

## Stupid Cupid



**When I saw “Patrulha Espacial” I thought I could code something similar in 1K and hires. This is the result. I only made a bit more friendly. Cupid dropped his bag with love hearts. Can he shoot them with his arrows? I also managed to shorten the default initialization with a few bytes. Still bytes left, so no further optimization needed.**

```
; Stupid Cupid
; Gameplay based on "Patrulha Espacial"
; Controls
; 1 = fire
; 9 = left
; 0 = right
; NL = start game

? * TORNADO *

                ORG   #4009                ;#4009
                DUMP  49161

screen          EQU   init+33

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

initl     JP    init

; 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 17              ; LD DE,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    initl          ; useable

prcc      DEFB 188              ; used by ZX81
sposn     DEFB 33,24            ; used by ZX81
cdflag    DEFB 64               ; used by ZX81

udg       DEFB 6,96,24          ; hearts udg and arrow
           DEFB 137,145,60
           DEFB 80,10,118
           DEFB 240,15,44
           DEFB 112,14,44
           DEFB 248,31,44
           DEFB 116,46,52
           DEFB 50,76,126
           DEFB 1,128,90
           DEFB 0,0,0

udgcupid  DEFB 62,0,24          ; cupid udg and arrow
           DEFB 65,0,60          ; arrow is double stored
           DEFB 158,0,118        ; for displaymethod
           DEFB 162,0,44
           DEFB 170,0,44
           DEFB 163,0,44
           DEFB 230,0,52
           DEFB 162,0,126
           DEFB 60,0,90
           DEFB 152,192,192
           DEFB 85,64,64
           DEFB 246,64,64
           DEFB 112,128,128
           DEFB 247,0,0
           DEFB 180,0,0,28

lbuf      DEFB 0                ; linebuffer hidden in UDG
           DEFB 0,20,0           ; 20 bytes double used
           DEFB 0,20,0

```

```

        DEFB 0,18,0
        DEFB 0,25,0
        DEFB 0,22,0
        DEFB 0,0,0,0
        JP   Z,cloop          ; all lines in block shown, 48K bug
        EXX
        JP   (IX)             ; stay in same block

tabp2   DEFB #3B,#3A,#39,#38 ; rotation table part2
        DEFB #48,#47,#57,#56
        DEFB #66,#65,#75,#76
        DEFB #86,#87,#97,#98
        DEFB #A8,#A9,#AA,#AB
        DEFB #9B,#9C,#8C,#8D
        DEFB #7D,#7E,#6E,#6D
        DEFB #5D,#5C,#4C,#4B

hr      LD   HL,lowres+#8000  ; the lowres display
        LD   BC,#209         ; minimum needed
        LD   A,#1E
        LD   I,A
        LD   A,#FB
        CALL #2B5

        EX   (SP),HL         ; sync hires with lowres
        EX   (SP),HL
        LD   A,(HL)

        LD   IX,blow         ; save 2 tstate in display
        LD   HL,lbuf+#8000   ; save 6 tstate in display
        LD   BC,#100A        ; 16 hearts, size 10 lines
        LD   E,C             ; save 3 tstate in display
        EXX
        LD   D,#40           ; actual hiresdata location
        LD   A,D
        LD   I,A
        LD   H,D             ; also same highbyte of udg
        LD   BC,screen       ; the screendata
        CALL line2-2         ; display topscreen

        LD   B,13            ; outline cupid with hearts
fill    DJNZ fill

        INC   BC              ; filler
        LD   BC,#116         ; 21 lines for 1 cupid
        EXX
        CALL line-1          ; display bottomscreen

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

cloop   DEC   B               ; decrease block counter
        LD   C,E             ; reset nr of lines next block
        RET   Z               ; return end reached
        EXX
        LD   L,udg*256/256   ; point to start of udg

line2   LD   A,(BC)           ; get current x
        LD   E,A             ; point to x on line
        LDI                     ; copy udg to xpos, dec next x
        LD   A,(BC)          ; get next x

```

```

LDI                ; copy second part 16 bits udg
LD  E,A            ; point to x next udg
LD  A,(HL)         ; get udg
LD  (DE),A         ; write udg

EXX
XOR  A             ; load start of data
DEC  C             ; signal line displayed
LD  R,A
JP  (HL)           ; display current line

blow              INC  C
                  INC  C            ; undo point next udgline xpos
                  INC  L            ; point to next udgdata

; 16 bytes to add an extra column in the display
line              LD  A,(BC)        ; get current x
                  LD  E,A          ; point to x on line
LDI                ; copy udg to xpos, dec next x
LD  A,(BC)         ; get next x
LDI                ; copy second part 16 bits udg
LD  E,A            ; point to x next udg
LD  A,(HL)         ; get udg
LD  (DE),A         ; write udg

EXX
XOR  A             ; point to start of data
DEC  C             ; decrease linecounter
LD  R,A
JP  (HL)           ; display current line

wait              LD  HL,frames      ; delay loop
                  LD  A,(HL)        ; used for display
wfr               SUB  B            ; and wait after death
                  CP   (HL)         ; or next level
JR   NZ,wfr
RET

rnd               PUSH HL           ; some randomness
rseed             LD  DE,0          ; in start of each new
                  LD  HL,(frames)   ; heart as well in the
DEC  DE            ; movement of each heart
ADD  HL,DE
LD  A,H
AND  #1F
LD  H,A
LD  (rseed+1),HL
LD  A,(HL)
POP  HL
RET

eog               LD  HL,score-1     ; test hiscore
                  LD  DE,hi-1
LD  BC,7
same              INC  HL
                  INC  DE
DEC  C
JR   Z,start      ; same score as hi
LD  A,(DE)
CP   (HL)
JR   Z,same        ; still same score
CALL C,#19F9       ; set new high with LDIR in ROM

start             LD  A,(lastk)      ; game over, wait for

```

```

SUB  %10111111      ; newline
JR   NZ,start

LD   SP,#4400        ; clear calls from stack

reset  INC  A
LD   (clrfire+1),A   ; allow fire from the start
LD   (steps+1),A

LD   HL,score        ; reset score
LD   B,6
clschr LD   (HL),28
INC  HL
DJNZ clschr

LD   A,34             ; "5" lives per game
LD   (lives),A        ; set "6"

LD   HL,gameitem+2    ; start in part1
LD   (loop+1),HL

dead  LD   HL,lives    ; 1 live lost
DEC  (HL)
LD   A,(HL)
CP   28
JR   Z,eog            ; game over when "0"

LD   HL,#0A0B
LD   (init),HL        ; cupid startposition

CALL clheart          ; erase hearts

LD   B,50             ; wait for gamestart
CALL wait

loop  LD   HL,gameitem ; which part in play?
steps LD   A,0
DEC  A
LD   (steps+1),A
JR   NZ,playitem

LD   (round+1),A      ; reset hearts in round

CALL clheart          ; erase on next part
INC  HL
INC  HL
LD   A,L
CP   lowres*256/256
JR   NZ,setnext       ; back to part1 test
LD   L,gameitem*256/256
setnext LD   (loop+1),HL ; point to next part

playitem LD   A,(HL)
INC  HL
LD   H,(HL)
LD   L,A
CALL #44              ; CALL (HL)

LD   HL,init
LD   BC,#102B         ; 16 arrows, top out of screen
arrowsup INC  HL

```

```

CALL hittest          ; test if arrow hits heart
INC HL
LD E,(HL)             ; get next position
LD (HL),C             ; write previous
LD C,E
DJNZ arrowsup        ; move them all

CALL readmove         ; read keyboard or joystick

LD HL,init+1
BIT 0,B               ; read result is in B
JR NZ,right
BIT 1,B
JR Z,movedone
left DEC (HL)
JR NZ,movedone        ; move ok to left
right INC (HL)

movedone LD A,(HL)
CP 20
JR Z,left             ; move false to right

DEC HL
LD E,A
INC E                 ; new arrowposition

clrfire LD A,1
DEC A
JR NZ,nofire         ; no fire allowed

LD (HL),E             ; show arrow on cupid

BIT 7,B
JR Z,delay            ; no fire pressed

LD A,(HL)             ; get current x arrow
LD (init+2),A         ; write it on screen

LD A,10               ; set 10 delays to shoot

nofire LD (HL),#2B     ; erase arrow on cupid
LD (clrfire+1),A      ; set altered wait counter

delay LD B,4           ; display delay
CALL wait
JP loop

readmove LD A,%11110111 ; port 1-5
IN A,(254)
RRA
JR C,lrread
LD B,128              ; bit 7 = fire
lrread LD A,%11101111  ; port 6-0
IN A,(254)
CPL
AND 3                 ; bit 0 = right, bit 1 = left
ADD A,B
LD B,A
RET NZ                ; keyboard key pressed

zxexpand LD BC,%1110000000000111
LD A,#A0
OUT (C),A
JR zx2

```

```

zx2      IN    A,(C)                ; read joystick
        CPL                      ; udlrf...
        RRCA                      ; .udlrf..
        RRCA                      ; ..udlrf.
        RRCA                      ; ...udlrf
        RRCA                      ; f...udlr
        LD     B,A                ; same as keyboard
        RET

clheart  LD     B,32
        LD     DE,screen
        LD     A,#2B
cls       LD     (DE),A            ; clear all hearts
        DEC    DE
        DJNZ   cls
        RET

part1    LD     B,#0F              ; 15 hearts, 1 falls off
        LD     HL,screen
        CALL   rnd
        AND    15
        JR     NZ,new             ; test to set heart
        CALL   rnd
        AND    15                ; xpos 0-15
        ADD    A,3                ; xpos 3-18
        LD     (HL),A            ; set random x-pos heart

new      LD     C,(HL)            ; get top
        LD     (HL),#2B          ; erase top

mvloop   DEC    HL                ; move 1 line down
        DEC    HL
        CALL   rnd
        AND    3                  ; 0,1,2,3
        JR     Z,zero            ; 1,2,3
        DEC    A                  ; 0,1,2
        DEC    A                  ; -1,0,1

zero     BIT    5,C                ; test out of screen
        LD     E,(HL)            ; get next
        JR     NZ,outscreen      ; no move out of screen
        ADD    A,C                ; do dx
        DEC    A
        CP     19                ; test within boundaries
        INC    A                  ; undo dec a
        JR     NC,outscreen      ; no change by move out screen
        LD     C,A                ; change to C
outscreen LD    (HL),C            ; write "new" xpos
        INC    HL
        LD     C,(HL)
        DEC    HL
        CALL   hittest           ; test dropped heart hit arrow
        LD     C,E                ; transport old x from here
        DJNZ   mvloop            ; do all hearts

        LD     HL,init+1
        LD     A,(HL)            ; x cupid
        SUB    C                  ; x bottom heart
        INC    A
        CP     3                  ; -1,0,1 is hit
        JP     C,dead            ; bottom heart drops on cupid
        ; return still on stack
        RET

```

```

hittest    BIT    5,(HL)                ; out of screen
           RET    NZ                    ; so no check needed
           LD     A,(HL)                ; get x
           SUB    C                     ; check fire
           JR     Z,hit
           INC    A                     ; could be 1 less
           RET    NZ                    ; no hit

hit        INC    A                     ; A=1
           PUSH   DE                    ; save original e
           LD     (clrfire+1),A          ; next round fire again
           LD     A,(HL)                ; get original x
           LD     C,#2B
           LD     (HL),C                ; erase heart

           INC    HL
           PUSH   HL
           LD     (HL),C                ; erase arrow

           LD     DE,screen+2            ; always add 1
           EX     DE,HL
           OR     A                     ; reset carry
           SBC    HL,DE
           LD     E,A
           LD     A,L
           RLCA                          ; divide 2 gives ypos
           RLCA
           RLCA
           AND    #F0
           ADD    A,E
           LD     HL,tabp2-1            ; table of displaypoints
           PUSH   BC
           LD     B,32

fhit       INC    HL
           CP     (HL)
           JR     NZ,fhit2              ; find if in table
           LD     DE,(part2+1)          ; translate position
           SBC    HL,DE                 ; to according displaybit
           LD     A,L                   ; this part is used
           JR     NC,lfound              ; in bonuslevel only
           ADD    A,32

lfound     ADD    A,A
           AND    #F8
           ADD    A,#C6                 ; opcode SET n,(HL)
           LD     (setbhl+1),A
           LD     HL,round+1

setbhl     SET    0,(HL)                ; signal heart deleted
           DEFB   #3A                  ; for speed only

fhit2      DJNZ   fhit                  ; check full table
           POP    BC
           POP    DE
           DEFB   254                  ; skip push, already here

add25      PUSH   HL
           LD     D,25                  ; 25 points to score

dpoint     LD     HL,score+6
           DEFB   #3A                  ; hide ten

ten        LD     (HL),28
           DEC    HL
           INC    (HL)
           LD     A,(HL)
           CP     38
           JR     Z,ten
           DEC    D

```



```

JR    NZ,dpoint
POP   HL
DEC   HL
RET

part2  LD    HL,tabp2
      INC   HL                ; rotate 1 position
      LD    A,L
      CP    hr*256/256        ; test back to start
      JR    C,seth1
      SUB   32
      LD    L,A                ; back to start
seth1  LD    (part2+1),HL
round  LD    DE,#800           ; 8 hearts, 0 deleted
set8   LD    A,(HL)           ; get x from table
      AND   15
      LD    C,A
      LD    A,(HL)
      RLCA
      RLCA
      RLCA
      RLCA
      AND   15
      LD    B,A                ; get ypos
      LD    A,L
      ADD   A,4
      CP    hr*256/256
      JR    C,hlok
hlok   LD    L,A
      PUSH  HL
findy  LD    HL,screen
      DEC   HL
      DEC   HL
      DJNZ  findy             ; get y on screentable
      DEC   HL
      LD    A,(HL)            ; get arrow x
      INC   HL
      SUB   C                  ; test hit with x heart
      JR    Z,sete0
      INC   A
      JR    NZ,rotate         ; not hit
sete0  BIT   0,E
      JR    NZ,rotate         ; already shot
      SET   0,E               ; signal shot
      PUSH  HL
      PUSH  DE
      CALL  add25
      LD    (HL),#2E          ; erase arrow
      POP   DE
      POP   HL
rotate RRC    E
      JR    NC,setheart
setheart LD    C,#2E           ; signal erase heart
      LD    (HL),C            ; set x heart
      POP   HL
      DEC   D
      JR    NZ,set8           ; set 8 hearts
      LD    A,E
      LD    (round+1),A       ; set status of 8 hearts
      INC   A                  ; test all deleted
      RET   NZ
      LD    A,(steps+1)       ; remaining time
      LD    E,A

```

```

addbonus    CALL add25
            INC HL
            LD A,E                ; in the end A=1
            DEC E
            JR NZ,addbonus
            LD (steps+1),A        ; signal next part to play
            RET

gameitem    DEFW part1
            DEFW part2

; the low and hires screens
x           EQU 101
lowres      DEFB 118
score       DEFB 28,28,28,28,28,28,0
            DEFB "S"+x,"T"+x,"U"+x,"P"+x,"I"+x,"D"+x,128
            DEFB "C"+x,"U"+x,"P"+x,"I"+x,"D"+x,0
hi          DEFB 28,28,28,28,28,28,0
lives       DEFB 33
            DEFB 118

space       EQU #439F-$

            DEFS space

; init must be same high as screen
; screen must remain at end of game
init        LD IX,hr              ; Hires mode
;          LD SP,#4400            ; Stack is cleared in game, not needed here
            LD H,#3F              ; #3fxx
            LD D,#BF              ; #bfxx
            LDIR                  ; repair 48K bug

clrdata     LD HL,#4018            ; clear the displayline
            DEC L
            LD (HL),C
            JR NZ,clrdata

            LD HL,init             ; clear initarea by
            LD (HL),#2B            ; setting all items
            LD DE,init+1           ; out of screen by
            LD C,34                ; hiscore-LDIR to prevent
            JP start-3             ; crash on 1st hires-display

vars        DEFB 128
?
last        EQU $

```