PREFACE

This manual is based on the arduino uno suite system and a tutorial to operate each module, the aim is to make user can quickly the grasp Arduino and familiar with the principle of each module and how to use it. This document does not give a specific application sample, we hope the guest to learn and be familiar with each module in the tutorial, and flexibly apply to your creative products. Of course, emakefun also use arduino modules to design particularly creative products, these will target different products and provide corresponding suite, detailed tutorials.

Introduction of Arduino UNO

What is arduino ?

Arudino is originally created by Italian teacher Massimo Banzi for the convenience of the electronic major students presenting their ideas through hardware. About in the winter of 2005, he united Spain chip engineer David Cuartielles, discussed the idea, so they two decided to design their own circuit boards, and recruited Banzi student David Mellis to write programming language for circuit board. It was named after the name of the Italian king Arduin. He named the Arduino and start to create Arduino. Subsequently, Banzi, Cuartielles and Mellis put the design online. Due to everyone only knew the open-source software, there was no heard of source hardware, and then they thought of the well-known Linux open-source software, so they hope the Arduino to open source like Linux.

The emergence of the Arduino required original microcontroller programmer to possess relevant electronic or software background, become normal amateurs and soon learned how to use the Arduino. His design concept attracted many professional or non-professional people right after being introduced worldwide.

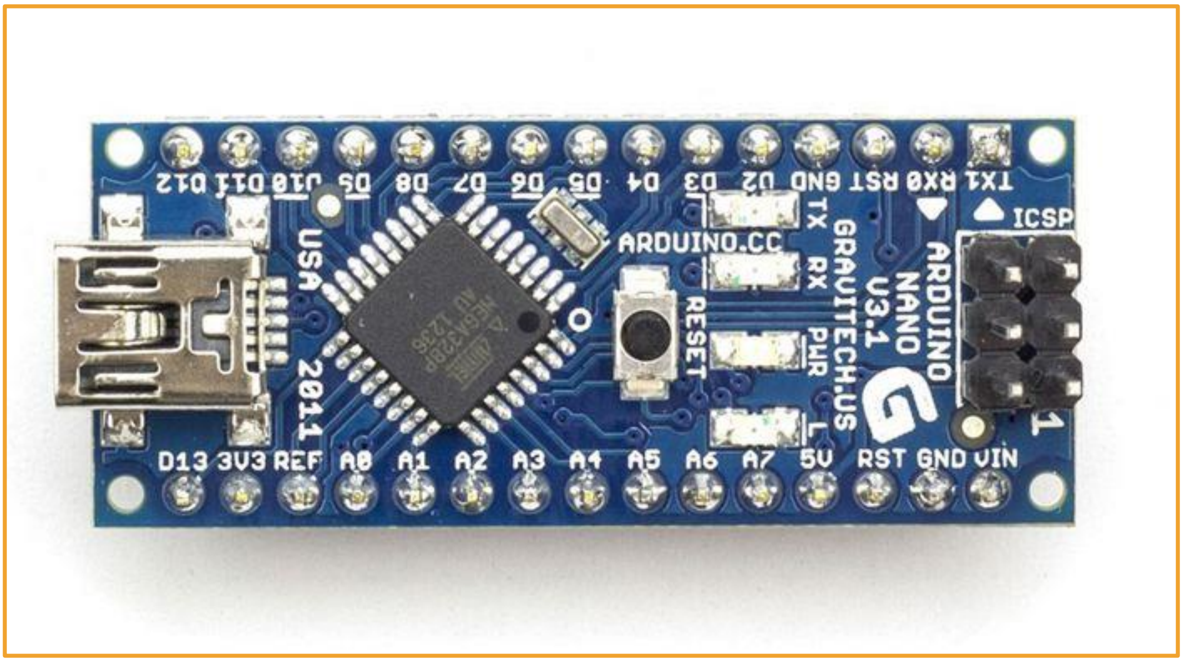
What is arduino uno Kit ?

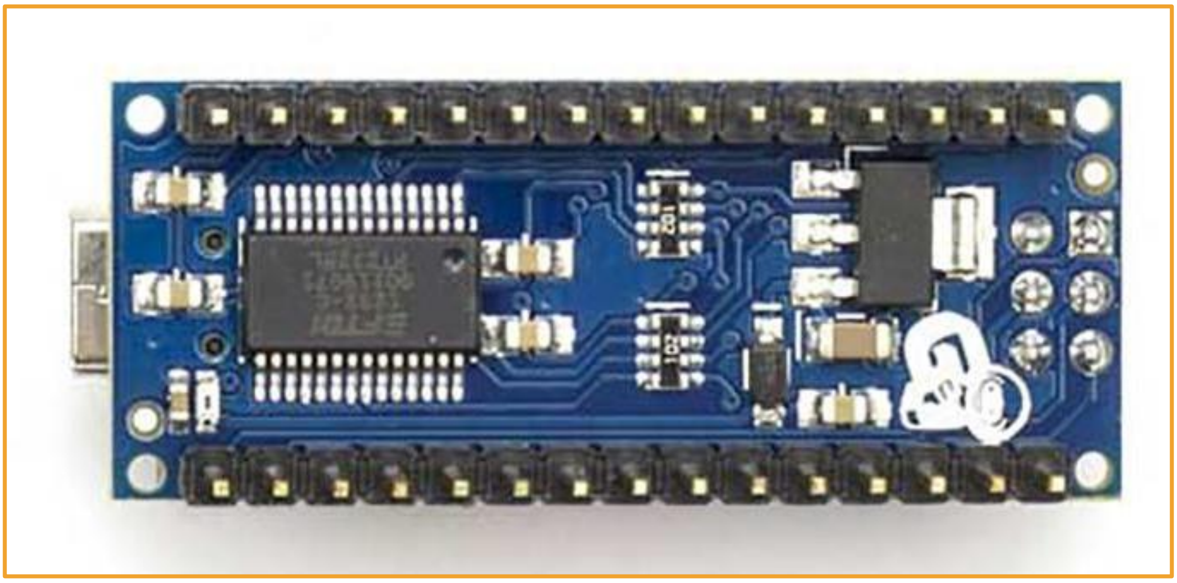
Arduino uno is arduino USB series reference standard template. UNO processor core is Atmel company produces Atmega328, at the same time, with 14 digital input/output port (of which six can be used as a pwm output), 6 analog input, a 16 MHz crystal oscillator, a USB port, and a power socket, a ICSP header and a reset button. UNO has been released to the third edition.

- The processor ATmega328
- Working voltage 5v
- Input voltage (recommended) 7-12v

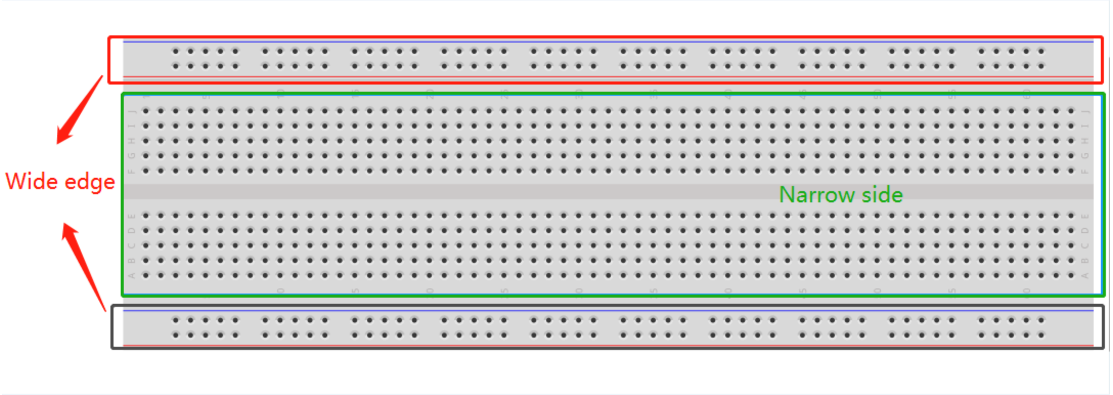
- Input voltage (range)   6-20v
- Digital IO pin 14 (of which six as a PWM output)
- Analog input pin 6
- IO pin DC 40 mA
- 3.3V pin DC 50 mA
- Flash Memory 32 KB (ATmega328, in which 0.5 KB for bootloader)
- SRAM 2 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- Clock 16 MHz

Picture of real products are as follows:


D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 GND RST RX0 TX1
ICSP
ARDUINO.CC
USA
2011
RESET
TX
RX
PWR
L
GRAVITECH.US
ARDUINO
NANO
V3.1
1
D13 3V3 REF A0 A1 A2 A3 A4 A5 A6 A7 5V RST GND VIN

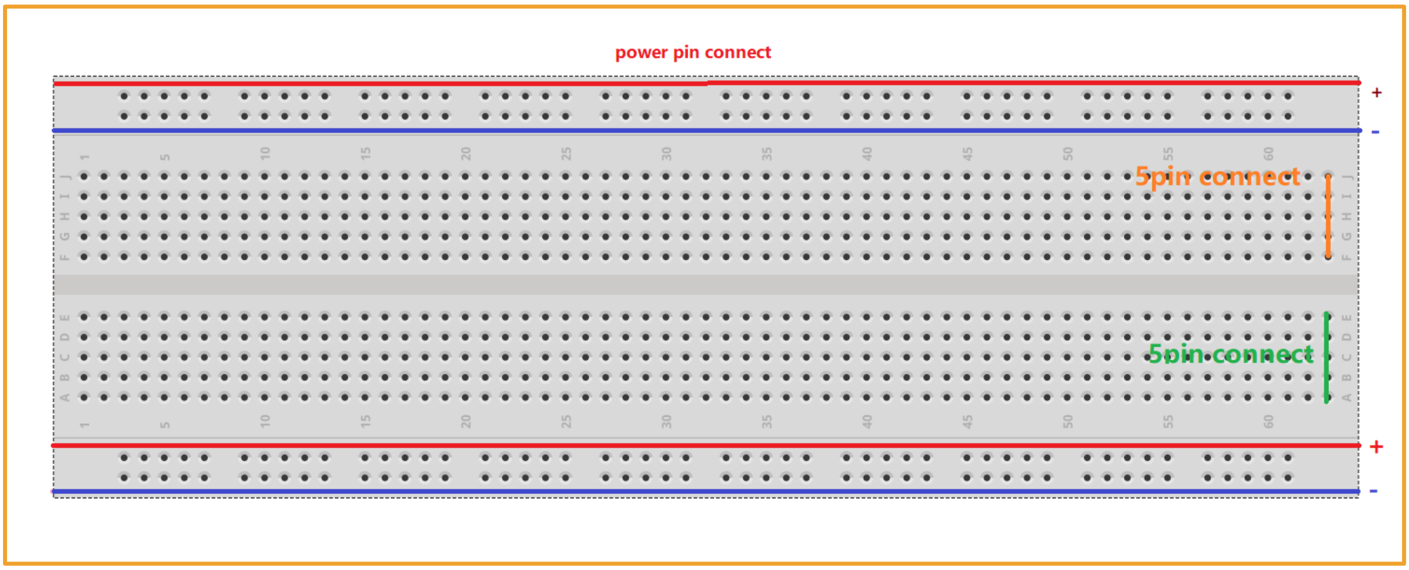


How To use breadboard



All power pin connect together as red line , but up and down is separate.

All GND pin connect together as blue line, but up and down is separate.

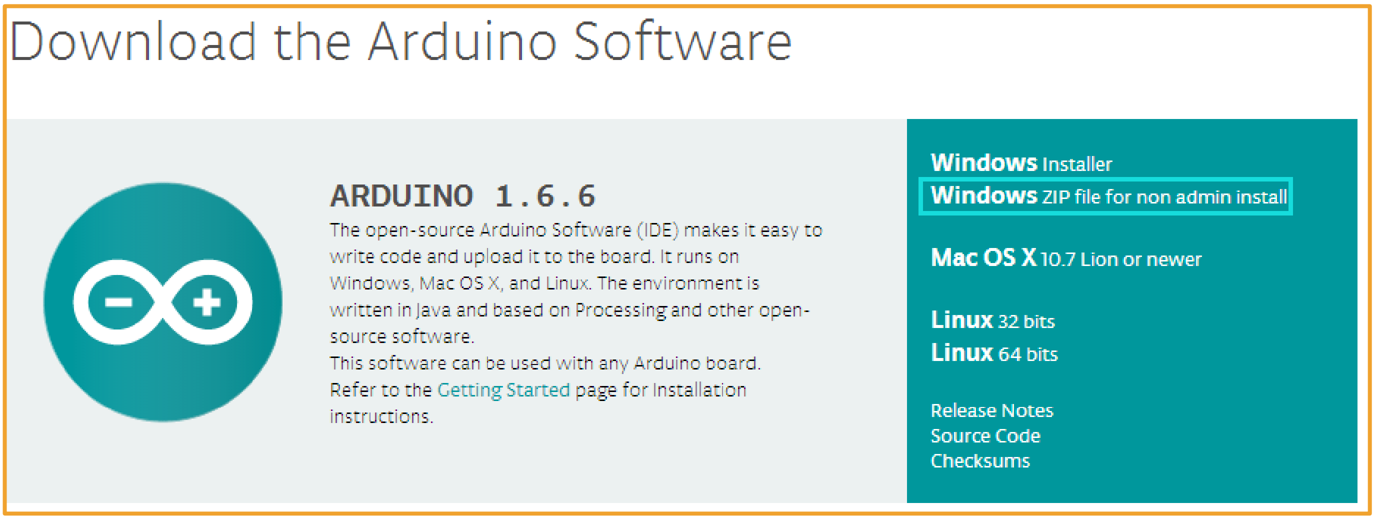


Development Environment Building

Development Software

Download link: https://www.arduino.cc/en/Main/Software

Windows, Linux, Mac are all available for downloading.



Download the Arduino Software

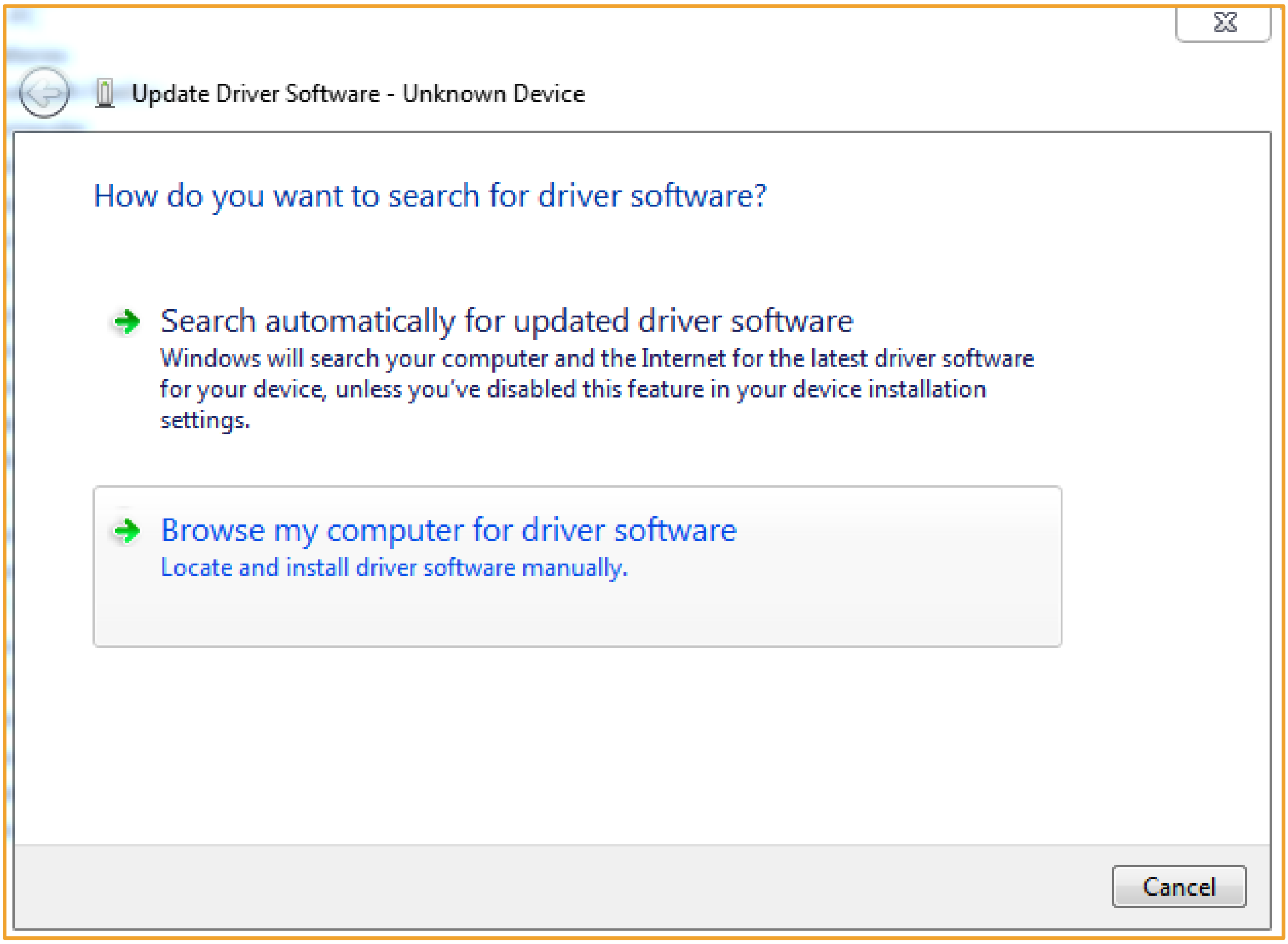
ARDUINO 1.6.6

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board.
Refer to the Getting Started page for Installation instructions.

The interface of Arduino IDE is simple and the operation is rather convenient. If you want get more please click https://www.arduino.cc/en/Guide/Environment

Install usb to serial port driver

Inserting USB cable will prompt as follows, choosing the specified location to install.



Update Driver Software - Unknown Device

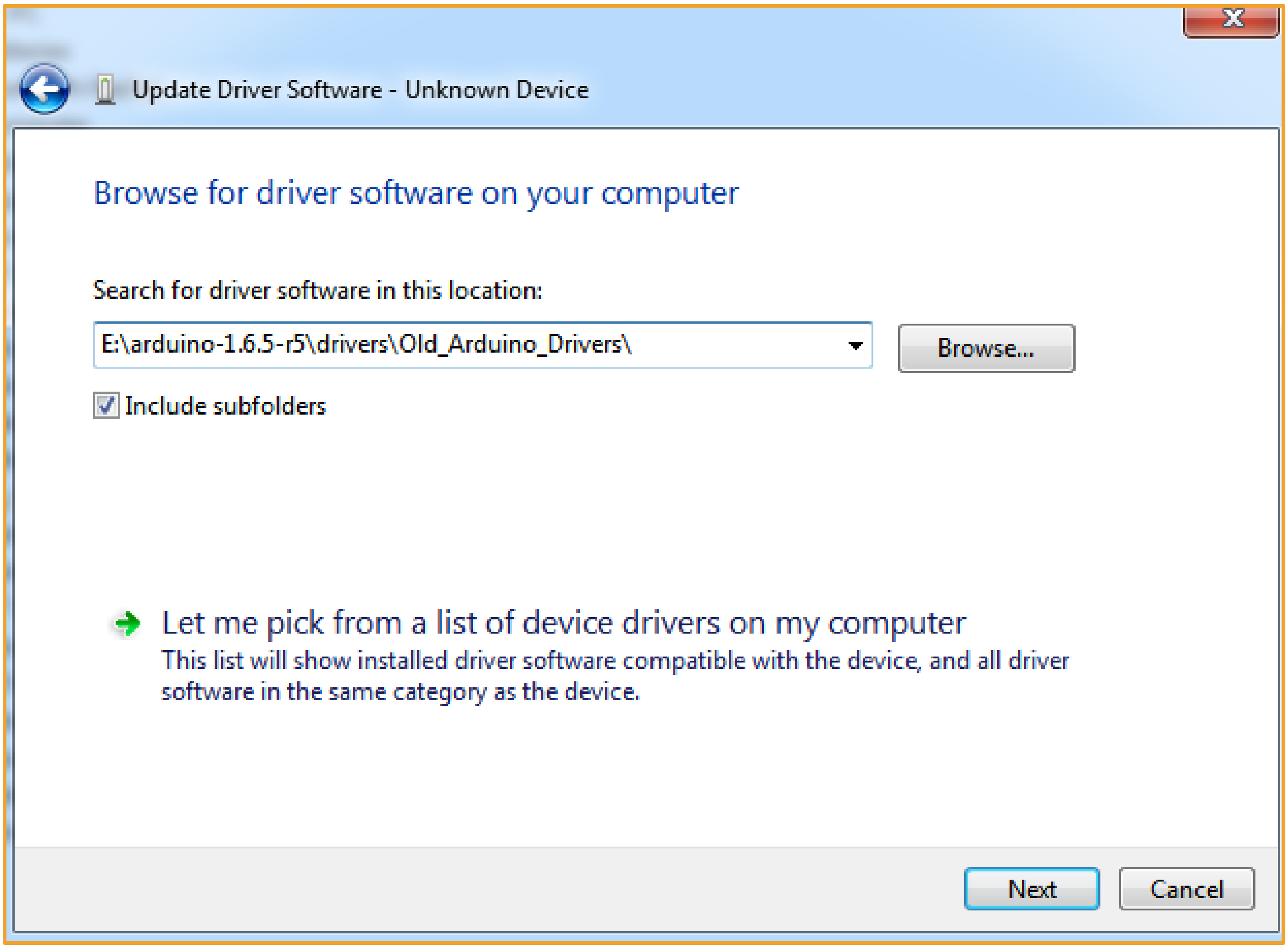
How do you want to search for driver software?

Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.

Selecting download Arduino ide file “E:\arduino-1.6.5-r5\drivers\Old_Arduino_Drivers\” Checking the type of USB serial chip on the board, if it is Atmel, then choose the following path; if it is FTDI, you should choose the arduino\drivers\FTDI USB Drivers path.

This kit use CH340 driver so install《CH341SER_for_64bit_win7.zip》



Update Driver Software - Unknown Device

Browse for driver software on your computer

Search for driver software in this location:

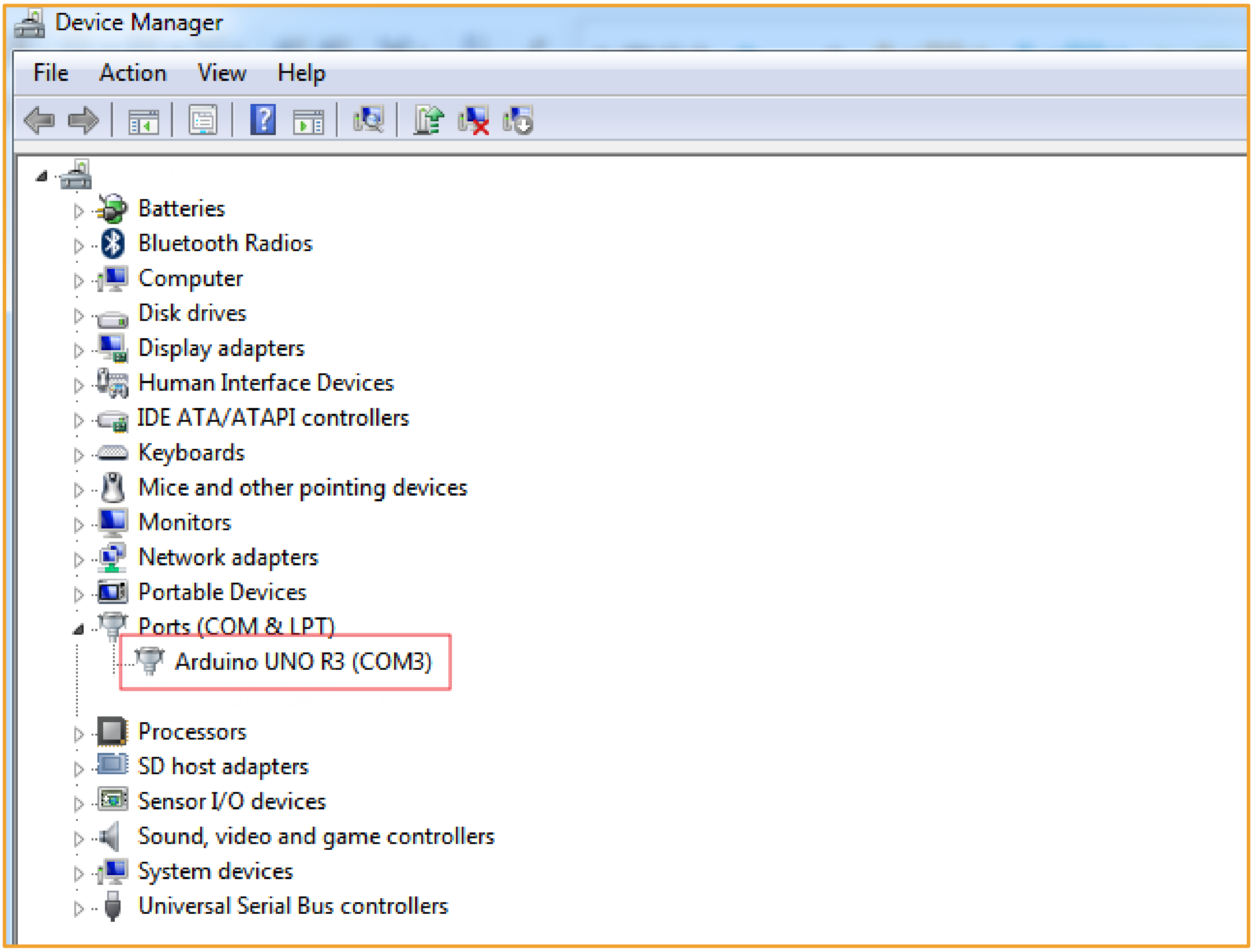
E:\arduino-1.6.5-r5\drivers\Old_Arduino_Drivers\

☑ Include subfolders

Let me pick from a list of device drivers on my computer

This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

Clicking on the next step, you will be prompted with a successful installation message. Now you can change to equipment management to see Arduino UNO R3



Linux Installation Environment Building

Download the Arduino Software select Linux 64bit and save.

```
Shell:# tar –vxf arduino-1.6.7-linux64.tar.xz
Shell:# cd arduino-1.67
Shell:# ./arduino
```

Mac Arduino Development Environment Building

You can get all source codes from https://github.com/emakefun/emakefun-uno-kit

Install CH340 driver ： CH34XUSB-SERIAL_for_mac\CH34x_Install_V1.3.pkg

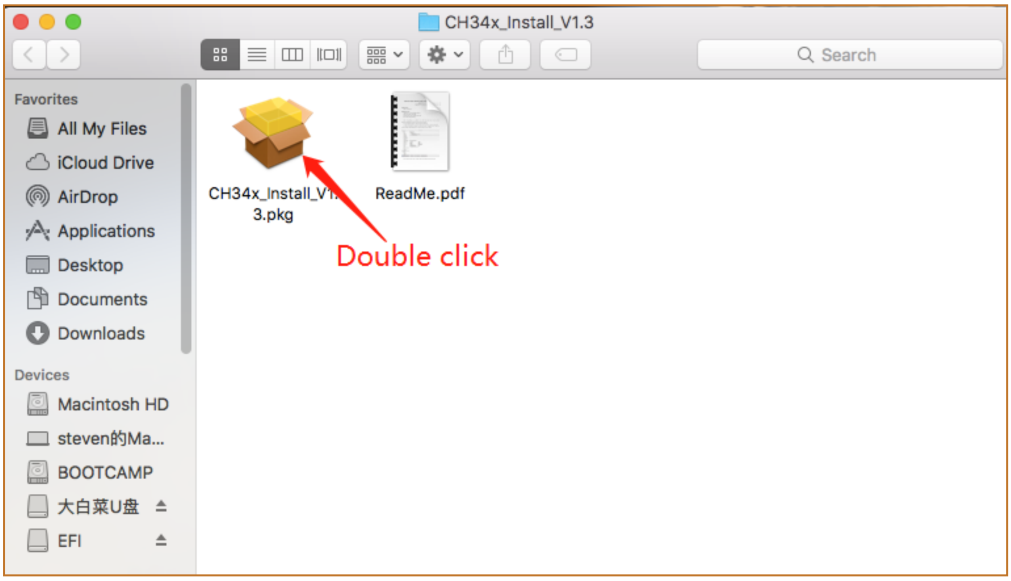
SupportSystem:

OSX10.9andabove

InstallationProcess:

Extract the contents of the zip file to a local installation

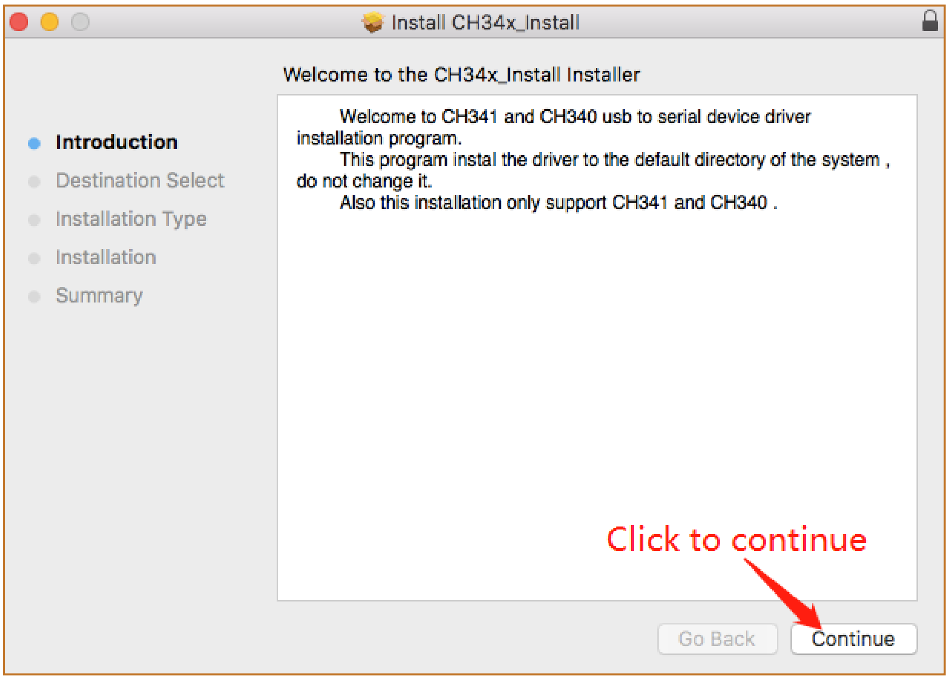
directory Double-clickCH34x_Install.pkg



Double click

Install according to the installation on

procedure



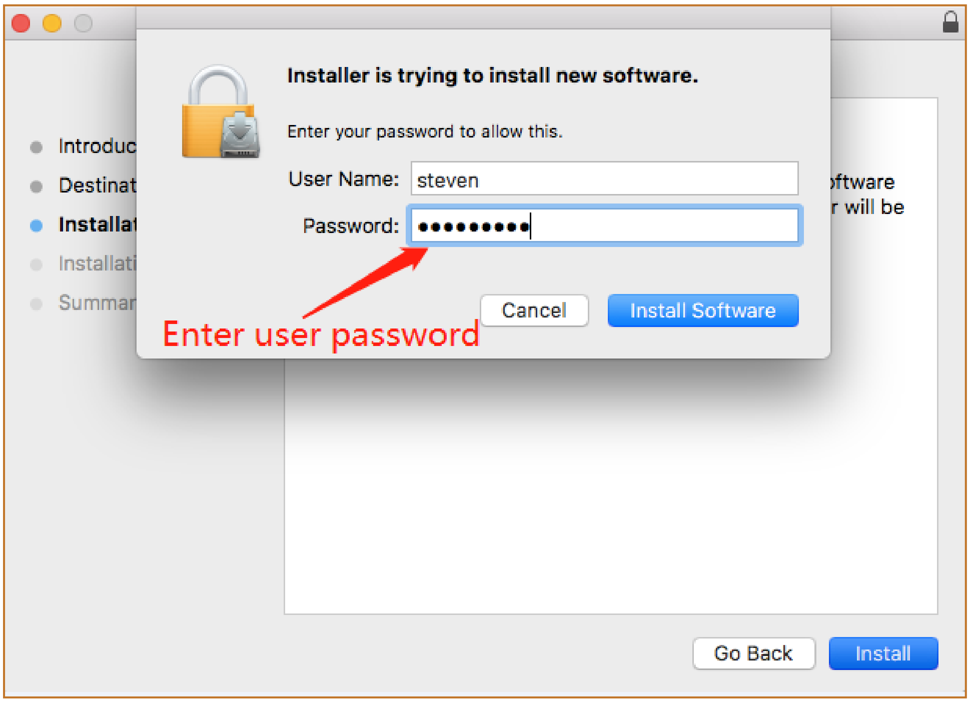
Welcome to the CH34x_Install Installer

Welcome to CH341 and CH340 usb to serial device driver installation program.

This program instal the driver to the default directory of the system , do not change it.

Also this installation only support CH341 and CH340 .

Click to continue



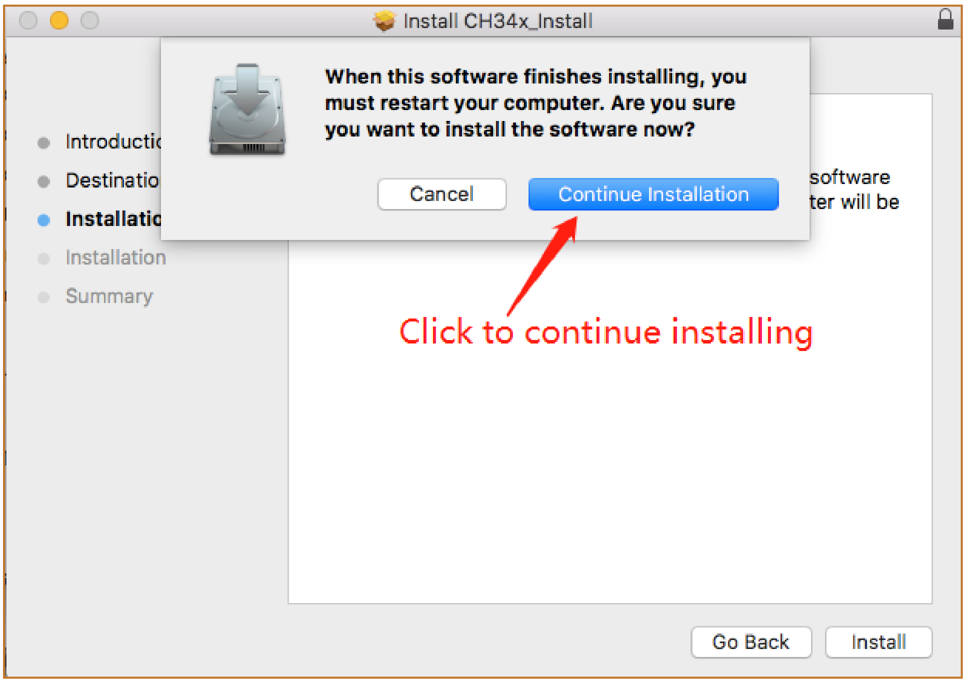
Installer is trying to install new software.

Enter your password to allow this.

User Name: steven

Password: •••••••••

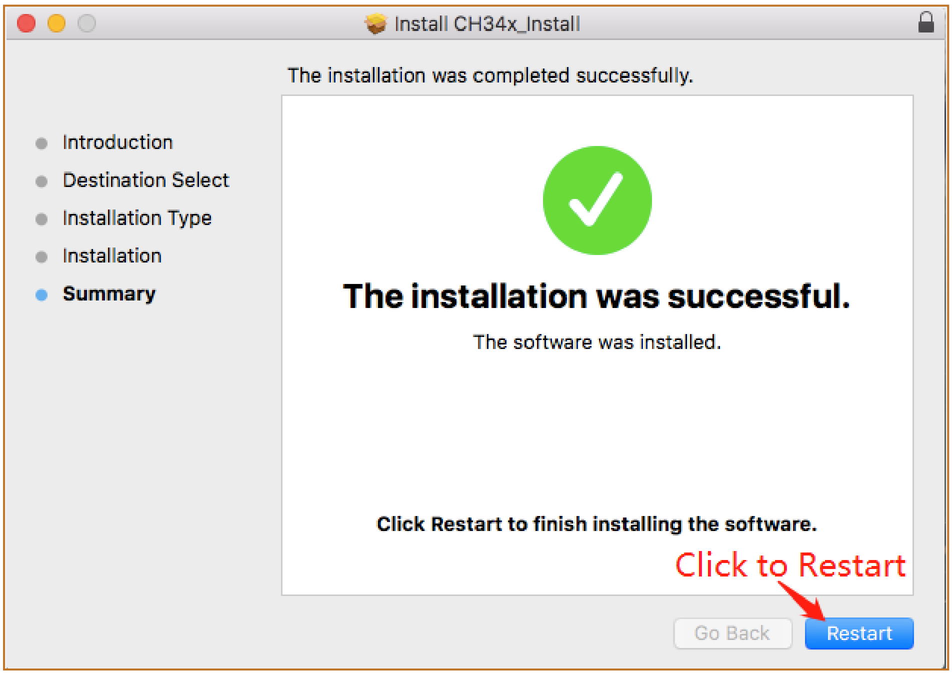
Enter user password



Install CH34x_Install

When this software finishes installing, you must restart your computer. Are you sure you want to install the software now?

Click to continue installing



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 finish installing the software.

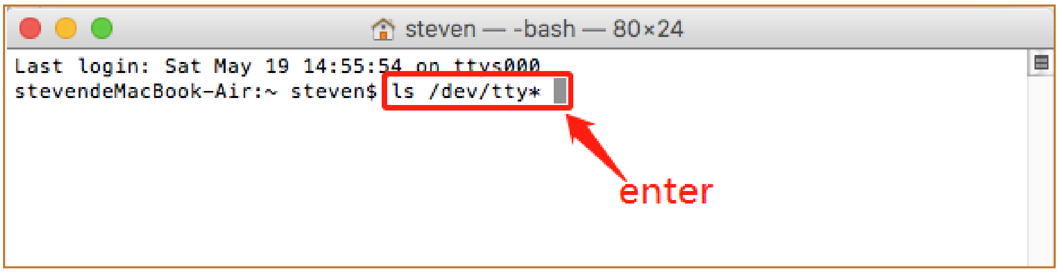
Click to Restart

Restart after finishing installing

After the installation is completed, you will find serial device in the device list(/dev/tty.wchusbserial*) ,and you can access it by serial tools.

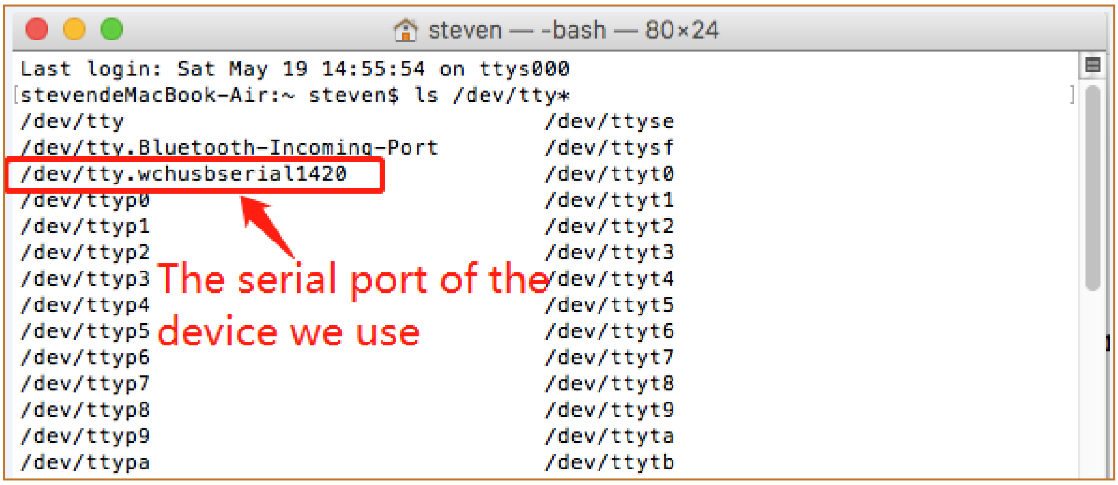
If you can’t find the serial port then you can follow the steps below:

Step1: Open terminal and type ‘ls/dev/tty*’ and see is there device like tty.wchusbserial;



```
Last login: Sat May 19 14:55:54 on ttys000
stevendeMacBook-Air:~ steven$ ls /dev/tty*
```

enter

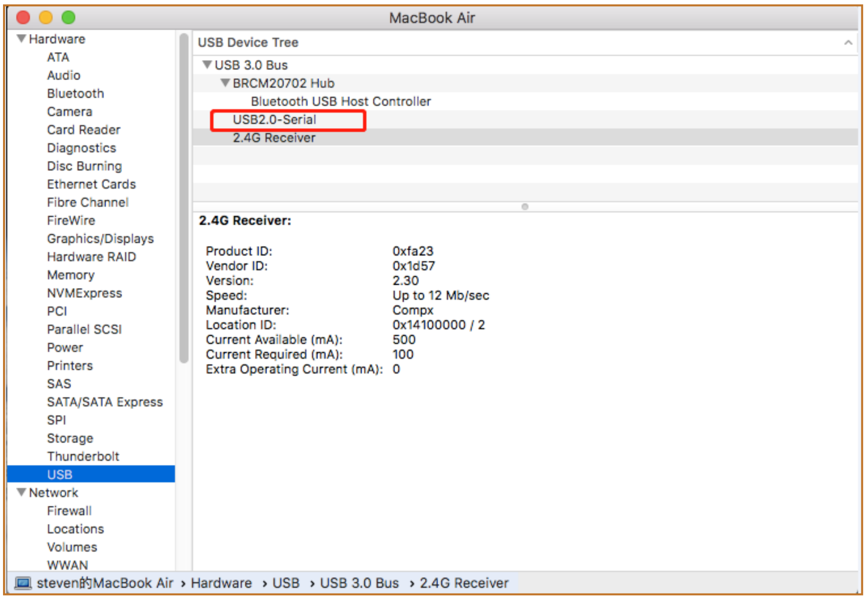


```
Last login: Sat May 19 14:55:54 on ttys000
stevendeMacBook-Air:~ steven$ ls /dev/tty*
/dev/tty                                    /dev/ttyse
/dev/tty.Bluetooth-Incoming-Port            /dev/ttysf
/dev/tty.wchusbserial1420                   /dev/ttyt0
/dev/ttyp0                                  /dev/ttyt1
/dev/ttyp1                                  /dev/ttyt2
/dev/ttyp2                                  /dev/ttyt3
/dev/ttyp3                                  /dev/ttyt4
/dev/ttyp4                                  /dev/ttyt5
/dev/ttyp5                                  /dev/ttyt6
/dev/ttyp6                                  /dev/ttyt7
/dev/ttyp7                                  /dev/ttyt8
/dev/ttyp8                                  /dev/ttyt9
/dev/ttyp9                                  /dev/ttyta
/dev/ttypa                                  /dev/ttytb
```

The serial port of the device we use

Step2: Open ‘System Report’->Hardware->USB, on the right side “USB Device Tree” there will Be device named “Vendor-Specific Device” and check if the Current is normal.

If the step supper don’t work at all, please try to install the package again.



2.4G Receiver:

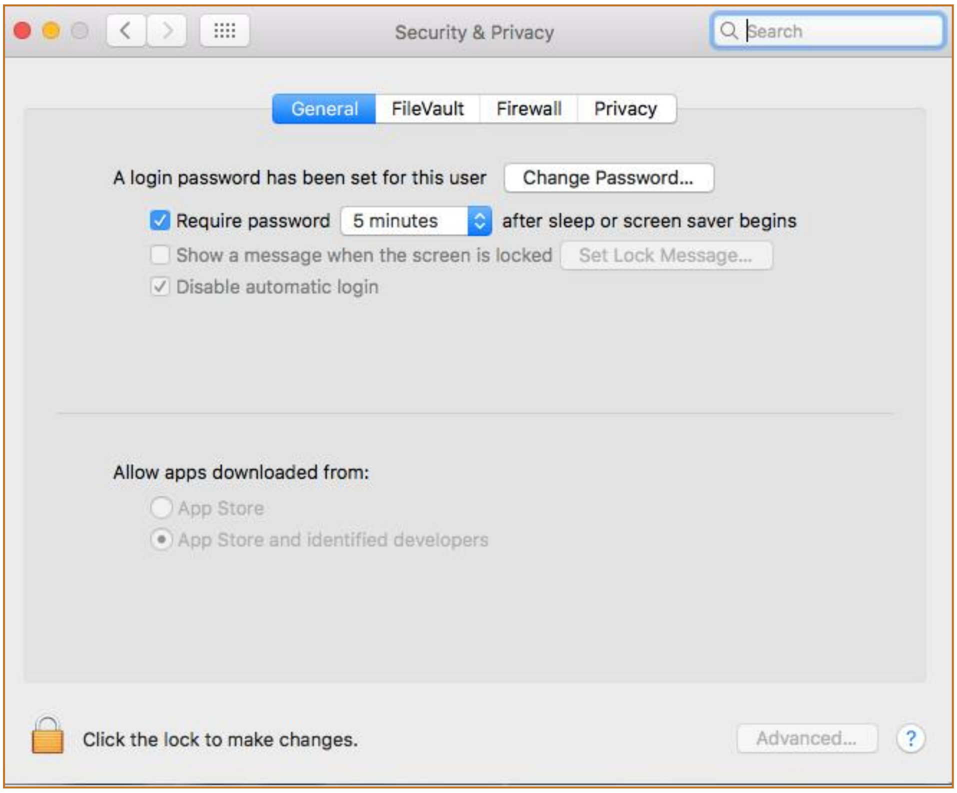
Product ID:	0xfa23
Vendor ID:	0x1d57
Version:	2.30
Speed:	Up to 12 Mb/sec
Manufacturer:	Compx
Location ID:	0x14100000 / 2
Current Available (mA):	500
Current Required (mA):	100
Extra Operating Current (mA):	0

Note:

Please enter “System Preferences”->”Security & Privacy”->”General”, below the

title” Allow apps downloaded from: ”you should choose the choice2->” Mac App Store

and identified developers ” so that our driver will work normally.



A login password has been set for this user

☑ Require password 5 minutes after sleep or screen saver begins

□ Show a message when the screen is locked

☑ Disable automatic login

Allow apps downloaded from:

○ App Store

● App Store and identified developers

Click the lock to make changes.

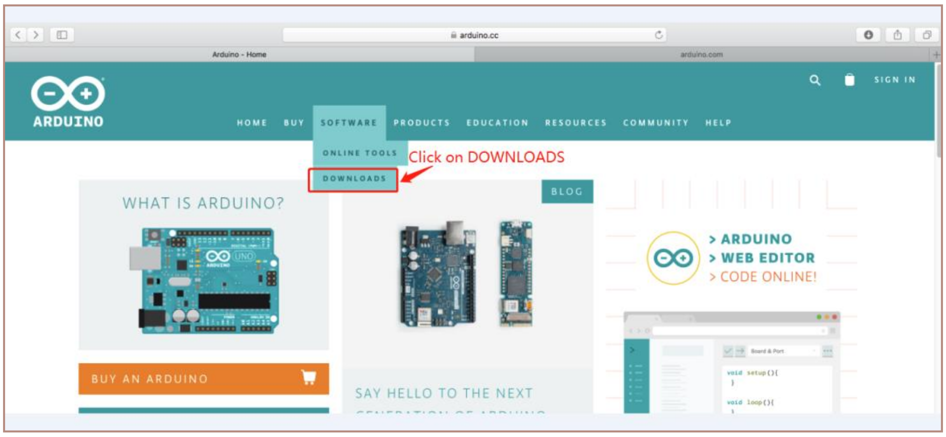
s

Mac Install Arduino IED

Download Arduino

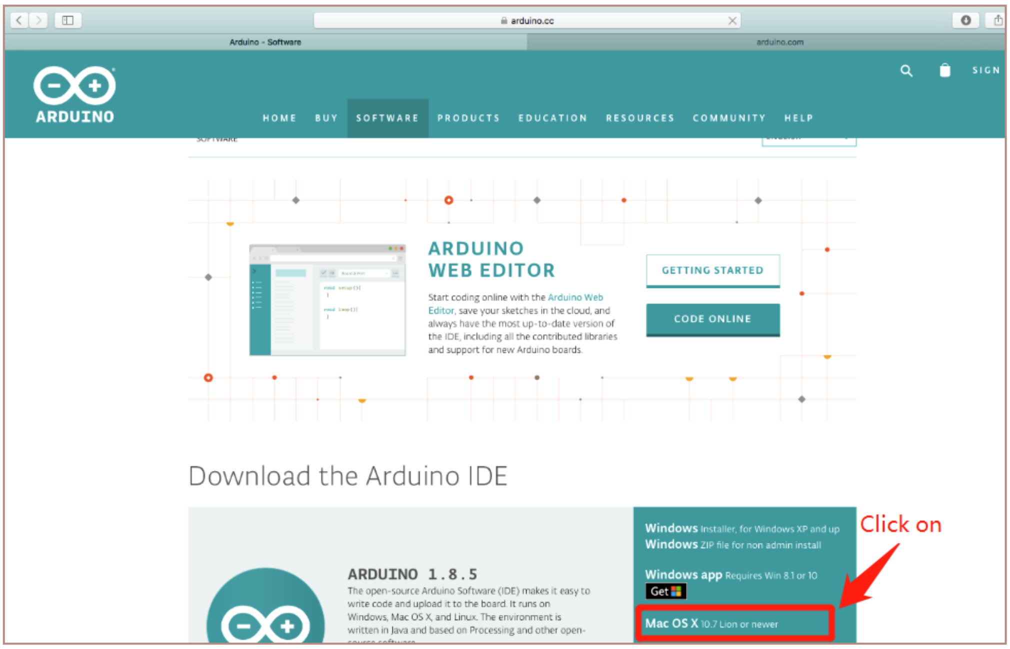
Step1 : Enter the URLarduino.cc

Step2: Enter the download page and click DOWNLOADS under SOFTWARE



Click on DOWNLOADS

Step3 :Select the system version software



Start coding online with the Arduino Web Editor, save your sketches in the cloud, and always have the most up-to-date version of the IDE, including all the contributed libraries and support for new Arduino boards.

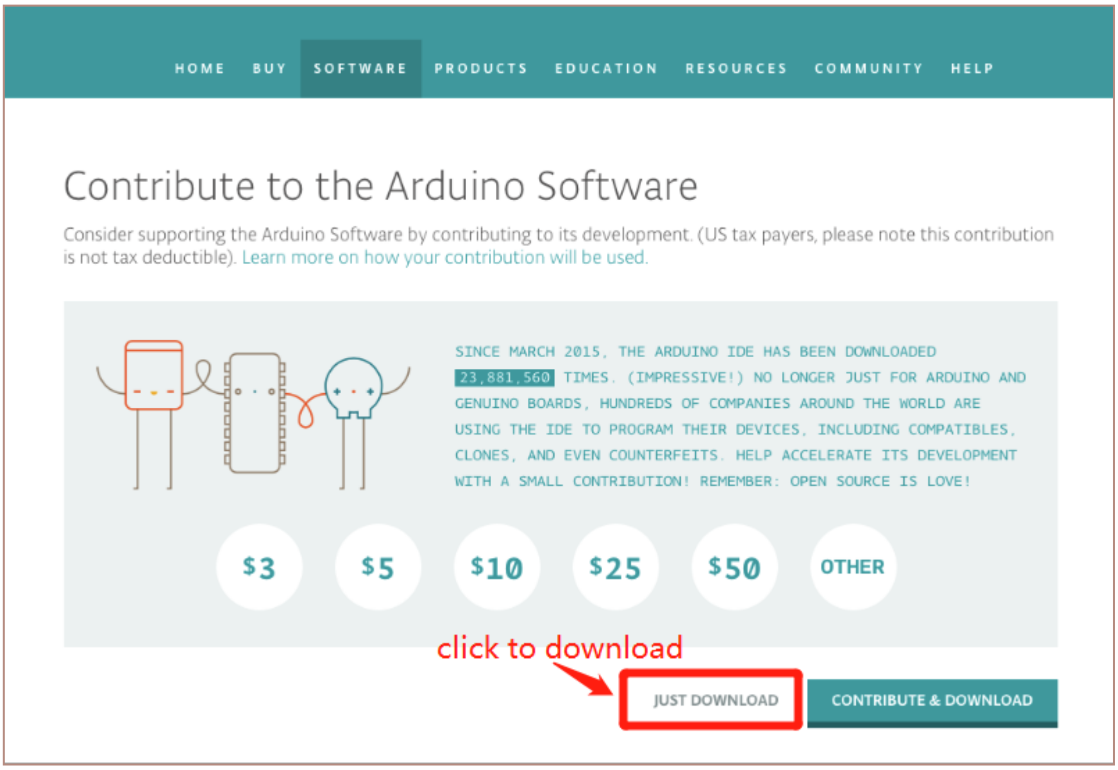
Download the Arduino IDE

ARDUINO 1.8.5

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-

Click on

Step4: Click to download



Contribute to the Arduino Software

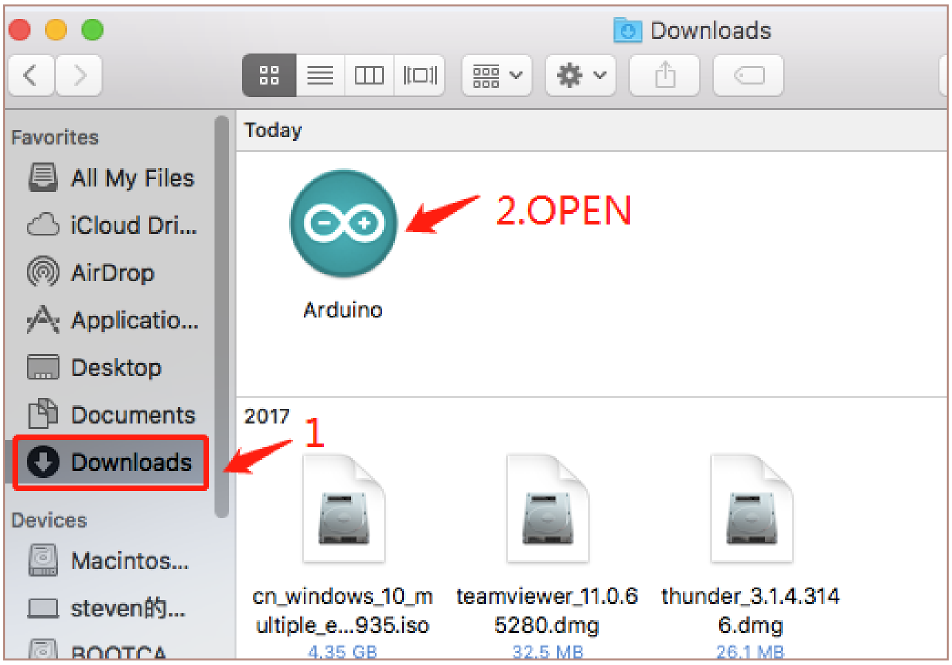
Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). Learn more on how your contribution will be used.

SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED 23,881,560 TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

click to download

Arduinon IDE Tutorial

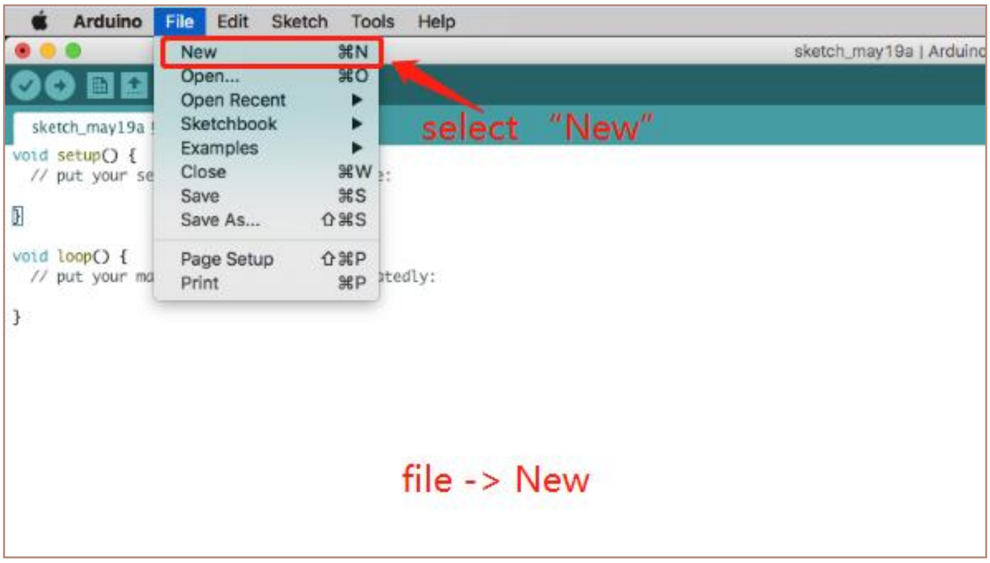
Step1:Open Arduino



2.OPEN

1

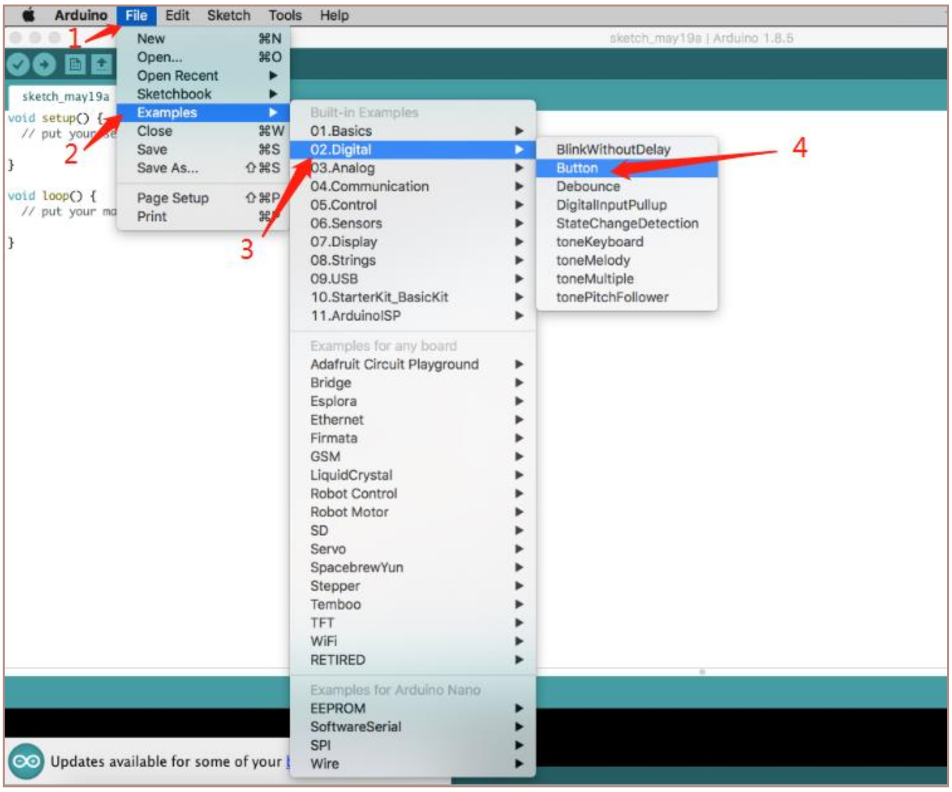
Step2: New construction



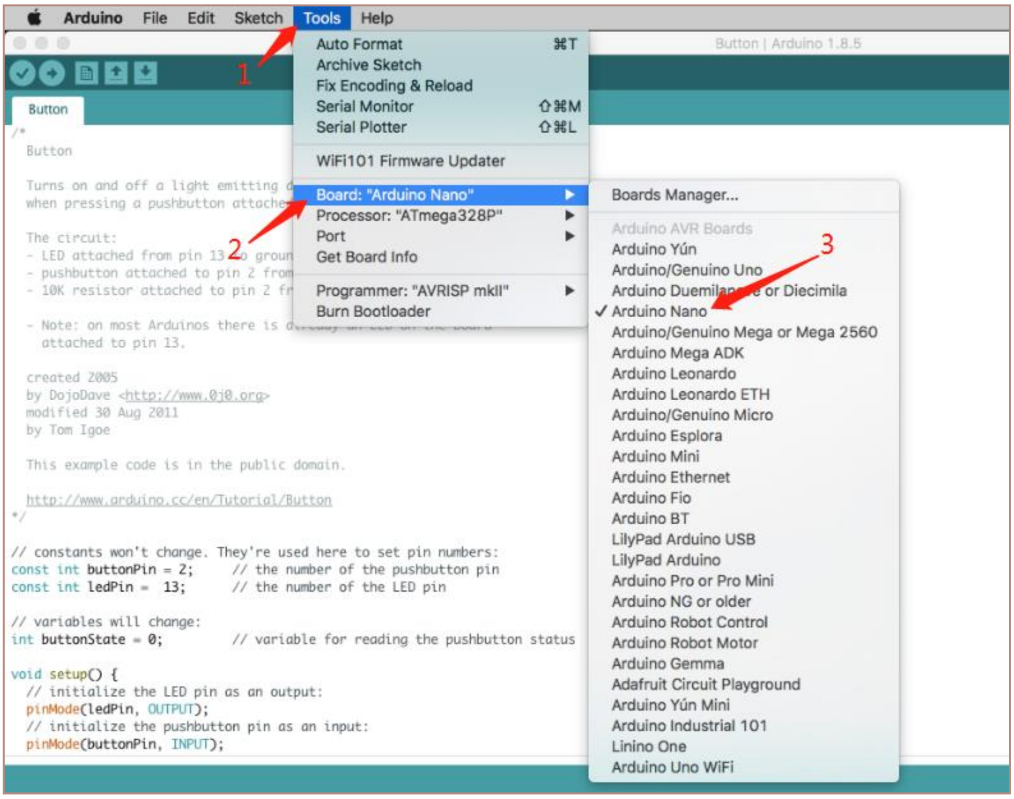
select "New"

file -> New

Step3:Open an example

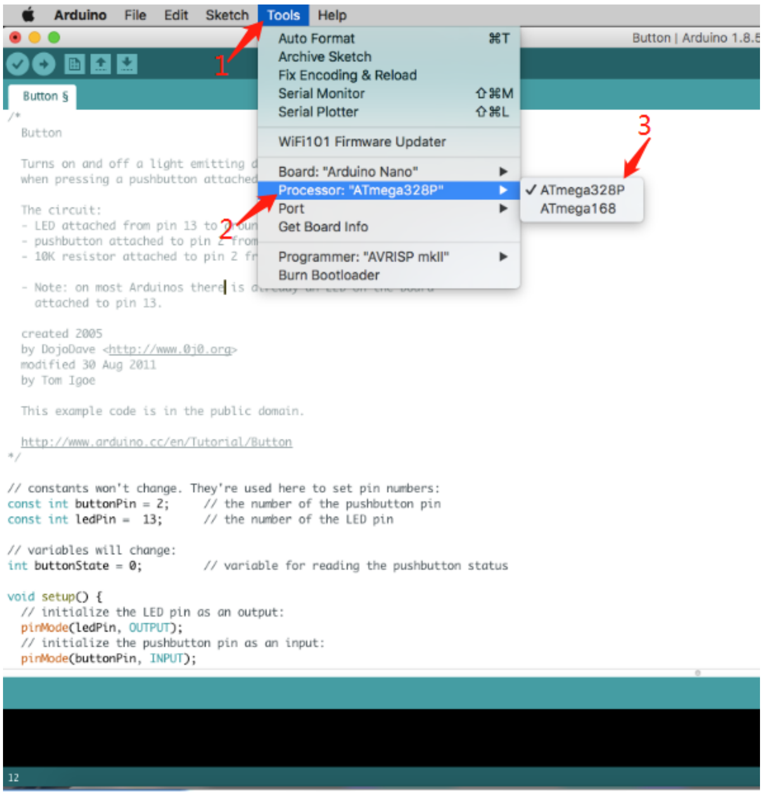


Step4: Select the Arduino Board to Write (we are Nano board)



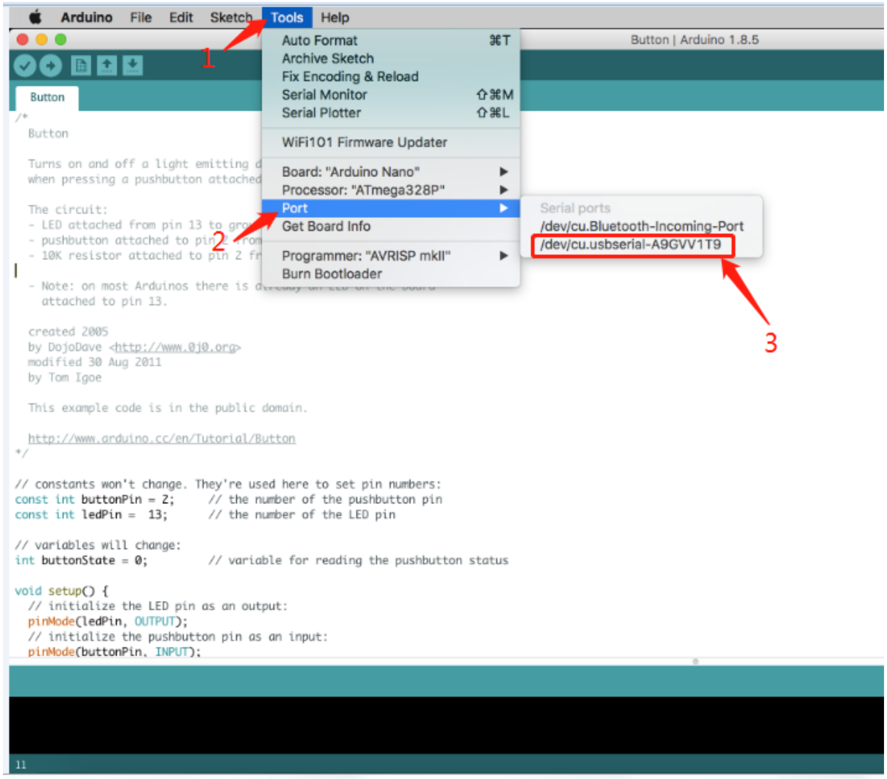
```
/*
  Button

  Turns on and off a light emitting d
  when pressing a pushbutton attache

  The circuit:
  - LED attached from pin 13 to groun
  - pushbutton attached to pin 2 from
  - 10K resistor attached to pin 2 fr

  - Note: on most Arduinos there is a
    attached to pin 13.

  created 2005
  by DojoDave <http://www.0j0.org>
  modified 30 Aug 2011
  by Tom Igoe

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
```

Step5: Select the chip model (the chip of our Nano board is Atmega328P)



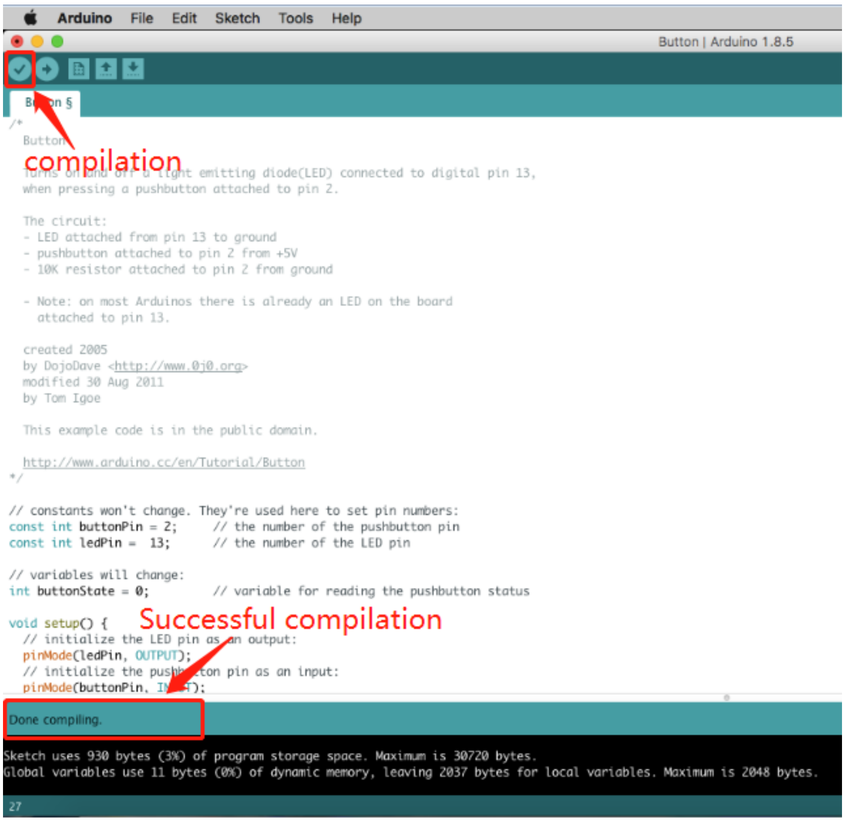
```
/*
  Button

  Turns on and off a light emitting d
  when pressing a pushbutton attached

  The circuit:
  - LED attached from pin 13 to groun
  - pushbutton attached to pin 2 from
  - 10K resistor attached to pin 2 fr

  - Note: on most Arduinos there is a
    attached to pin 13.

  created 2005
  by DojoDave <http://www.0j0.org>
  modified 30 Aug 2011
  by Tom Igoe

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
```

Step6:Select the serial port



```
/*
  Button

  Turns on and off a light emitting d
  when pressing a pushbutton attached

  The circuit:
  - LED attached from pin 13 to gro
  - pushbutton attached to pin 2 from
  - 10K resistor attached to pin 2 fr

  - Note: on most Arduinos there is a
    attached to pin 13.

  created 2005
  by DojoDave <http://www.0j0.org>
  modified 30 Aug 2011
  by Tom Igoe

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
```

Step7:Compile the software



compilation

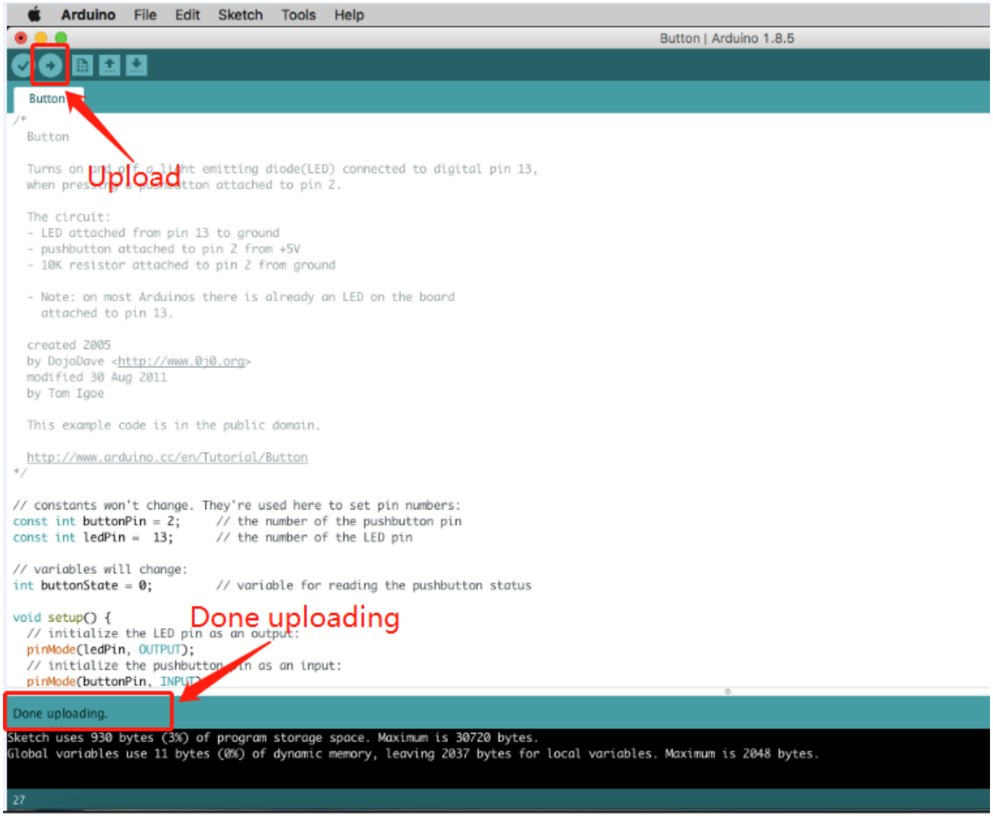
Successful compilation

```
/*
  Button

  Turns on and off a light emitting diode(LED) connected to digital pin 13,
  when pressing a pushbutton attached to pin 2.

  The circuit:
  - LED attached from pin 13 to ground
  - pushbutton attached to pin 2 from +5V
  - 10K resistor attached to pin 2 from ground

  - Note: on most Arduinos there is already an LED on the board
    attached to pin 13.

  created 2005
  by DojoDave <http://www.0j0.org>
  modified 30 Aug 2011
  by Tom Igoe

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
```

Done compiling.

```
Sketch uses 930 bytes (3%) of program storage space. Maximum is 30720 bytes.
Global variables use 11 bytes (0%) of dynamic memory, leaving 2037 bytes for local variables. Maximum is 2048 bytes.
```

Step8:Software download to the Nano board



Upload

Done uploading

```
/*
  Button

  Turns on and off a light emitting diode(LED) connected to digital pin 13,
  when pressing a pushbutton attached to pin 2.

  The circuit:
  - LED attached from pin 13 to ground
  - pushbutton attached to pin 2 from +5V
  - 10K resistor attached to pin 2 from ground

  - Note: on most Arduinos there is already an LED on the board
    attached to pin 13.

  created 2005
  by DojoDave <http://www.0j0.org>
  modified 30 Aug 2011
  by Tom Igoe

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
```

Done uploading.

```
Sketch uses 930 bytes (3%) of program storage space. Maximum is 30720 bytes.
Global variables use 11 bytes (0%) of dynamic memory, leaving 2037 bytes for local variables. Maximum is 2048 bytes.
```