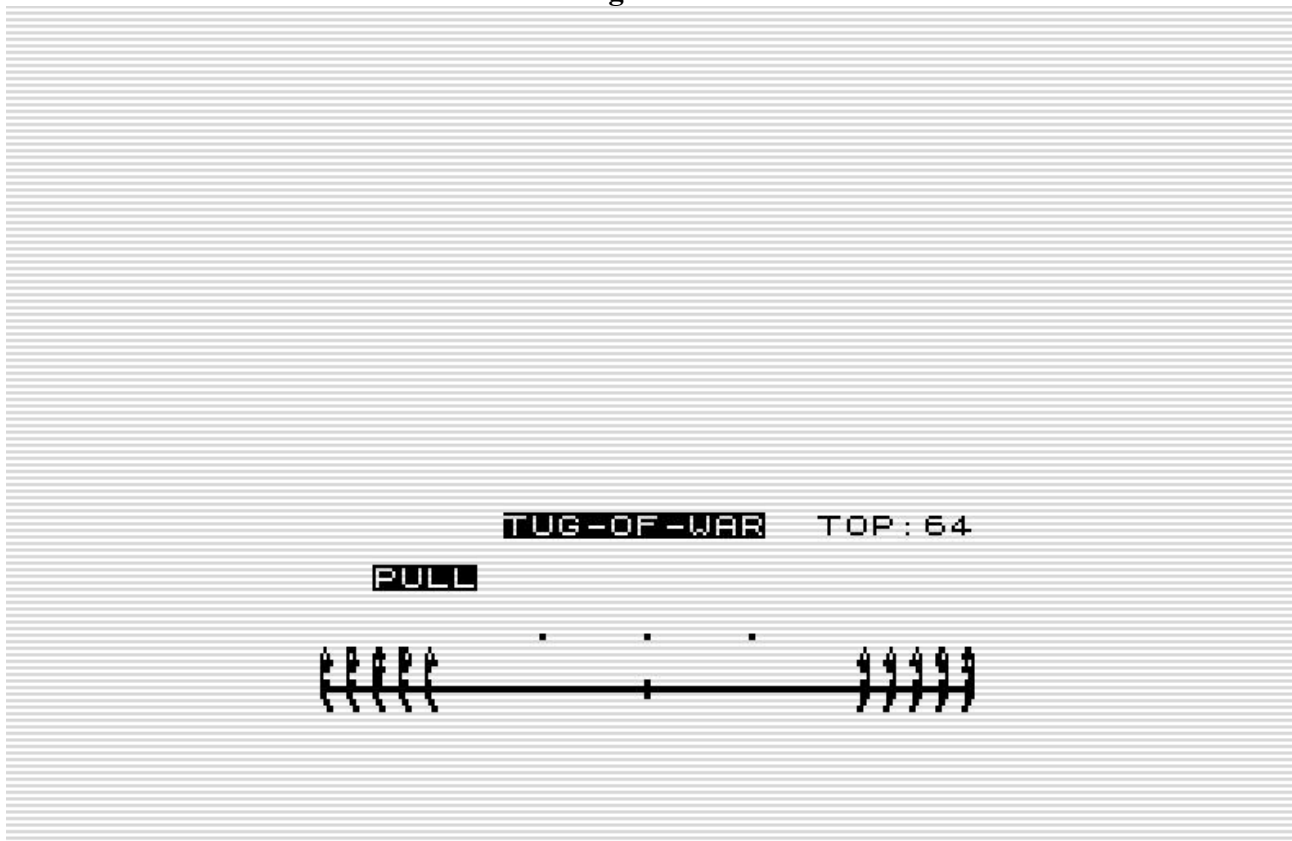


## Tug of War



**This is a simple hires routine. It is a fixed screen that can shift per 2 pixels left or right. The screen is at the bottom so there is no need to end the screen on a fixed position. Therefore only at the start of the hirescreen the outlining of the display is done.**

```
; Tug of War
; Game 64 in 1K hires for the ZX81

? * TORNADO *

                ORG  #4009
                DUMP 49161

basic           LD    D,#C0                ; preset for 48K bug
                JR     init0              ; this game has no 48K bug

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

eline           DEFW last                  ; needed to load
chadd           DEFW last-1
xptr            DEFW 0
stkbot          DEFW last                  ; needed to load
stkend          DEFW last                  ; needed to load
berg            DEFB 0
mem             DEFW 0
                DEFB 0

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      LD  E,L               ; delay intrupts by
           DEFB #26              ; LD H,64
flagx      DEFB 64              ; clever setting of flags

           XOR  A                ; intruptcounter reset
           EX  AF,AF'

taddr      DEFW 0                ; used by ZX81,no hurting code
           LD  B,4               ; frames is set ok

frames     DEFW #DD*256+1        ; used by ZX81, clever IX set
coprcc     LD  HL,hr             ; set IX
sposn      JR  init1
cdflag     DEFB 64              ; used by zx81

delaytab   NOP
           RET  C
           DEC  HL
           LD  A,(HL)

pulltxt    DEFB "P"+x,"U"+x,"L"+x,"L"+x

; routine to write PULL on screen on 2 locations
copytxt    LD  HL,pulltxt
           PUSH BC
           LD  BC,4
           LDIR
           POP  BC
           RET

testwin    OR  C
           JR  NZ,start          ; gameover, computer won
           LD  HL,playloop+1
           DEC (HL)              ; 1 level up
           LD  A,H
           CALL delay            ; show end screen a while
           LD  HL,delaycnt+1
           DEC (HL)              ; faster response each level
           JR  levelup

start      LD  A,%10111111      ; game over, wait for
           IN  A,(254)
           RRA                   ; newline
           JR  C,start

           LD  A,16              ; start delay
           LD  (delaycnt+1),A

           RRCA
           LD  (playloop+1),A    ; start with best of 64

levelup    LD  HL,playloop+1
           LD  B,(HL)
           LD  HL,topnr-4

findnr     INC  HL
           INC  HL
           DJNZ findnr
           LD  DE,level
```

```

LDI                ; copy correct top number
LDI
LD    A,L
CP    topnr*256/256+2
JR    Z,start      ; check best of world

LD    C,16         ; startpos of rope

newshout    CALL    makescreen      ; draw the screen

LD    A,C
AND    31
JR    Z,testwin    ; left or right on screen

LD    HL,plcom
LD    B,20
clpull      LD    (HL),0            ; reset possible text
INC    HL
DJNZ    clpull

newrnd      CALL    rnd              ; get rnd 0-3
oldrnd      CP    0                  ; test against previous
JR    Z,newrnd      ; not the same, get new rnd
LD    (oldrnd+1),A    ; save new as next old

LD    B,A            ; store number in B
RRA                ; test bit 0
LD    DE,plcom
CALL    C,copytxt    ; show PULL player
LD    E,aicom*256/256
RRA                ; test bit 1
CALL    C,copytxt    ; show PULL computer

LD    A,B            ; get displaynumber
XOR    2             ; swap bit computer
AND    2             ; save false bit computer only

delaycnt    LD    E,16
LD    HL,storinp+1
LD    (HL),A        ; player ok, comp false move

CALL    rnd+2        ; some displaytime
ADD    A,E           ; add minimal 16
LD    D,A           ; loopcounter to D

playloop    LD    E,8            ; computer level
CALL    rnd+2
LD    HL,storinp+1
JR    NZ,moveplay    ; only Z will move comp

skip        LD    A,0
DEC    A
AND    3
LD    (skip+1),A    ; but only 1/8 moves is done
JR    NZ,moveplay    ; not too often a right move

; repair the false move
RES    1,(HL)        ; signal no pull
BIT    1,B           ; test if pull is needed
JR    Z,moveplay     ; if not, ready
SET    1,(HL)        ; set computer reacts ok

moveplay    LD    A,%01000000    ; all keys but H-NewLine
IN    A,(254)
CPL

```

```

        AND    31
        JR     Z,delaylp          ; no key pressed

playingp    SET    0,(HL)          ; signal key pressed

delaylp     LD     A,254           ; some delay for display
            CALL  delay

            DEC    D
            JR     NZ,playloop     ; random delay counter

storingp    LD     A,0             ; get keys comp and player
            XOR    B               ; test against needed values
            RRA                   ; test move player
            JR     NC,ok1          ; no C is matched move
            INC    C               ; move to computer
ok1          RRA                   ; test move computer
            JR     NC,ok2          ; no C, computer moved ok
            DEC    C               ; move to player
ok2          JP     newshout       ; get a new coach shout

space       EQU    #4103-$        ; screen must start on ..03
            DEFS   space

; the screen with men and rope
screen      DEFB   64,224,96,224,64
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   2,2,2,6,6
s2          DEFB   160,160,240,208,160
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   5,5,5,13,15
s3          DEFB   240,240,160,240,240
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   15,15,15,15,5
s4          DEFB   224,224,240,224,192
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   7,7,3,7,15
s5          DEFB   128,128,224,128,128
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   1,1,1,1,3
s6          DEFB   224,224,224,224,192
            DEFB   0,0,0,0,0,0,0,24
            DEFB   0,0,0,0,0,0,0
            DEFB   7,3,7,7,7
s7          DEFB   255,255,255,255,255,255,255,255
            DEFB   255,255,255,255,255,255,255,255
            DEFB   255,255,255,255,255,255,255,255
s8          DEFB   192,192,192,192,192
            DEFB   0,0,0,0,0,0,0,24
            DEFB   0,0,0,0,0,0,0
            DEFB   7,7,7,7,7
s9          DEFB   96,96,96,96,96
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   6,6,6,6,6
sa          DEFB   48,48,48,48,48
            DEFB   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
            DEFB   12,12,12,12,12

hr          LD     HL,lowres+#8000 ; the lowres display
            LD     BC,#689         ; minimum lines in this game
            LD     A,#1E           ; needed to prevent scrolling
            LD     I,A
            LD     A,#FB

```

```

CALL #2B5                ; show lowres

hr00    LD    B,8          ; filler
        DJNZ hr00

setjump  JP    tstate47-1    ; start of display alters

tstate47 NOP
        DEFW 0,0,0,0
        XOR   A              ; RESET CARY
        NOP                ; changed for display
        RET   C              ; 5 tstate filler
        LD    A,screen/256
        LD    I,A

        LD    B,10          ; 10 lines
        LD    C,2           ; each line 2x
        LD    L,screen*256/256

double  PUSH HL
        POP  HL
        XOR  A
        RET  C

display LD    A,L
        CALL lbuf+#8000

        PUSH HL
        POP  HL

        DEC  C
        JR   NZ,double

        LD    A,L
        ADD  A,25
        LD    L,A
        LD    C,2
        DJNZ display

        CALL #292            ; back from intrupt
        CALL #220
        LD    IX,hr
        JP    #2A4

lbuf    LD    R,A
        DEFW 0,0,0,0,0,0,0,0,0,0,0,0,0
        DEFB 0
        RET

x       EQU   101
n       EQU   27

lowres  DEFB 118

        DEFW 0,0,0,0,0,0

        DEFB "T"+x,"U"+x,"G"+x,150,"O"+x,"F"+x,150
        DEFB "W"+x,"A"+x,"R"+x,0,0
        DEFB "T"-n,"O"-n,"P"-n,14

level  DEFB 34,32          ; "64"

```

```

        DEFB 118,118
        DEFB 0,0,0,0,0,0,0
plcom    DEFB "P"+x,"U"+x,"L"+x,"L"+x

        DEFW 0,0,0,0,0,0

aicom    DEFB "P"+x,"U"+x,"L"+x,"L"+x

        DEFB 118,118
        DEFB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,27
        DEFB 0,0,0
        DEFB 27,0,0,0,27
        DEFB 118

rnd       LD     E,4
        PUSH  DE
        LD     A,(frames)
rseed    ADD     A,#44
        LD     D,A
        RRCA
        RRCA
        RRCA
        XOR    #1F
        ADD    A,D
        SBC    A,#FF
        LD     (rseed+1),A
        POP    DE
sube     SUB     E
        JR     NC,sube
        ADD    A,E
        RET

makescreen LD     A,C
        LD     HL,tstate47-1
fhl      DEC     HL
        SUB     4
        JR     NC,fhl
        LD     (setjump+1),HL    ; 4 tstate delay set ok
        AND     3
        LD     HL,delaytab
        ADD    A,L
        LD     L,A
        LD     A,(HL)
        LD     (tstate47),A      ; last matching tstates set
        RET

delay    LD     HL,frames
        ADD    A,(HL)
wfr      CP      (HL)
        JR     NZ,wfr
        RET

init     LDIR                                ; repair 48K bug, standard
        JP     start                  ; not in this game

; 01 02 04 08 16 32 64
topnr    DEFB 28,29,28,30,28,32,28,36,29,34
        DEFB 31,30,34,32

vars     DEFB 128
last     EQU    $

```