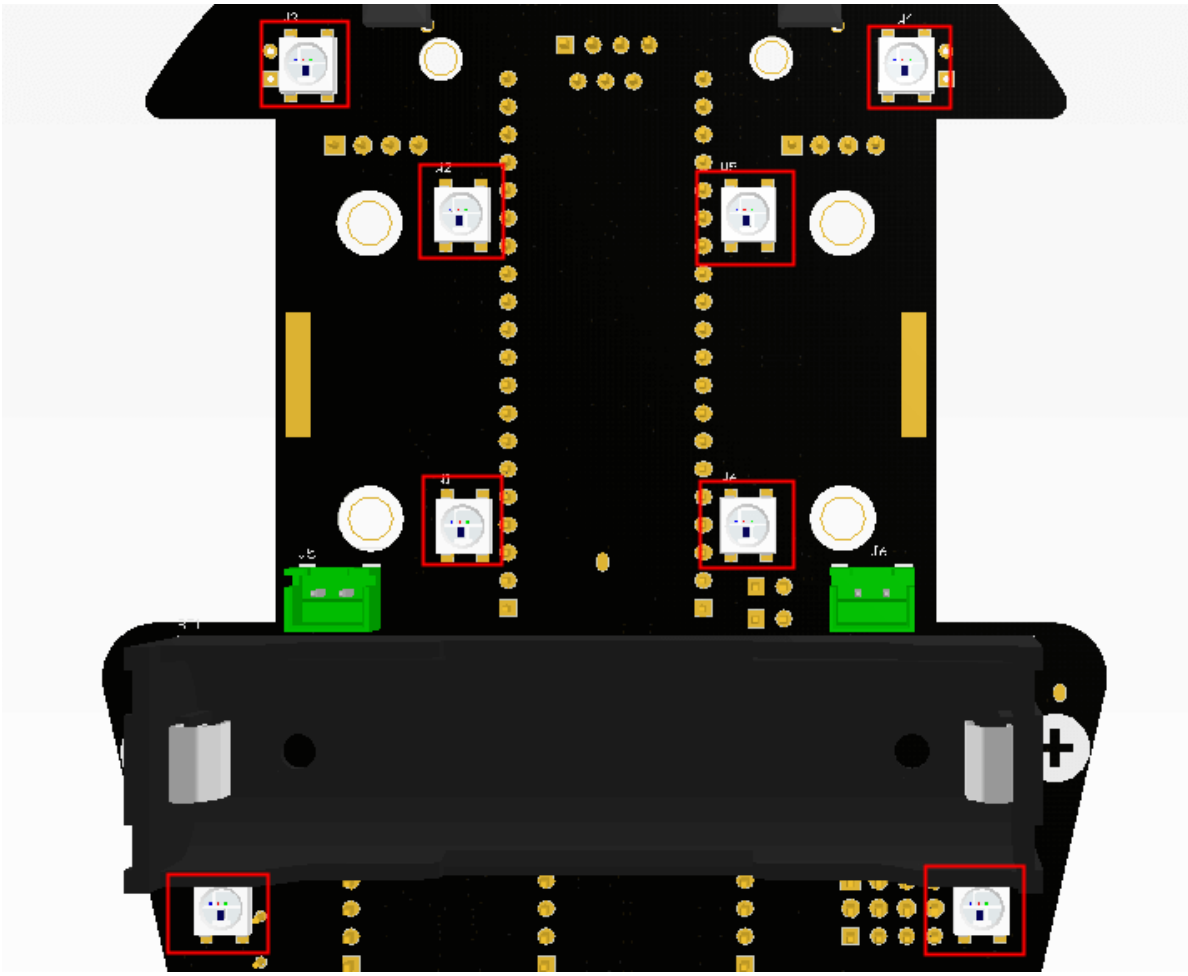
3.4 Breathing Light

I. Learning Objectives

1. Learn to combine the programmable RGB lights of the Raspberry Pi Pico 2/Pico mainboard and the car expansion board for experiments.
2. Understand the use of programmable RGB lights to achieve the breathing light effect.

II. Hardware Usage

This course uses the Pico 2/Pico mainboard and the car onboard programmable RGB lights



The car has 8 programmable RGB lights onboard, which can achieve colorful lighting effects. The 8 programmable lights have built-in ws2812 chips. Only one port is needed to control 8 lights at the same time through timing control. The timing control function is encapsulated in the library. We only need to call to set the color of the light.

3. Program Analysis

Code path: Code -> 1.Basic course -> 4. Breathing light.py

```
import time
from pico_car import ws2812b

num_leds = 8  # Number of NeoPixels
# Pin where NeoPixels are connected
pixels = ws2812b(num_leds, 0)
# Set all led off
```

```
pixels.fill(0,0,0)
pixels.show()
# Define variables
i = 0
brightness = 0
fadeAmount = 1
# Breathing
while True:
    for i in range(num_leds):
        pixels.set_pixel(i,0,brightness,brightness)
    pixels.show()
    brightness = brightness + fadeAmount
    if brightness <= 0 or brightness >= 200:
        fadeAmount = -fadeAmount
    time.sleep(0.005)
```

from pico_car import ws2812b

Because we only want to turn on the lights, we only use the ws2812b from pico_car.

import time

The "time" library. This library handles everything to do with time, from measuring it to inserting delays into the program. The unit is seconds.

pixels = ws2812b(num_leds, 0)

Initialize the RGB lights. We have 8 RGB lights, so here num_leds is set to 8.

pixels.fill(0,0,0)

Set all lights to 0,0,0, that is, turn off all lights. The parameters are (red, green, blue), and the color brightness is 0-255.

pixels.show()

Display the set lights.

pixels.set_pixel(i,0,brightness,brightness)

Set the color of each light through this function. The parameters are (light number, red, green, blue). The light number starts from 0 and the color brightness is 0-255. For example, the first bright red pixel.set_pixel(0,255,0,0).

brightness = 0 fadeAmount = 1

The breathing light effect is achieved by controlling the addition and subtraction of these two values.

IV. Experimental phenomenon

After the program is downloaded, we can see that the lights under the car cycle on and off in cyan with a breathing effect.