Roboduino car patrol line

1. Experimental preparation

1. Knowledge reserve

- Good programming ability (mainly C language)

2. Material preparation

- Roboduino car *1
- Eight-way tracking module *1
- 7.4V battery *1
- Several Dupont wires
- Several M3 copper pillars and M3 screws

2. Car wiring

After assembling the car, it is as shown below

2.1 Wiring of Roboduino and infrared sensor (this example uses I2C communication)

Roboduino	Infrared sensor
SDA	SDA
SCL	SCL
5V	5V

Roboduino	Infrared sensor
GND	GND

Main procedures

```
void setup()
{
    init_x_PID(); //PID Initialization
    Serial.begin(115200);  // start serial for output
    Wire.begin();        // join i2c bus (address optional for master)

    pinMode(KEY_PIN, INPUT_PULLUP);  // Set the key(button) pin to pull-up input
mode
    delay(100);

    while (getKeyState(KEY_PIN) != Press_KEY) ;//Press key1 to start line patrol

    Motor_init();//Motor initialization

}


void loop()
{
  I2Cdata();
  Car_line_track();

}
```

The main function is to perform PID processing for line patrol according to the value of the infrared probe, so that line patrol can be completed on the map with black lines and white background.

In line_pid.cpp, there is a parameter for adjusting PID line patrol. If you want to increase or decrease the speed and optimize the effect, you can adjust the macro definition value inside.

```

#define KPx  (15) //P
#define KIx  (0.0001) //I
#define KDx  (1) //D

#define Speed_Line (30) //Line patrol speed
```

- KPx: P value of pid line patrol
- KIx
- KDx
- Speed_Line: Speed of line patrol

Experimental phenomenon:

On the premise of ensuring that the wiring and installation are correct, after the 8-way line patrol module is calibrated, press the key button on the car to start line patrol.

If the 8-way module probe cannot detect the black and white lines normally, you need to wait for the module to work normally before pressing the key button on the car