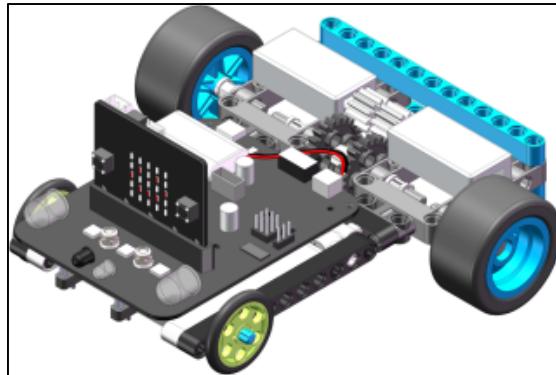


Python course Following car ---“Tracking”

! Note: This experiment must be done indoors to reduce the interference of sunlight on the infrared sensor.



1. Learning goals

After downloading the program, open the power switch of the Following car and put Following car on the tracking map, the car will walk along the black track.



2. Preparation before class

We needs to be ready:

Building Block Following car *1

Tracking map *1

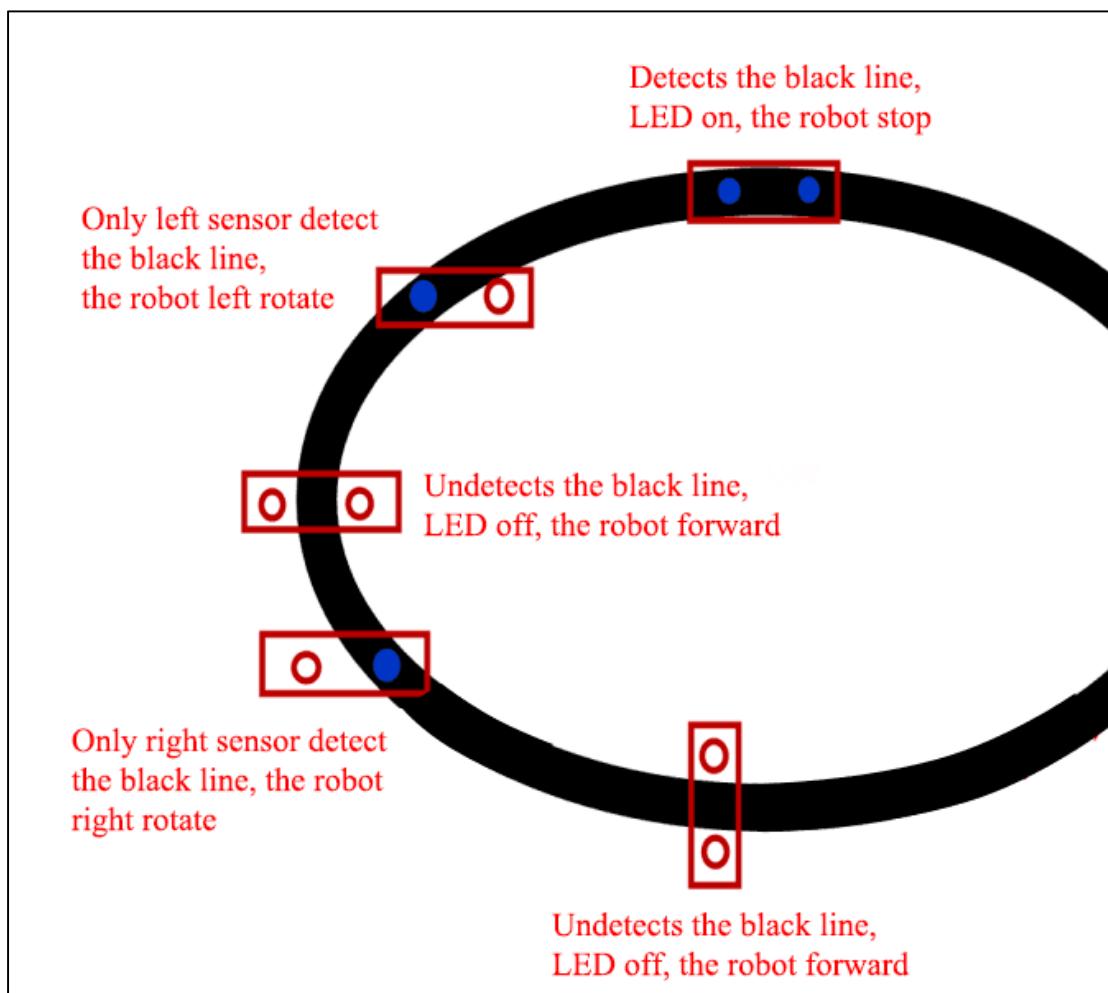
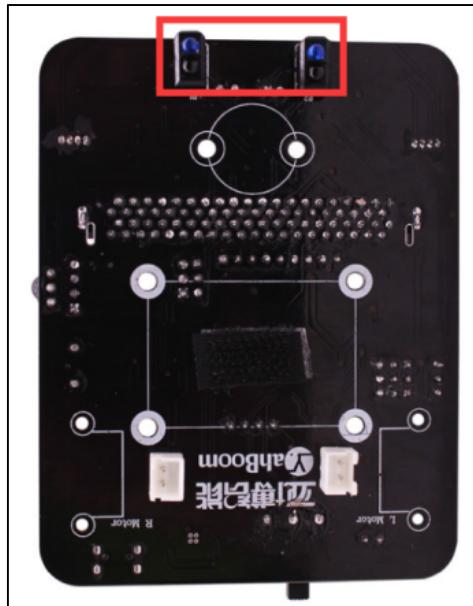
USB data cable *1

3. The principle of tracking

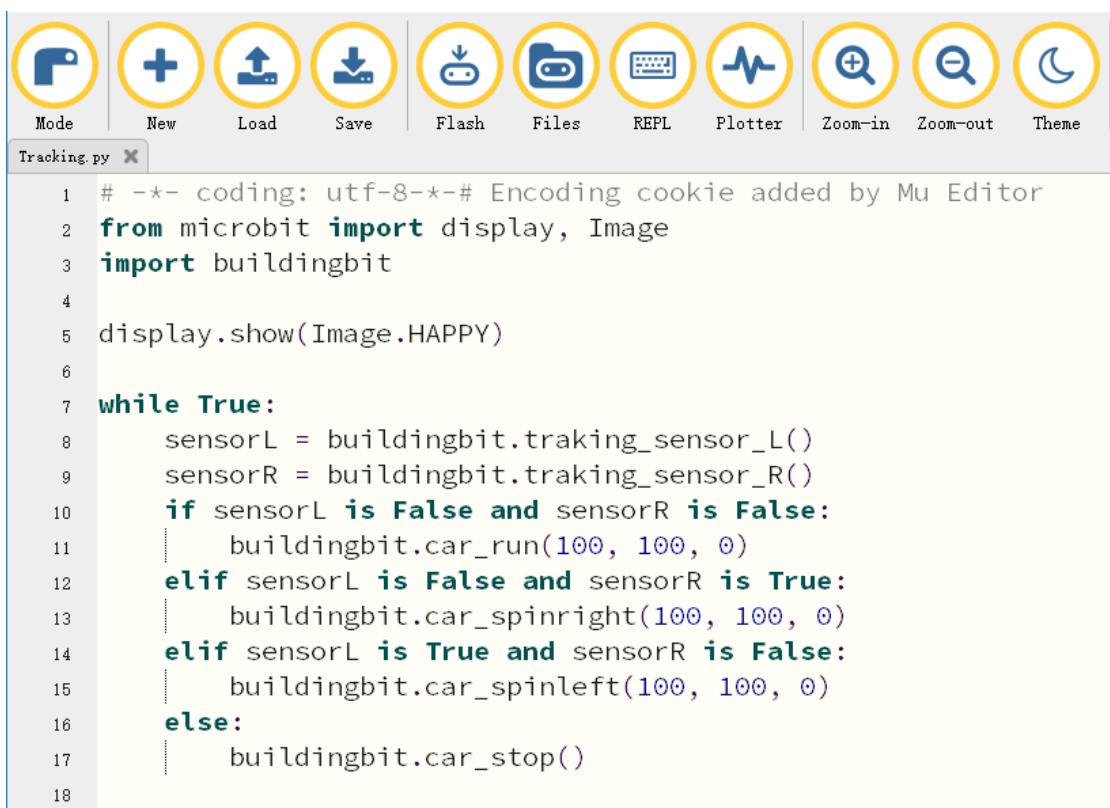
The basic principle of infrared sensor tracking is to use the reflective nature of the object. The purpose of our experiment is to make the car walk along the black line.

When the infrared light is emitted to the black line, it will be absorbed by the black line. When it is emitted to other colors, it will be reflected to the infrared receiving tube. When the tracking module of the car detects a black line, the

indicator light is on, and when the white object is detected, the indicator light is off. According to the this difference, we write the corresponding code to complete the car tracking function.



4. Programming



```

1 # -*- coding: utf-8-*# Encoding cookie added by Mu Editor
2 from microbit import display, Image
3 import buildingbit
4
5 display.show(Image.HAPPY)
6
7 while True:
8     sensorL = buildingbit.traking_sensor_L()
9     sensorR = buildingbit.traking_sensor_R()
10    if sensorL is False and sensorR is False:
11        buildingbit.car_run(100, 100, 0)
12    elif sensorL is False and sensorR is True:
13        buildingbit.car_spinright(100, 100, 0)
14    elif sensorL is True and sensorR is False:
15        buildingbit.car_spinleft(100, 100, 0)
16    else:
17        buildingbit.car_stop()
18

```

1) Import buildingbit library: `import buildingbit`

We also need to use display library: `from microbit import display`

2) `display.show(Image.HAPPY)` Display smile.

3) `buildingbit.car_run(255, 255, 0)` make robot advance.

4) `buildingbit.traking_sensor_L()`: If the black line sensor on the left detects a black line, it returns True; if it detects white, it returns False;

5) `buildingbit.traking_sensor_R()`: If the black line sensor on the right detects a black line, it returns True; if it detects white, it returns False;

If both the left and right sensors do not detect the black line, the cart moves advance;

If black line is not detected on the left and a black line is detected on the right, the car will spin right;

If a black line is detected on the left and a black line is not detected on the right, the car spin left;

In other cases, if both sensors of the cart detect a black line, the cart stops.

Code as shown below:

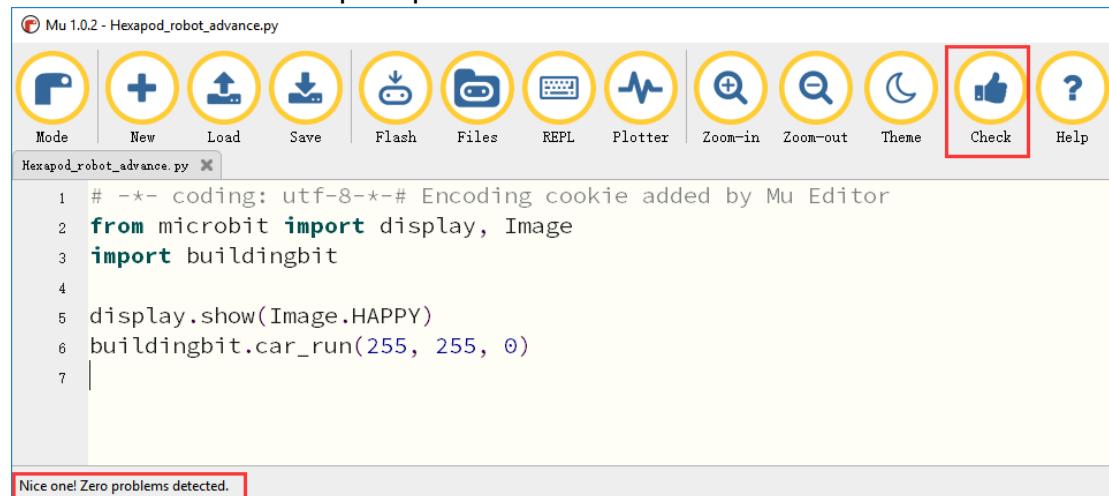
```

1 # -*- coding: utf-8-*-# Encoding cookie added by Mu Editor
2 from microbit import display, Image
3 import buildingbit
4
5 display.show(Image.HAPPY)
6
7 while True:
8     sensorL = buildingbit.traking_sensor_L()
9     sensorR = buildingbit.traking_sensor_R()
10    if sensorL is False and sensorR is False:
11        buildingbit.car_run(100, 100, 0)
12    elif sensorL is False and sensorR is True:
13        buildingbit.car_spinright(100, 100, 0)
14    elif sensorL is True and sensorR is False:
15        buildingbit.car_spinleft(100, 100, 0)
16    else:
17        buildingbit.car_stop()
18

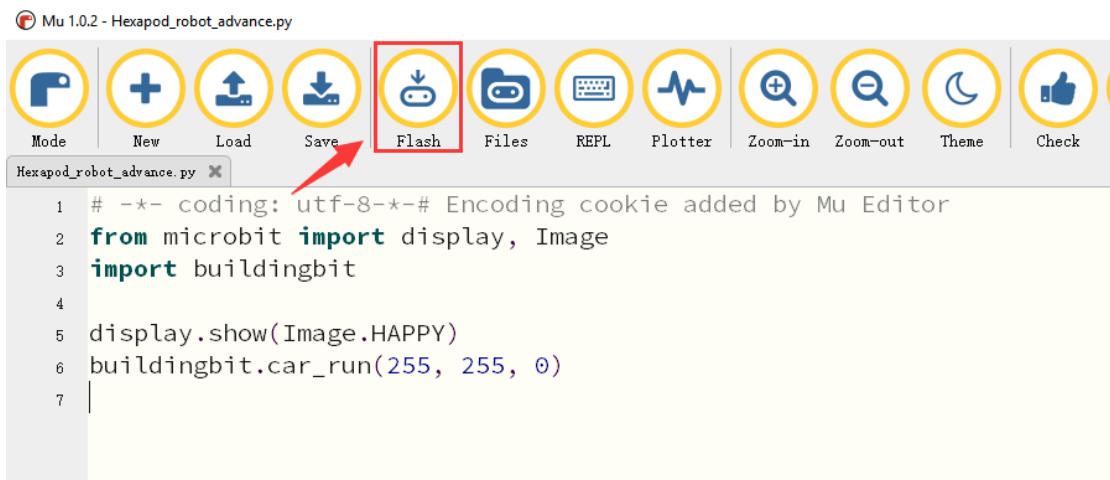
```

5. Download program

- 5.1 After programming is complete, please connect the computer and the micro:bit board with a Micro USB data cable.
- 5.2 You need to click the **【Check】** button to check if our code has an error. If a line appears with a cursor or an underscore, the program indicating this line is wrong. If there is no cursor or underline, it means that the code is correct, and the bottom left will prompt that the check is OK.



- 5.3 Click the **【Flash】** button to download the program to the micro:bit board of the building block following car.



If the program is wrong or the experimental phenomenon is wrong after downloading, please confirm whether you have downloaded the Buildingbit library hex file we provided to the micro: bit board.

For the specific method of adding library files, please refer to **【1.Preparation before class】---【Python programming】**