3.11 Color tracking

1. Learning Objectives

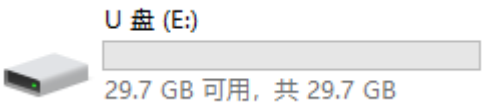
In this course. We will learn how to make Tiny:bit pro realize color tracking.

Put the color object that needs to be followed in front of the camera of the car for learning. After the learning is completed, when we move the colored object, the car will follow the colored object.

2. Preparation before class

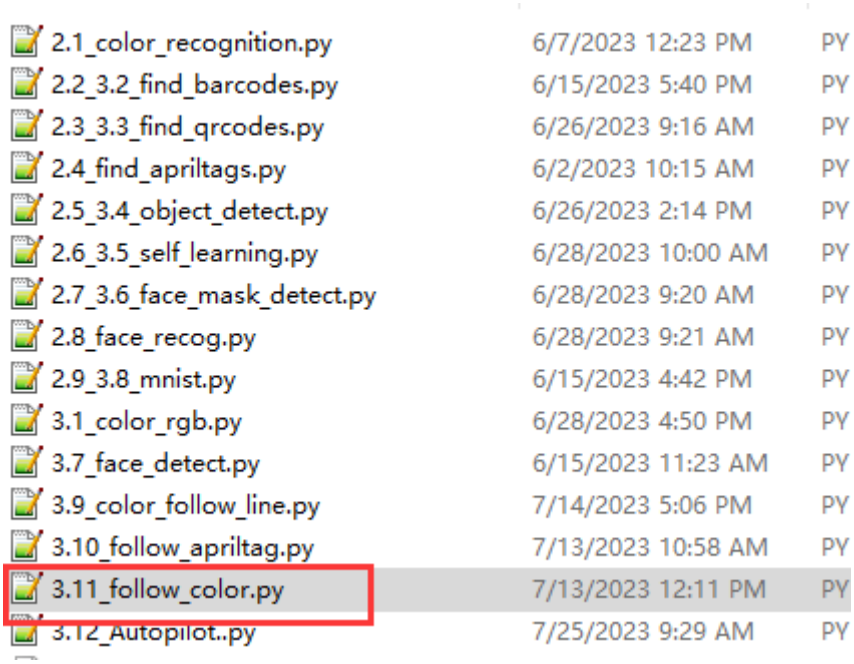
1. Take out the TF card from the k210 vision module and insert it into the card reader.

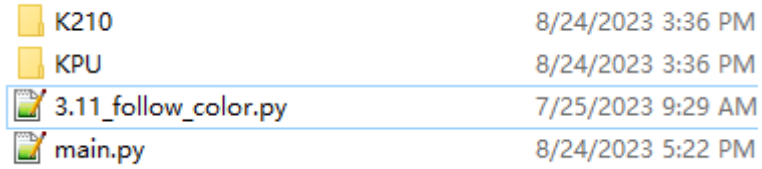
2. Plug the card reader into the computer, and wait for the computer to recognize the USB disk.



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

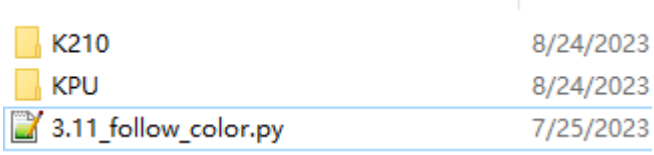
4. Go to the k210 folder and find the **3.11_follow_color.py** file from the folder and copy it to the root directory.





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

   Then, re-name the **3.11_follow_color.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

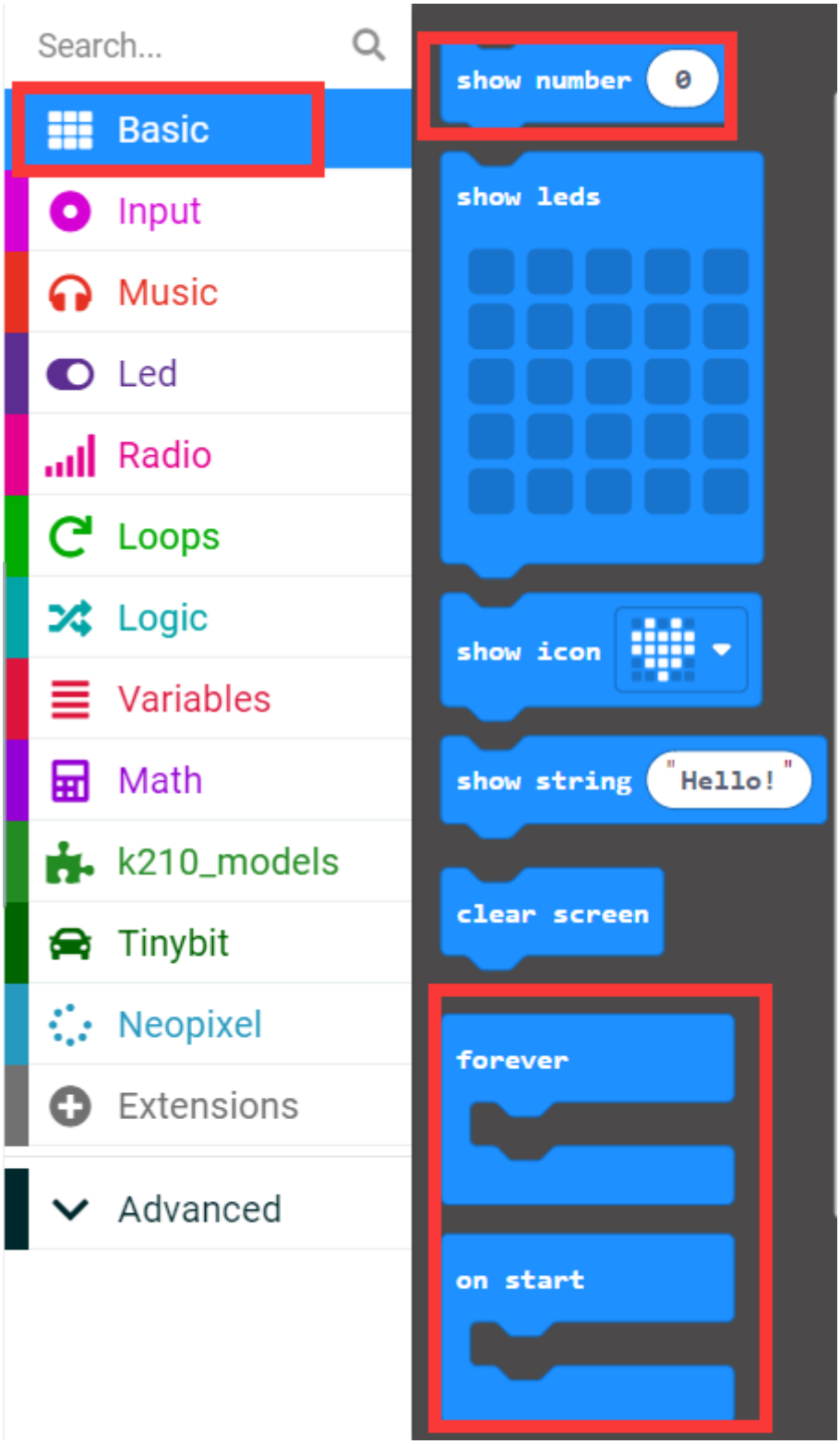
Online programming: first copy this URL https://makecode.microbit. and enter the online programming interface.

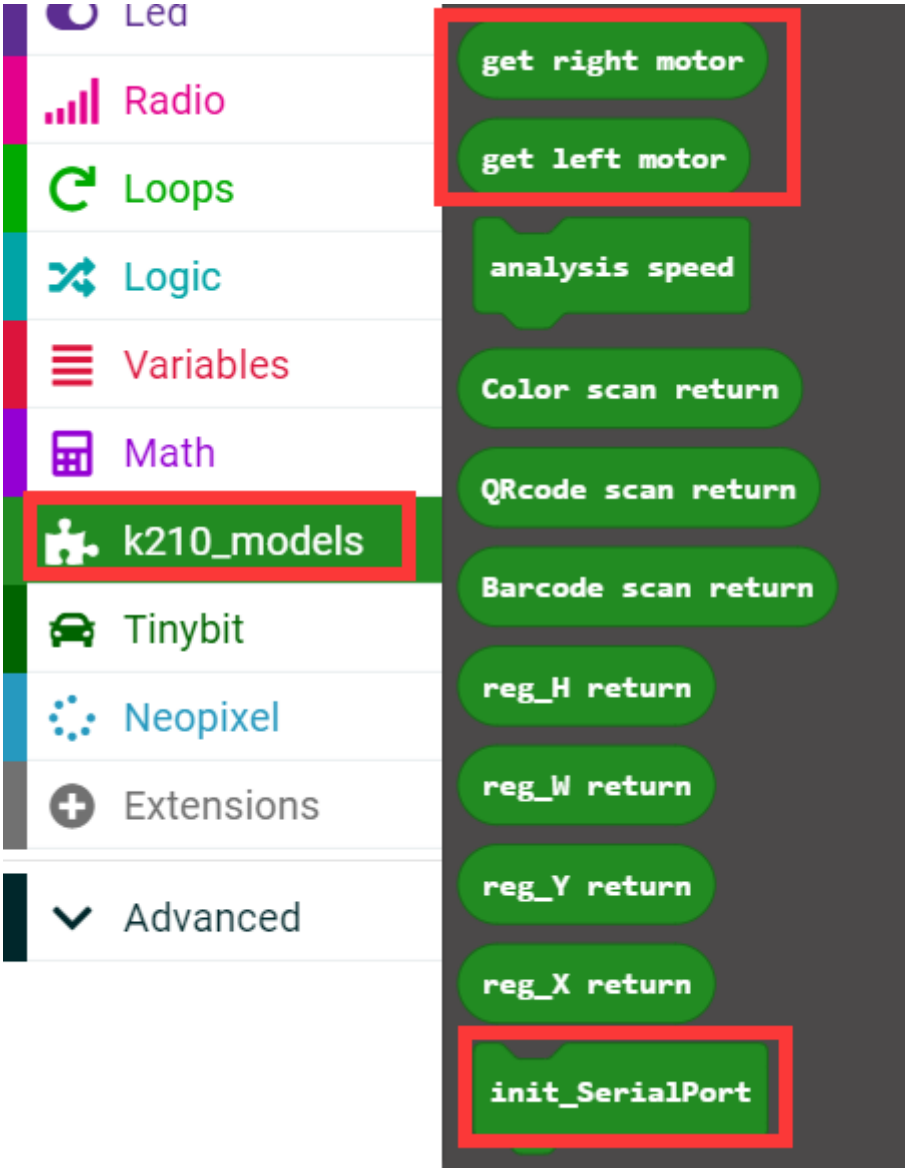
Click Extension, copy the package URL: https://github.com/YahboomTechnology/K210-Module.git to the input field, click Confirm to add package,

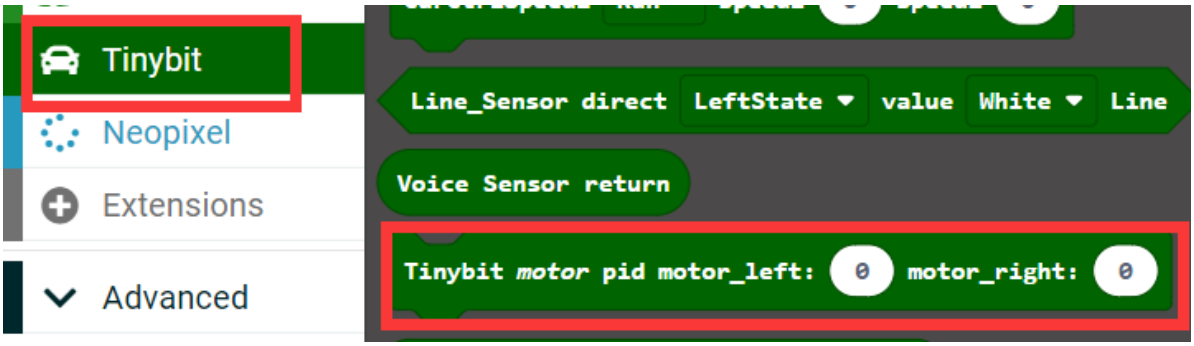
Click Extension again, copy the package URL: https://github.com/YahboomTechnology/Tiny-bitLib to the input field, click Confirm to add the package.

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

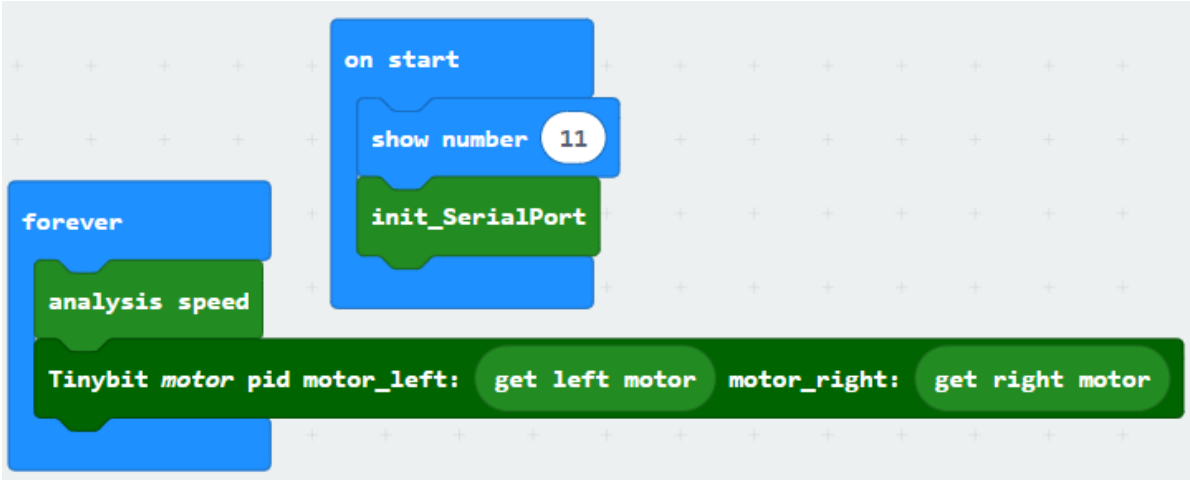
4. Blocks


Search...
Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
k210_models
Tinybit
Neopixel
Extensions
Advanced
show number 0
show leds
show icon
show string "Hello!"
clear screen
forever
on start


Led
Radio
Loops
Logic
Variables
Math
k210_models
Tinybit
Neopixel
Extensions
Advanced
get right motor
get left motor
analysis speed
Color scan return
QRcode scan return
Barcode scan return
reg_H return
reg_W return
reg_Y return
reg_X return
init_SerialPort


Tinybit
Neopixel
Extensions
Advanced
Line_Sensor direct LeftState value White Line
Voice Sensor return
Tinybit motor pid motor_left: 0 motor_right: 0

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_follow_color.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

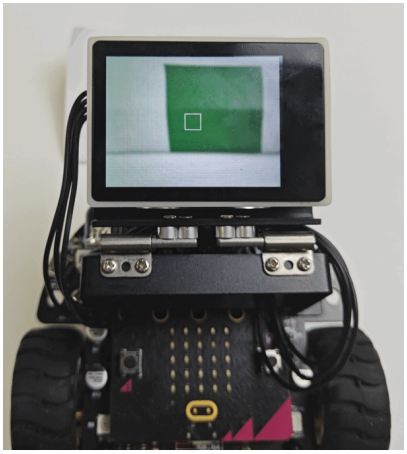
After the car starts, the dot matrix of the microbit scrolls to display the number 11. Wait for the screen to display the camera image.
According to the position of the white frame, point the camera at the colored block to be recognized, and a green frame will appear at this time, indicating that the color is being learned. Wait for the green frame to turn into a white frame, and the + symbol appears in the white frame, which means that the learning is completed.

We move the colored object to make sure it moves within the camera range.

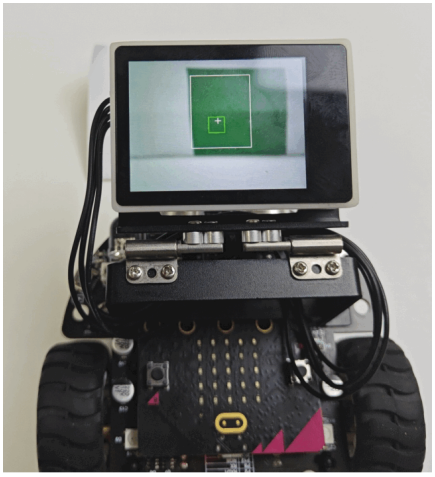
When the white frame at the top of the screen, the car will move forward;
When the white frame at the bottom of the screen, the car will move backwards;
When the white frame on the left side of the screen, the car will turn left;
When the white frame on the right side of the screen, the car will turn right.

Note: The speed of the moving object should not be too fast, because it is greatly affected by the light. When the white box disappears, or when multiple white boxes appear at the same time, the effect is not good, change a light uniform ring environment, and then re-boot, this can improve the effect.

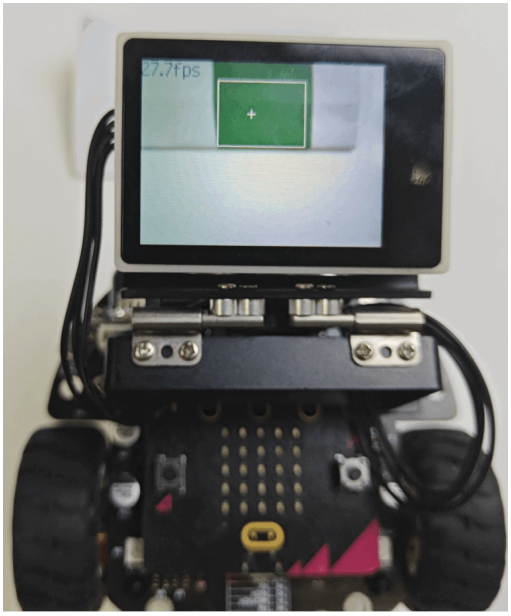
As shown below.



A green frame appears to indicate that the color is being learned

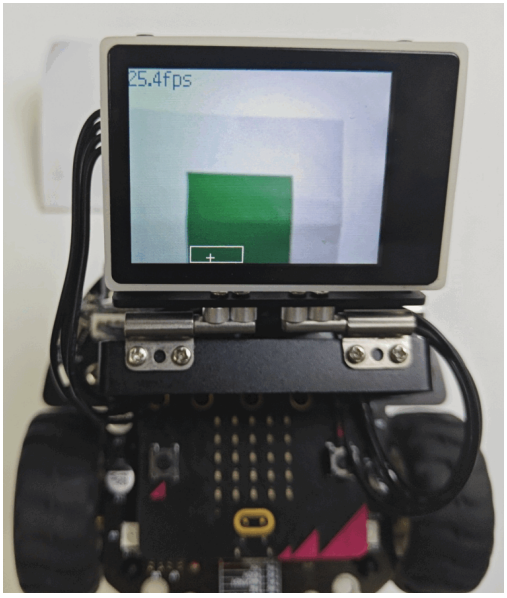


Wait for the green frame to turn into a white frame and for the + symbol to appear inside the white frame, which means is that learning is complete.

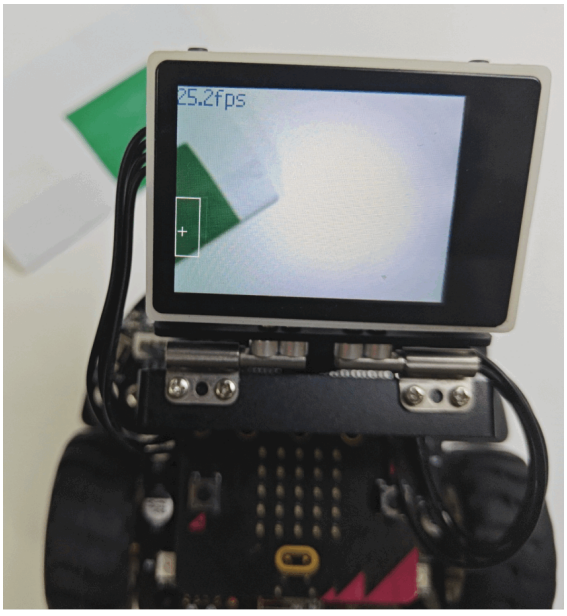


When the white frame is at the top of the screen, the car will move forward

When the white frame is at the bottom of the screen, the car will move backward



When the white frame is on the left side of the screen, the car will turn left



When the white frame is on the right side of the screen, the car will turn right