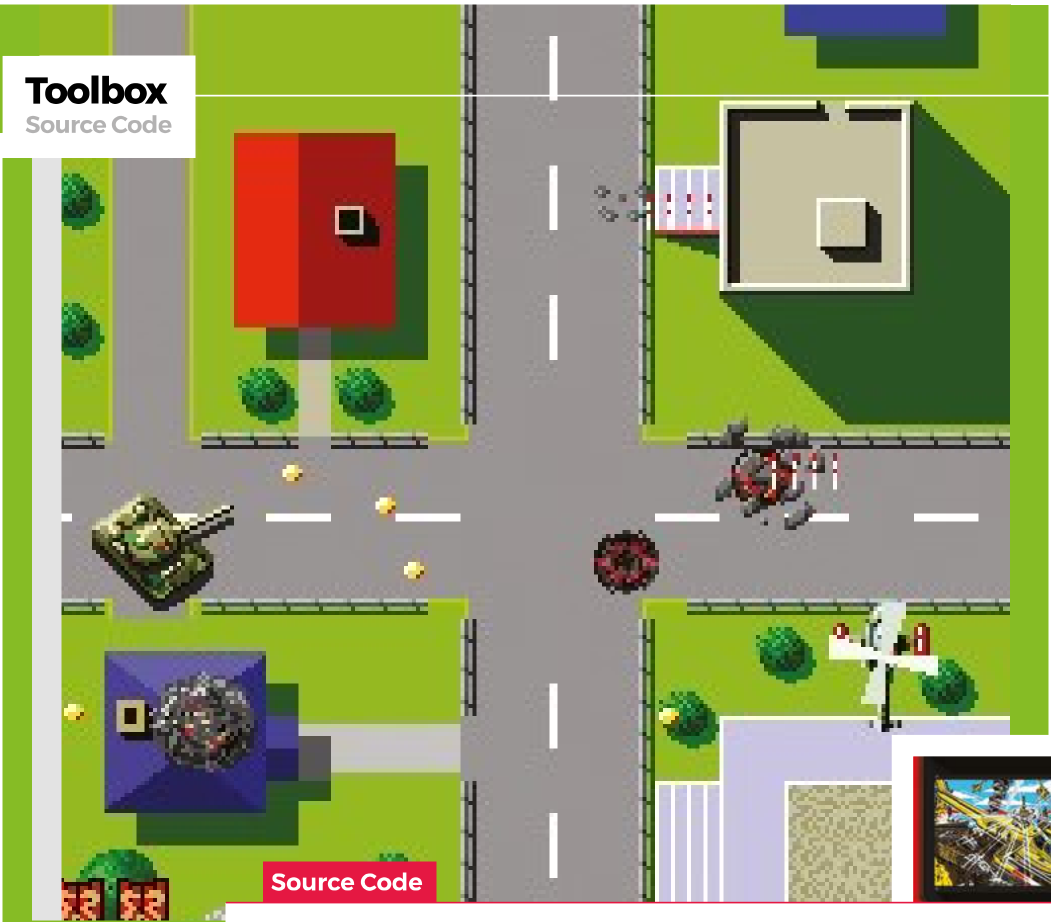


Fly over the military targets, firing missiles and dropping bombs.

Tiger-Heli was developed by Toaplan and published in Japan by Taito and by Romstar in North America.

Source Code

AUTHOR
MARK VANSTONE

Recreate Tiger-Heli's bomb mechanic

Code an explosive homage to Toaplan's classic blaster

Released in 1985, *Tiger-Heli* was one of the earliest games from Japanese developer Toaplan: a top-down shoot-'em-up that pitted a lone helicopter against relentless waves of enemy tanks and military installations. Toaplan would go on to refine and evolve the genre through the eighties and nineties with such titles as *Truxton* and *Fire Shark*, so *Tiger-Heli* served as a kind of blueprint for the studio's legendary blasters.

Tiger-Heli featured a powerful secondary weapon, too: as well as a regular shot, the game's attack helicopter could also drop a deadly bomb capable of destroying everything within its blast radius. The mechanic was one that first appeared as far back as Atari's *Defender* in 1981, but Toaplan quickly made it its own, with variations on the bomb becoming one of the signatures in the studio's later games.

For our *Tiger-Heli*-style Pygame Zero code, we'll concentrate on the unique bomb aspect, but first, we need to get the basic scrolling background and helicopter on the screen. In a game like this, we'd normally make the background out of tiles that can be used to create a varied but continuous scrolling image. For this example, though, we'll keep things simple and have one long image that we scroll down the screen and then display a copy above it. When the first image goes off the screen, we just reset the co-ordinates to display it above the second image copy. In this way, we can have an infinitely scrolling background.

The helicopter can be set up as an Actor with just two frames for the movement of the rotors. This should look like it's hovering above the ground, so we blit a shadow bitmap to the bottom right of the helicopter. We can set up keyboard events to move the Actor left, right, up, and down, making sure we don't allow it to go off the screen.

Now we can go ahead and set up the bombs. We can predefine a list of bomb Actors but only display them while the bombs are active. We'll trigger a bomb drop with the **SPACE** bar and set all the bombs to the co-ordinates of the helicopter. Then, frame by frame, we move each bomb outwards in different directions so that they spread out in a pattern. You could try adjusting the number of bombs or their pattern to see what effects can be achieved. When the bombs get to frame 30, we start changing the image so that we get a flashing, expanding circle for each bomb.

It's all very well having bombs to fire, but we could really do with something to drop them on, so let's make some tank Actors waiting on the ground for us to destroy. We can move them with the scrolling background so that they look like they're static on the ground. Then if one of our bombs has a collision detected with one of the tanks, we can set an animation going by cycling through a set of explosion frames, ending with the tank disappearing.

We can also add in some sound effects as the bombs are dropped, and explosion sounds if the tanks are hit. And with that, there you have it: the beginnings of a *Tiger-Heli*-style blaster.

Download the code from GitHub: **wfmag.cc/wfmag45**

Making bombs in Python

Here's Mark's code for a *Tiger Heli*-style shooter, complete with deadly bombs. To get it running on your system, you'll need to install Pygame Zero – full instructions can be found at **wfmag.cc/pgzero**.

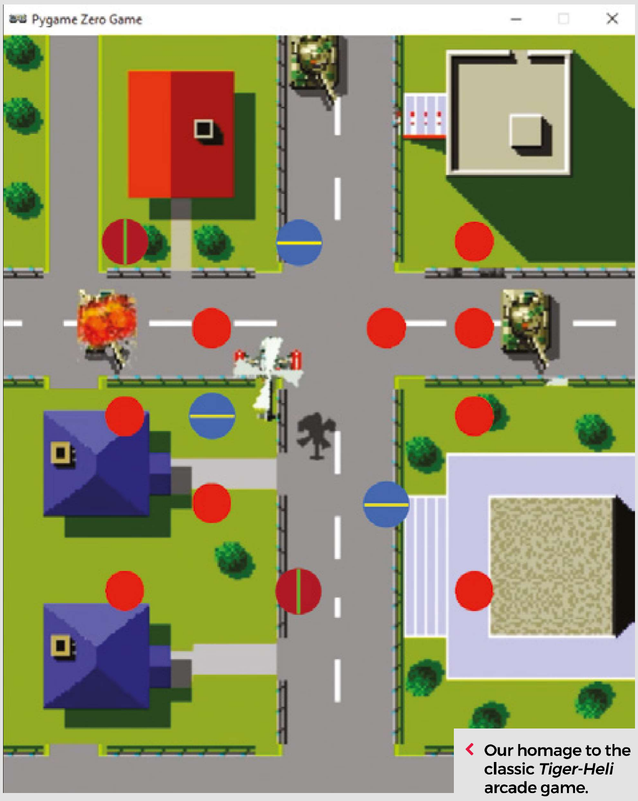
```
# Tiger-Heli

WIDTH = 600
HEIGHT = 800
backgroundY = count = 0
heli = Actor(‘heli1’, center=(300, 650))
bombActive = False
bombs = []
bombDirs = [(0,1),(1,1),(1,0),(0,0),(0,-1),(-1,-1),(-1,0),(-
1,1),(1,-1),(-0.5,0),(0.5,0.5),(-0.5,-0.5),(0.5,-0.5),(0,-
0.5),(-0.5,0.5),(-0.5,1),(1,-0.5),(-1,-0.5),(0.5,-1)]
for b in range(0, 18):
    bombs.append(Actor(‘bomb1’, center=(0,0)))
    bombs[b].frame = 0
tankLocations = [(500,-250),(100,-250),(300,-500)]
tanks = []
for t in range(0,3):
    tanks.append(Actor(‘tank0’, center=(tankLocations[t]
[0],tankLocations[t][1])))
    tanks[t].frame = 0

def draw():
    screen.blit(‘background’,(0,backgroundY))
    screen.blit(‘background’,(0,backgroundY-1400))
    screen.blit(“helishadow”+str(count%2 + 1),(heli.x+10,heli.
y+10))
    for t in range(0,3):
        if tanks[t].frame < 10:
            tanks[t].draw()
    if bombActive == True:
        for b in range(0, 18):
            bombs[b].draw()
    heli.draw()

def update():
    global backgroundY, count,bombActive
    backgroundY += 1
    if backgroundY > 1400: backgroundY = 0
    heli.image = “heli”+str(count%2 + 1)
    if keyboard.left and heli.x > 50 : heli.x -= 2
    if keyboard.right and heli.x < 550 : heli.x += 2
    if keyboard.up and heli.y > 50 : heli.y -= 2
    if keyboard.down and heli.y < 650 : heli.y += 2
    if keyboard.space : fireBomb()
    for t in range(0,3):
        tanks[t].y = (tankLocations[t][1] + backgroundY)
        if tanks[t].y > 850: tanks[t].frame = 0
        if tanks[t].frame > 0 and tanks[t].frame < 10 :
tanks[t].frame += 0.2
        tanks[t].image = “tank”+str(int(tanks[t].frame))
    if bombActive == True:
        for b in range(0, 18):
            bombs[b].y += 1
            bombs[b].x += bombDirs[b][0]*5
            bombs[b].y += bombDirs[b][1]*5
            bombs[b].frame += 1
            if bombs[b].frame > 30:
                bombs[b].image = “bomb”+str(bombs[b].frame-30)
                for t in range(0,3):
                    if bombs[b].collidepoint(tanks[t].pos) and
tanks[t].frame == 0:
                        tanks[t].frame = 1
                        sounds.explosion.play()
            if bombs[b].frame == 40:
                bombActive = False
    count += 1

def fireBomb():
    global bombActive
    if bombActive == False :
        bombActive = True
        sounds.launch.play()
        for b in range(0, 18):
            bombs[b].frame = 1
            bombs[b].pos = heli.pos
            bombs[b].image = “bomb1”
```



Our homage to the classic *Tiger-Heli* arcade game.