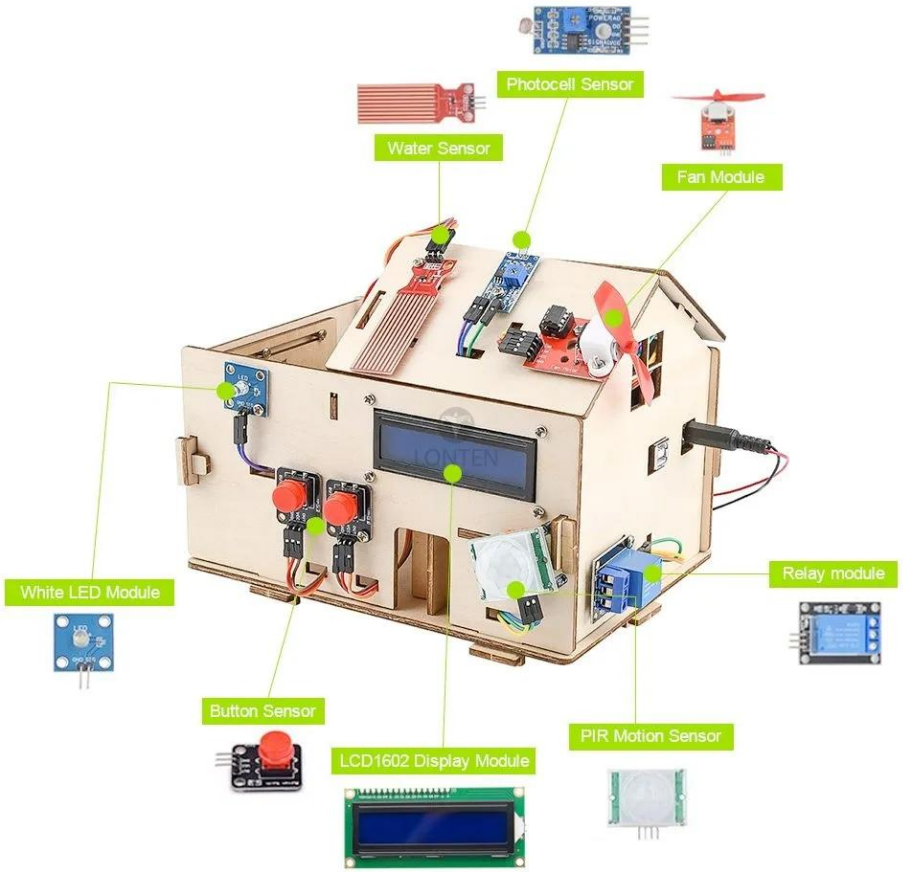
LROBRUYA

Smart Home Learning Kit

LROBRUYA


Photocell Sensor
Water Sensor
Fan Module
White LED Module
Relay module
Button Sensor
LCD1602 Display Module
PIR Motion Sensor

Content

Preface

Company Profile

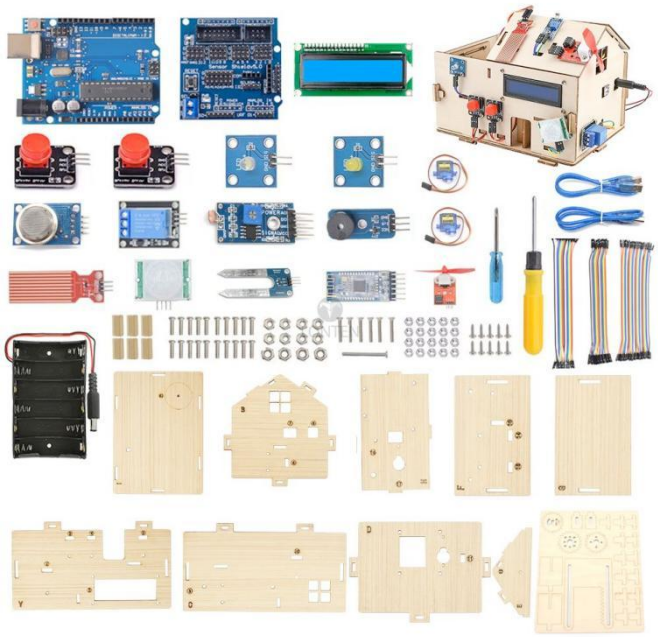
Founded in 2014, Shenzhen Lonten Technology Co., Ltd. focuses on the design, research production of Electronics Module for robotics related products. Consisting of professional researchers and skilled engineers, our R&D team constantly strives for creative function and excellent user experience. The company's R&D investments on arduino kits raspberry pi kits, as well as 3D printer and robots that back up STEAM education.

Customer Service

Our self-owned factory is certificated with BSCI and SO, covering an area of 5,000 square meters, and achieving an annual production capacity of over 10,000 units. Our products are all certified to CE, FCC, and ROHS standards, have exported to more than 100 countries including, but not limited to France, the United States of America, Australia, Russia, the United Kingdom, Germany, Singapore, Egypt, and India, bringing technological innovation to all walks of life.

LROBRUYA

Packing list



How to Install Arduino IDE

Introduction

The Arduino Integrated Development Environment (IDE) is the software side of the Arduino platform.

In this Project, you will learn how to setup your computer to use Arduino and how to set about the Projects that follow.

The Arduino software that you will use to program your Arduino is available for Windows, Mac and Linux. The installation process is different for all three platforms and unfortunately there is a certain amount of manual work to install the software.

STEP 1: Go to https://www.arduino.cc/en/software.

The version available at this website is usually the latest version, and the actual version may be newer than the version in the picture.

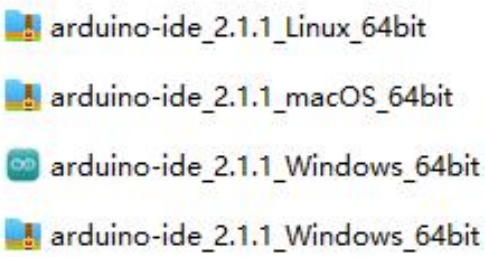
STEP2：Download the development software that is compatible with the operating.

system of your computer. Take Windows as an example here.

Click Windows Win 10 and newer,64 bits.

Click JUST DOWNLOAD.

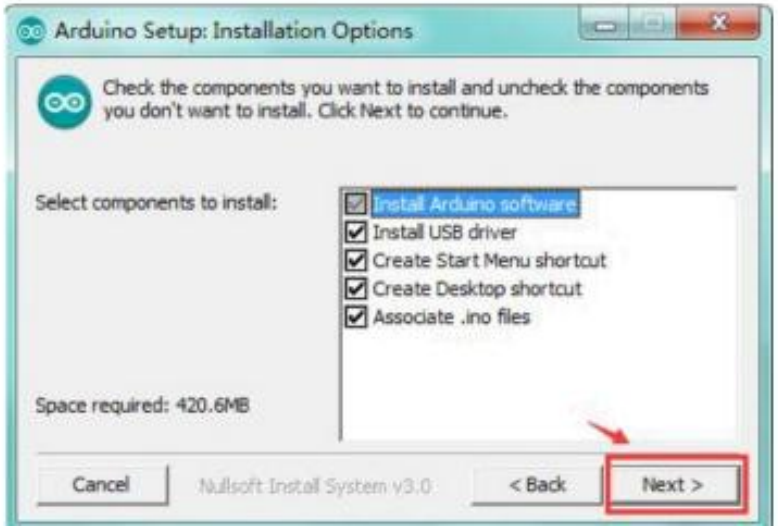
Also version 2.1.1 is available in the material we provided, and the versions of our materials are the latest versions when this course was made.



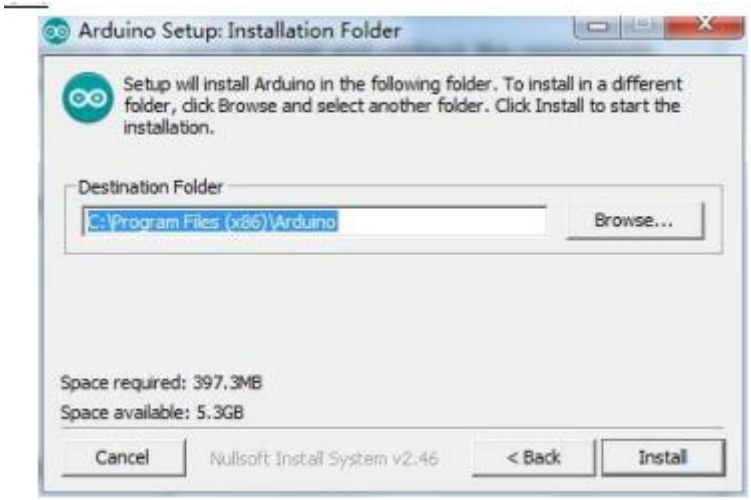
Installing Arduino (Windows)

Install Arduino with the exe. Installation package.

Click I Agree to see the following interface.

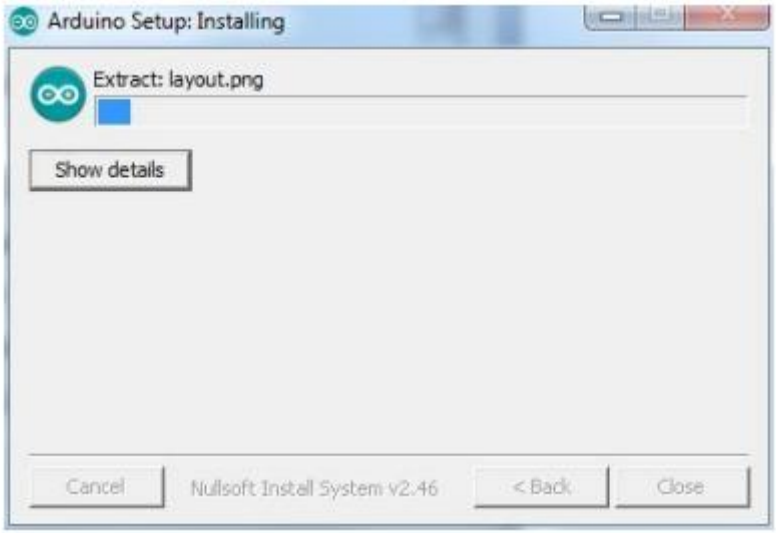

Arduino Setup: Installation Options
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
Install Arduino software
Install USB driver
Create Start Menu shortcut
Create Desktop shortcut
Associate .ino files
Space required: 420.6MB
Cancel
Nullsoft Install System v3.0
< Back
Next >

Click Next



You can press Browse… to choose an installation path or directly type in the directory you want.

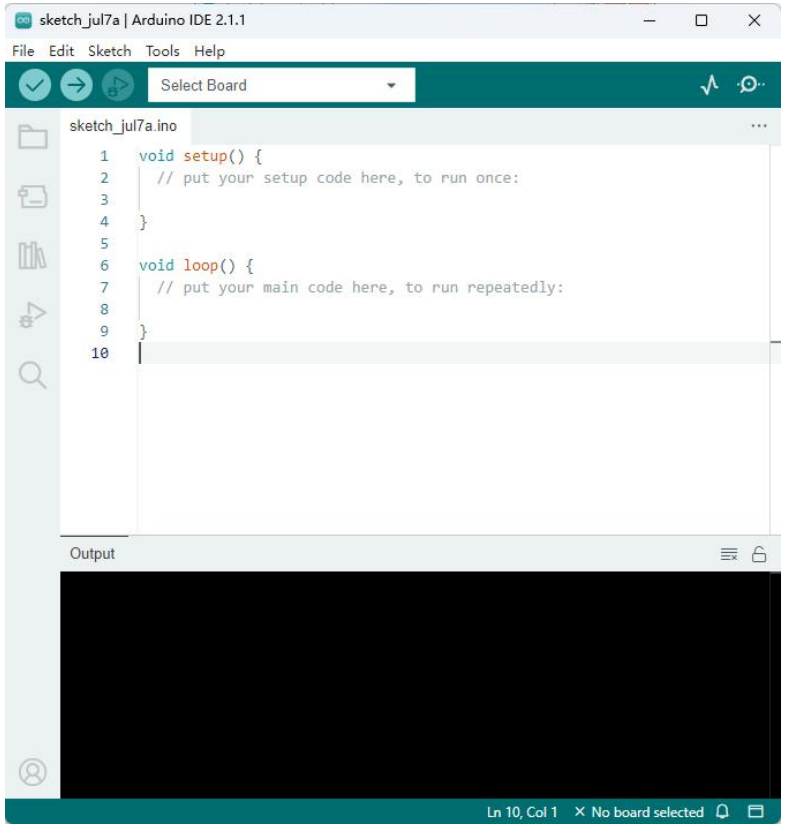
Click Install to initiate installation



Finally, the following interface appears, click Install to finish the installation.

Next, the following icon appears on the desktop

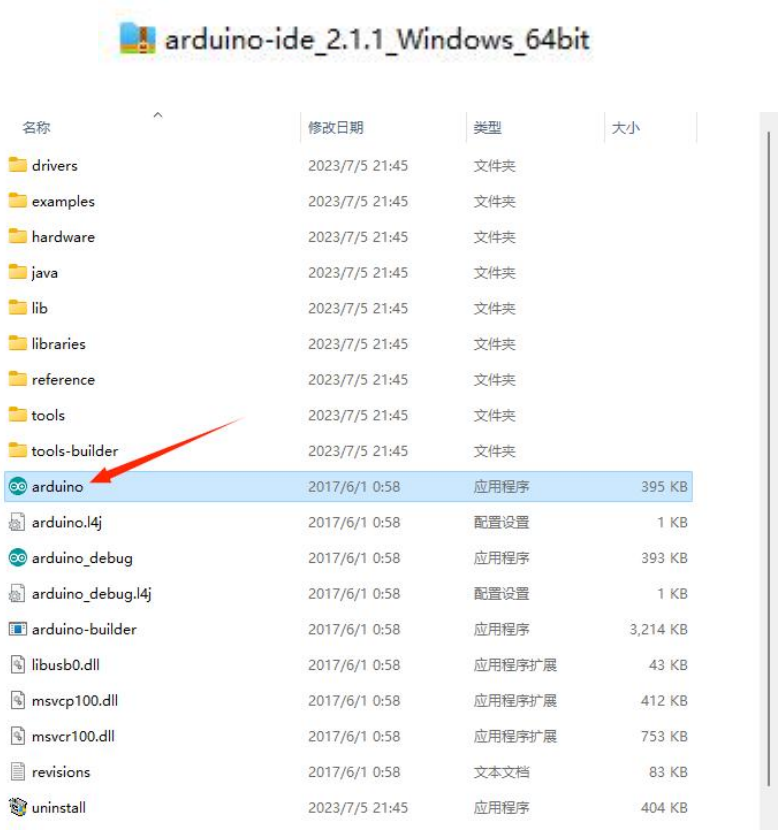
Double-click to enter the desired development environment



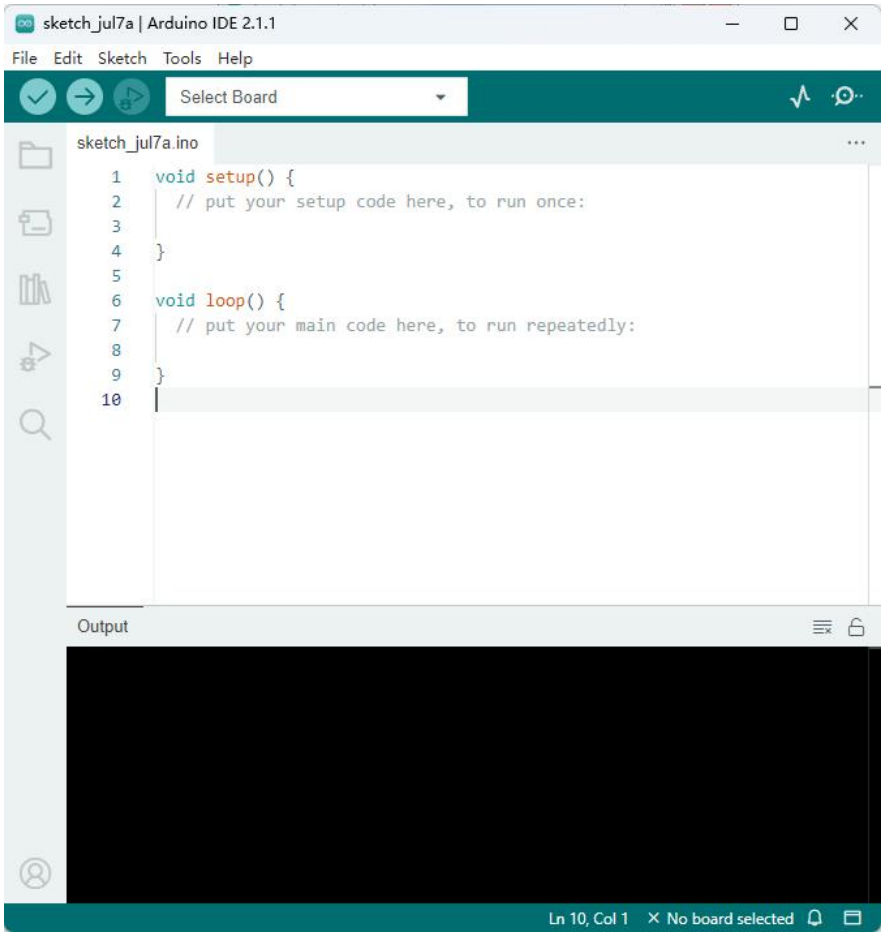
You may directly choose the installation package for installation and skip the contents below and jump to the next section.

But if you want to learn some methods other than the installation package, please continue to read the section.

Unzip the zip file downloaded, Double-click to open the program and enter the desired development environment.

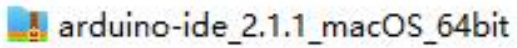


LROBRUYA


sketch_jul7a | Arduino IDE 2.1.1
File Edit Sketch Tools Help
Select Board
sketch_jul7a.ino
1 void setup() {
2 // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7 // put your main code here, to run repeatedly:
8
9 }
10
Output
Ln 10, Col 1
No board selected

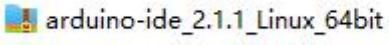
Installing Arduino (Mac OS X)

Download and Unzip the zip file, double click the Arduino.app to enter Arduino IDE; the system will ask you to install Java runtime library if you don't have it in your computer. Once the installation is complete you can run the Arduino IDE.



Installing Arduino (Linux)

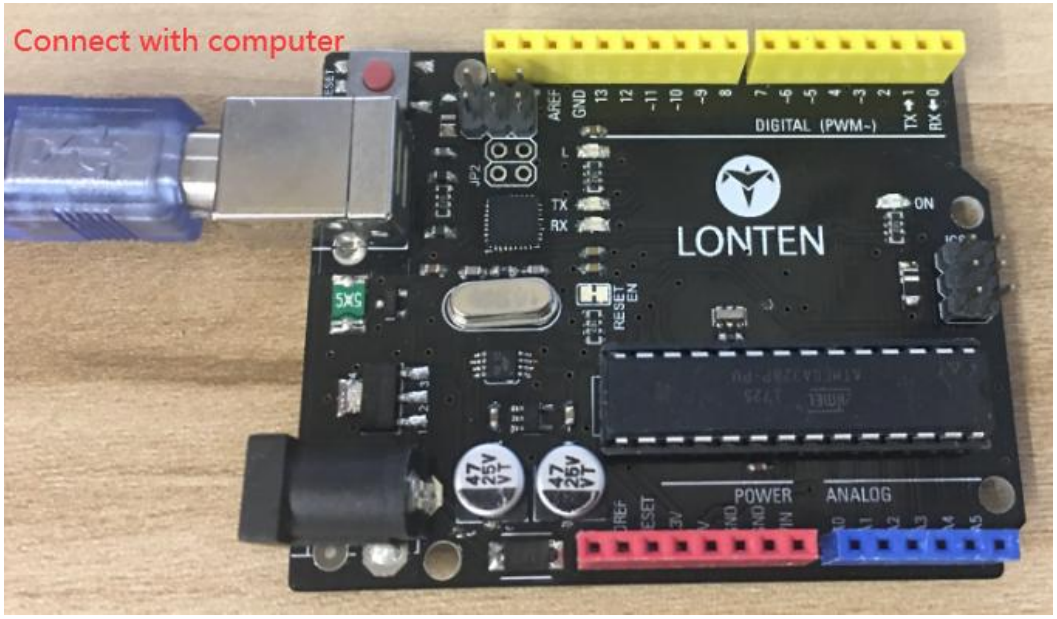
You will have to use the make install command. If you are using the Ubuntu system, it is recommended to install Arduino IDE from the software center of Ubuntu.



How to Install Arduino Driver

For Windows

Arduino UNO R3 board



Serial communication interface: D0 is RX, D1 is TX

PWM interface (pulse width modulation): D3 D5 D6 D9 D10 D11

External interrupt interface: D2 (interrupt 0) and D3 (interrupt 1)

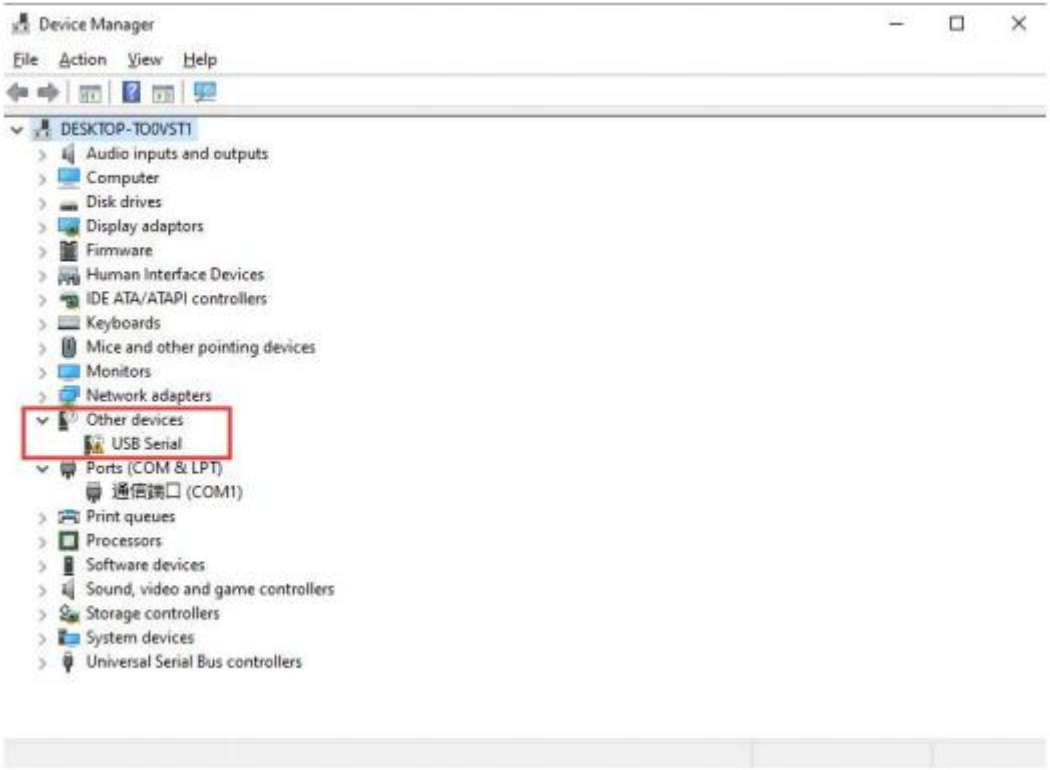
SPI communication interface: D10 is SS, D11 is MOSI, D12 is MISO, D13 is SCK

IIC communication port: A4 is SDA, A5 is SCL

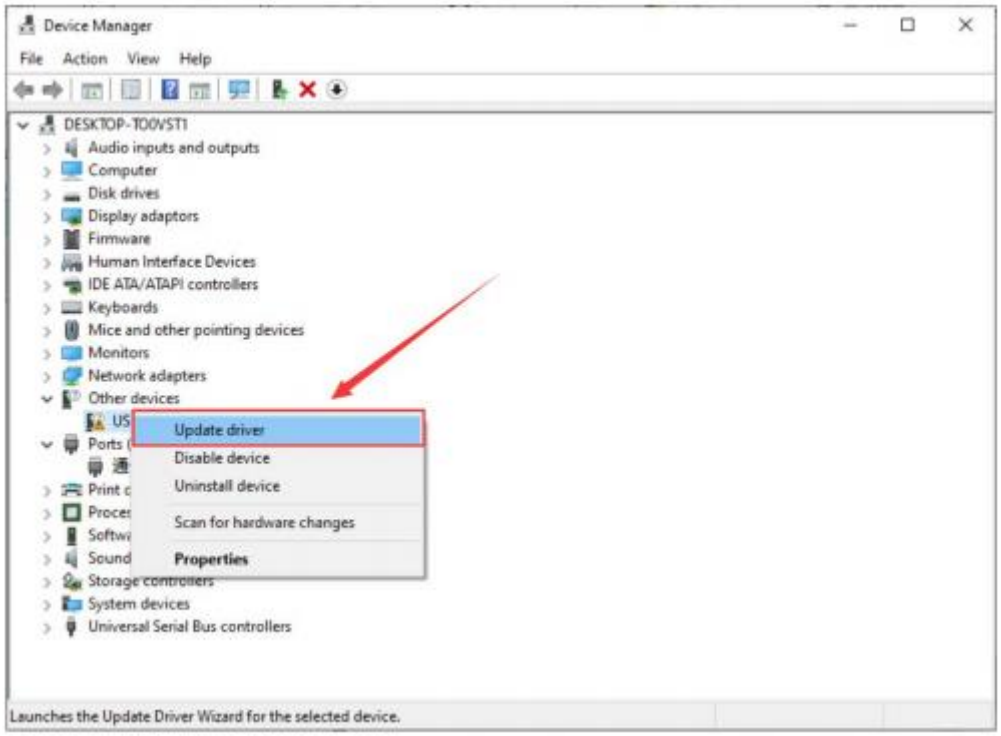
In different systems, the driver installation is similar. Here we start to install the driver on the Win10 system. You can find the “USB_Drive_CH341_3_1” folder in the information we provide, this is the driver file we want to install.

Plug one end of your USB cable into the Arduino UNO R3 Board and the other into a USB socket on your computer.

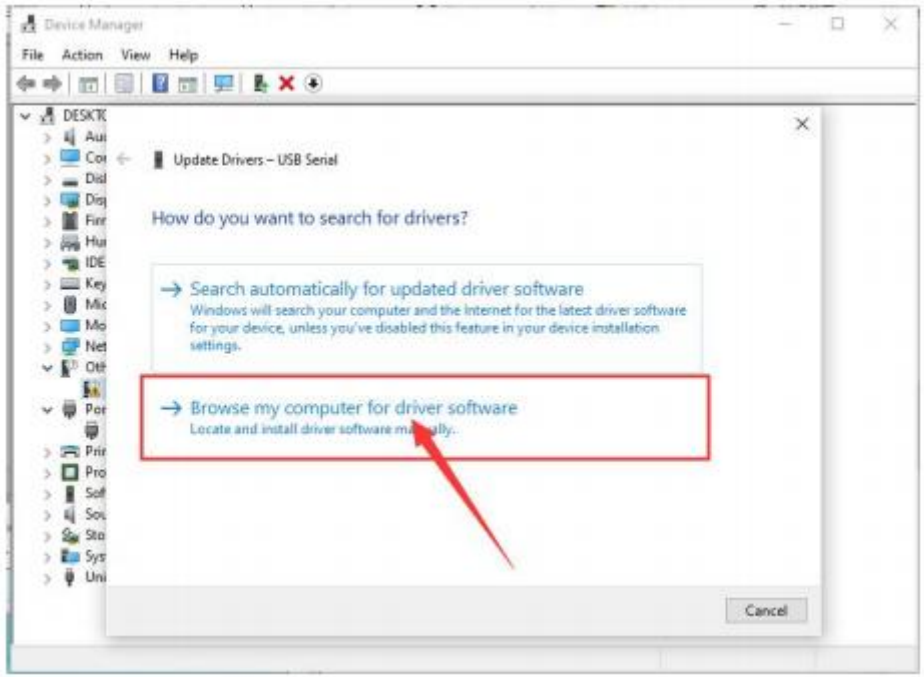
When you connect the Arduino UNOR3 Board to your computer at the first time, right click your “My Computer”—>for “Properties”—>click the “Device manager”, under Other devices, you should see the “USB-Serial” or “Unknown device ”.Or you can search for "devi" in your computer, or you can open the device manager of your computer.



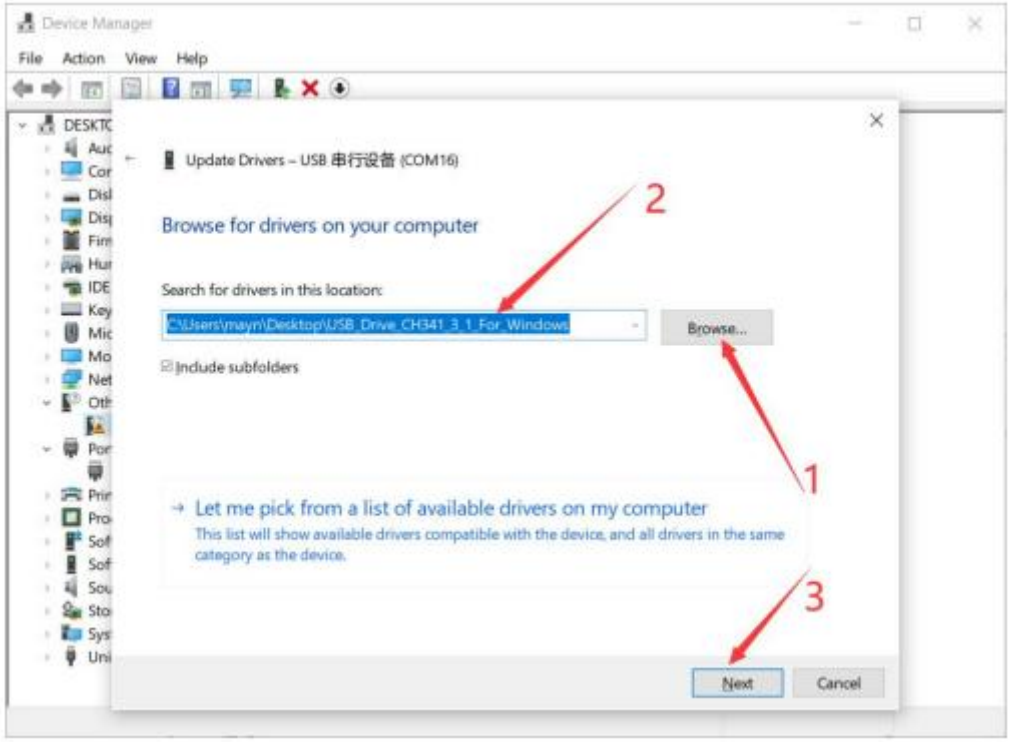
Then right-click on the device and select the top menu option (Update Driver Software...) shown as the figure below.



Then it will be prompted to either “Search Automatically for updated driver software” or “Browse my computer for driver software”. Shown as below. In this page, select “Browse my computer for driver software”.

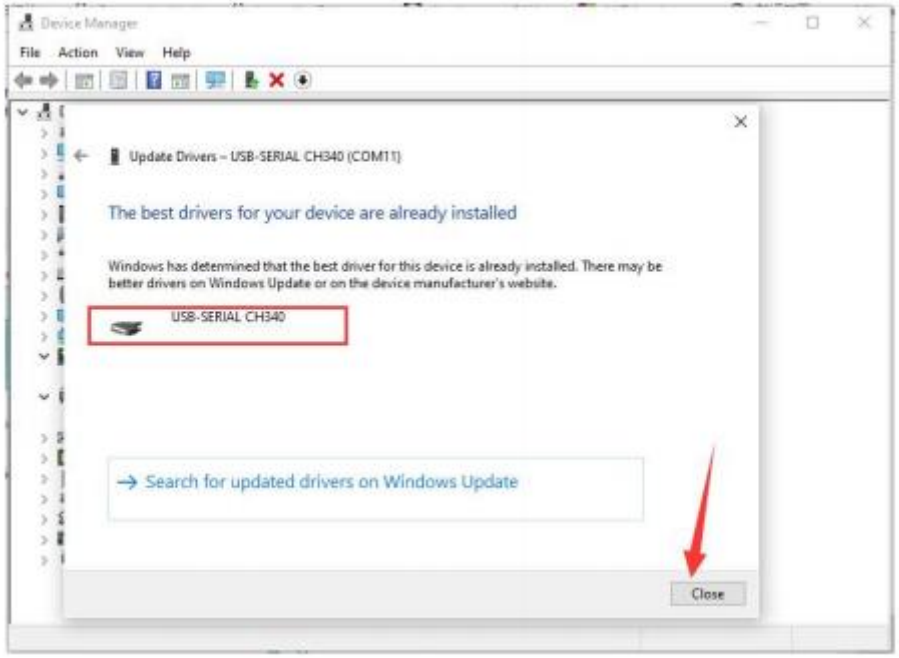


After that, select the browse option and navigate to the drive folder "USB_Drive_CH341_3_1", which can be found in the information we provide.(Note that the file path selects the location of the .For example, I store this driver file on the computer desktop, so the file path I choose is C:\Users\mayn\Desktop\USB_Drive_CH341_3_1_For_Windows)

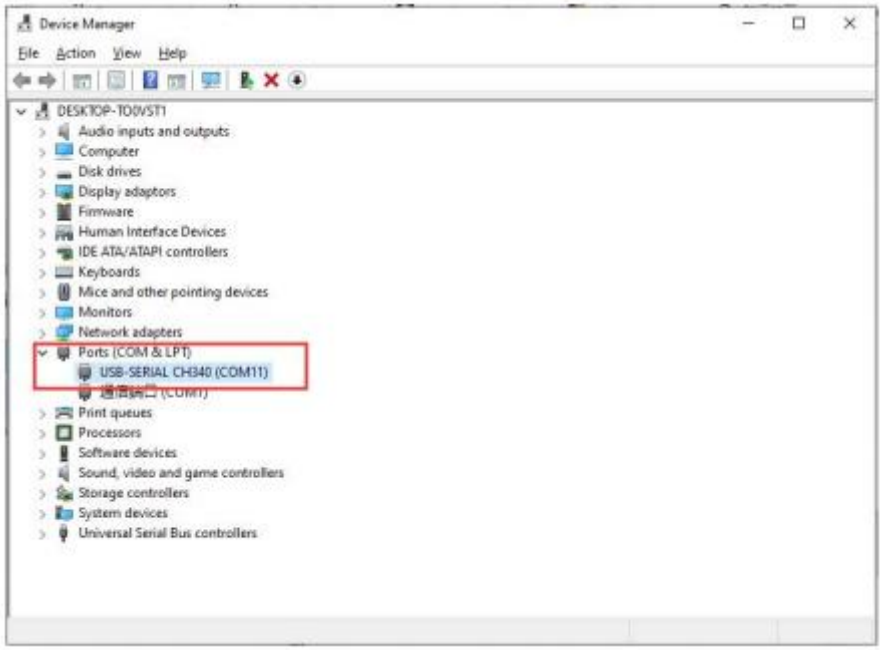


Once the software has been installed, you will get a confirmation message.

Installation completed, click “Close”.



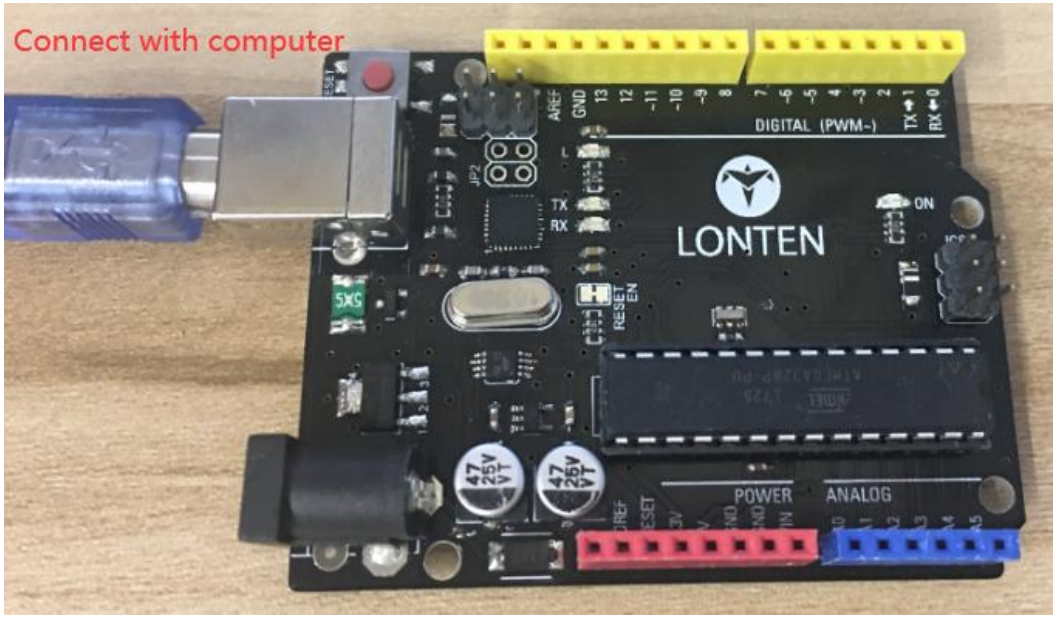
Up to now, the driver is installed well. Then you can right click “My Computer”—>for “Properties”—>click the “Device manager”, you should see the device as the figure shown below. Or you can search for "devi" in your computer, or you can open the device manager of your computer.



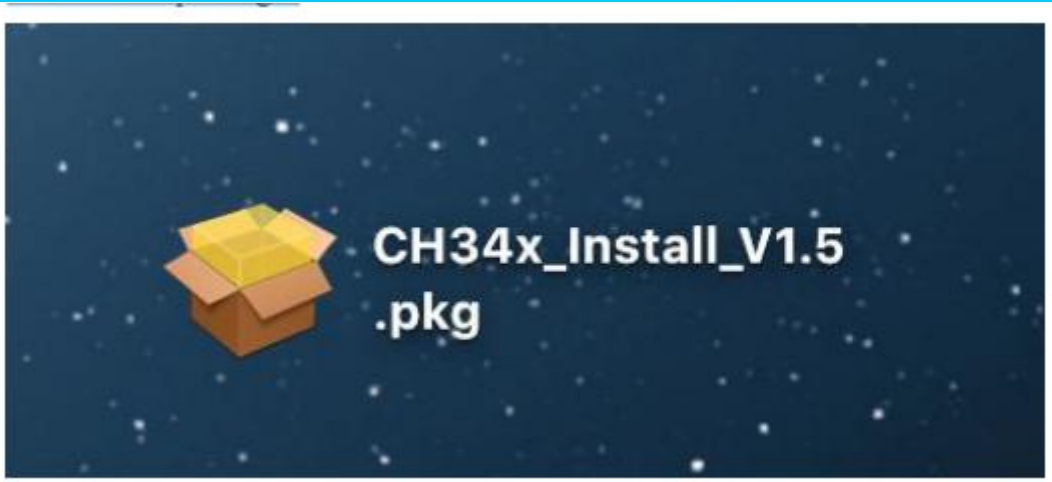
For MAC System

Arduino UNO R3 board

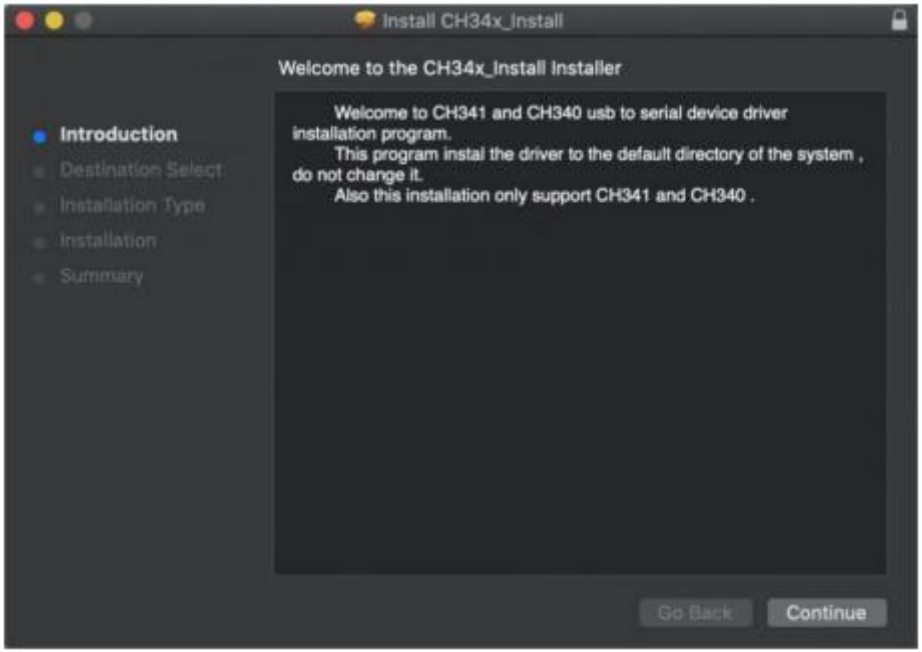
Plug one end of your USB cable into the Arduino UNO R3 Board and the other into a USB socket on your computer.



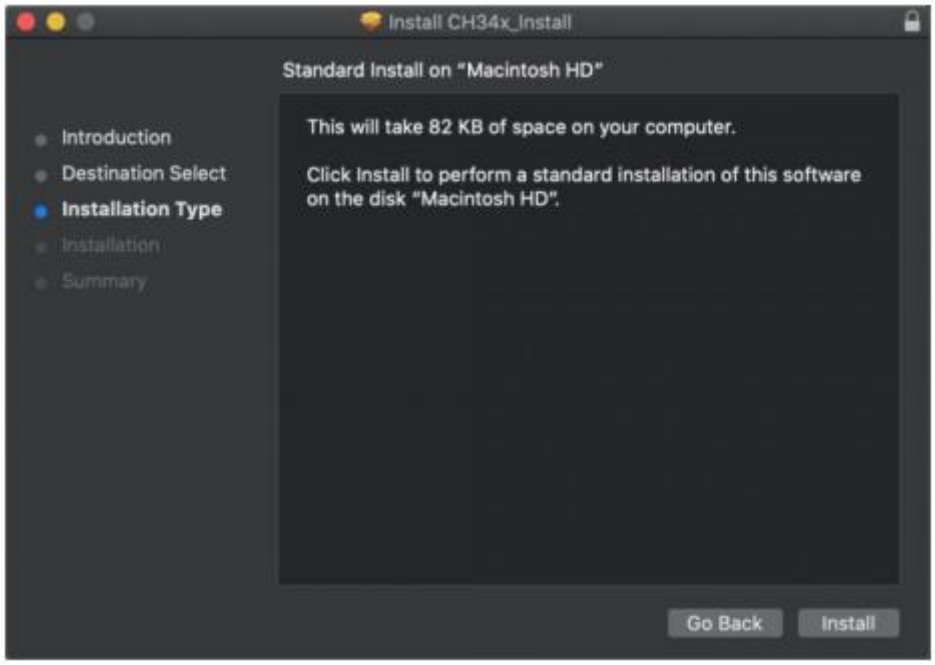
The driver file of the R3 of the MAC system is provided in the tutorial data package.



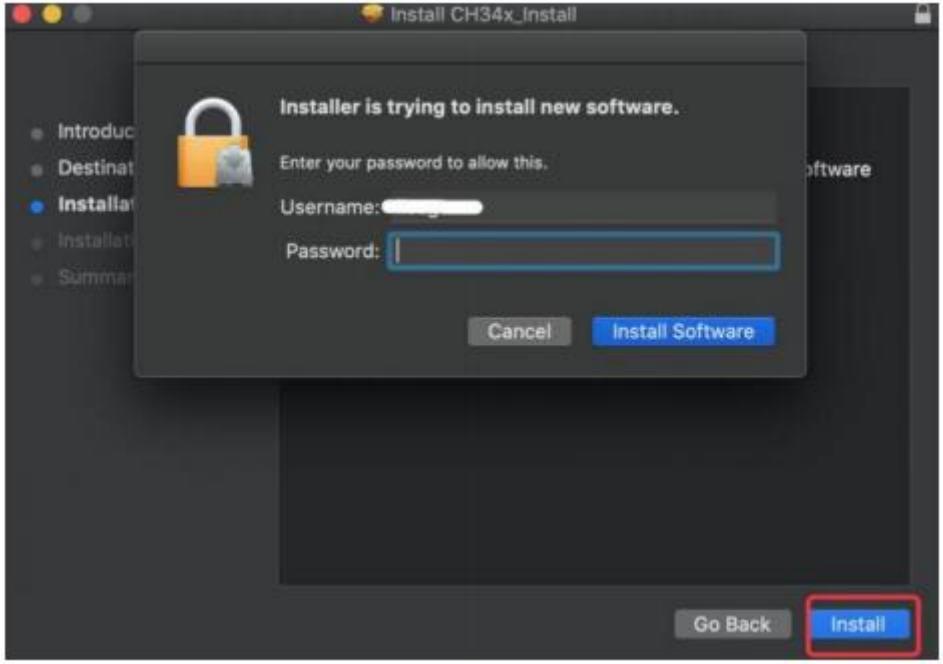
Double-click installation package and tap Continue



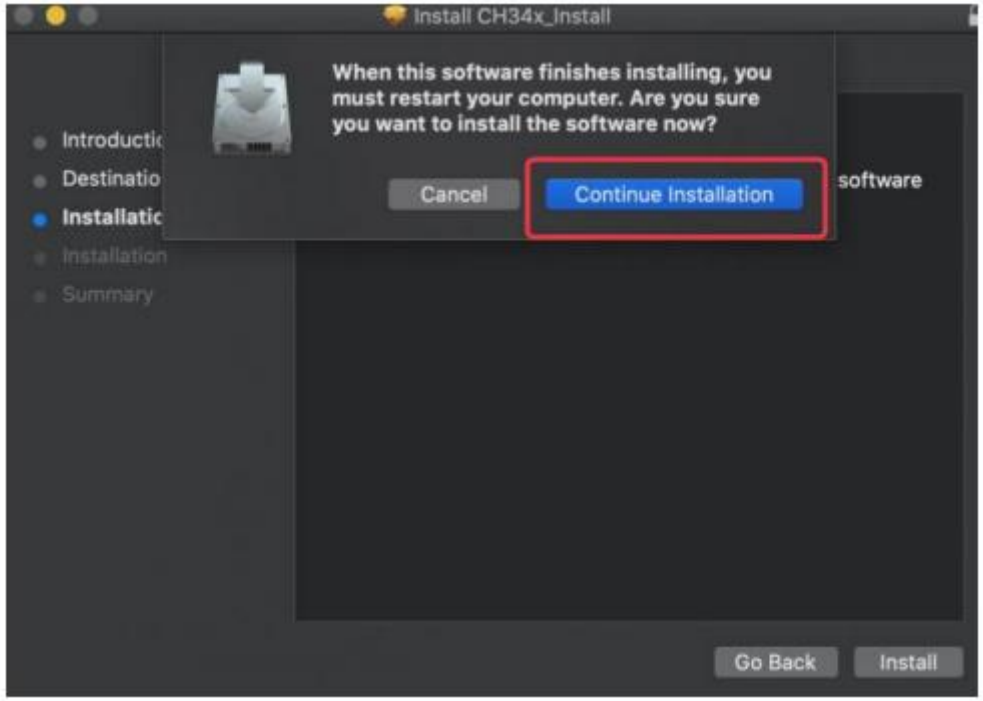
Click Install



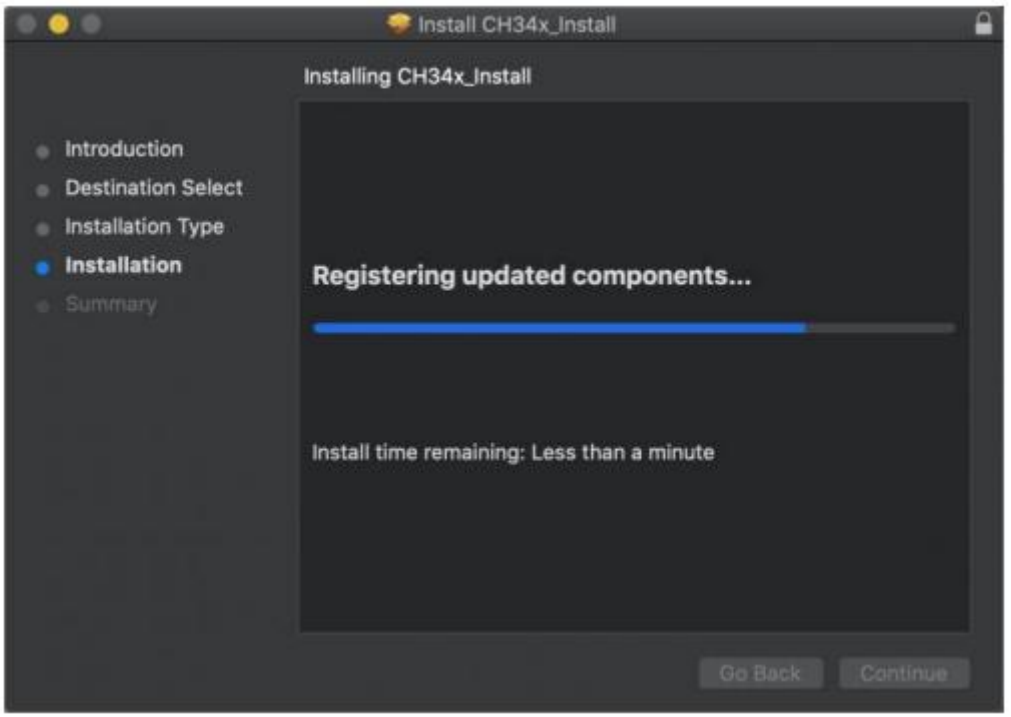
Input your user password and click Install Software



Tap Continue Installation

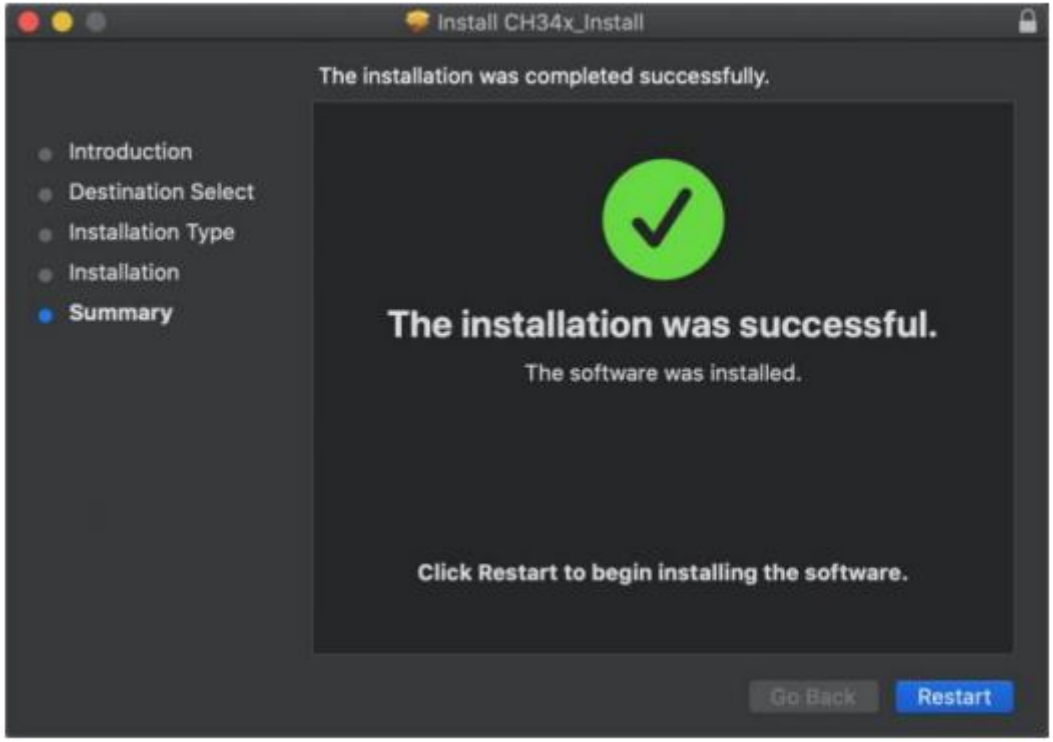


Wait to install



Click Restart after the installation is finished

LROBRUYA


Install CH34x_Install
The installation was completed successfully.
Introduction
Destination Select
Installation Type
Installation
Summary
The installation was successful.
The software was installed.
Click Restart to begin installing the software.
Go Back
Restart

How to Add Arduino Libraries

Installing Additional Arduino Libraries

Once you are comfortable with the Arduino software and using the built-in functions, you may want to extend the ability of your Arduino with additional libraries.

What are Libraries?

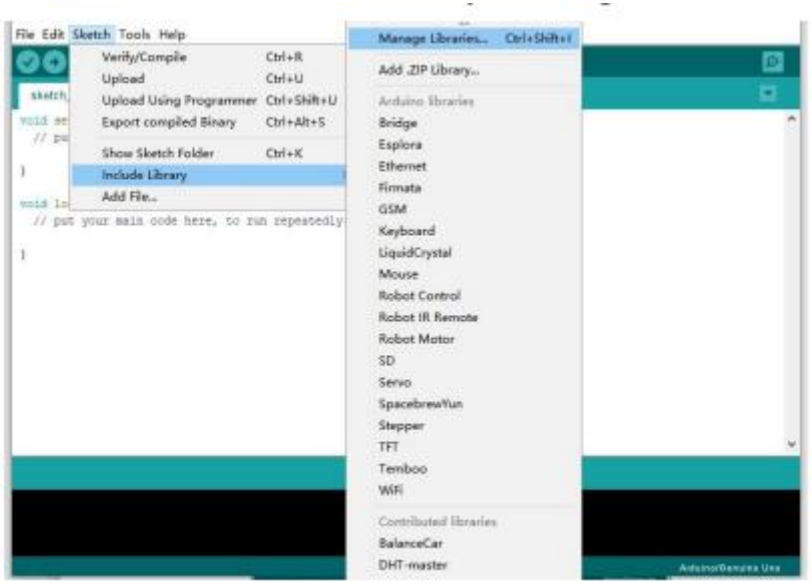
Libraries are a collection of code that makes it easy for you to connect to a sensor, display, module, etc. For example, the built-in Liquid Crystal library makes it easy to talk to character LCD displays. There are hundreds of additional libraries available on the Internet for download.

The built-in libraries and some of these additional libraries are listed in the reference. To use the additional libraries, you will need to install them.

How to Install a Library

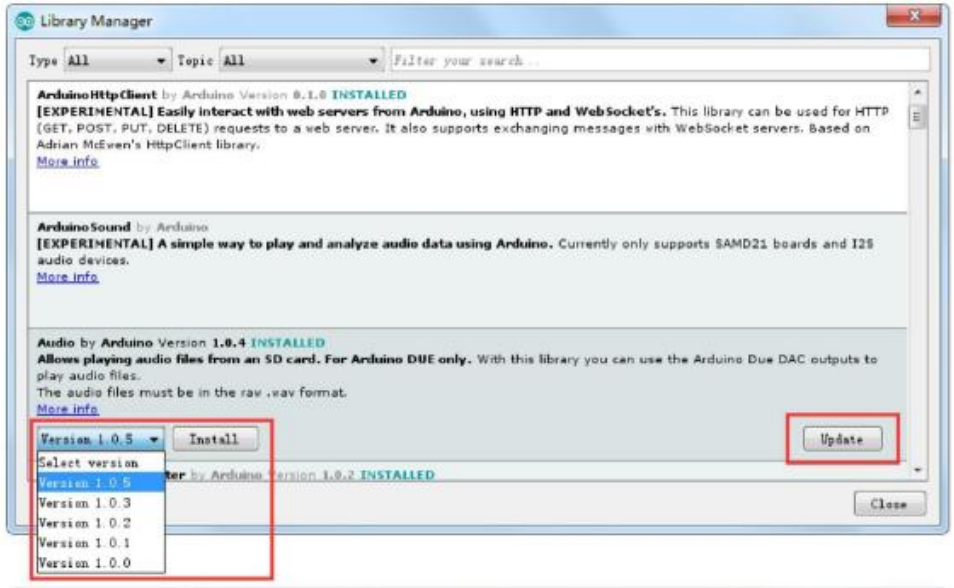
Using the Library Manager

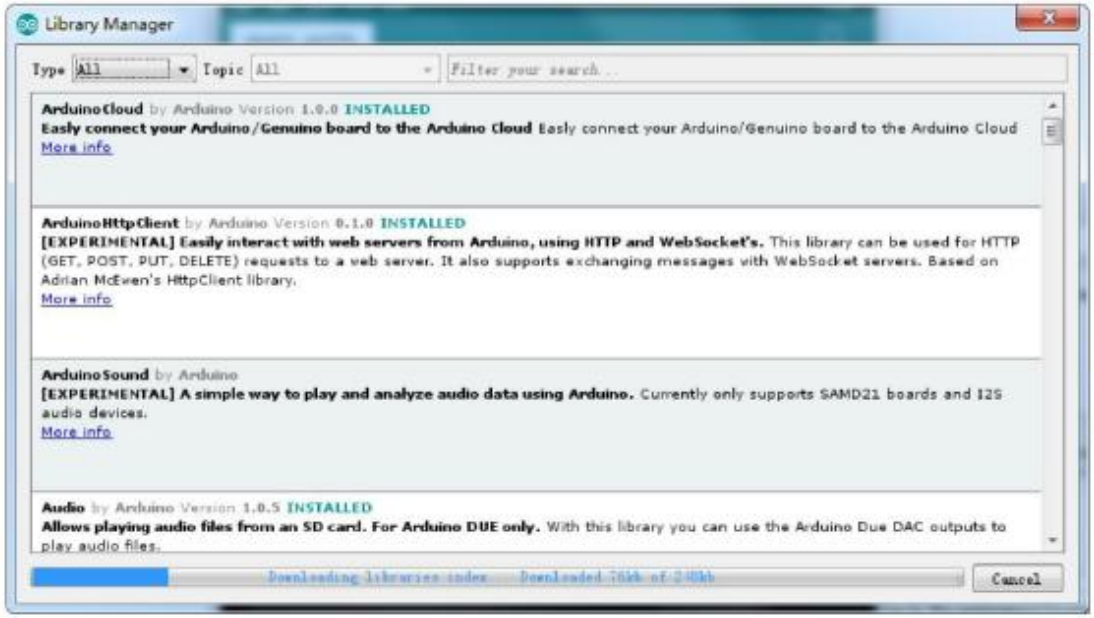
To install a new library into your Arduino IDE you can use the Library Manager (available from IDE version 1.8.0). Open the IDE and click to the "Sketch" menu and then Include Library > Manage Libraries.



Then the library manager will open and you will find a list of libraries that are already installed or ready for installation. In this example we will install the Bridge library. Scroll the list to find it, then select the version of the library you want to install. Sometimes only one version of the library is available. If the version selection menu does not appear, don't worry: it is normal.

There are times you have to be patient with it, just as shown in the figure. Please refresh it and wait.

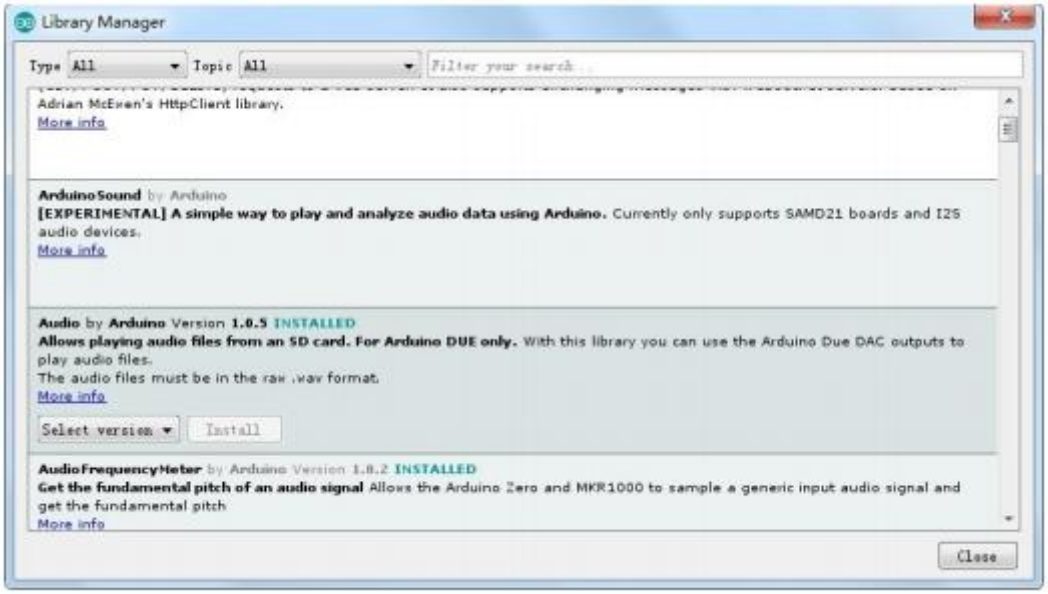




Finally click on install and wait for the IDE to install the new library.

Downloading may take time depending on your connection speed. Once it has finished, an Installed tag should appear next to the Bridge library.

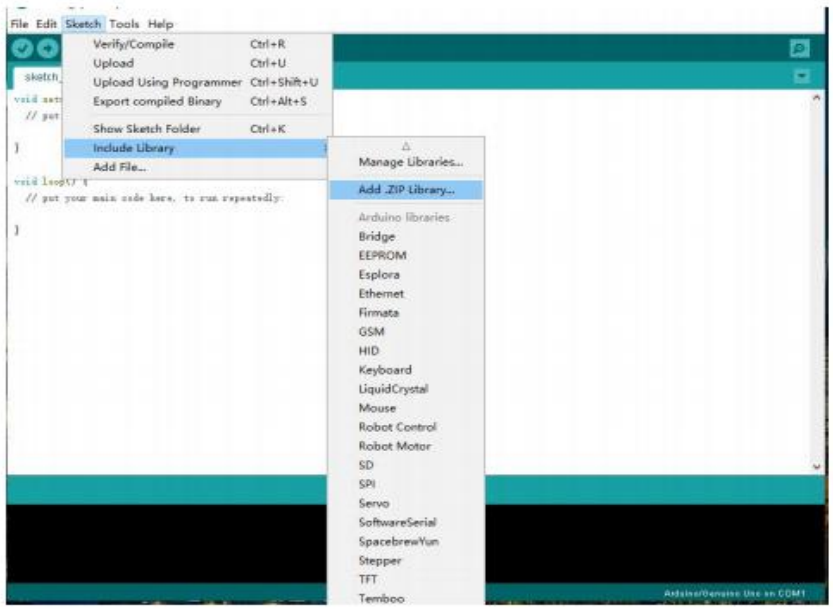
You can close the library manager.



You can now find the new library available in the Include Library menu. If you want to add your own library open a new issue on Github.

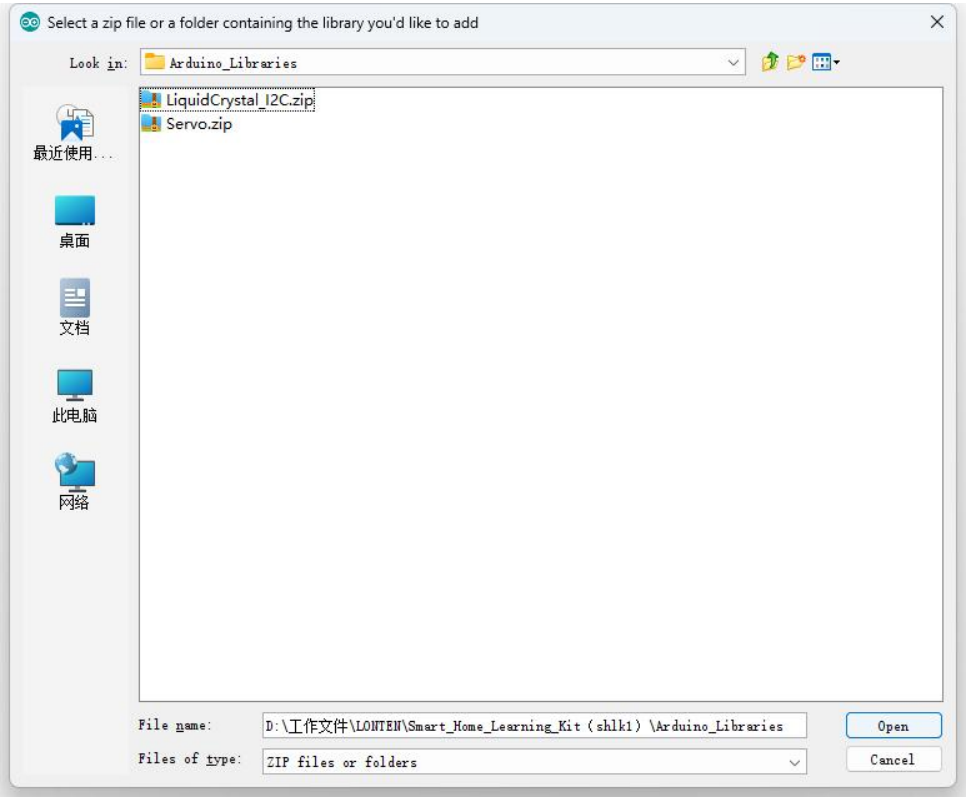
Importing a .zip Library

Libraries are often distributed as a ZIP file or folder. The name of the folder is the name of the library. Inside the folder will be a .cpp file, a .h file and often a keywords.txt file, examples folder, and other files required by the library. Starting with version 1.0.5, you can install 3rd party libraries in the IDE. Do not unzip the downloaded library, leave it as is. In the Arduino IDE, navigate to Sketch > Include Library. At the top of the drop down list, select the option to "Add .ZIP Library".

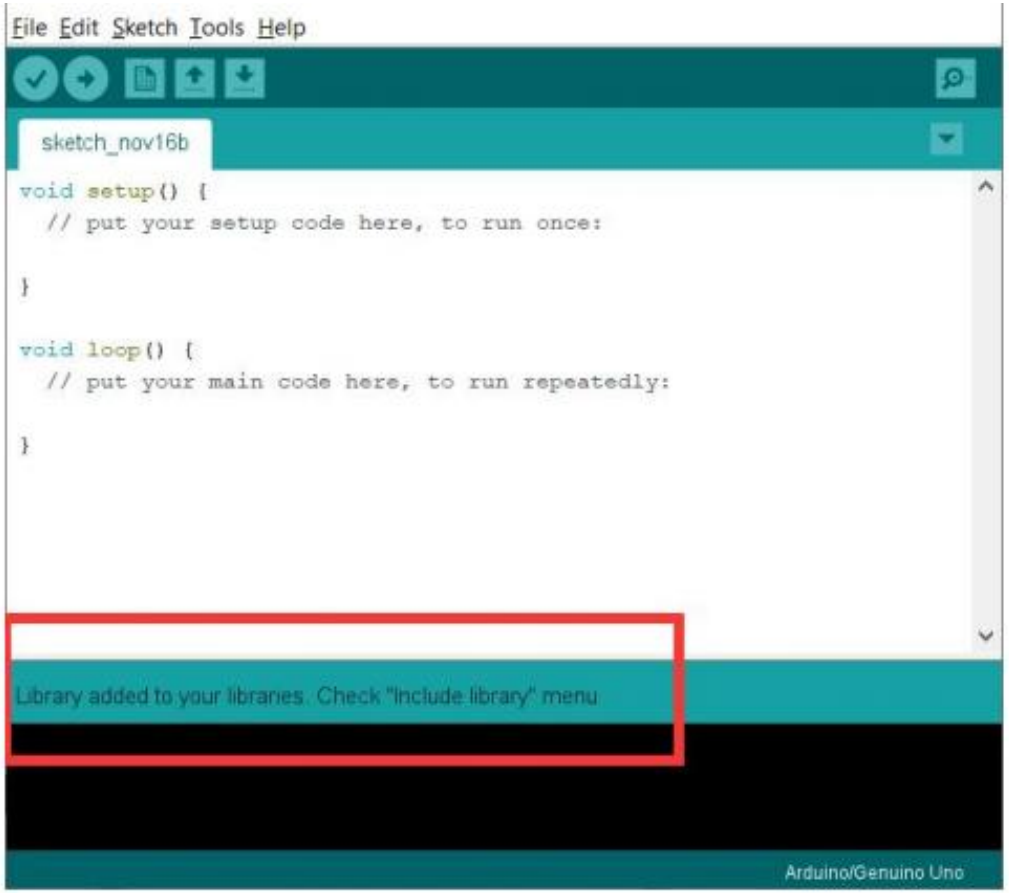


You will be prompted to select the library you would like to add.

Navigate to the .zip file's location and open it.



LROBRUYA


File Edit Sketch Tools Help
sketch_nov16b
void setup() {
// put your setup code here, to run once:
}
void loop() {
// put your main code here, to run repeatedly:
}
Library added to your libraries. Check "Include library" menu
Arduino/Genuino Uno

Return to the Sketch > Import Library menu. You should now see the library at the bottom of the drop-down menu. It is ready to be used in your sketch. The zip file will have been expanded in the libraries folder in your Arduino sketches directory. NB: the Library will be available to use in sketches, but examples for the library will not be exposed in the File > Examples until after the IDE has restarted.

Blink Test

Overview

In this Project, you will learn how to program your UNO R3 controller board to blink the Arduino's built-in LED, and how to download programs by basic steps.

Component Required:

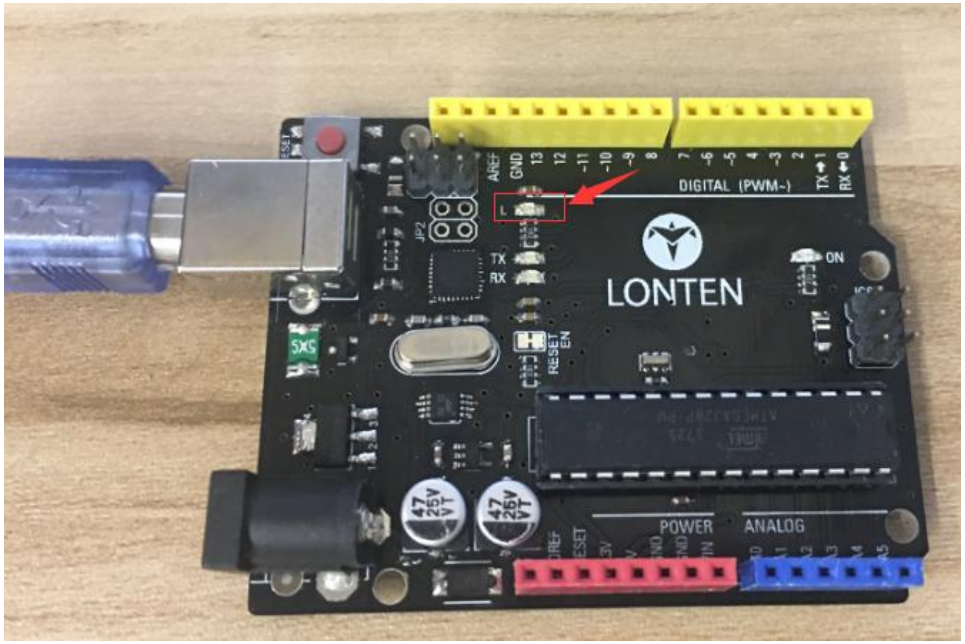
LONTEN Uno R3 Board* 1

Principle

The UNO R3 board has rows of connectors along both sides that are used to connect to several electronic devices and plug-in 'shields' that extends its capability.

It also has a single LED that you can control from your sketches. This LED is built onto the UNO R3 board and is often referred to as the 'L' LED as this is how it is labeled on the board.

LROBRUYA

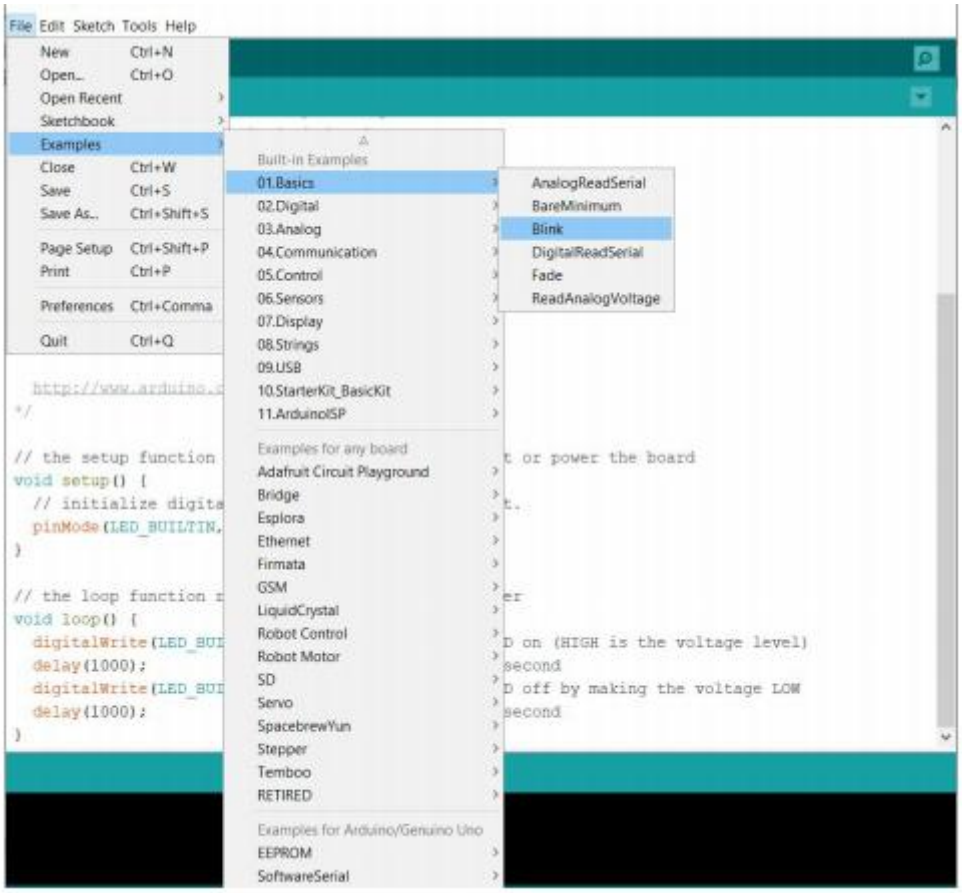


In this Project, we will reprogram the UNO board with our own Blink sketch and then change the rate at which it blinks.

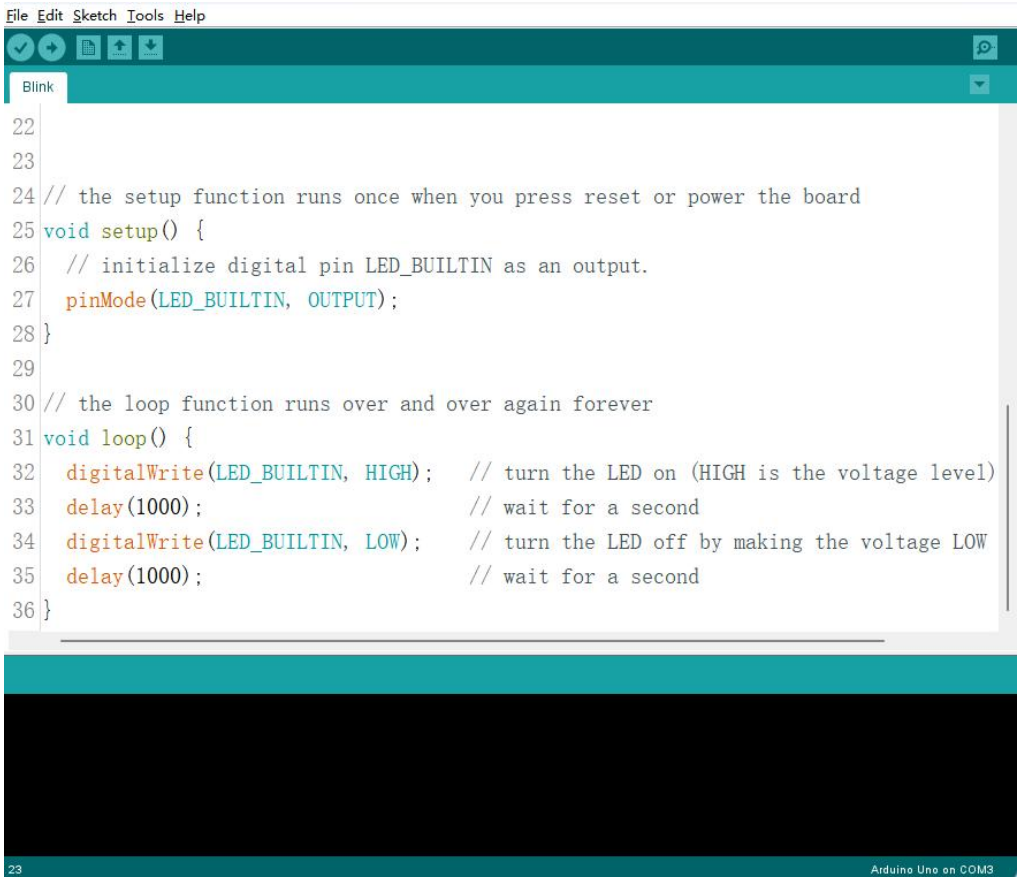
In the previous chapter-How to install Arduino IDE, you set up your Arduino IDE and made sure that you could find the right serial port for it to connect to your UNO board. The time has now come to put that connection to the test and program your UNO board.

The Arduino IDE includes a large collection of example sketches that you can load up and use. This includes an example sketch for making the 'L' LED blink.

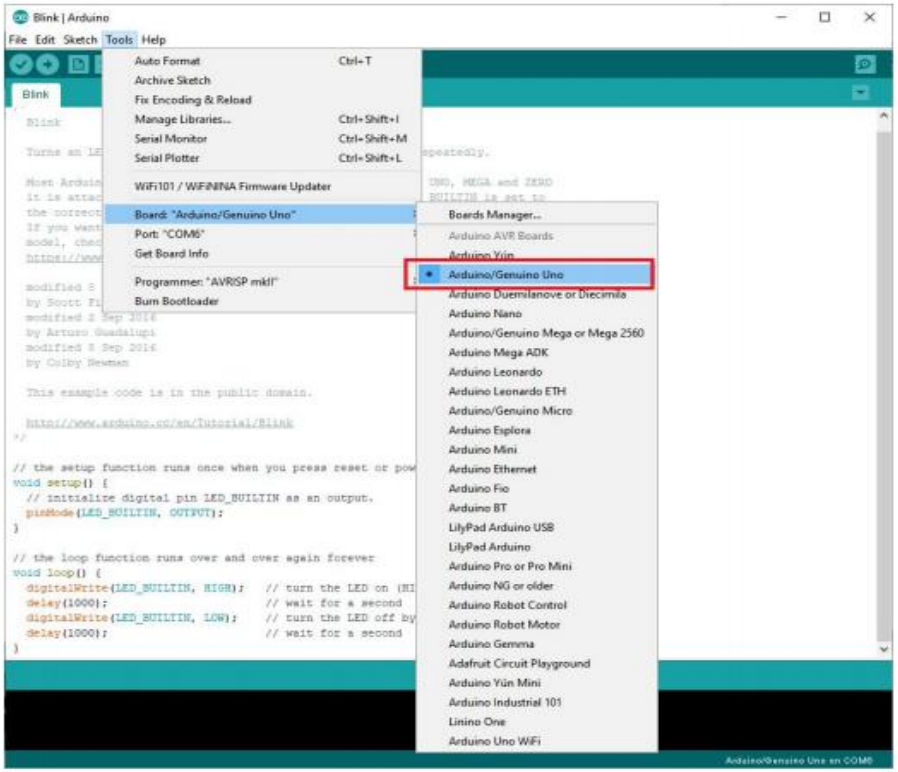
Load the 'Blink' sketch that you will find in the IDE's menu system under File > Examples > 01.Basics>Blink



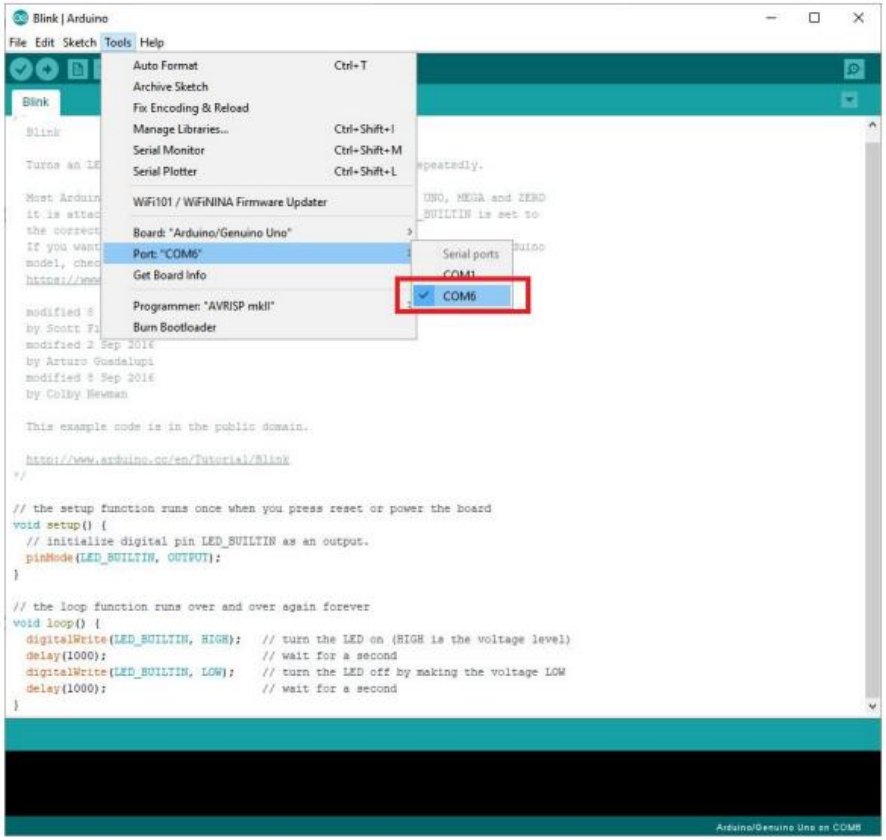
When the sketch window opens, enlarge it so that you can see the entire sketch in the window.

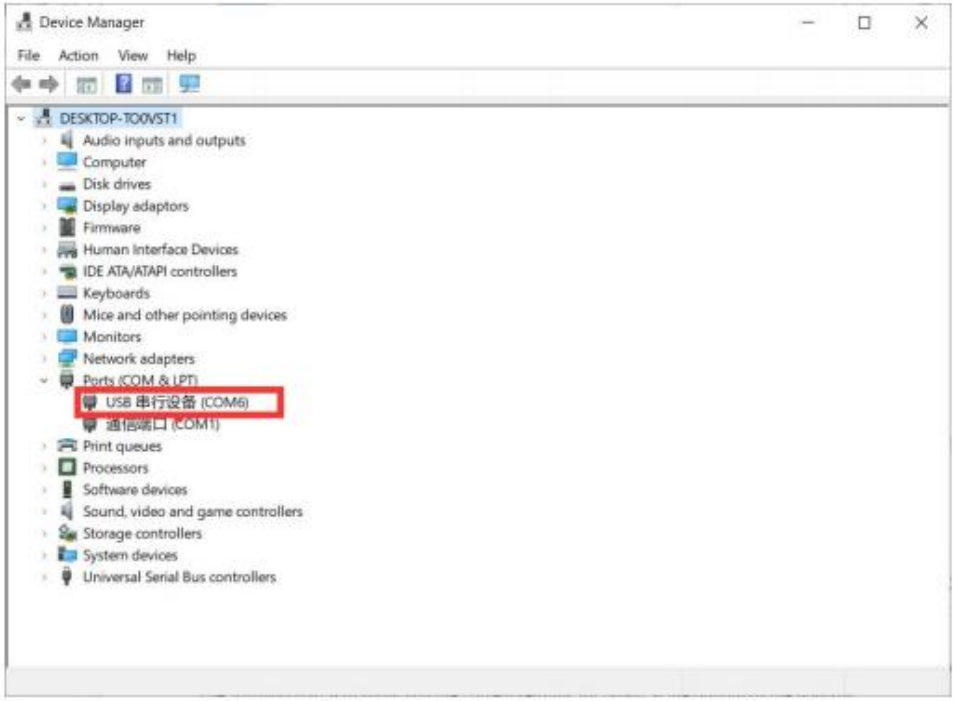

File Edit Sketch Tools Help
Blink
22
23
24 // the setup function runs once when you press reset or power the board
25 void setup() {
26 // initialize digital pin LED_BUILTIN as an output.
27 pinMode(LED_BUILTIN, OUTPUT);
28 }
29
30 // the loop function runs over and over again forever
31 void loop() {
32 digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
33 delay(1000); // wait for a second
34 digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
35 delay(1000); // wait for a second
36 }
23
Arduino Uno on COM3

Attach your Arduino board to your computer with the USB cable and check that the 'Board Type' and 'Serial Port' are set correctly.



LROBRUYA


Blink | Arduino
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino/Genuino Uno"
Port: "COM6"
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Serial ports
COM1
COM6
Blink
by Arturo Guadalupi
modified 8 Sep 2016
by Colby Newman
This example code is in the public domain.
http://www.arduino.cc/en/Tutorial/Blink
// the setup function runs once when you press reset or power the board
void setup() {
// initialize digital pin LED_BUILTIN as an output.
pinMode(LED_BUILTIN, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
delay(1000); // wait for a second
digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
delay(1000); // wait for a second
}
Arduino/Genuino Uno on COM6

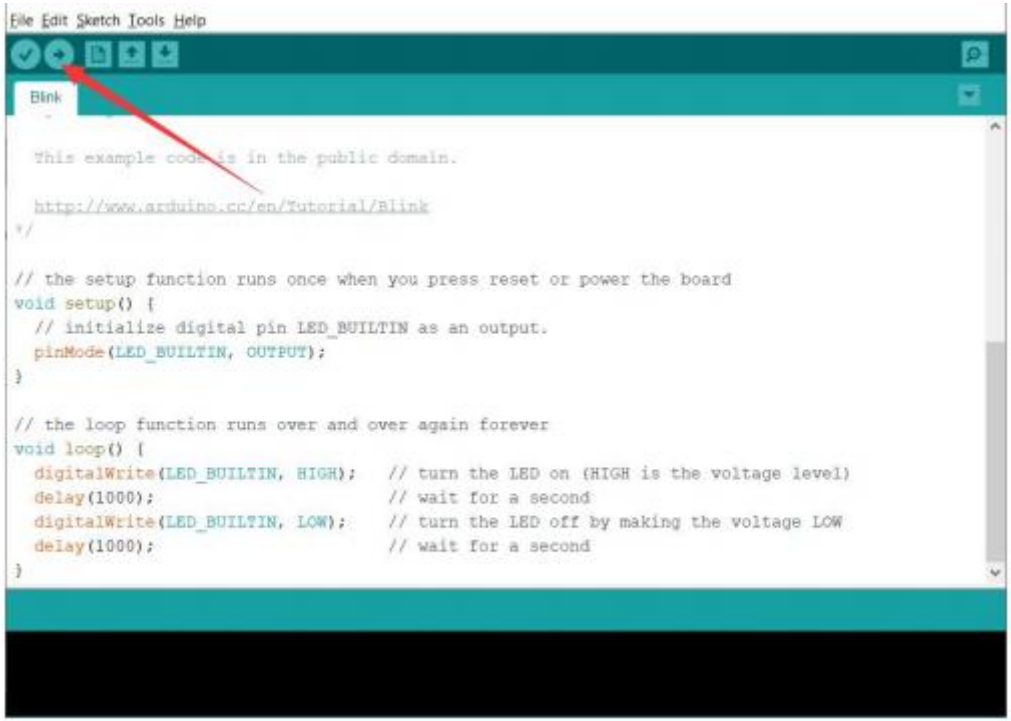


Note: The Board Type and Serial Port here are not necessarily the same as shown in picture. If you are using UNO, then you will have to choose Arduino UNO as the Board Type, other choices can be made in the same manner. And

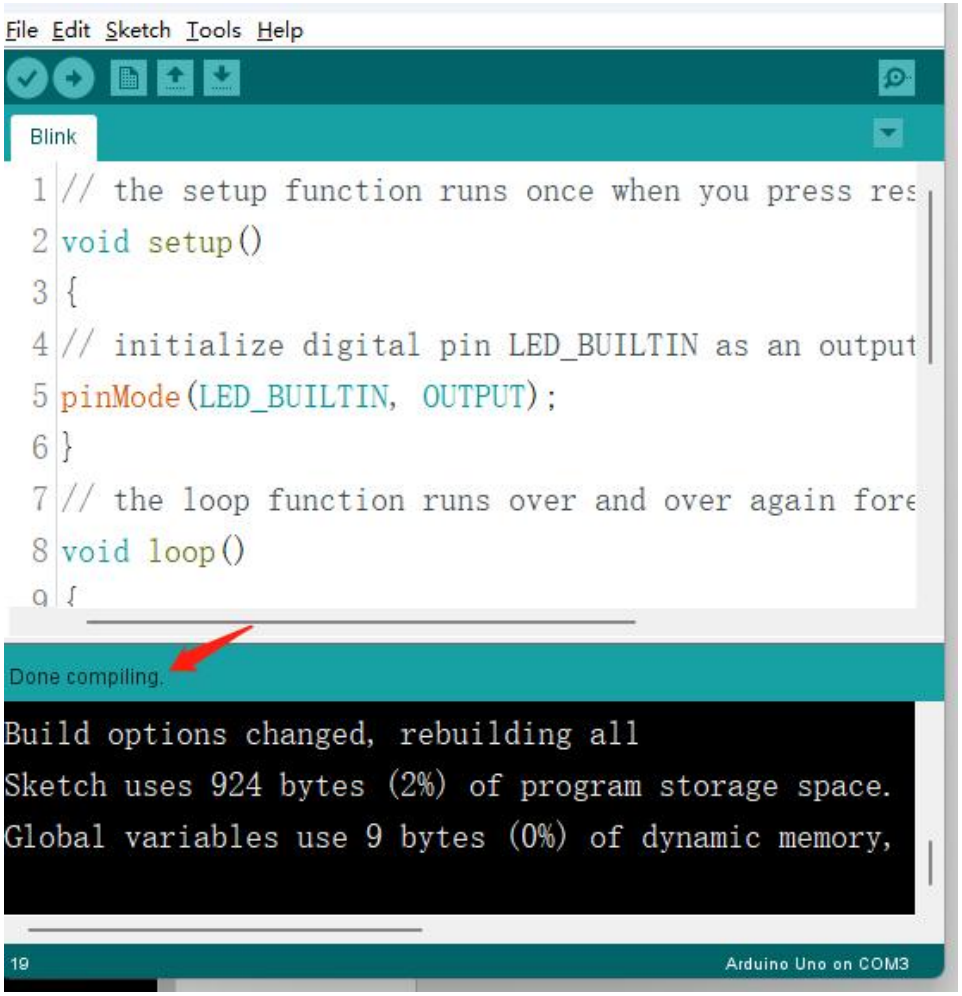
the Serial Port displayed for everyone is different, despite COM 6 chosen here, it could be COM3 or COM4 on your computer. A right COM port is supposed to be COMX (arduino XXX), which is by the certification criteria.

The Arduino IDE will show you the current settings for board at the bottom of the window.

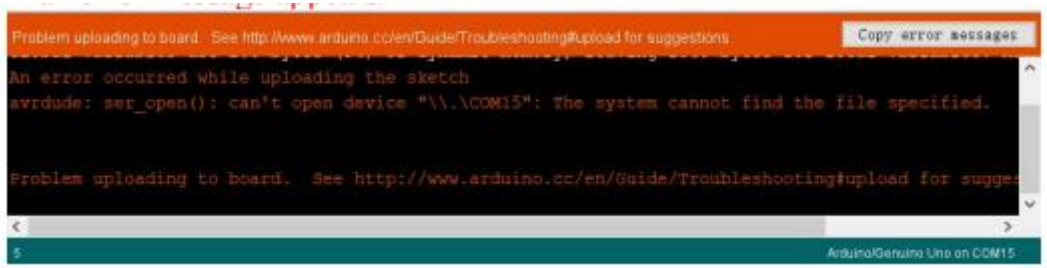
Click on the 'Upload' button. The second button from the left on the toolbar.



When the status bar prompts "Done uploading", it means the code upload is successful


File Edit Sketch Tools Help
Blink
1 // the setup function runs once when you press res
2 void setup()
3 {
4 // initialize digital pin LED_BUILTIN as an output
5 pinMode(LED_BUILTIN, OUTPUT);
6 }
7 // the loop function runs over and over again fore
8 void loop()
Done compiling.
Build options changed, rebuilding all
Sketch uses 924 bytes (2%) of program storage space.
Global variables use 9 bytes (0%) of dynamic memory,
19
Arduino Uno on COM3

If an error message appears.

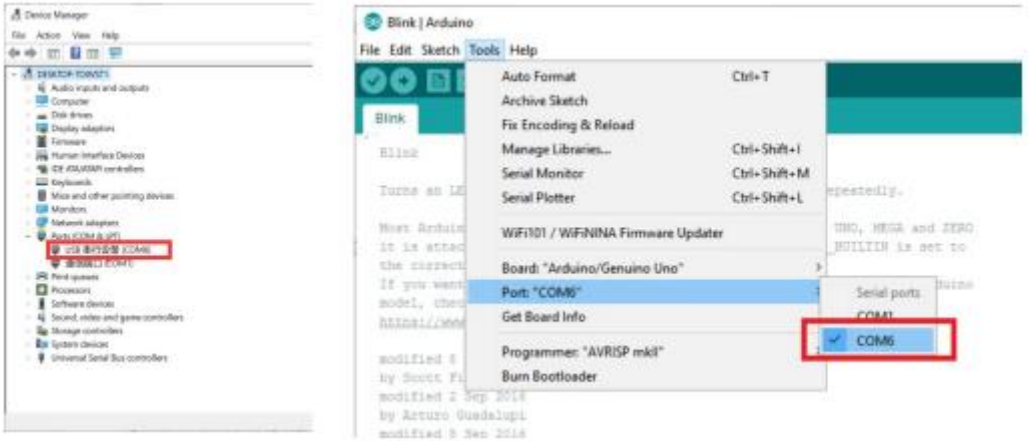


There can be several reasons:

1. The arduino uno driver software is not installed successfully, please refer to the course for the installation steps: How to Install Arduino Driver.

2. The communication serial port selection of arduino uno is wrong;

you can check the communication port COMx of your arduino uno in the computer in the device manager.

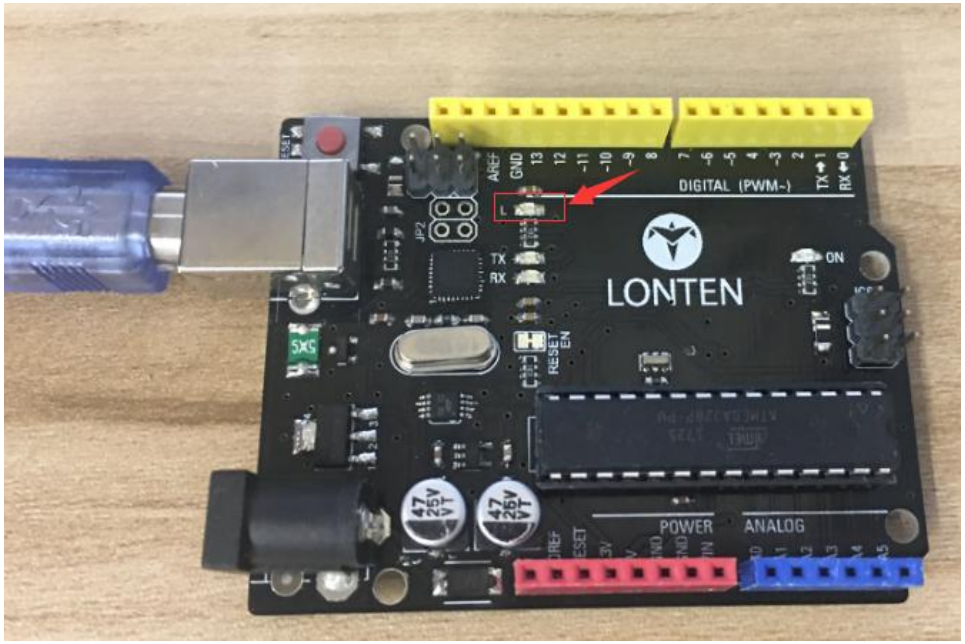


3. If your Arduino uno is connected to a Bluetooth module, it will occupy the communication serial port. You need to remove the Bluetooth module connection before uploading the code.

4. The USB data cable is not firmly connected. Check if there are any of the above problems. After correcting, follow the previous steps to re-operate.

Test Code

```
void setup() // the setup function runs once when you press reset or power the board
{
    pinMode(LED_BUILTIN, OUTPUT); // initialize digital pin LED_BUILTIN as an output.
}
void loop() // the loop function runs over and over again forever
{
    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(1000); // wait for a second
    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
    delay(1000); // wait for a second
}
```



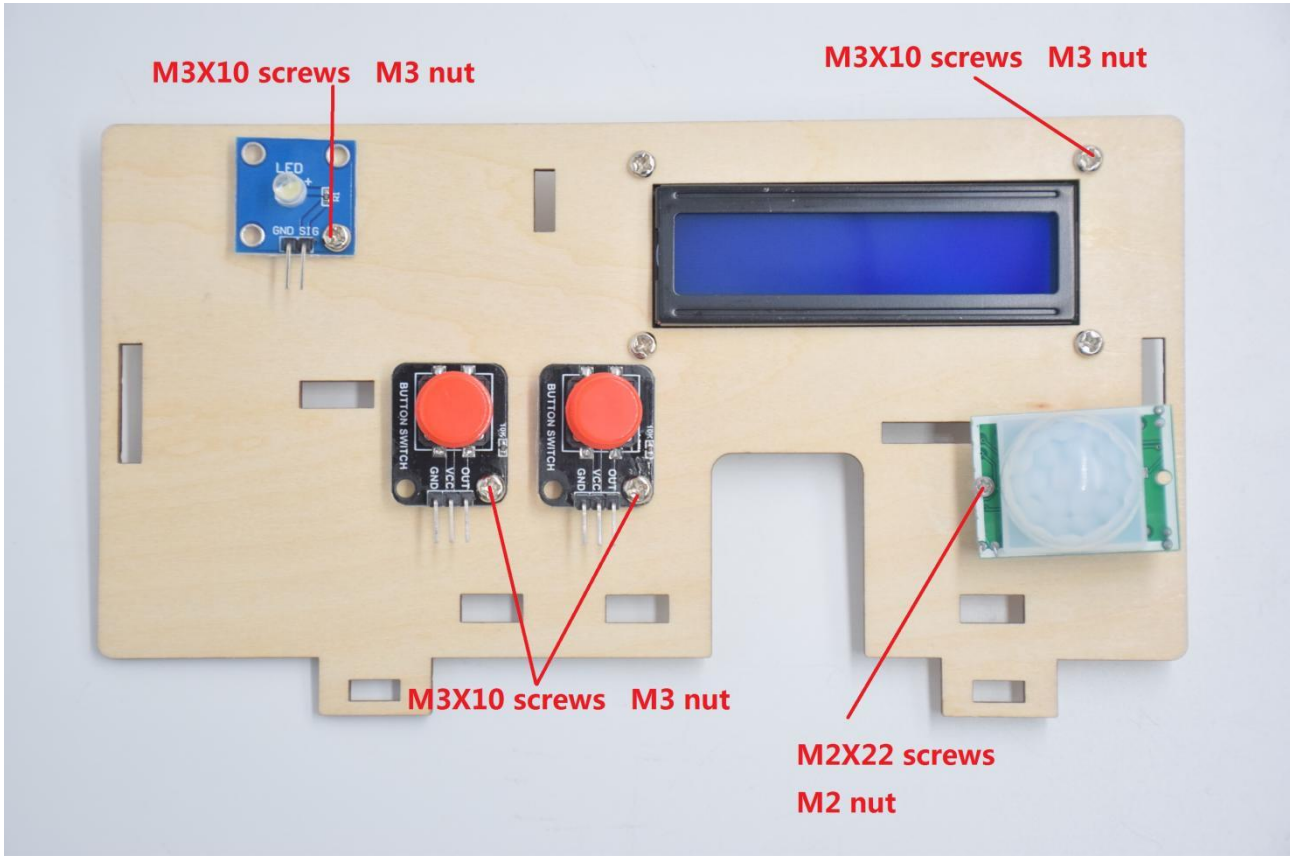
After the code is successfully uploaded, the "L" character LED will flash once per second. So far, you have completed the testing process of your first program.

LROBRUYA

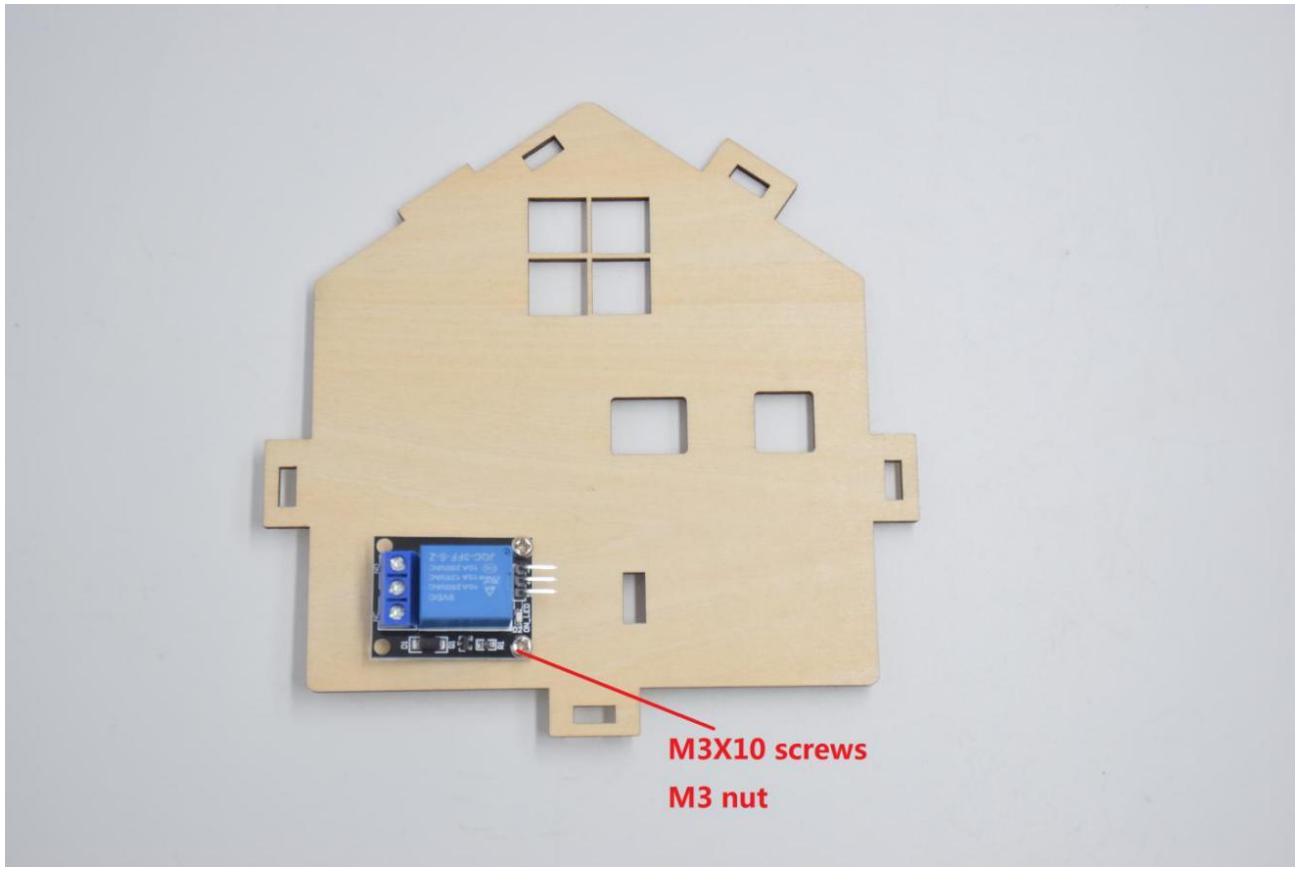
Installation Method

Check the board A~J and parts firstly

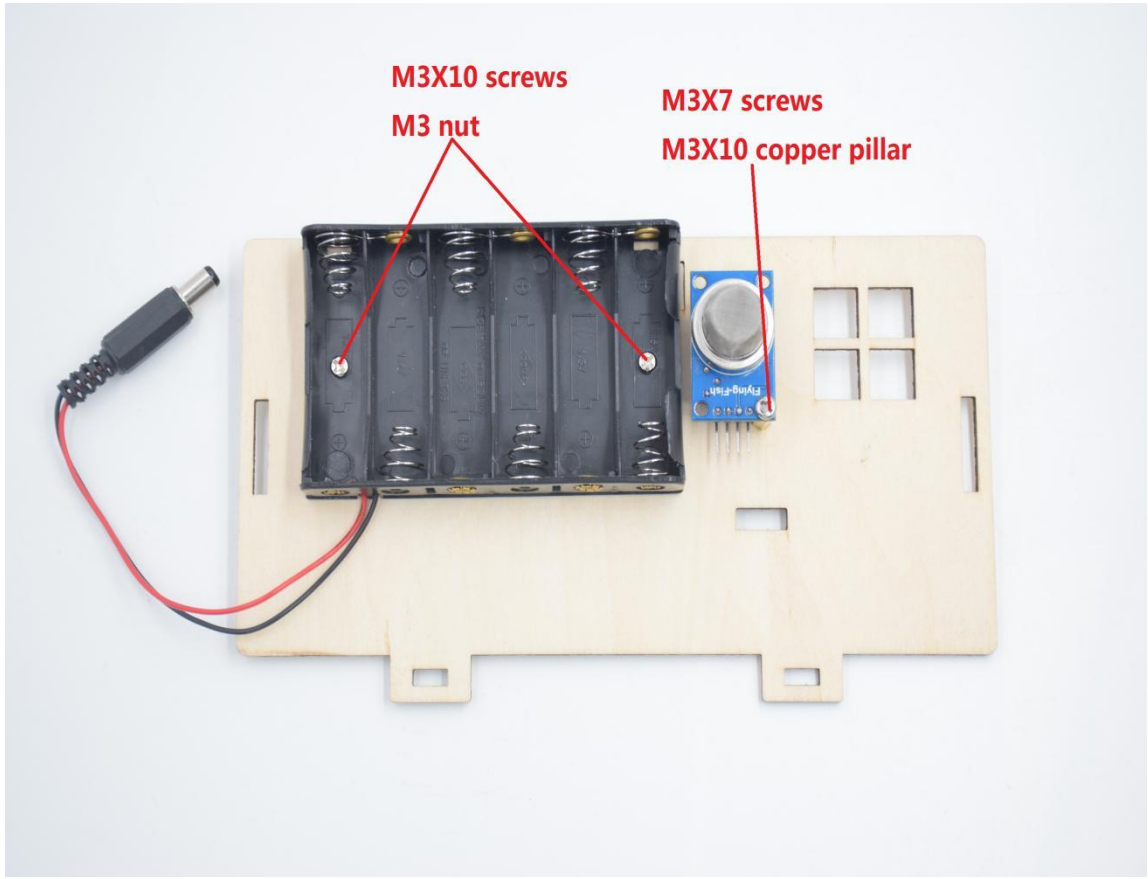
Step1:Install sensors of A board



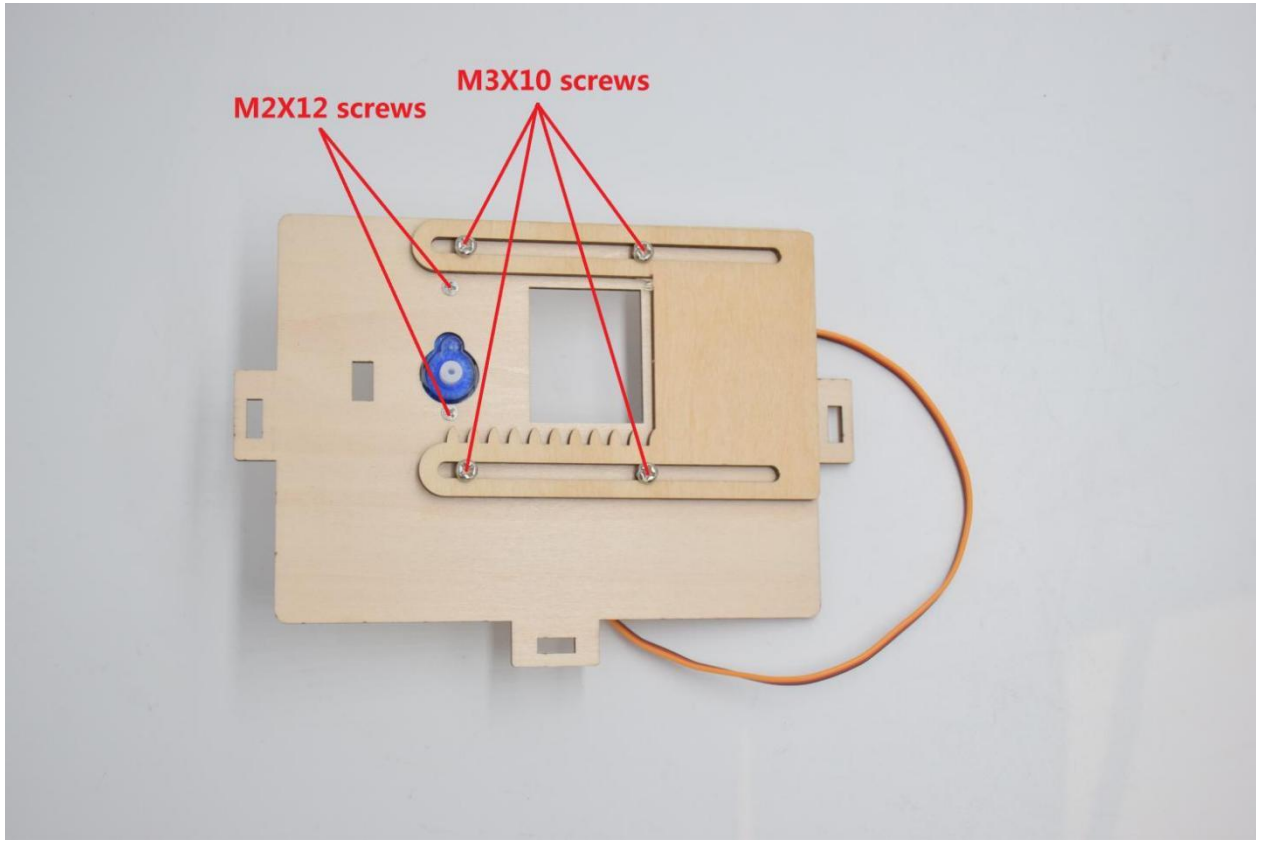
Step2:Install the sensor of B board

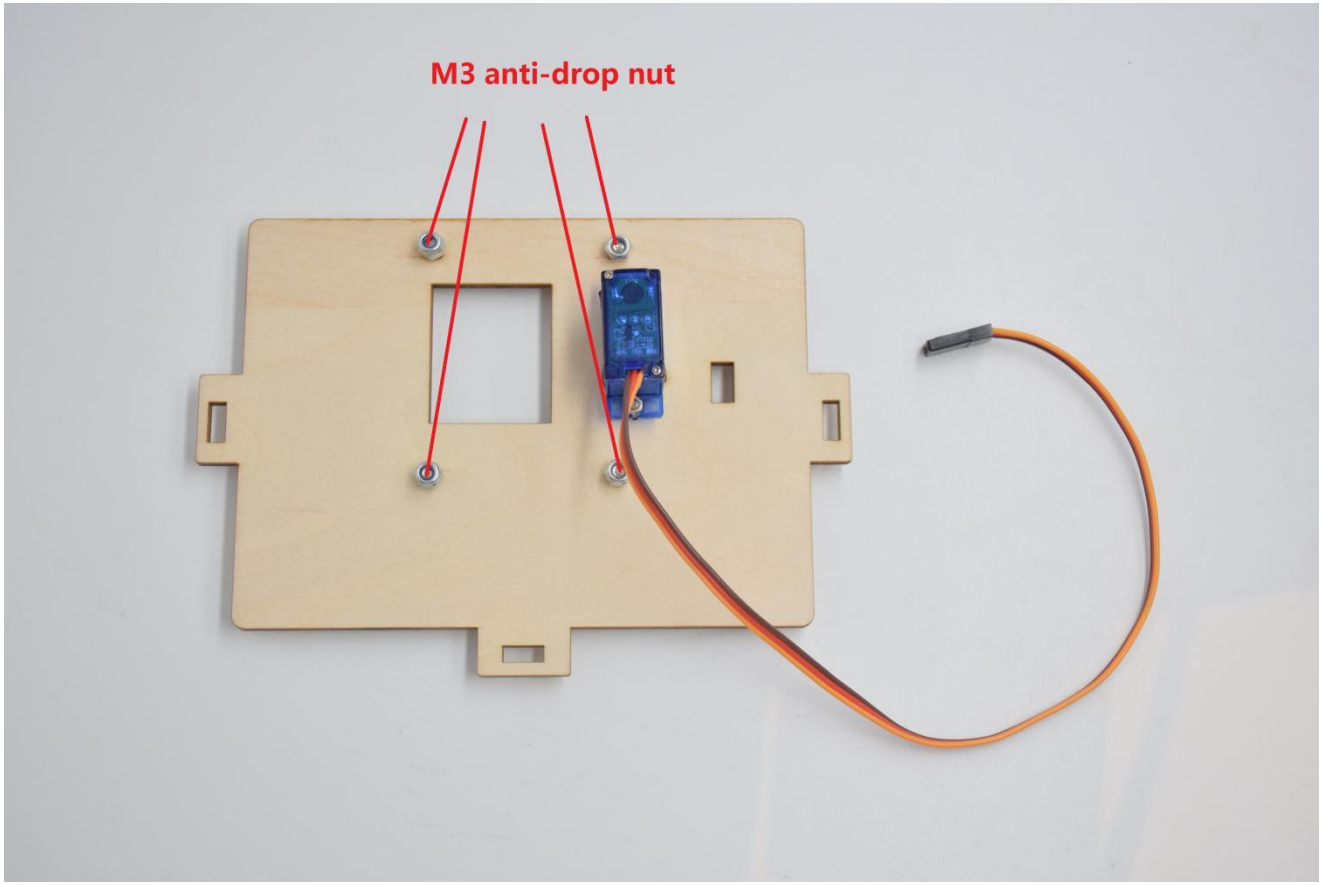


Step3: Assemble the sensors and battery holder of C board

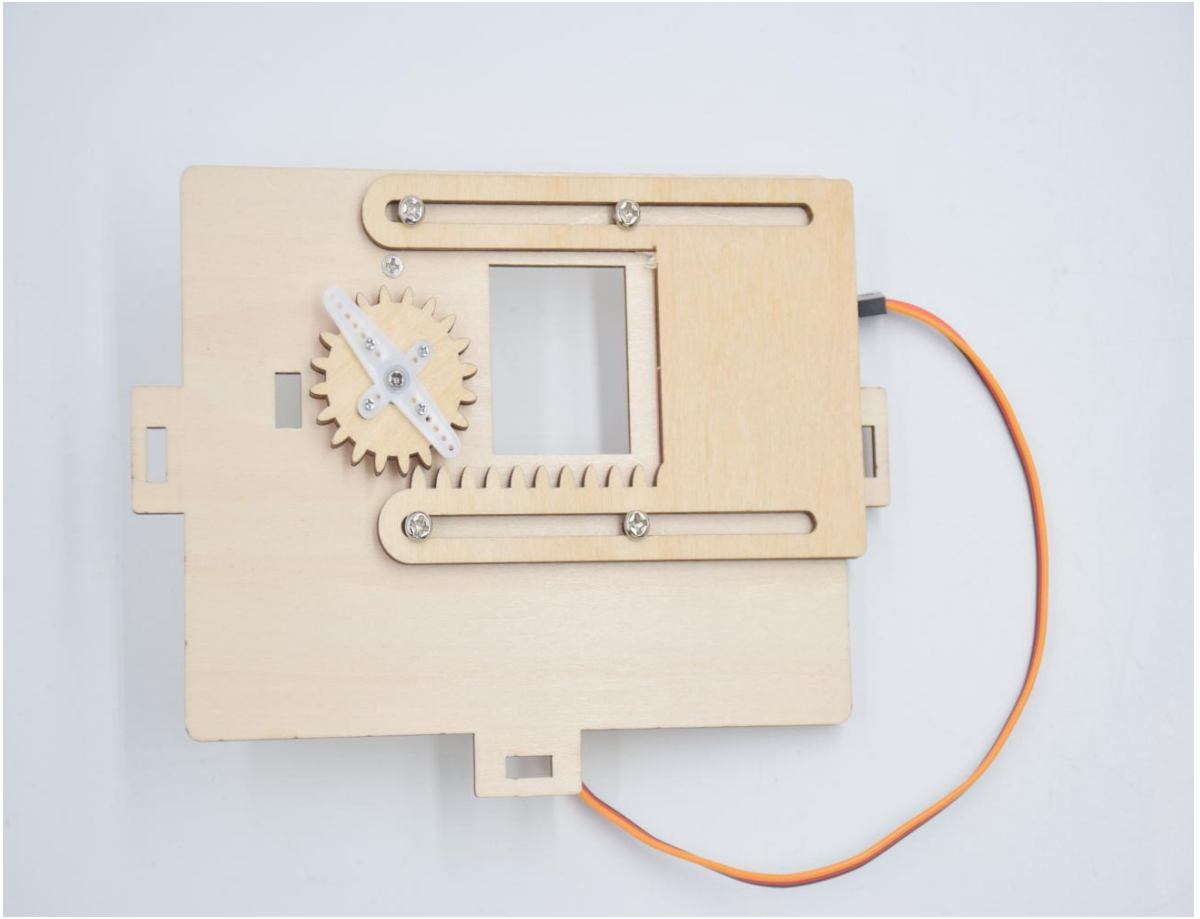


Step4:Install the sensors and parts of D board

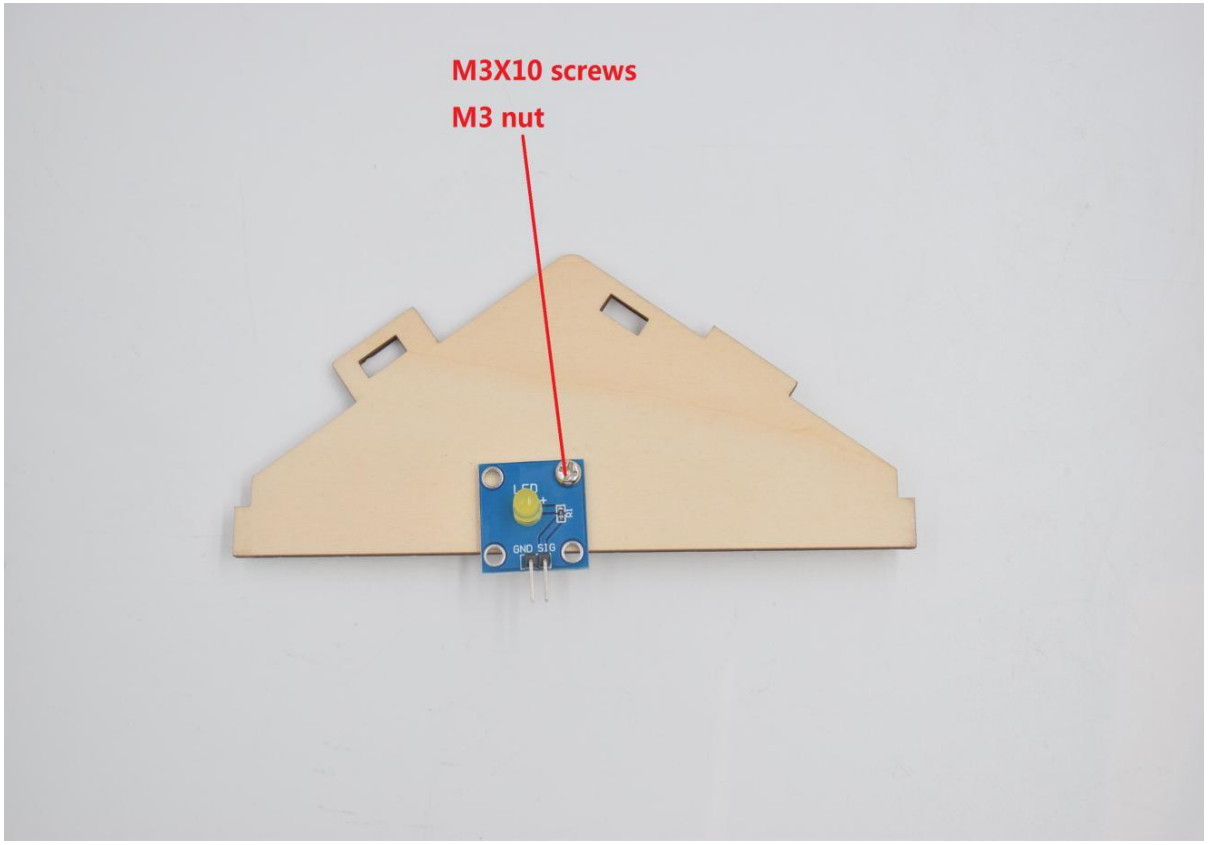



M3 anti-drop nut

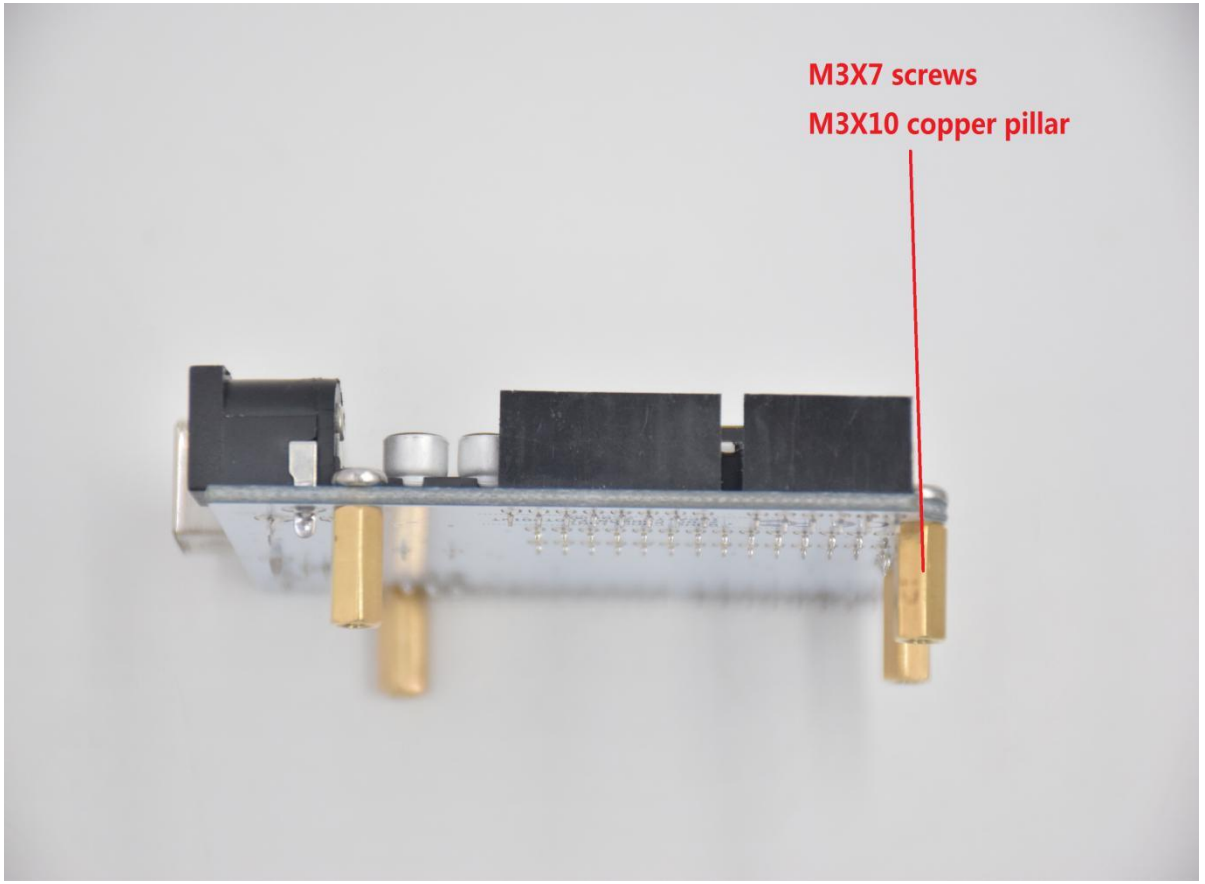
LROBRUYA



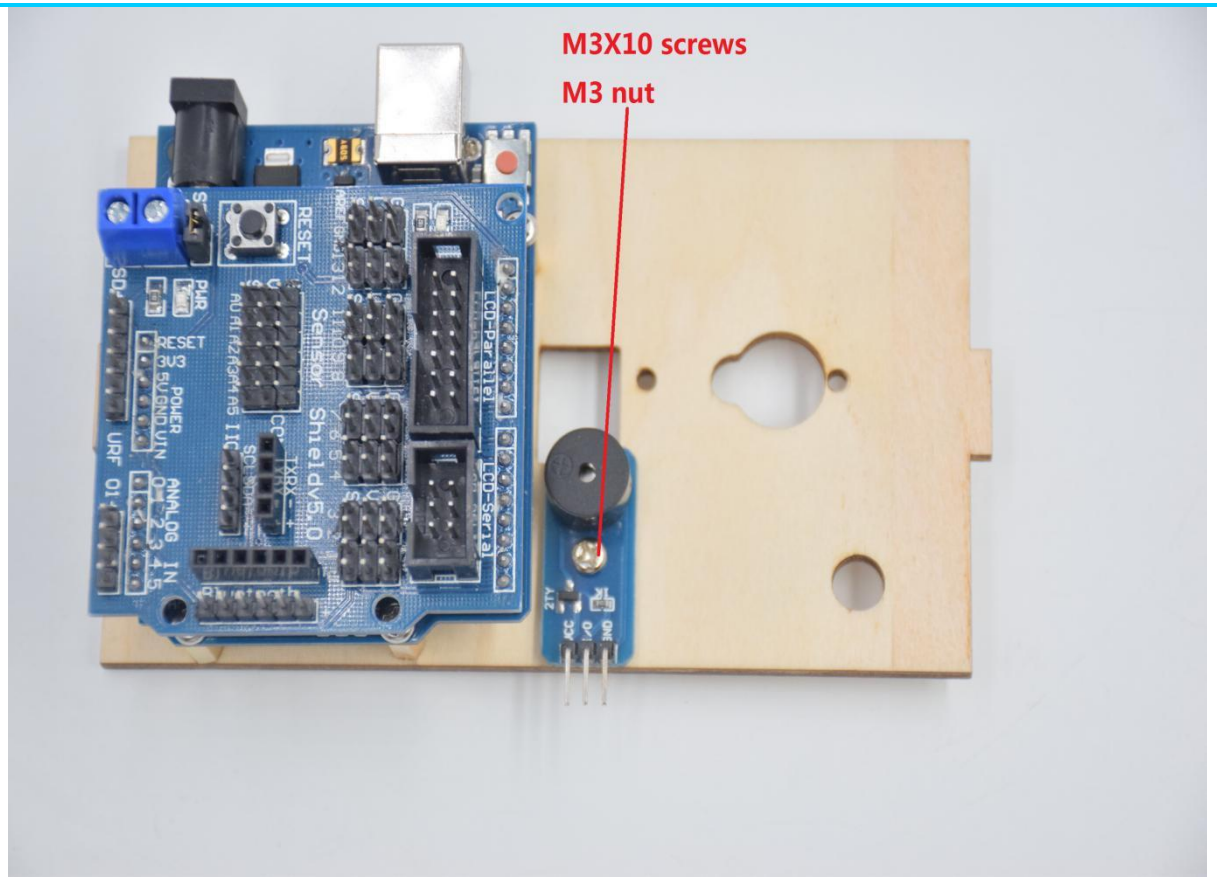
Step 5: install the sensor of E board



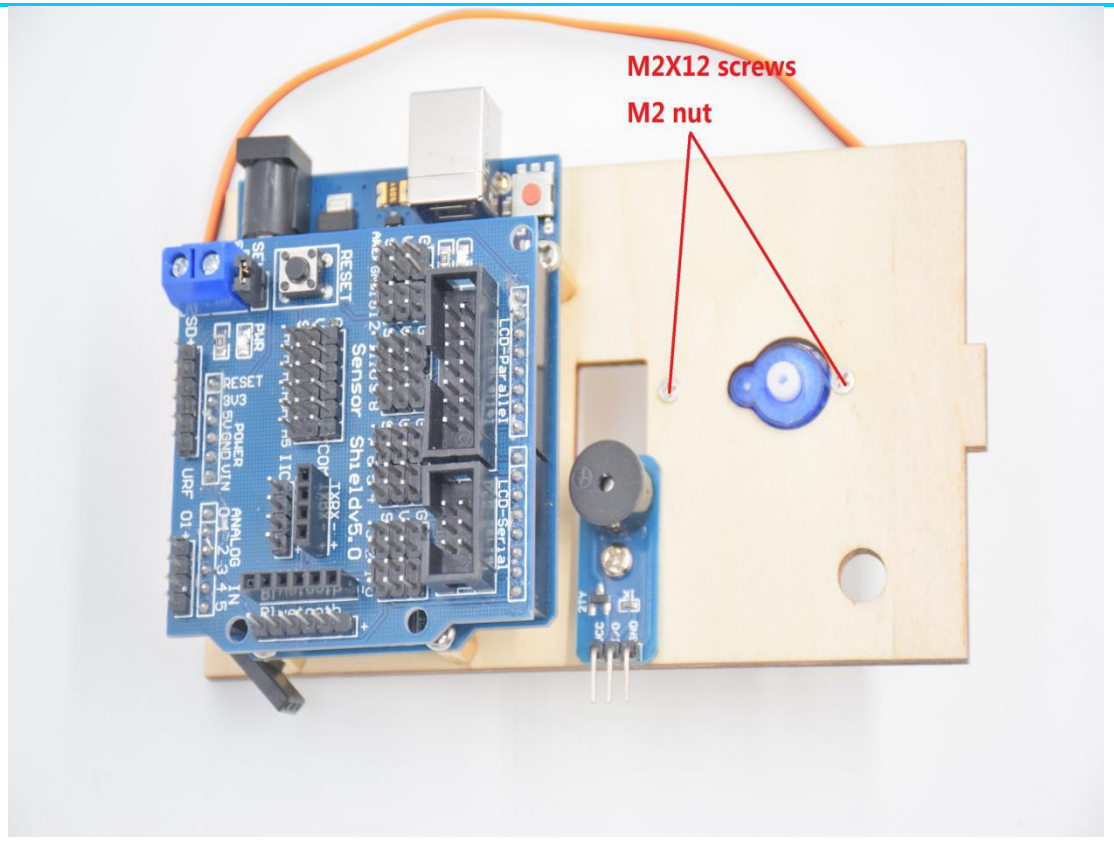
Step 6: Install control board, sensors and parts of F board



LROBRUYA

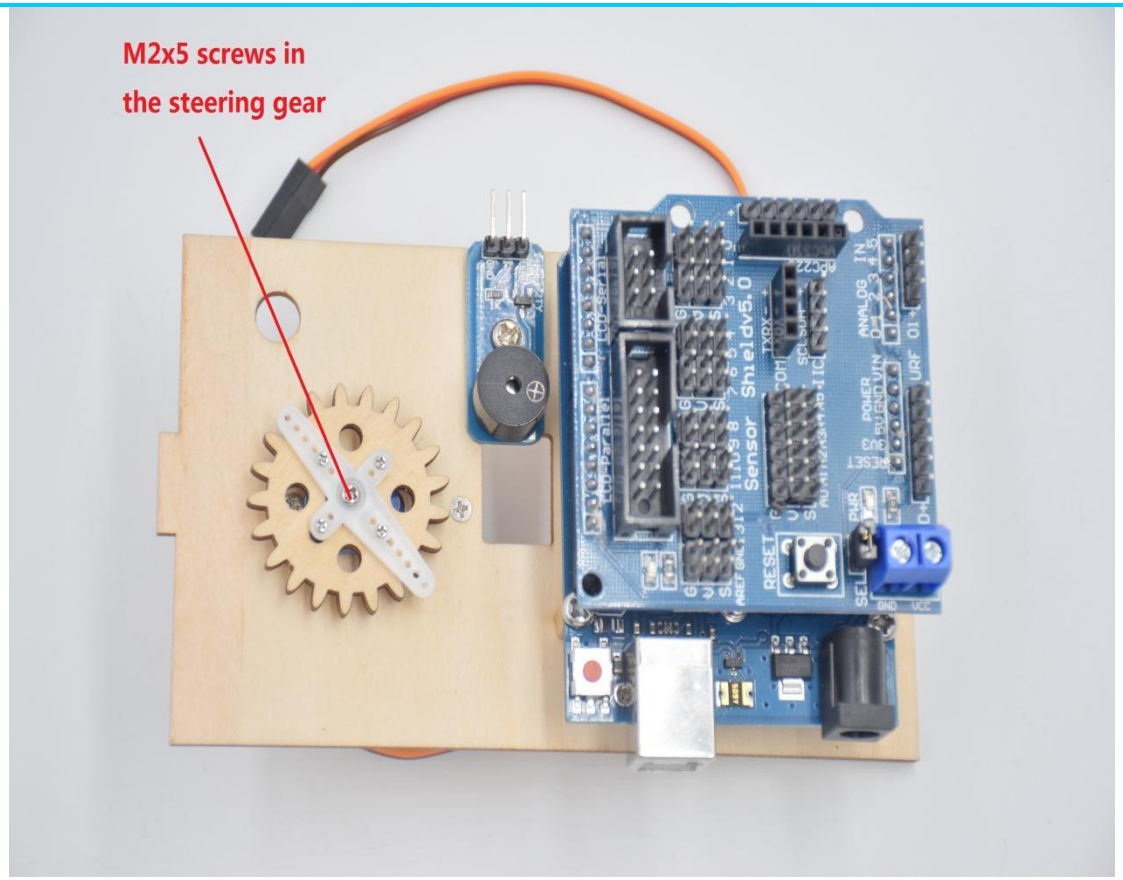


LROBRUYA


M2X12 screws
M2 nut
RESET
Sensor Shield v5.0
LCD-Parallel
LCD-Serial

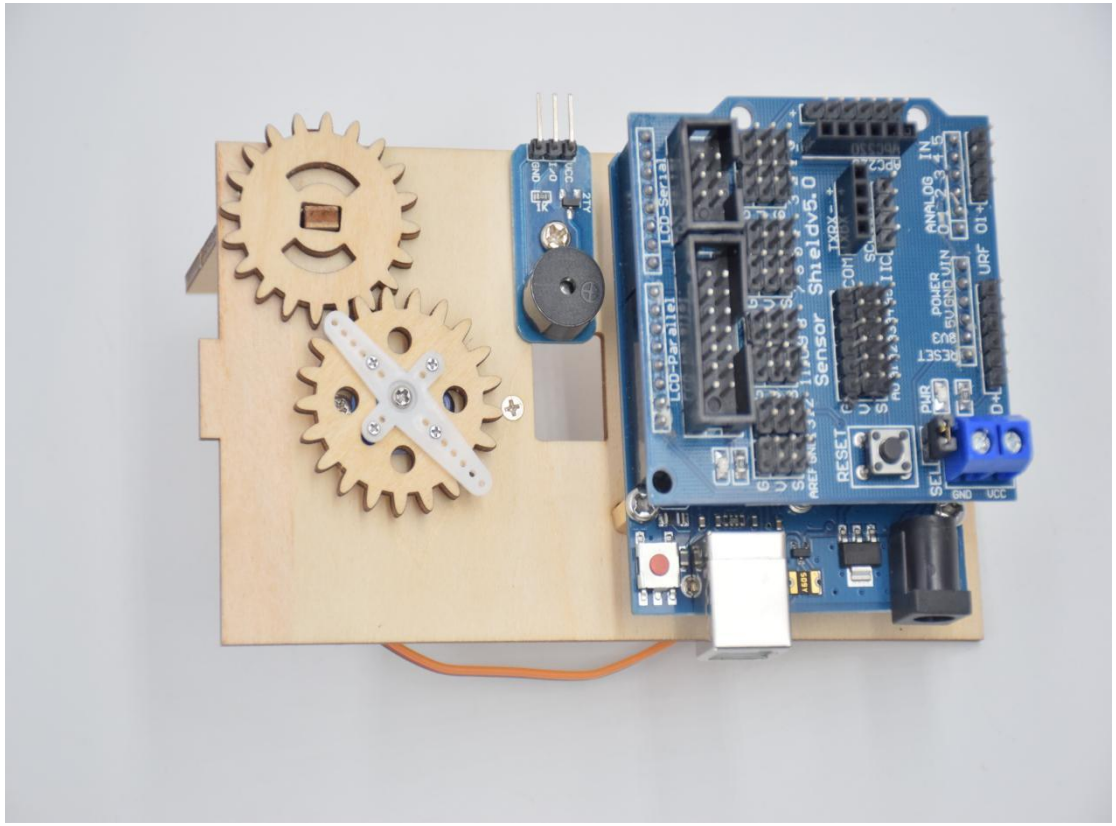

M1.2X5 self-tapping screws

LROBRUYA

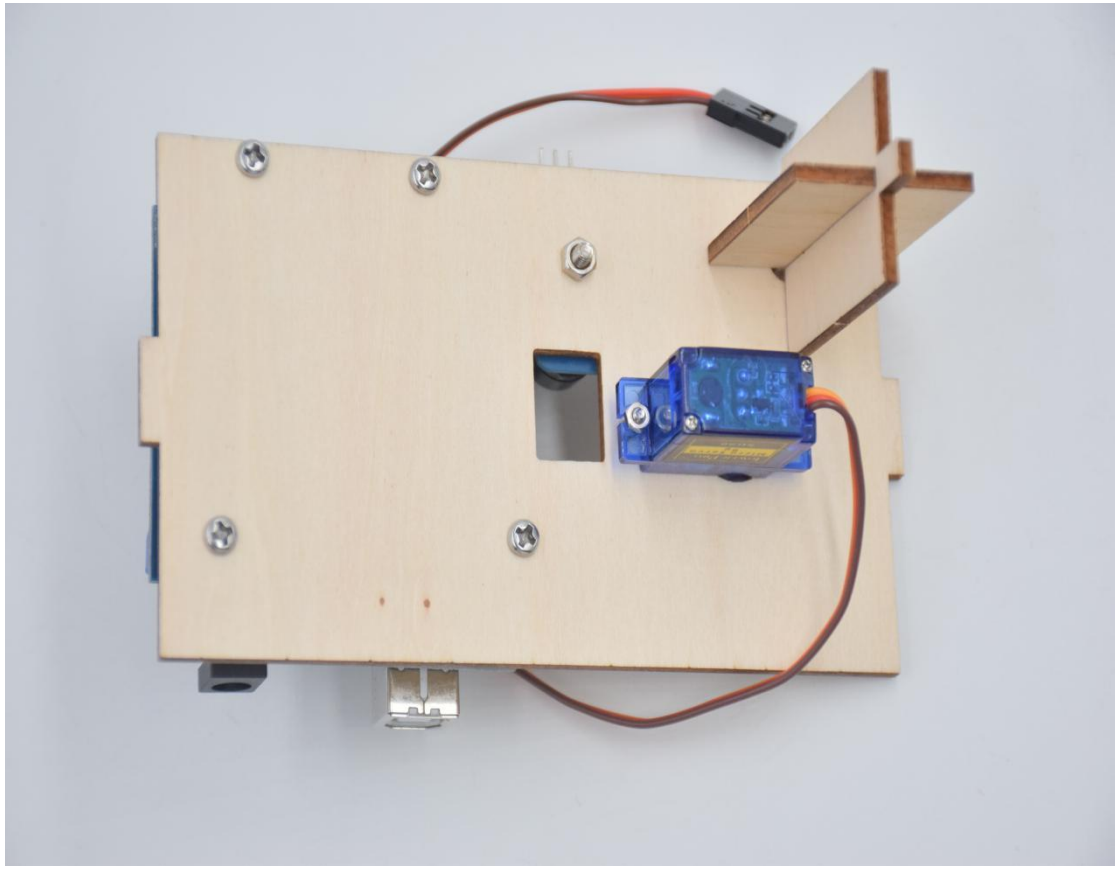


LROBRUYA

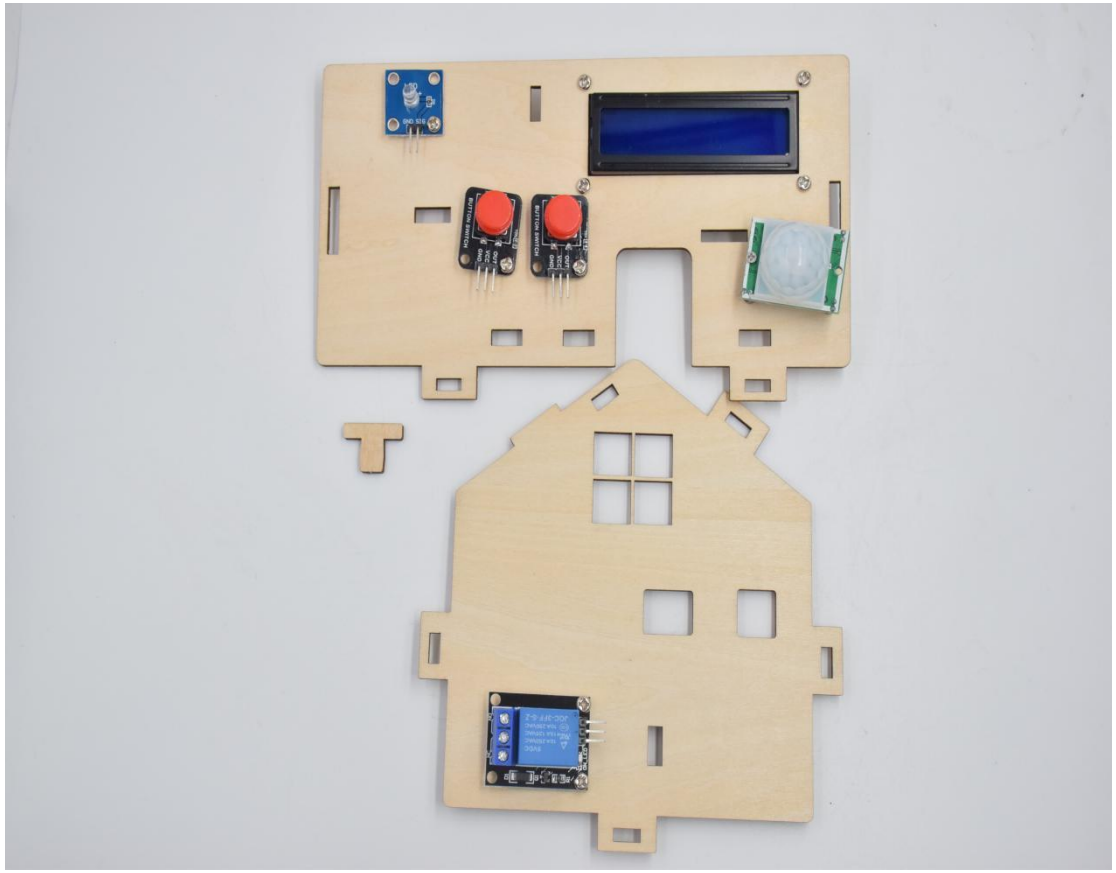
LROBRUYA


LCD-Serial
LCD-Parallel
Sensor Shieldv5.0
RESET
POWER
ANALOG IN
PWR
SEL
GND
VCC

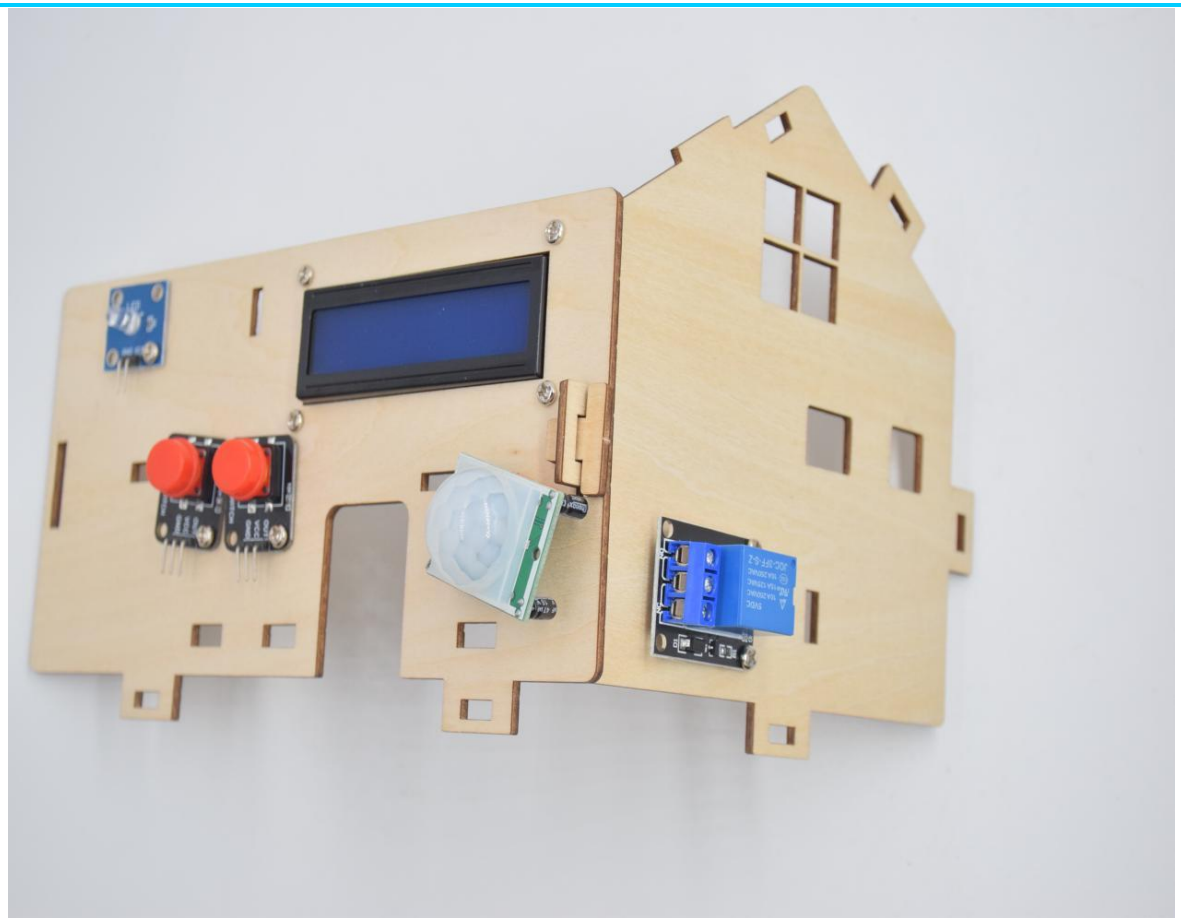
LROBRUYA



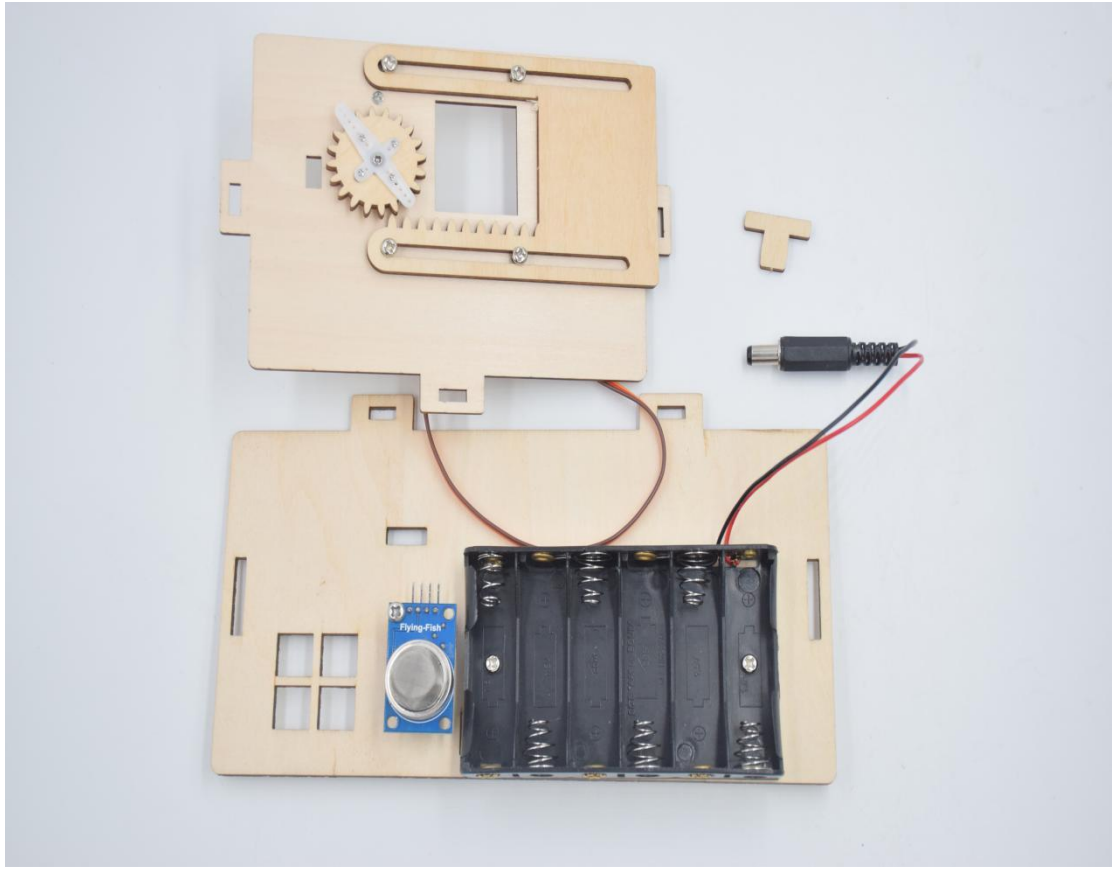
Step 7: Fix A board and B board together with a “T” bolt.



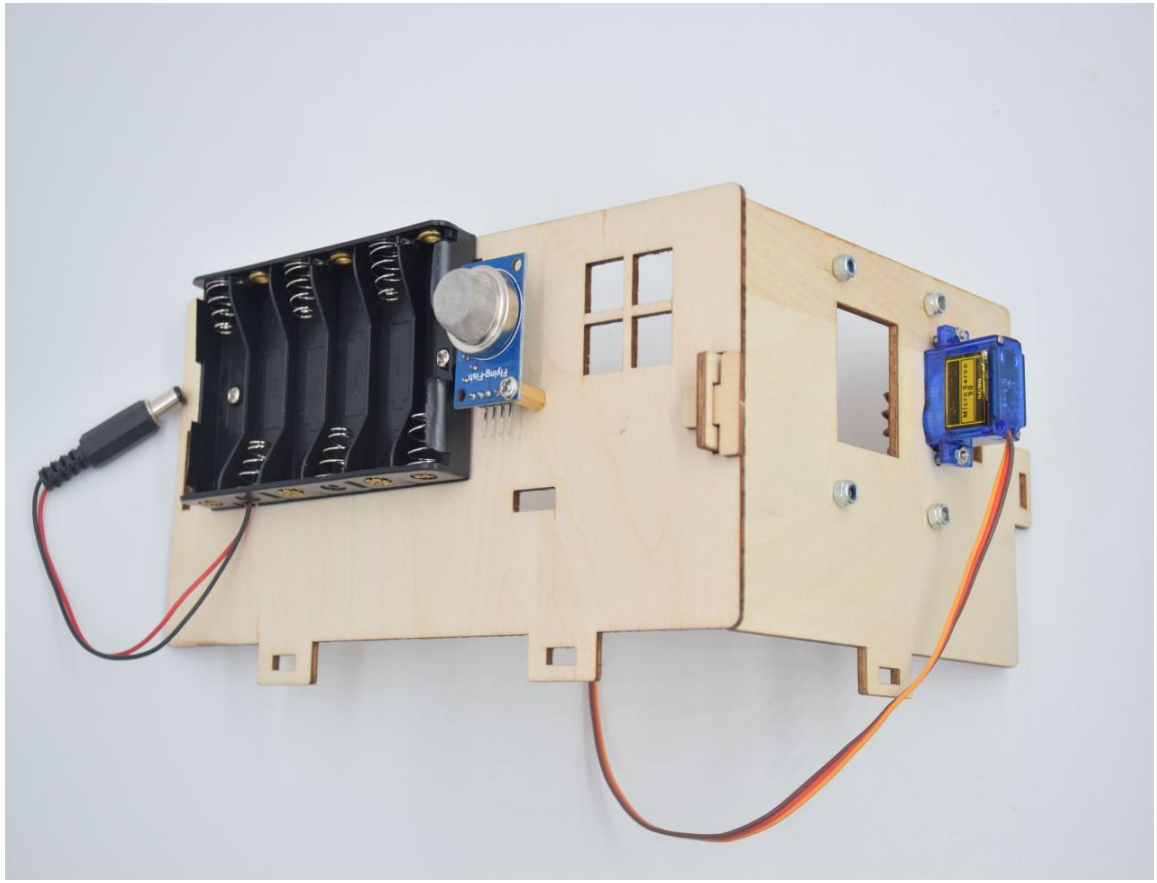
LROBRUYA



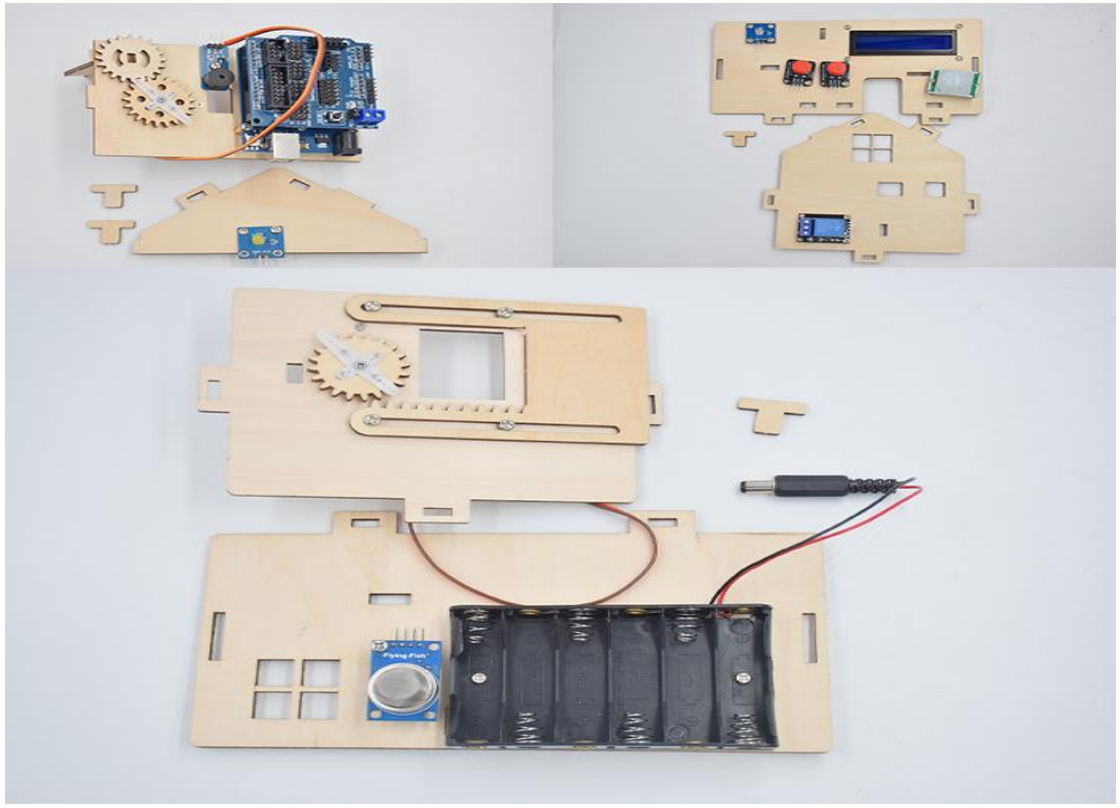
Step 8: Assemble C board with D board by a “T” type bolt.



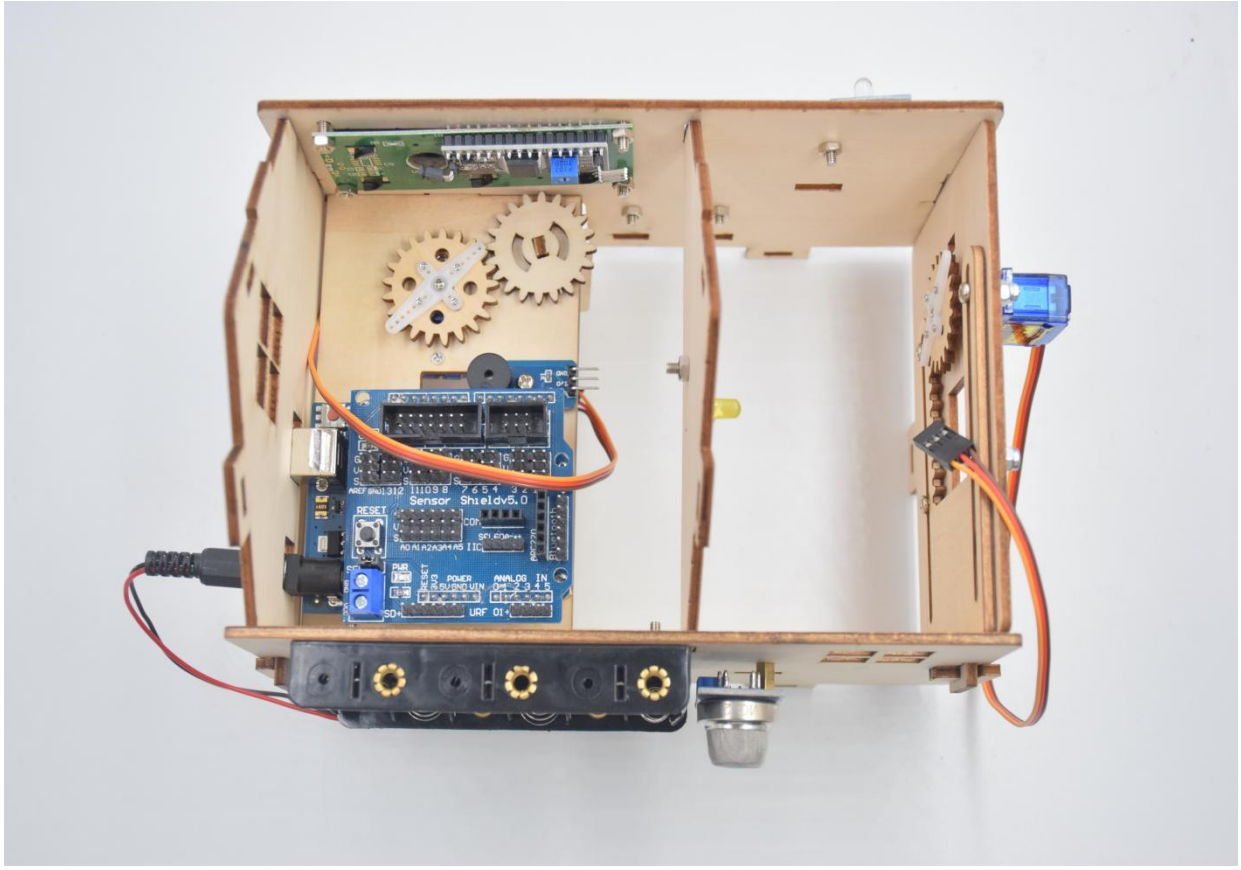
LROBRUYA



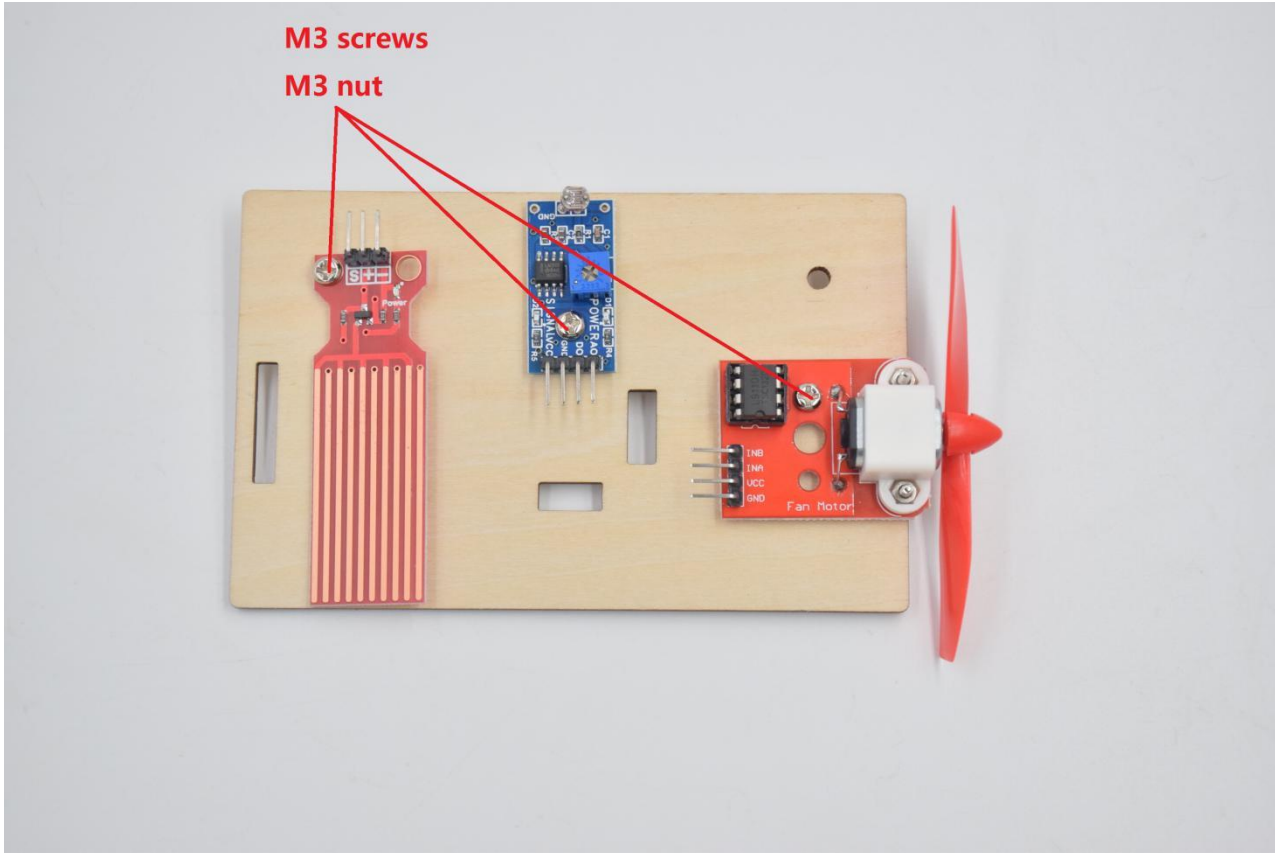
Step 9: Assemble A, B, C, D, E and H board together, then fix them with 2 “T” type bolts.



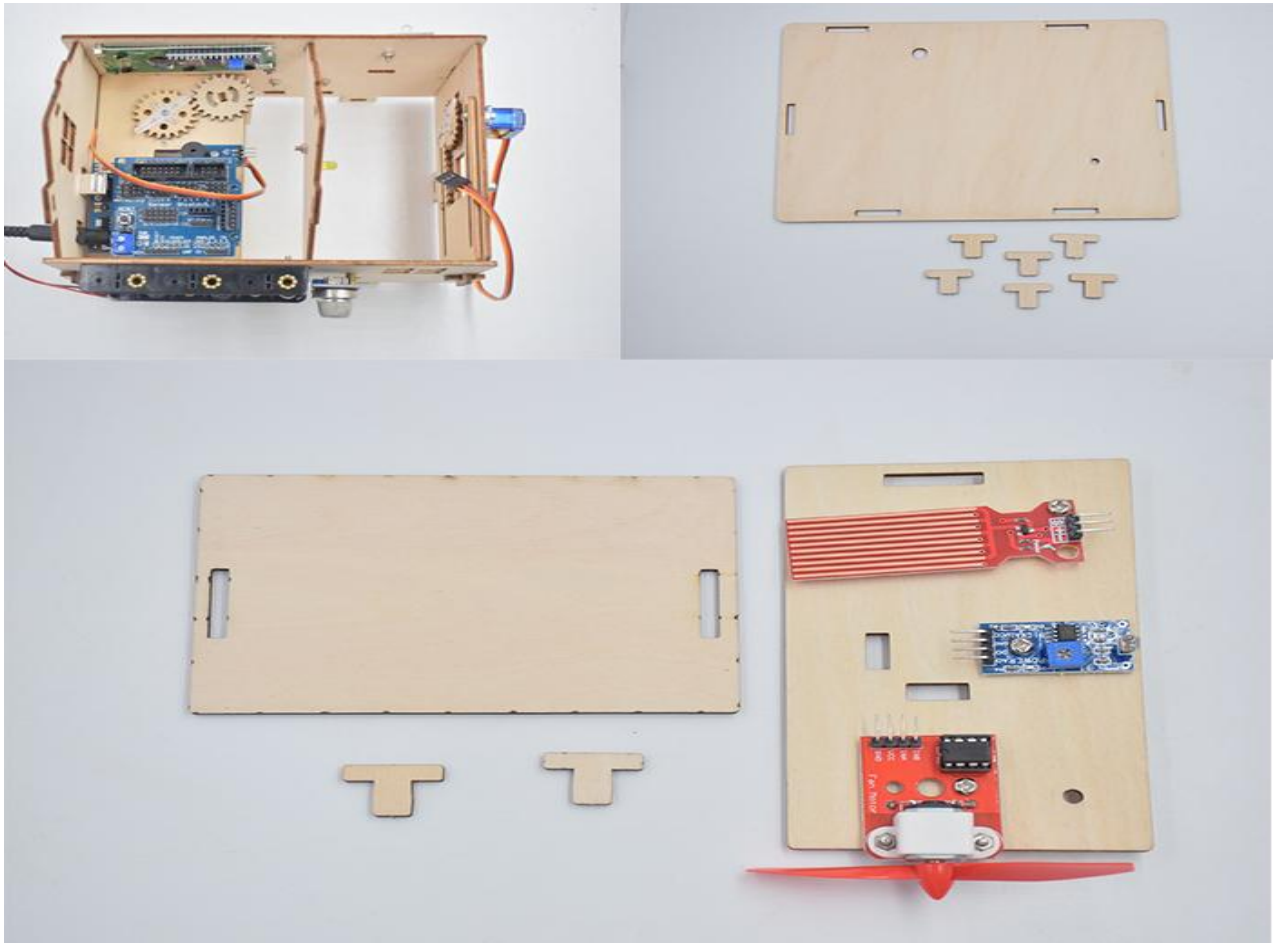
LROBRUYA



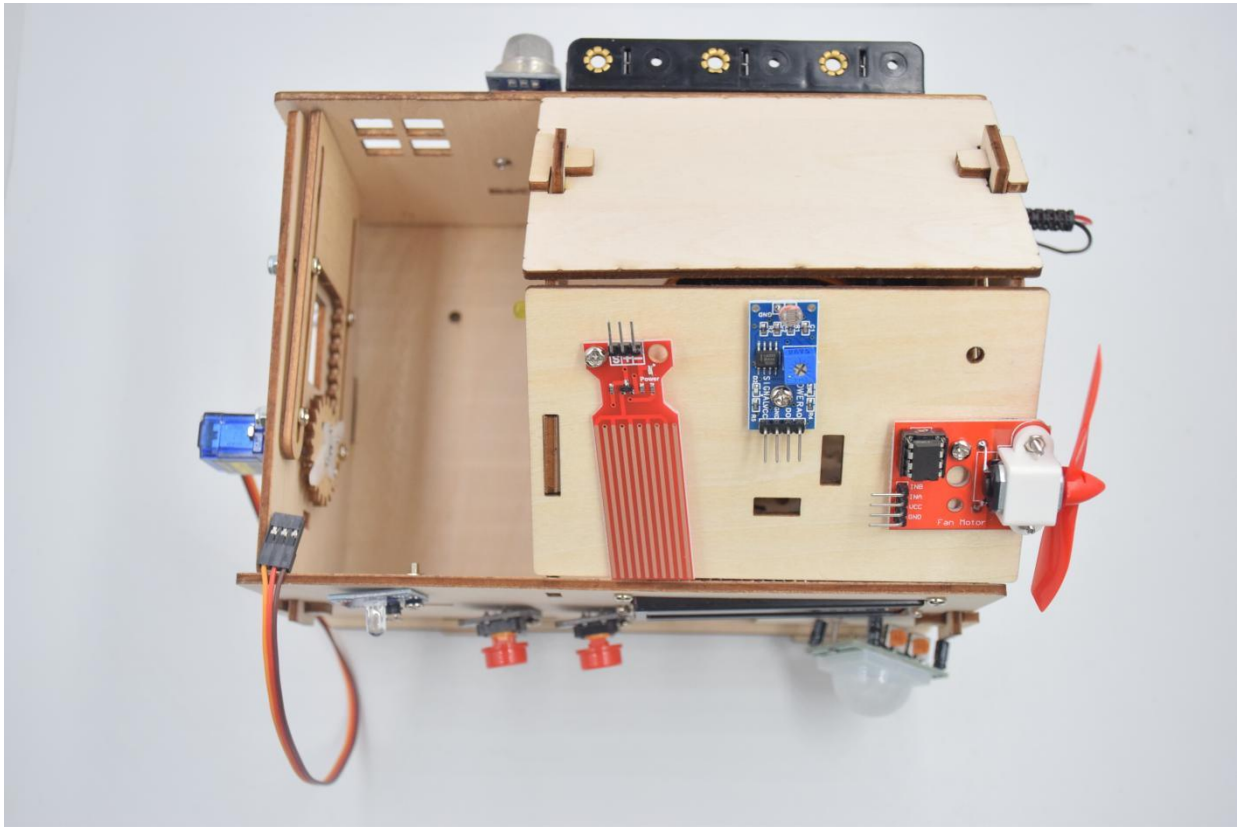
Step 10: Install the sensor of G board



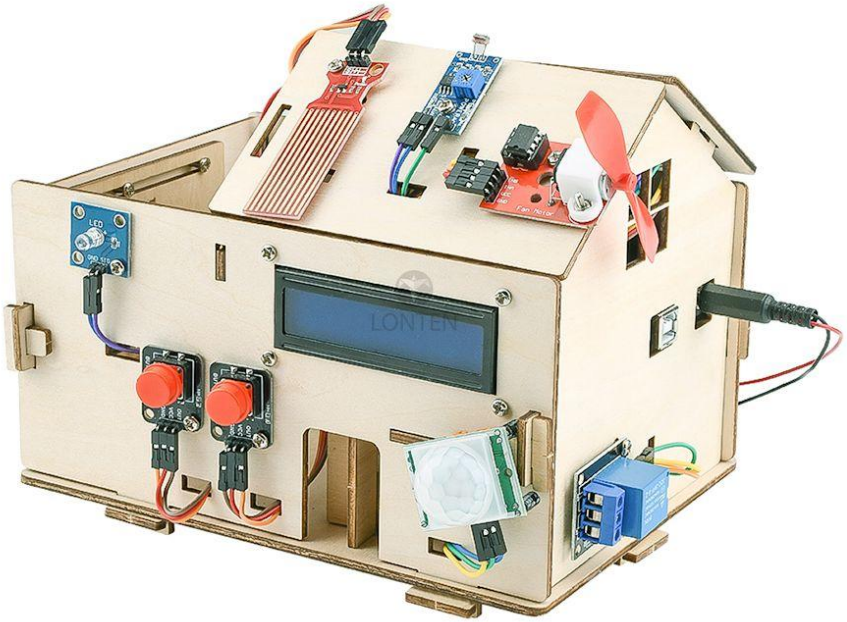
LROBRUYA



LROBRUYA



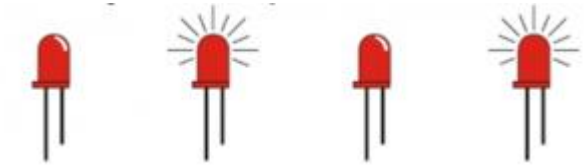
LROBRUYA



LROBRUYA

Project 1：LED Blink

Description：



we start from simple projects. For this project we will perform “Arduion blinks LED”, which is the basic practice for starter. We provide a test code to control LED to perform blinking effect. In the code, you could set distinct flashing scene by changing the time of lighting on and off. Power on GND and VCC, the LED will light on when signal end S is high level, on the contrary, LED will turn off when signal end S is low level.

In addition, the different blinking frequency can be presented by adjusting the delayed time.

Specifications：

- Control interface: digital port
- Working voltage: DC 3.3-5V
- Pin pitch: 2.54mm
- LED display color: white
- Size: 30 * 20mm
- Weight: 3g

Materials

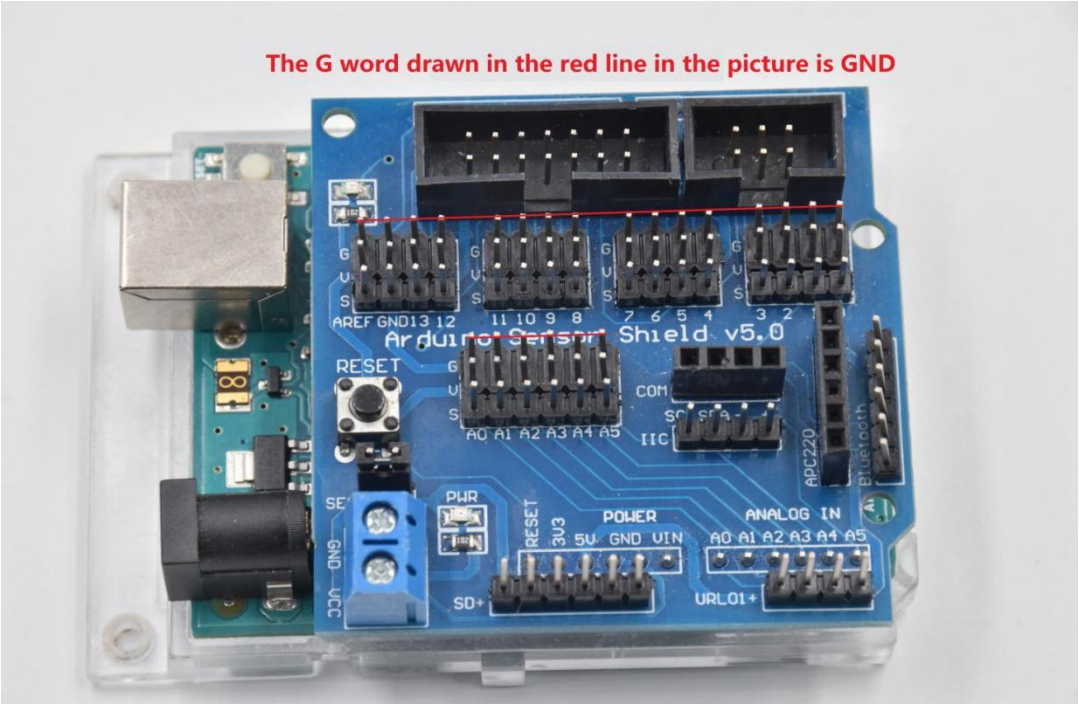
UNO R3 control Board	Sensor Shield v5.0	White LED Module	Female to Female Dupont wire
			

Sensor Shield

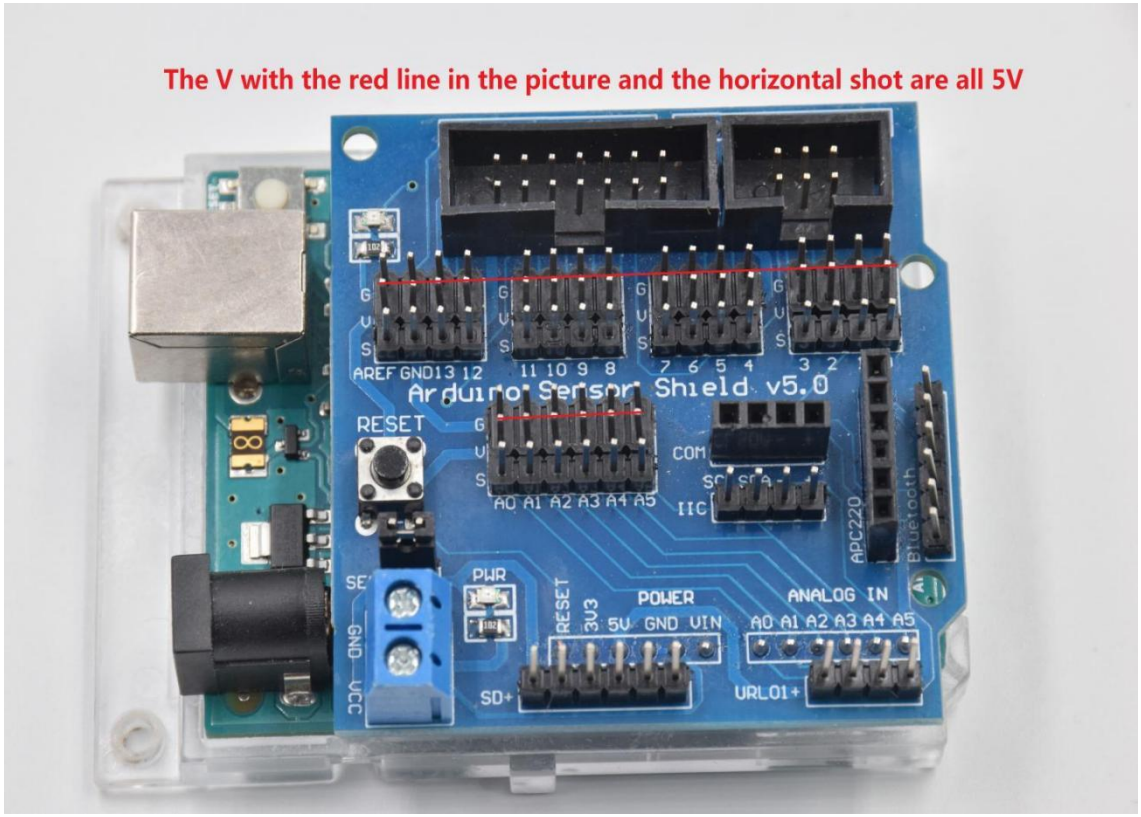
We usually combine Arduino control board with a large number of sensors and modules. However, the pins and ports are limited on control board.

To cope with this disadvantage, we just need to stack V5 sensor board on LONTEN control board.

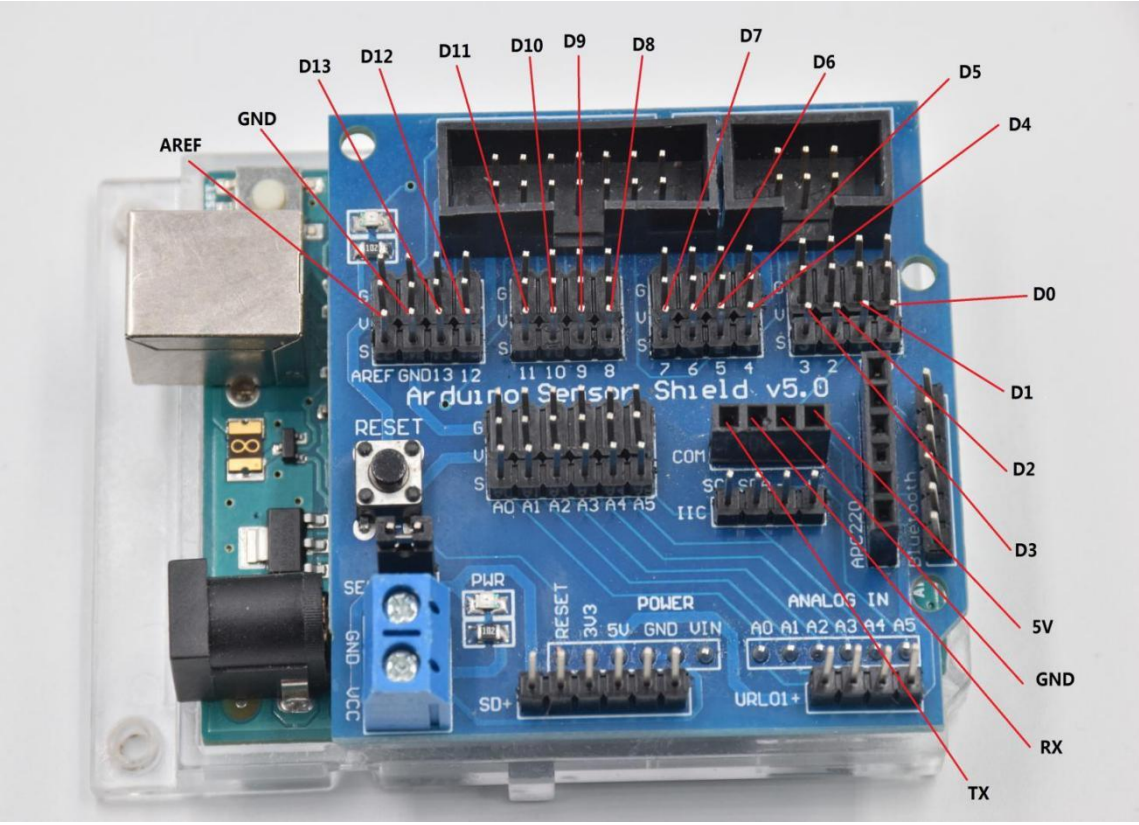
This V5 shield can be directly attached to sensors with 3 pin connectors, and be extended the commonly used communication ports as well, such as serial communication, IIC communication and SPI communication ports. What's more,the shield comes with a reset button and 2 signal lights.

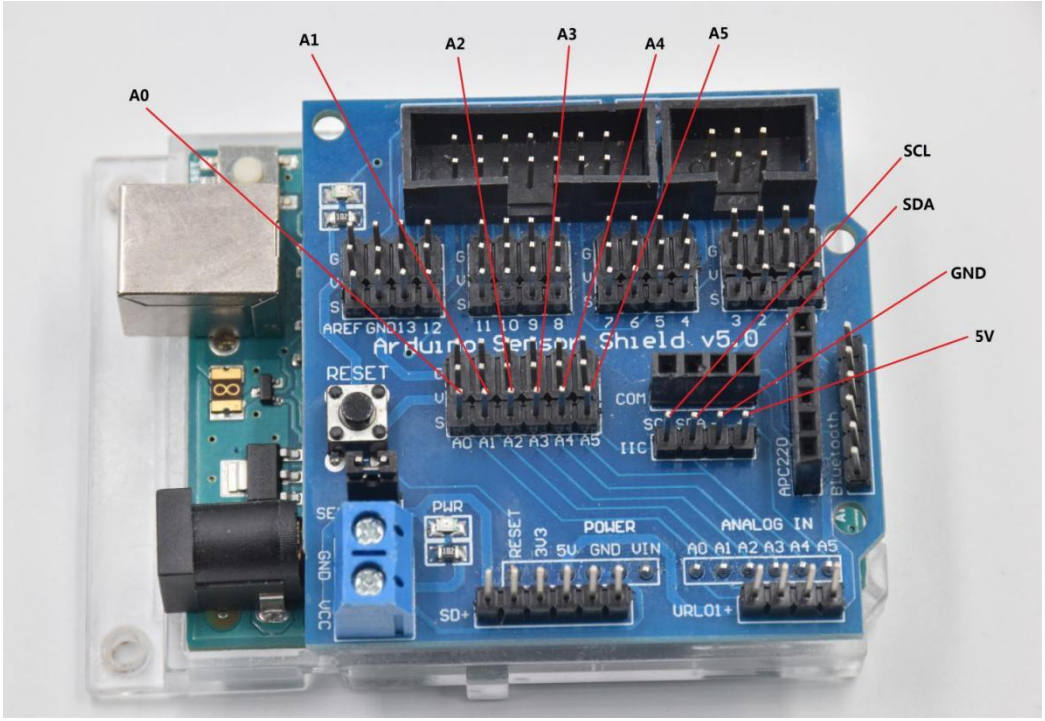


LROBRUYA


The V with the red line in the picture and the horizontal shot are all 5V
AREF GND13 12 11 10 9 8 7 6 5 4 3 2
Arduino Sensor Shield v5.0
RESET
A0 A1 A2 A3 A4 A5
COM
IIC
PWR
POWER
3V3 5V GND VIN
ANALOG IN
A0 A1 A2 A3 A4 A5
SD+
URL01+
APC220
Bluetooth

LROBRUYA


D13
D12
D11
D10
D9
D8
D7
D6
D5
D4
D0
D1
D2
D3
5V
GND
RX
TX
GND
AREF
Arduino Sensor Shield v5.0
RESET
PWR
POWER
ANALOG IN
COM
IIC
SD+
URL01+

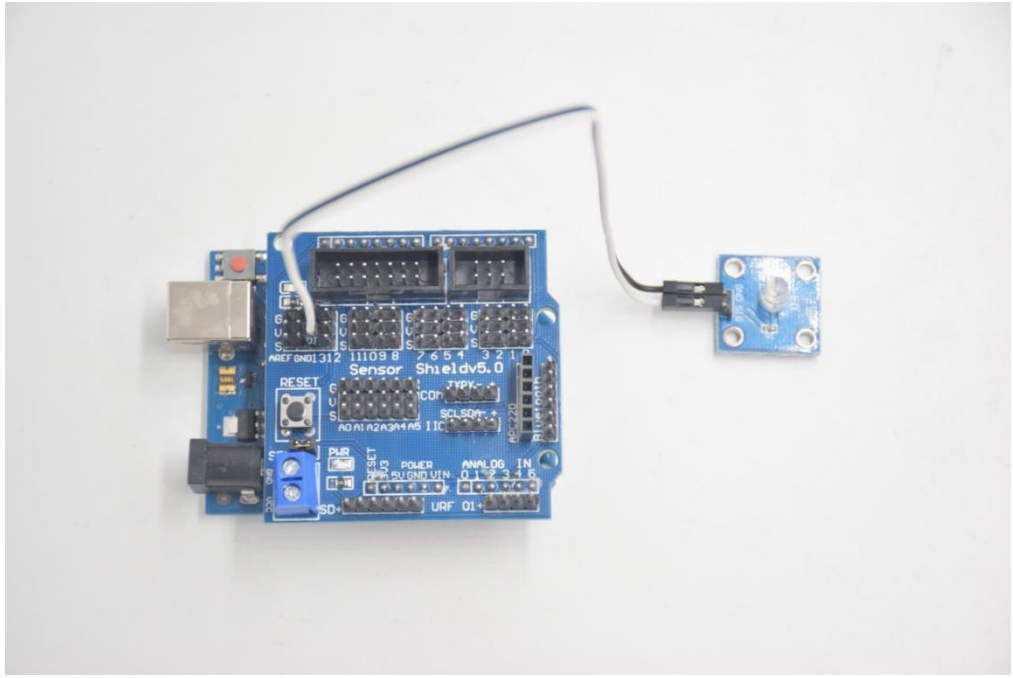


Connection

White LED:

GND -- GND

SIG -- D13



Test Result:

Upload test code successfully, white LED starts blinking, lights on for 1000ms, lights off for 1000ms, alternately.

Code analysis

The code looks long and clutter, but most of which are comment. The grammar of Arduino is based on C.

Comments generally have two forms of expression:

/* .....*/ : suitable for long paragraph comments

// : suitable for mono line comments

So the code contains the many vital information, such as the author, the issued agreement, etc.

Most people omit comments, starter should develop a good habit of looking through code. Firstly, check comments. They contain the provided information and do help you understand test code quickly. Secondly, form the habit of writing comments

```
// the setup function runs once when you press reset or power the board

void setup() {
```

```
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}
```

According to comments, we will find that author define the D13 pin mode as digital output in setup() function. Setup() is the basic function of Arduino. It will execute once in the running of program, usually as definition pin, define and ensure the variables.

```
// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
```

```
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

Loop() is the necessary function of Arduino, it can run and loop all the time after “setup()” executes once

In the loop()function.

digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)

digitalWrite(): set the output voltage of pin to high or low level. We make D13 output high level, then the LED lights on.

delay(1000); // wait for a second

Delay function is used for delaying time, 1000ms is 1s, unit is ms

```
digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
```

Similarly, we make D13 output low level, LED will turn off.

```
delay(1000); // wait for a second
```

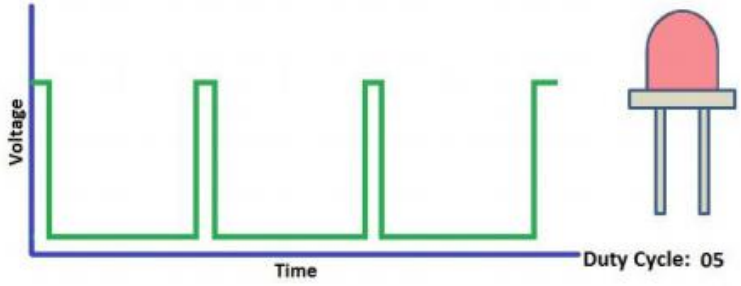
Delay for 1s, light on LED--keep on 1s--light off LED--stay on 1s, iterate the process. LED flashes with 1-second interval. What if you want to make LED flash rapidly? You only need to modify the value of delay block. Reducing the delay value implies that the time you wait is shorter, that is, flashing rapidly. Conversely, you could make LED flash slowly.

LROBRUYA

Project 2： Breathing Light

Breathing light

Description



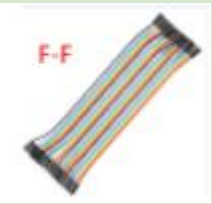
In the previous project, we control LED on and off and make it blink. In this project, we will control LED brightness through PWM to simulate breathing effect. Similarly, you can change the step length and delay time in the code so as to demonstrate different breathing effect.

PWM is a means of controlling the analog output via digital means. Digital control is used to generate square waves with different duty cycles (a signal that constantly switches between high and low levels) to control the analog output.In general, the input voltage of port are 0V and 5V. What if the 3V is required? Or what if switch among 1V, 3V and 3.5V? We can't change resistor constantly. For this situation, we need to control by PWM.

For the Arduino digital port voltage output, there are only LOW and HIGH, which correspond to the voltage output of 0V and 5V. You can define LOW as 0 and HIGH as 1, and let the Arduino output five hundred 0 or 1 signals within 1 second. If output five hundred 1, that is 5V; if all of which is 1, that is 0V. If output 010101010101 in this way then the output port is 2.5V, which is like showing movie. The movie we watch are not completely continuous. It actually outputs 25 pictures

per second. In this case, the human can't tell it, neither does PWM. If want different voltage, need to control the ratio of 0 and 1. The more 0,1 signals output per unit time, the more accurately control.

Materials:

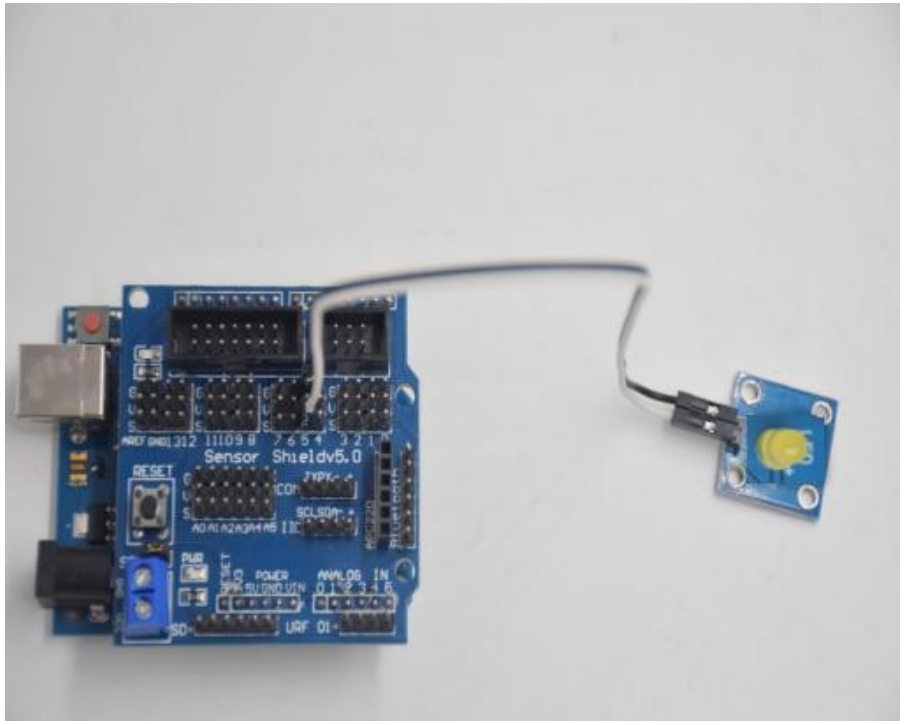
UNO R3 control Board	Sensor Shield v5.0	Yellow LED Module	Female to Female Dupont wire
			

Connection:

Yellow LED:

GND --- GND

SIG --- D5

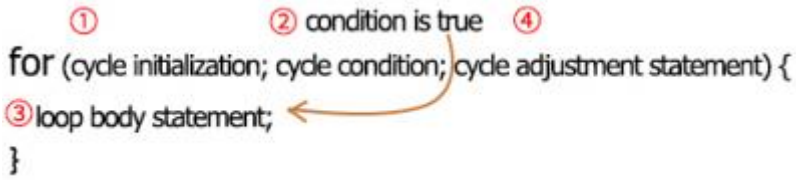


Test Result：

LED smoothly changes its brightness from dark to bright and back to dark, continuing to do so, which is similar to a lung breathing in and out.

Code analysis

When we need to repeat some statements, we have to use “for” statement For statement format as follows:



“for” cyclic sequence:

Round 1：1 → 2 → 3 → 4

Round 2：2 → 3 → 4

… Until number 2 is not established, “for”loop is over,

After knowing this order, go back to code:

```
for (int value = 0; value < 255; value=value+1){
        ...
}
for (int value = 255; value >0; value=value-1){
        ...
}
```

The two “for”statement make value increase from 0 to 255, then reduce from 255 to 0, then increase to 255,....infinite loop

There is a new function in “for” statement ----- analogWrite()

We know that digital port only has two state of 0 and 1. So how to send an analog value to a digital value? Here, we need this function, observe the Arduino board and you will find 6 pins with “~”. They are different from other pins and can

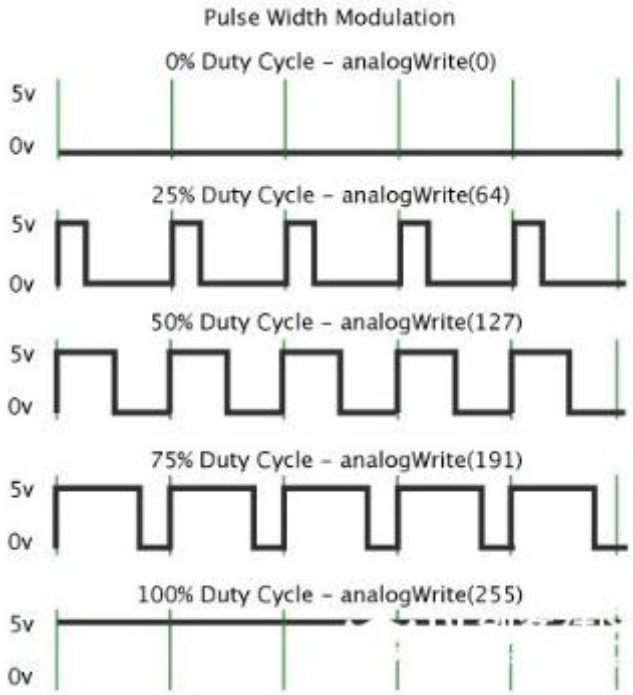
output PWM signals.

Function format as follows:

analogWrite(pin,value) analogWrite() is used to write an analog value from 0~255 for PWM port, so the value is in the range of 0~255, attention that you only write the digital pins with PWM function, such as pin 3, 5, 6, 9, 10, 11.

PWM is a technology to obtain analog quantity through digital method. Digital control forms a square wave, and the square wave signal only has two states of switching (that is, high or low levels of our digital pins). By controlling the ratio of the duration of on and off, a voltage varying from 0 to 5V can be simulated. The time taken(academically referred to as high level) is called pulse width, so PWM is also called pulse width modulation.

Through the following five square waves, let’s know more about PWM.



In the above figure, the green line represents a period, and value of analogWrite() corresponds to a percentage which is called Duty Cycle as well. Duty cycle implies that high-level duration is divided by low-level duration in a cycle. From top

to bottom, the duty cycle of first square wave is 0% and its corresponding value is 0. The LED brightness is lowest, that is, turn off. The more time high level lasts, the brighter the LED. Therefore, the last duty cycle is 100%, which correspond to 255, LED is brightest. 25% means darker.

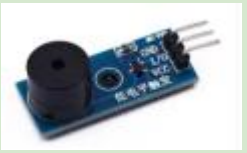
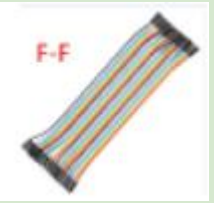
Project 3： Passive Buzzer

Description

There are prolific interactive works completed by Arduino. The most common one is sound and light display. We always use LED to make experiments. For this project, we design circuit to emit sound. The universal sound components are buzzer and horns. Buzzer is easier to use. And buzzer includes about active buzzer and passive buzzer. In this experiment, we adopt passive buzzer. While using passive buzzer, we can control different sound by inputting square waves with distinct frequency. During the experiment, we control code to make buzzer sound, begin with “tick, tick” sound, then make passive

buzzer emit “do re mi fa so la si do”, and play specific songs.

Materials:

UNO R3 control Board	Sensor Shield v5.0	Passive Buzzer Sensor	Female to Female Dupont wire
			

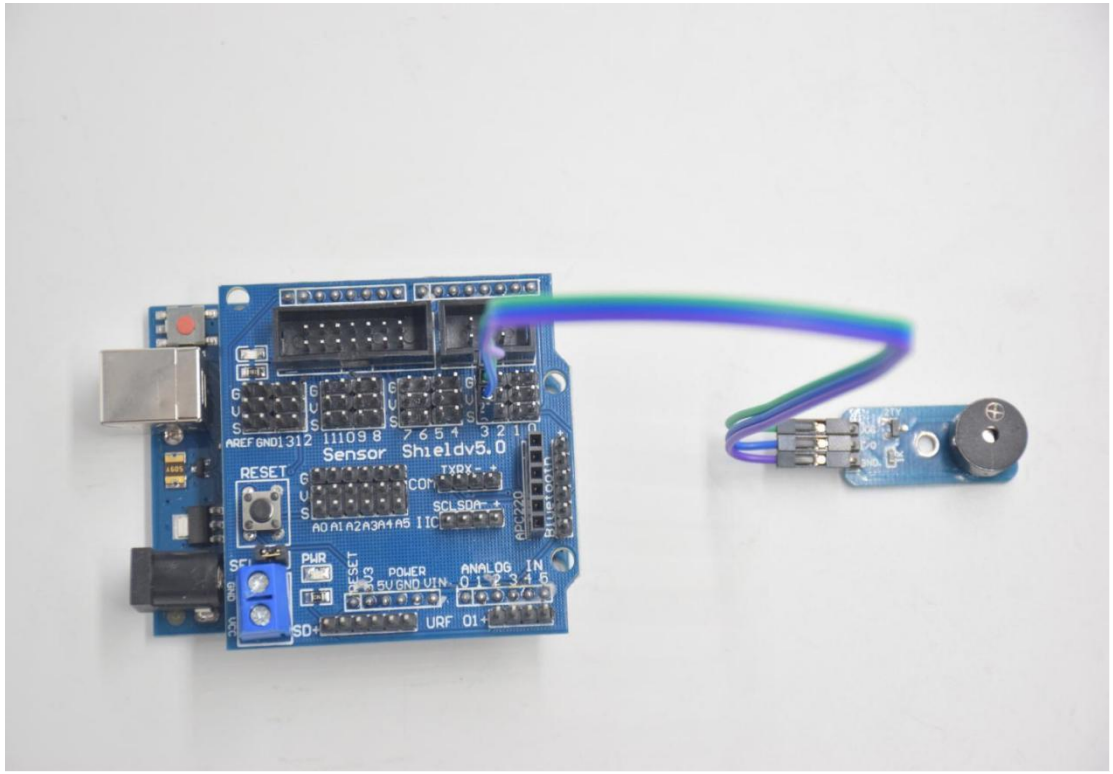
Connection:

Passive Buzzer:

VCC --- 5V

I/O --- D3

GND --- GND



Test Result:

From the above code, 80 and 100 decide frequency in “for” statement. Delay controls duration, like the beat in music.

We will play fabulous music if we control frequency and beats well, so let’s figure out the frequency of tones. As shown below:

LROBRUYA

Bass:

Tone Note	1#	2#	3#	4#	5#	6#	7#
A	221	248	278	294	330	371	416
B	248	278	294	330	371	416	467
C	131	147	165	175	196	221	248
D	147	165	175	196	221	248	278
E	165	175	196	221	248	278	312
F	175	196	221	234	262	294	330
G	196	221	234	262	294	330	371

LROBRUYA

Alto:

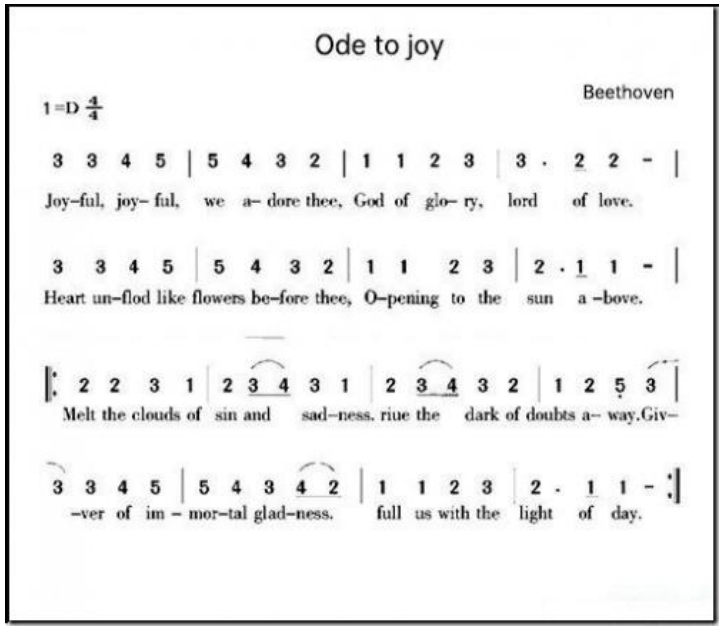
Tone Note	1	2	3	4	5	6	7
A	441	495	556	589	661	742	833
B	495	556	624	661	742	833	935
C	262	294	330	350	393	441	495
D	294	330	350	393	441	495	556
E	330	350	393	441	495	556	624
F	350	393	441	495	556	624	661
G	393	441	495	556	624	661	742

LROBRUYA

Treble:

Tone Note	1#	2#	3#	4#	5#	6#	7#
A	882	990	1112	1178	1322	1484	1665
B	990	1112	1178	1322	1484	1665	1869
C	525	589	661	700	786	882	990
D	589	661	700	786	882	990	1112
E	661	700	786	882	990	1112	1248
F	700	786	882	935	1049	1178	1322
G	786	882	990	1049	1178	1322	1484

After knowing the frequency of tone, next to control the time the note plays. The music will be produces when every note plays a certain amount of time. The note rhythm is divided into one beat, half beat, 1/4 beat, 1/8 beat, we stipulate the time for a note to be 1, half beat is 0.5, 1/4 beat is 0.25, 1/8 beat is 0.125....., Therefore, the music is played. We will take example of “Ode to joy”



From notation, the music is 4/4 beat.

There are special notes we need to explain:

1.Normal note, like the first note 3, correspond to 350(frequency), occupy 1 beat

2.The note with underline means 0.5 beat

3.The note with dot(3.)means that 0.5 beat is added, that is 1+0.5 beat

4.The note with"—" represents that 1 beat is added, that is 1+1 beat.

5.The two successive notes with arc imply legato, you could slightly modify the frequency of the note behind legato(need to debug it yourself), such like reducing or increasing some values, the sound will be more smoother.

Upload test code on the development board.

We will hear "Ode to joy".

Project 4： Controlling LED By Button Module

Description：

In this project, we will control LED to light on and off via button module.The button switch is ordinary in our life. It belongs to switch quantity(digital quantity)components. Composed of normally open contact and normally closed contact, it is similar to ordinary switch.

When the button is pressed, the signal end outputs low level (0); when released, the signal end of sensor keeps high level(1).

LROBRUYA

Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Yellow LED Module	Button Sensor
				

Connection:

key module：

GND -- GND

VCC -- 5V

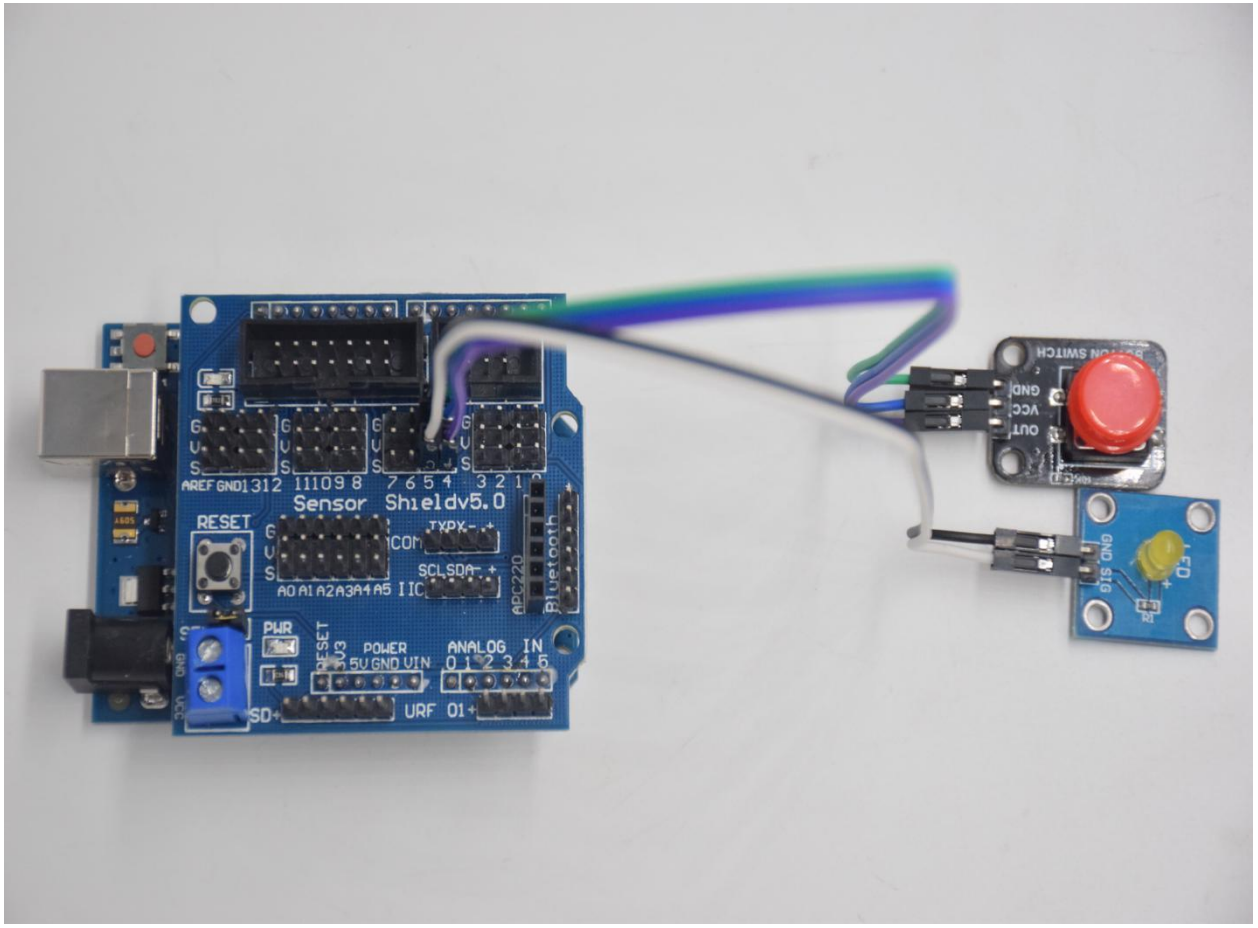
OUT -- D4

LED:

GND -- GND

SIG -- D5

LROBRUYA


Sensor Shieldv5.0
RESET
PWR
POWER
ANALOG IN
SD+
URF
Bluetooth
APC220
LED
GND
SIG
VCC
OUT

Test Result:

This experiment is pretty simple, and widely applied to various of circuits and electrical appliances. In our life, you could find this principle on any device, such as the backlight is on when press any buttons, which is the typical appliance.

Code analysis:

we make LED on by button. Comparing with previous experiments, we add a conditional judgement statement. We use if statement. The written sentences of Arduino is based on C language, therefore, the condition judgement statement of C is suitable for Arduino, like while, swich, etc.

For this project, we take simple “if” statement as example to demonstrate:

If button is pressed, digital 4 is low level, then we make digital 5 output high level , then LED will be on; conversely, if the button is released, digital 4 is high level, we make digital 5 output low level, then LED will go off.

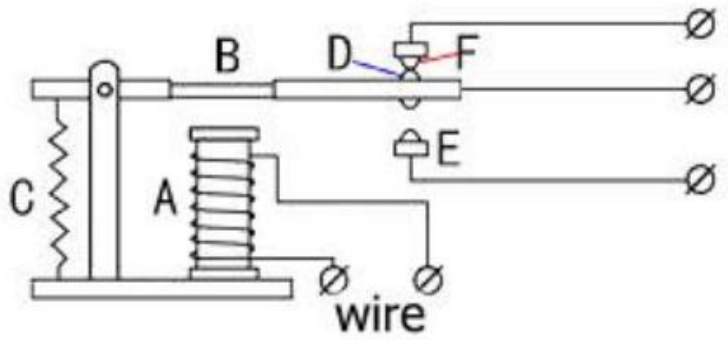
Project 5：1-channel Relay Module

Description

This module is an Arduino dedicated module, and compatible with arduino sensor expansion board. It has a control system (also called an input loop) and a controlled system (also called an output loop). Commonly used in automatic control circuits, the relay module is an "automatic switch" that controls a larger current and a lower voltage with a smaller current and a lower voltage.

Therefore, it plays the role of automatic adjustment, safety protection and conversion circuit in the circuit. It allows Arduino to drive loads below 3A, such as LED light strips, DC motors, miniature water pumps, solenoid valve pluggable interface.

The main internal components of the relay module are electromagnet A, armature B, spring C, moving contact D, static contact (normally open contact) E, and static contact (normally closed contact) F, (as shown in the figure).



As long as a certain voltage is applied to both ends of the coil, a certain current will flow through the coil to generate electromagnetic effects, and the armature will attract the iron core against the pulling force of the return spring under the action of electromagnetic force attraction, thereby driving the moving contact and the static contact (normally open contact) to attract. When the coil is disconnected, the electromagnetic suction will also disappear, and the armature will return to the original position under the reaction force of the spring, releasing the moving contact and the original static contact (normally closed contact). This pulls in and releases, thus achieving the purpose of turning on and off in the circuit. The

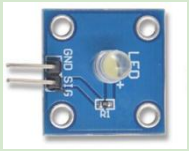
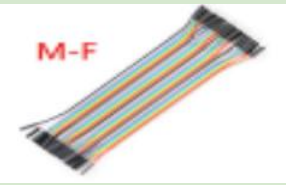
"normally open and closed" contacts of the relay can be distinguished in this way: the static contacts on disconnected state when the relay coil is powered off are called "normally open contacts"; the static contacts on connected state are called "normally closed contact". The module comes with 2 positioning holes for you to fix the module to other equipment.

Specifications：

- Working voltage: 5V (DC)
- Interface: -, +, S interface
- Input signal: digital signal (high level 1, low level 0)
- Contacts: static contacts (normally open contacts, normally closed contacts) and moving contacts
- Rated current: 10A (NO) 5A (NC)
- Maximum switching voltage: 150 V (AC) 24 V (DC)
- Electric shock current: less than 3A

- Weight: 15g
- Contact action time: 10ms

Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Relay Module	White LED Module	Male to Female Dupont wire
					

Connection：

Relay：

S -- D12

\+ -- 5V

\- -- GND

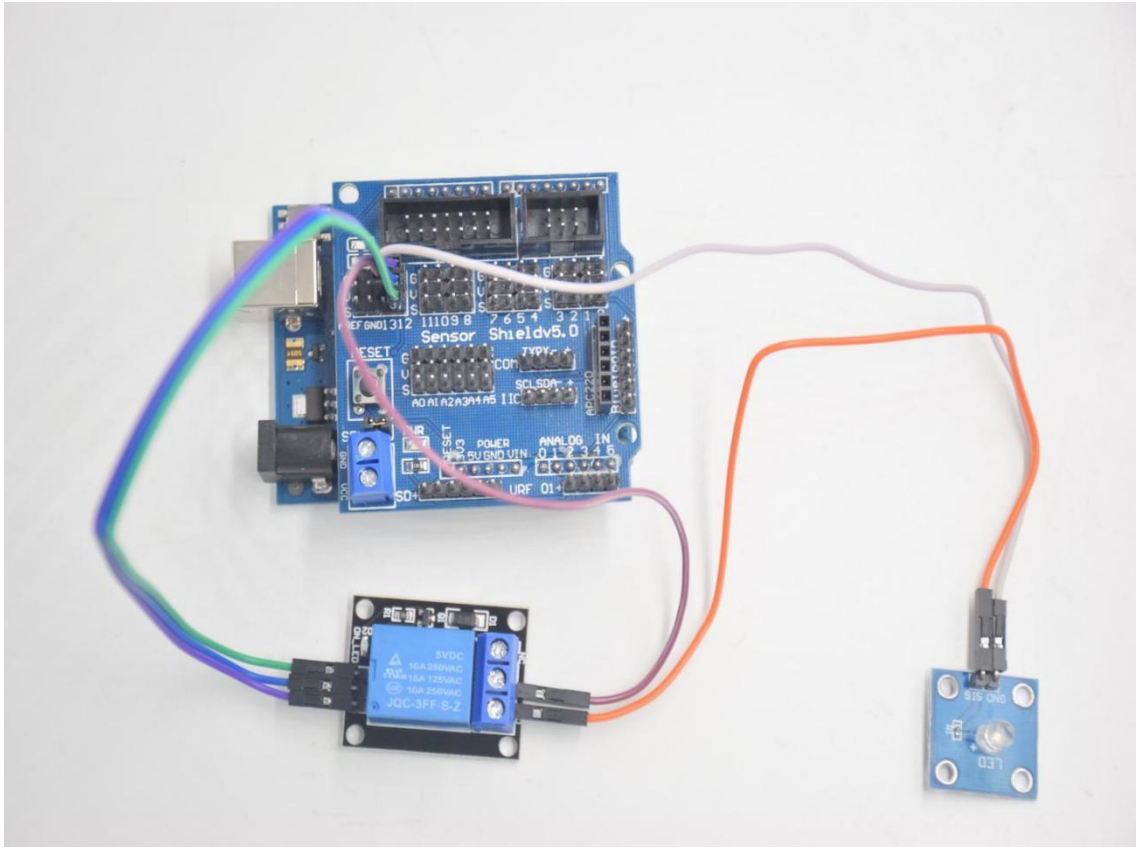
COM -- 5V

LED:

SIG -- NO

GND -- GND

LROBRUYA



Test Result：

Wire, power up and upload test code. The relay is connected(“NO” is on , NC is off) for 0.5s, then disconnected for 0.5s (NC is on, NO is off), and alternately. When the relay is connected, the white LED will be on, conversely, the white LED will go off.

Project 6： Photocell Sensor

Description

The photocell sensor (photoresistor) is a resistor made by the photoelectric effect of a semiconductor. It is very sensitive to ambient light, thus its resistance value vary with different light intensity. We use its features to design a circuit and generate a photoresistor sensor module. The signal end of the module is connected to the analog port of the microcontroller. When the light intensity increases, the resistance decreases, and the voltage of the analog port rises, that is, the analog value of the

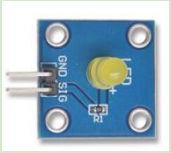
microcontroller also goes up. Otherwise, when the light intensity decreases, the resistance increases, and the voltage of the analog port declines. That is, the analog value of the microcontroller becomes smaller. Therefore, we can use the photoresistor sensor module to read the corresponding analog value and sense the light intensity in the environment. It is commonly applied to light measurement, control and conversion, light control circuit as well.

Specifications：

- Working voltage: 5V (DC)
- Interface: 4PIN interface
- Output signal: analog signal
- Weight: 2.3g

LROBRUYA

Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Photocell	Yellow LED Module
				

Connection：

Photosensitive：

VCC -- 5V

GND -- GND

DO -- A1

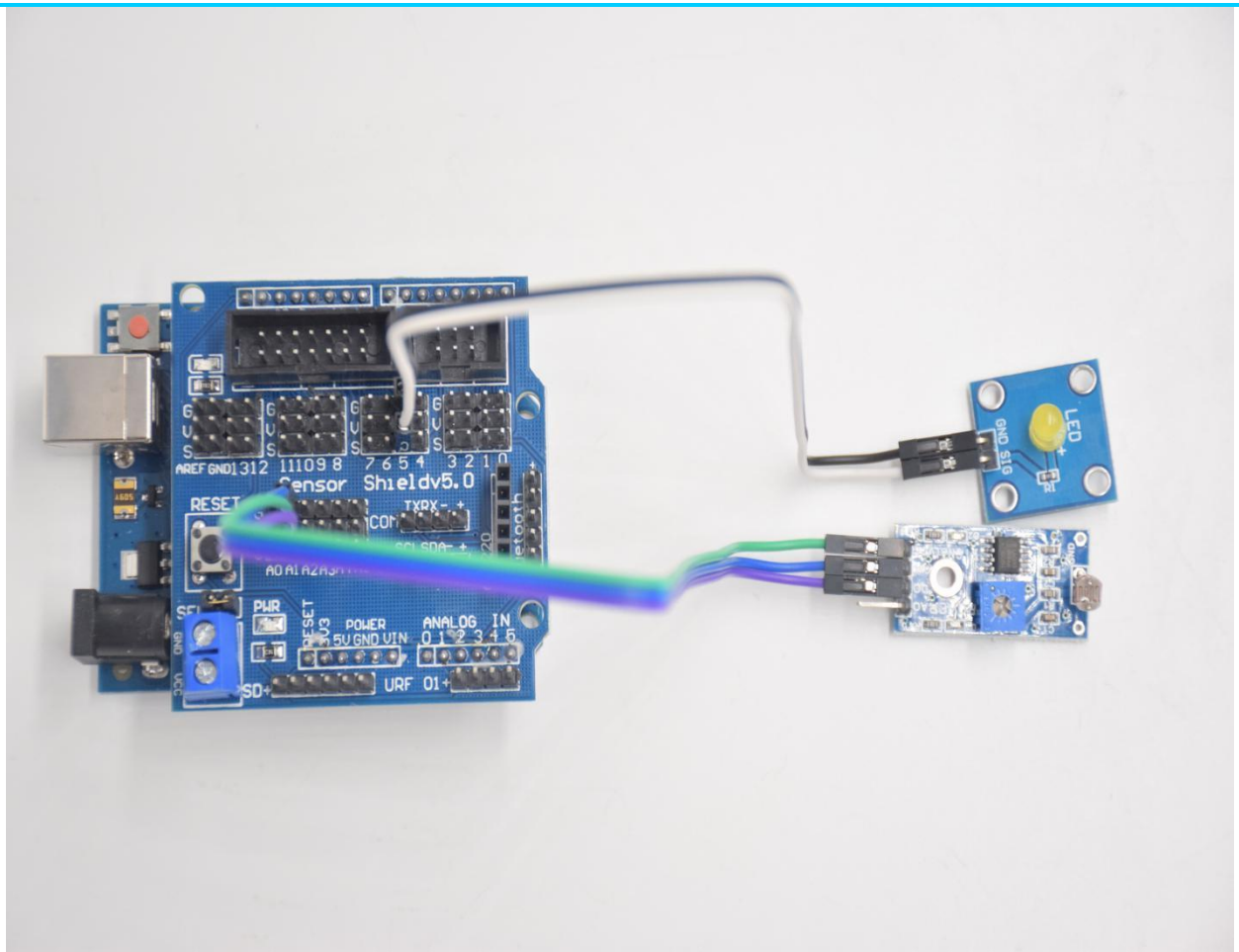
AO -- No Answer

LED:

SIG -- D5

GND -- GND

LROBRUYA



Test Result:

LED will be on after uploading test code, point at the photocell sensor with flashlight (or the flash from cellphone), you'll find that LED is automatically off. However, take away the flashlight, LED will be on again.

Review For this code string, it is simply. We read value through analog port, please attention that analog quantity doesn't need input and output mode.Read the analog value of photocell sensor by analog port.

The analog value will gradually decreases once there is light, the value is up to 1000, this value can be chosen according to brightness you need. Select method: put the whole device in the environment where LED is off, open serial monitor to check shown value, replace 1000 with this value. Read value from serial monitor is a good way to modulate code

Project 7： Adjusting Motor Servo Angle

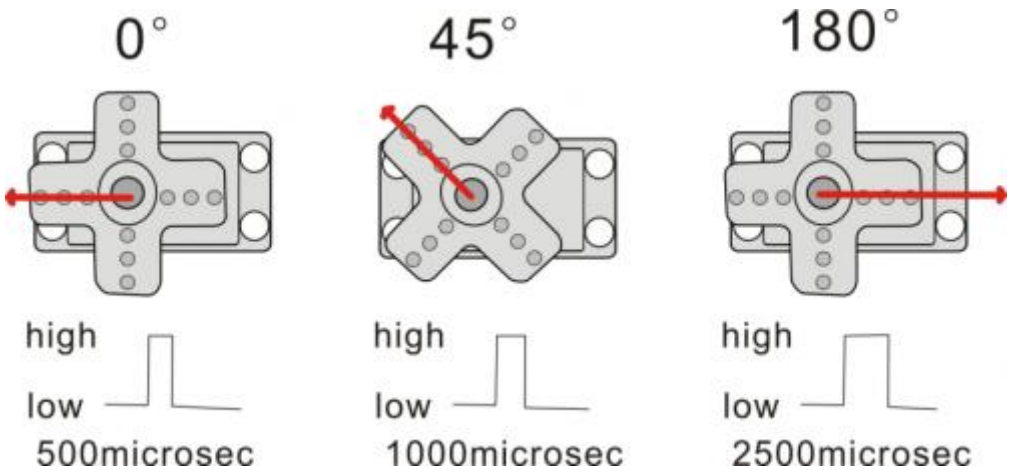
Description

When we make this kit, we often control doors and windows with servos. In this course, we'll introduce its principle and how to use servo motors. Servo motor is a position control rotary actuator. It mainly consists of housing, circuit board, core-less motor, gear and position sensor. Its working principle is that the servo receives the signal sent by MCU or receiver and produces a reference signal with a period of 20ms and width of 1.5ms, then compares the acquired DC bias voltage to the voltage of the potentiometer and outputs a voltage difference.

Servo motor comes with many specifications. But all of them have three connection wires, distinguished by brown, red, orange colors (different brand may have different color). Brown one is for GND, red one for power positive, orange one for signal line.

The rotation angle of servo motor is controlled by regulating the duty cycle of PWM (Pulse-Width Modulation) signal. The

standard cycle of PWM signal is 20ms (50Hz). Theoretically, the width is distributed between 1ms-2ms, but in fact, it's between 0.5ms-2.5ms. The width corresponds the rotation angle from 0° to 180°. But note that for different brand motor, the same signal may have different rotation angle.



There are two ways to control a servomotor with Arduino. One is to use a common digital sensor port of Arduino to produce square wave with different duty cycle to simulate PWM signal and use that signal to control the positioning of the motor. Another way is to directly use the Servo function of the Arduino to control the motor. In this way, the program will be easier

but it can only control two-contact motor because for the servo function, only digital pin 9 and 10 can be used. The Arduino drive capacity is limited. So if you need to control more than one motor, you will need external power.

Specifications:

- Working voltage: DC 4.8V ~ 6V
- Operating angle range: about 180 ° (at 500 → 2500 μsec)
- Pulse width range: 500 → 2500 μsec
- No-load speed: 0.12 ± 0.01 sec / 60 (DC 4.8V) 0.1 ± 0.01 sec / 60 (DC 6V)
- No-load current: 200 ± 20mA (DC 4.8V) 220 ± 20mA (DC 6V)
- Stopping torque: 1.3 ± 0.01kg · cm (DC 4.8V) 1.5 ± 0.1kg · cm (DC 6V)
- Stop current: ≦ 850mA (DC 4.8V) ≦ 1000mA (DC 6V)
- Standby current: 3 ± 1mA (DC 4.8V) 4 ± 1mA (DC 6V)

- Lead length: 250 ± 5 mm Appearance size: 22.9 * 12.2 * 30mm
- Weight: 9 ± 1 g (without servo horn)
- Storage temperature: -20 ℃ ~ 60 ℃
- Operating temperature: -10 ℃ ~ 50 ℃

Materials：

UNO R3 control Board	Sensor Shield v5.0	Servo motor
		

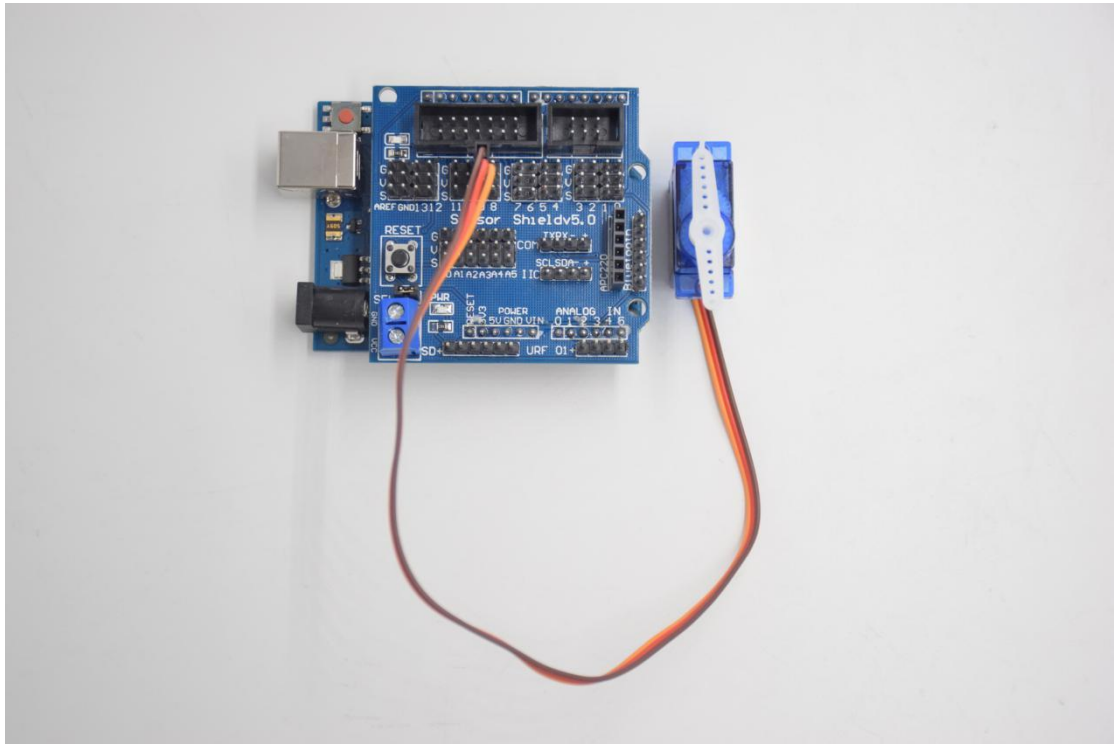
Connection：

Steering gear

Yellow line -- D9

red line -- 5V

Brown thread -- GND



Test Result:

Upload code, wire according to connection diagram, and power on. The servo rotates from 0° to 180° then from 180°~0°

Project 8：Fan Module

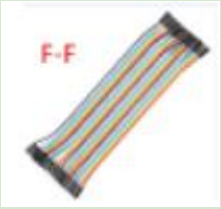
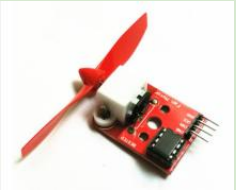
Description

The L9110 fan module adopts L9110 motor control chip, it can control the rotation direction and speed of the motor. Moreover, this module is efficient and with high quality fan, which can put out the flame within 20cm distance. Similarly, it is an important part of fire robot as well.

Specifications:

- Fan diameter: 75mm
- Working voltage: 5V

Materials：

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Fan Module
			

Connection：

Motor fan module

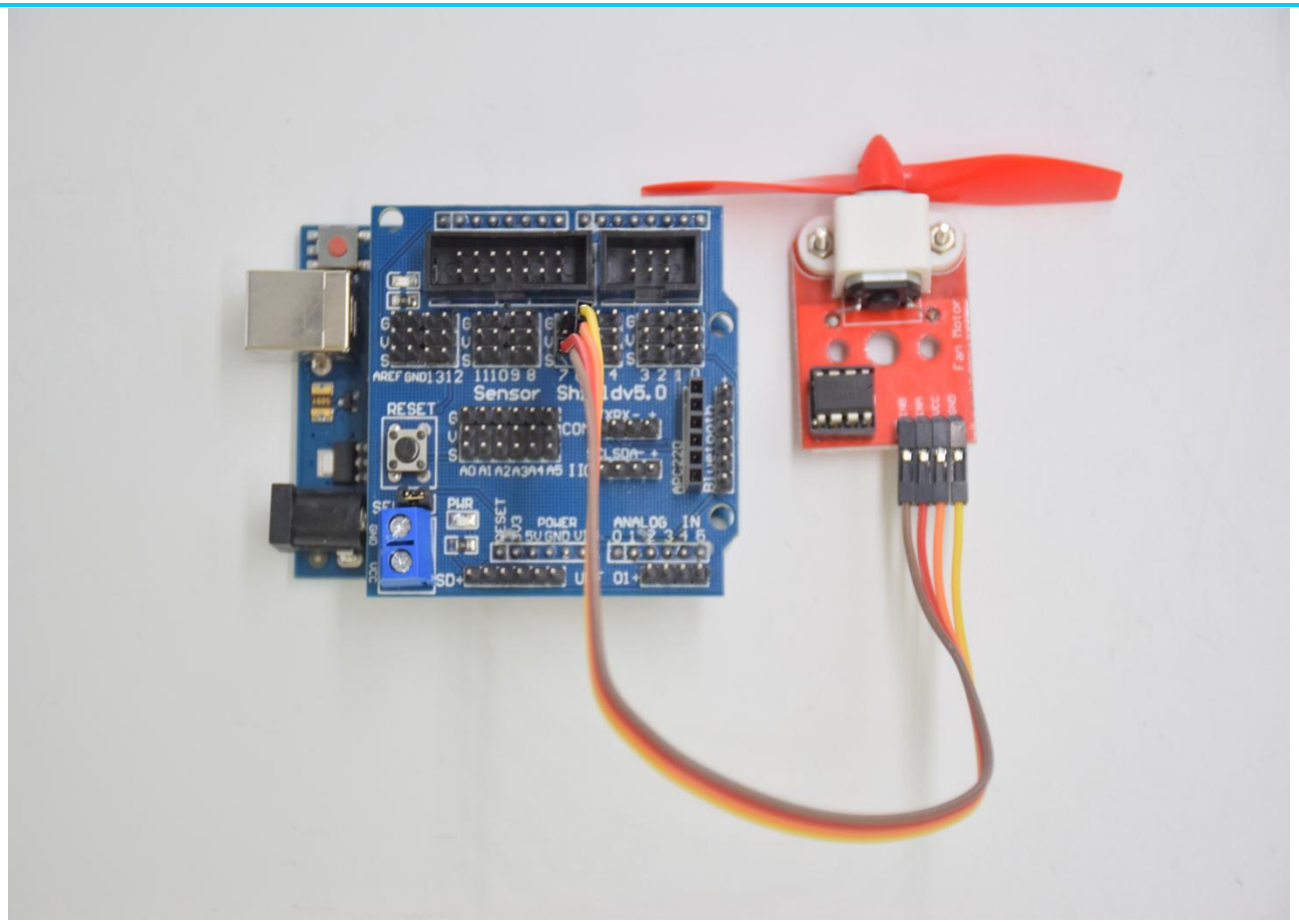
INB -- D6

INA -- D7

VCC -- 5V

GND -- GND

LROBRUYA



Test Result：

Upload test code, wire according to connection diagram, the DIP switch is dialed to right side and power on. The fan rotates counterclockwise for 3000ms, stops for 1000ms, then rotates clockwise for 3000ms.

Project 9： Steam Sensor

Description

This is a commonly used water level sensor. The principle is to detect the amount of water through the exposed printed parallel lines on the circuit board. The more water, the more wires are connected. As the conductive contact area increases, the output voltage will gradually increase. The water level sensor can be used as a rain detector switch. When the humidity on the sensor surface increases sharply, the output voltage will increase.

The sensor is compatible with various microcontroller control boards, such as Arduino series microcontrollers. When using, we provide a guide for operating the water level sensor and Arduino control board. Connect the signal end of the sensor to the analog port of the microcontroller, sense the change of the analog value, and display the corresponding analog value on the serial monitor.

Note: The connection part is not waterproof, please do not immerse it in water.

Specifications:

- Working voltage: DC 3.3-5V
- Working current: <20mA
- Operating temperature range: -10 ℃ ～ ＋ 70 ℃;
- Control signal: analog signal output
- Interface: 2.54mm 3pin pin interface

- Size: 35 * 20 * 8mm
- Weight: 2.2g
- S: signal output
- +: Power supply (VCC)
- -: Ground (GND)

Materials：

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Water sensor
			

Connection：

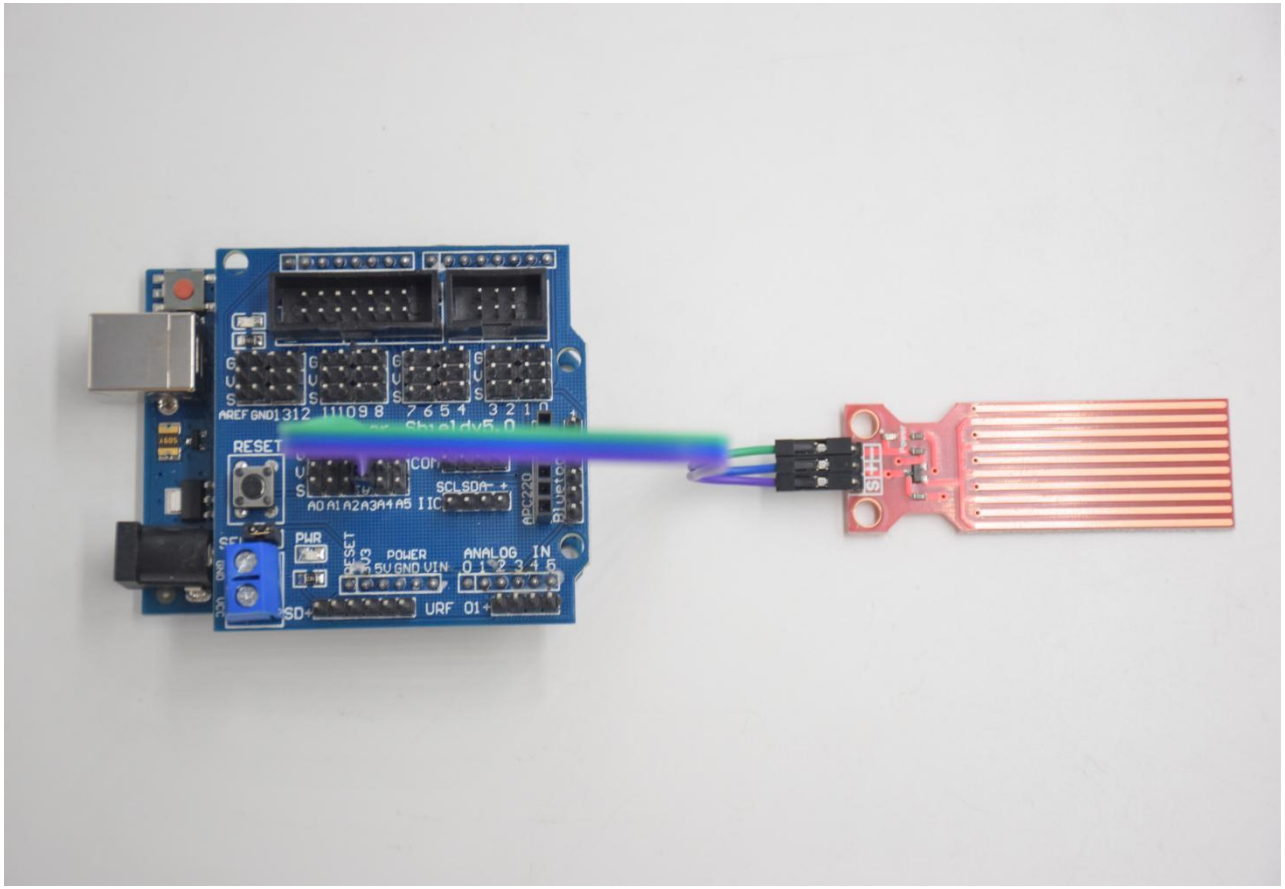
Water level module:

- -- GND

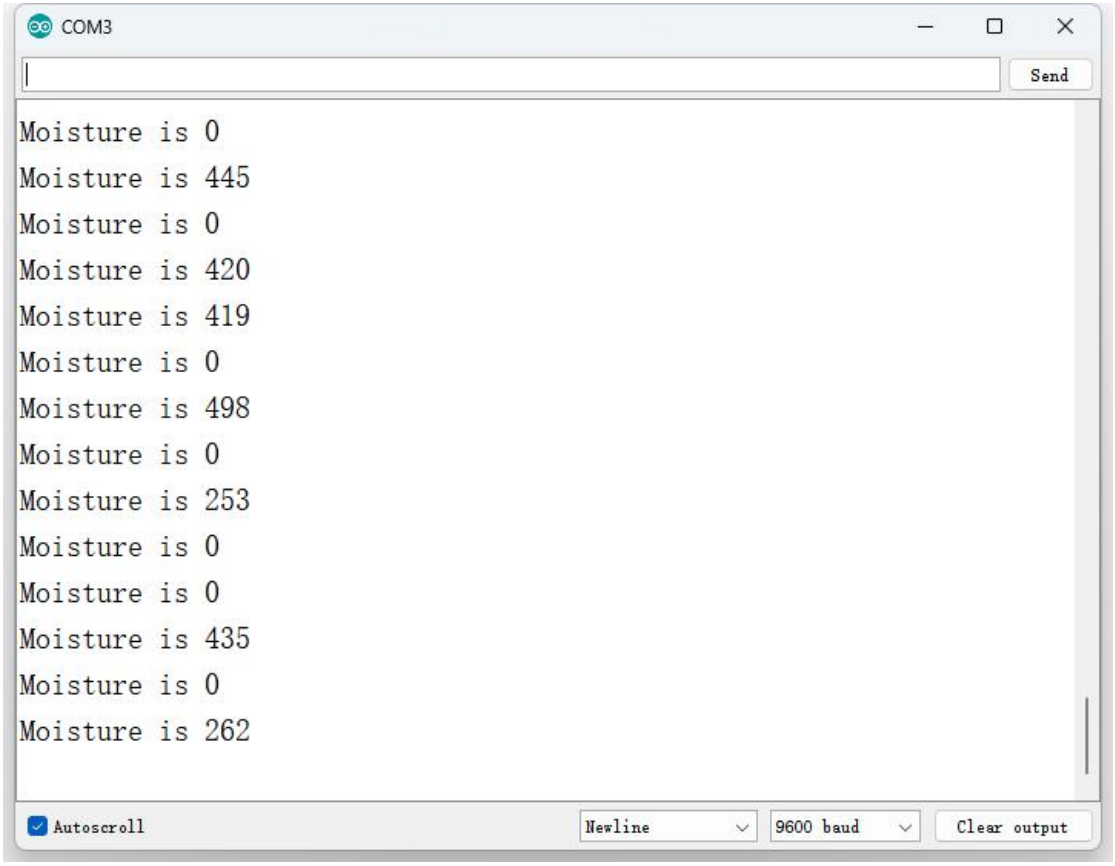
+ -- 5V

S -- A3

LROBRUYA



Test Result：


COM3
Send
Moisture is 0
Moisture is 445
Moisture is 0
Moisture is 420
Moisture is 419
Moisture is 0
Moisture is 498
Moisture is 0
Moisture is 253
Moisture is 0
Moisture is 0
Moisture is 435
Moisture is 0
Moisture is 262
Autoscroll
Newline
9600 baud
Clear output

Project 10： PIR Motion Sensor

Description

The human body infrared motion sensor can detect infrared signals from moving people or animals, and output switching signals. It can be applied to various occasions to detect human movement. The conventional pyroelectric infrared sensor has a larger volume, a complicated circuit and a lower reliability. Now, we have introduced a human body infrared motion sensor specially designed for Arduino. The sensor integrates an integrated digital pyroelectric infrared sensor and connection pins. It has higher reliability, lower power consumption and simpler peripheral circuits.

Specifications:

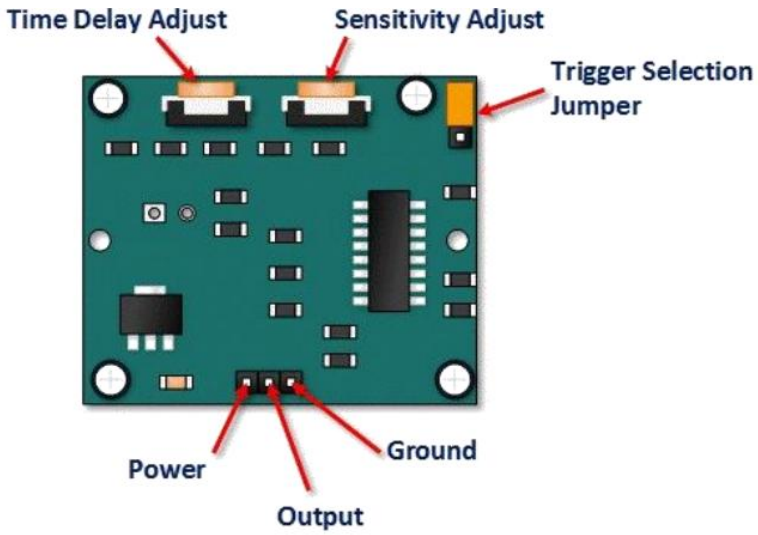
- Input voltage: DC 5V
- Working current: 15uA

- Working temperature: -20 ~ 85 degrees Celsius
- Output voltage: high 3 V, low 0 V
- Detection angle: about 140 °
- Detection distance: 3-4 meters
- Pin limit current: 100mA

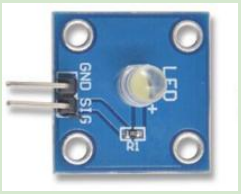
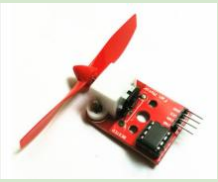
Special note：

- 1. The maximum distance is 4-5 meters during testing.
- 2. When testing, the sensor needs to be covered with white lens, otherwise it will affect the distance.
- 3.Uncover the white cap to see the port label
- 4. The distance is best at 25℃, and the detection distance is shortened when it exceeds 30℃.

- 5. Done powering up and uploading the code, you need to wait 5-10 seconds then start testing, otherwise it is not sensitive.
- 6.Use a screwdriver to adjust the human body induction potentiometer according to the figure below



Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	White LED Module	PIR Motion Sensor	Fan Module
					

Connection：

Human body induction：

GND -- GND

OUT -- D2

VCC -- 5V

Motor fan module:

INB -- D6

INA -- D7

VCC -- 5V

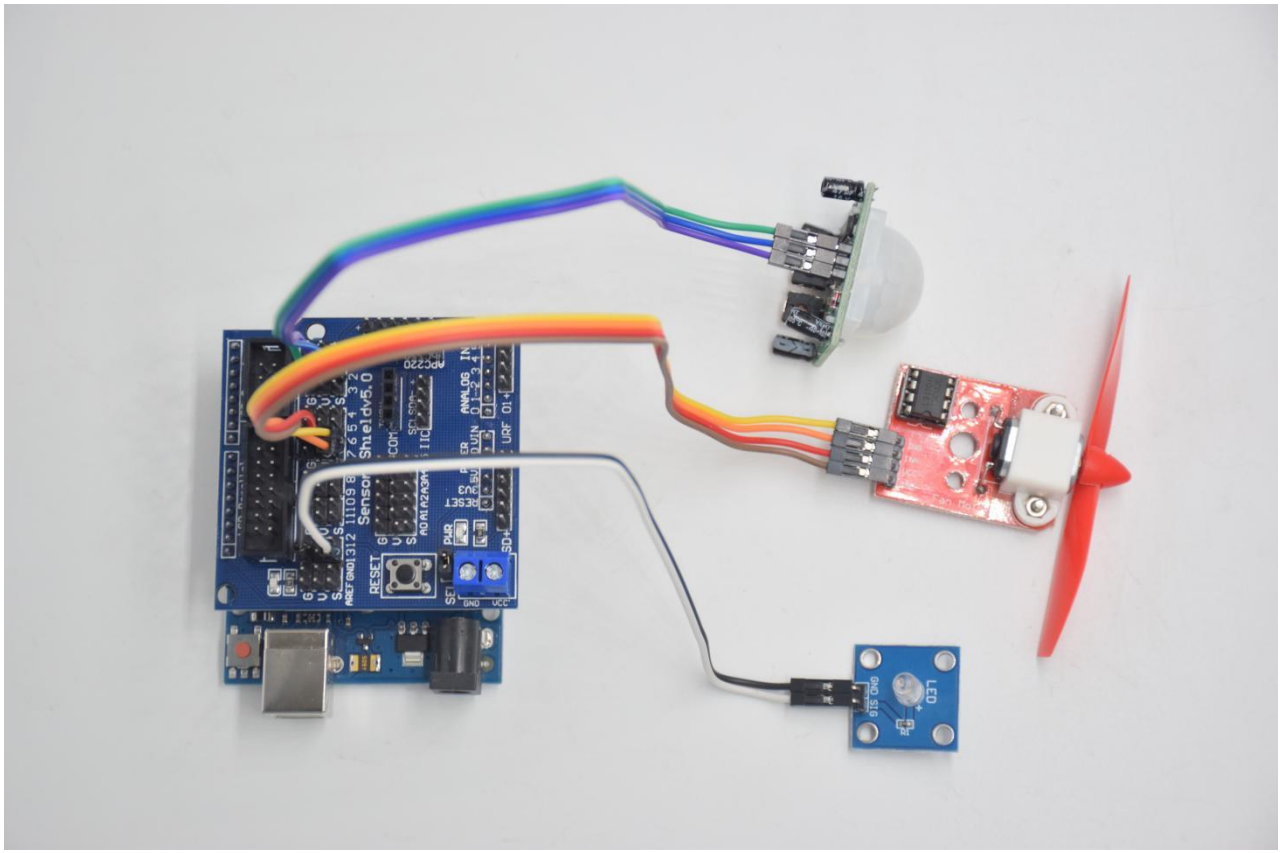
GND -- GND

LED:

GND -- GND

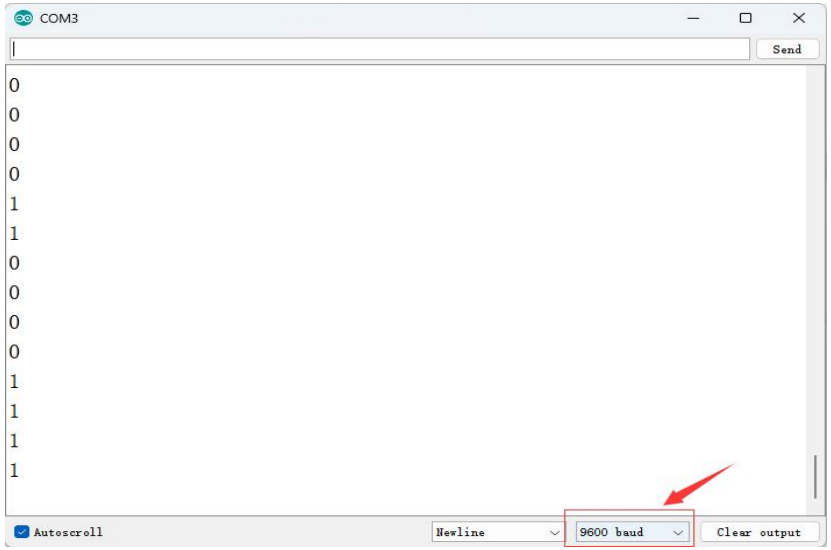
SIG -- D13

LROBRUYA



Test Result:

Upload the test code, open the serial monitor, and set the baud rate to 9600. If the human infrared sensor detects people around, the serial monitor displays "1", D13 and the white LED indicator light up at the same time, and the fan rotates. If there is no one around, the serial monitor displays "0" and the D13 indicator and white LED are off. The fan stopped spinning.



Project 11：Analog（MQ-2）Sensor

Description

This gas sensor is used for household gas leak alarms, industrial combustible gas alarms and portable gas detection instruments. And it is suitable for the detection of liquefied gas, benzene, alkane, alcohol, hydrogen, etc., and widely used in various fire alarm systems. The MQ-2 smoke sensor can be accurately a multi-gas detector, and has the advantages of high sensitivity, fast response, good stability, long life, and simple drive circuit. It can detect the concentration of flammable gas and smoke in the range of 300~10000ppm.Meanwhile, it has high sensitivity to natural gas, liquefied petroleum gas and other smoke, especially to alkanes smoke. It must be heated for a period of time before using the smoke sensor, otherwise the output resistance and voltage are not accurate. However, the heating voltage should not be too high, otherwise it will cause my internal signal line to blow.

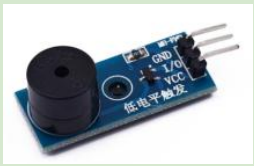
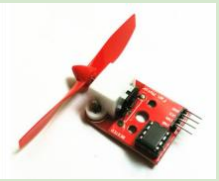
It is belongs to the tin dioxide semiconductor gas-sensitive material, and belongs to the surface ion type N-type semiconductor. At a certain temperature, tin dioxide adsorbs oxygen in the air and forms negative ion adsorption of oxygen, reducing the electron density in the semiconductor, thereby increasing its resistance value. When in contact with flammable gas in the air and smog, if the potential barrier at the grain boundary is adjusted by the smog, it will cause the surface conductivity to change. With this, information about the presence of smoke or flammable gas can be obtained. The greater the concentration of smoke or flammable gas in the air, the greater the conductivity, and the lower the output resistance, the larger the analog signal output. The sensor comes with a positioning hole, which is convenient for you to fix the sensor to other devices. In addition, the sensitivity can be adjusted by rotating the potentiometer.

Specifications:

- Working voltage: 3.3-5V (DC)
- Interface: 4 pins (VCC, GND, D0, A0)

- Output signal: digital signal and analog signal
- Weight: 7.5g

Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	Passive Buzzer Sensor	MQ-2 Gas Sensor	Fan Module
					

Connection：

Passive buzzer：

VCC -- 5V

I/O -- D3

GND -- GND

Smoke sensor:

AO -- A0

GND -- GND

VCC -- 5V

Motor fan module:

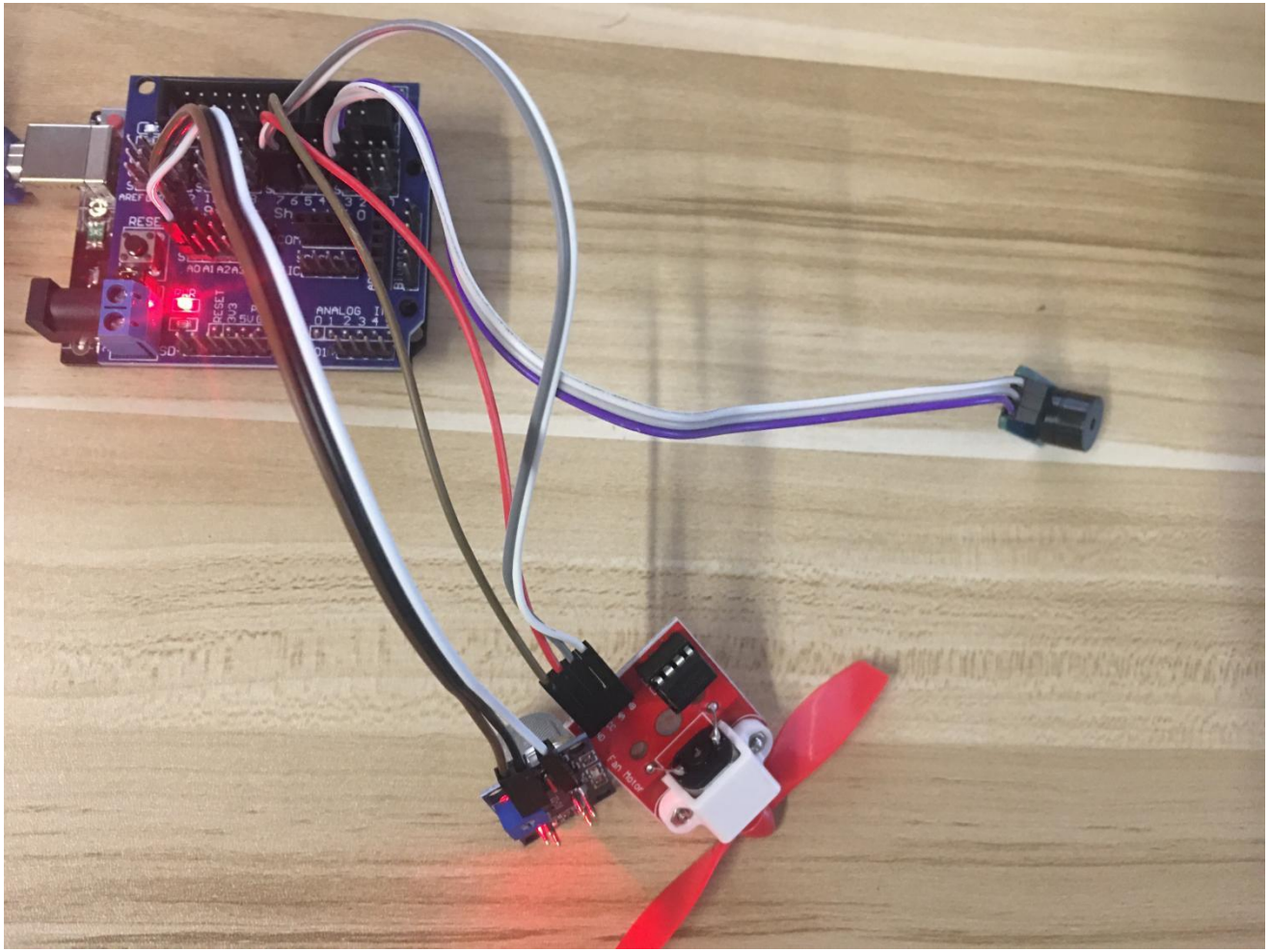
INB -- D6

INA -- D7

VCC -- 5V

GND -- GND

LROBRUYA



Test Result：

Upload test code, wire up components according to connection diagram and power on. When the detected value of flammable gas is greater than 80, the passive buzzer will emit sound and reverse rotation of the fan motor, however, when there is no flammable gas, the passive buzzer won't emit a sound and the motor stops rotating.

Project 12： 1602 LCD Display

Description

With I2C communication module, this is a display module that can show 2 lines with 16 characters per line. It shows blue background and white word and connects to I2C interface of MCU, which highly save the MCU resources. On the back of LCD display, there is a blue potentiometer for adjusting the backlight. The communication address defaults to 0x27. The original 1602 LCD can start and run with 7 IO ports, but ours is built with Arduino IIC/I2C interface, saving 5 IO ports. Alternatively, the module comes with 4 positioning holes with a diameter of 3mm, which is convenient for you to fix on other devices.

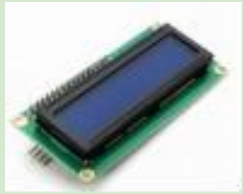
Notice that when the screen gets brighter or darker, the characters will become more visible or less visible.

Specifications:

- I2C address: 0x27
- Backlight (blue, white)
- Power supply voltage: 5V

- Adjustable contrast
- GND: A pin that connects to ground
- VCC: A pin that connects to a +5V power supply
- SDA: A pin that connects to analog port A4 for IIC communication
- SCL: A pin that connects to analog port A5 for IIC communication

Materials:

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	LCD1602 display module
			

Connection:

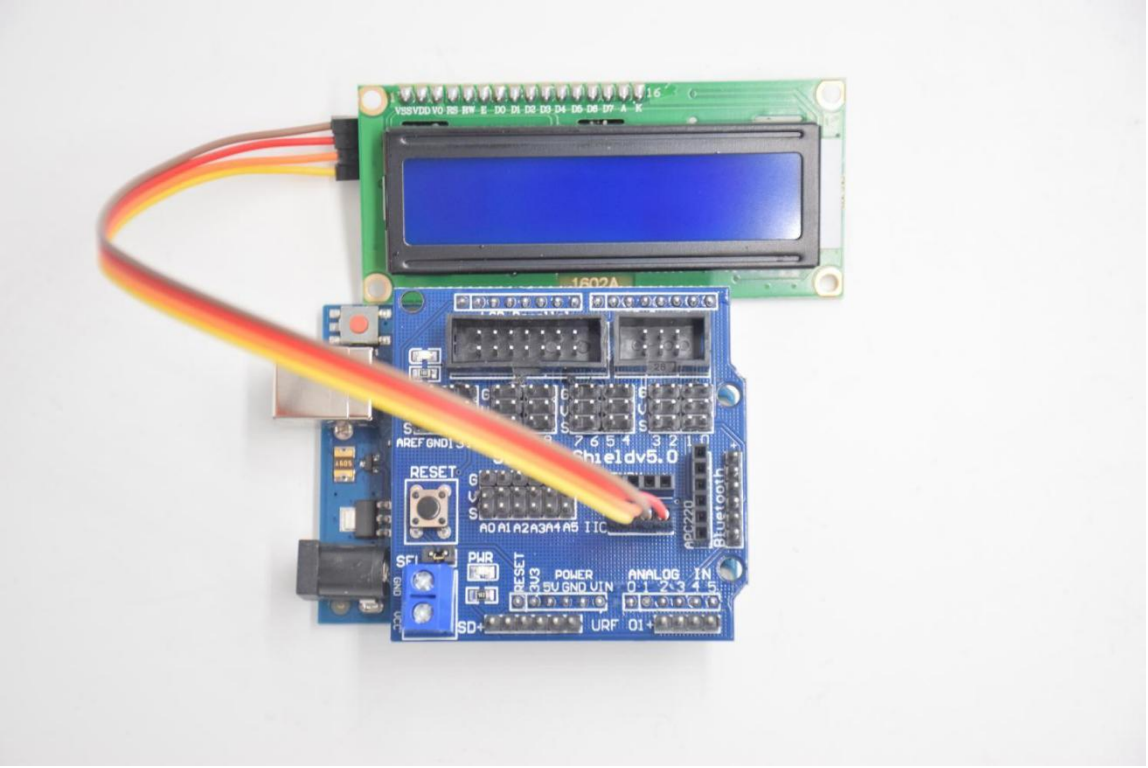
LCD1602 display:

SCL -- SCL

SDA -- SDA

VCC -- +

GND -- -



Test Result:

After connection and uploading sample code, the first line on LCD prints "Hello, world!", second line prints "LONTEN", with a potentiometer to adjust LCD backlight.

Project 13： Soil Humidity Sensor

Description

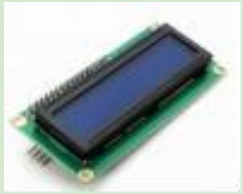
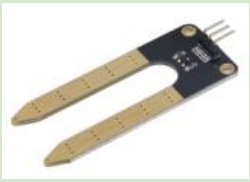
This is a simple soil humidity sensor aims to detect the soil humidity. If the soil is in lack of water, the analog value output by the sensor will decrease; otherwise, it will increase. If you use this sensor to make an automatic watering device, it can detect whether your botany is thirsty to prevent it from withering when you go out. Using the sensor with Arduino controller makes your plant more comfortable and your garden smarter. The soil humidity sensor module is not as complicated as you might think, and if you need to detect the soil in your project, it will be your best choice. The sensor is set with two probes inserted into the soil, then with the current go through the soil, the sensor will get resistance value by reading the current changes between the two probes and convert such resistance value into moisture content. The higher moisture (less resistance), the higher conductivity the soil has. Insert it into the soil and then use the AD converter to read it.

With the help of this sensor, the plant can remind of you: I need water.

Specification

- Power Supply Voltage: 3.3V or 5V
- Working Current: ≤ 20mA
- Sensor type: Analog output
- Interface definition: OUT- signal, GND- GND, VCC- VCC
- Packaging : Electrostatic bag sealing

Materials：

UNO R3 control Board	Sensor Shield v5.0	Female to Female Dupont wire	LCD1602 display module	Soil Humidity Sensor
				

Connection:

I2C 1602 display:

SCL -- SCL

SDA -- SDA

\- -- GND

\+ -- 5V

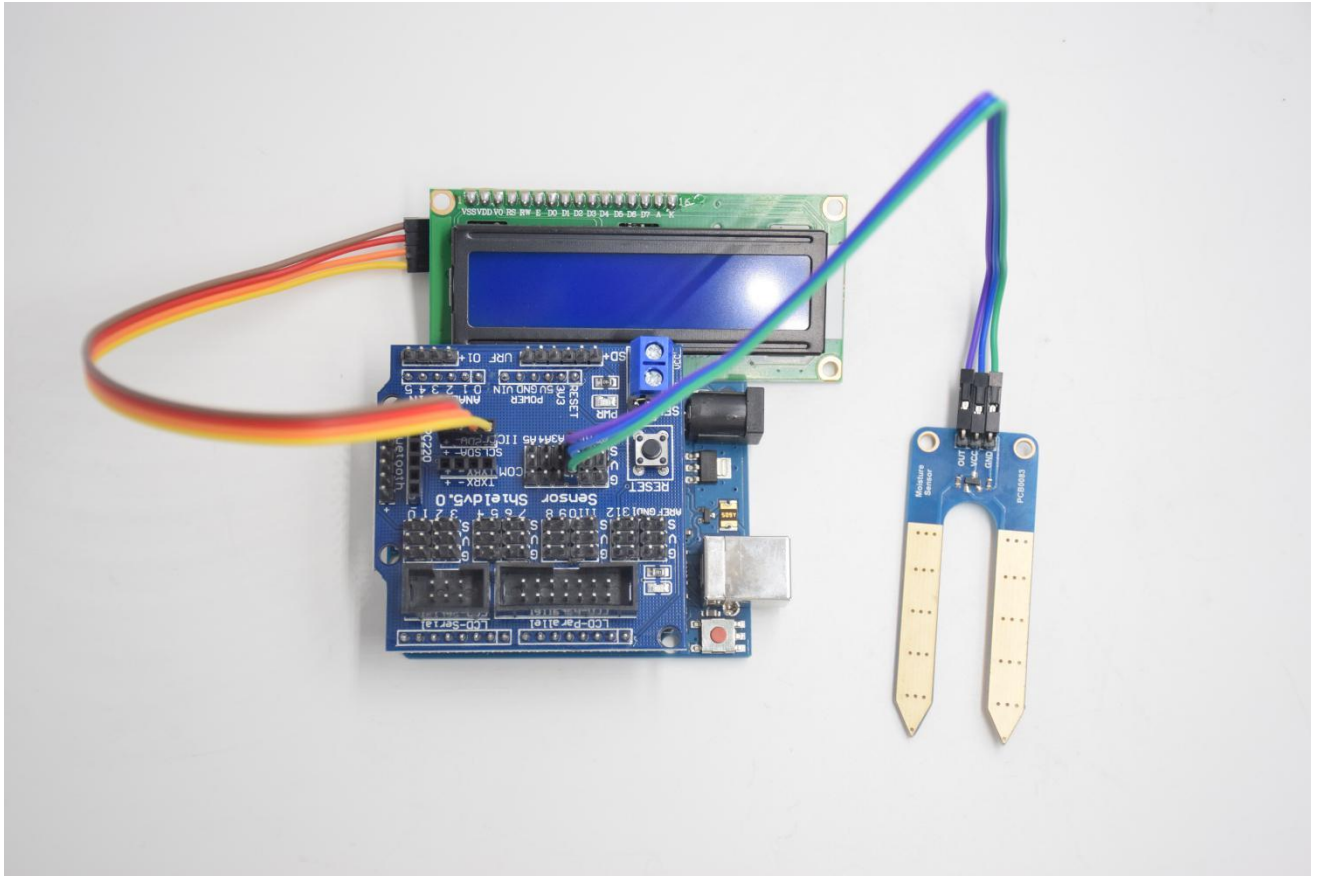
Soil module:

OUT -- A2

VCC -- 5V

GND -- GND

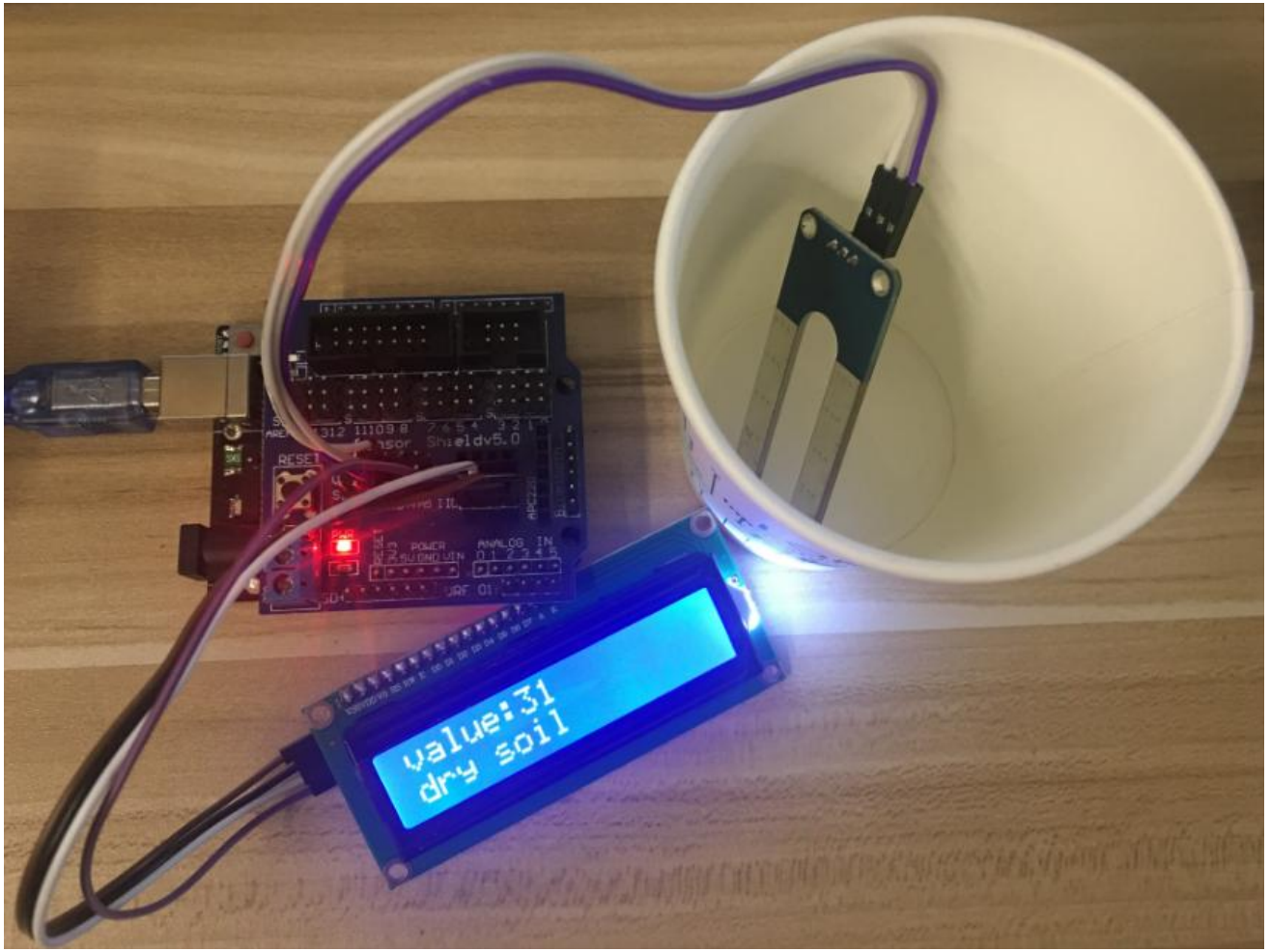
LROBRUYA



Test Result:

Connect according to wiring diagram, and burn the program and power on. Open the serial monitor and insert the soil humidity sensor into the soil. The greater the humidity is, the bigger the number, in the range of 0-1023. The soil sensor is inserted into the soil and water with different humidity, and the 1602LCD displays the corresponding value.

LROBRUYA



Project 14： Bluetooth Test

Description

In 20th century, technology has changed our life. People can work at home with wireless device like mouse, earphone, printer and speaker, which highly enhances our life standard. Bluetooth can make work at home easily, as well as the entertainment. Users can control wirelessly the audio file from PC or Apple iPod within 30 inches. Bluetooth technology can also be used in adapters, allowing people to share their daily life with friends from internet and social media.

Bluetooth Remote Control

Bluetooth technology is a wireless standard technology that enables short-distance data exchange between fixed devices, mobile devices, and building personal area networks (using UHF radio waves in the ISM band of 2.4 to 2.485 GHz). This kit is equipped with the HM-10 Bluetooth module, which is a master-slave machine. When use as the Host, it can send

commands to the slave actively; when use as the Slave, it can only receive commands from the host. The HM-10 Bluetooth module supports the Bluetooth 4.0 protocol, which not only supports Android mobile, but also supports iOS system. In the experiment, we default use the HM-10 Bluetooth module as a Slave and the cellphone as a Host. We install the Bluetooth APP on the mobile phone, connecting the Bluetooth module; finally use the Bluetooth APP to control the parts of smart home kit. We also provide you with 2 types of mobile APP, for Android and iOS system.

Parameters of HM-10 Bluetooth Module:

Pins	Description
BRK	As input pin, short press control, or input single pulse of 100ms low level to achieve the following functions: 1. When module is in sleep state: Module is activated to normal state, if open AT+NOTI, serial port will send OK+WAKE. 2. When in connected state: Module will actively request to disconnect When in standby mode: Module will be in initial state
RXD	Serial data inputs
TXD	Serial data outputs
GND	ground lead
VCC	Positive pole of power, input 5V
STATE	As output pin, show the working state of module Flash slowly in standby state——repeat 500ms pulse; Always light in connected state——high level You could set to no flashing in standby state, always light in connected state

- Bluetooth protocol: Bluetooth Specification V4.0 BLE

- No byte limit in serial port Transceiving
- In open environment, realize 100m ultra-distance communication with iphone4s
- USB protocol: USB V2.0
- Working frequency: 2.4GHz ISM band
- Modulation method: GFSK(Gaussian Frequency Shift Keying)
- Transmission power: -23dbm, -6dbm, 0dbm, 6dbm, can be modified by AT command.
- Sensitivity: ≤-84dBm at 0.1% BER
- Transmission rate: Asynchronous: 6K bytes ; Synchronous: 6k Bytes
- Security feature: Authentication and encryption
- Supporting service: Central & Peripheral UUID FFE0, FFE1
- Power consumption: Auto sleep mode, stand by current 400uA~800uA, 8.5mA during transmission.

- Power supply: 5V DC
- Working temperature: –5 to +65 Centigrade

Using Bluetooth APP

In the previous project, we've introduced the basic parameter principle of HM-10 Bluetooth module. In this project, let's show you how to use the HM-10 Bluetooth module. In order to efficiently control this kit by HM-10 Bluetooth module, we specially designed an APP, as shown below.

There are 12 control buttons and 4 sliders on app. When we connect the HM-10 Bluetooth module and app, only press control button of APP, and the Bluetooth of cellphone sends a control character. The Bluetooth module will receive a corresponding control character. When programming, we set the corresponding function of each sensor or module according to the corresponding key control character. Next, let's test 16 buttons on app.

APP for Android mobile：

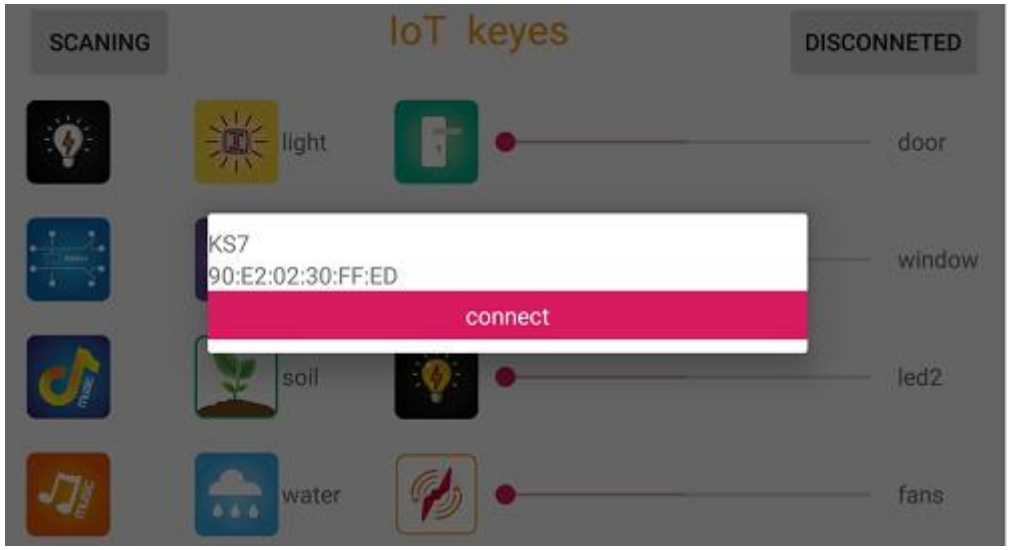
Note: You need to enable the location information before connecting to HM-10 Bluetooth module via cellphone, otherwise, Bluetooth may not be connected.

App see toolkit:APP for Android/

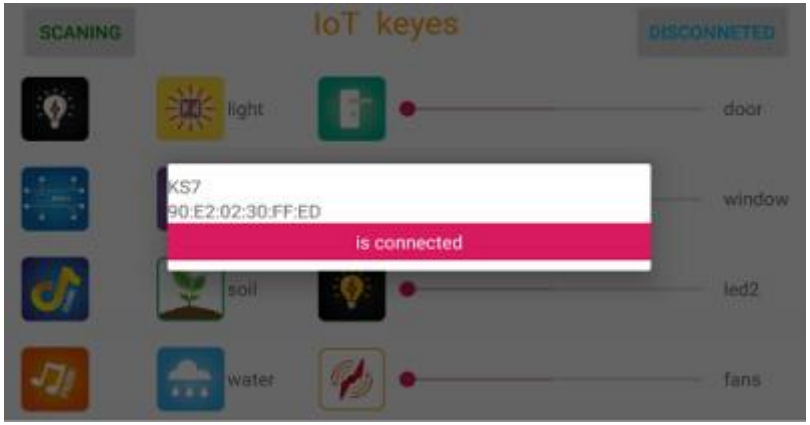
After installing and open the app 

， the interface pops up as below:

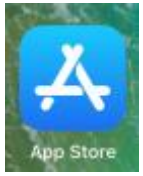
Upload code and power on, Led of Bluetooth module blinks. Start Bluetooth and open App to click “CONNECT” to connect.



Click to “Connect”, Bluetooth is connected successfully. As shown below, the LED of Bluetooth module is normally on.

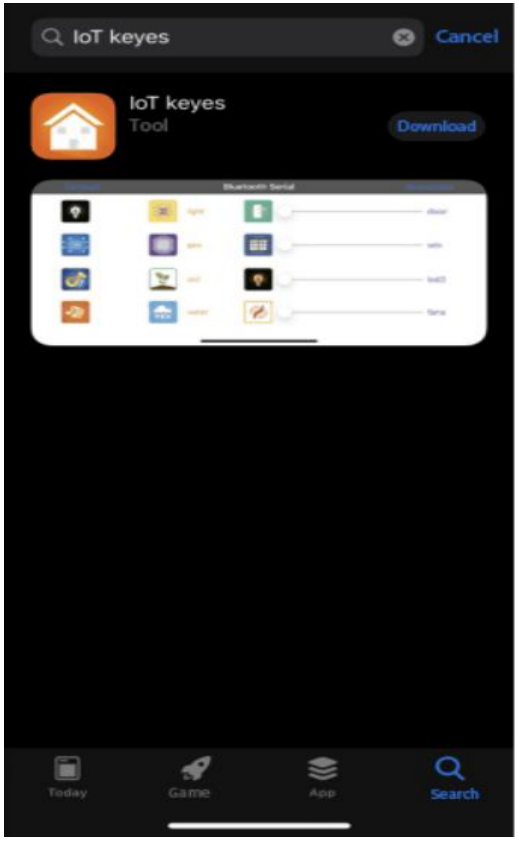


For IOS system:

(1) Open App Store 

(2) Search “IoT keyes”on APP Store， then click “downlaod”.

LROBRUYA


IoT keyes
Cancel
IoT keyes
Tool
Download
Today
Game
App
Search

(3) After installing successfully and open 

, the interface is shown below:

(4)Upload the test code successfully, insert the Bluetooth module and power on. LED of Bluetooth module is flashing. Start Bluetooth on cellphone, then click “connect” on the left to search Bluetooth and pair. After paring successfully, the LED of Bluetooth module is on. Note: Remove the Bluetooth module please, when uploading the test code. Otherwise, the program

will fail to upload. Connect the Bluetooth and Bluetooth module to pair after uploading the test code.

Connection

Bluetooth module:

STATE -- No answer

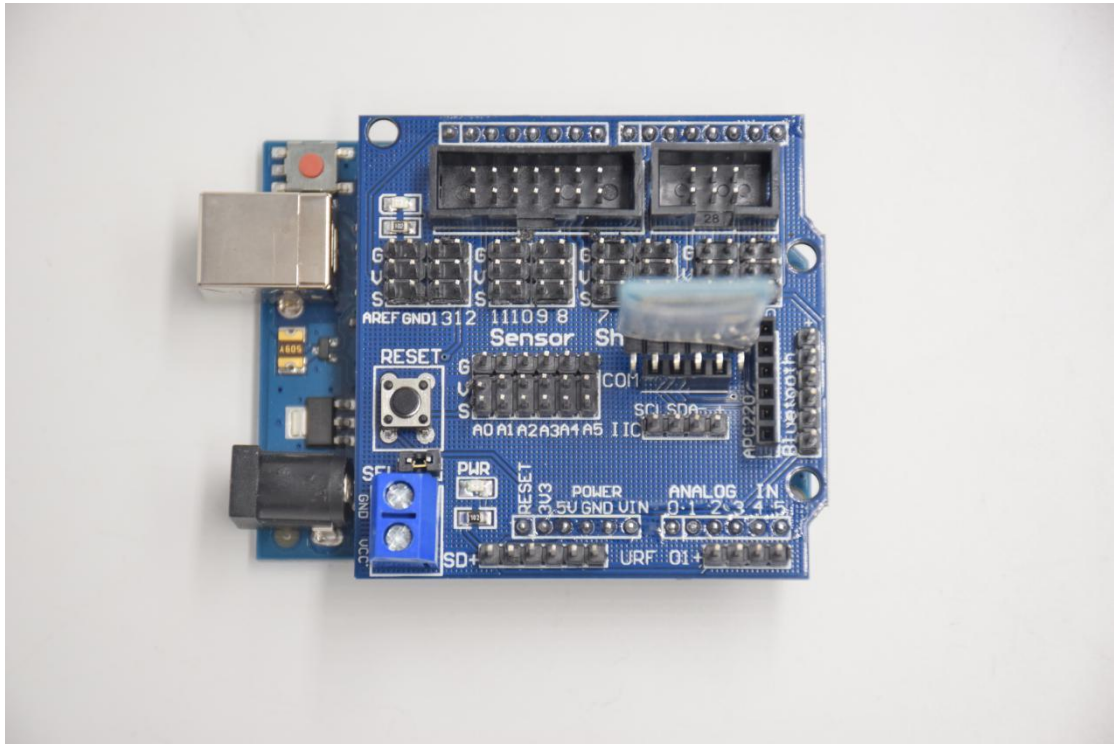
RXD -- TX

TXD -- RX

GND -- -

VCC -- +

EN -- No answer



The function of corresponding character and button is shown below:

LROBRUYA

No.	Button	Control Character	Function	No	Button	Control Character	Function
1	SCANNING	Pair and connect to HM-10 Bluetooth module		2	DISCONNECT	Disconnect Bluetooth	
3		Click to send "a", click again to send "b"	Click to turn on white LED, click again to turn off LED	4		Click to send "c", click again to send "d"	Clcik to turn on relay module; click again to turn off relay module
5		Hold and press to send "e" release to send "g"	Click to play music	6		Hold and press to send "f" release to send "g"	Click to play music (alternative song)
7		Click to send"h", click again to send "s"	Click to turn on photocell sensor, light shows the data; click again to turn off photocell sensor	8		Click to send "i" click again to send "S"	Click to turn on gas sensor, gas displays the detected data; click again to turn off gas sensor
9		Click to send"j" click again to send"S"	Click to turn on soil humidity sensor, soil shows data, click again to turn off soil humidity sensor	10		Click to send "k" click again to send "S"	Click to turn on steam sensor, water displays the detected data; click again to turn off steam sensor
11		Click to send "l" ; click again to send"m"	Click to open the door; click again to close the door	12	door	Drag slider to send "t 50 #", 't' represents initial character; 50 is the angle of servo 1 ; '#'implies termination character	Slider controls the angle of servo 1 to rule the door, door displays the angle value of servo 1
13		Click to send "n"; click again to send"o"	Click to open the window; click again to close the window	14	window	Drag slider to send "u 34 #", 'u'represents initial character; 34 is the angle of servo 2; '#' stands for termination character	Slider controls the angle of servo 2 to rule the window, win shows the angle value of servo 2
15		Click to send "p" ; click again to send"q"	Click to turn on LED; click again to turn off LED	16	led2	Drag slider to send "v 100 #", 'v'represents initial character; 100 is the PWM value of led2; '#' stands for termination character	Slider controls LED brightness, led2 displays brightness value
17		Click to send "r" ; click again to send"s"	Click to turn on fan; click again to turn off fan	18	fans	Drag slider to send "w 153 #", 'w'represents initial character; 153 is the PWM value of fan ; '#'stands for termination character	Slider controls rotation speed, fans indicates the rotation speed value

Project 15： Multi-purpose Smart Home Kit

Description

In the previous projects, we introduce how to use sensors, modules and HM-10 Bluetooth module. For this project, we will perform all functions We will achieve the effect as follows:

- Photocell sensor, PIR motion sensor and LED. When at night, someone passes by, LED is on; nobody is around, the LED is off.

- There are 1602LCD display, 2 buttons, 1 servo on the board. Press button1 to enter the password(you can set password in the Project_15_Multi-purpose_Smart_Home_Kit), the 1602LCD will show “*”, then press button2 to “ensure”. If the password is correct, the 1602LCD will show “open”, the door will be open. However, if the password is wrong, the “error” pops up , after 2s, “error” will turn into “again” , you can enter password again.

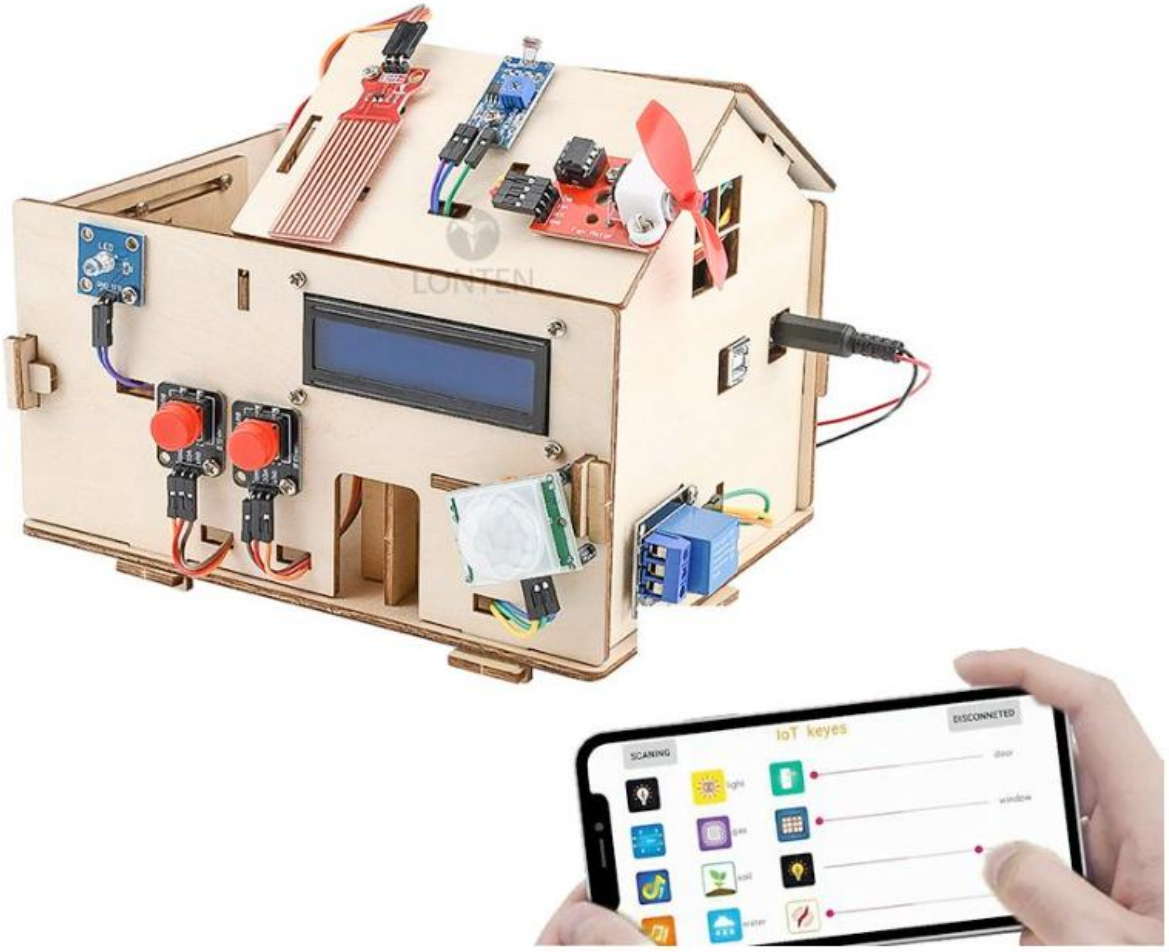
The door will be closed when PIR motion sensor doesn't detect people around. What's more, press and hold button2, buzzer will sound, LCD displays "wait". (If the password is right, the servo will rotate to 180°, otherwise, the servo don't rotate)

Note: The correct password is "*++*+*" which means that **short press button1, long press button1, long press button1, short press button1, long press button1, short press button1.**

"* "means **long press button1**, "+"means **short press button1**

- Insert soil humidity into plant pot, when the soil is too dry, buzzer will alarm and you will get the notification on app.
- When the gas sensor detects the gas with high concentration, the buzzer emits a "tick,tick" alarm sound.
- When steam sensor detects rains, the servo 2 will be activated, the window will be closed automatically, otherwise, the window will be open.

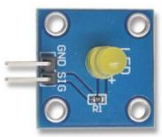
LROBRUYA

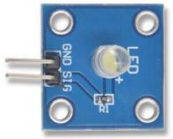


LROBRUYA

Materials：











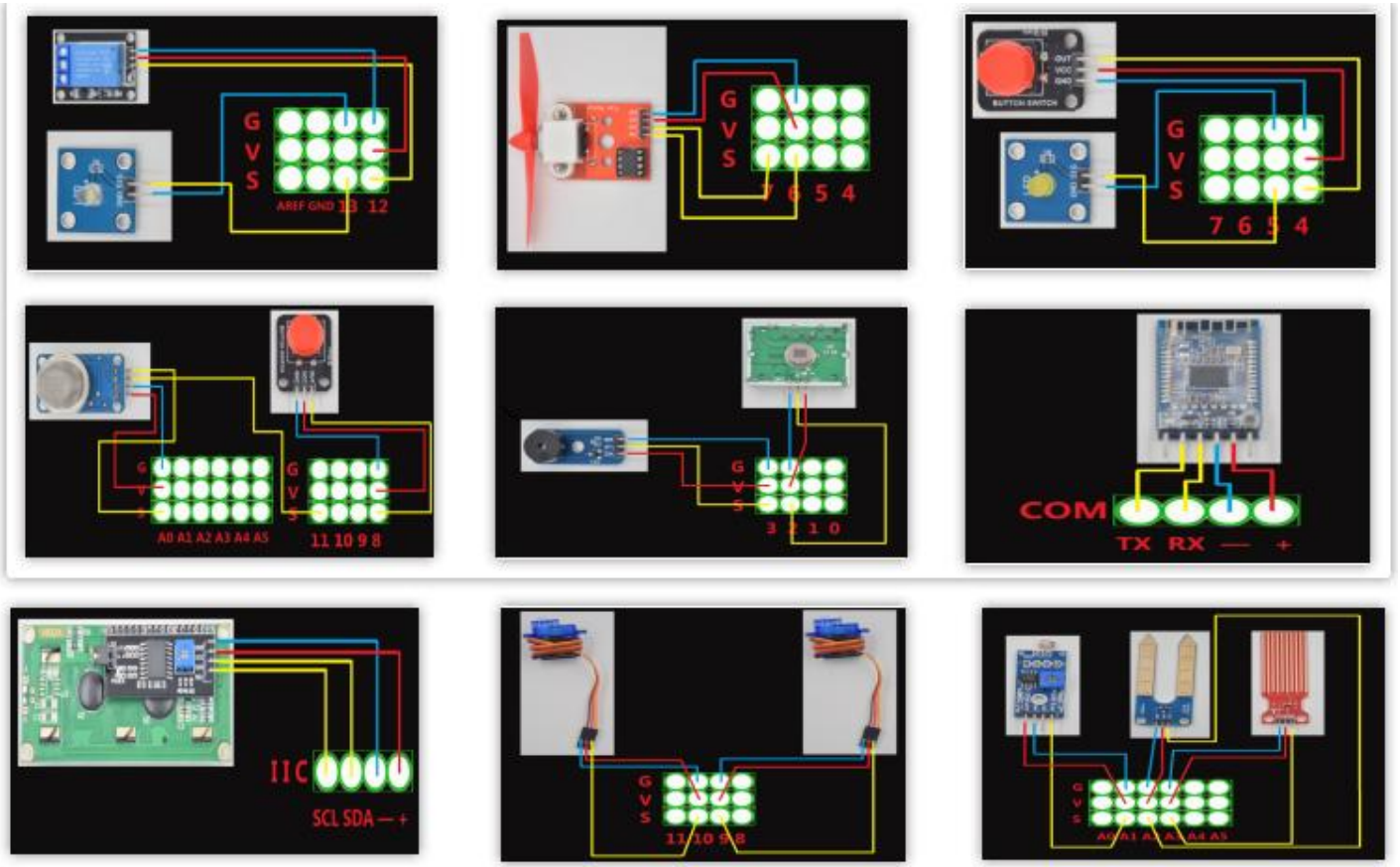







LROBRUYA

Connection：



Name	The corresponding interfaces of sensors and sensor shield	
PIR Motion Sensor	G/V/S	G/V/2
Passive buzzer	G/V/S	G/V/3
Button module 1	G/V/S	G/V/4
Yellow LED	G/V/S	G/V/5
Fan module	GND/VCC/INA/INB	G/V/7/6
Button module 2	G/V/S	G/V/8
Servo 1 controlling the door	Brown/Red/Orange wire	G/V/9
Servo 2 controlling the windows	Brown/Red/Orange wire	G/V/10
MQ-2 Gas Sensor	GND/VCC/D0/A0	G/V/11/A0
Relay Module	G/V/S	G/V/12
White LED	G/V/S	G/V/13
LCD1602 Display	GND/VCC/SDA/SCL	GND/5V/SDA/SCL
Photocell Sensor	G/V/S	G/V/A1
Soil humidity sensor	G/V/S	G/V/A2
Steam sensor	G/V/S	G/V/A3

Test Result:

Upload the code and see the result!

Note: Remove the Bluetooth module please, when uploading the test code. Otherwise, the program will fail to upload.

Connect the Bluetooth and Bluetooth module to pair after uploading the test code.

Upload the test code, stack expansion board on PLUS Control Board, and power on. After pairing and connecting Bluetooth successfully, we can control the smart home through app.