RGB Light Bar Driver

I. Learning Objectives

- Master the Jetson Nano series motherboards to control RGB light bars
- Control RGB light bar with CubeNano driver library
- Control RGB light bar through Jupyter lab

II. Preparation before experiment

Install the Jetson CUBE Nano chassis according to the assembly video tutorial, or refer to the "Jetson CUBE Nano Chassis_Hardware Wiring" tutorial document for installation, here we show the chassis expansion board and RGB light bar and fan hardware connections.

灯条/风扇接线图
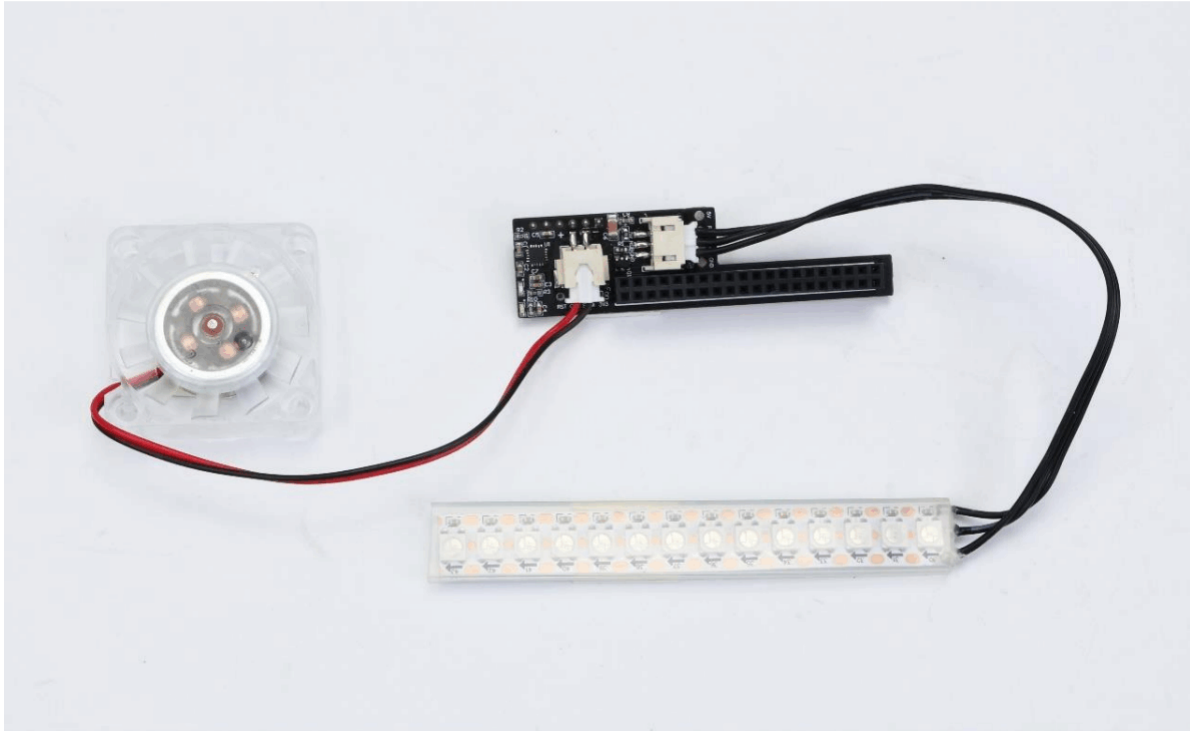
Chassis expansion board	3 Pin connector	2Pin connector
	RGB light bar	Fan

III. Using CubeNano driver library to control RGB light bar

1. Import the CubeNano driver library and create the objects.

The name of CubeNano driver library is CubeNanoLib, use CubeNanoLib to import the library in the programme.

```
from CubeNanoLib import CubeNano
```

The CubeNanoLib library functions needed to control the RGB light bar:

```
set_RGB_Effect(effect)    #设置RGB灯特效#Setting up RGB light effects
set_RGB_Speed(speed)      #设置RGB灯特效速度#Setting the speed of RGB light effects
set_RGB_Color(color)      #设置RGB灯特效颜色# Setting the colours of RGB light
effects
set_Single_Color(index, r, g, b)    #设置单个RGB灯颜色# Setting individual RGB
light colours
```

2. Setting up RGB light effects

set_RGB_Effect(effect): set RGB light effects

Effect value	RGB Light Effect
0	Turn off effects
1	Single colour breathing light
2	runaway light
3	rainbow light
4	dazzling light
5	flowing light
6	cycling breathing light

3. Setting the speed of RGB light effects

set_RGB_Speed(speed): set the speed of RGB light effects

Speed value	RGB light effects speed
1	low speed
2	medium speed
3	high speed

4. Setting the RGB light effect colours

Note: This function can only set the effect in single-colour breathing light and running light.

set_RGB_Color(colour): set RGB light effect colour

Color value	RGB light effect color
0	red
1	green
2	blue
3	yellow
4	purple
5	cyan
6	white

5. Setting individual RGB light colours

Note: All colours can be combined from different brightness levels of red, green and blue;

set_Single_Color(index, r, g, b): set a single RGB light colour

parametric	Individual RGB light colour
index	Serial number (0 - 13, 255)
r	Red (0 - 255)
g	Green (0 - 255)
b	Blue (0 - 255)

index parameter: 0-13 corresponds to the serial number of different lamp beads respectively; where index = 255, means set all RGB lamp colours.

r, g, b parameter: 0-255 means different colour brightness level.

6. Code Demonstration

- **control RGB light bar effects and functions (python interactive interface: each statement needs to be run separately)**

```
from CubeNanoLib import CubeNano
bot = CubeNano(i2c_bus=1)

bot.set_RGB_Effect(0)
bot.set_RGB_Effect(1)
bot.set_RGB_Effect(2)
bot.set_RGB_Effect(3)
bot.set_RGB_Effect(4)
bot.set_RGB_Effect(5)
bot.set_RGB_Effect(6)
```

```
bot.set_RGB_Speed(1)
bot.set_RGB_Speed(2)
bot.set_RGB_Speed(3)

bot.set_RGB_Effect(1)

bot.set_RGB_Color(0)
bot.set_RGB_Color(1)
bot.set_RGB_Color(2)
bot.set_RGB_Color(3)
bot.set_RGB_Color(4)
bot.set_RGB_Color(5)
bot.set_RGB_Color(6)

bot.set_Single_Color(5,0,0,255)
bot.set_Single_Color(255,255,255,255)
del bot
```


- **RGB Light Effects Test (RGB_Test.py)**
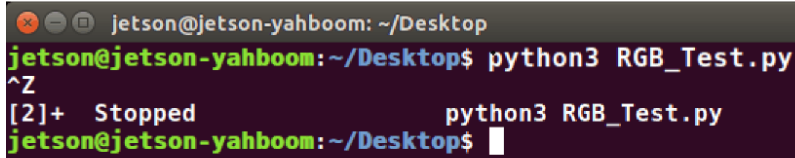
```
import time
from CubeNanoLib import CubeNano

if __name__ ==  '__main__':
    bot = CubeNano(i2c_bus=1)
    while True:
        bot.set_RGB_Effect(0) # 关闭特效5秒 Turn off the effect for 5 seconds
        time.sleep(5)
        bot.set_RGB_Effect(1) # 单色呼吸灯5秒 Monochromatic breathing lamp for 5
sec
        time.sleep(5)
        bot.set_RGB_Effect(2) # 跑马灯5秒 Marquee for 5 seconds
        time.sleep(5)
        bot.set_RGB_Effect(3) # 彩虹灯5秒 Rainbow lights for 5 seconds
        time.sleep(5)
```

```
        bot.set_RGB_Effect(4) # 炫彩灯5秒 Dazzle lights for 5 seconds
        time.sleep(5)
        bot.set_RGB_Effect(5) # 流水灯5秒 Running water lamp for 5 seconds
        time.sleep(5)
        bot.set_RGB_Effect(6) # 循环呼吸灯5秒 Circulate the lamp for 5 sec
        time.sleep(5)
```


Note: To terminate the run you can press Ctrl + Z

IV. Using Jupyter lab to control RGB light bar

In the Jupyter lab interface to create a new file and create the following code block, respectively, edit the following content (Fan_Test.ipynb):

1. Import the CubeNano driver library and create objects

```
# 导入CubeNano驱动库 Import the CubeNano driver library
from CubeNanoLib import CubeNano
# 创建对象 Create an object
bot = CubeNano(i2c_bus=1)
```

2. Setting up RGB light effects

```
bot.set_RGB_Effect(0) # 关闭特效 Turn off effects
```

```
bot.set_RGB_Effect(1) # 单色呼吸灯 Monochrome breathing light
```

```
bot.set_RGB_Effect(2) # 跑马灯 Marquee
```

```
bot.set_RGB_Effect(3) # 彩虹灯 Rainbow lights
```

```
bot.set_RGB_Effect(4) # 炫彩灯 Dazzling lights
```

```
bot.set_RGB_Effect(5) # 流水灯 Running water lamps
```

```
bot.set_RGB_Effect(6) # 循环呼吸灯 Circulation breathing lights
```

3. Setting the speed of RGB light effects

```
bot.set_RGB_Speed(1) # 低速 low speed
```

```
bot.set_RGB_Speed(2) # 中速 Medium speed
```

```
bot.set_RGB_Speed(3) # 高速 high speed
```

4. Setting the RGB light effect colours

```
bot.set_RGB_Effect(1) # 单色呼吸灯 Monochrome breathing light
```

```
bot.set_RGB_Color(0) # 红色 red
```

```
bot.set_RGB_Color(1) # 绿色 green
```

```
bot.set_RGB_Color(2) # 蓝色 blue
```

```
bot.set_RGB_Color(3) # 黄色 yellow
```

```
bot.set_RGB_Color(4) # 紫色 purple
```

```
bot.set_RGB_Color(5) # 青色 cyan
```

```
bot.set_RGB_Color(6) # 白色 white
```
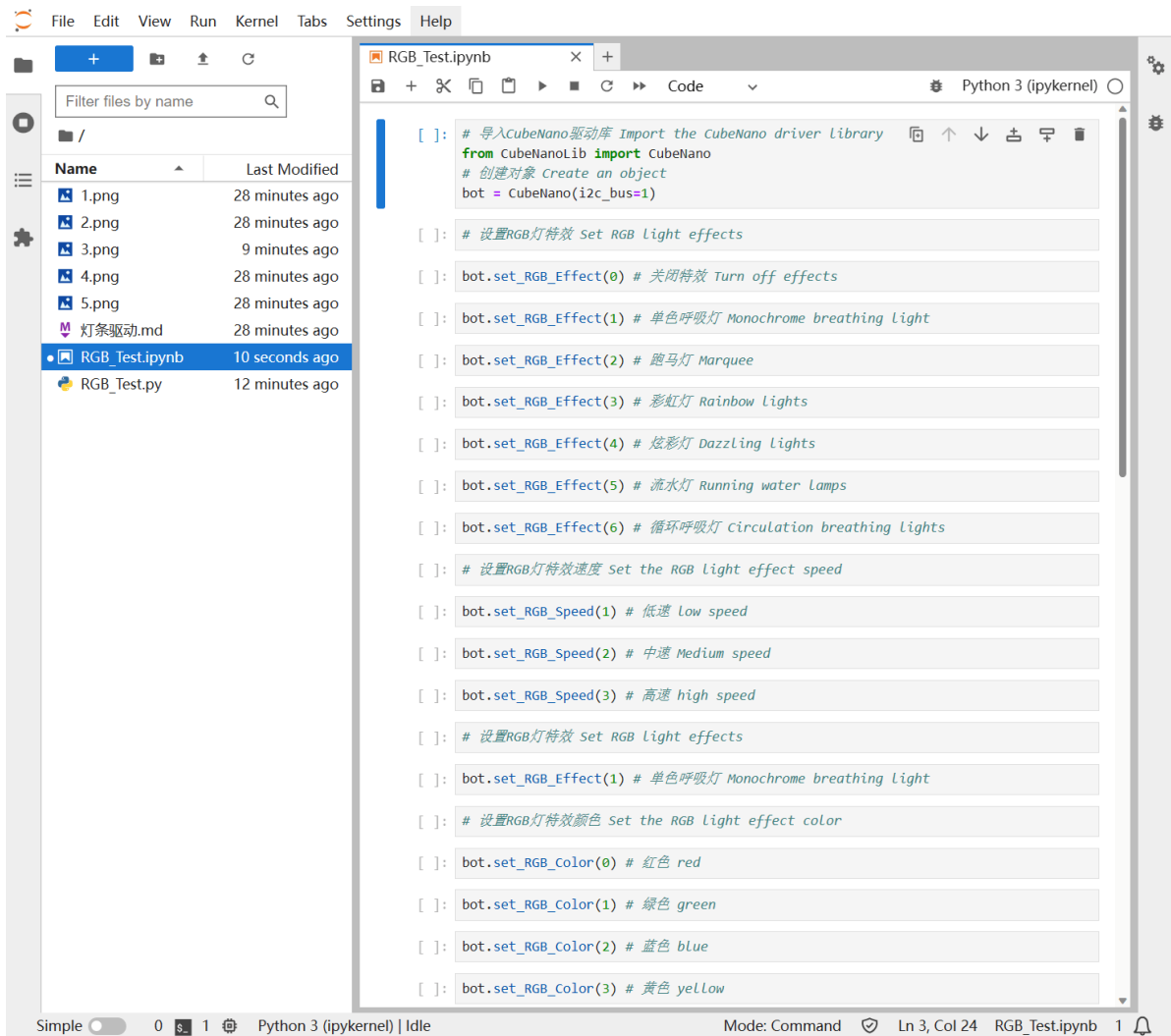
5. Setting individual RGB light colours

```
bot.set_Single_Color(5,255,0,0) # RGB灯条第6个灯珠显示红色
# RGB lights Article 6 lamp beads are displayed in red
```

```
bot.set_Single_Color(255,255,255,255) # 所有RGB灯珠显示白色
# All RGB beads are shown in white
```
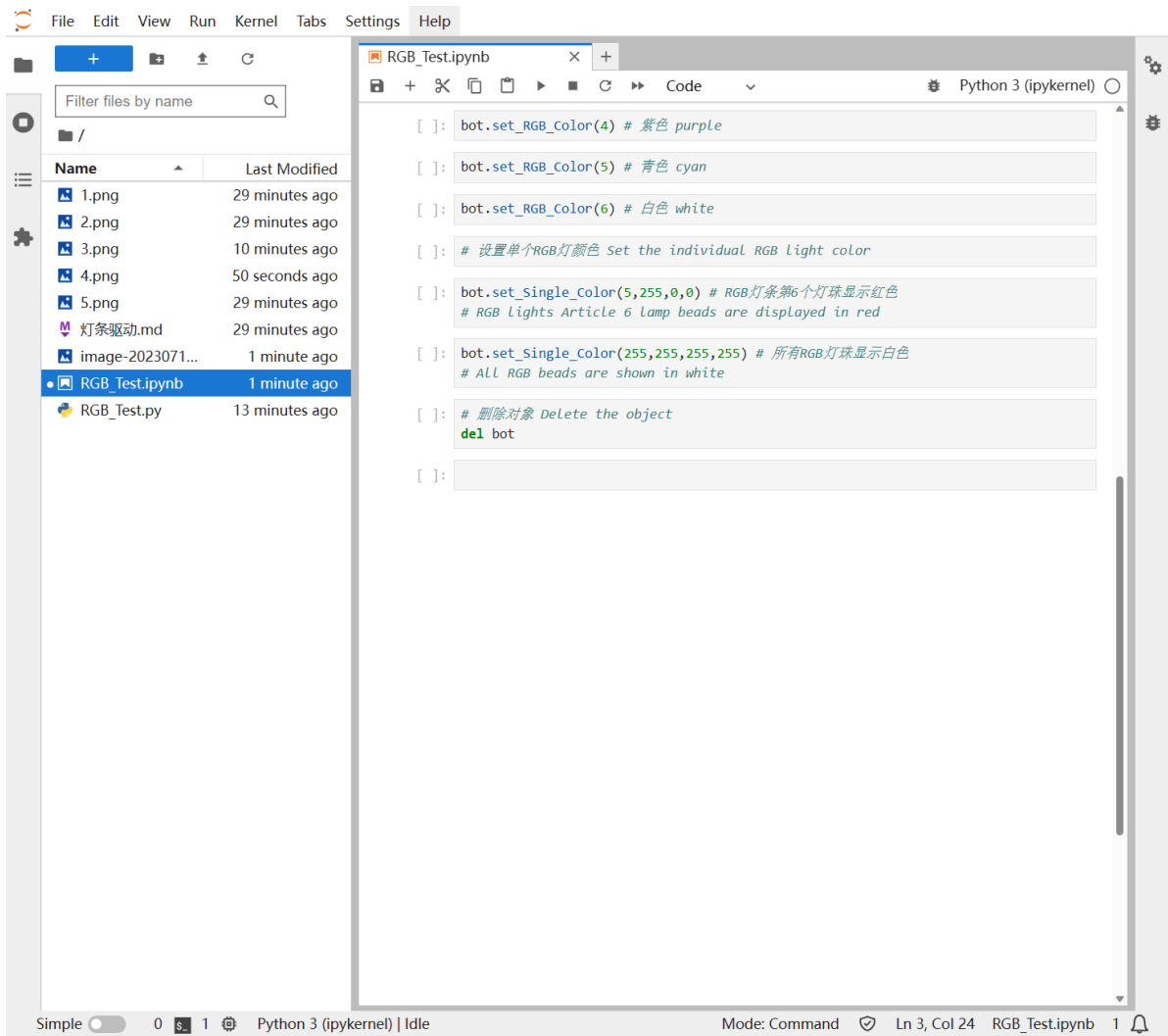
6. Deleting objects

```
# 删除对象 Delete the object
del bot
```

7. Code Demo

RGB_Test.ipynb

Code   Python 3 (ipykernel)

```python
# 导入CubeNano驱动库 Import the CubeNano driver library
from CubeNanoLib import CubeNano
# 创建对象 Create an object
bot = CubeNano(i2c_bus=1)
```

```python
# 设置RGB灯特效 Set RGB light effects
```

```python
bot.set_RGB_Effect(0) # 关闭特效 Turn off effects
```

```python
bot.set_RGB_Effect(1) # 单色呼吸灯 Monochrome breathing light
```

```python
bot.set_RGB_Effect(2) # 跑马灯 Marquee
```

```python
bot.set_RGB_Effect(3) # 彩虹灯 Rainbow lights
```

```python
bot.set_RGB_Effect(4) # 炫彩灯 Dazzling lights
```

```python
bot.set_RGB_Effect(5) # 流水灯 Running water lamps
```

```python
bot.set_RGB_Effect(6) # 循环呼吸灯 Circulation breathing lights
```

```python
# 设置RGB灯特效速度 Set the RGB light effect speed
```

```python
bot.set_RGB_Speed(1) # 低速 low speed
```

```python
bot.set_RGB_Speed(2) # 中速 Medium speed
```

```python
bot.set_RGB_Speed(3) # 高速 high speed
```

```python
# 设置RGB灯特效 Set RGB light effects
```

```python
bot.set_RGB_Effect(1) # 单色呼吸灯 Monochrome breathing light
```

```python
# 设置RGB灯特效颜色 Set the RGB light effect color
```

```python
bot.set_RGB_Color(0) # 红色 red
```

```python
bot.set_RGB_Color(1) # 绿色 green
```

```python
bot.set_RGB_Color(2) # 蓝色 blue
```
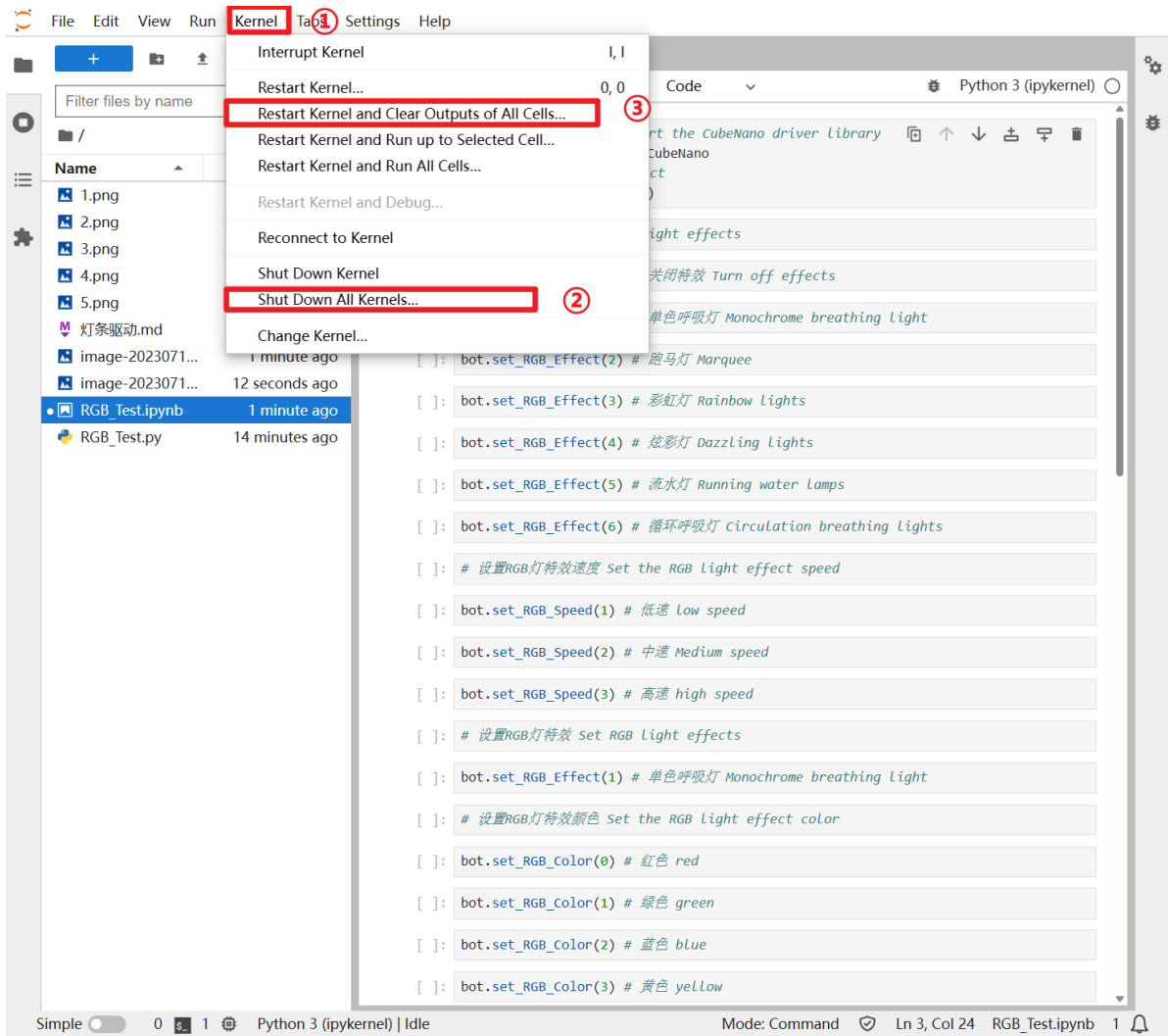
```python
bot.set_RGB_Color(3) # 黄色 yellow
```

8. Jupyter lab basic operation

If you find that the code runs abnormally, it is recommended that you follow the steps in the figure once, and then retest the code block.

V. Experimental phenomena

Call the functions in the CubeNano driver library to achieve the corresponding phenomenon, in which you need to set the RGB light effects before setting the colour and speed of the RGB light effects.

For example, after executing the function "Set single RGB light colour", directly using "Set RGB light effect colour" will not take effect, you need to "Set RGB light effect 1 or 5" first. effects 1 or 5" and then "set RGB light effect colour".

(The effect of using Jupyter lab to control the RGB light bar can be seen in the demo video under this folder.)