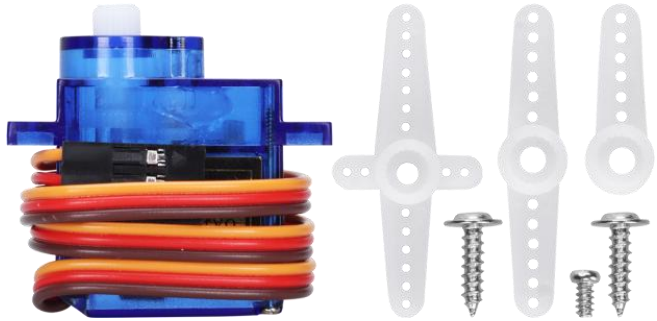
Project 19-Infrared remote control door servo

1. project description

With this simple Arduino project, you can control a servo using a remote. All you need is a ZY-type-c Nano, a G90 servo motor, an IR remote control and an IR receiver.

2. Introduction to modules

2.1 SG90 steering gear



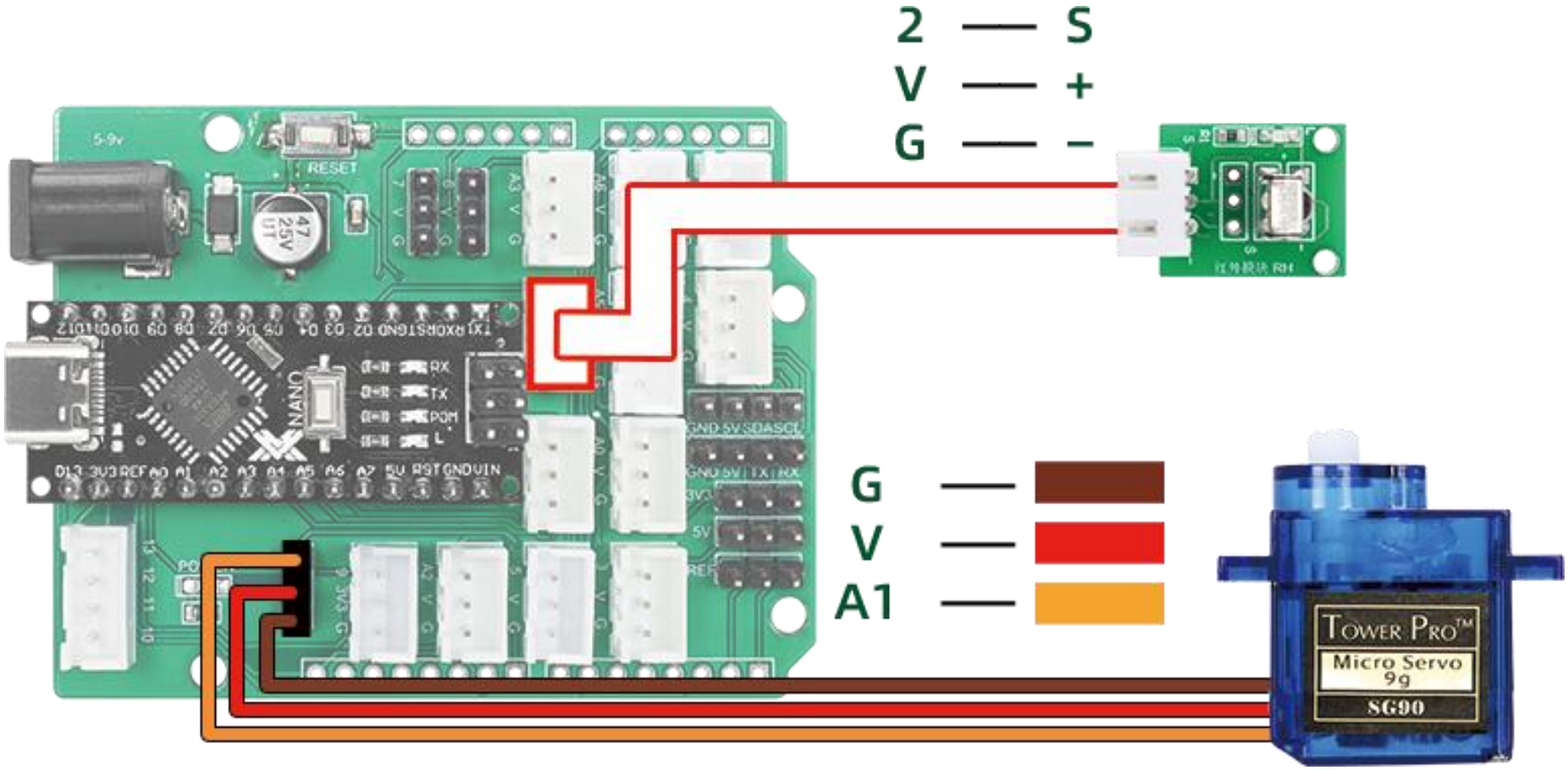
SG90 is a low-cost, high-output power servo motor. It can rotate up to 180 degrees and up to 90 degrees per step. Plus, it's small enough to easily fit into your robotics projects. Most importantly, it only requires an output pulse signal to control its movement.

The servo motor control pulse signal period is a 20MS pulse width modulation signal (PWM), the pulse width is from 0.5ms to 2.5ms, and the corresponding steering position changes linearly from 0 to 180 degrees. In other words, if a certain pulse width is provided to the steering gear, its output shaft will maintain a certain corresponding angle. No matter how the external torque changes, it will not change the output angle to a new corresponding position until another pulse signal is provided to it.

There is a reference circuit inside the steering gear, and the generation period is 20ms . Theoretically, the pulse width distribution should be between 1ms and 2ms, but in fact it can be between 0.5ms and 2.5ms. The pulse width corresponds to the rotation angle from 0° to 180°. There is a comparator that compares the external signal with the reference signal to determine the direction and size, thereby generating a motor rotation signal.

The steering gear is a position servo driver, and its rotation range is generally 180° or 360° . Here, we are using a 180° servo .

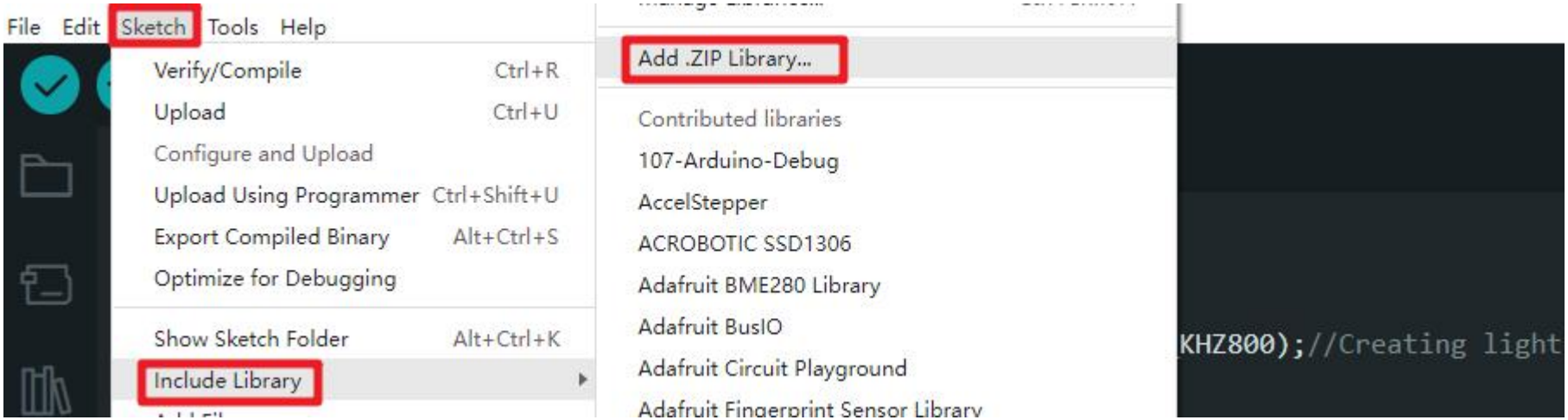
3. Project wiring diagram



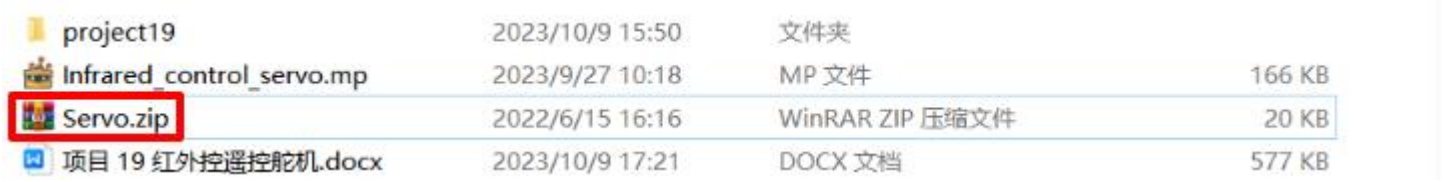
4. Add servo library

Here, the servo library used is Servo.zip .

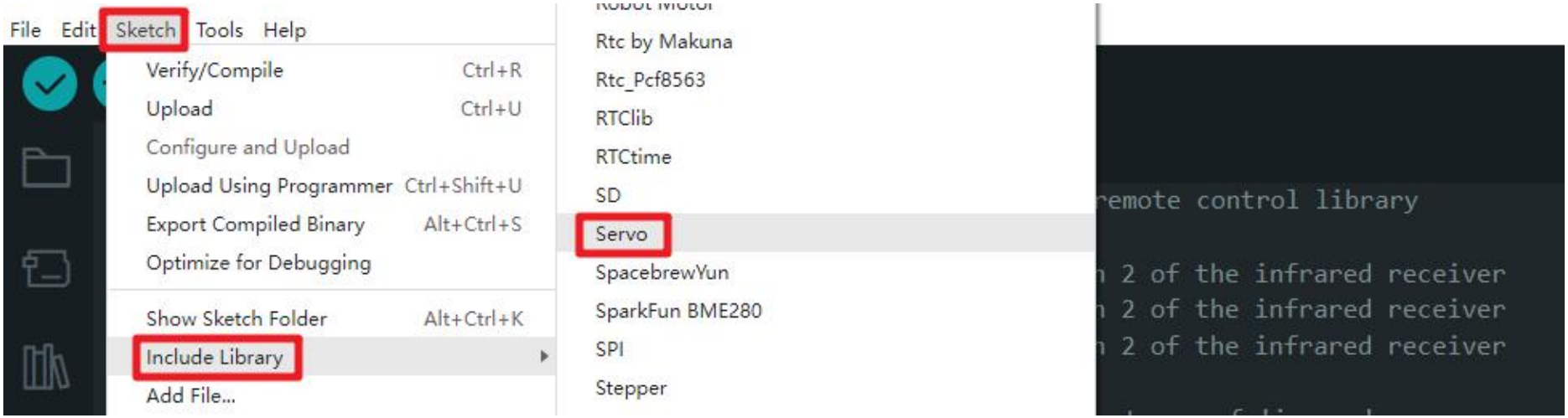
In the Arduino IDE, navigate to Sketch > Include Library > Add .ZIP Library and at the top of the drop-down list, select the "Add .ZIP Library" option.



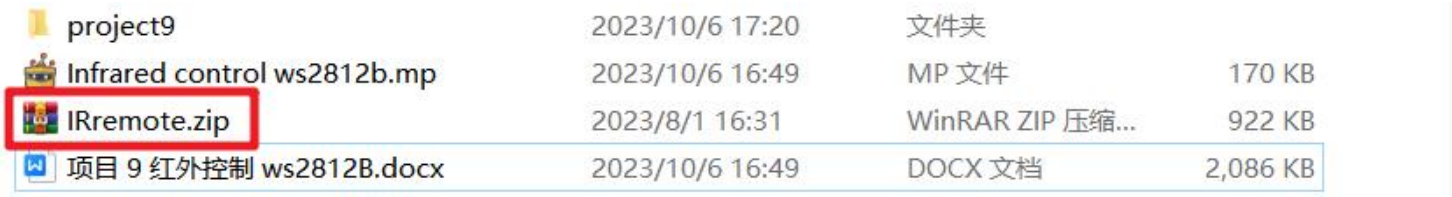
The system will prompt you to select the library to be added , as shown below, navigate to the path location of the *Servo* .zip file saved in the computer (*Item 19 Infrared Remote Control Servo\Servo.zip*) and open it .



Open the Sketch > Include Library menu. You should now see Libraries at the bottom of the drop-down menu. It's ready to use in your sketches .

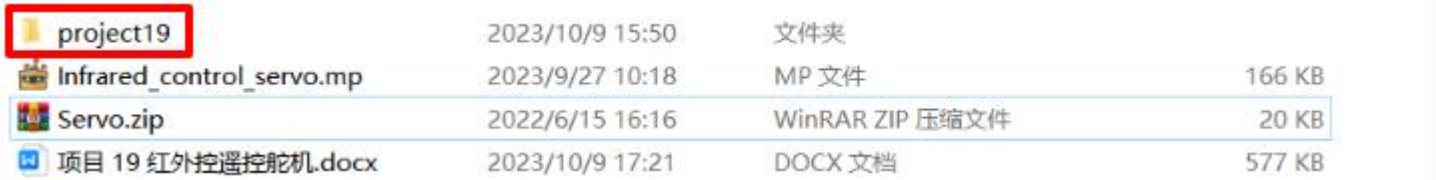


Also confirm that the infrared remote control library file IRremote.zip has been successfully added. If it has not been added, please go back to item 9 to see how to add the library.



5. Download Arduino code

Open the project Arduino code file (path: project 19 infrared remote control servo\project19\project19.ino)



Connect the main control board to the computer using USB, select the board type as Nano, select the newly displayed COM number, click "Download" to start compiling and downloading the program to the main control board.

Code analysis:

```
#include <IRremote.h>      //添加红外遥控库    Add infrared remote control library
#include <Servo.h>         //添加舵机库    Add servo library
#define receiver 2         //定义红外接收器引脚2   Define pin 2 of the infrared receiver
#define doorOpen 0XFFE01F //定义按钮 7 的红外接收编码值   Define the infrared received coded value of button 7
#define doorClose 0XFFA857//定义按钮 8 的红外接收编码值   Define the infrared receive coded value of button 8

IRrecv irrecv(receiver);  //创建` irrecv `实例     create instance of 'irrecv'
decode_results results;   //创建` decode_results `实例     create instance of 'decode_results'
Servo myservo;            //实例化舵机为myservo create instance of 'myservo'
```

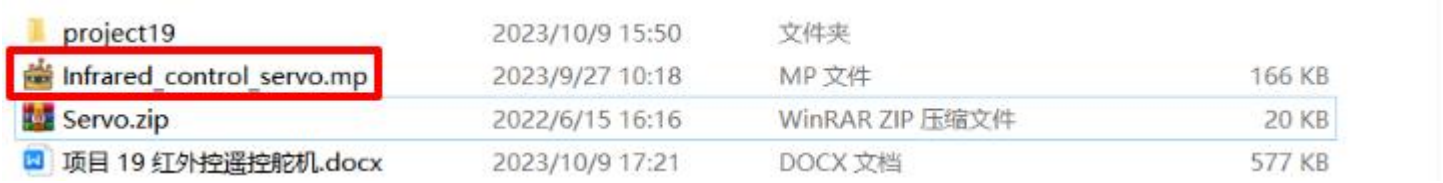
```
void setup()
{
    Serial.begin(9600);   //设置波特率9600    Set the baud rate to 9600
    irrecv.enableIRIn();  //启动红外接收器    Start the receiver
    myservo.attach(15);   //设置舵机引脚为15(A1)   attaches the servo on pin 15 to the servo object
    myservo.write(10);    //初始化设置舵机角度为10   The servo Angle is initially set to 10
}
void loop()
{
    if (irrecv.decode(&results)){
        Serial.print("value = ");
        Serial.println(results.value, HEX);   //将接收到红外编码值打印输出到串口监视器  Print the received infrared coded value
        if (results.value == doorOpen){       //根据您的IR遥控按钮号码更改  change according to your IR remote button number
            myservo.write(90);                //改变舵机的转动角度  Change the degree of rotation of the steering gear
            delay(15);
        }
        if (results.value == doorClose){      //根据您的IR遥控按钮号码更改  change according to your IR remote button number
            myservo.write(10);
            delay(15);
        }
        irrecv.resume();                      //接收下一个红外信号   receive the next value
    }
}
```

Project effect:

When the infrared remote control button "7" is pressed, the servo connected to pin A1 (15) rotates to 90 (opening the door); when the infrared remote control button "8" is pressed, the servo rotates to 10 (closing the door) .

4. Download Mind+ graphical code

Open the project Mind+code file (path: Project 19 Infrared remote control servo\Infrared_control_servo.mp)



Connect the main control board to the computer with a USB cable and select the newly appeared CH340 serial port COM number. Click "Upload to Device" to complete the code upload.

Programming analysis:

This project uses a third-party library. The code in the attachment has integrated the library. When you need to customize a new infrared remote control servo graphical code file, you need to add a servo library first. Click "Extend", then select "Actuator" and then add "Servo Module"

After the addition is completed, you can see that there is an "actuator" category in the building block area, and you can program the servo.

Total code: (For the infrared coding value corresponding to the button, please see the comparison table in item 9)


Nano 主程序开始
循环执行
设置 红外编码 的值为 读取红外接收模块 数字引脚 2
如果 变量 红外编码 = FFE01F 那么执行
设置 door 的值为 1
否则 如果 变量 红外编码 = FFA857 那么执行
设置 door 的值为 0
如果 变量 door = 1 那么执行
设置 A1 引脚伺服舵机为 90 度
如果 变量 door = 0 那么执行
设置 A1 引脚伺服舵机为 10 度
接收红外遥控器发来的信息保存到变量 "红外编码"
根据编码值控制继电器通断