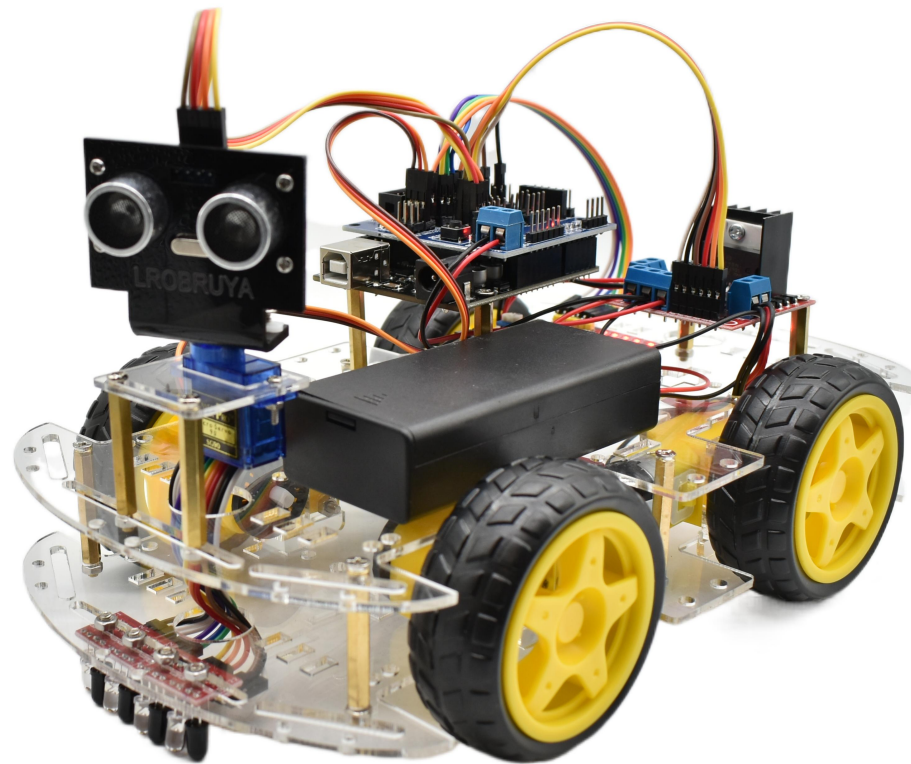


LROBRUYA

**DIY 4WD Smart Robot Car Kit**

# LROBRUYA

---





---

## Content

Preface .....	5
How to Install Arduino IDE .....	7
How to Install Arduino Driver .....	22
How to Add Arduino Libraries .....	39
Blink Test .....	47
Lesson 1 Servo .....	62
Lesson 2 Ultrasonic Sensor Module .....	66
Lesson 3 L298N Motor Driver .....	73
Lesson 4 Ultrasound Obstacle Avoidance Car .....	77
Lesson 5 Follow Car .....	80

# LROBROYA

---

Lesson 6 Line-Tracking sensor .....	83
Lesson 7 Line-Tracking Car .....	88



---

## 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.



---

By the way, We also look forward to hearing from you and any of your critical comment or suggestions. Pls email us by [lonten3@qq.com](mailto:lonten3@qq.com) or [info@lontentech.com](mailto:info@lontentech.com), if you have any questions or suggestions. As a continuous and fast growing company. We keep striving our best to offer you excellent products and quality service.

## **Our Store**

store: <https://www.lontentech.com/>

Brand: LONTEN

## **Product Catalog**

<https://www.lontentech.com/collections/steam-robot>

## **Tutorial**

This tutorial include codes,libraries and lessons. It is designed for beginners. It will teach every users how to assembly the robot kit and use UNO controller board, sensors,servo.



---

## How to Install Arduino IDE

### Introduction

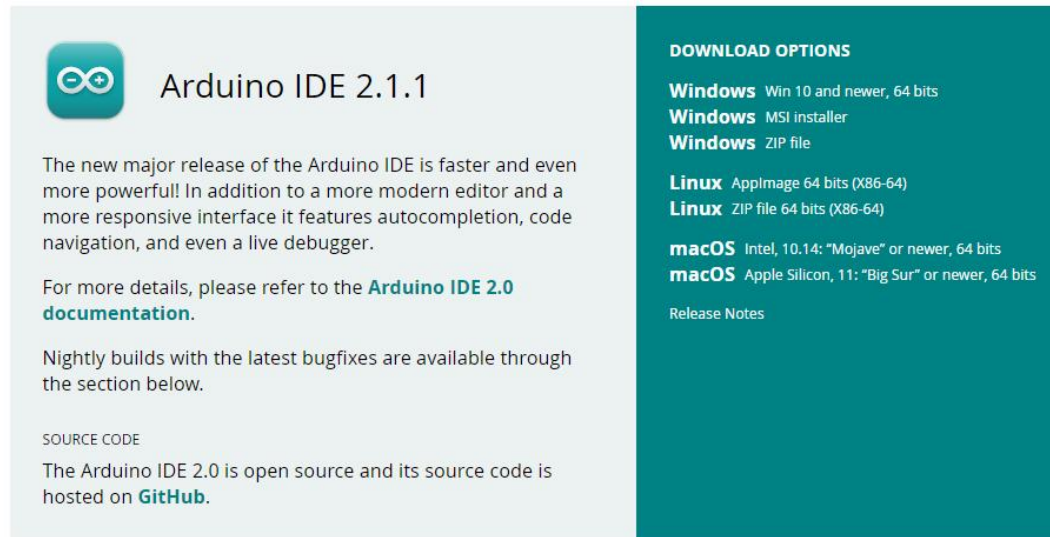
The Arduino Integrated Development Environment (IDE) is the software side of the Arduino platform.

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>.

# LROBROYA



 **Arduino IDE 2.1.1**

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

**SOURCE CODE**

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

**DOWNLOAD OPTIONS**

**Windows** Win 10 and newer, 64 bits  
**Windows** MSI installer  
**Windows** ZIP file

**Linux** AppImage 64 bits (X86-64)  
**Linux** ZIP file 64 bits (X86-64)

**macOS** Intel, 10.14: "Mojave" or newer, 64 bits  
**macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

**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.



# LROB RUYA

---

## DOWNLOAD OPTIONS

**Windows** Win 10 and newer, 64 bits

**Windows** MSI installer

**Windows** ZIP file

**Linux** Appliance 64 bits (X86-64)

**Linux** ZIP file 64 bits (X86-64)

**macOS** Intel, 10.14: "Mojave" or newer, 64 bits

**macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

Click Windows Win 10 and newer, 64 bits.

# LROBBUYA





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.


# LROBROYA

---

 arduino-ide\_2.1.1\_Linux\_64bit

 arduino-ide\_2.1.1\_macOS\_64bit

 arduino-ide\_2.1.1\_Windows\_64bit

 arduino-ide\_2.1.1\_Windows\_64bit

## Installing Arduino (Windows)

**Install Arduino with the exe. Installation package.**

 arduino-ide\_2.1.1\_Windows\_64bit

# LROBROYA

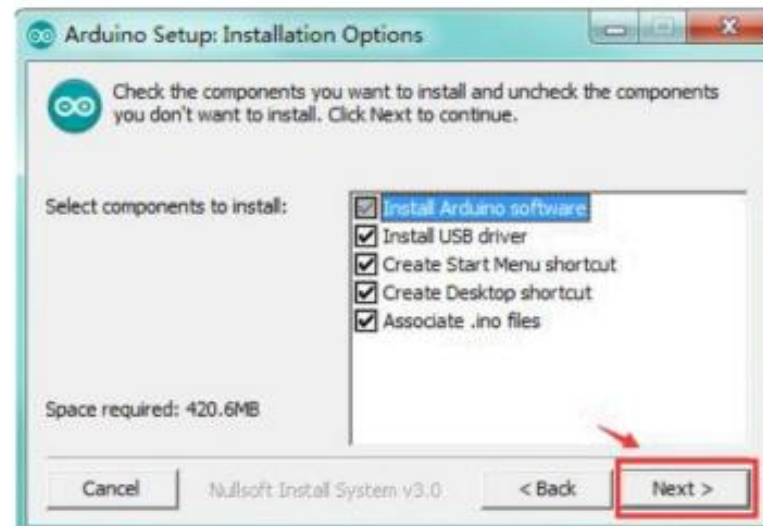
---



Click I Agree to see the following interface.

# LROBROYA

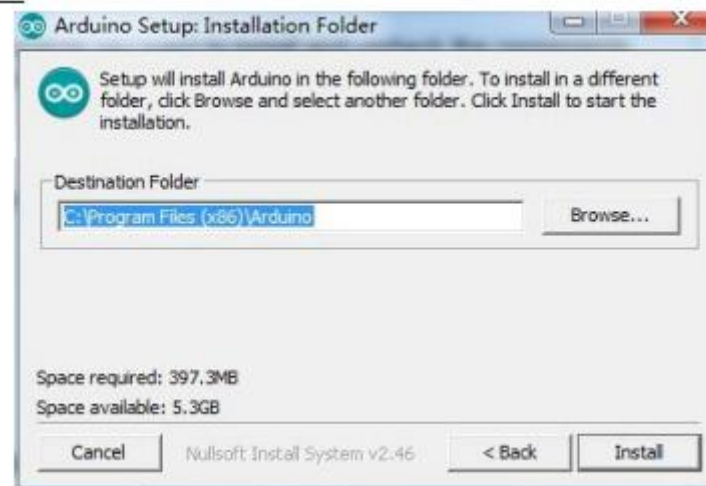
---



Click Next

# LROBROYA

---



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

# LROBROYA

---



Click Install to initiate installation

# LROBROYA

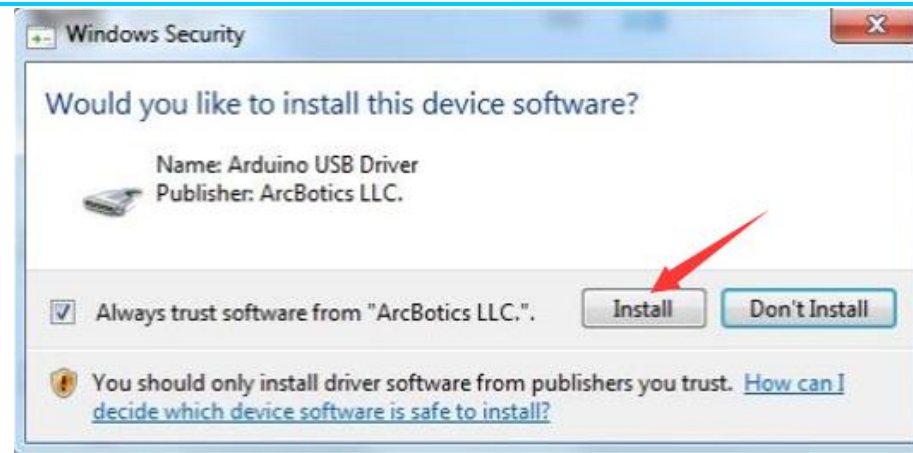
---



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



# LROBROYA

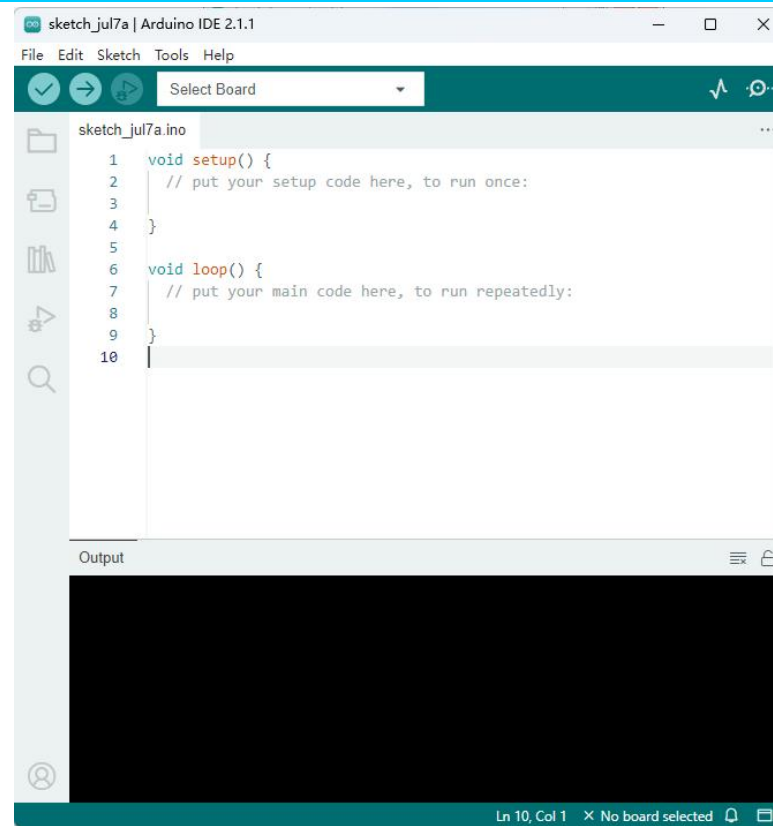


Next, the following icon appears on the desktop



Double-click to enter the desired development environment

# LROBROYA



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.

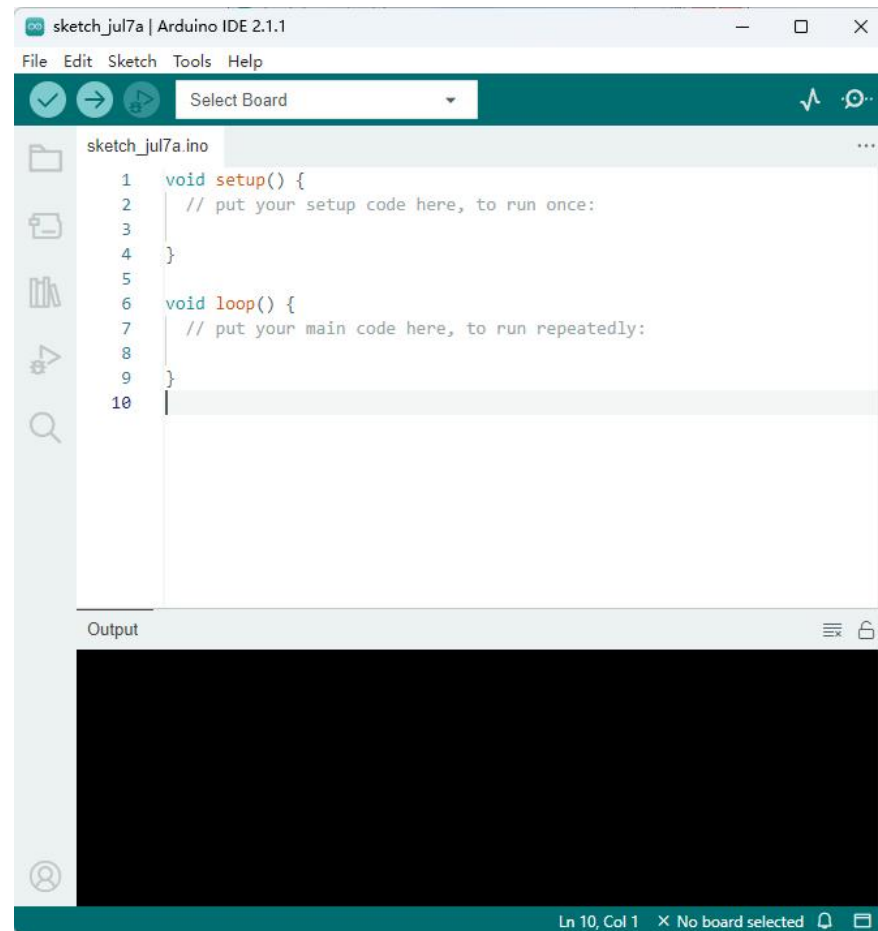
# LROBROYA

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

arduino-ide\_2.1.1\_Windows\_64bit

名称	修改日期	类型	大小
drivers	2023/7/5 21:45	文件夹	
examples	2023/7/5 21:45	文件夹	
hardware	2023/7/5 21:45	文件夹	
java	2023/7/5 21:45	文件夹	
lib	2023/7/5 21:45	文件夹	
libraries	2023/7/5 21:45	文件夹	
reference	2023/7/5 21:45	文件夹	
tools	2023/7/5 21:45	文件夹	
tools-builder	2023/7/5 21:45	文件夹	
arduino	2017/6/1 0:58	应用程序	395 KB
arduino.l4j	2017/6/1 0:58	配置设置	1 KB
arduino_debug	2017/6/1 0:58	应用程序	393 KB
arduino_debug.l4j	2017/6/1 0:58	配置设置	1 KB
arduino-builder	2017/6/1 0:58	应用程序	3,214 KB
libusb0.dll	2017/6/1 0:58	应用程序扩展	43 KB
msvcp100.dll	2017/6/1 0:58	应用程序扩展	412 KB
msvcr100.dll	2017/6/1 0:58	应用程序扩展	753 KB
revisions	2017/6/1 0:58	文本文档	83 KB
uninstall	2023/7/5 21:45	应用程序	404 KB

# LROBROYA

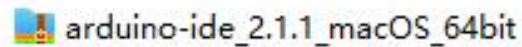




---

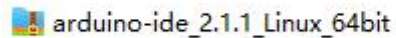
## Installing Arduino (Mac OS X)

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.

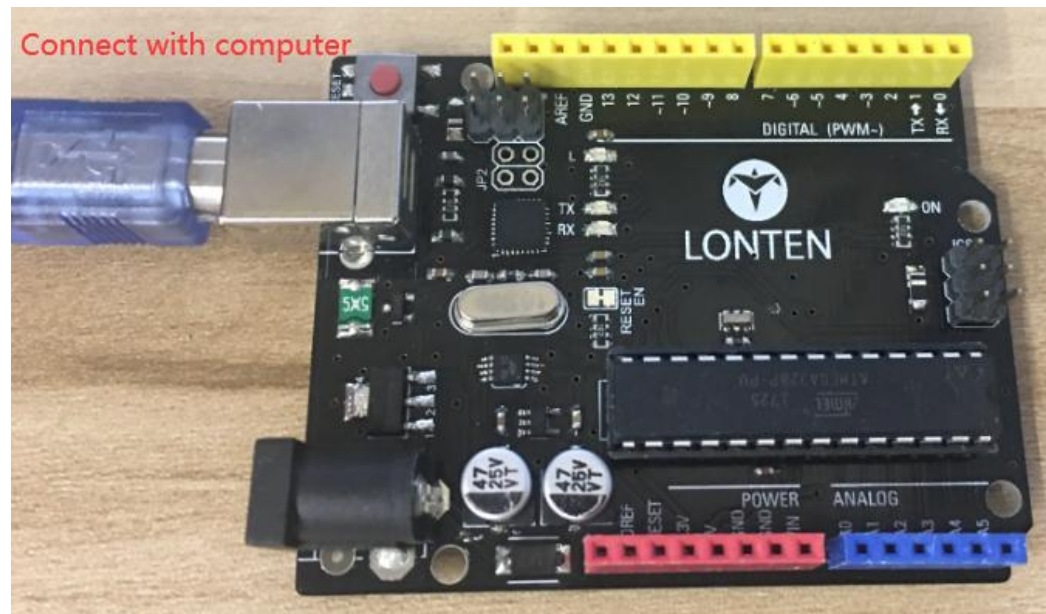


# LROBRUYA

## How to Install Arduino Driver

**For Windows**

**Arduino UNO R3 board**





---

Serial communication interface: D0 is RX, D1 is TX

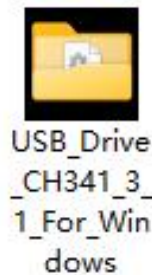
PWM interface (pulse width modulation): D3 D5 D6 D9 D10 D11

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.



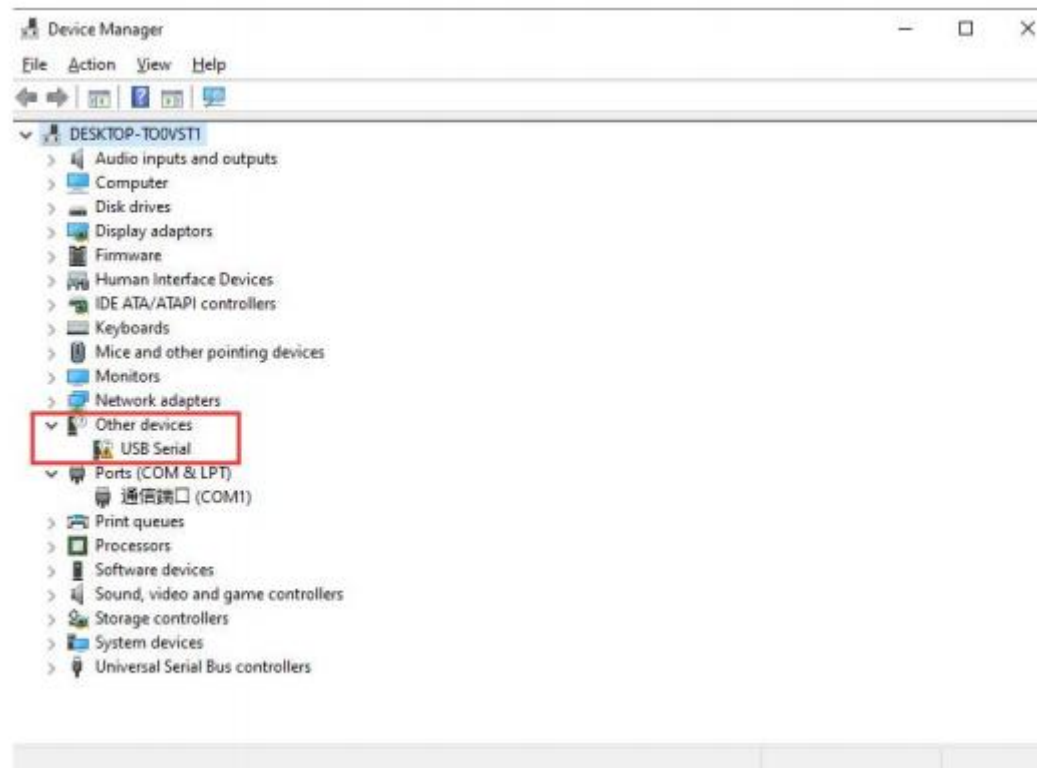
Plug one end of your USB cable into the Arduino UNO R3 Board and the other into a USB socket on your computer.



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.

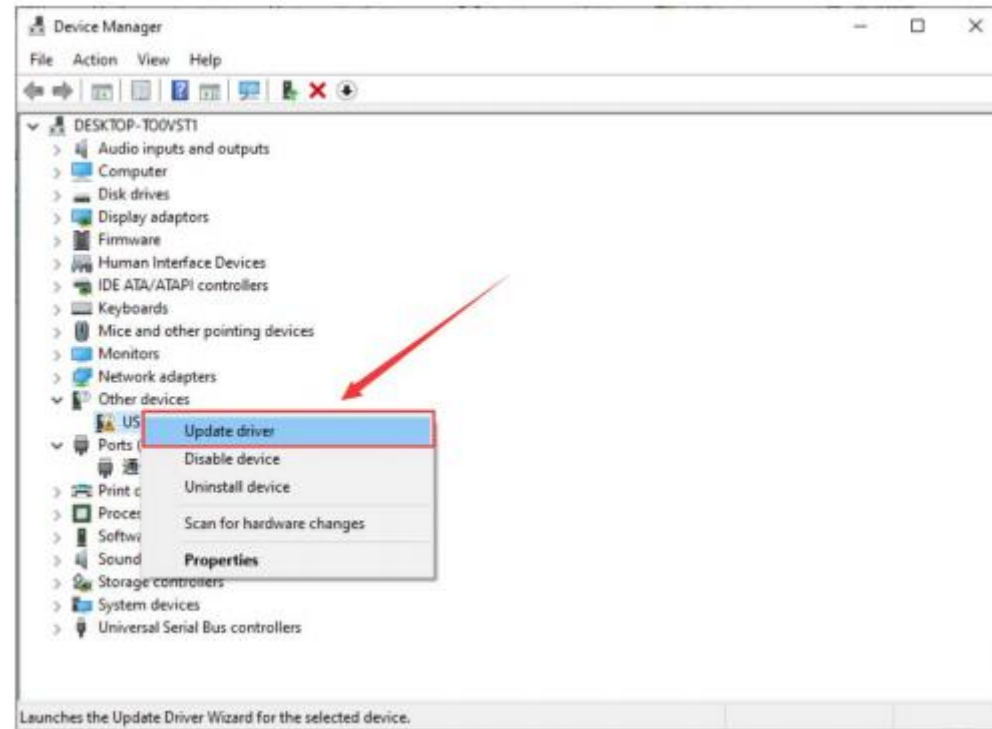


# LROBROYA



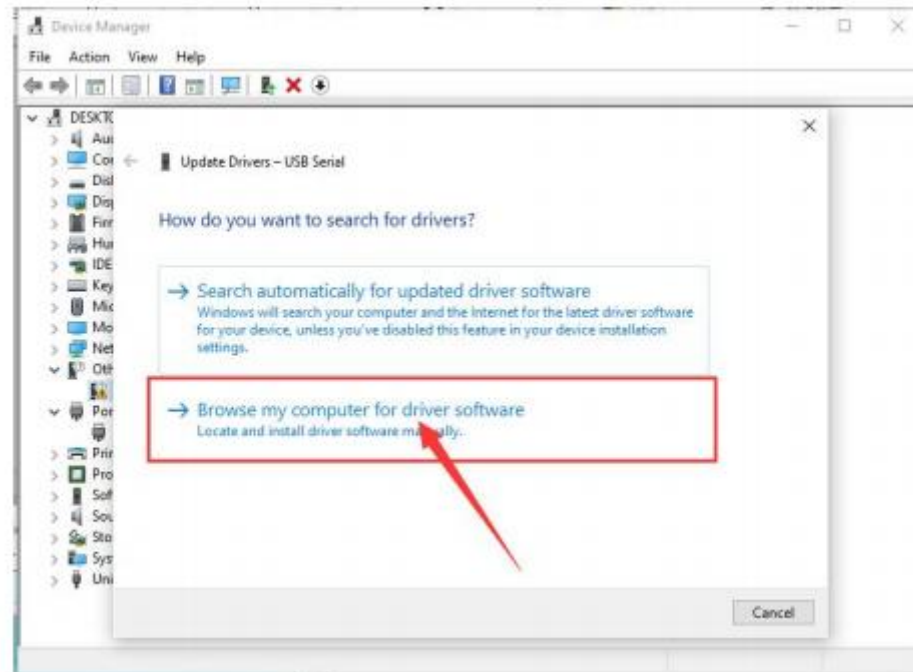
Then right-click on the device and select the top menu option (Update Driver Software...) shown as the figure below.

# LROBBUYA



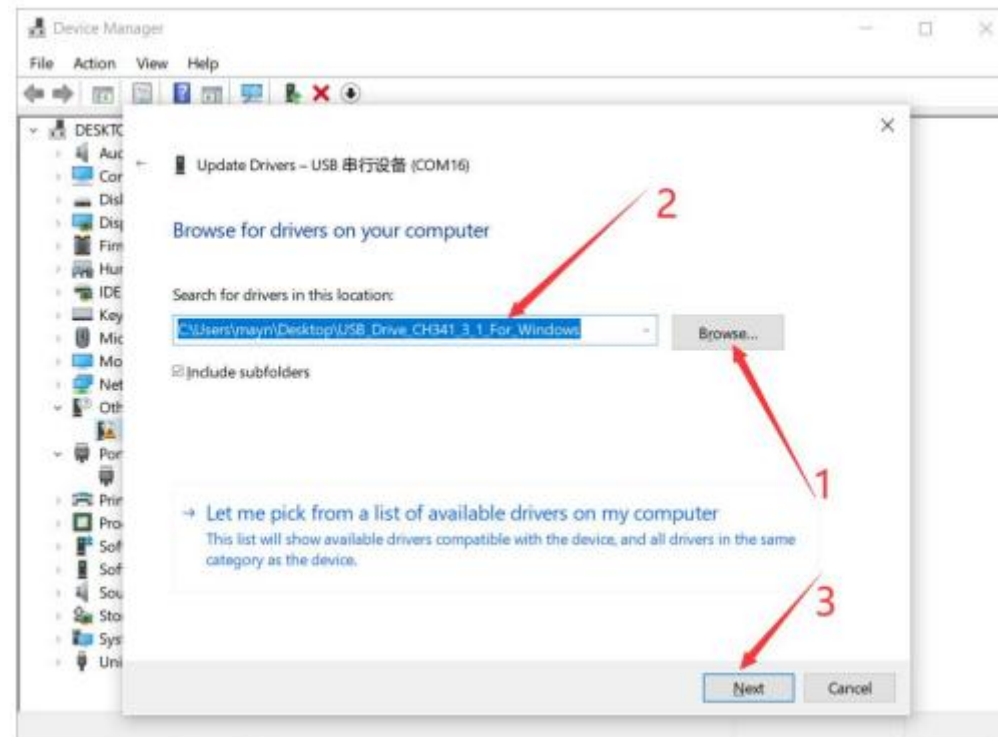
Then it will be prompted to either “Search Automatically for updated driver software” or “Browse my computer for driver software”. Shown as below. In this page, select “Browse my computer for driver software”.

# LROBRUYA



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` )

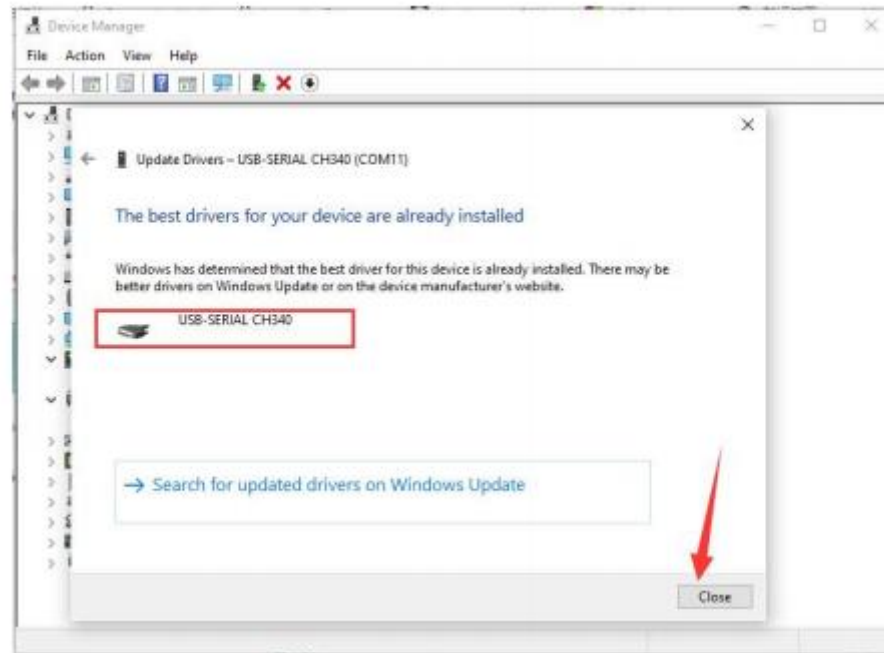
# LROBROYA



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

Installation completed, click “Close”.

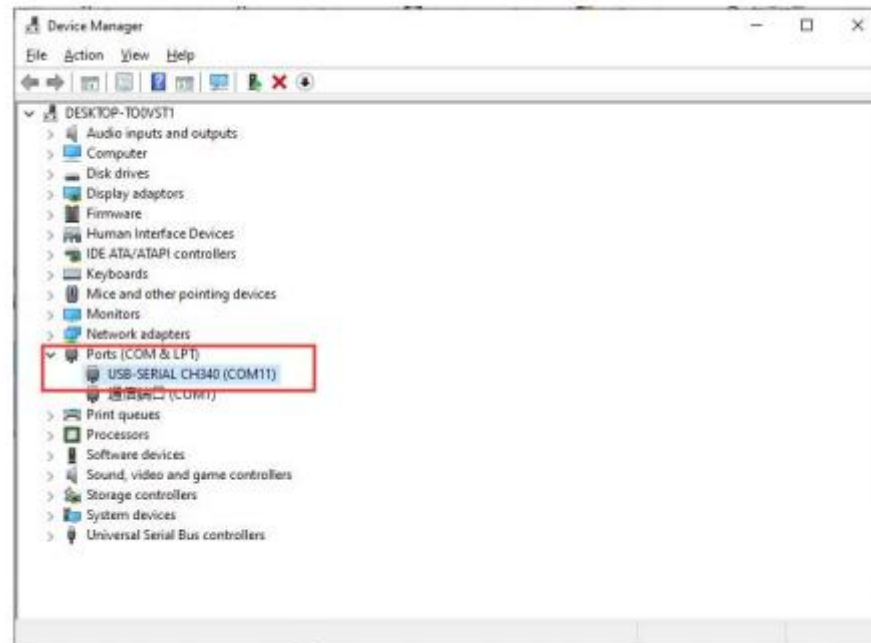
# LROBROYA



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.

# LROB RUYA

---

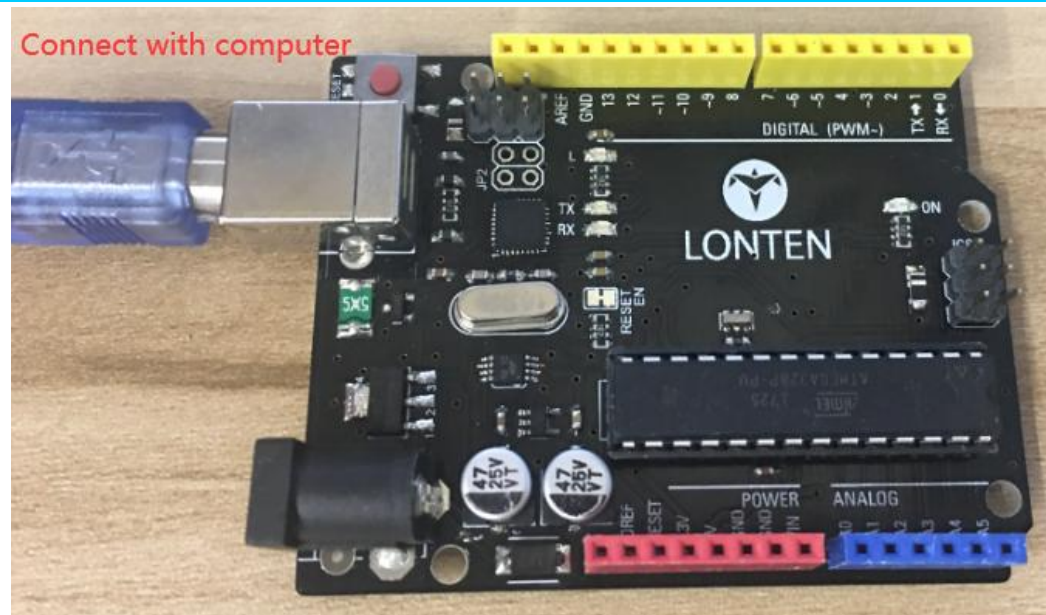


**For MAC System**

**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.

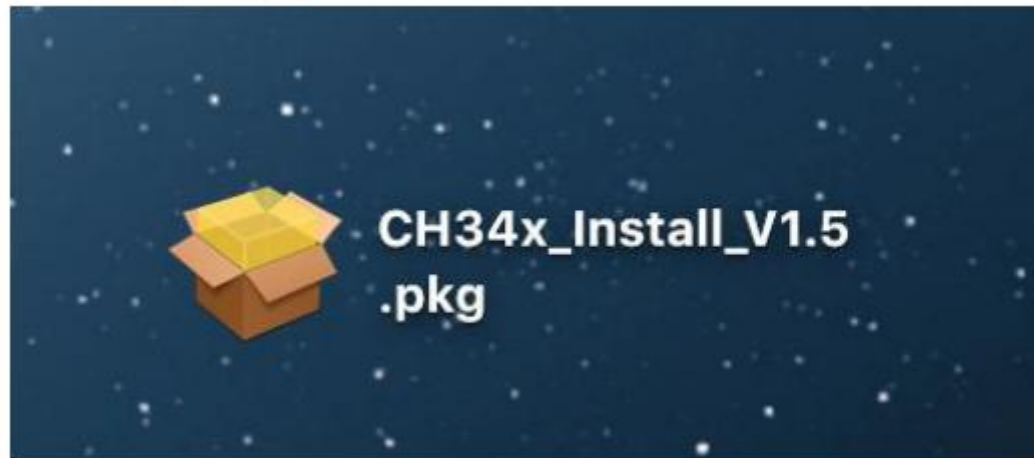
# LROBRUYA



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

# LROB RUYA

---

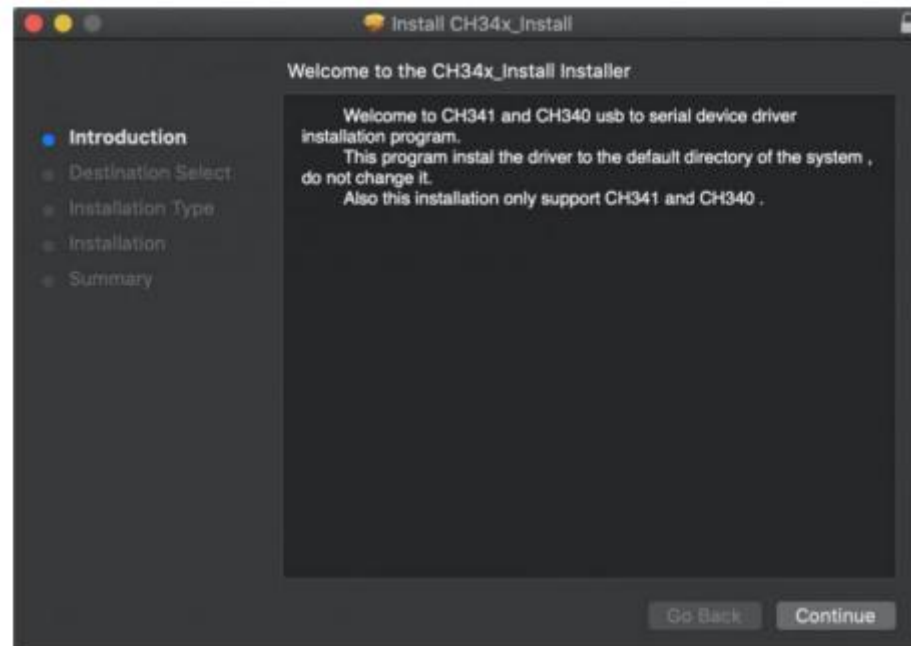


Double-click installation package and tap Continue



# LROB RUYA

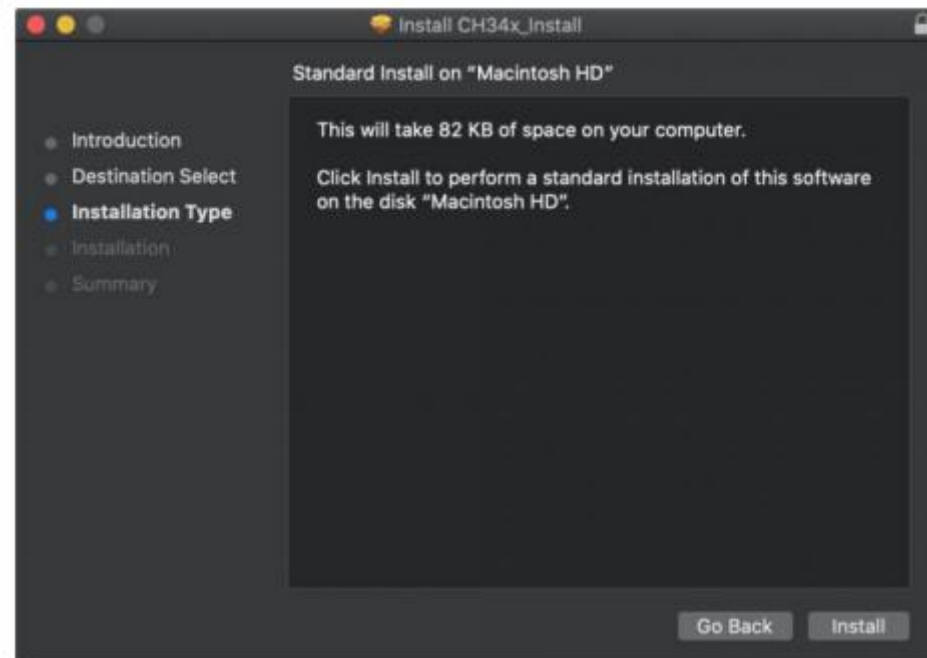
---



Click Install

# LROBRUYA

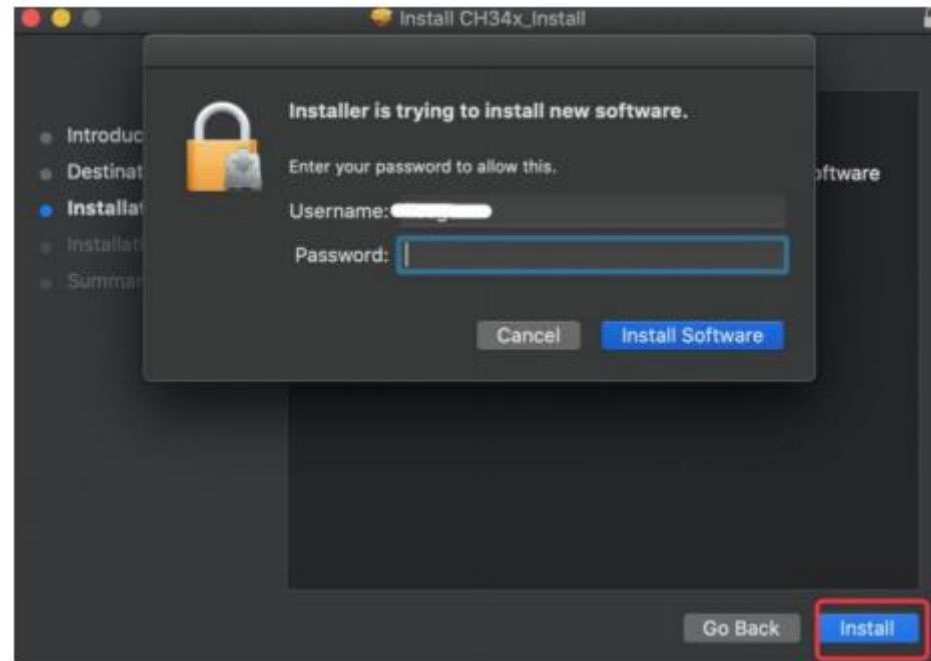
---



Input your user password and click Install Software

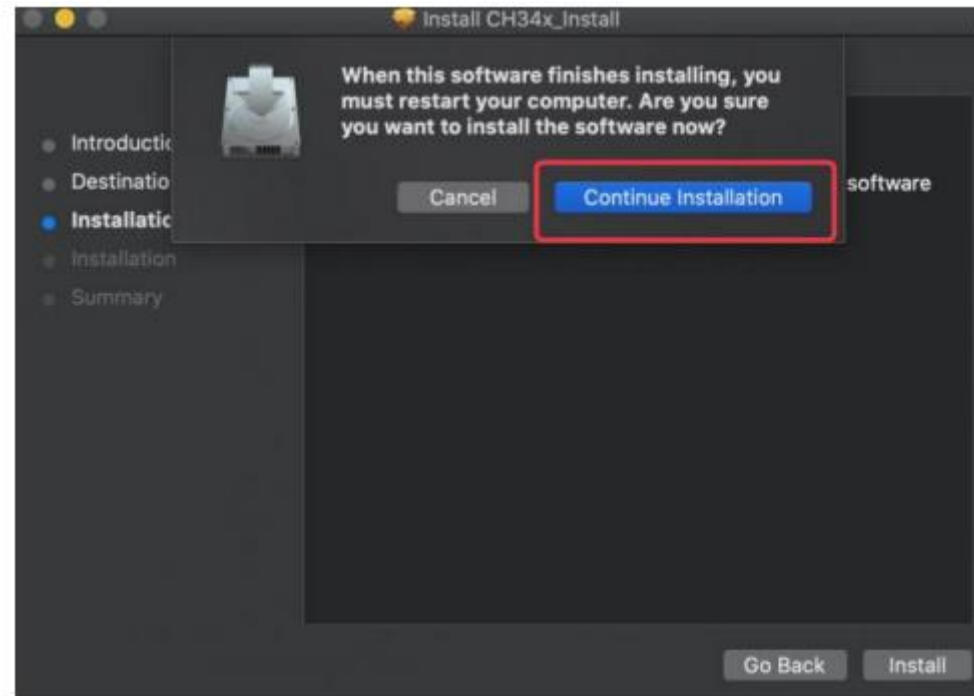
# LROBROYA

---



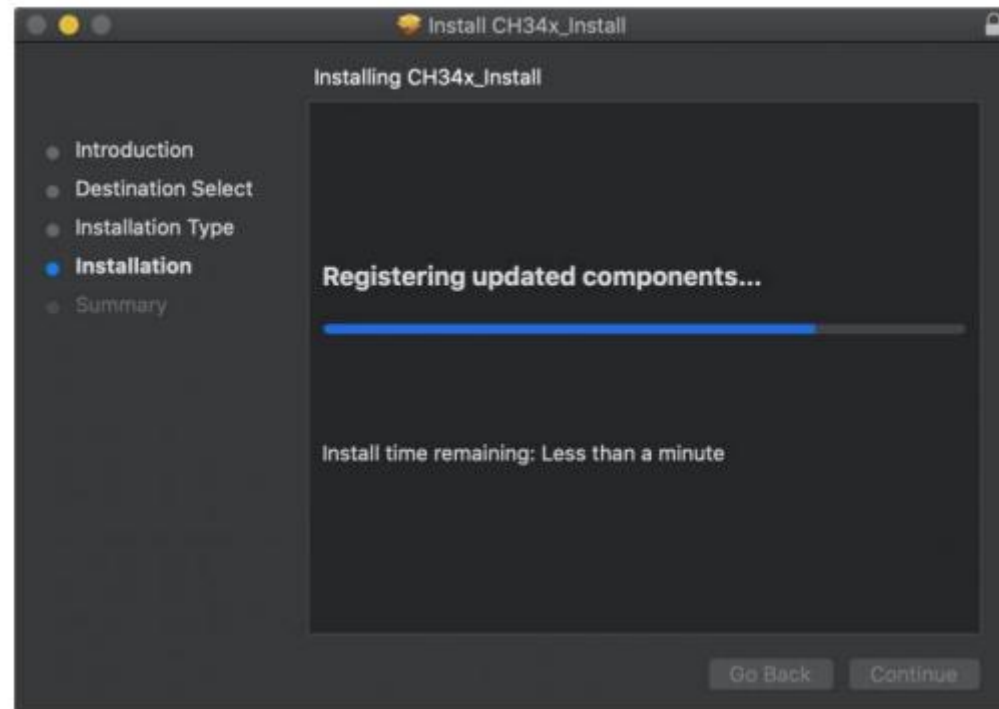
Tap Continue Installation

# LROBROYA



Wait to install

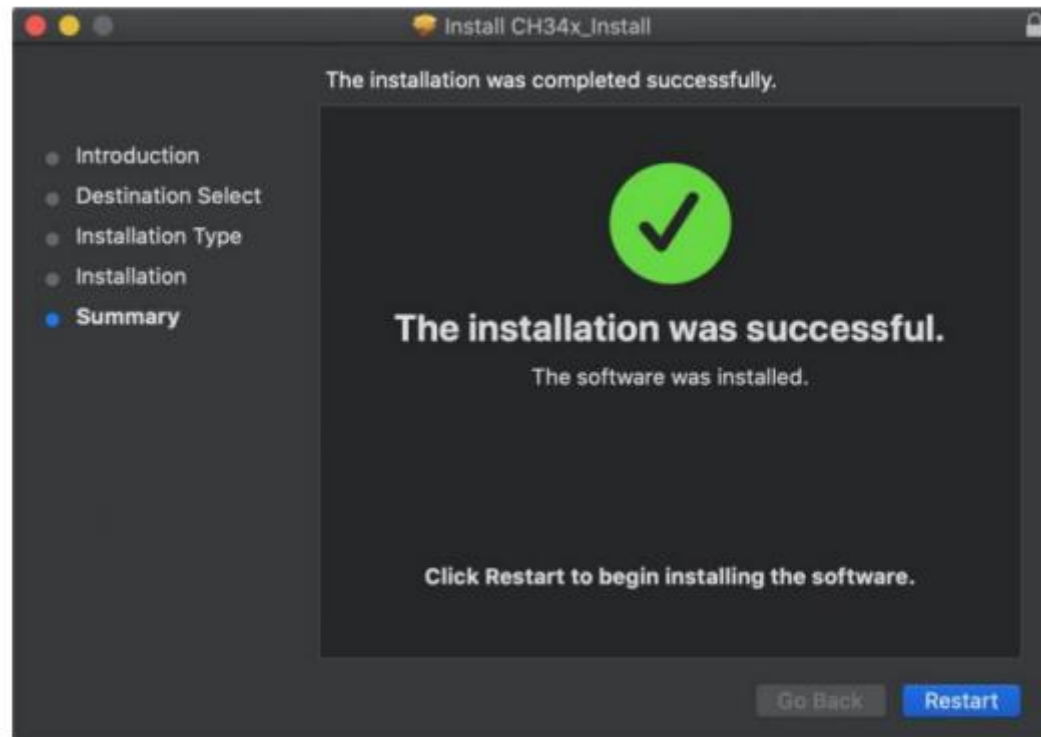
# LROBROYA



Click Restart after the installation is finished

# LROBBUYA

---





---

## 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.

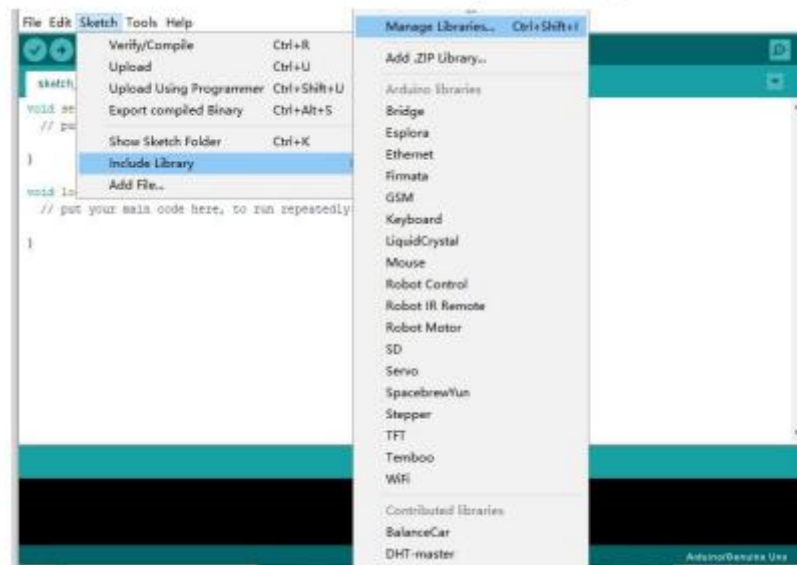
The built-in libraries and some of these additional libraries are listed in the reference. To use the additional libraries, you will need to install them.

# LROBROYA

## 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.

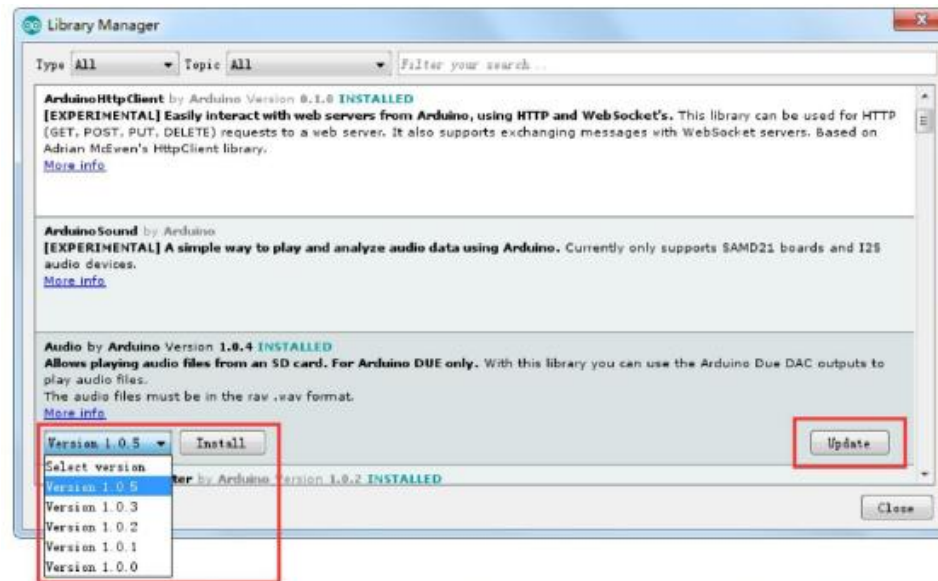




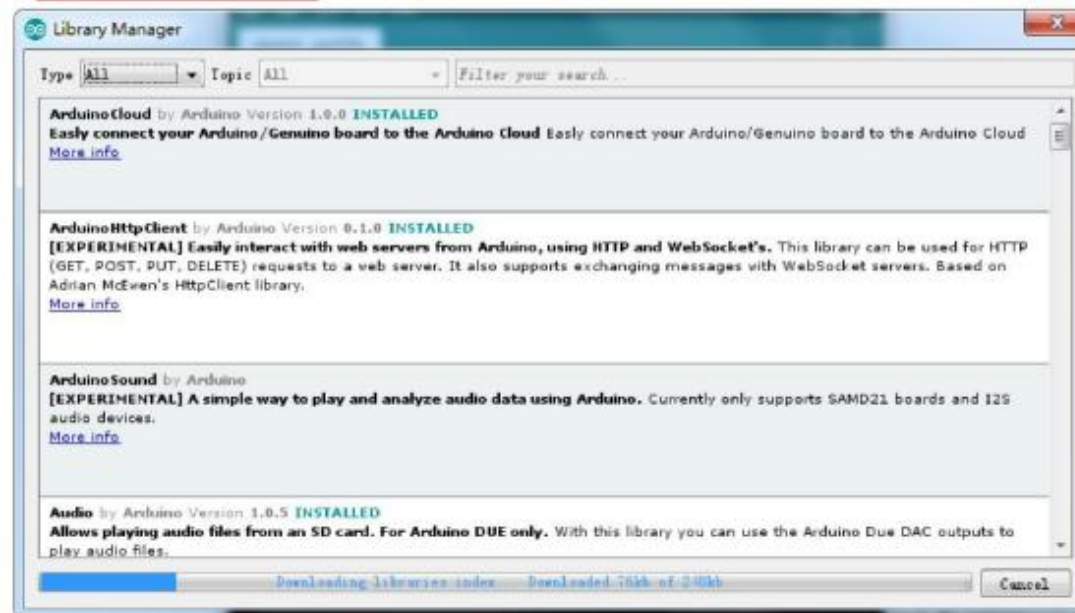
# LROBROYA

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.



# LROBRUYA



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.

# LROBROYA

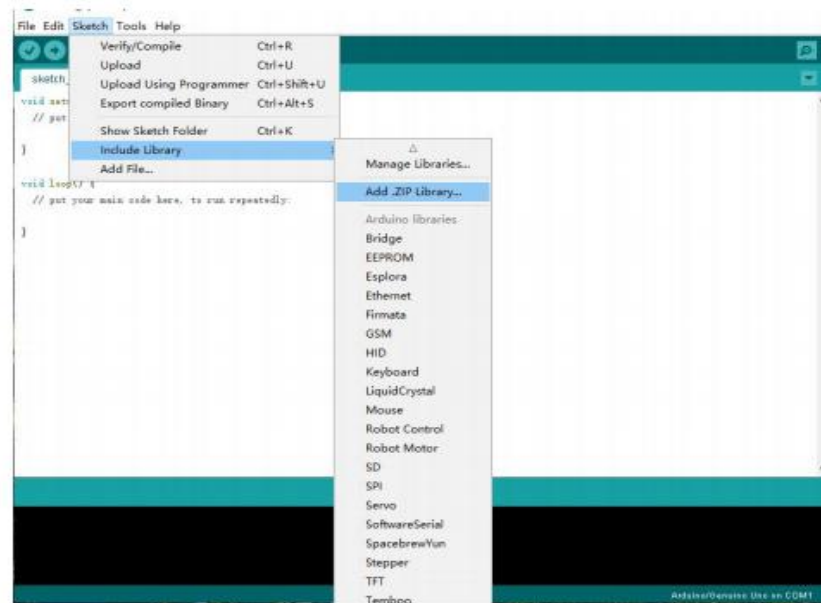


You can now find the new library available in the Include Library menu. If you want to add your own library open a new issue on [Github](#).

## Importing a .zip Library

# LROBROYA

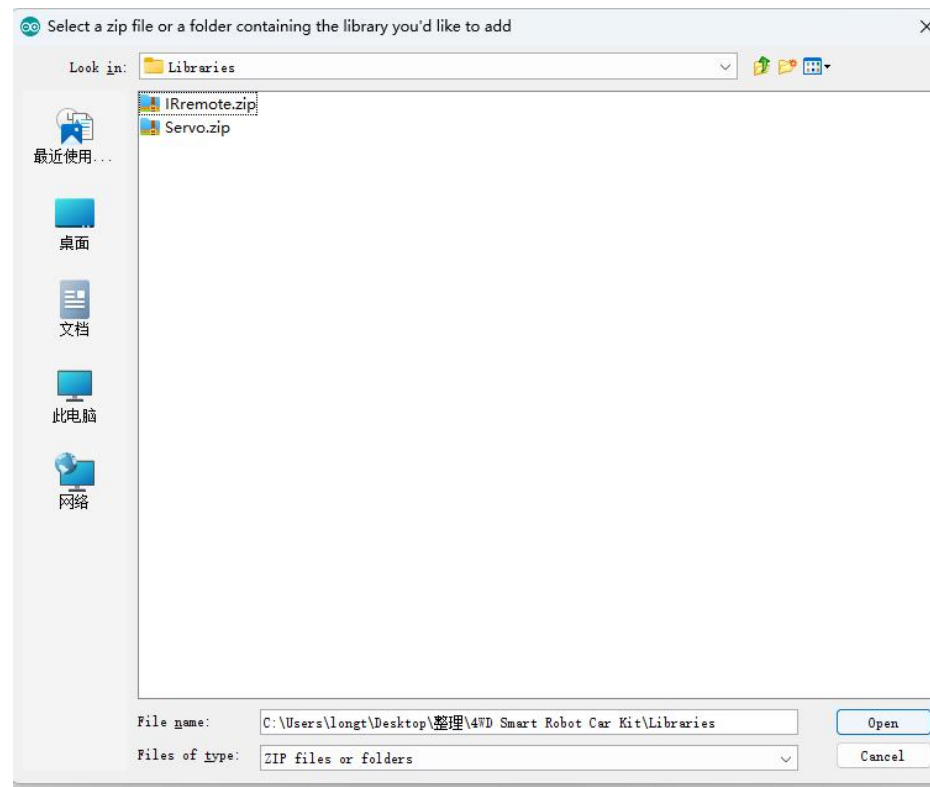
Libraries are often distributed as a ZIP file or folder. The name of the folder is the name of the library. Inside the folder will be a .cpp file, a .h file and often a keywords.txt file, examples folder, and other files required by the library. Starting with version 1.0.5, you can install 3rd party 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".



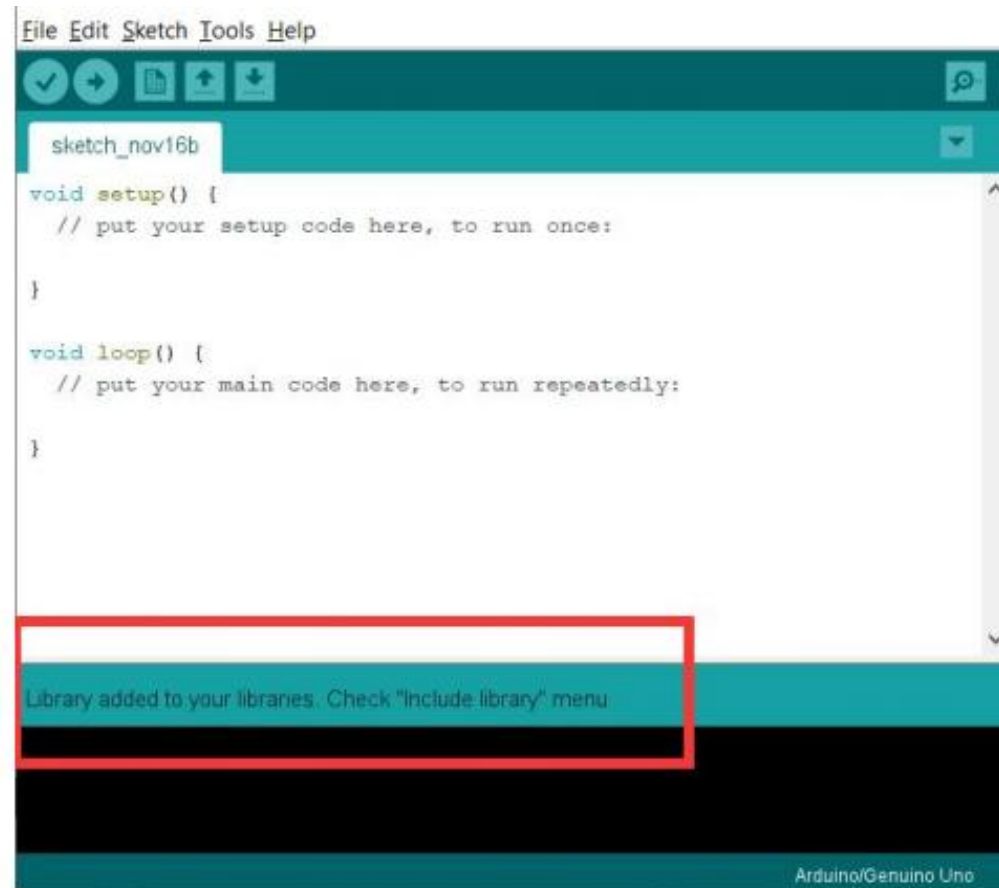
# LROBROYA

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

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



# LROBROYA





---

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:

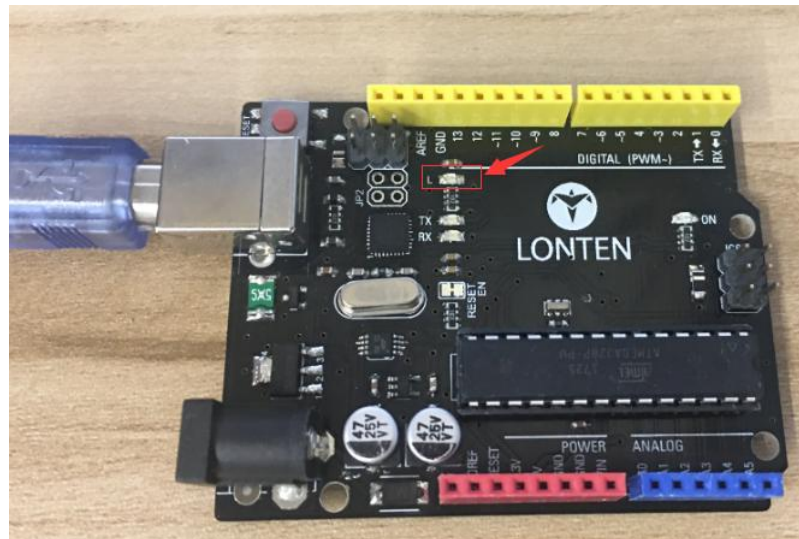
LONTEN Uno R3 Board\* 1

# LROBRUYA

## 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.







---

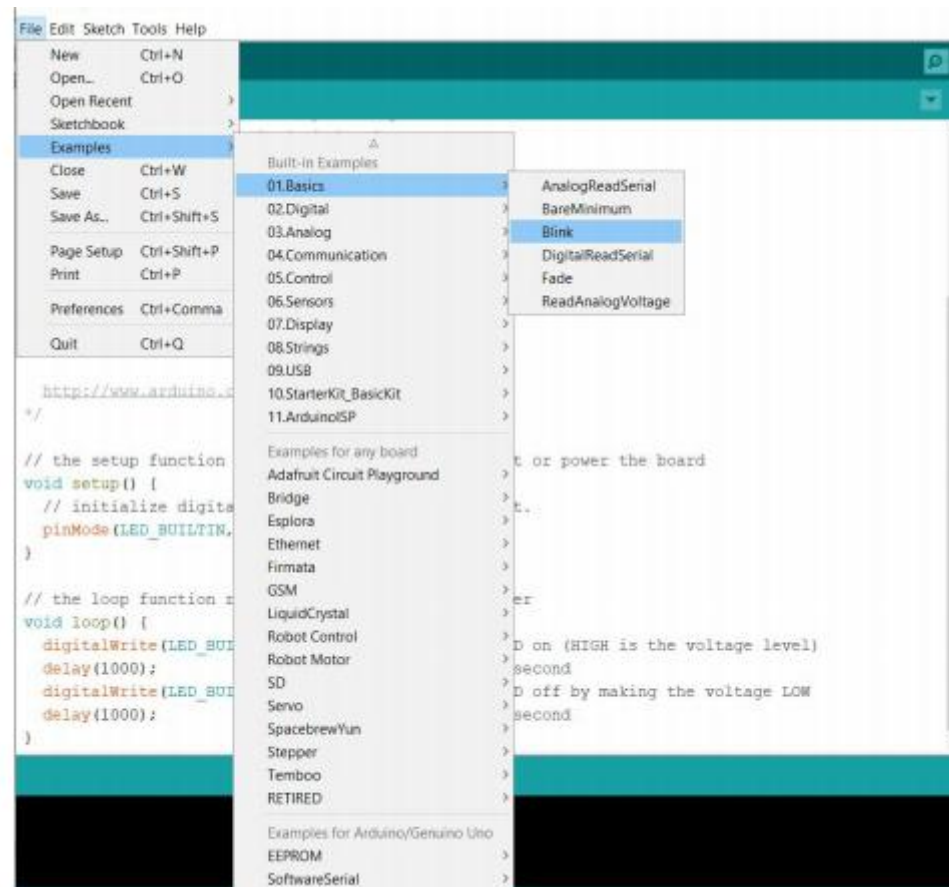
In this Project, we will reprogram the UNO board with our own Blink sketch and then change the rate at which it blinks.

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

# LROBROYA



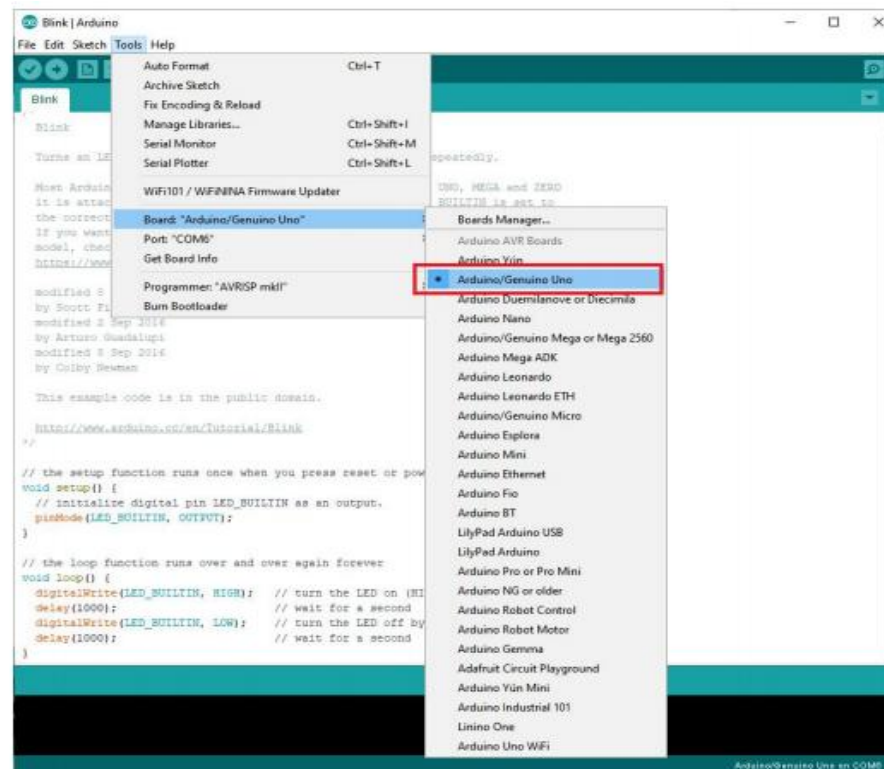
When the sketch window opens, enlarge it so that you can see the entire sketch in the window.

# LROBROYA

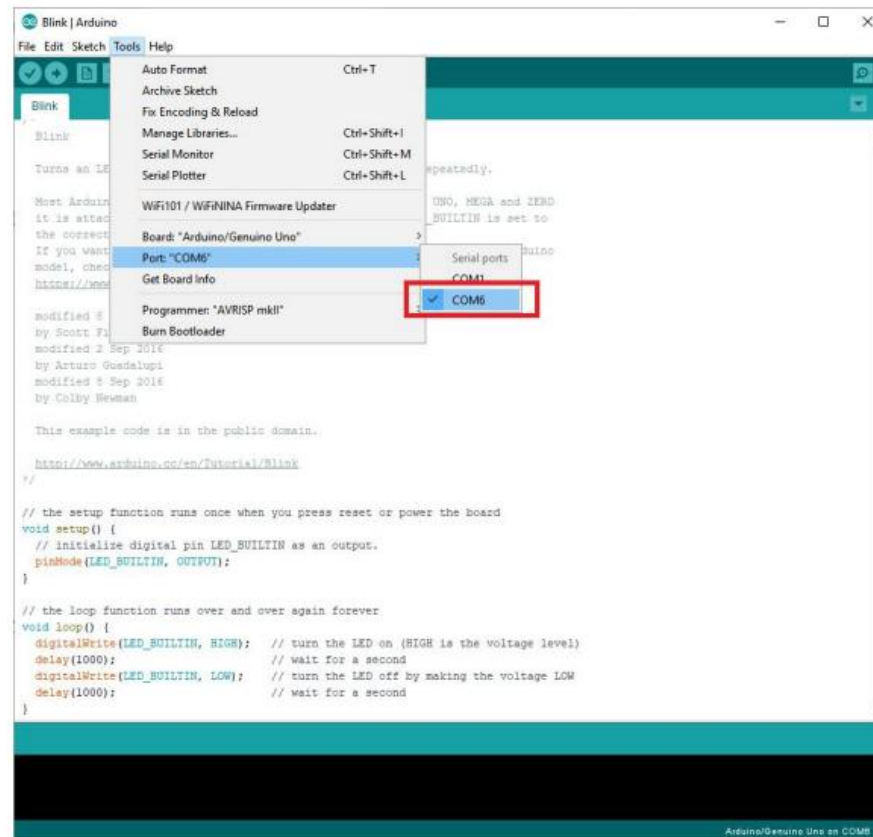
```
File Edit Sketch Tools Help
[Icons] [Serial Monitor Icon]
Blink [Close Icon]
22
23
24 // the setup function runs once when you press reset or power the board
25 void setup() {
26   // initialize digital pin LED_BUILTIN as an output.
27   pinMode(LED_BUILTIN, OUTPUT);
28 }
29
30 // the loop function 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 }
[Serial Monitor Area]
23 Arduino Uno on COM3
```

# LROBROYA

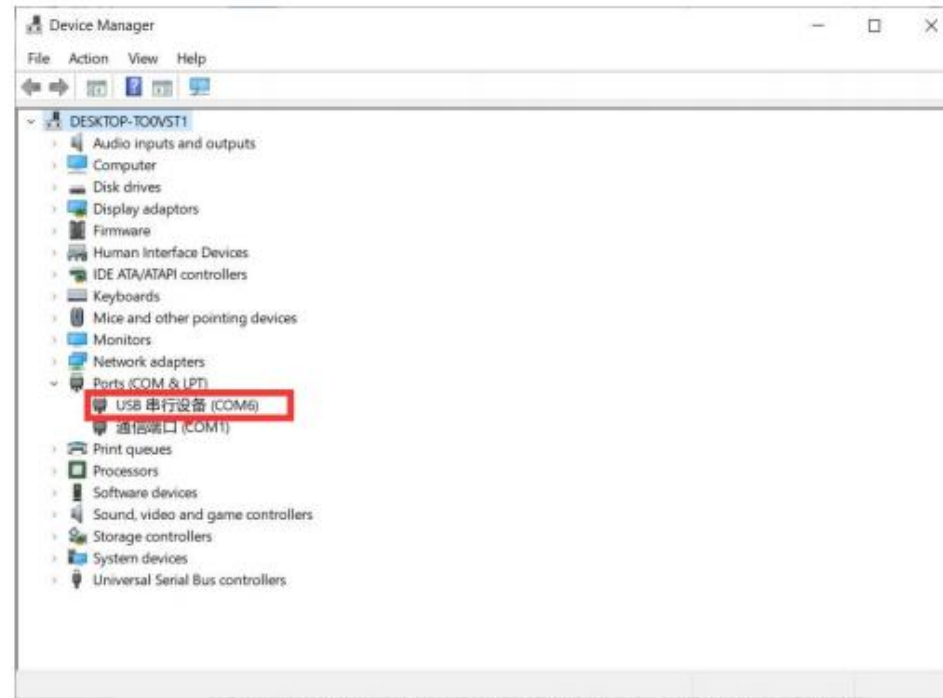
Attach your Arduino board to your computer with the USB cable and check that the 'Board Type' and 'Serial Port' are set correctly.



# LROBBUYA



# LROBROYA



**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**

# LROBROYA

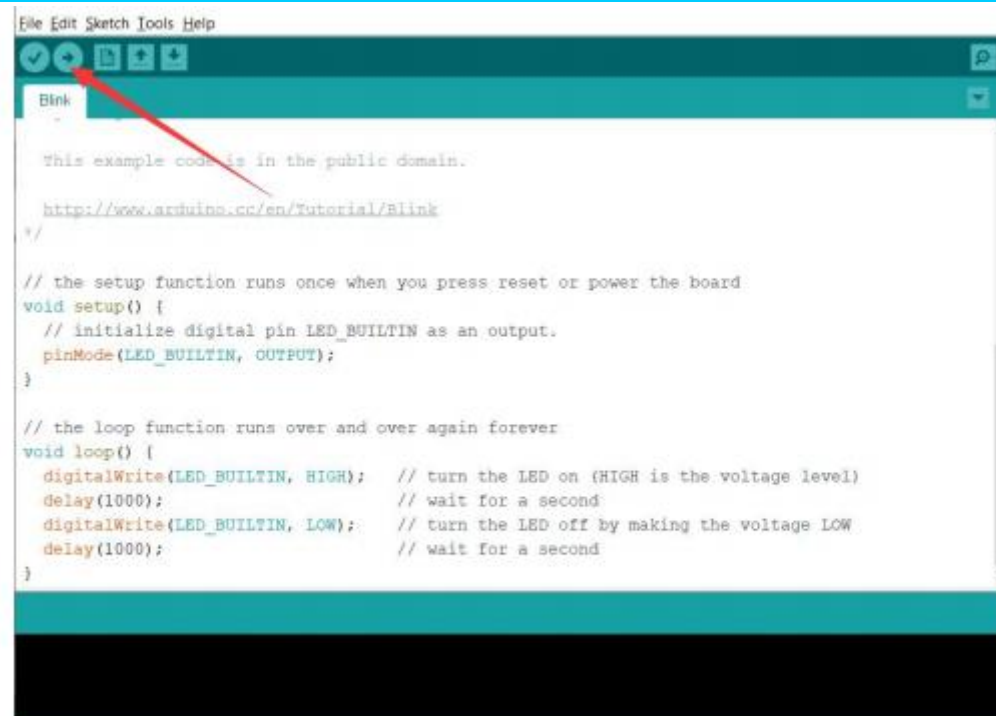
the Serial Port displayed for everyone is different, despite COM 6 chosen here, it could be COM3 or COM4 on your computer. A right COM port is supposed to be COMX (arduino XXX), which is by the certification criteria.

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.

# LROBROYA



```
File Edit Sketch Tools Help

Blink

This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink

*/

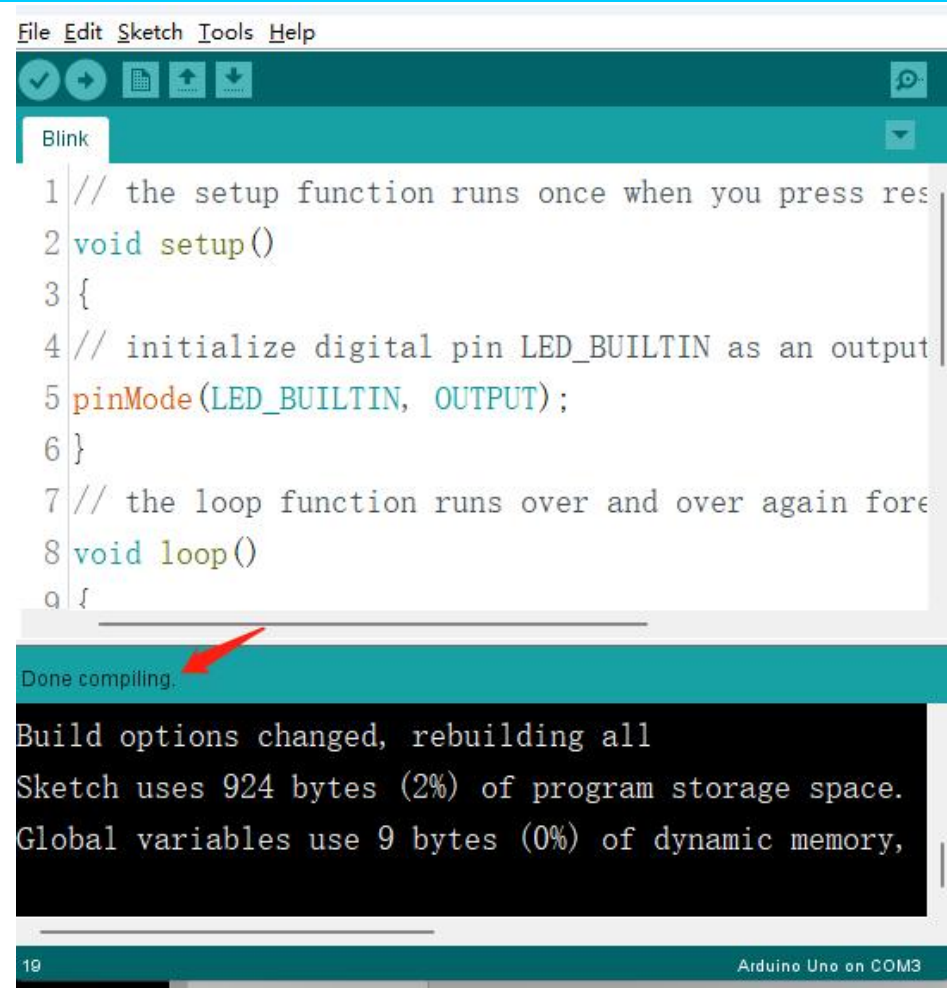
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// 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
}
```

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



# LROBROYA



The screenshot shows the Arduino IDE interface. The menu bar at the top includes File, Edit, Sketch, Tools, and Help. Below the menu bar is a toolbar with icons for saving, undo, redo, and other functions. The main text area displays the code for the 'Blink' sketch. The code is as follows:

```
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()
9 {
```

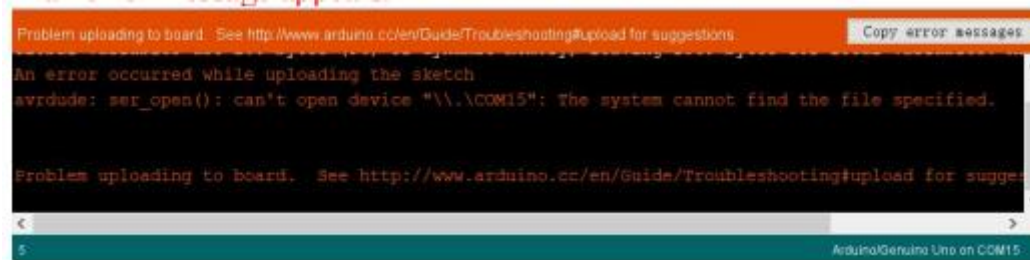
Below the code editor, a status bar indicates 'Done compiling.' with a red arrow pointing to it. Below the status bar, a black box displays the following text:

```
Build options changed, rebuilding all
Sketch uses 924 bytes (2%) of program storage space.
Global variables use 9 bytes (0%) of dynamic memory,
```

At the bottom of the IDE, the status bar shows '19' on the left and 'Arduino Uno on COM3' on the right.

# LROBROYA

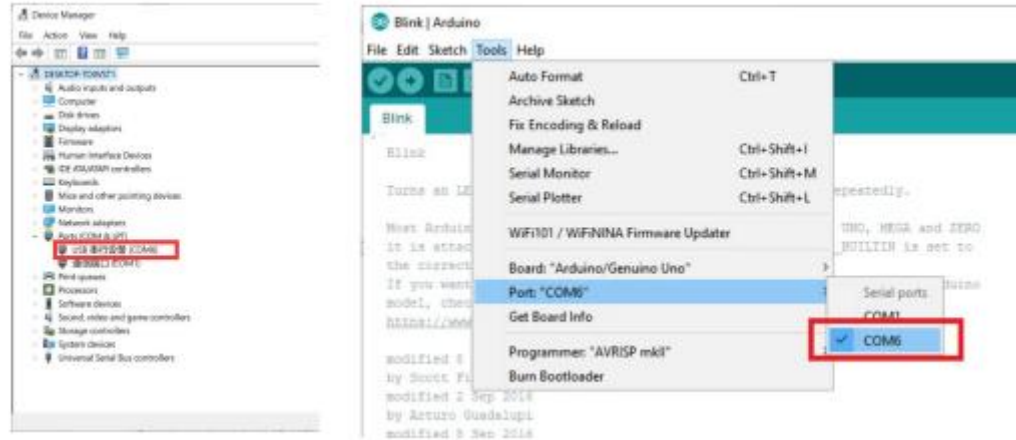
If an error message appears.



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.

# LROBROYA



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.



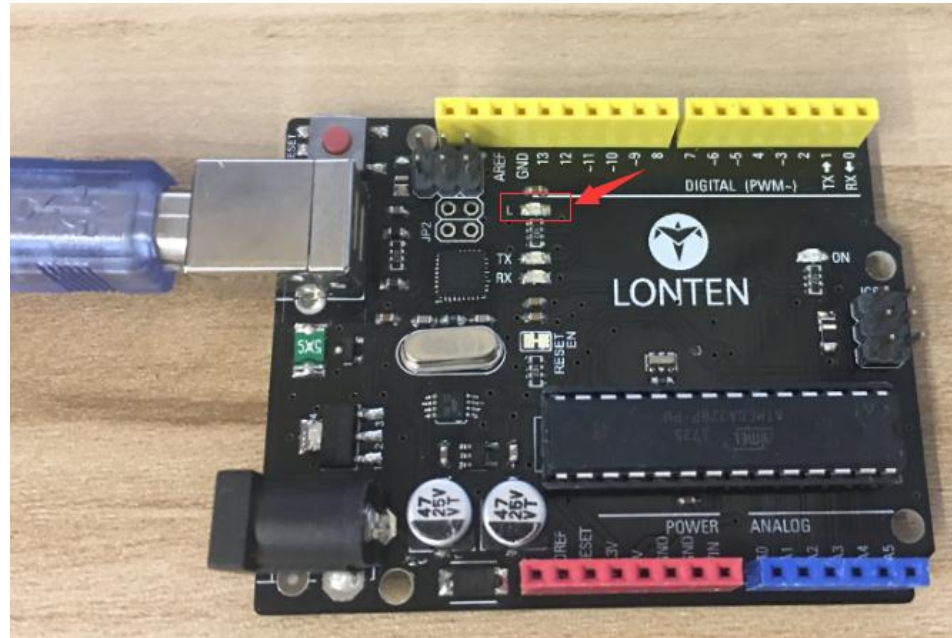
---

## 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
}
```

# LROBRUYA



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.

# LROB RUYA

---

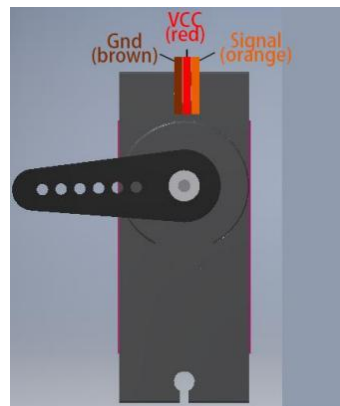
## Lesson 1 Servo

### About this lesson:

In this lesson, you will learn how to control a servo motor using Arduino UNO development board.

The servo motor has three leads. The color of the leads varies between servo motors, but the red lead is always 5V and GND will either be brown. The red one is the power wire and should be connected to the 5v port and this is usually orange.

This control lead is connected to digital pin 13.



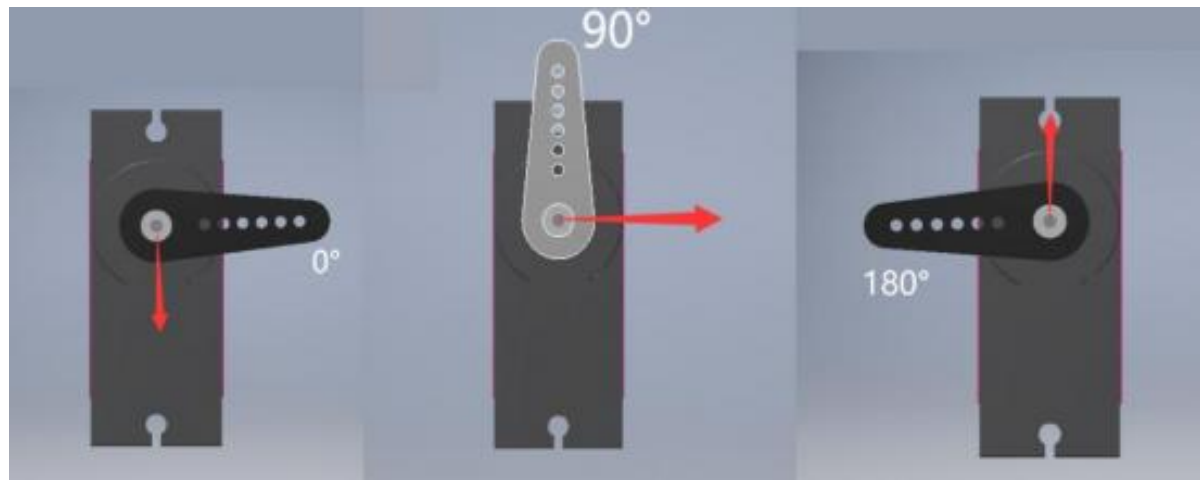
# LROBRUYA

---

## Introduction

Servo motors are great devices that can turn to a specified position.

Usually, they have a servo arm that can turn 180 degrees. Using the Arduino, we can tell a servo to go to a specified position and it will go there. As simple as that!



Servo motors were first used in the Remote Control (RC) world, usually to control the steering of RC cars or the flaps on a RC plane. With time, they found their uses in robotics, automation, and of course, the Arduino world.



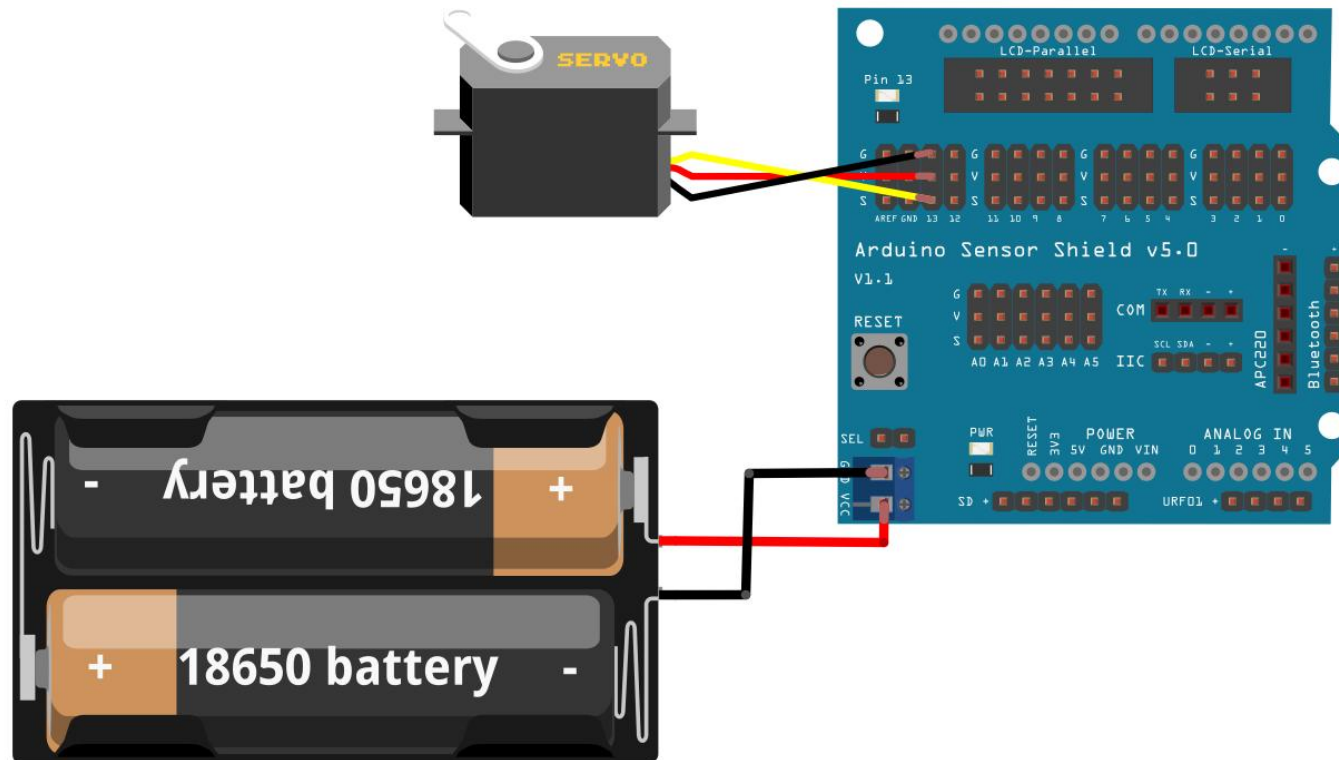
---

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. The Arduino drive capacity is limited. So if you need to control more than one motor, you will need external power.



# LROBRUYA

## Connection diagram





---

### **Attention**

After connecting, please open the the program and load up the code - Lesson\_1\_Servo onto your Arduino board. Before you can run this, make sure that you have installed the < Servo> library or re-install it, if necessary. Otherwise, your code won't work.

### **Result**

After uploading the code, the servo motor rotates from 0 degrees to 180 degrees, 1 degree at a time. Then rotate from 180 degrees to 0 degrees, one degree at a time.

## **Lesson 2 Ultrasonic Sensor Module**

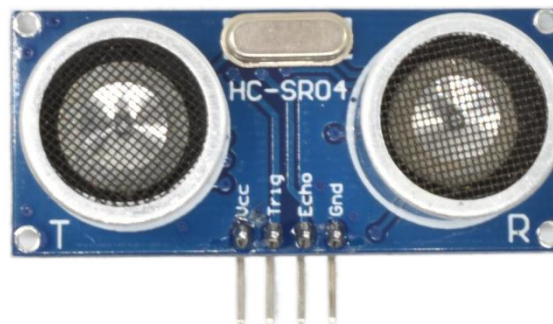
### **About this lesson:**

Ultrasonic sensor is great for all kind of projects that need distance measurements, avoiding obstacles as examples.

# LROB RUYA

It works like a bat's eye. Determine the distance of obstacles in front after receiving and receiving high-frequency sound waves.

As the following picture shown, it is our ultrasonic module. It has two something like eyes. One is transmitting end, the other is receiving end.



## **Introduction:**

Ultrasonic sensor module HC-SR04 provides 2cm-400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit.

The basic principle of work:



---

(1) Using IO trigger for at least 10us high level signal

(2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.

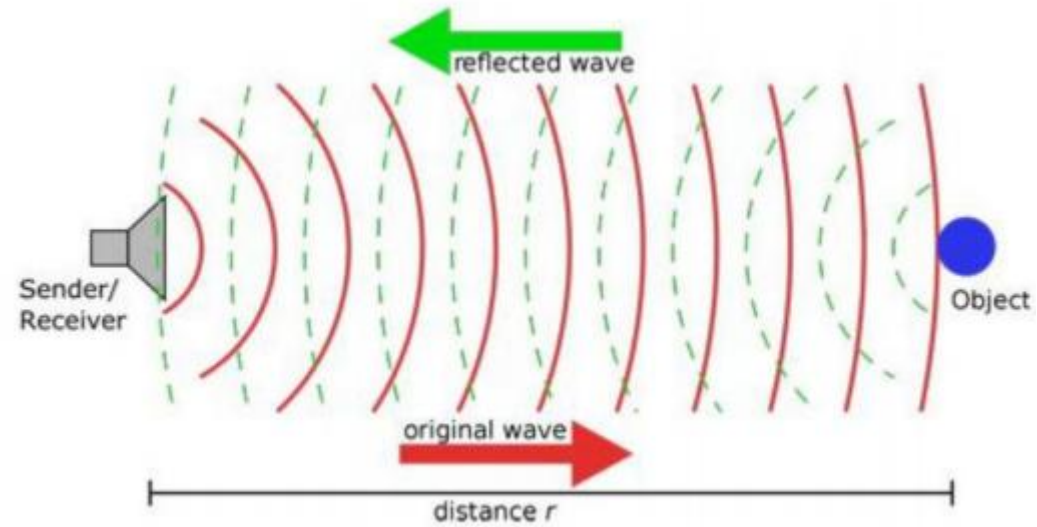
(3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Test distance = (high level time × velocity of sound (340m/s) / 2

The Timing diagram is shown below. You only need to supply a short 10us pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula:  $\text{us} / 58 = \text{centimeters}$  or  $\text{us} / 148 = \text{inch}$ ; or: the range = high level time \* velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.

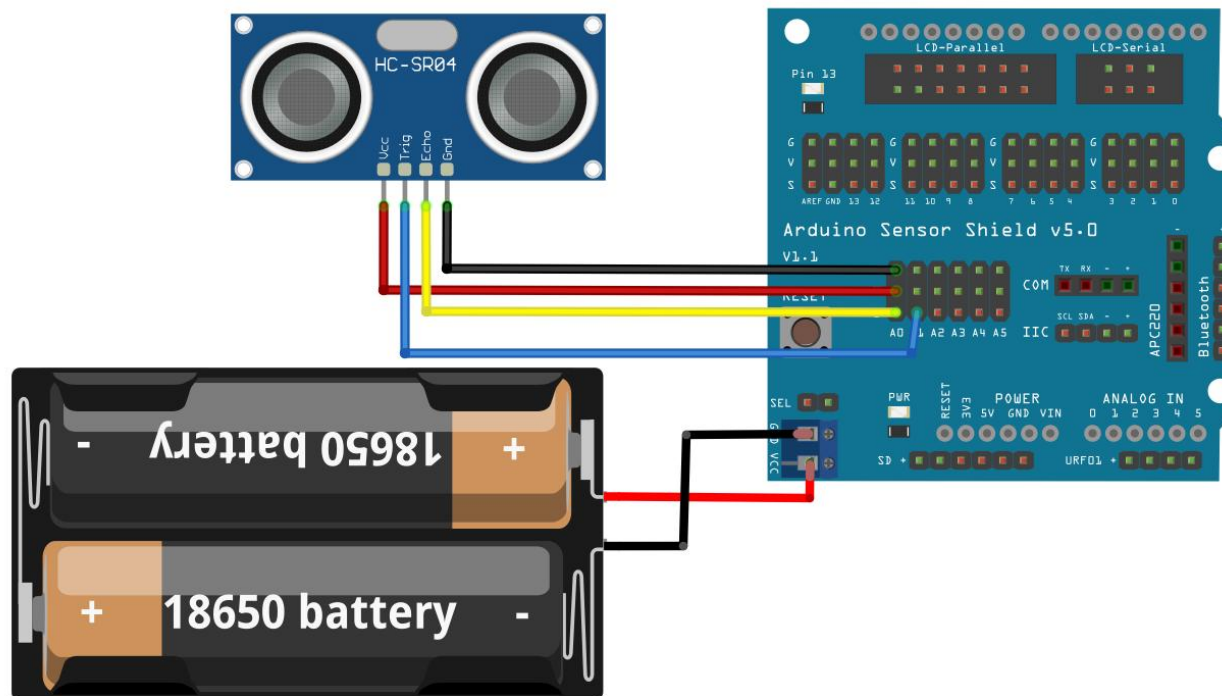
# LROBRUYA

---



# LROBRUYA

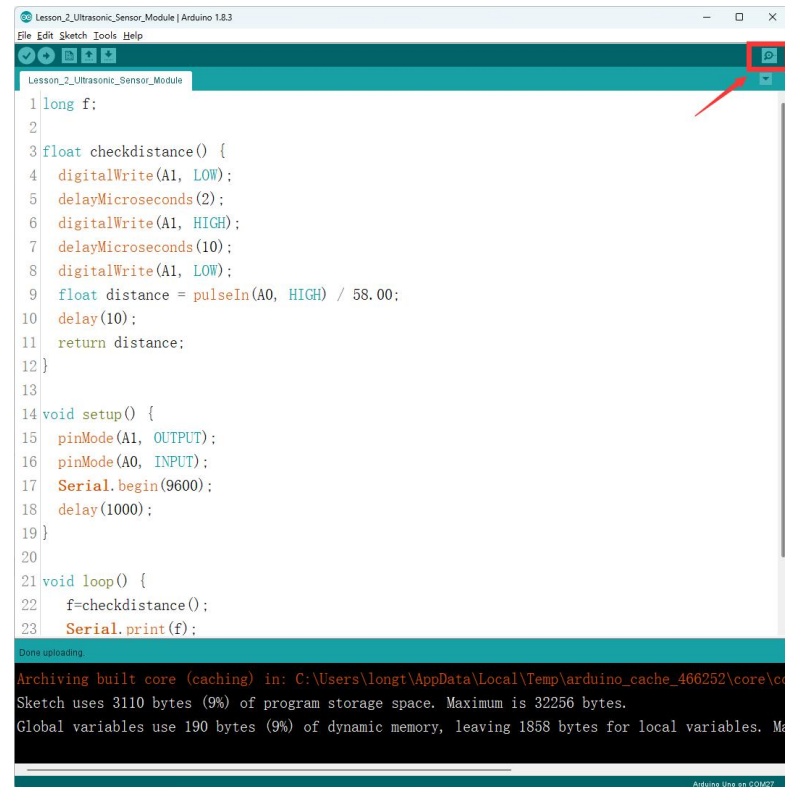
## Wiring diagram



# LROBROYA

## Result

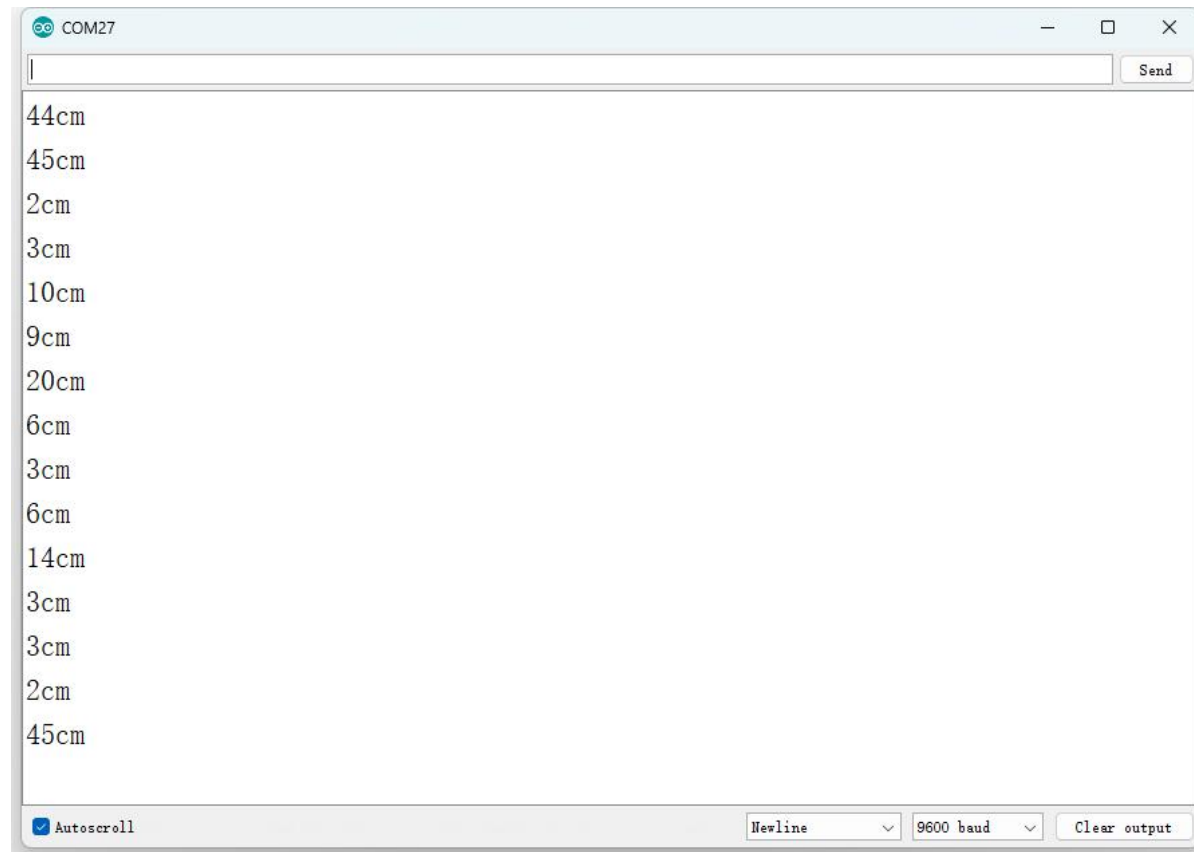
After uploading the code, click the button in the upper right corner to open the serial monitor to view the measured distance.



```
Lesson_2_Ultrasonic_Sensor_Module | Arduino 1.8.3
File Edit Sketch Tools Help
Lesson_2_Ultrasonic_Sensor_Module
1 long f;
2
3 float checkdistance() {
4   digitalWrite(A1, LOW);
5   delayMicroseconds(2);
6   digitalWrite(A1, HIGH);
7   delayMicroseconds(10);
8   digitalWrite(A1, LOW);
9   float distance = pulseIn(A0, HIGH) / 58.00;
10  delay(10);
11  return distance;
12 }
13
14 void setup() {
15   pinMode(A1, OUTPUT);
16   pinMode(A0, INPUT);
17   Serial.begin(9600);
18   delay(1000);
19 }
20
21 void loop() {
22   f=checkdistance();
23   Serial.print(f);
24 }
Done uploading.
Archiving built core (caching) in: C:\Users\longt\AppData\Local\Temp\arduino_cache_466252\core\co
Sketch uses 3110 bytes (9%) of program storage space. Maximum is 32256 bytes.
Global variables use 190 bytes (9%) of dynamic memory, leaving 1858 bytes for local variables. Ma
Arduino Uno on COM27
```

# LROBRUYA

Then you can see the data as blow:







---

## Lesson 3 L298N Motor Driver

### **About this lesson:**

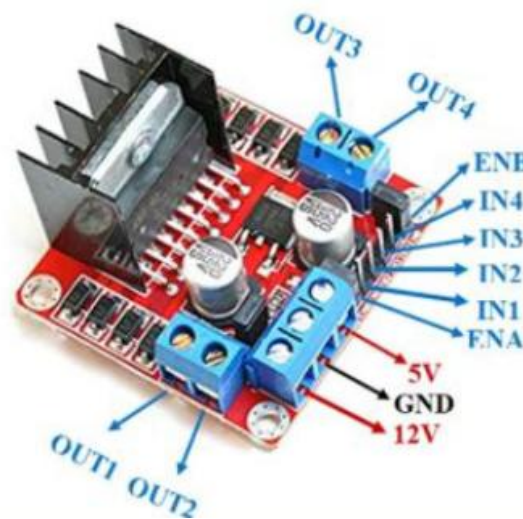
In this lesson, you will learn how to use a L298N Motor Driver module.

### **Component Introduction**

This L298N Motor Driver Module is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control.

# LROBRUYA

---



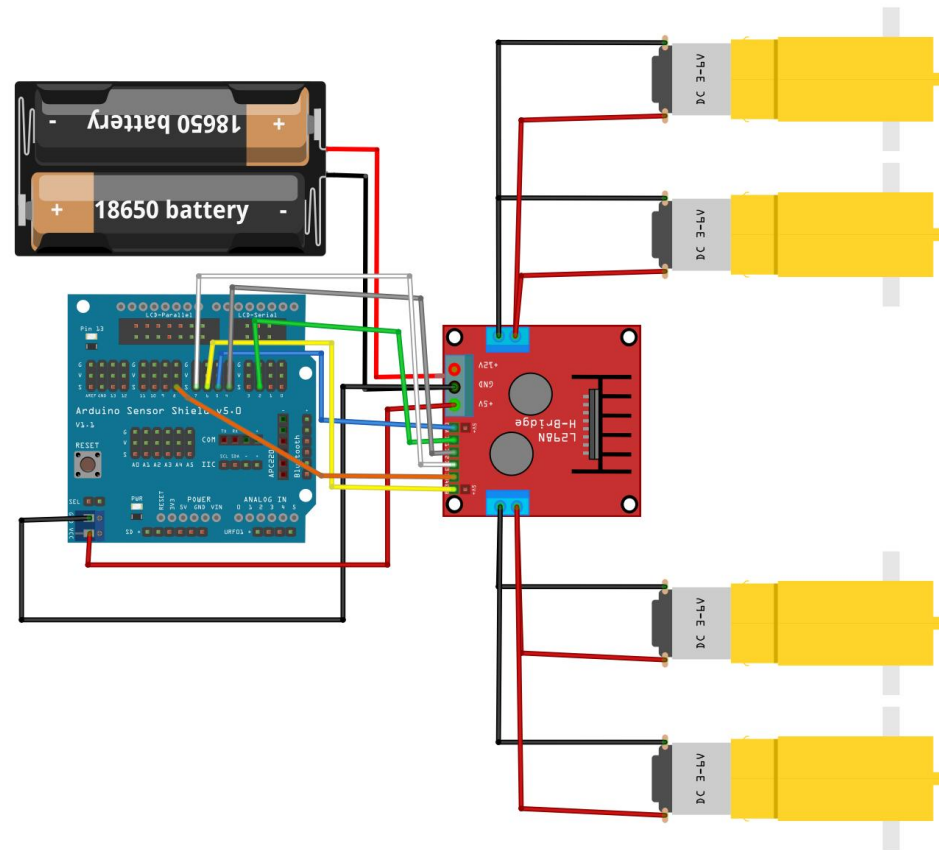
Using L298N made by ST Company as the control chip, the module has characteristics of strong driving ability, low calorific value and strong anti-interference ability.

This module can use built-in 78M05 for electric work via a driving power supply part. But to avoid the damage of the voltage stabilizing chip, please use an external 5V logic supply when using more than 12V driving voltage.

Using large capacity filter capacitor, this module can follow current to protect diodes, and improve reliability.

# LROBRUYA

## Wiring diagram





---

### **Attention**

When connecting, the jumper caps of ENA and ENB must be removed.

After wiring, please open the program in the code folder- Lesson\_3\_L298N\_Motor\_Driver and click UPLOAD to upload the program.

### **Result**

After connection and power-on, two motors rotate clockwise for 2 second at a speed of 200 (PWM value is 200) and then stop for 2 second; two motors rotate anticlockwise for 2 second at a speed of 200 (PWM value is 200) and then stop for 2second; circulating like this.



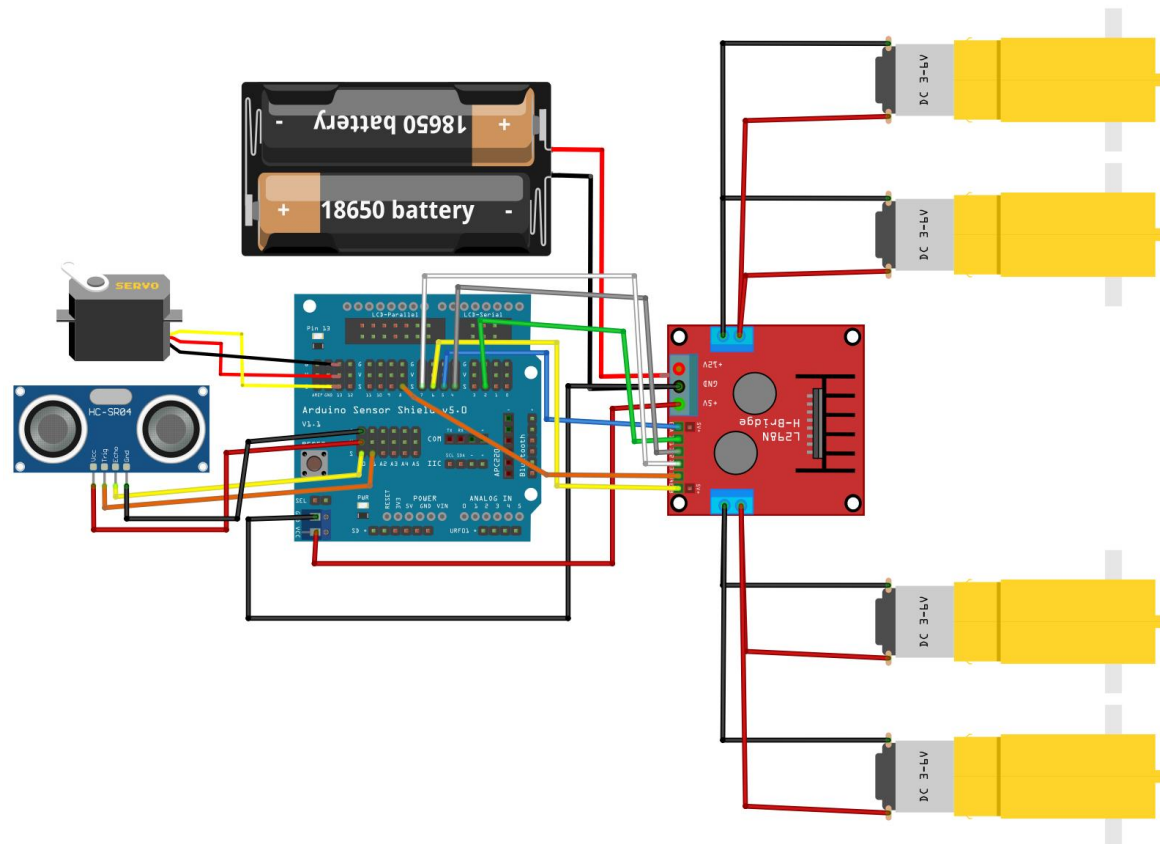
---

## Lesson 4 Ultrasound Obstacle Avoidance Car

### **About this lesson:**

This lesson ,regarding Arduino as main control, detect front obstacle by ultrasonic sensor and platform motor, and send the feedback to Arduino. Arduino will analyses the feedback signal and then control the driver motor to adjust the car diversion. Finally the car is able to avoid obstacle automatically and keep going.

## Wiring diagram





---

## Attention

After connecting, please open the the program and load up the code - Lesson\_4\_Ultrasound\_Obstacle\_Avoidance\_Car onto your Arduino board. Before you can run this, make sure that you have installed the < Servo> library or re-install it, if necessary. Otherwise, your code won't work.

## Result

1. Ultrasonic detecting distance: one port emits high level more than 10 us. Once it outputting level, open potentiometer to time. When the port becomes low level, read out current value. Use the time of detecting distance to calculate distance.
2. Use ultrasonic to detect the distance between obstacle and car, so that control the motion of the car according to the data.
3. When there is no obstacle in front of the robot car, the car keeps walking straight. When the distance of the obstacle in front of the robot car is less than 20cm, the robot car stops, then detects whether there is an obstacle in the left front and right front, and then turns in the opposite direction.



---

## Lesson 5 Follow Car

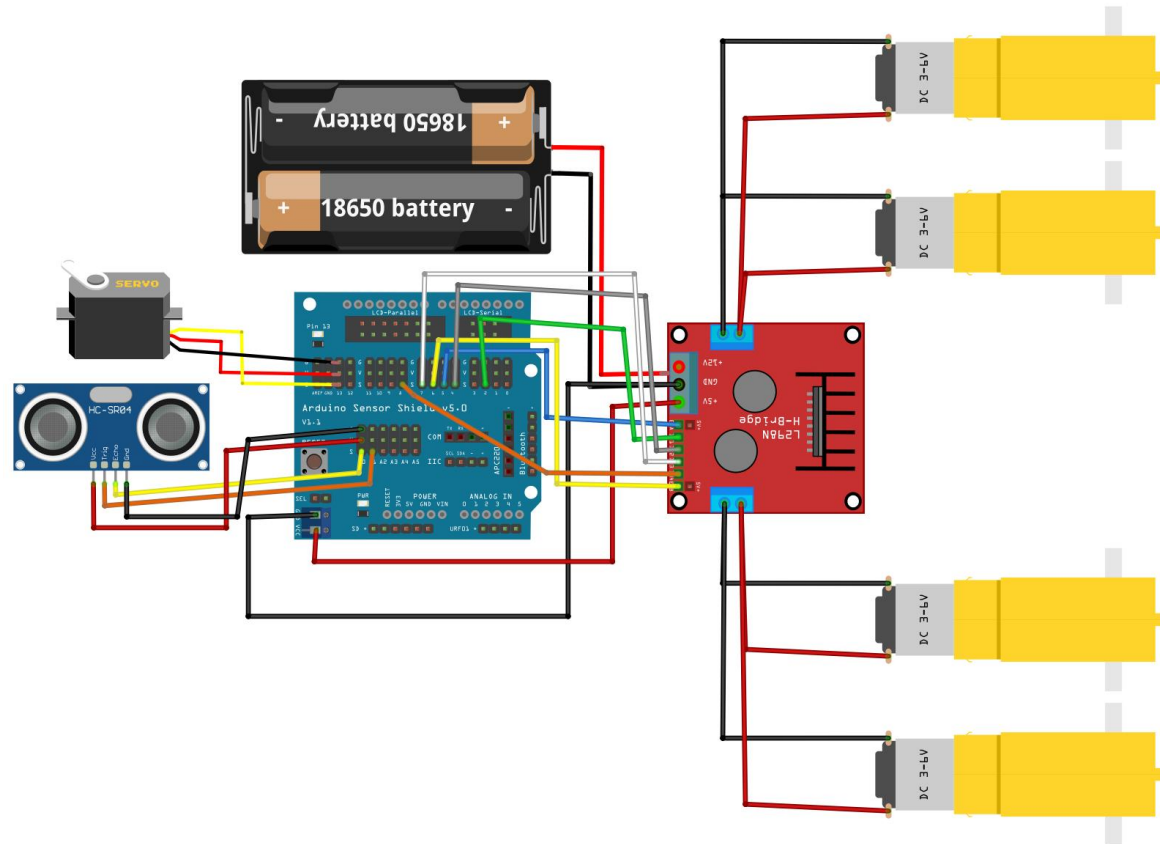
### **About this lesson:**

In this lesson, we learn the use of ultrasonic sensors and then use ultrasonic sensors to detect distances to achieve robots to follow his master.



# LROBRUYA

## Wiring diagram





---

## **Attention**

After connecting, please open the the program and load up the code - Lesson\_6\_Follow\_Car onto your Arduino board.

Before you can run this, make sure that you have installed the < Servo> library or re-install it, if necessary. Otherwise, your code won't work.

## **Result**

After downloading the program and observing the distance information printed by the serial port window, unplug the USB data cable and turn on the power switch of the robot. you can move your hand close to the front of the ultrasonic sensor.

When the robot is less than 10 cm away from your hand, he will stop. Move your hand straight forward. When the robot is more than 20cm away from your hand, the robot will approach your hand.



---

## Lesson 6 Line-Tracking sensor

### About this lesson:

In this lesson, you will learn how to use a Tracking Sensor. we will use an obstacle avoidance sensor module and an LED attached to pin of the Arduino Uno board to build a simple circuit to make a tracking light.





---

## Component Introduction

This Line Tracking Sensor can detect white lines in black and black lines in white. The single line-tracking signal provides a stable output signal TTL for a more accurate and more stable line. Multi-channel option can be easily achieved by installing required line-tracking robot sensors. The tracking sensor is actually an infrared sensor. The component used here is the TCRT5000 infrared tube. Its working principle is to use the different reflectivity of infrared light to the color, then convert the strength of the reflected signal into a current signal. During the process of detection, black is active at HIGH level, but white is active at LOW level. **The detection height is 0-3 cm. By rotating the adjustable potentiometer on the sensor, it can adjust the detection sensitivity of the sensor.**

### Specification:

Power Supply: +5V Operating Current: <10mA

Operating Temperature Range: 0°C ~ + 50 °C

Output Interface: 3-wire interface (1 - signal, 2 - power, 3 - power supply negative) Output Level: TTL level



---

## **How to use the Line-Tracking sensor**

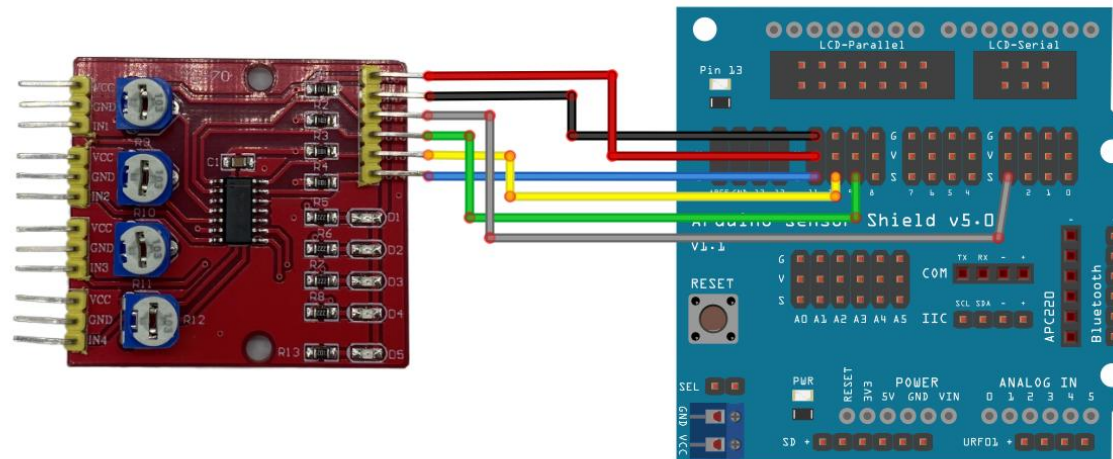
we read the signal level of Line-Tracking sensor to judge whether detect black or not.

When detects black, sensor's signal pin outputs HIGH (display 1).The red indicator light will turn off ; otherwise, output LOW (display 0),the red indicator light will turn on.

Show the result on the serial monitor.

# LROBROYA

## Wiring diagram



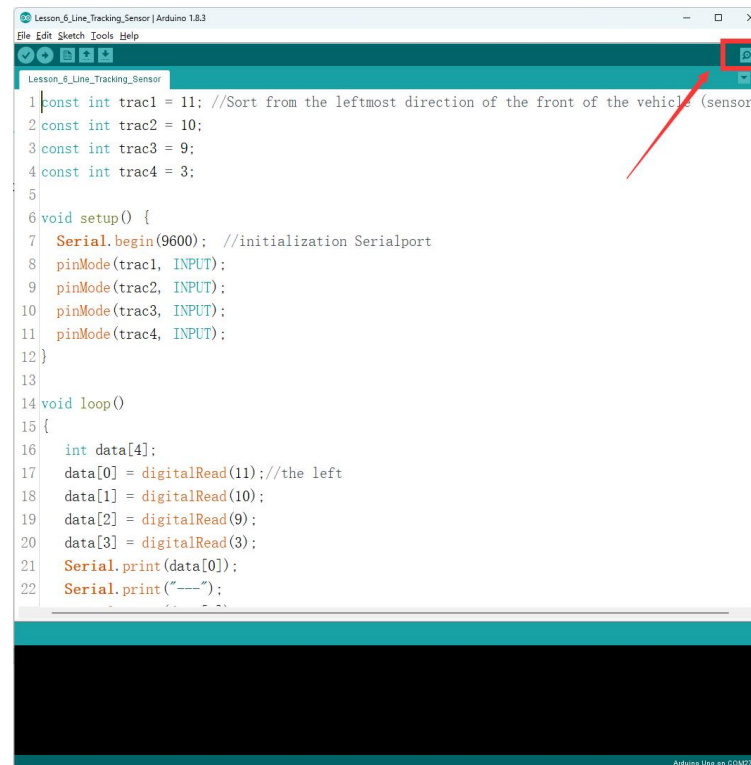
## Attention

The Line Tracking sensor adjusts the detection sensitivity of the sensor according to the actual situation by rotating the adjustable potentiometer on the sensor.

# LROBROYA

## Result

After uploading the code, click the button in the upper right corner to open the serial monitor to view the value.



```
Lesson_6_Line_Tracking_Sensor | Arduino 1.8.3
File Edit Sketch Tools Help

Lesson_6_Line_Tracking_Sensor
1 const int trac1 = 11; //Sort from the leftmost direction of the front of the vehicle (sensor
2 const int trac2 = 10;
3 const int trac3 = 9;
4 const int trac4 = 3;
5
6 void setup() {
7   Serial.begin(9600); //initialization Serialport
8   pinMode(trac1, INPUT);
9   pinMode(trac2, INPUT);
10  pinMode(trac3, INPUT);
11  pinMode(trac4, INPUT);
12 }
13
14 void loop()
15 {
16   int data[4];
17   data[0] = digitalRead(11); //the left
18   data[1] = digitalRead(10);
19   data[2] = digitalRead(9);
20   data[3] = digitalRead(3);
21   Serial.print(data[0]);
22   Serial.print("----");
23 }
```



---

When the line patrol sensor detects black, the serial monitor will receive "1", and the digital port will receive a high level, otherwise the serial monitor will receive "0", and the digital port will receive a low level. If the four sensors on the left and right and middle detect black at the same time, the serial monitor receives four "1"s. If the left sensor detects black and the other three sensors detect non-black, the signals received by the serial monitor are "1---0---0---0".

## Lesson 7 Line-Tracking Car

### About this lesson:

In this lesson, we will learn a simple and automatic line tracking system of a car.

**Step 1:** Prepare a black track on white ground. (the width of the black track is more than 20mm and less than 30mm).

Please note, the bend angle of the track can't be larger than 90 degree. If the angle is too large, the car will move out of the track.





---

**Step 2:** Adjust the sensitivity of tracking sensor modules.

Turn on and hold the car to adjust the potentiometer on the tracking sensor with Phillips screwdriver until you get the best sensitivity status: the signal indicate LED light will turn on when sensor is above white ground, and the signal LED will turn off when the sensor is above black track.

Signal Indicate LED ON: White Ground

Signal Indicate LED OFF: Black Track

**Step 3:** Turn on the car and put the car over the black track, then the car will move along the black track.

If the car can't move, please check the following:

If adjusted well the sensibility of the tracking sensor

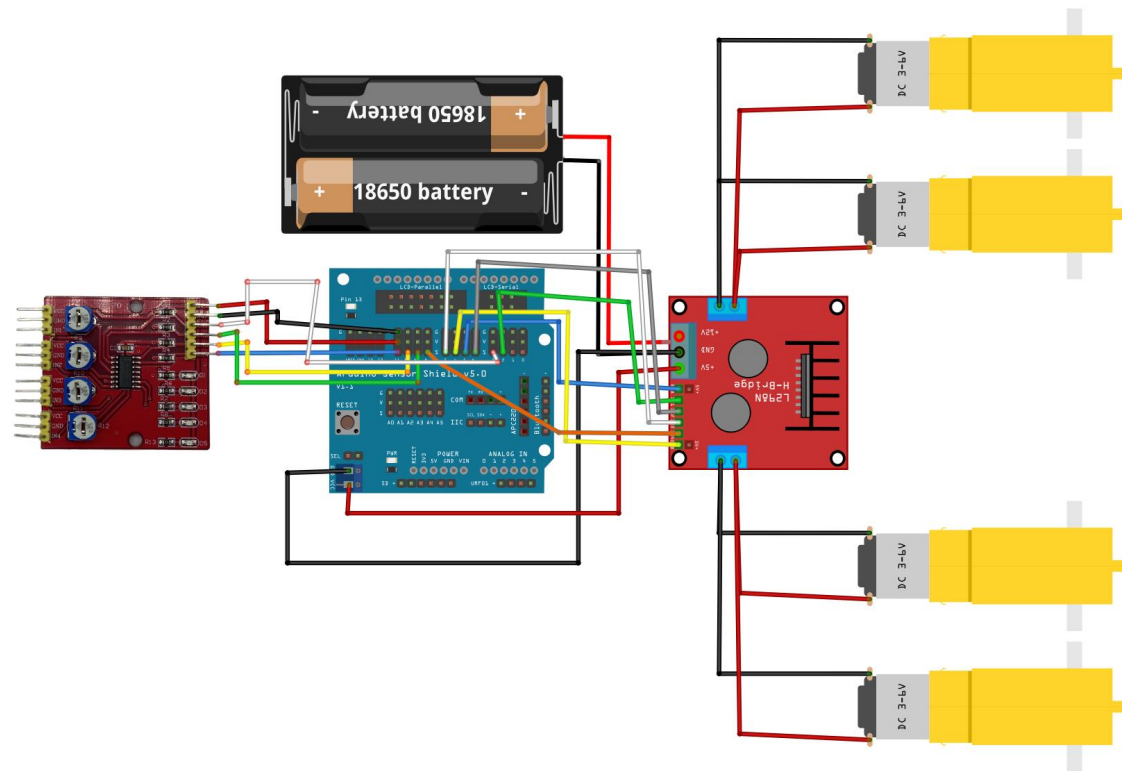
### **Car tracking flow chart**

The car entered the tracking mode, namely began constantly scanning and detector connected to the I/O port of the SCM, once detected a signal of a I/O port, enter judgment processing procedures, to determine which one of 3 detectors detect the

# LROBRUYA

black line.

## Wiring diagram





## **Result**

Turn the POWER switch ON. The robot car will move forward along the black track.