

3.9 Colour patrol

3.9 Colour patrol

1. Learning objectives
2. Preparation for the lesson
3. Programming Methods
4. Blocks
5. Code
6. Download code
7. Experimental phenomena
8. Notes:

1. Learning objectives

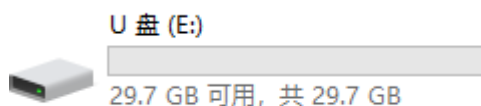
In this course. We will learn how to make the k210 vision module performs color recognition. After recognition, the car will move along the recognized color track.

2. Preparation for the lesson

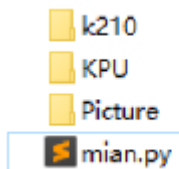
1. Remove the TF card from the k210 vision module and insert it into the card reader.



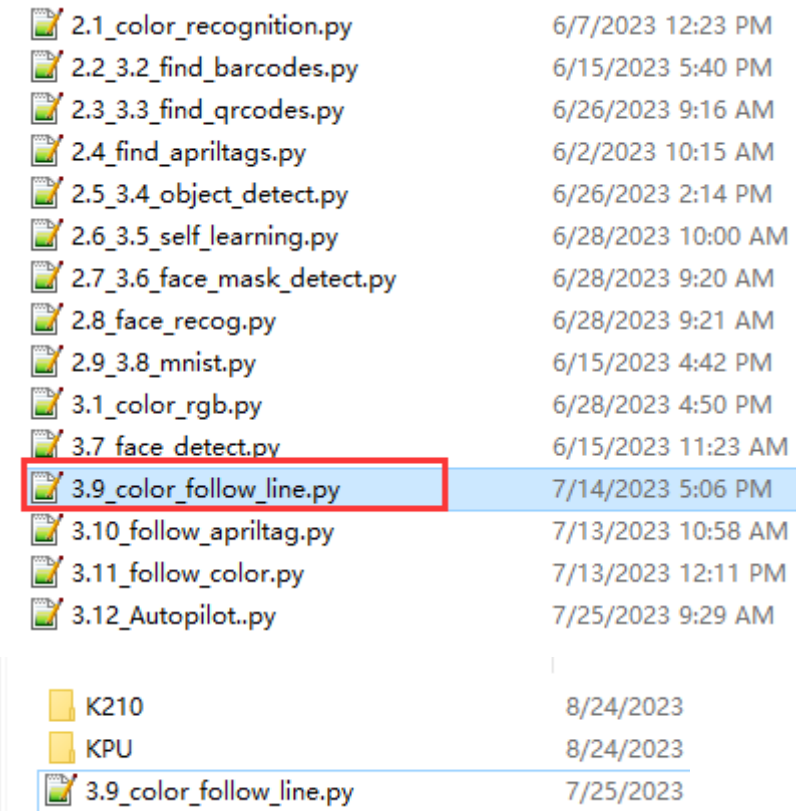
2. Plug the card reader into the computer, and wait for the computer to recognize the USB disk.



3. Then, enter the TF card. You will see following content.



4. Go to the k210 folder, find the **3.9_color_follow_line.py** file from the folder and copy it to the root directory.



5. Delete the original **main.py** file.

Then, re-name the **3.9_color_follow_line.py** file as the **main.py** file.

6. After re-name, pull out the card reader, remove the TF card and insert it back into the k210 vision module.

3. Programming Methods

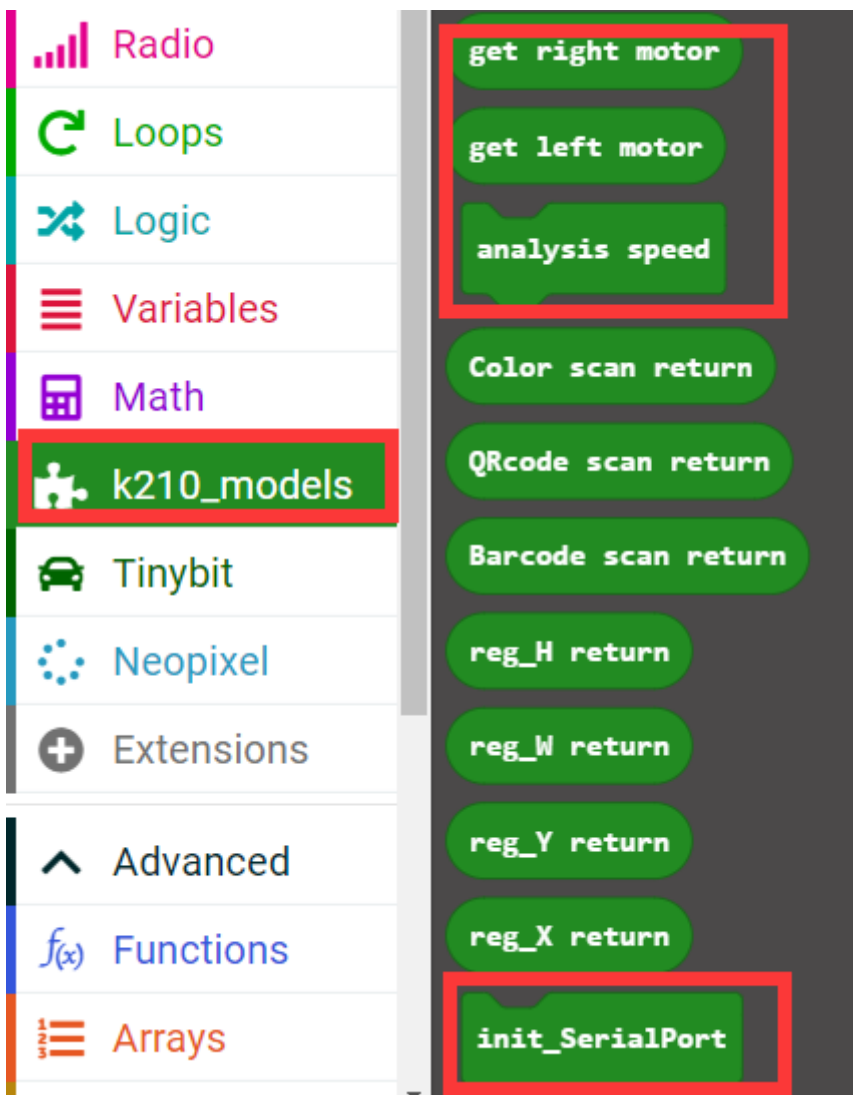
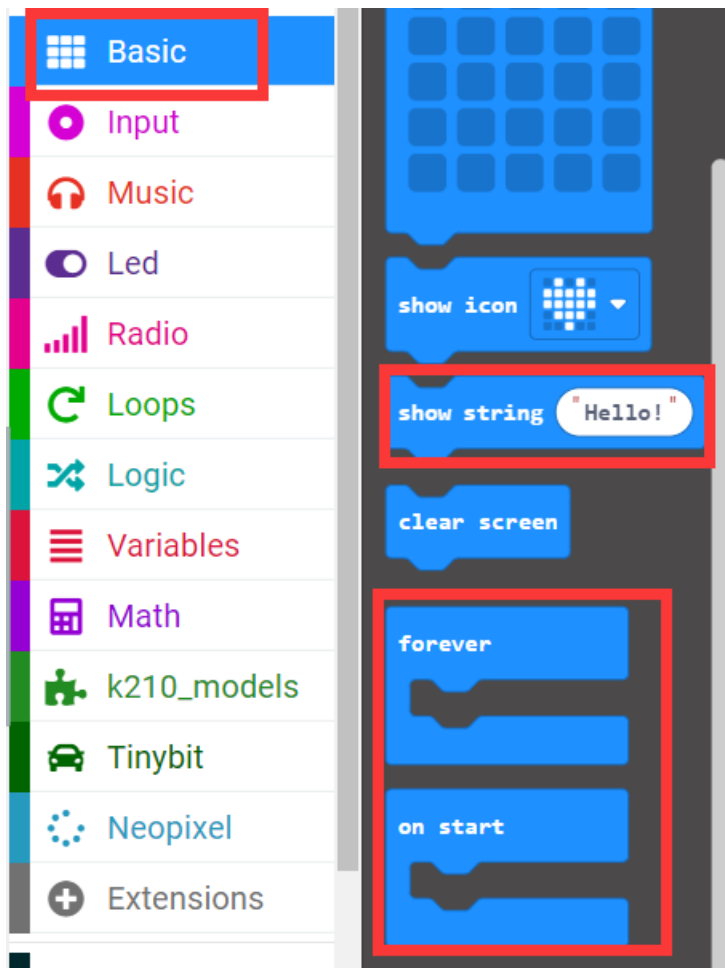
Online programming: first copy this URL <https://makecode.microbit>. and enter the online programming interface.

Click Extension, copy the package URL: <https://github.com/YahboomTechnology/K210-Module.git> to the input field, click Confirm to add package,

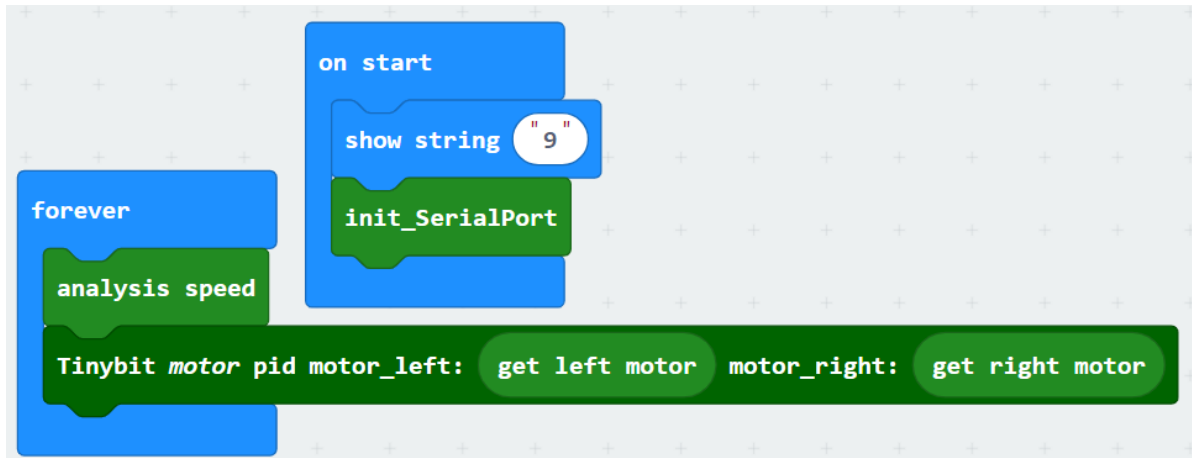
Click Extension again, copy the package URL: <https://github.com/YahboomTechnology/Tiny-bitLib> to the input field, click Confirm to add the package.

Finally you can use the K210 Vision Module package and Tinybit's building blocks.

4. Blocks



5.Code



6. Download code

Connect the Micro:bit board to the computer via microusb cable, the computer will pop up a USB stick.

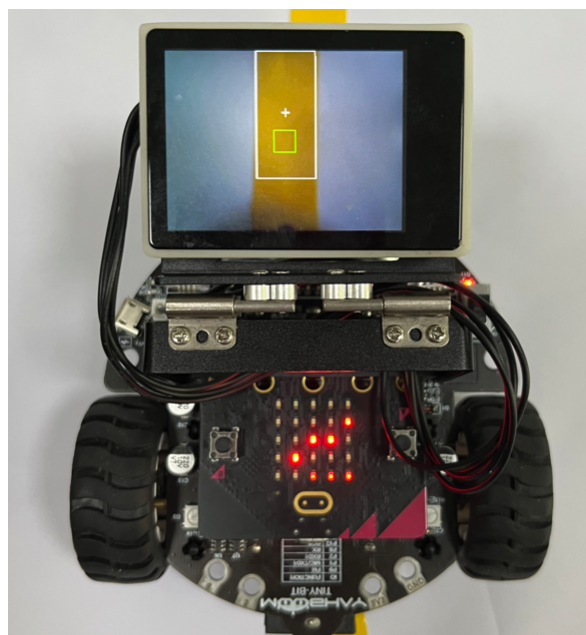
Then, select the **k210_color_line.hex** code and right click to send it to the Micro:bit U disk.

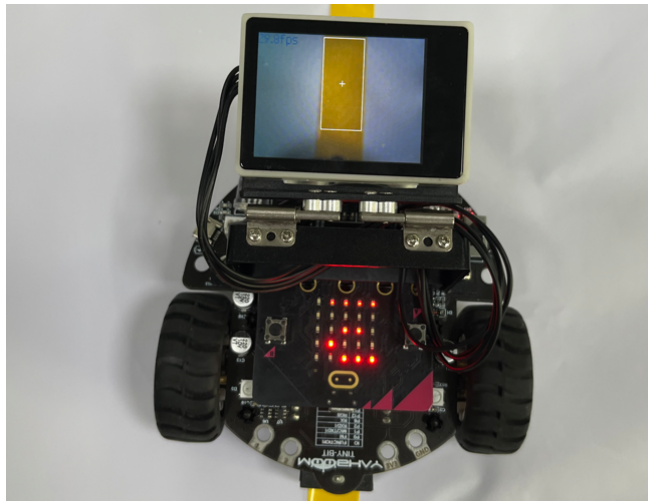
Wait until sending is complete and unplug the Micro:bit usb cable. Plug the Micro:bit board into the car.

7. Experimental phenomena

After the car starts, wait for the screen to display the camera image.

After the screen is displayed, point the camera at the color to be recognized, and the car will move along the color track after recognition.





8. Notes:

Due to the differences in the tracks and the lighting, when performing this experiment, if you are using the code but the cart is not cruising well, you will need to fine-tune the PID as well as the speed of the cart.

Only the boxed parts need to be modified.

```
1 import sensor, image, time, lcd
2
3 from modules import ybserial
4
5 speed_line = 27 #巡线速度
6
7 #FollowLinePID = (22, 0, 2)
8 FollowLinePID = (30, 0, 2)#15, 0, 2
9 SCALE = 100.0
10
11 ser = ybserial()
12
13 class PID(object):
14     def __init__(self, target, P, I, D):
15
16         self.Kp = P
17         self.Ki = I
18         self.Kd = D
19         self.setPoint = target
20         self.err = 0
21         self.err_next = 0
22         self.err_last = 0
23         self.last_result = 0
24
25
26     def __del__(self):
27         print("DEL PID")
28
29 # 重新设置目标值
30 def reset_target(self, target):
31     self.setPoint = target
32
```

