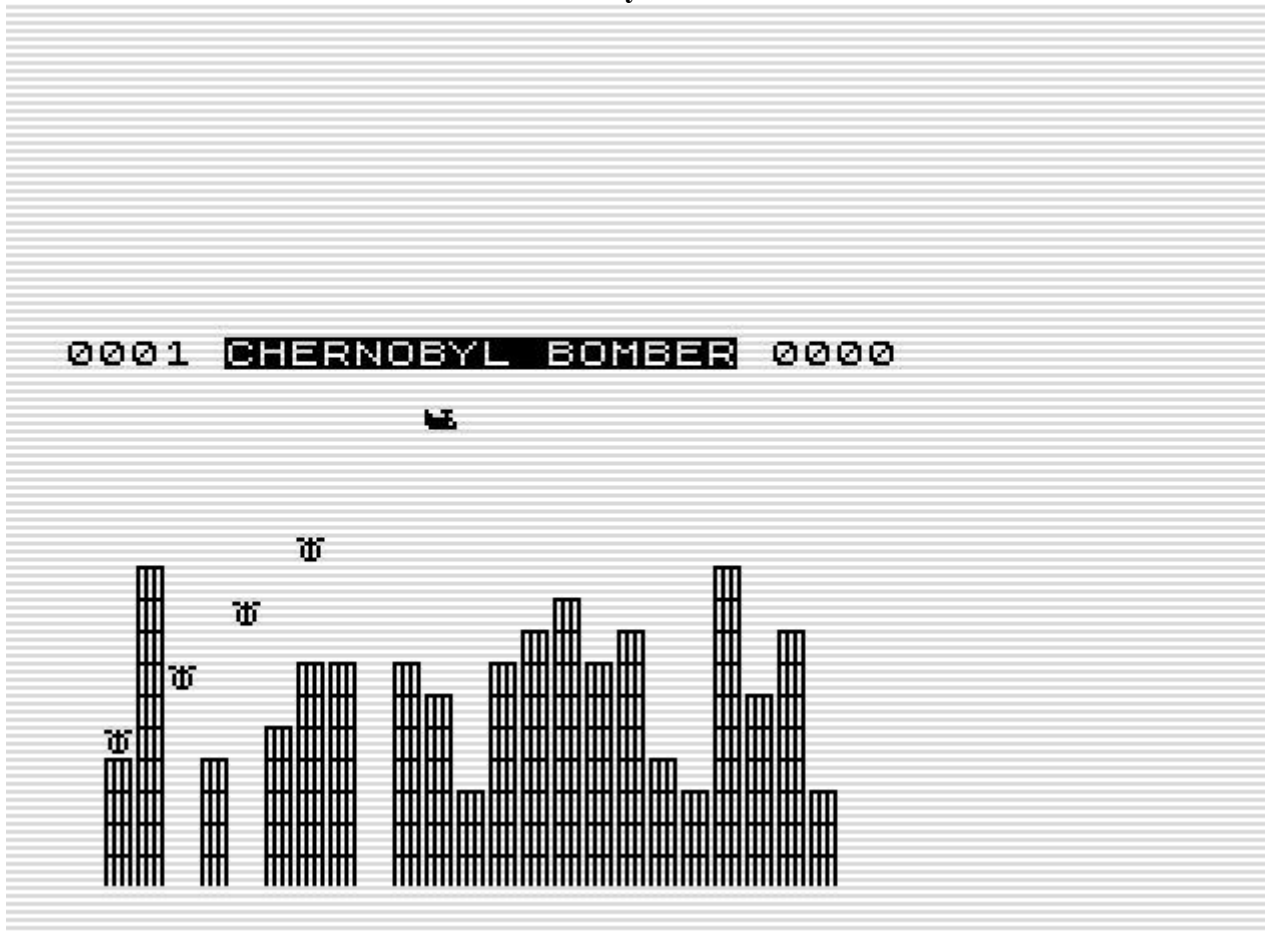


Chernobyl Bomber



Based on a request I thought of a way to code this game. 16 lines that will show a building in each column or not. 1 UDG can be overwritten per line. This is a plane or a bomb.

```
; Chernobyl Bomber
; Game 59 in 1K hires for the ZX81
```

```
? * TORNADO *
```

```
bomb      EQU    #401C
plane     EQU    nxtlin
bdelay    EQU    coprcc
```

```
ORG    #4009                ;#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                ; 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    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,0          ; 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

dbuf2     DEFB 170,170,170,170,170,170,170,170
        DEFB 170,170,170,170,170,170,170,170
        DEFB 170,170,170,170,170,170,170,170

hr        LD      HL,lowres+#8000 ; the lowres display
        LD      BC,#241      ; minimum needed
        LD      A,#1E
        LD      I,A
        LD      A,#FB
        CALL    #2B5

        EX      (SP),HL
        EX      (SP),HL
        LD      A,(HL)
        LD      A,(HL)

        LD      H,n2/256
        LD      A,H
        LD      I,A
        EXX                    ; program uses shadowregs
        PUSH    BC            ; must be saved too
        PUSH    DE
        PUSH    HL

        LD      (savesp+1),SP
        LD      SP,dispstack
        LD      C,16
        JR      cont

n2        LD      (DE),A
        EXX
        OR      A
        DJNZ    bdelay
        DEC     C
        JR      Z,savesp

cont      EXX
        POP     IX            ; get lbuf
        POP     BC            ; get udg

```

```

        POP    DE                ; get pos udg
        LD     L,n1*256/256
        XOR    A                ; top always space

        LD     R,A
        LD     (DE),A
        JP     (IX)            ; goto current lbuf

n1      LD     A,254
        LD     (DE),A          ; repair roof
        NOP
        INC    DE
        POP    DE                ; get new xpos in next databuf
        EXX
        LD     B,7
        EXX

bloop2  LD     A,(BC)
        LD     (DE),A
        INC    BC
        LD     L,n2*256/256
        LD     A,dbuf2*256/256-3 ; databuffer2
        LD     R,A
        LD     A,170            ; repair wall of building
        JP     (IX)            ; goto current lbuf

savesp  LD     SP,0            ; retrieve stack
        POP    HL
        POP    DE
        POP    BC
        EXX

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

start   LD     A,(lastk)        ; game over, wait for
        SUB    %10111111        ; newline
        JR     NZ,start

        LD     HL,played+3
        CALL   addl            ; increase played games

cls      LD     HL,lbuf
        LD     DE,lbuf+25
        LD     BC,15*25
        LDIR

setbuild
rseed   LD     C,24
        LD     HL,(frames)
        LD     DE,0
        ADD    HL,DE
        DEC    HL
        LD     A,H
        AND    #1F
        LD     H,A
        LD     (rseed+1),HL
        LD     A,(HL)
        AND    7
        ADD    A,3
        LD     B,A

yloop   CALL   lbuffld
        LD     (HL),B          ; show field

```

```

XOR  A
CALL mkdispfld
LD   (HL),A
INC  HL
INC  HL
LD   (HL),A
DJNZ yloop          ; set random sized tower
DEC  C
LD   A,C
DEC  A              ; no tower on final position
JR   NZ,setbuild

LD   B,#10          ; 16 lines

plyloop
plxloop
LD   C,24           ; 24 columns
LD   A,plane*256/256
CALL setx           ; show plane

BIT  6,(HL)         ; something visible?
JR   Z,start        ; then collision, game over

LD   (HL),B         ; show plane
EX   DE,HL          ; save plane position

LD   HL,frames      ; delay
LD   A,(HL)
SUB  7
CP   (HL)
JR   NZ,wfr

LD   A,H             ; #40
LD   (DE),A         ; hide plane
PUSH BC             ; save plane position

LD   A,C
DEC  A              ; no bomb on position right
JR   Z,drop

oldk
LD   A,(lastk)
CP   0
JR   Z,drop         ; not same key repeat

LD   (oldk+1),A     ; save altered (no) key
INC  A

LD   A,bomb*256/256
CALL NZ,setx        ; set new bomb on screen

drop
LD   B,1            ; drop all bombs 1 line
                        ; start at bottom, go up

dropline
CALL dispfld
LD   A,(HL)
SUB  bomb*256/256    ; has this line a bomb?
JR   NZ,nextbomb

PUSH BC             ; save current line

LD   (HL),A         ; erase bomb
INC  HL
INC  HL
LD   A,27
SUB  (HL)
LD   C,A            ; x calculated

```

```

XOR  A
LD   (HL),A
INC  HL
INC  HL
LD   (HL),A           ; current line erased

CALL lbuffld
LD   (HL),#40         ; hide position

DEC  B                ; go 1 line down
JR   Z,nextb
CALL lbuffld
LD   (HL),B           ; show next field

LD   A,bomb*256/256   ; show bomb 1 line lower
CALL setx

nextb  POP  BC         ; get original line back

nextbomb INC  B        ; go 1 line up
LD   A,B
CP   17
JR   NZ,dropline

POP  BC               ; get xy plane

DEC  C                ; move plane right
JR   NZ,plxloop
DJNZ plyloop         ; go 1 line down

LD   HL,succes+3      ; here succesfull landing
CALL add1

JP   start            ; perfect landing

setx   CALL mkdispfld
LD   A,27
SUB  C
LD   (HL),A
ADD  A,dbuf2*256/256-3
INC  HL
INC  HL
LD   (HL),A

lbuffld LD  HL,#4400+25
PUSH BC
LD   DE,65535-24      ; -25
lbuf1  ADD  HL,DE
DJNZ lbuf1
SBC  HL,BC
POP  BC
RET

ten    LD   (HL),28
DEC  HL
add1   INC  (HL)
LD   A,(HL)
CP   38
JR   Z,ten
RET

dispfld PUSH AF
LD   A,B
LD   HL,lbuf+2

```

```

fdisp      LD    DE,65535-7
           ADD   HL,DE
           DEC   A
           JR    NZ,fdisp
           POP   AF
           RET

mkdispfld  CALL  dispfld
           LD    (HL),A
           INC   HL
           INC   HL
           RET

x           EQU   101

lowres     DEFB  118
played     DEFB  28,28,28,28,0

           DEFB  "C"+x,"H"+x,"E"+x,"R"+x,"N"+x,"O"+x,"B"+x
           DEFB  "Y"+x,"L"+x,128
           DEFB  "B"+x,"O"+x,"M"+x,"B"+x,"E"+x,"R"+x,0
succes     DEFB  28,28,28,28
           DEFB  118

space      EQU   #4400-528-$
           DEFS  space

lbufh      EQU   lbuf+#8000

dispstack  DEFW  lbufh,plane,#4000,#4000
           DEFW  lbufh+25,plane,#4000,#4000
           DEFW  lbufh+50,plane,#4000,#4000
           DEFW  lbufh+75,plane,#4000,#4000
           DEFW  lbufh+100,plane,#4000,#4000
           DEFW  lbufh+125,plane,#4000,#4000
           DEFW  lbufh+150,plane,#4000,#4000
           DEFW  lbufh+175,plane,#4000,#4000
           DEFW  lbufh+200,plane,#4000,#4000
           DEFW  lbufh+225,plane,#4000,#4000
           DEFW  lbufh+250,plane,#4000,#4000
           DEFW  lbufh+275,plane,#4000,#4000
           DEFW  lbufh+300,plane,#4000,#4000
           DEFW  lbufh+325,plane,#4000,#4000
           DEFW  lbufh+350,plane,#4000,#4000
           DEFW  lbufh+375,plane,#4000,#4000

lbuf       DEFW  #4040,#4040,#4040,#4040
           DEFW  #4040,#4040,#4040,#4040
           DEFW  #4040,#4040,#4040,#4040
           JP    (HL)

init
setroof    LD    HL,#4018+3
           LD    (HL),254
           DEC   L
           JR    NZ,setroof
           LD    SP,dispstack
           LD    DE,bomb
           LD    HL,bombc
           LD    C,7
           LDIR

           LD    HL,planec
           LD    DE,plane
           LD    C,7

```

```

        LDIR

        LD    HL,bdelayc
        LD    DE,bdelay
        LD    C,5
        LDIR

        LD    HL,start
        PUSH HL
        LD    DE,init
        LD    HL,lbuf
        LD    BC,15*25
        JP    #19F9

; after loading copy below over sysvar
; saves 19 bytes
bombc    DEFB 214,56,84,84,84,56,0
planec    DEFB 142,196,254,253,127,0,0
bdelayc    EXX
            LD    A,(HL)
            JP    bloop2

vars      DEFB 128
?
last      EQU    $

```