

智能系统实验1——传感器与通信

一、实验内容

- 连接传感器
- 获取传感器数据
- 发送指令至传感器
- 数据包设计、生成与解析
- 下位机与上位机通信

二、实验要求

- 完成硬件连接
- 使用一种程序设计语言实现数据（指令）获取、发送、接收以及解析程序
- 完成软硬件测试
- 完成实验报告

三、完成时间

- 实验时间：2021.4.24, 2021.5.8
- 检查时间：2021.5.8
- 2021 年 5 月 10 日 23:59 之前提交实验报告

四、实验指导

实验准备工作

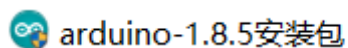
实验环境

准备好python的语言环境，并安装 serial 库，安装命令如下

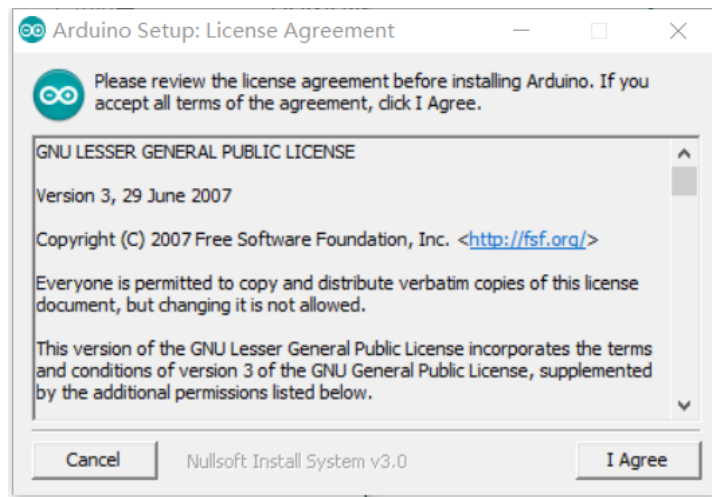
```
pip install pyserial
```

arduino IDE安装

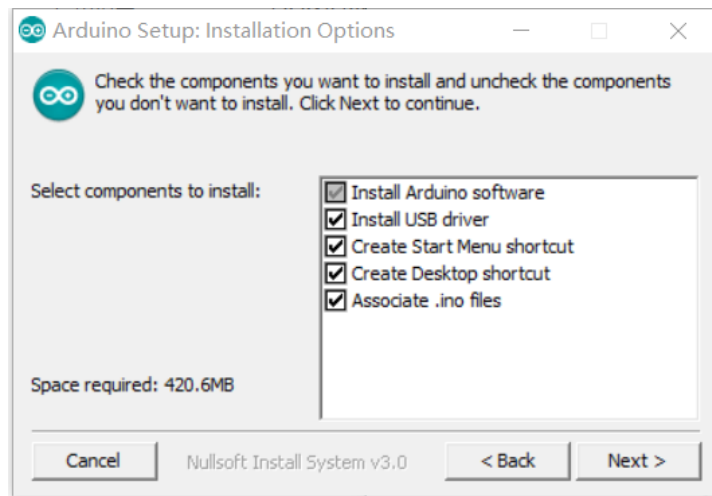
双击实验文件夹下的arduino安装包



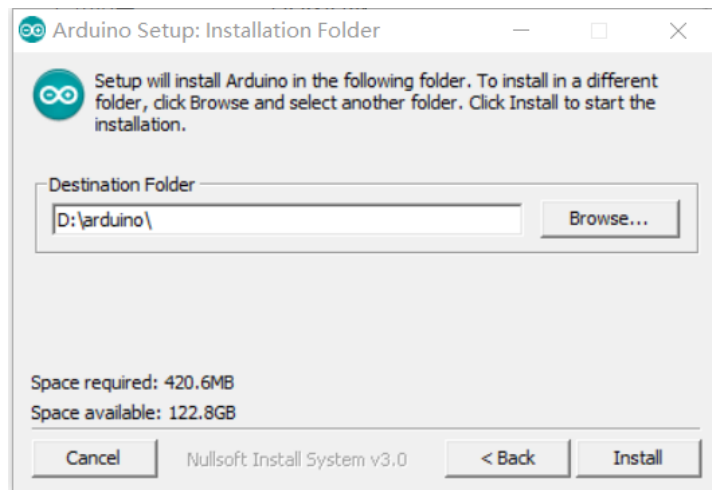
I Agree



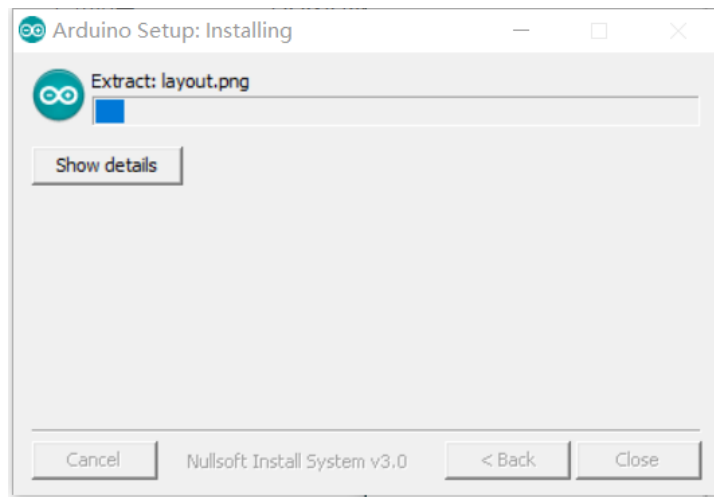
Next



选择自定义的目录D:\arduino\（无则事先创建文件夹），点击Install安装



等待装完成



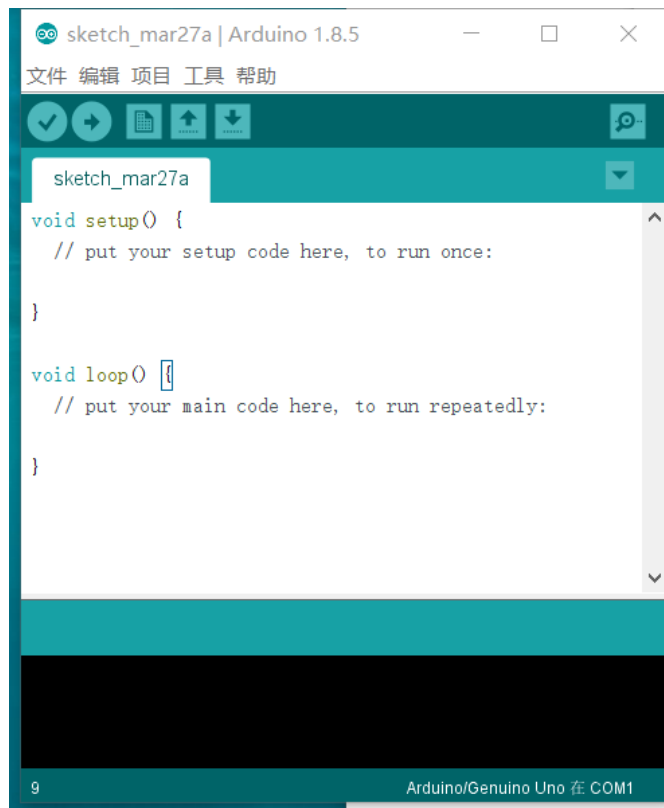
(可能出现如下的三个选择，点击安装)



最后点击close完成完成安装，桌面出现如下图标



双击打开如下模样



红外避障传感器使用及小灯点亮实验

1.实验准备

准备arduino uno r3开发板、交通灯模块、红外避障模块和杜邦线。

2.定义实验效果

前方无障碍物，此时避障传感器为输出接口为低电平，设定LED 为输出接口为低电平灯灭；

将避障模块的红外探头对准障碍物，距离为3 - 30cm厘米以内，此时避障传感器为输出接口为高电平，设定LED 为输出接口为高电平灯亮。

3.定义使用的变量

定义连接交通灯的引脚为13口，定义避障传感器接口为3，定义数字变量val(后续会使用)。

```
int Led = 13;           //定义LED 接口
int buttonpin = 3;      //定义避障传感器接口
int val;                //定义数字变量val
```

4.设置初始化函数

定义setup函数，此函数将在机器运行时运行一次，初始化环境，在其中初始化led 13口为输出，初始化buttonpin 3口为输入。

```
void setup()
{
    pinMode(Led, OUTPUT);           //定义LED 为输出接口
    pinMode(buttonpin, INPUT);      //定义避障传感器为输入接口
}
```

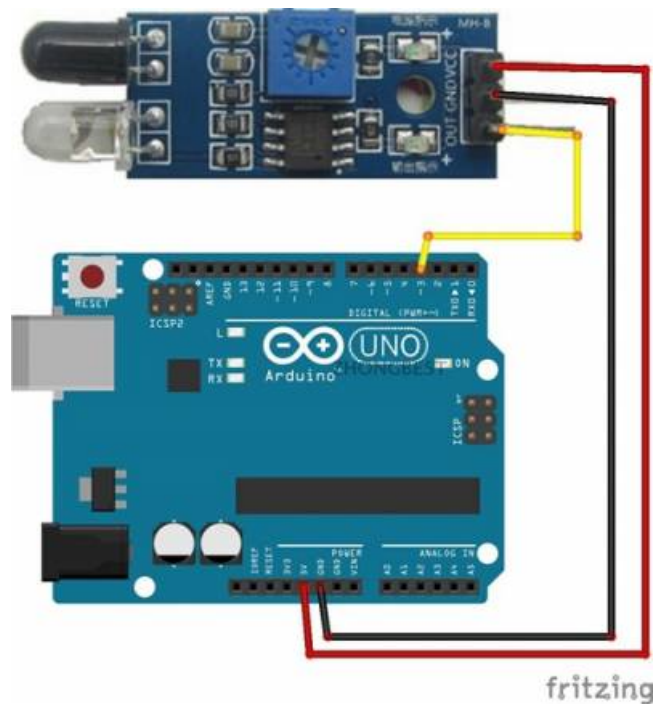
5.定义循环函数

此函数必须为loop，表示机器运行后一直会循环执行此函数。内部逻辑为每次循环首先使用digitalRead函数从buttonpin引脚读出变量，即红外避障传感器的数据，若为低电平，则表示前方无障碍物，这时使用digitalWrite函数向led引脚输出低电平，不亮小灯；反之输出高电平，并且延迟信号1s使得亮明显一些。

```
void loop()
{
    val = digitalRead(buttonpin); //将数字接口3的值读取赋给val
    if (val == LOW)                //当避障传感器检测低电平时，LED 灭
    {
        digitalWrite(Led, LOW);
    }
    else
    {
        digitalWrite(Led, HIGH);
        delay(1000);               //延迟1s，亮灯时间明显
    }
}
```

6.硬件连接

红外避障传感器与arduino连接如下，模块的out连接到引脚13号，电压VCC连接到开发板VCC，GCD接地信号连接到开发板GCD。



交通灯模块将G、Y、R任意一个连接到开发板3号引脚，将GND连接到开发板GND。

最后将arduino开发板连接上电脑，发现开发板通电。

7.烧录

如下图中红色框，第一个位编译代码，第二个为烧录代码到开发板



烧录成功后如下图区域会显示烧录成功提示。



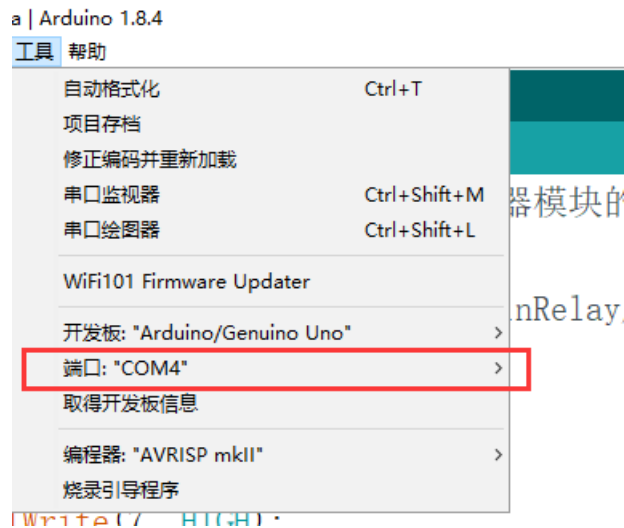
8.源代码

```
/*
 * 将模块与UNO控制器正确连接，切勿接错；
 * 程序效果：首先使得前方无障碍物，此时避障传感器为输出接口为低电平，设定LED 为输出接口为低电平
灯灭；
 *          将避障模块的红外探头对准障碍物，距离为3 - 30cm厘米以内，此时避障传感器为输出接口
为高电平，设定LED 为输出接口为高电平灯亮；
 */
int Led = 13;          //定义LED 接口
int buttonpin = 3;     //定义避障传感器接口
int val;               //定义数字变量val
void setup()
{
    pinMode(Led, OUTPUT);          //定义LED 为输出接口
    pinMode(buttonpin, INPUT);     //定义避障传感器为输入接口
}
void loop()
{
    val = digitalRead(buttonpin); //将数字接口3的值读取赋给val
    if (val == LOW)               //当避障传感器检测低电平时，LED 灭
    {
        digitalWrite(Led, LOW);
    }
    else
    {
        digitalWrite(Led, HIGH);
        delay(1000);           //延迟1s，亮灯时间明显
    }
}
```

Python上位机与下位机通信实验

1.实验准备

接着第一个实验将开发板继续连着电脑，点击如下图按钮查看开发板连接电脑的串口号为COM4或其他。



然后打开python编辑器。

2.实验效果

首先使得前方无障碍物，此时避障传感器为输出接口为低电平，设定LED 为输出接口为低电平灯灭；

将避障模块的红外探头对准障碍物，距离为3 - 30cm厘米以内，此时避障传感器为输出接口为高电平，设定LED 为输出接口为高电平灯亮。

3.编辑上位机程序

首先引入库，然后定义1中知晓的串口号，设置波特率为9600（与arduino 烧录的程序对应），最后连接上串口。

```
import serial

serialPort = "COM4" # 串口
baudRate = 9600 # 波特率
ser = serial.Serial(serialPort, baudRate, timeout=0.5) # 连接串口
print("参数设置: 串口=%s , 波特率=%d" % (serialPort, baudRate))
```

定义bytes类型的0,1ascii码数据，方便串口传输。

```
demo0 = b"0" # 将0转换为bytes类型的ASCII码方便发送
demo1 = b"1" # 同理
```

定义循环，每次循环首先接收下位机传输的字符串，然后打印出来，用户再输入0或1，若x==1，则发送b"1",否则发送b"0"给下位机，1表示开启灯，0为不开灯。

最后关闭串口。

```
while 1:
    str = ser.readline() # 接收下位机上传的
    print(str)
    x = int(input("请输入0 or 1: "))
    if(x == 1):
        ser.write(demo1) # 发送字节1
    else:
        ser.write(demo0) # 发送字节0

ser.close()
```

4.编辑下位机程序

在第一个实验代码中基础上，首先增加chr变量定义

```
int Led = 13;           //定义LED 接口
int buttonpin = 3;      //定义避障传感器接口
int val;                //定义避障传感器接口的数字变量val
char chr;               //定义上位机的输入字符信号
```

setup()函数不变。

loop函数中，首先接收传感器型号，若为低电平则向上位机传输NO的字符串，表示无障碍，否则传输YES表示有障碍；然后接收上位机的字符，若为0则不开灯，为1则开灯1s。

```
void loop()
{
    // 接收传感器信号然后发送上位机
    val = digitalRead(buttonpin); //将数字接口3的值读取赋给val

    if (val == LOW)                //当避障传感器检测低电平时，LED 灭
    {
        Serial.print("NO\n");
    }
    else
    {
        Serial.print("YES\n");
    }

    // 接收上位机信号控制小灯开关
    chr = Serial.read();
    if (chr == '0')                //当避障传感器检测低电平时，LED 灭
    {
        digitalWrite(Led, LOW);
    }
    else
    {
        digitalWrite(Led, HIGH);
        delay(1000);              //延迟1s，亮灯时间明显
    }
}
```

5.物理连接

此处连接和第一个实验相同。

6.源码

python

```
"""
@name: lab1.py
@Describe: 通过python的串口库与下位机单片机通信
"""

import serial

serialPort = "COM4" # 串口
```



```

baudRate = 9600 # 波特率
ser = serial.Serial(serialPort, baudRate, timeout=0.5) # 连接串口
print("参数设置: 串口=%s , 波特率=%d" % (serialPort, baudRate))

demo0 = b"0" # 将0转换为bytes类型的ASCII码方便发送
demo1 = b"1" # 同理

while 1:
    str = ser.readline() # 接收下位机上传的
    print(str)
    x = int(input("请输入0 or 1: "))
    if(x == 1):
        ser.write(demo1) # 发送字节1
    else:
        ser.write(demo0) # 发送字节0

ser.close()

```

arduino

```

/*
 * 将模块与UNO控制器正确连接，切勿接错；
 * 程序效果：首先使得前方无障碍物，此时避障传感器为输出接口为低电平，将“NO”发送给上位机；
 *           将避障模块的红外探头对准障碍物，距离为3 - 30cm厘米以内，将“YES”发送给上位机；
 *           然后接受上位机信号，若为0，则设定LED 为输出接口为低电平灯灭，否则高电平灯亮
 */

int Led = 13; //定义LED 接口
int buttonpin = 3; //定义避障传感器接口
int val; //定义避障传感器接口的数字变量val
char chr; //定义上位机的输入字符信号
void setup()
{
    pinMode(Led, OUTPUT); //定义LED 为输出接口
    pinMode(buttonpin, INPUT); //定义避障传感器为输出接口
    Serial.begin(9600); //连接上位机，波特率为9600
}
void loop()
{
    // 接收传感器信号然后发送给上位机
    val = digitalRead(buttonpin); //将数字接口3的值读取赋给val

    if (val == LOW) //当避障传感器检测低电平时，LED 灭
    {
        Serial.print("NO\n");
    }
    else
    {
        Serial.print("YES\n");
    }

    // 接收上位机信号控制小灯开关
    chr = Serial.read();
    if (chr == '0') //当避障传感器检测低电平时，LED 灭
    {
        digitalWrite(Led, LOW);
    }
}

```

```
else
{
    digitalWrite(Led, HIGH);
    delay(1000);           //延迟1s，亮灯时间明显
}
}
```