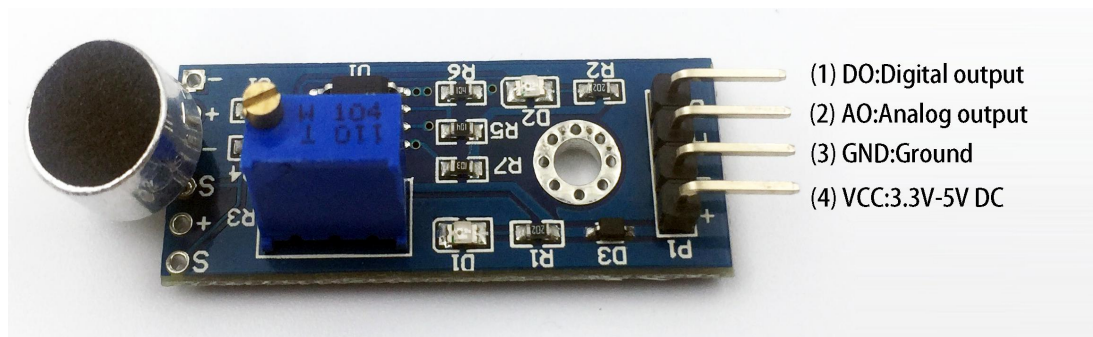# Lesson 27 Sound Sensor Module

Introduction

In this lesson, you will learn how to use a sound sensor module. This module has two outputs:

AO: Analog output, real-time output voltage signal of the microphone.

DO: Digital output, when the intensity of the sound reaches a certain threshold, the output is a high or low level signal. The threshold sensitivity can be achieved by adjusting the potentiometer.

To make sure the microphone can detect your voice normally, please try to change its sensitivity by turning the blue precise potentiometer on the module. Given its preciseness, it takes at least 10 circles for you to get some response.

(1) DO:Digital output
(2) AO:Analog output
(3) GND:Ground
(4) VCC:3.3V-5V DC

Hardware Required

✓ 1 * RuiiGuu Uno R3

✓ 1 * Sound sensor module

✓ 4 * F-M Jumper Wires

✓ 1 * 5mm GREEN LED

Principle

Microphone

This module allows you to detect when the sound has exceeded a set point you select. Sound is detected via a microphone and fed into an LM393 op-amp.

The sound level set point is adjusted via an on board potentiometer. When the sound level exceeds the set point, an LED on the module is illuminated and the output is sent low.

Uses for the Arduino Sound Detector Given that this device measures whether or not a sound has exceeded a threshold,   you're basically left with determining what it is you want to do.   What I mean by this is that you can do something when it is quiet and/or you can do something when it is loud.   For example:

You could detect whether or not a motor is running.

You could set a threshold on pump sound so that you know whether or not there is cavitation.

In the presence of no sound,   you might want to create an ambiance by turning on music.

In the presence of no sound and no motion, you may go into an energy savings mode and turn off the lights.

Arduino Sound Detection Sensor Pin Outs

The image and table below detail the controls, pin outs, and other key components.

When referring sensitivity, I mean this:

When less sensitive,   it takes more sound to trigger the device

When more sensitive, it takes less sound to trigger the device

Code interpretation

This module provides two signal output modes, for which we wrote two codes: digital_signal_output and analog_signal_output. The code of digital_signal_output works when the voice reaches a certain value, it will trigger a digital signal and the dig #11 pin on Arduino will output a high level and the indicator L will be lit up at the same time. This triggering value may be changed according to the sensitivity adjustment method mentioned above. The code of analog_signal_output will read the analog value of the module and directly display it on the serial monitor, likewise, this value can also be changed according to the sensitivity adjustment method mentioned above.

A：digital_signal_Code

int Led=13;**//define LED port**

int buttonpin=3; **//define switch port**

;int   val;//define digital variable val

void   setup()

{

pinMode(Led,OUTPUT);**//define LED as a output port**

pinMode(buttonpin,INPUT);**//define switch as a output port**

}

void   loop()

{ val=digitalRead(buttonpin);**//read the value of the digital interface 3 assigned to val**

```
if(val==HIGH)//when the switch sensor have signal, LED blink

{

digitalWrite(Led,HIGH);

}

else

{

digitalWrite(Led,LOW);

}

}
```

B：analog_signal_Code

```
int   sensorPin   =   A0;     // select the input pin for the
potentiometer //

int   ledPin   =   13;        // select the pin for the LED

int   sensorValue =   0;     // variable to   store   the value   coming
from   the sensor//


void setup()

{

pinMode(ledPin,OUTPUT);

Serial.begin(9600);

}

void loop(){
```
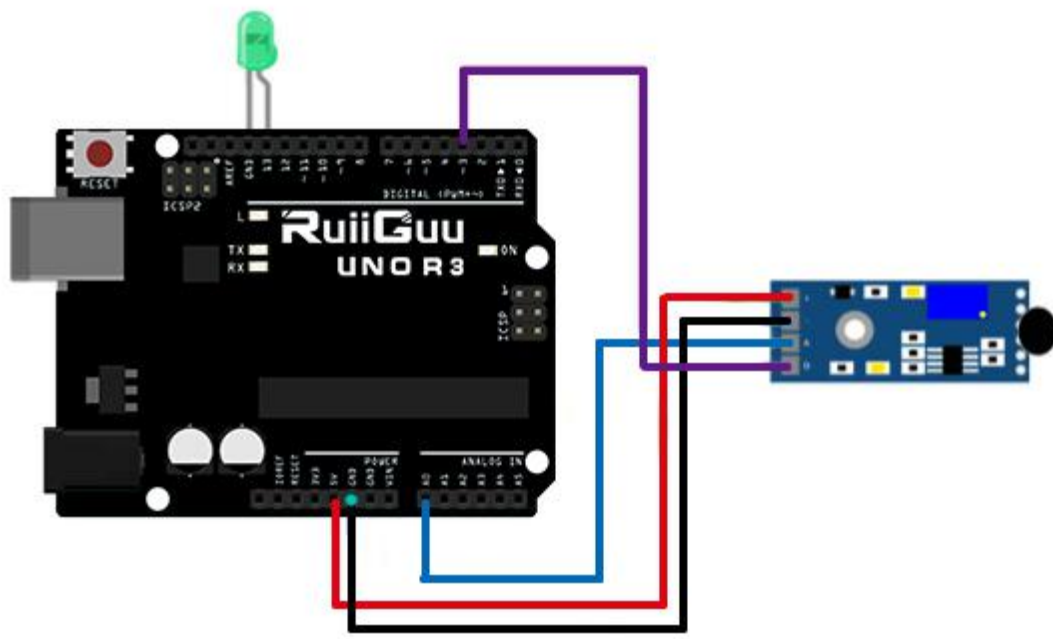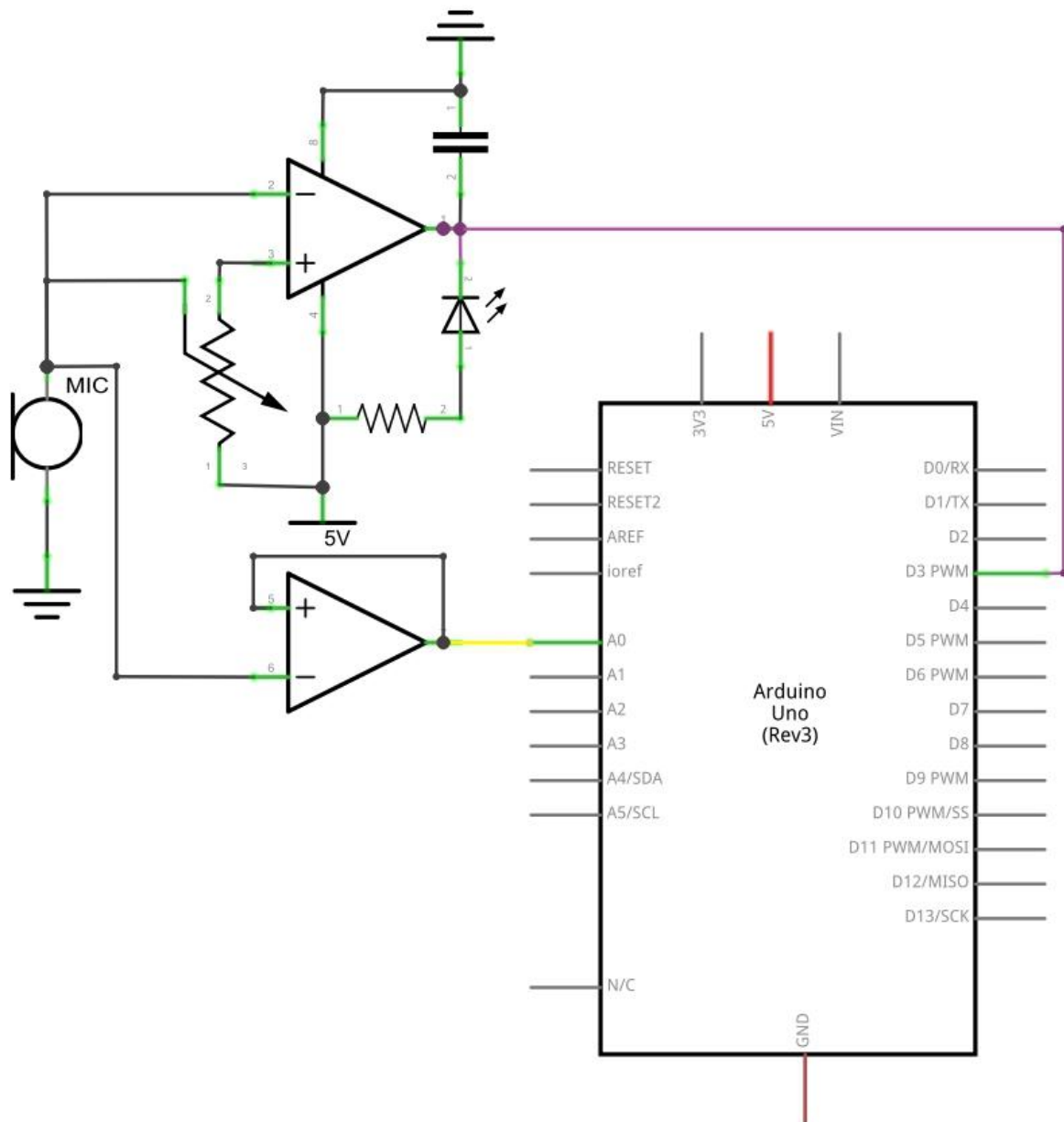
```
sensorValue =    analogRead(sensorPin);

digitalWrite(ledPin,    HIGH);

delay(sensorValue);

digitalWrite(ledPin,    LOW);

delay(sensorValue);

Serial.println(sensorValue,    DEC);

}
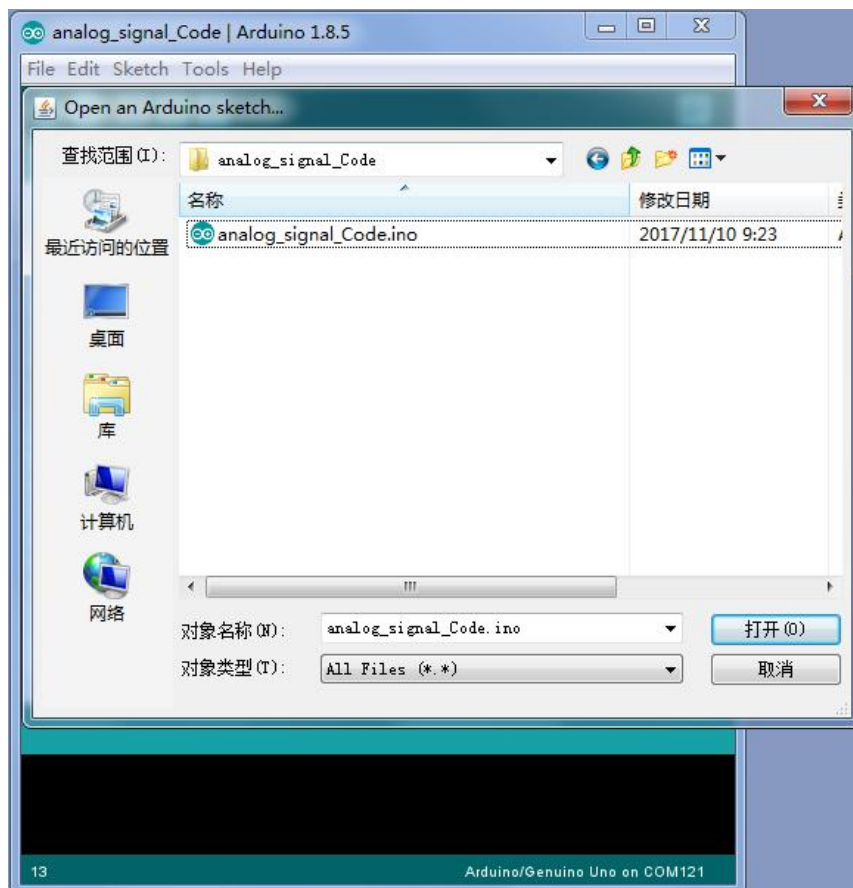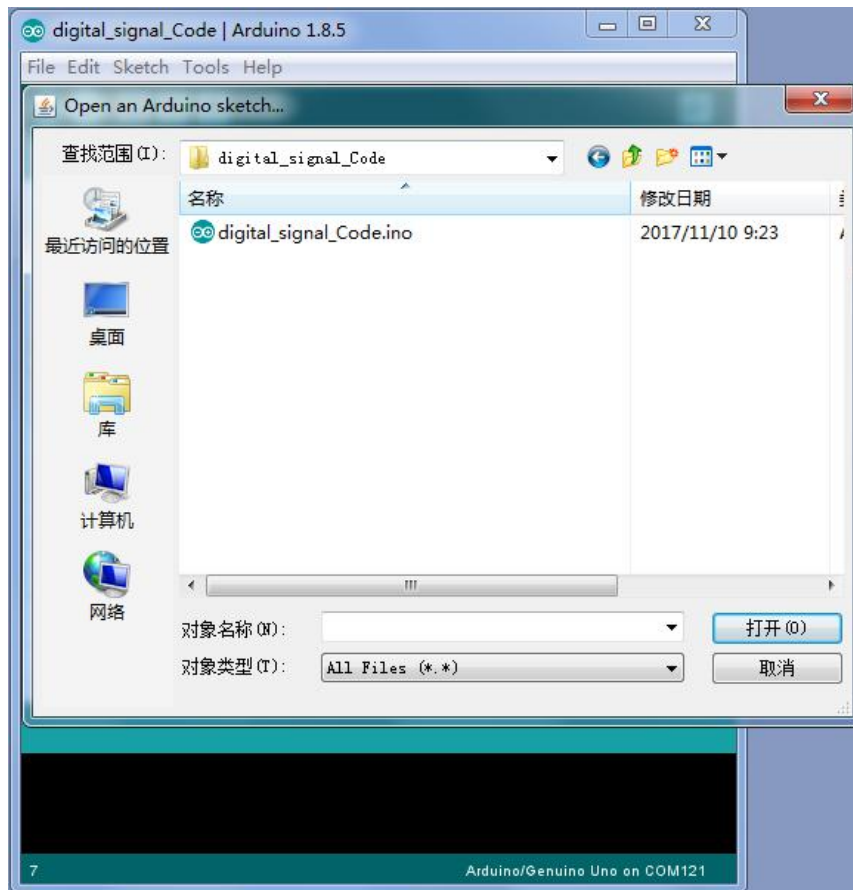```

# Experimental Procedures

## Step 1:Build the circuit



## Schematic Diagram

**Step 2: Open the code:**

**digital_signal_Code** **or** **analog_signal_Code**

**Step 3: Attach Arduino UNO R3 board to your computer via USB cable and check that the 'Board Type' and 'Serial Port' are set correctly.**
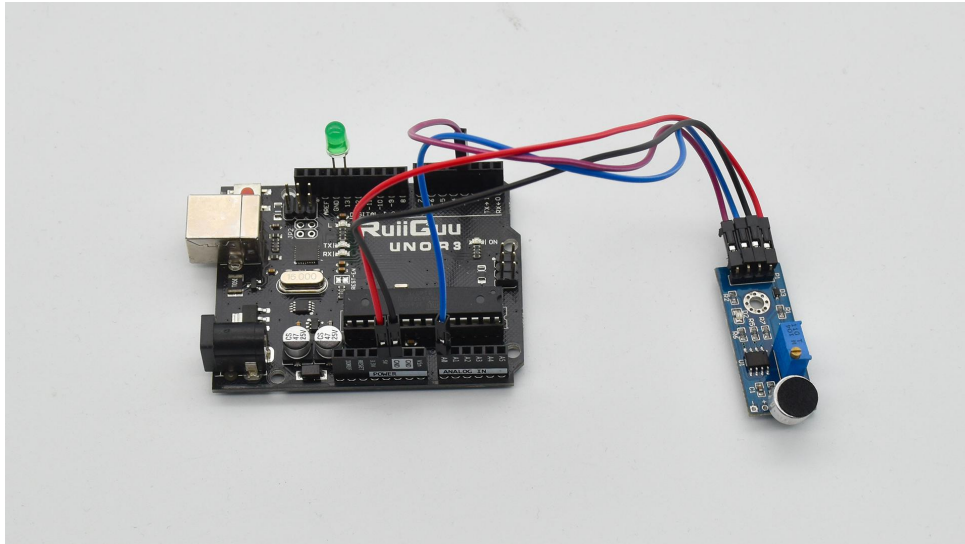
**Step 4: Upload the code to the RuiiGuu UNO R3 board.**

**Then, you can see that the louder the sound, the brighter the LED.**

**You can also open the Serial Monitor to see the data changes with sound as below:**

**(How to use the Serial Monitor is introduced in details in Lesson 0 Preface)**
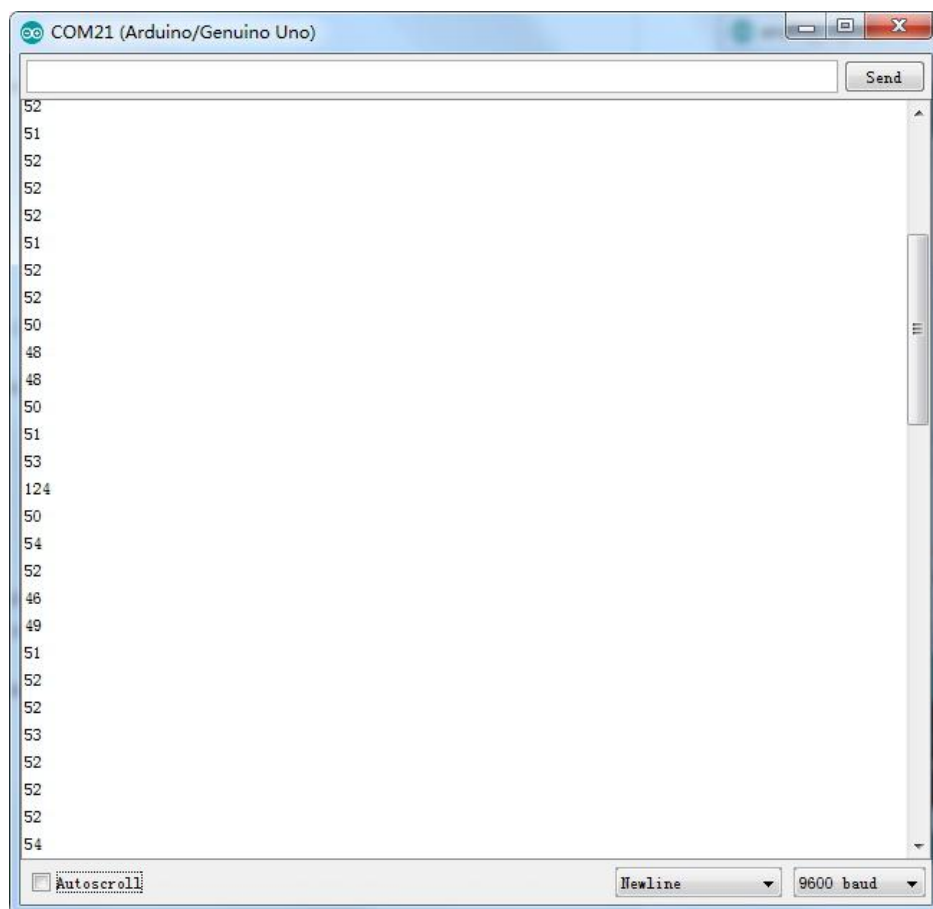
If it isn't working, make sure you have assembled the circuit correctly, verified and uploaded the code to your board. For how to upload the code and install the library, check Lesson 0 Preface.