Lesson 2 Light Detection and Placement

Please prepare a light-blocking prop for this game. You can use 3D printer to print a prop shown in the following image or use hard material to make one.

1. Project Principle

This game uses light sensor to sense the ambient light intensity to judge if it is covered by object. Then robot arm is controlled to place object in the corresponding area.

A photosensitive sensor is a sensitive device that responds or converts to external light signals or light radiation. The sensor mainly contains a QT523C (photodiode) and an LM358 chip (voltage comparator).

When it works, sensor uses QT523C to convert light signal to an electrical signal output, which is then converted to a voltage ranging 0 to 5V and received by the data collector after A/D conversion in the range of 0-1023. The greater the external brightness intensity, the smaller the output voltage, so the brightness is inversely proportional to the output voltage.

The path of the program file: “6. Secondary Devlopment/Sensor-extension Game/ Python Development/ Program Files/ Sound Detection and Placement/main.py”

```
if __name__ == '__main__':
  arm.go_home() #机械臂复位，回到初始位置
  nozzle.set_angle(0,1000) #吸嘴角度置0
  time.sleep_ms(2000)
  num = 0 #木块计数变量
  angle = [12, 35, 55] #角度补偿
  while True:
    light = light_sendor.read() #光线传感器检测函数
    print(light)
    if light > 900: # 光线传感器被挡
      print('num:',num+1)
      buzzer.setBuzzer(100) #蜂鸣器响一下
      time.sleep_ms(500)
      arm.set_position((0,-165,100),1200) #机械臂移动到吸取位置上方，等待2秒后吸取
      time.sleep_ms(2000)
      arm.set_position((0,-165,86),600) #吸取木块
      nozzle.on() # 打开气泵
      time.sleep_ms(1000)
      arm.set_position((0,-165,180),1000) #机械臂抬起来
      time.sleep_ms(1000)
      arm.set_position((120,-20-60*num,180),1500) # 移动到放置位置上方
      nozzle.set_angle(angle[num],1500) #设置角度补偿，使木块放正
      time.sleep_ms(1500)
      arm.set_position((120,-20-60*num,88),1000) #放置木块
      time.sleep_ms(1200)
      nozzle.off() # 关闭气泵
      arm.set_position((120,-20-60*num,200),1000) # 机械臂抬起来
      time.sleep_ms(1000)
      arm.go_home() #机械臂复位，回到初始位置
      nozzle.set_angle(0,1800) #吸嘴角度置0
      time.sleep_ms(2000)
      num += 1       #木块计数变量加1
      if num >= 3:
        num = 0
        buzzer.setBuzzer(80) #蜂鸣器响一下
        time.sleep_ms(100)
        buzzer.setBuzzer(80) #蜂鸣器响一下

```

Use light sensor to detect the light intensity, and then send the signal to microcontroller for processing. When the light intensity is lower than the set

threshold in program, buzzer will sound for responding, and MaxArm will perform the corresponding action.

The first detected object will be placed at position 1. The second detected object will be placed at position 2. The third detected object will be place at position 3. As the figure shown below:

Then, execute the functions for controlling action, buzzer and air pump to control robot arm to move object and place it to the corresponding position on the left side. (Take robotic arm as the first person view)

Note:

1. Since light sensor is susceptible to the ambient light, you may need to adjust the potentiometer of sensor based on the actual situation.

2. Due to the swing of the robot arm, the specific object landing point may be deviated, but the overall range will be as shown above.

3. When strong light is detected, the blue LED light on sensor will keep on. As the light becomes weak, the blue LED light will be off. Please note that this has nothing to do with the brightness of the sensor being covered by the object.

---

2. Preparation

2.1 Hardware

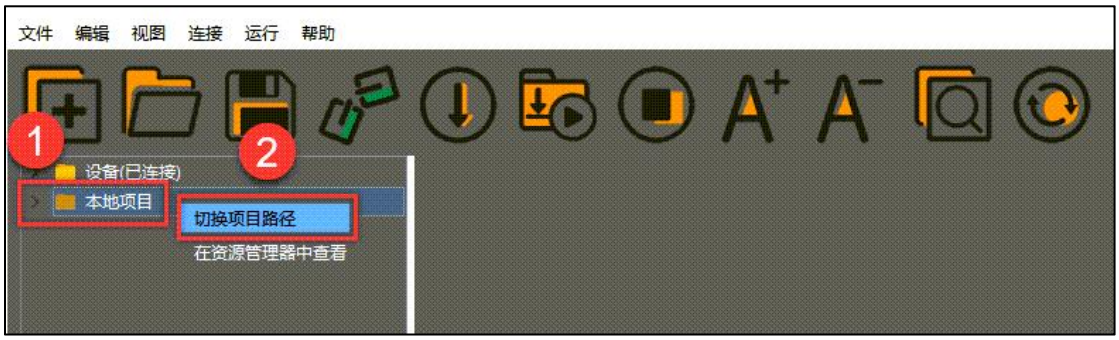
Please assemble sound sensor to the corresponding position on MaxArm according to the tutorial in folder “Lesson 1 Sensor Assembly” under the same directory.

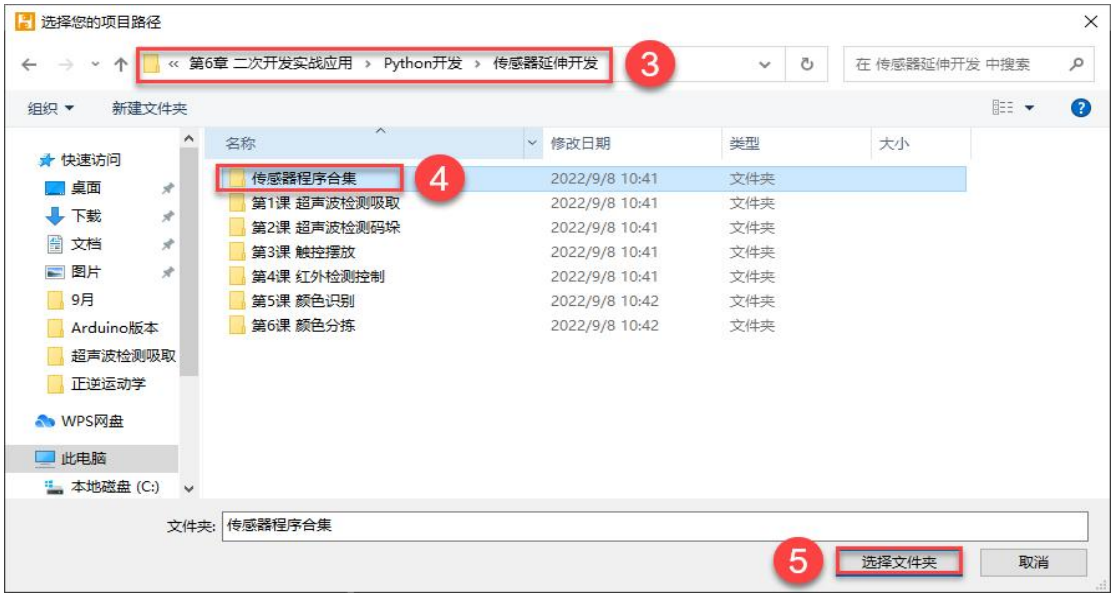
2.2 Software

Please connect MaxArm to Python editor according to the tutorial in folder “4. MaxArm Underlying Program Learning/Python Development/Lesson 1 Set Development Environment”.

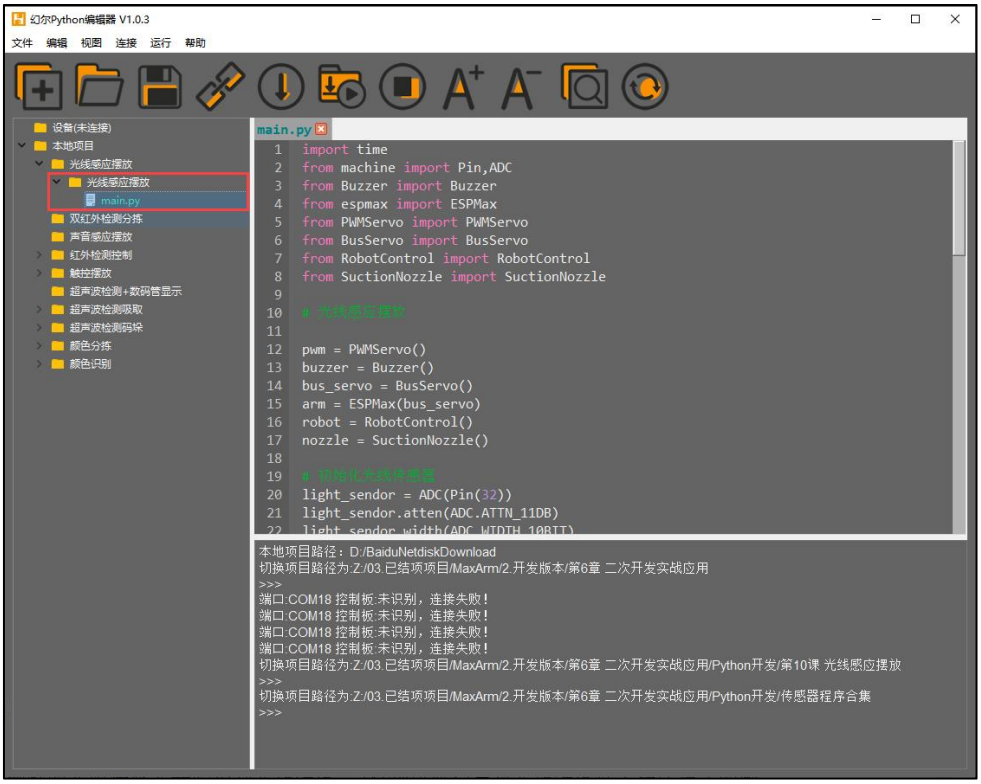
3. Program Download

1) After connecting, change the path of Workspace to “6. Secondary Development / Python Development/Sensor-extension Game”, and select “Program Files”.

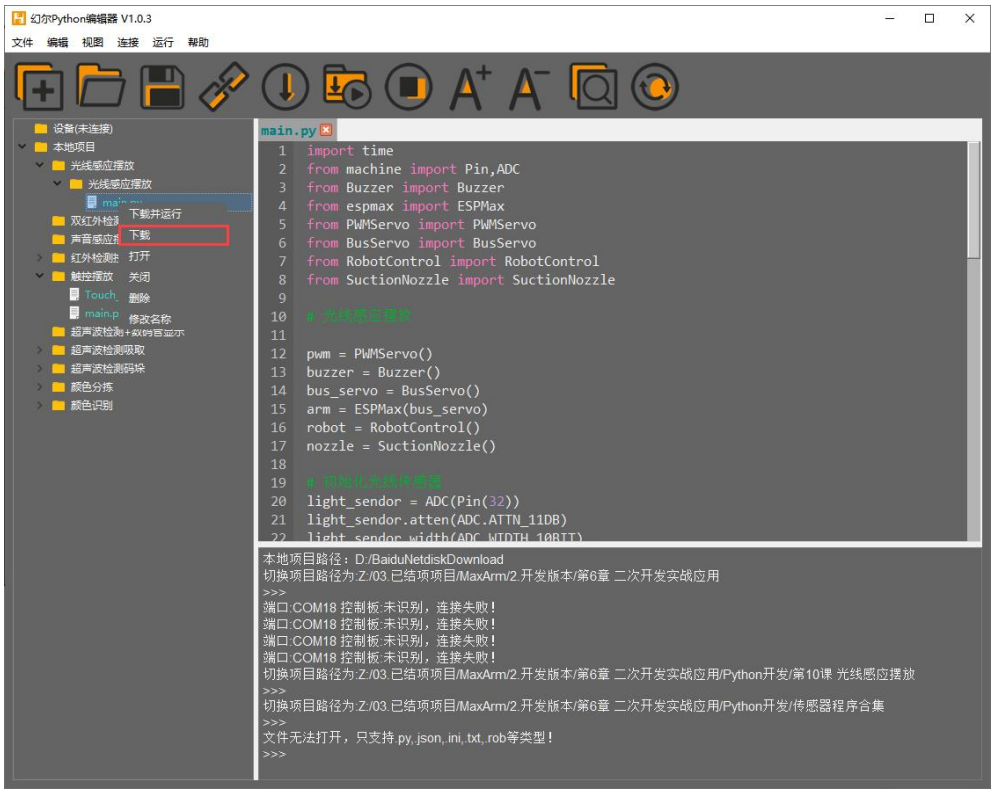



选择您的项目路径
« 第6章 二次开发实战应用 › Python开发 › 传感器延伸开发
3
在 传感器延伸开发 中搜索
组织
新建文件夹
名称
修改日期
类型
大小
快速访问
桌面
下载
文档
图片
9月
Arduino版本
超声波检测吸取
正逆运动学
WPS网盘
此电脑
本地磁盘 (C:)
传感器程序合集
4
2022/9/8 10:41
文件夹
第1课 超声波检测吸取
2022/9/8 10:41
文件夹
第2课 超声波检测码垛
2022/9/8 10:41
文件夹
第3课 触控摆放
2022/9/8 10:41
文件夹
第4课 红外检测控制
2022/9/8 10:41
文件夹
第5课 颜色识别
2022/9/8 10:42
文件夹
第6课 颜色分拣
2022/9/8 10:42
文件夹
文件夹: 传感器程序合集
5
选择文件夹
取消

2) Click the folder “Sound Detection and Placement” and select all the program files.

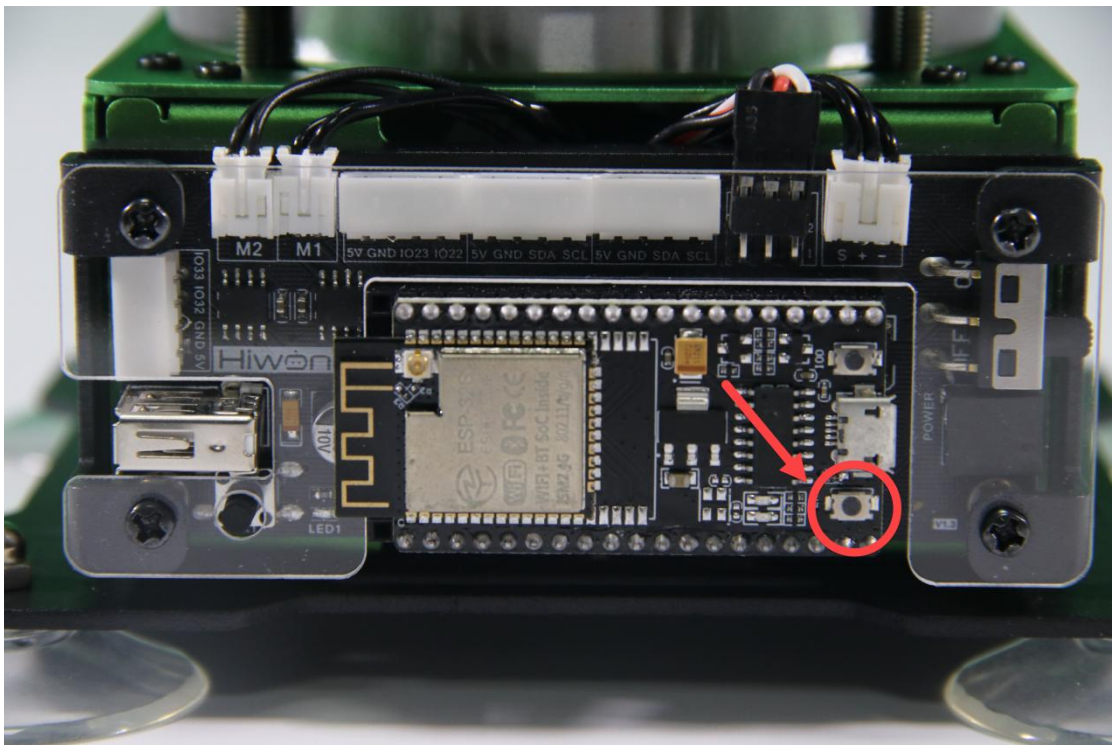


3) Then right click to download all the program files to the controller.



When the terminal prints the prompt as shown in the image below, it means download completed.

4) After downloading, click on the reset icon or press the reset button on ESP32 controller to run program.



4. Project Outcome

When light sensor starts to detecting light, it covered by object resulting in the light intensity is lower than the threshold set in program. At this time, microcontroller will receive a command, and then robot arm will turn on air pump and suck the object and place it to corresponding position. After that, turn off the air pump, reset and wait for the next command.

5. Program Instruction

5.1 Import function library and Initialize

The path of the program file: “6. Secondary Development/Sensor-extension Game/ Python Development/Program Files/ light Detection and Placement/ LightSensor_put/ LightSensor_put.ino”. If the program is modified, you can find a backup file in Appendix.

Before running the program, the related function libraries including buzzer, PWM servo, bus servo, infrared sensor, and air pump are required to imported first.

```
import time
from machine import Pin,ADC
from Buzzer import Buzzer
from espmax import ESPMax
from PWMServo import PWMServo
from BusServo import BusServo
from RobotControl import RobotControl
from SuctionNozzle import SuctionNozzle
```

Then, initialize the library files and robotic arm.

```
pwm = PWMServo()
buzzer = Buzzer()
bus_servo = BusServo()
arm = ESPMax(bus_servo)
robot = RobotControl()
nozzle = SuctionNozzle()

# 初始化光线传感器
light_sendor = ADC(Pin(32))
light_sendor.atten(ADC.ATTN_11DB)
light_sendor.width(ADC.WIDTH_10BIT)
```

5.2 Light Detection

Use analogRead() function to read the analog amount of detected light intensity. When the analog amount reaches the set threshold, the program will place object at position 1. When the next object is detected, it will be placed at position 2. After recognizing and placing three time, game will enter the next round.

```
if __name__ == '__main__':
  arm.go_home() #机械臂复位，回到初始位置
  nozzle.set_angle(0,1000) #吸嘴角度置0
  time.sleep_ms(2000)
  num = 0 #木块计数变量
  angle = [12, 35, 55] #角度补偿
  while True:
    light = light_sendor.read() #光线传感器检测函数
    print(light)
    if light > 900: # 光线传感器被挡
      print('num:',num+1)
      buzzer.setBuzzer(100) #蜂鸣器响一下
      time.sleep_ms(500)
      arm.set_position((0,-165,100),1200) #机械臂移动到吸取位置上方，等待2秒后吸取
      time.sleep_ms(2000)
      arm.set_position((0,-165,86),600) #吸取木块
      nozzle.on() # 打开气泵
      time.sleep_ms(1000)
      arm.set_position((0,-165,180),1000) #机械臂抬起来
      time.sleep_ms(1000)
      arm.set_position((120,-20-60*num,180),1500) # 移动到放置位置上方
      nozzle.set_angle(angle[num],1500) #设置角度补偿，使木块放正
      time.sleep_ms(1500)
      arm.set_position((120,-20-60*num,88),1000) #放置木块
      time.sleep_ms(1200)
      nozzle.off() # 关闭气泵
      arm.set_position((120,-20-60*num,200),1000) # 机械臂抬起来
      time.sleep_ms(1000)
      arm.go_home() #机械臂复位，回到初始位置
      nozzle.set_angle(0,1800) #吸嘴角度置0
      time.sleep_ms(2000)
      num += 1       #木块计数变量加1
      if num >= 3:
        num = 0
        buzzer.setBuzzer(80) #蜂鸣器响一下
        time.sleep_ms(100)
        buzzer.setBuzzer(80) #蜂鸣器响一下
```

5.3 Light Detection Feedback

When cover is detected by sensor, MaxArm will execute the corresponding action.

```
if __name__ == '__main__':
  arm.go_home() #机械臂复位，回到初始位置
  nozzle.set_angle(0,1000) #吸嘴角度置0
  time.sleep_ms(2000)
  num = 0 #木块计数变量
  angle = [12, 35, 55] #角度补偿
  while True:
    light = light_sendor.read() #光线传感器检测函数
    print(light)
    if light > 900: # 光线传感器被挡
      print('num:',num+1)
      buzzer.setBuzzer(100) #蜂鸣器响一下
      time.sleep_ms(500)
      arm.set_position((0,-165,100),1200) #机械臂移动到吸取位置上方，等待2秒后吸取
      time.sleep_ms(2000)
      arm.set_position((0,-165,86),600) #吸取木块
      nozzle.on() # 打开气泵
      time.sleep_ms(1000)
      arm.set_position((0,-165,180),1000) #机械臂抬起来
      time.sleep_ms(1000)
      arm.set_position((120,-20-60*num,180),1500) # 移动到放置位置上方
      nozzle.set_angle(angle[num],1500) #设置角度补偿，使木块放正
      time.sleep_ms(1500)
      arm.set_position((120,-20-60*num,88),1000) #放置木块
      time.sleep_ms(1200)
      nozzle.off() # 关闭气泵
      arm.set_position((120,-20-60*num,200),1000) # 机械臂抬起来
      time.sleep_ms(1000)
      arm.go_home() #机械臂复位，回到初始位置
      nozzle.set_angle(0,1800) #吸嘴角度置0
      time.sleep_ms(2000)
      num += 1       #木块计数变量加1
```

Take setBuzze() function, set_position() function, Pump_on() function and go_home() for example.

The setBuzze() is a function to control buzzer. Call “Buzzer.h” function in the same directory as “InfraredSensor_sorting.ino” program. Fill in the parentheses with the duration time of buzzer, and the unit is ms. The code "setBuzzer(100)" means that the buzzer will respond for 100ms after the infrared sensor detects an object, and then the next set_potision () function will be executed.

```
buzzer.setBuzzer(100)
```

The set_position() is a function to call robotic arm. Call “**ESPMax.h**” function in the same folder as “**LightSensor_put.ino**" program. Fill in the parentheses with the coordinate values and duration time. Take the code "set_position(pos,1500)" for example, pos[0], pos[1], pos[2] is the representative of the robot arm corresponding to the position on the XYZ axis, 1500 is the running time, the unit is milliseconds (ms). After that, the Pump_on() function is executed in the next step.

```
arm.set_position((0,-165,100),1200)
```

The nozzle.on() function is used to turn on air pump while the nozzle.off() function to turn off air pump. Then proceed to execute arm.go_home() function.

```
nozzle.on()
```

```
nozzle.off()
```

```
arm.go_home() #机械臂复位，回到初始位置
```

The go_home() is a function to reset robotic arm. After turning off air pump, this function will be executed to get robot arm back to initial position. After that, the program will execute a function for 2000ms delay and then end the game and wait for the next recognition signal.

6. Adjust Sensitivity

There is an adjustable potentiometer knob on light sensor for adjusting the measuring distance. When performing the related games, if the measurement effect it not good enough, the measurement sensitivity of sensor can be adjusted by adjusting the knob.

It is recommended to use phillips screwdriver. Rotate the knob clockwise, as the figure shown below, to increase the measurement distance; rotate it counterclockwise to decrease the measurement distance.

Please take notice of the following two tips:

1) Since the warm light (incandescent light, sunlight, etc) contains more infrared component, there will be a certain interference the detection result. When using it, special attention should be paid to the surrounding environment. Therefore, you can not adjust the sensitivity alone.

2) The sensitivity adjustment has is set a threshold, which you can regard it as a critical point. If the sensitivity exceeds the critical point, its value will return to the initial state.

3) Sensitivity adjustment needs to be based on the actual needs of the project. It is better to be adjusted to the most suitable conditions.