

5.5 Drift movement

1. Learning goal:

Using Drifting function to control the car.

2. Experimental phenomena:

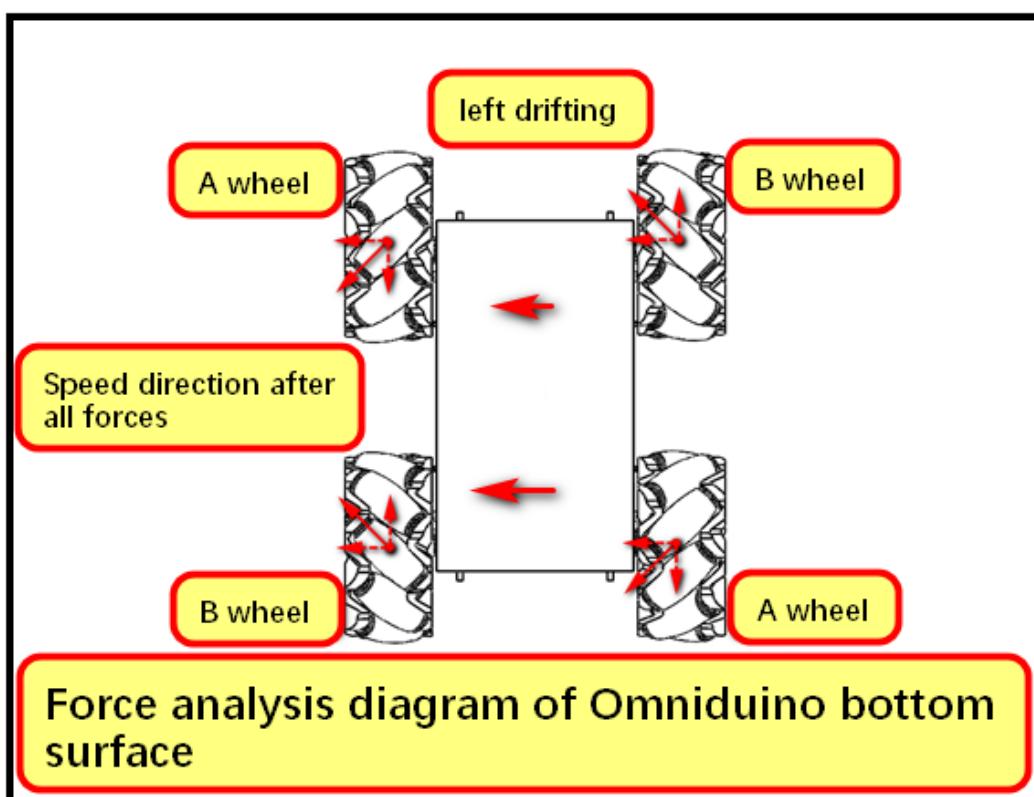
After we open the power of switch, press the button(K1) to start the program. The car will left drifting.

3. Force Analysis:

Analysis of the force of the car left drifting.

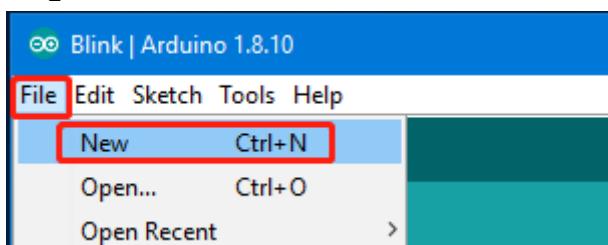
Drift to the left: When the A wheel reverse and the B wheel forward, the car will left drifting. If the speed of the front two wheels is less than the speed of the rear two wheels, car will move to the left and will also produce a centripetal force that points to the front of the car.

Therefore, the car will have a drift-like motion.

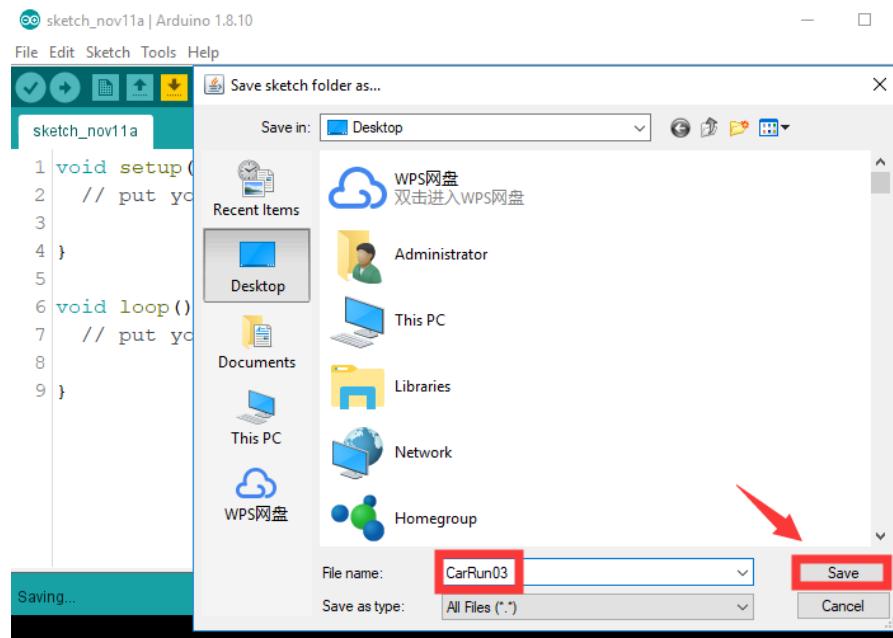


4. Create new project

4.1 Click 【File】-->【New】.



4.2 Press **Ctrl+S** to save and rename CarRun. As shown below.



4.3 We can see that there is a Serial folder with **CarRun03.ino** on the computer desktop.

4.4 We will **CarRun03.ino** as shown below.

```

void setup() {
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
}

```

The `setup()` function only runs once when the car is turned on or when the reset button is pressed, and the program for initializing the relevant content can be written;

The `loop()` function is the main loop function of the car and most of the data processing and logic processing are done in this function.

5. Programming

5.1 New create `carDrifting()` function, add parameters to modify the front and rear wheel speeds in the left drifting program of the cart.

```

void carDrifting(int Speed_Front, int Speed_Rear)
{
    Speed_Front = map(Speed_Front, 0, 160, 0, 2560);
    Speed_Rear = map(Speed_Rear, 0, 160, 0, 2560);
    pwm.setPWM(10, 0, Speed_Front); //Right front forward
    pwm.setPWM(11, 0, 0);
    pwm.setPWM(8, 0, 0);
    pwm.setPWM(9, 0, Speed_Rear); //Right rear reserve

    pwm.setPWM(13, 0, 0);
    pwm.setPWM(12, 0, Speed_Front); //Left front reserve
    pwm.setPWM(15, 0, Speed_Rear); //Left rear forward
    pwm.setPWM(14, 0, 0);
}

```

5.2 New create carRun03() function, which can achieve the function of the car movement, car left drifting.

```

void carRun03()
{
    //Reduce the front wheel speed so that the rear wheel speed is larger than the front wheel speed
    carDrifting(CarSpeedControl*0.6, CarSpeedControl);
}

```

5.3 In the loop() main loop function detection K1 button, when the K1 button is pressed, car start left drifting, when we press K1 button again, car will stop.

```

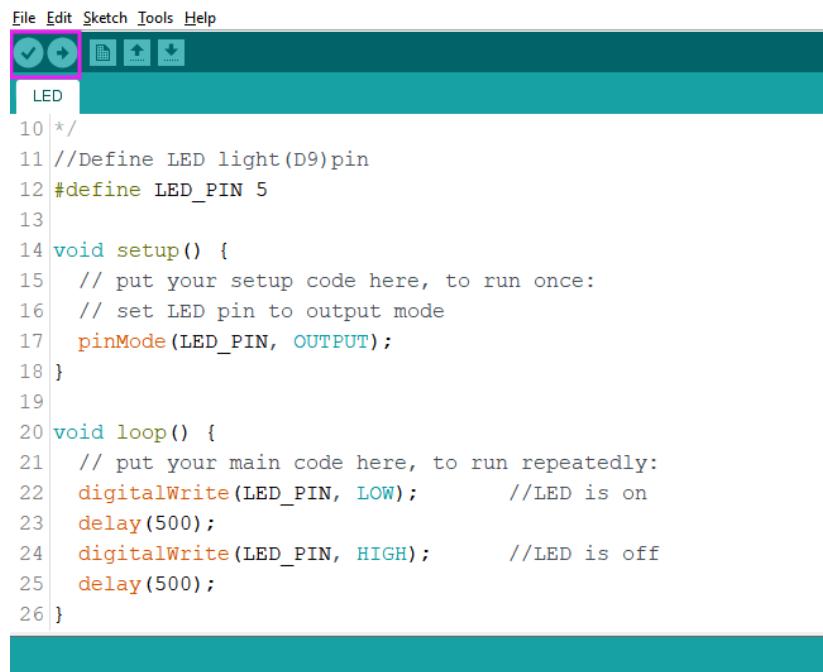
void loop() {
    // put your main code here, to run repeatedly:
    keyscan();

    if (button_press)
    {
        carRun03();
    }
    else
    {
        brake();
    }
}

```

6. Compiling and downloading code

6.1 After the code is written, press Ctrl+S to save, then click the “√” button to compile. If there is no problem, click “→” to upload (the car must be connected to the computer via the USB cable).

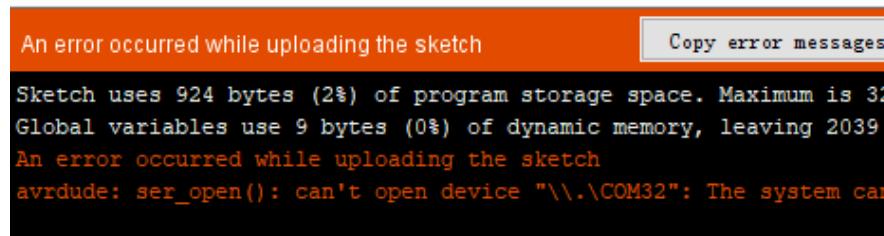


```

File Edit Sketch Tools Help
LED
10 */
11 //Define LED light(D9)pin
12 #define LED_PIN 5
13
14 void setup() {
15   // put your setup code here, to run once:
16   // set LED pin to output mode
17   pinMode(LED_PIN, OUTPUT);
18 }
19
20 void loop() {
21   // put your main code here, to run repeatedly:
22   digitalWrite(LED_PIN, LOW);      //LED is on
23   delay(500);
24   digitalWrite(LED_PIN, HIGH);     //LED is off
25   delay(500);
26 }

```

6.2 If the compilation passes normally, but the following error occurs during uploading, the reason may be that the wrong serial port or the serial port is occupied.



Solution: Open the device manager to see if there is a serial port with CH340 tag. If not, please restart the Omniduino car, then, re-plug the USB cable or replace a USB cable; If there is a serial port number, we need to close the other serial port or assistant software, avoid serial port occupation, and then re-select the serial port to ArduinolDE 【Tool】-->【Port】.

6.3 If there is an error like the following, it means that the library file is missing. Please copy the library file provided by the omniduino omnibus to the library file directory compiled by arduinolDE.

please refer to 【3.Development Environment Construction】 ---- 【3.4 Add additional library files】

```
Adafruit_PWMServoDriver.h: No such file or directory

CarRun:2:10: error: Adafruit_PWMServoDriver.h: No such file or directory
#include <Adafruit_PWMServoDriver.h>
^
compilation terminated.

exit status 1
Adafruit_PWMServoDriver.h: No such file or directory
```