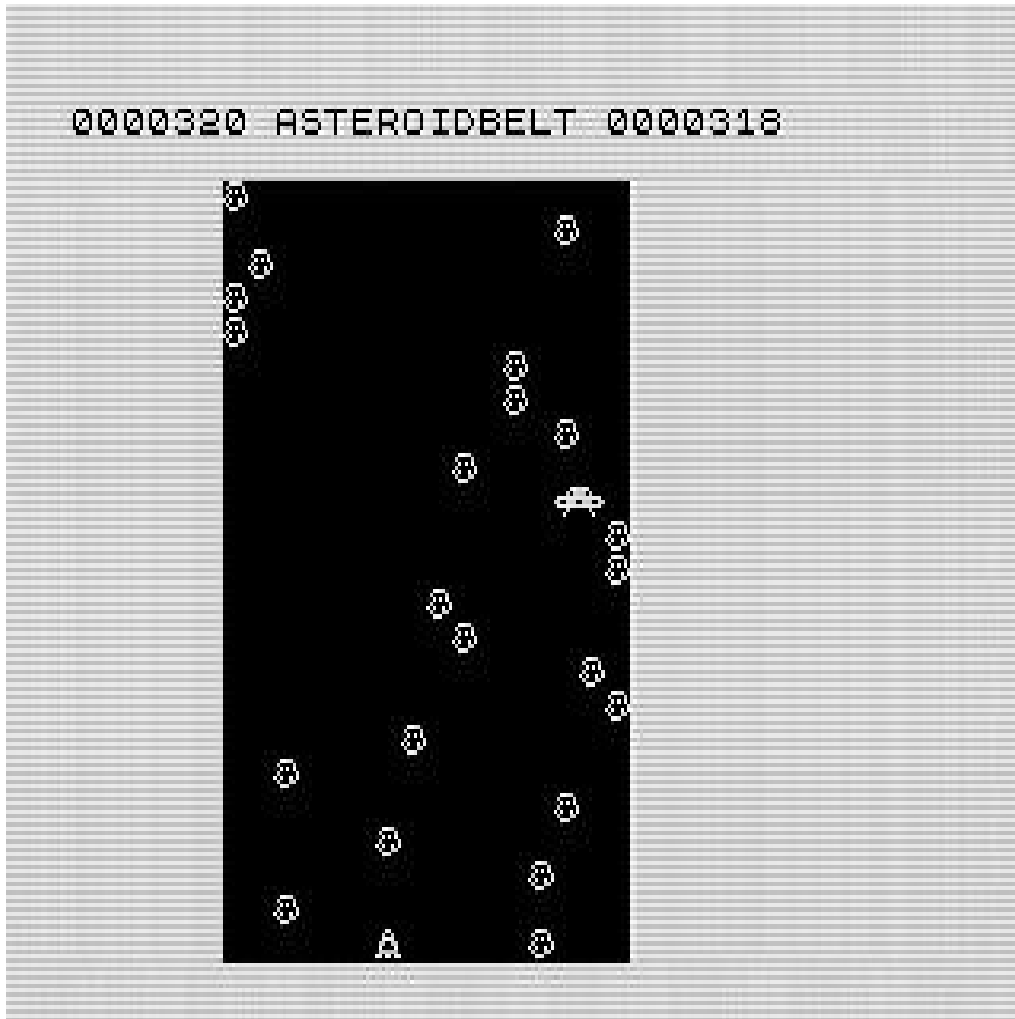


Asteroidbelt



Where previous games ends, the next idea starts. 2D MonsterMaze displayed a large UDG.

Question is: How many UDG can you load on a wide screen.

Problem is setting right pointers and erase old UDG. 2 UDG's can beset on random places when the screen is half size. This made this game possible. You have an asteroid and a ship on a line or the collided. Game is ready.

Later a UFO is added. Just meet them for extra points.

```
; Spacetrip, the classic spacetravel game in 1K hires
; Controls
; 1 = left
; 2 = right
; Newline = (re)start game

; additional controls ZXPAND
```

```
? * TORNADO *
```

```
ORG #4009
```

```
; #4009
```

DUMP 49161

JP init

d_file	DEFW dfile	
dfcc	DEFW dfile+1	
var	DEFW vars	
dest	DEFW 0	
eline	DEFW last	
chadd	DEFW last-1	
xptr	DEFW 0	
stkbot	DEFW last	
stkend	DEFW last	; memory above reused for data

berg	DEFB 0
mem	DEFW 0
	DEFB 0
	DEFB 2
	DEFW 1

lastk	DEFB 255,255,255	; used by ZX81
-------	------------------	----------------

margin	DEFB 55	
nxtlin	DEFW basic	
count	DEFB 0	
picnr	DEFB 0	
flagx	DEFB 0	; x
strlen	DEFW 0	

taddr	DEFW 3213
-------	-----------

seed	DEFW 0	
frames	DEFW 65535	; used by ZX81
coords	DEFB 0,0	
prcc	DEFB 188	
sposn	DEFB 33,24	
cdflag	DEFB 64	

star	EQU starudg*256/256
ship	EQU shipudg*256/256
ufo1	EQU ufo1udg*256/256
ufo2	EQU ufo2udg*256/256

starudg	EQU \$-1
	DEFB %00000000
	DEFB %00111000
	DEFB %01000110
	DEFB %10100101
	DEFB %01000001
	DEFB %00101010
	DEFB %01000010
	DEFB %00111100

```

shipudg    EQU    $-1
           DEFB    %00000000
           DEFB    %11011011
           DEFB    %01111110
           DEFB    %01000010
           DEFB    %01100110
           DEFB    %00111100
           DEFB    %00100100
           DEFB    %00011000

ufo1udg    EQU    $-1
           DEFB    0
           DEFB    %00010000
           DEFB    %00001000
           DEFB    %01111111
           DEFB    %11001110
           DEFB    %01111111
           DEFB    %00001011
           DEFB    %00000111

ufo2udg    EQU    $-1
           DEFB    0
           DEFB    %00001000
           DEFB    %00010000
           DEFB    %11111110
           DEFB    %01110011
           DEFB    %11111110
           DEFB    %11010000
           DEFB    %11100000

hr          LD      B,14                ; the screenroutine
h1          DJNZ   h1                ; start on screen

           LD      HL,lowres+#8000    ; start with lowres
           LD      BC,#301
           LD      A,#1E
           LD      I,A
           LD      A,#F5
           CALL    #2B5

hr2         LD      B,4                ; the screenroutine
h2         DJNZ   h2                ; start on screen
           INC     HL
           DEC     HL

           LD      IX,lbuf+#8000
           LD      D,#40
           LD      A,D
           LD      I,A
           LD      HL,screen-1

```

```

        LD      B,#18

bloop   INC     L
        LD      C,8
        DJNZ   nline

        CALL   #292                ; and back to program
        CALL   #220
        LD      IX,hr              ; set hr start
        JP      #2A4

cloop   DEC     HL
        DEC     L
        DEC     L
nline   LD      A,(HL)              ; fetch udg1
        ADD     A,C                ; add linenr for udg
        LD      E,A               ; point to right data
        LD      A,(DE)            ; get udg data
        INC     L
        LD      E,(HL)            ; fetch position for udg
s1      LD      A,(DE)            ; ld (de),a ; set udg1
        INC     L                ; goto next udg
        LD      A,(HL)            ; fetch udg2
        ADD     A,C
        LD      E,A
        LD      A,(DE)            ; data udg2
        INC     L
        LD      E,(HL)            ; next position
s2      LD      A,(DE)            ; ld (de),a ; set udg2

        XOR     A                ; lowbyte of dataline, at nn00
        DEC     C                ; preset flags for return
        JP      (IX)              ; linedisplay

lbuf    LD      R,A
        DEFW   #8080,#8080,#8080,#8080
        DEFW   #8080,#8080,#8080,#8080
        JP      Z,bloop           ; back to low, next char line
        JP      cloop            ; back to low, next udg line

dead    LD      B,15              ; flashloop
dflash  LD      (HL),D            ; set star/ship
        EXX
        LD      HL,frames
        LD      A,(HL)
        SUB     3
deadflash CP      (HL)            ; timedelay

```

```

        JR    NZ,deadflash
        EXX
        LD    A,D                ; swap udgs
        LD    D,E                ; ship to star
        LD    E,A                ; star to ship
        DJNZ  dflash             ; do flashloop

start    LD    A,(lastk)         ; entry from dead or init
        CP    %10111111
        JR    NZ,start          ; start game with NEWLINE
        LD    HL,score
        LD    B,7

resetsc  LD    (HL),28           ; score to zero
        INC   HL
        DJNZ  resetsc

        LD    HL,rstspd+1       ; undo speed up msg
        LD    (HL),B

        CALL  cls
        LD    A,7
        LD    (delay+1),A       ; slowest start
        LD    BC,#0909          ; both same
gl       PUSH  BC                ; save coordinates

rstspd   LD    A,0
        DEC   A
        JP    M,done            ; 255, ready
        LD    (rstspd+1),A      ; lower counter
        AND   1                 ; 0 / 1
        DEC   A                 ; 255 / 0
done      AND   118              ; 118 / 0
        LD    (spdtxt),A        ; NL or SPACE
                                   ; end with NL

        LD    HL,score-1
        LD    DE,hiscore-1
        LD    BC,8
findhi   DEC   C
        JR    Z,gameloop
        INC   HL
        INC   DE
        LD    A,(DE)
        CP    (HL)
        JR    Z,findhi          ; same sofar
        JR    NC,gameloop       ; no high
        LDIR                    ; sync hiscore

gameloop LD    DE,screen+91
        LD    HL,screen+87
        LD    C,88              ; B already 0
        LDDR                    ; scroll down

```

```

rndjp      INC    HL
setufo     JR      setstar
           LD      (HL),ufo1
           INC    HL
           CALL   rnd
           JR      Z,fielddok
           DEC    A
fielddok   LD      (HL),A
           INC    HL
           LD      (HL),ufo2
           INC    HL
           INC    A
           LD      (HL),A
           JR      zxpand

setstar    LD      (HL),star      ; set new asteroid
           INC    HL
           CALL   rnd
           LD      (HL),A
           INC    HL
           INC    HL
           LD      (HL),17
zxpand     LD      HL,screen+89

           LD      BC,%1110000000000111
           LD      A,#A0
           OUT    (C),A          ; signal ZXPAND
           PUSH   HL
           POP    HL            ; the delay for the ZXPAND
           IN     A,(C)         ; read joystick
           CPL
           AND    #F8
           POP    BC
           JR      Z,readkb      ; not used, check keyboard
                                   ; 5 4
           RRCA                ; 4>3>2>1>0
           RRCA                ; 3>2>1>0
           RRCA                ; 2>1>0
           RRCA                ; 1>0, 5 to 1 4 to 0
           JR      testa

readkb     LD      A,%11110111
           IN     A,(254)
testa      BIT    1,A
           JR      Z,noleft
           DEC    C              ; move left
noleft     BIT    0,A
           JR      Z,noright
           INC    C              ; move right
noright    LD      A,15
           CP     C
           JR      NC,ok        ; not out of screen

```

```

ok      LD    C,B                ; undo move
        LD    B,C                ; save new
        LD    DE,star*256+ship
        LD    A,B
        CP    (HL)
        JR    Z,hit1             ; hit on first field
        INC   HL
        INC   HL
hit1     CP    (HL)               ; test second
        DEC   HL                 ; point to udg
        JR    NZ,nohit
        LD    A,(HL)
        CP    star               ; test hit on asteroid
        JP    Z,dead
ufofnd  LD    A,(delay+1)
        INC   A
        LD    HL,score+4         ; hundreds of score
        AND   7
        JR    Z,sc1              ; already lowest speed
        LD    (delay+1),A
        JR    sc1

nohit   LD    (HL),E             ; set ship
        INC   HL
        LD    (HL),B

sc1      LD    HL,score+6         ; units of score
        INC   (HL)
        JR    sc2
rescnt  LD    (HL),28
        DEC   HL
sc2      INC   (HL)
        LD    A,(HL)
        CP    38
        JR    Z,rescnt
        LD    A,pos1000*256/256
        CP    L
        JR    C,ufoctest

speedup LD    HL,delay+1
        LD    A,(HL)
        DEC   A
        JR    Z,udgstar          ; you keep 1 frame delay
        LD    (HL),A
        LD    A,40                ; set counter speedmessage
        LD    (rstspd+1),A
;        JR    timedelay

ufoctest INC   A
        CP    L
        JR    NZ,udgstar          ; not on 100
        LD    A,(HL)
        SUB   31                  ; on x300

```

```

        JR    Z,udgset
        SUB   4                      ; on x700
        JR    Z,udgset
udgstar  LD    A,setstar-setufo
udgset   LD    (rndjp+1),A

timedelay LD   HL,frames
         LD    A,(HL)
delay    SUB   0                      ; speed is set during play
wfr      CP    (HL)
         JR    NZ,wfr

        JP    gl

rnd       LD    DE,0                  ; rnd-seed
         LD    A,(frames)             ; add frames
         INC   DE                     ; next seed
         ADD   A,E
         LD    E,A
         LD    A,0
         ADC   A,D
         AND   #1F                    ; within ROM
         LD    D,A
         LD    (rnd+1),DE             ; save seed
         LD    A,(DE)
         AND   15                     ; position on screen
         RET

n        EQU   27

lowres   DEFB  118
score    DEFB  28,28,28
pos1000  DEFB  28,28,28,28,0
         DEFB  "A"-n,"S"-n,"T"-n,"E"-n,"R"-n,"O"-n,"I"-n
         DEFB  "D"-n,"B"-n,"E"-n,"L"-n,"T"-n,0
hiscore  DEFB  28,28,28,28,28,28,28
         DEFB  118
spdtxt   DEFB  118                    ; can be altered to space
         DEFB  0,0,0,0,0,0,0,0,0,0
         DEFB  "S"-n,"P"-n,"E"-n,"E"-n,"D"-n,0,"U"-n,"P"-n
         DEFB  118

;space   EQU   #4200-$                ; screen must be in 256 bnds
;        DEFS  space

cls       LD    HL,screen-1
cls0      LD    B,96
cls1      INC   HL

```



```

LD      (HL),17          ; UDG print out of line
DJNZ   cls1             ; so not visible
RET

```

```

screen    EQU    $
; the screen holds a UDG and the position per line
; with initially code on the screen, the hires routine
; would kill the code due to incorrect pointers to dataline

```

```

init1     POP    HL          ; hires will crash on initcode
          NOP          ; but not when s1 and s2
          LD      (HL),A      ; have invalid value and are
          NOP          ; repaired to correct value
          POP    HL          ; after cls init. Return will
          NOP          ; clear init1, but init1 is
          LD      (HL),A      ; coded so it won't crash HR
          NOP          ; before it is cleared (NOPS)
          RET
          NOP
          DEFB    star        ; a fake asteroid for correct
          NOP                ; clear after NEWLINE

```

```

init      LD      IX,hr       ; directly to 'hires'
          LD      HL,#4016
ccline    DEC     L
          LD      (HL),0
          JR      NZ,ccline   ; displayline cleared
          LD      HL,#4000     ; 48K bug repair
          LD      DE,#C000
          LD      BC,#400
          LDIR
          LD      HL,start     ; the actual start
          PUSH    HL

          LD      HL,s2
          PUSH    HL
          LD      HL,s1
          PUSH    HL
          LD      A,18
          LD      HL,init1     ; but first 'repair' hires
          PUSH    HL
          LD      HL,init-1
          JP      cls0

```

```

basic     DEFB    0,1         ; only used to start program
          DEFW    0
          DEFB    249,212,28
          DEFB    126
          DEFB    143,0,18,0,0
dfile     EQU     $
          DEFB    118,0,0

```

```
vars      DEFB 128
last      EQU  $
```