Self-stabilizing mode

1. Learning goal

In this lesson, we will learn how to obtain gyroscope data, record the gyroscope angle of the cart itself, and when external forces change the cart angle, the cart can adjust angle by itself.

2. Looking for building blocks

The following is the location of the building blocks required for this programming.


2.1 The content in the Omniduino setup block will only run once when the Omniduino is turned on or the reset button is pressed.

We can write into the initialization and other content in this block.

The content in the loop is the main loop function of the Omniduino car, most of the data processing and logic processing are completed in this function.

2.2 Buzzer Play Tone block, you can choose Tone and Beats.
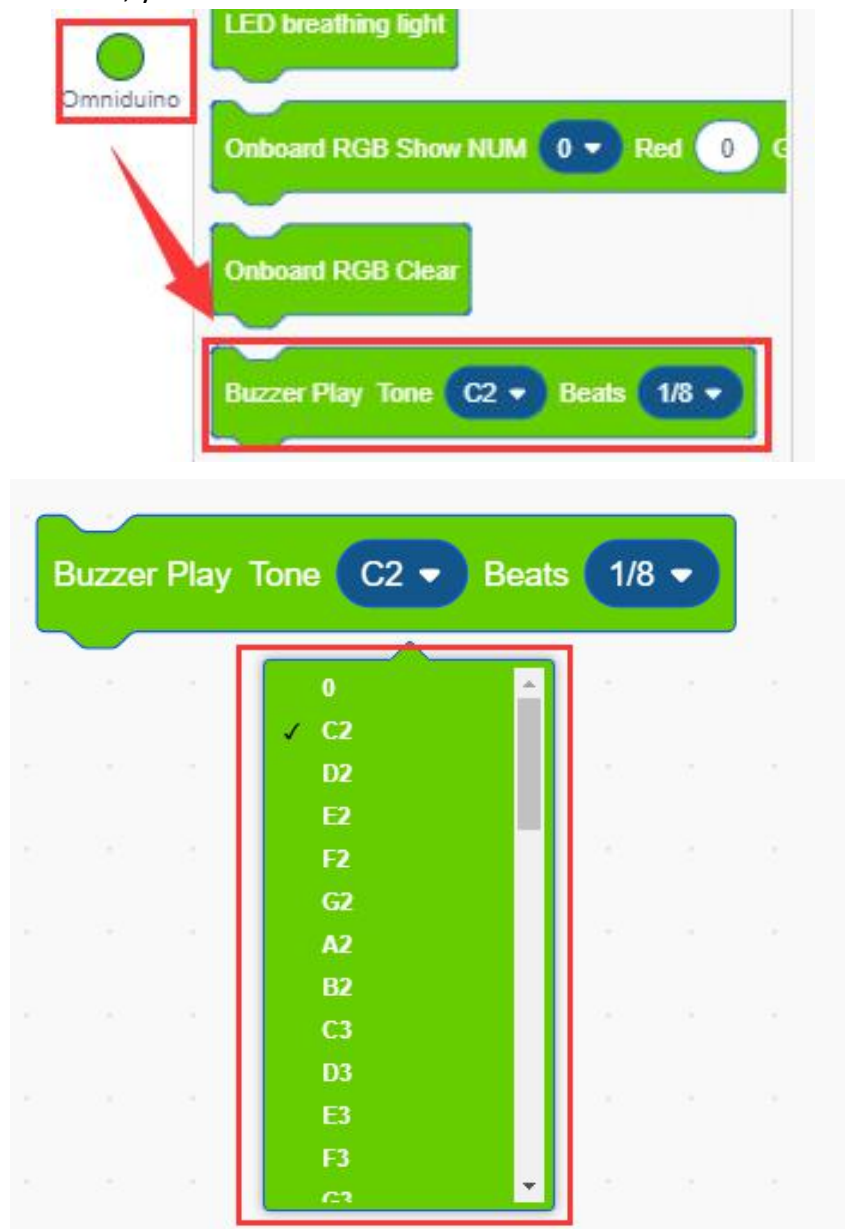
Due to the initialization of MPU6050 takes 5~6 seconds, we can set a buzzer. When the initialization of MPU6050 is successful, the buzzer will make a sound to prompt.
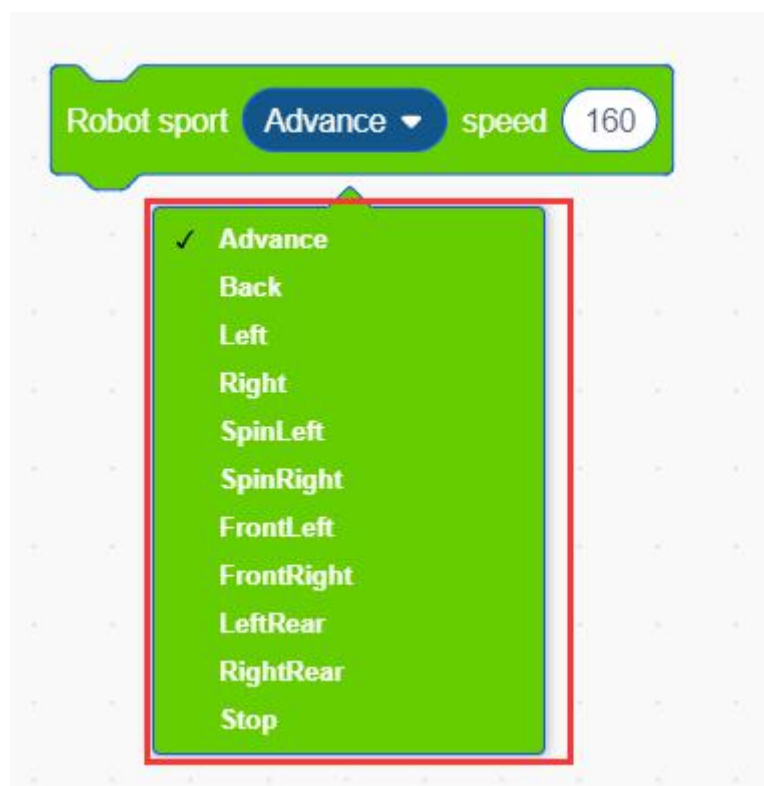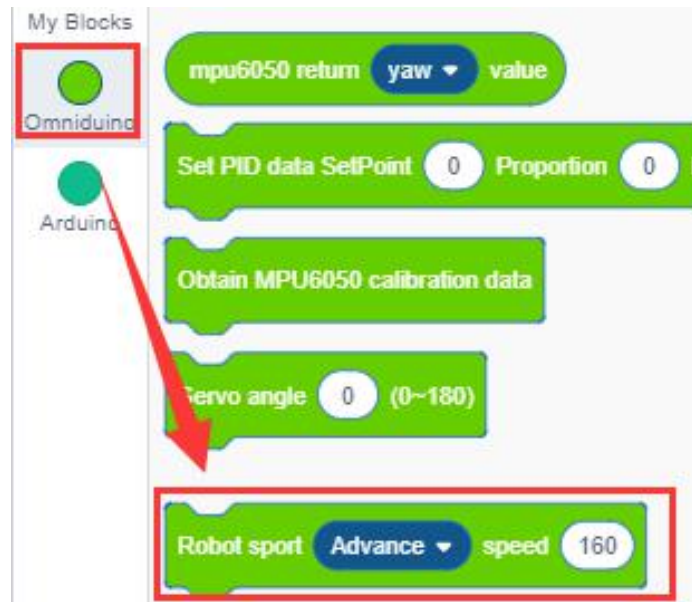
2.3 Setting MPU6050PID parameters block

User can adjust this parameter according to the actual situation.

2.4 Obtain the correction data calculated by the MPU6050 sensor.

Obtain MPU6050 calibration data

2.5 Robot sport building block, we can choose movement status, the speed is adjustable between 0-160.

My Blocks

Omniduino

Arduino

mpu6050 return yaw value

Set PID data SetPoint 0 Proportion 0

Obtain MPU6050 calibration data

Servo angle 0 (0~180)

Robot sport Advance speed 160

Robot sport Advance speed 160

✓ Advance
Back
Left
Right
SpinLeft
SpinRight
FrontLeft
FrontRight
LeftRear
RightRear
Stop

In this course, we need to set the speed to 0, because we are in the self-stabilization mode.

The car does not need to move at any speed, it will always remain in place.
(It is not the state where the motor is completely stopped. When the external force changes the angle of the car, the motor will rotate to make the car return to the initial position)

Combine blocks

3. Compiling and uploading the program

3.1 After building the blocks, click the **[code mode]** in the upper right corner of the Helloblock programming interface. We can see the corresponding Arduino code.

```cpp
#include "Arduino.h"
#include "YahBoom_Omniduino.h"

YahBoom_Omniduino Omniduino;
double button_press;
YahBoom_Omniduino_MotorServo cMotor;
YahBoom_Omniduino_Button cButton(8);

void setup()
{
    Omniduino.YahBoom_Omniduino_Init();
    cMotor.YahBoom_Omniduino_MotorServo_Init();
    cButton.YahBoom_Omniduino_Button_Init();
button_press=0;
    cMotor.Stop(0);

}

void loop()
{
    if(cButton.Get_Button_State())
    {
        button_press += 1;
        if(button_press == 2)
        {
            button_press=0;
        }
    }
    if(button_press == 1)
    {
```

```
DEBUG StatusLogger Shut down RollingFileManager
C:\Users\Administrator\AppData\Local\Arduino15/logs
/application.log, all resources released: true
>TRACE StatusLogger XmlConfiguration stopped 2
remaining Appenders.
TRACE StatusLogger XmlConfiguration cleaning
Appenders from 2 LoggerConfigs.
>DEBUG StatusLogger Stopped
```

3.2 Then, you need to connect Omniduino car to your computer. Select the CH340 port number identified in the previous step in the upper right corner. Then, click the up arrow to start compiling and uploading the program.
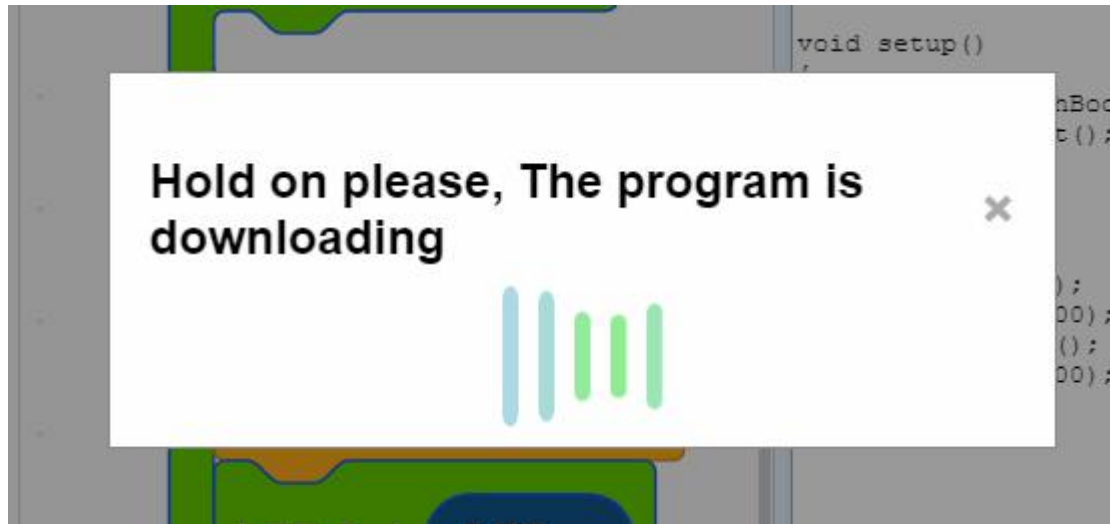
3.3 When the words **"Done compiling   Done uploading"** appear in the lower right corner of the programming interface, which means the program has been uploaded.
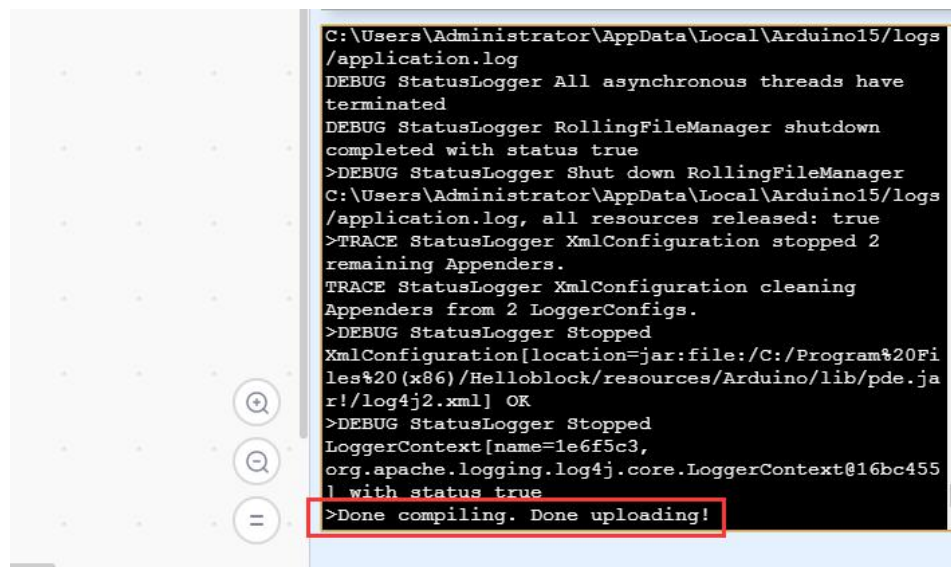
4. Experimental phenomenon

After the program is downloaded. Put car on a level ground.

Open the power switch of the car and wait for 5~6 seconds. After the MPU6050 is initialized successfully, the buzzer will make a sound. When an external force changes the angle of the car, the car can adjust back by itself.