        JR     NZ,nline

; fixed end of HR-routine
        CALL #292                ; back from intrupt
        CALL #220
        LD    IX,hr
        JP     #2A4

gameend LD    HL,score-1
        LD    DE,hiscore-1
        LD    BC,4
fihi    DEC    C
        JR     Z,start
        INC    HL
        INC    DE
        LD    A,(DE)
        CP     (HL)
        JR     Z,fihi
        JR     NC,start
        LDIR

start   LD    A,127
        IN     A,(254)
        RRA
        JR     NC,play          ; SPACE = start

        LD    A,254
        IN     A,(254)
        RRA
        JR     C,start

; redefine routine
        LD    HL,showkey

```

```

LD    DE,keys

defloop SET    7,(HL)
waitup LD    A,(lastk)
      INC    A
      JR     NZ,waitup

waitdown LD    BC,(lastk)
        LD    A,C
        INC    A
        JR     Z,waitdown
        PUSH  HL
        PUSH  DE
        CALL  #7BD
        POP   DE
        LD    (DE),A
        LD    HL,#7D
        ADD   A,L
        LD    L,A
        LD    A,(HL)
        POP   HL
        LD    (HL),A
        INC   DE
        LD    BC,8
        ADD   HL,BC
        LD    A,L
        CP    endred*256/256
        JR    C,defloop
        JR    start

play    LD    HL,round
        LD    (HL),36          ; reset 1/8 finals

        LD    HL,#1C1C        ; reset score
        LD    (score),HL
        LD    (score+1),HL

nxtlev  LD    HL,#6363        ; 99 energy player and comp
        LD    (basic),HL

        LD    HL,enrpl
        CALL  setenrgy-2

        LD    HL,enrcp
        CALL  setenrgy-2

; reset start xy and retract punch
LD    HL,#400
LD    (ypl+1),HL
LD    (ycp+1),HL

playlp  LD    BC,(lastk)
        LD    A,C
        INC    A
        JR     Z,setold      ; no key pressed
        CALL  #7BD          ; translate to ascii
old      CP    0              ; test old keypress
setold  LD    (old+1),A      ; save next key

        LD    DE,(ypl+1)    ; get Y and punchpointer
        LD    B,D           ; save original Y
        LD    HL,keys

        CP    (HL)

```

```

        INC    HL

        LD     E,1                ; preset down punch
        JR     Z,punch2          ; punch up, test on twice

        CP     (HL)
        INC    HL

        JR     NZ,nopunch        ; other key than punch

punch2  LD     E,#10              ; preset up punch
        CP     0
        JR     NZ,uptest         ; punch allowed, moved

nopunch LD     E,0                ; reset punch value

uptest  CP     (HL)
        INC    HL
        JR     NZ,downtest

        DEC    D                ; move up

downtest CP    (HL)
        JR     NZ,moveout

        INC    D                ; move down

moveout LD     A,D
        AND    7
        JR     NZ,save           ; do not move out of screen
        LD     D,B              ; undo move

save    LD     HL,(ypl+1)
        SBC    HL,DE             ; test valid key pressed
        LD     (ypl+1),DE        ; save "altered" Y and punch
        LD     A,(old+1)         ; get pressed key
        JR     Z,movecp          ; no valid key pressed
        LD     (punch2+1),A      ; save new key pressed

movecp  CALL    cprnd
        JR     NZ,nomove

        LD     A,(ypl+2)
        LD     HL,ycp+2
        SUB    (HL)
        JR     Z,nopu1
        LD     C,2               ; punch up
        JR     C,pu2
        LD     C,#20             ; punch down
pu2     ADD    A,3
        CP     7
        JR     NC,nopu1

        CALL    cprnd
        JR     NZ,nopu1

        LD     A,C
        LD     (ycp+1),A
        JR     nomove

nopu1   LD     A,2               ; up or down
        CALL    rnd
        LD     HL,ycp+2
        LD     A,(HL)

```

```

        JR    Z,downcpu
        INC   A
        INC   A
downcpu  DEC   A
        AND   7
        JR    Z,nomove
        LD    (HL),A

nomove   LD    HL,udglines      ; erase old values on hiscreen
        LD    B,39
cls      LD    (HL),z0         ; set spaces
        DEC   HL
        DJNZ  cls

        LD    HL,udglines+3
ypl      LD    BC,#00          ; #11 is punch up / down
        LD    DE,dispord-1
        CALL  clpl            ; display player

ycp      LD    BC,#00          ; #22 is punch up / down
        LD    HL,udglines+1
        CALL  clpl            ; display computer

wait     LD    HL,frames      ; game delay
        LD    A,(HL)
        SUB   4
wfr      CP    (HL)
        JR    NZ,wfr

        LD    A,(ypl+1)
        OR    A
        LD    DE,basic
        JR    Z,pucomp

        LD    A,(DE)
        SUB   3
        JR    C,dead
        INC   A
        LD    (DE),A
        LD    HL,enrpl
        CALL  setenrgy

pucomp   INC   DE
        LD    A,(ycp+1)
        OR    A
        JR    Z,testphit

        LD    A,(DE)          ; also for computer
        SUB   3
        JR    C,nxtrnd
        INC   A
        LD    (DE),A
        LD    HL,enrcp
        CALL  setenrgy

testphit LD    HL,(ypl+1)
        LD    BC,(ycp+2)      ; only Y in C needed
        LD    B,1
        CALL  hardhit
        LD    A,(DE)          ; energy computer
        SUB   B
        INC   A
        LD    (DE),A
        DEC   DE

```

```

        JR    C,nxtrnd
        LD    HL,enrcp
        CALL  setenrgy

testchit  LD    HL,(ycp+1)
          LD    BC,(ypl+2)
          LD    B,2
          CALL  hardhit
          LD    A,(DE)
          SUB   B
          INC   A
          LD    (DE),A
dead      JP    C,start
          LD    HL,enrpl
          CALL  setenrgy

          XOR   A
          LD    (ypl+1),A      ; retract punch
          LD    (ycp+1),A      ; retract punch

          JP    playlp

nxtrnd    LD    A,(DE)          ; get remaining energy
          LD    B,A            ; add as points
addsc     LD    HL,score+3
          DEFB  17
ten       LD    (HL),28
          DEC   HL
          INC   (HL)
          LD    A,(HL)
          CP    38
          JR    Z,ten
          DJNZ  addsc

          LD    A,(round)      ; get current final
          ADD   A,28           ; add "0" to current
          RRA                ; divide by 2
          CP    28
          JP    Z,gameend      ; final won!
          LD    (round),A      ; number is halved

          LD    HL,frames
          LD    A,(HL)
          INC   A
wfr2     CP    (HL)            ; >5 sec rest for
          JR    NZ,wfr2        ; next round starts
          JP    nextlev

cprnd     LD    A,(round)
          SUB   28
          ADD   A,A
          ADD   A,A

rnd       LD    HL,(frames)
          LD    B,A
rseed     LD    DE,0
          ADD   HL,DE
          DEC   HL
          LD    A,H
          AND   #1F
          LD    H,A
          LD    (rseed+1),HL
          LD    A,(HL)
frnd      SUB   B

```



```

        JR    NC,frnd
        ADD   A,B
        RET

hardhit  LD    A,L
        CP    B                ; did I punch?
        LD    B,1
        RET    C                ; no punching
HIER     LD    A,H                ; Y player
        JR    Z,frnd
        ADD   A,4
fnd      INC    C                ; face is 1 position more
        SUB   C
        CP    3                ; 0,1,2 gives C, no hit others
        RET   NC
        DEC   A                ; test 1, full hit
        LD    B,11
        RET   Z
        LD    B,6                ; partial hit
        RET

setenrgy LD    A,99
ft       LD    (HL),27+128
        INC   (HL)
        SUB   10
        JR    NC,ft
        INC   HL
        ADD   A,38+128
        LD    (HL),A
        RET

keys     DEFB  25,31,10,5

clpl     DEC   HL
        DEC   HL
        DEC   HL
        DJNZ  clpl                ; find start of display

disp     LD    B,5                ; do 5 displays
        DEC   HL
        DEC   HL
        DEC   HL
        INC   DE
        LD    A,(DE)                ; get display
        LD    (HL),A                ; set on screen
        LD    A,C                ; get punch value
        AND   15
        JR    Z,noset                ; test upperpunch
        LD    (HL),pu                ; show arm
        PUSH  HL                ; save pointer
        DEC   A
        JR    Z,cpunch                ; test computer or player
        INC   HL
        INC   HL
cpunch   DEC   HL
        LD    A,(DE)                ; again get fist
        LD    (HL),A                ; show on other position
        POP   HL

noset    LD    A,C                ; get punchpointer
        AND   #F0                ; take off upperpunch
        LD    C,A                ; save pointer
        DJNZ  disp                ; do full display

```

```

LD      A,C                ; get lowpunch
AND     #F0
RET     Z                  ; ready on no punch
LD      (HL),pu            ; show arm
CP      #10                ; again test player / computer
JR      Z,cpu2
INC     HL
INC     HL
cpu2    DEC     HL
LD      A,(DE)             ; get fist
LD      (HL),A            ; show fist
RET

n       EQU     27
x       EQU     101

lowres  DEFB 118

        DEFB "S"-n,"P"-n,"A"-n,"C"-n,"E"-n,20
        DEFB "S"-n,"T"-n,"A"-n,"R"-n,"T"-n,118

        DEFB "S"-n,"H"-n,"I"-n,"F"-n,"T"-n,20
        DEFB "R"-n,"E"-n,"D"-n,"E"-n,"F"-n
        DEFB "I"-n,"N"-n,"E"-n,118,118

Showkey DEFB "P"-n,20,"L"-n,"E"-n,"F"-n,"T"-n,0,0
        DEFB "L"-n,20,"R"-n,"I"-n,"G"-n,"H"-n,"T"-n,0

        DEFB "Q"-n,20,"U"-n,"P"-n,0,0,0,0
        DEFB "A"-n,20,"D"-n,"O"-n,"W"-n,"N"-n,0

endred  DEFB 118,118,118,118,118

score   DEFB 28,28,28,0
enrpl   DEFB 28,28,0
        DEFB 29,24
round   DEFB 29,0
        DEFB "F"-n,"I"-n,"N"-n,"A"-n,"L"-n,0
        DEFB "B"-n,"O"-n,"X"-n,"I"-n,"N"-n,"G"-n,0
enrcp   DEFB 28,28,0
hiscore DEFB 28,28,28
        DEFB 118

dispord DEFB lu,h1,ml,h2,d1
dispcp  DEFB ru,h3,mr,h4,rd

; BASIC initialization done on the screen
screen  EQU     $
init    LD      IX,hr        ; 04 Hires mode
        LD      SP,#4400     ; 07
        LD      H,#3F        ; 09 #3fxx
        LD      D,#BF        ; 11 #bfxx
        LDIR             ; 13 repair 48K bug

        LD      HL,screen
        LD      (HL),z0
        LD      DE,screen+1
        LD      C,39
        JP      start-2

vars    DEFB 128                ; 39

last    EQU     $

```