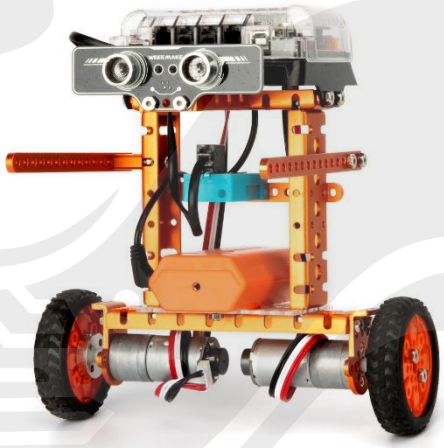
L. Self-balancing Robot

Material

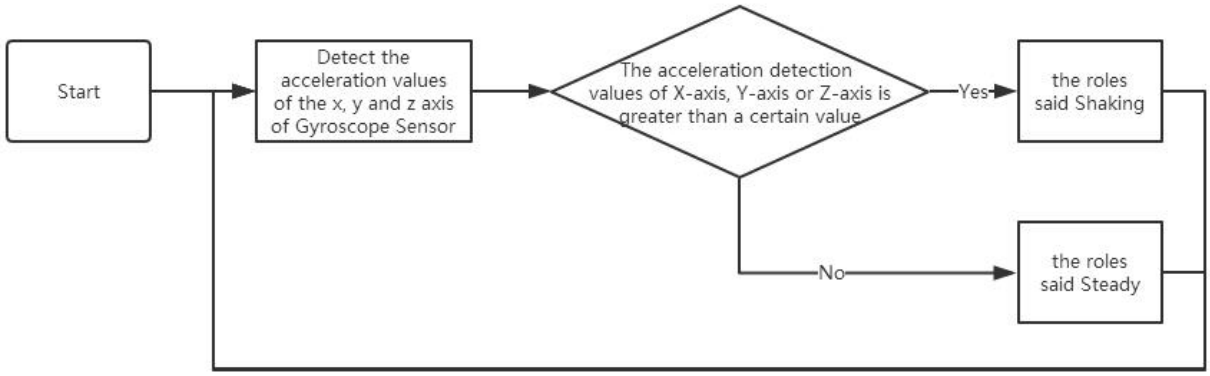
1. Self-balancing robot
2. Computer



Project #1: Mastering the use of gyroscope sensor

Use the block " Gyro sensor... acceleration..." to detect the accelerations of the X axis, Y axis and Z axis of the gyroscope, so as to learn changes of the acceleration of Gyroscope Sensor, and combine the roles with it. When the gyroscope is stationary or shaking, the roles can perform corresponding actions.

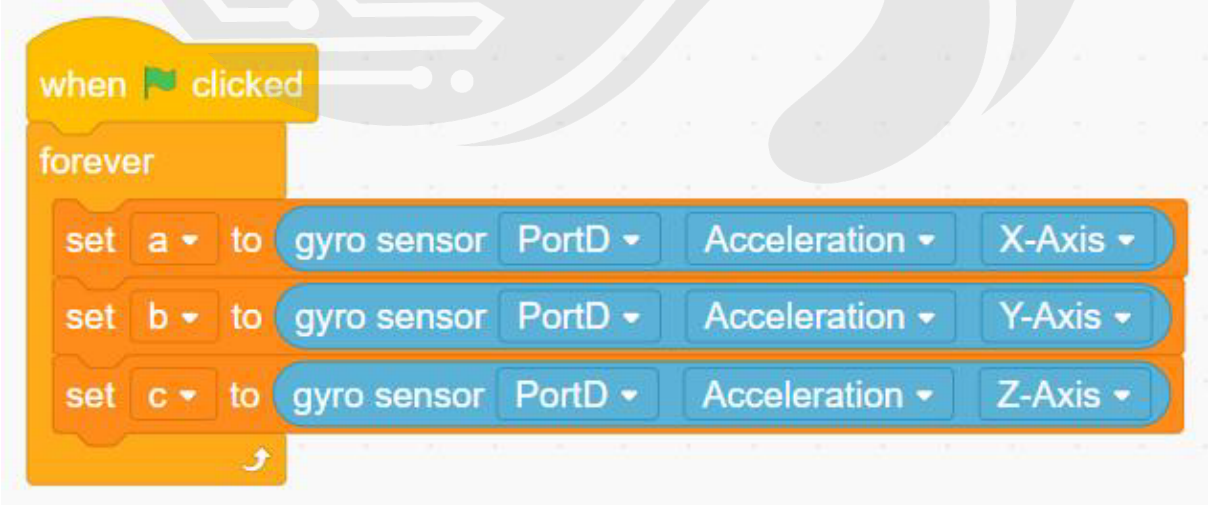
Program Idea



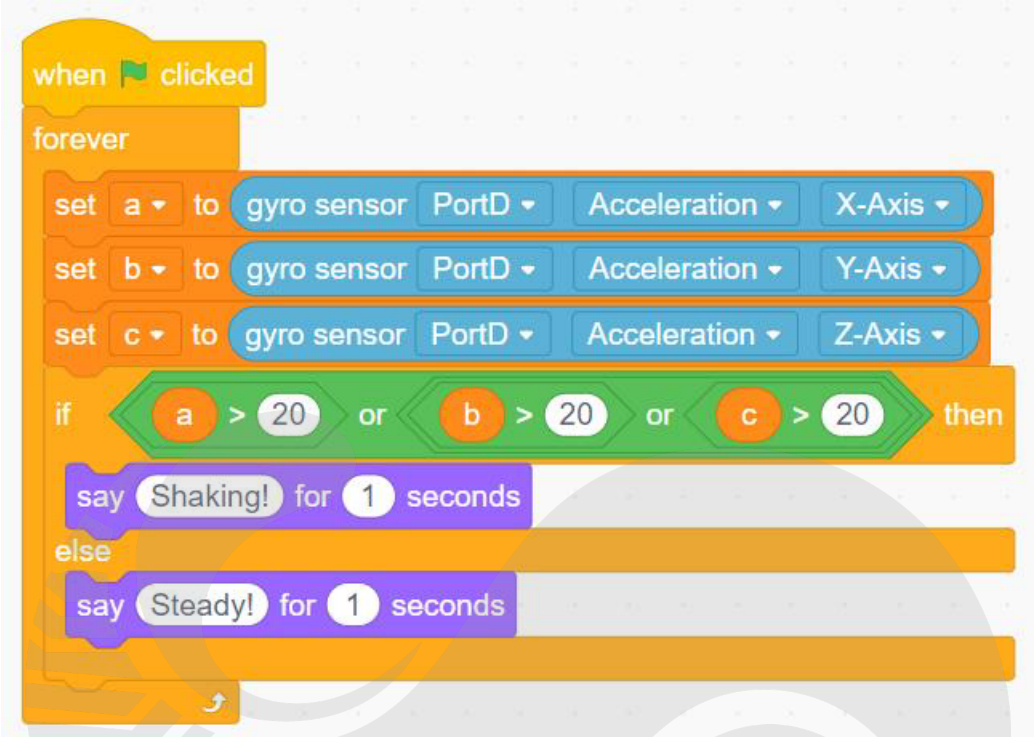
Reference Code

gyro sensor PortA Gyration X-Axis	Obtaining the rotation angle/rotational acceleration value on a certain axis.

Acceleration Measurement of Gyro Sensor



Application of Gyro Sensor Acceleration



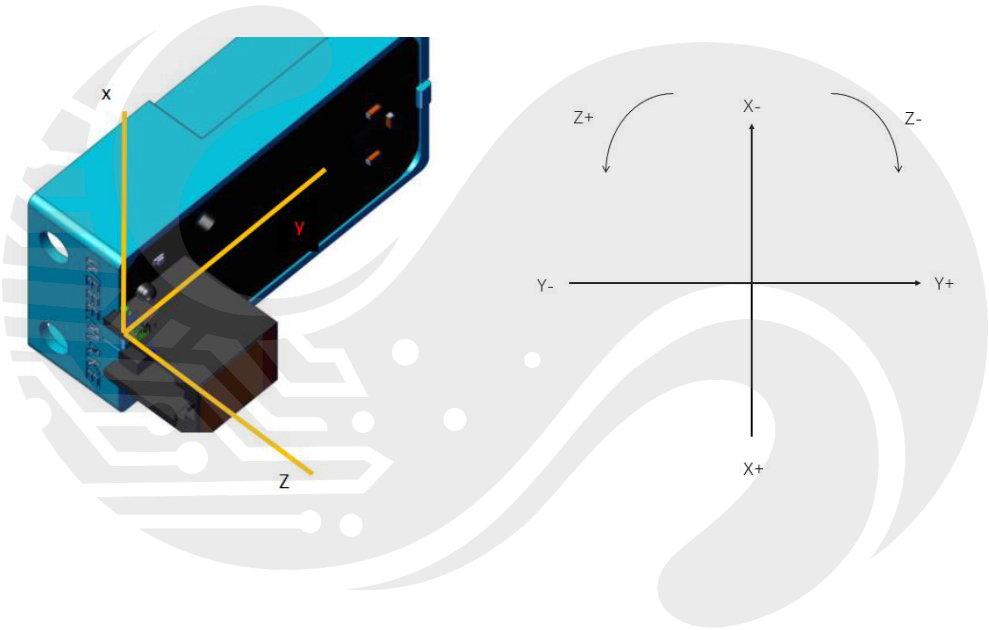
Note: To detect the acceleration value of the axis, it is best to disassemble the sensor from the robot.

After running the program, when the gyroscope is stationary, the roles will say “Steady!” for 1 second, when the gyroscope moves, the roles will say “Shaking!” for 1 second.

Project #2: Testing and application programs for gyroscope sensor

Use the block " Gyro sensor ... " to detect the angles of the X axis, Y axis and Z axis of the gyroscope, so as to learn the changes of the gyroscope sensor angle, and combine the roles with it. When the gyroscope tilts or rotates, the roles can perform corresponding actions.

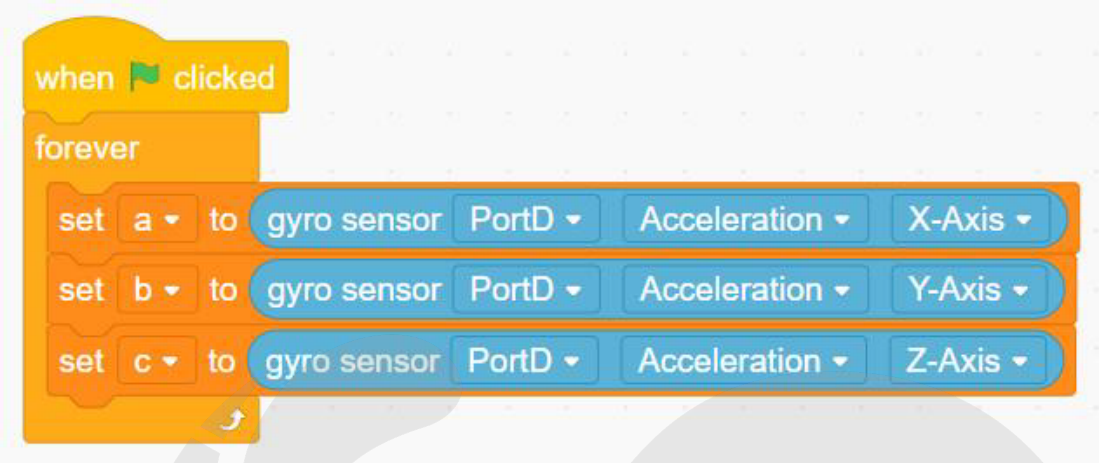
Program Idea



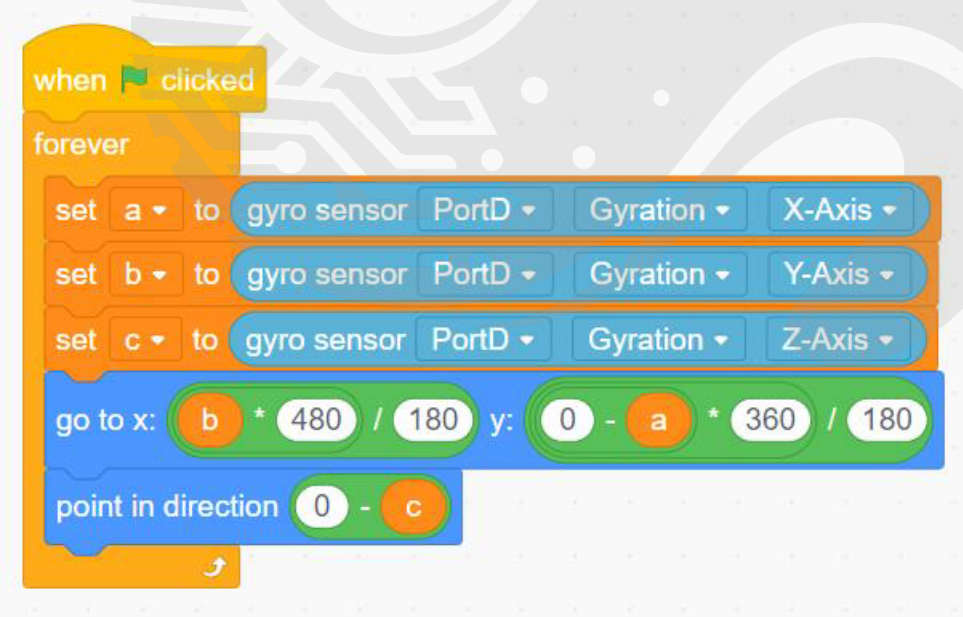
Note: In the context of the gyroscope sensor, **acceleration** refers to the change in speed that occurs when it rotates around an axis. We can think of it as rotational acceleration. Similarly, when the gyroscope rotates around an axis, it also experiences a change in angle. So, we can say that it measures the rotational angle.

Reference Code

Angle Measurement of Gyro Sensor



Application of Gyro Sensor Angle



After executing the program, when the gyroscope leans forward, the role moves upward. When the gyroscope leans backward, the role moves down. When the gyroscope leans to the left, the role moves to the left. When the gyroscope moves to the right, the role moves to the right.

Project #3: Experience the self-balancing robot.

It is difficult to write the program for a self-balancing robot by graphical programming as it requires a very complicated processing of the data detected by gyroscope. So in here we just observe how the self-balancing robot works。

We should restore the factory program onto the mainboard firstly, select the 12-in-1 storm in the factory firmware. Then we make the robot upright, turn on the power switch, and press the D key of the remote controller to enter the self-balancing mode. We can learn the self-balancing robot keeps walking upright. At the same time, other robots we made before also can realize rapid experience by restoring the factory firmware.

Button A——Manual Mode
Button B——Obstacle Avoidance
Button C——Line-following Mode
Button D——Self-balancing Mode
Button E——Robot Arm Mode
(In this mode:
Button ←——Port 3 Encoder Motor A clockwise
Button →——Port 3 Encoder Motor A anticlockwise
Button ↑——Port 4 Encoder Motor B clockwise
Button ↓——Port 4 Encoder Motor B anticlockwise)
Button F——RGB LED Circle
(In this mode:
Button 1——Rotating water light;
Button 2——Rainbow light;
Button 3——Turn off the light)
Button ↑——Forward
Button ↓——Backward
Button ←——Turn left
Button →——Turn right

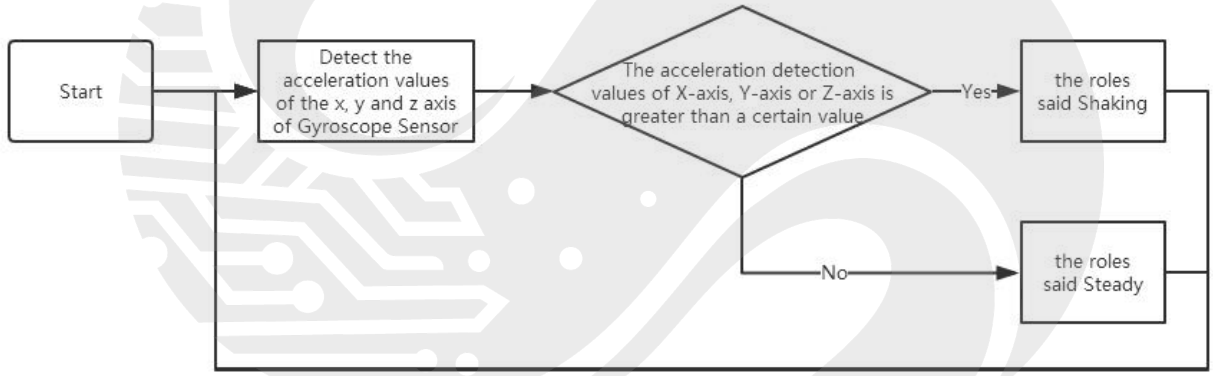
Button OK——RGB on ultrasonic sensor
Number Button 0——Speed 0
Number Button 1——Speed 1
Number Button 2——Speed 2
Number Button 3——Speed 3
Number Button 4——Speed 4
Number Button 5——Speed 5
Number Button 6——M1 clockwise
Number Button 7——Port D/M2 gripper clamp
Number Button 8——Port D/M2 gripper release
Number Button 9——M1 anti-clockwise

ARDUINO

Project #1: Mastering the Use of Gyroscope Sensor Arduino

Use the block " Gyro sensor... acceleration..." to detect the accelerations of the X axis, Y axis and Z axis of the gyroscope, so as to learn changes of the acceleration of Gyroscope Sensor, and combine the roles with it. When the gyroscope is stationary or shaking, the roles can perform corresponding actions.

Program Idea



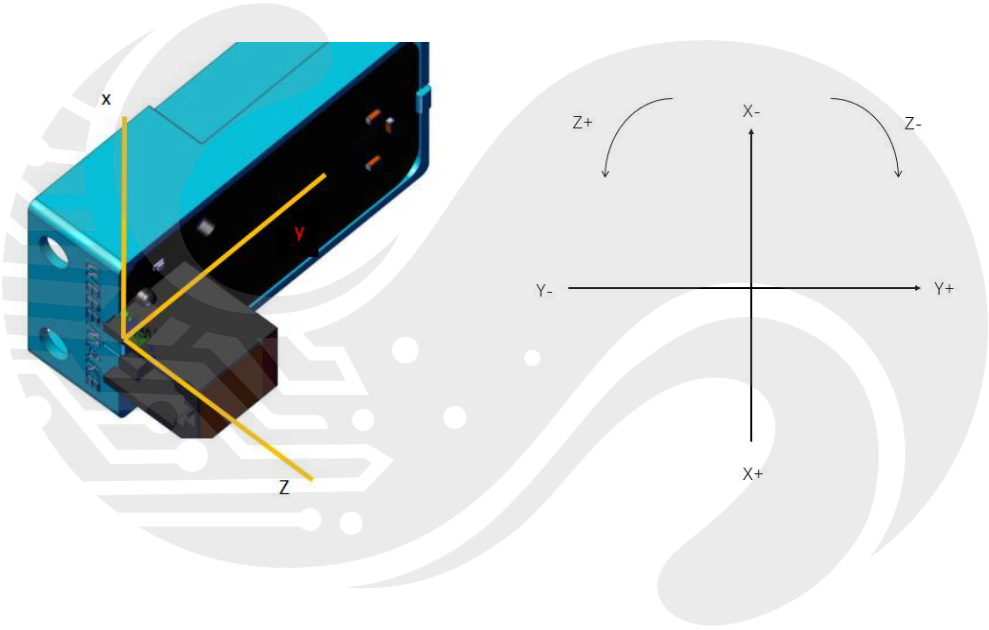
Reference Code

```
#include<WeELF328P.h>

void update();

WeGyroSensor gyro_D(PORT_D);
double v_a; //a;
double v_b; //b;
double v_c; //c;

void setup(){
        gyro_D.begin();

        //event_whenflagclicked();
}

void loop(){
        update();
        v_a = gyro_D.getGyroX();
        v_b = gyro_D.getGyroY();
        v_c = gyro_D.getGyroZ();
        if(v_a > 20 || (v_b > 20 || v_c > 20)){
                //looks_sayforsecs();
        }else{
                //looks_sayforsecs();
        }
}

void update(){
        gyro_D.update();
}
```

Project #2: Testing and Application Programs for Gyroscope Sensor Arduino

Use the block " Gyro sensor ... " to detect the angles of the X axis, Y axis and Z axis of the gyroscope, so as to learn the changes of the gyroscope sensor angle, and combine the roles with it. When the gyroscope tilts or rotates, the roles can perform corresponding actions.

Program Idea



Note: In the context of the gyroscope sensor, **acceleration** refers to the change in speed that occurs when it rotates around an axis. We can think of it as rotational acceleration. Similarly, when the gyroscope rotates around an axis, it also experiences a change in angle. So, we can say that it measures the rotational angle.

Reference Code

```
#include<WeELF328P.h>

void update();

WeGyroSensor gyro_D(PORT_D);
double v_a; //a;
double v_b; //b;
double v_c; //c;

void setup(){
        gyro_D.begin();

        //event_whenflagclicked();
}

void loop(){
        update();
        v_a = gyro_D.getAngleX();
        v_b = gyro_D.getAngleY();
        v_c = gyro_D.getAngleZ();
        //motion_gotoxy();
        //motion_pointindirection();
}

void update(){
        gyro_D.update();
}
```