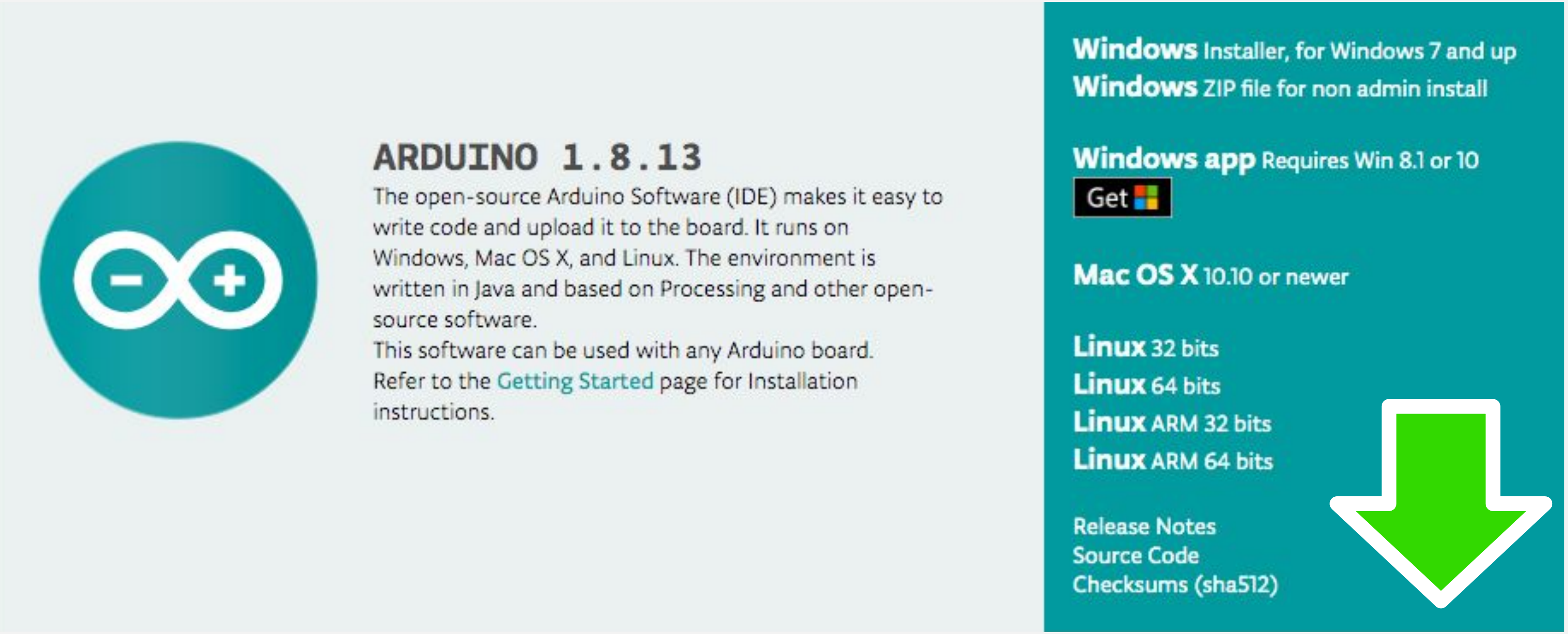
Arduino setup guide for Otto robots

Download & install Arduino IDE



ARDUINO 1.8.13

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.

Windows Installer, for Windows 7 and up

Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10

Mac OS X 10.10 or newer

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Release Notes

Source Code

Checksums (sha512)

If you have Otto Blockly installed and working on your PC **you do not need to install Arduino**

Download Otto DIY Libraries*

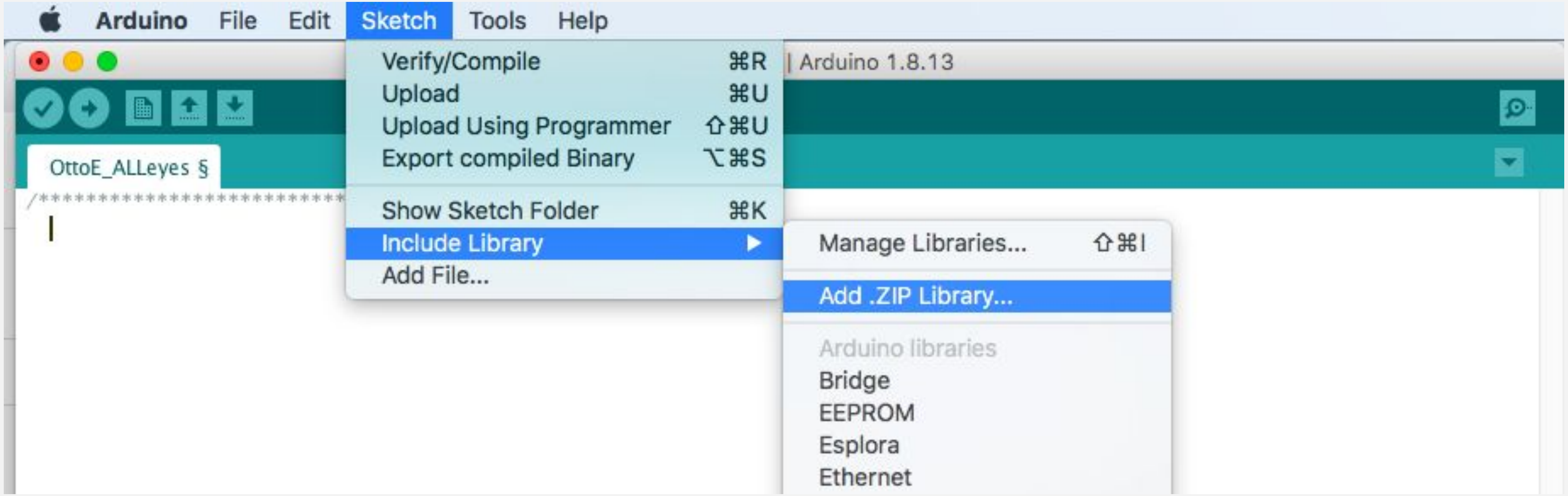
click me!

*If you have already installed Blockly locally, then you do not need to do this, skip to this slide and upload codes directly.

Link source: https://github.com/OttoDIY/OttoDIYLib/

Install Otto Libraries in Arduino

Open Arduino and select **Sketch** from the menu bar, then **Include Library** followed by selecting **Add .ZIP Library**.

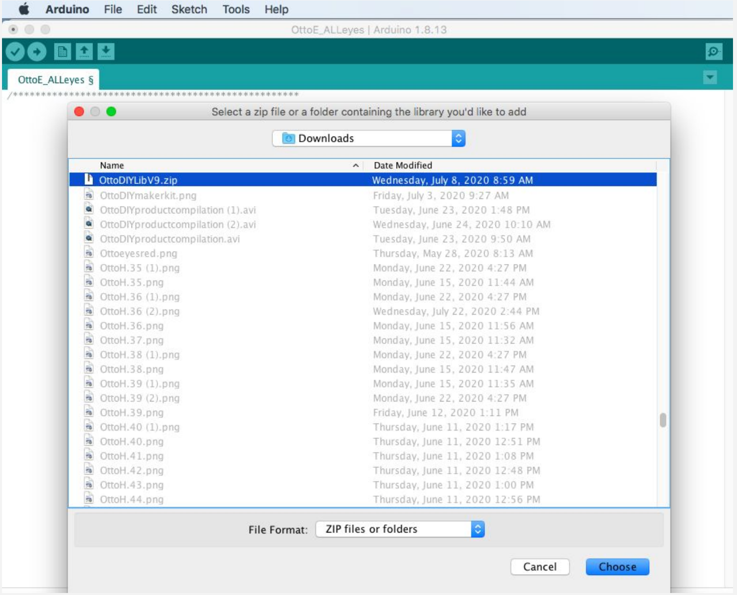


Include the Libraries

You will be prompted to select the library. Navigate to the .zip file's location, that you just downloaded and open as it is.

Navigate under downloads until you see the OttoDIYLibv9.zip then select **that file**.

To check if it was uploaded, select **Sketch** again > Include Library > & scrolling down to the bottom you should see **OttoDIY_Lib**



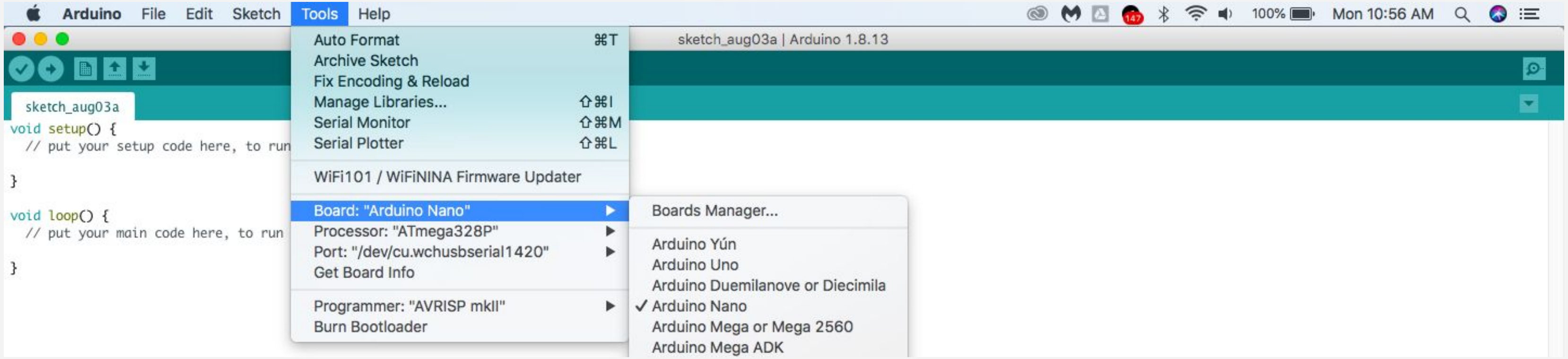
Select a zip file or a folder containing the library you'd like to add

Name	Date Modified
OttoDIYLibV9.zip	Wednesday, July 8, 2020 8:59 AM
OttoDIYmakerkit.png	Friday, July 3, 2020 9:27 AM
OttoDIYproductcompilation (1).avi	Tuesday, June 23, 2020 1:48 PM
OttoDIYproductcompilation (2).avi	Wednesday, June 24, 2020 10:10 AM
OttoDIYproductcompilation.avi	Tuesday, June 23, 2020 9:50 AM
Ottoeyesred.png	Thursday, May 28, 2020 8:13 AM
OttoH.35 (1).png	Monday, June 22, 2020 4:27 PM
OttoH.35.png	Monday, June 15, 2020 11:44 AM
OttoH.36 (1).png	Monday, June 22, 2020 4:27 PM
OttoH.36 (2).png	Wednesday, July 22, 2020 2:44 PM
OttoH.36.png	Monday, June 15, 2020 11:56 AM
OttoH.37.png	Monday, June 15, 2020 11:32 AM
OttoH.38 (1).png	Monday, June 22, 2020 4:27 PM
OttoH.38.png	Monday, June 15, 2020 11:47 AM
OttoH.39 (1).png	Monday, June 15, 2020 11:35 AM
OttoH.39 (2).png	Monday, June 22, 2020 4:27 PM
OttoH.39.png	Friday, June 12, 2020 1:11 PM
OttoH.40 (1).png	Thursday, June 11, 2020 1:17 PM
OttoH.40.png	Thursday, June 11, 2020 12:51 PM
OttoH.41.png	Thursday, June 11, 2020 1:08 PM
OttoH.42.png	Thursday, June 11, 2020 12:48 PM
OttoH.43.png	Thursday, June 11, 2020 1:00 PM
OttoH.44.png	Thursday, June 11, 2020 12:56 PM

File Format: ZIP files or folders

Select Board, Processor & Port Settings

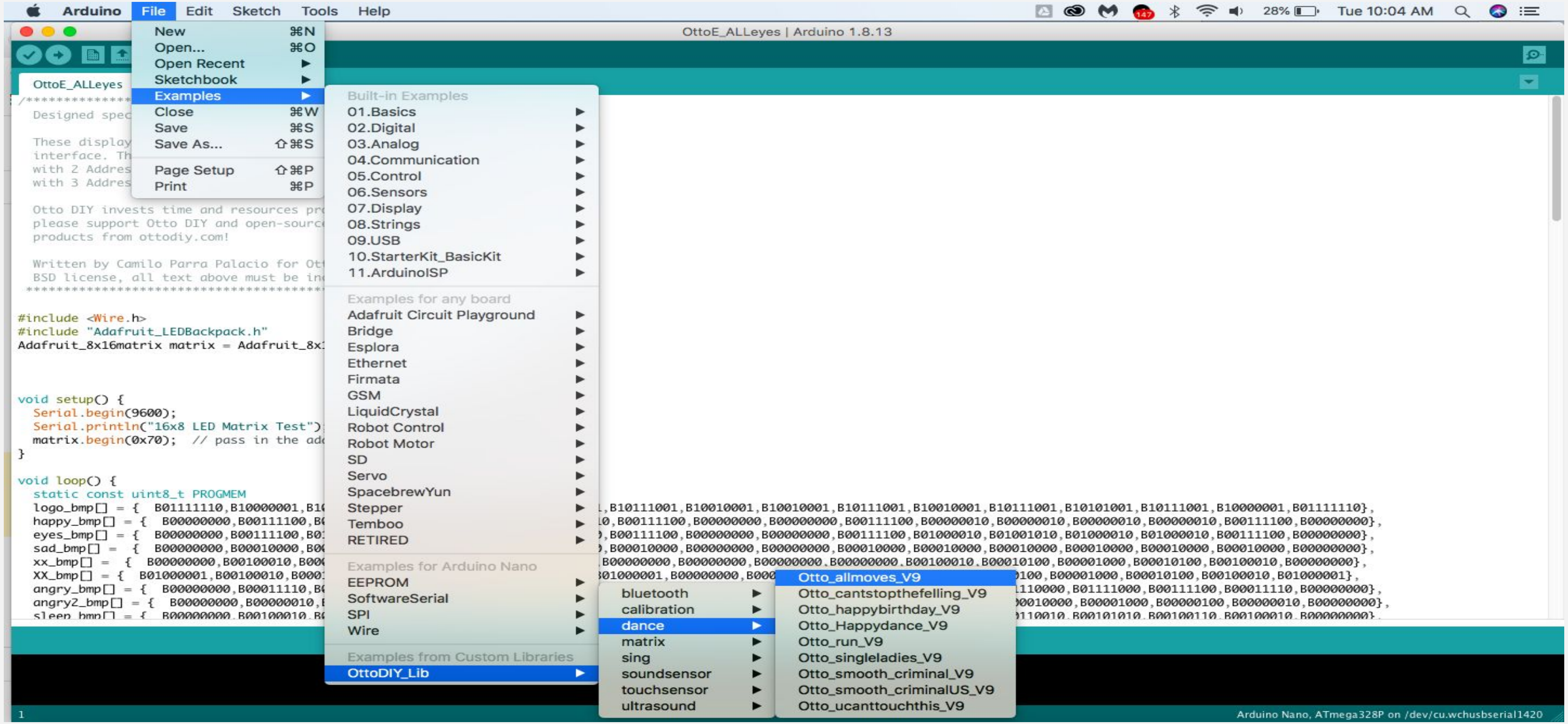
1) Select tools from menu bar
2) For Board select “Arduino Nano”
3) For Processor select “Atmega328P” (choose Old bootloader if error)
4) For Port select “COM #” (where your Otto is connected, this can be different in your computer) if not visible the driver CH340 is not installed



```
void setup() {
  // put your setup code here, to run

}

void loop() {
  // put your main code here, to run

}
```

Board: "Arduino Nano"
Processor: "ATmega328P"
Port: "/dev/cu.wchusbserial1420"
Programmer: "AVRISP mkII"

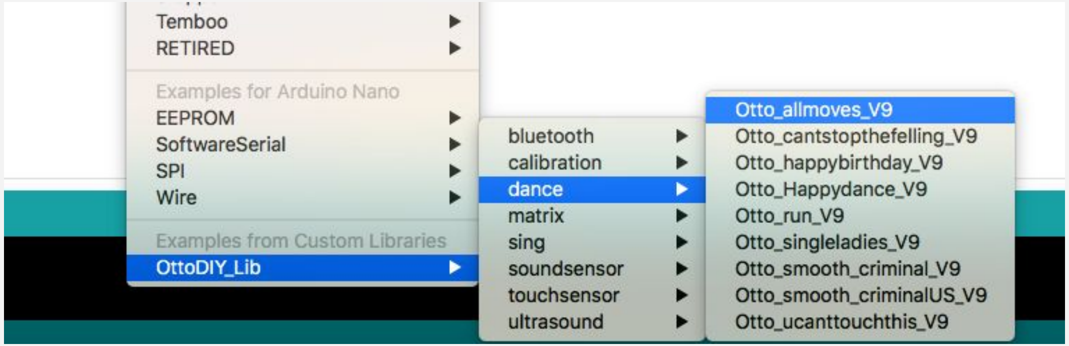
File/Examples/OttoDIYLib/dance/Otto_allmoves

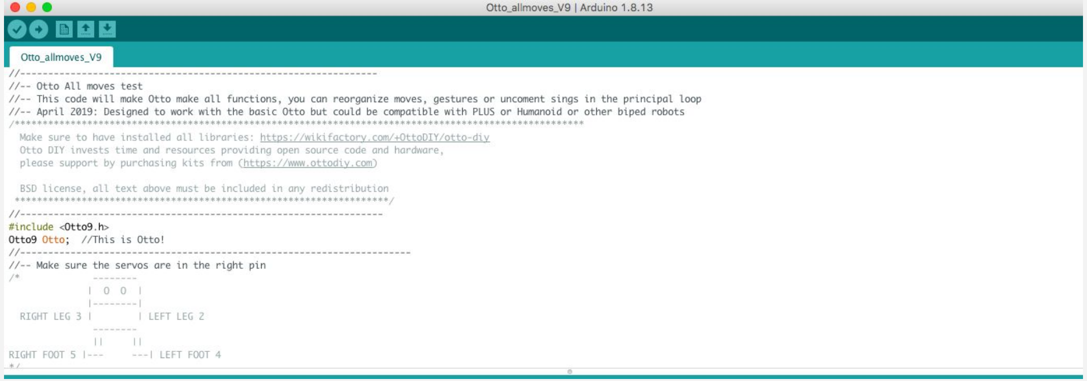


```
/***************
  Designed spec

  These display
  interface. Th
  with 2 Addres
  with 3 Addres

  Otto DIY invests time and resources pro
  please support Otto DIY and open-source
  products from ottodiy.com!

  Written by Camilo Parra Palacio for Ott
  BSD license, all text above must be inc
 ****************************************

#include <Wire.h>
#include "Adafruit_LEDBackpack.h"
Adafruit_8x16matrix matrix = Adafruit_8x1



void setup() {
  Serial.begin(9600);
  Serial.println("16x8 LED Matrix Test");
  matrix.begin(0x70);  // pass in the add
}

void loop() {
  static const uint8_t PROGMEM
  logo_bmp[] = {  B01111110,B10000001,B10
  happy_bmp[] = {  B00000000,B00111100,B0
  eyes_bmp[] = {  B00000000,B00111100,B01
  sad_bmp[] =  {  B00000000,B00010000,B00
  xx_bmp[] =  {  B00000000,B00100010,B000
  XX_bmp[] = {  B01000001,B00100010,B0001
  angry_bmp[] = {  B00000000,B00011110,B0
  angry2_bmp[] = {  B00000000,B00000010,B
  sleep_bmp[] = {  B00000000,B00100010,B0
```

```
1,B10111001,B10010001,B10010001,B10111001,B10010001,B10111001,B10101001,B10111001,B10000001,B01111110},
10,B00111100,B00000000,B00000000,B00111100,B00000010,B00000010,B00000010,B00000010,B00111100,B00000000},
0,B00111100,B00000000,B00000000,B00111100,B01000010,B01001010,B01000010,B01000010,B00111100,B00000000},
0,B00010000,B00000000,B00000000,B00010000,B00010000,B00010000,B00010000,B00010000,B00010000,B00000000},
,B00000000,B00000000,B00000000,B00000000,B00100010,B00010100,B00001000,B00010100,B00100010,B00000000},
01000001,B00000000,B000
0100,B00001000,B00010100,B00100010,B01000001},
1110000,B01111000,B00111100,B00011110,B00000000},
00010000,B00001000,B00000100,B00000010,B00000000},
0110010,B00101010,B00100110,B00100010,B00000000},
```

Code from examples tab

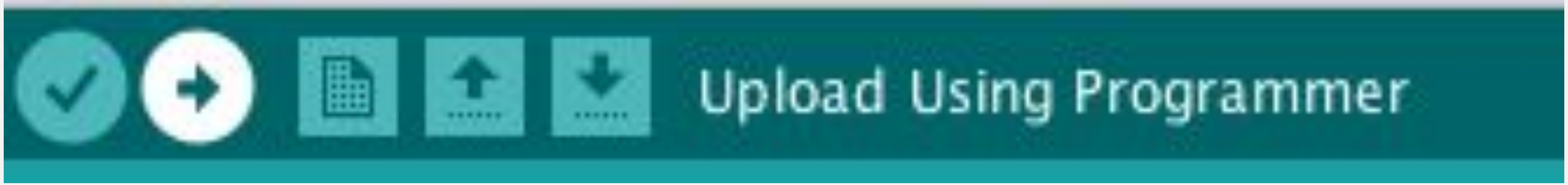




```
//----------------------------------------------------------------
//-- Otto All moves test
//-- This code will make Otto make all functions, you can reorganize moves, gestures or uncoment sings in the principal loop
//-- April 2019: Designed to work with the basic Otto but could be compatible with PLUS or Humanoid or other biped robots
/******************************************************************************************************
  Make sure to have installed all libraries: https://wikifactory.com/+OttoDIY/otto-diy
  Otto DIY invests time and resources providing open source code and hardware,
  please support by purchasing kits from (https://www.ottodiy.com)

  BSD license, all text above must be included in any redistribution
 ********************************************************************/
//-----------------------------------------------------------------
#include <Otto9.h>
Otto9 Otto;  //This is Otto!
//---------------------------------------------------------------------
//-- Make sure the servos are in the right pin
/*             --------
              |  O  O  |
              |--------|
  RIGHT LEG 3 |        | LEFT LEG 2
               --------
               ||     ||
RIGHT FOOT 5 |---     ---| LEFT FOOT 4
*/
```

Upload code to Otto

Lastly select the arrow pointing to the right to upload.



This will automatically check and verify the code and upload the code to Otto.

If there were no problems your Otto is now a walking dancing machine! Well only the legs but that is ok because now we tested that everything is working ;) If not you need to check the previous steps again.

You are all setup!

Now you can code Otto using Arduino IDE
play with the other examples depending on your robot kit.

Install Driver on Mac

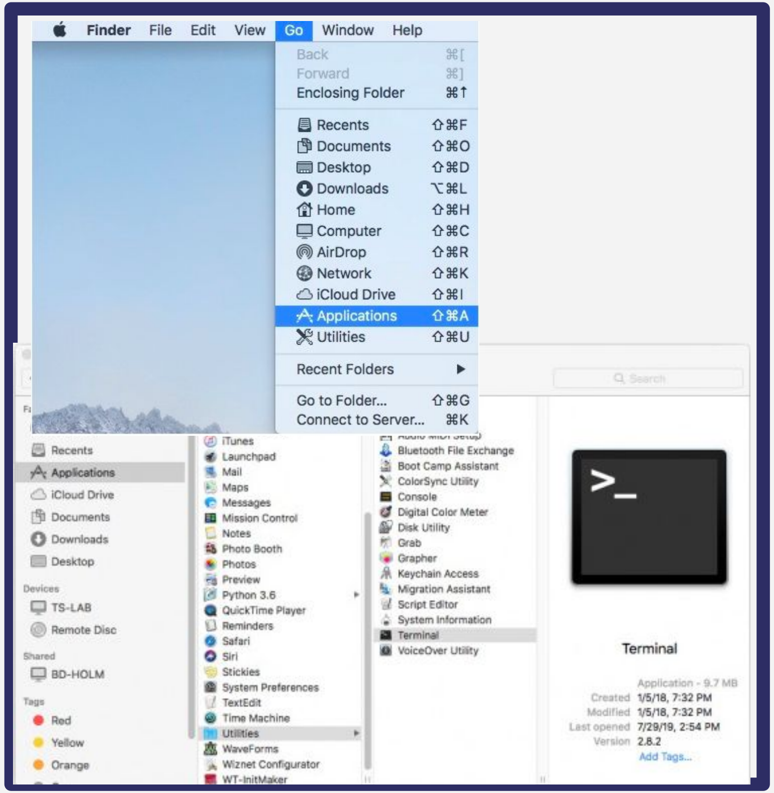
This slides are if you had problem seeing Otto when connected to USB

From your desktop click **go**, and then **applications**

Click **utilities**, then **terminal**

For Mac OSX v10.9+ type in the terminal and hit enter: **cd /Library/Extensions**

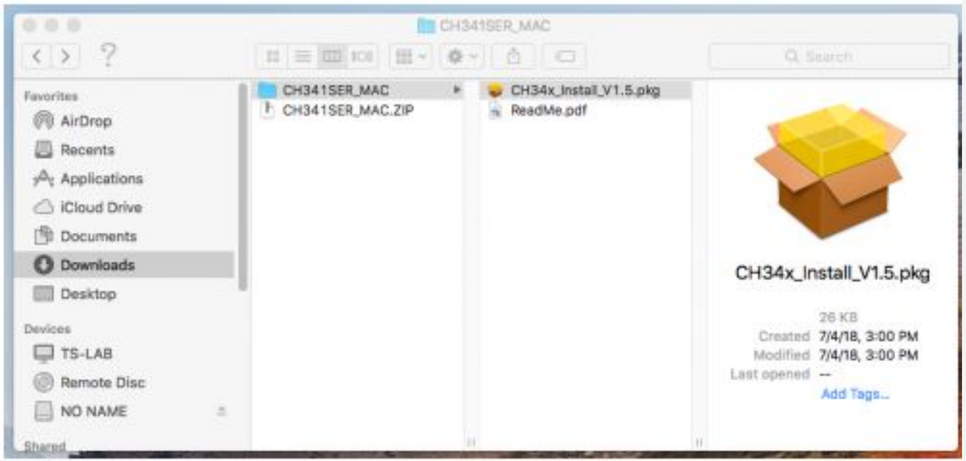
For Mac OSX v10.8 and below type and hit enter: **Cd /System/Library/Extension**



Install CH340 Mac

Next, download & install the CH340 Driver for Mac. You can get it from here.

Once installed you will need to restart the computer and done!



26 KB
Created 7/4/18, 3:00 PM
Modified 7/4/18, 3:00 PM
Last opened --

Lets double check

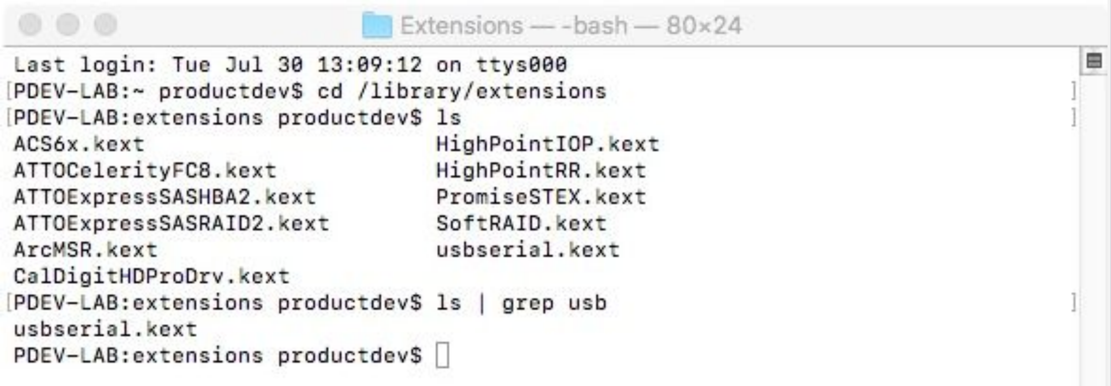
To check to see if the CH340 driver is in the correct path, use the following command to list the contents of the folder: ls

To look for CH340 driver files (i.e. **usb.kext** or **usbserial.kext**) in the path, you could use the following command:

```
ls | grep usb
```

You should see

something similar here



```
Last login: Tue Jul 30 13:09:12 on ttys000
[PDEV-LAB:~ productdev$ cd /library/extensions
[PDEV-LAB:extensions productdev$ ls
ACS6x.kext                      HighPointIOP.kext
ATTOCelerityFC8.kext            HighPointRR.kext
ATTOExpressSASHBA2.kext         PromiseSTEX.kext
ATTOExpressSASRAID2.kext        SoftRAID.kext
ArcMSR.kext                     usbserial.kext
CalDigitHDProDrv.kext
[PDEV-LAB:extensions productdev$ ls | grep usb
usbserial.kext
PDEV-LAB:extensions productdev$ 
```

Troubleshooting driver continued

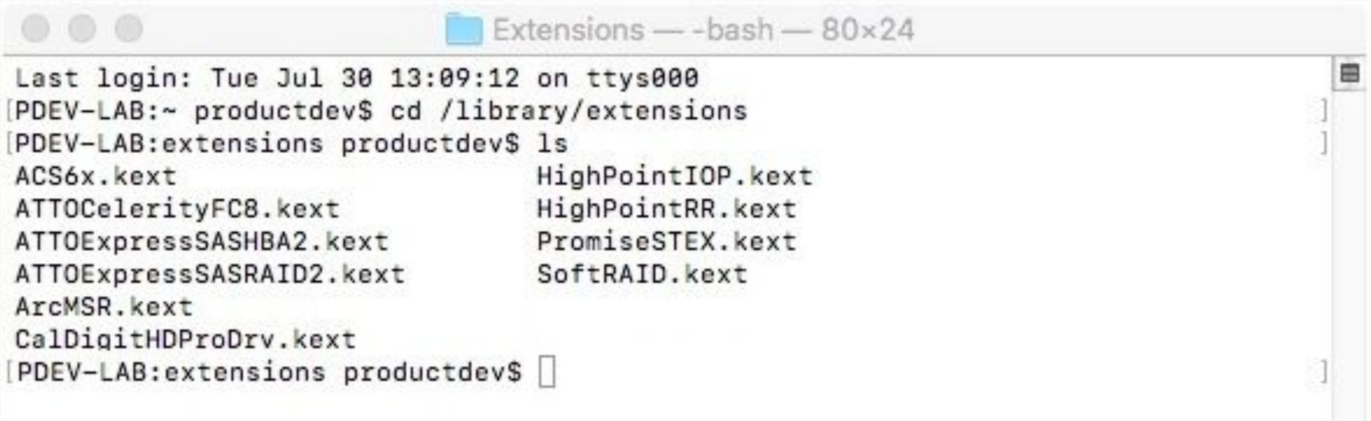
If you have found the file in the path, you will need to run each of the following commands below in the CLI/Terminal to remove old CH340 drivers. In this case, there was only the **usbserial.kext** file but it does not hurt to run both commands. Make sure to have administrative privileges to ensure that the drivers are removed.

```
sudo rm -rf /Library/Extensions/usb.kext
sudo rm -rf /Library/Extensions/usbserial.kext
```

Troubleshooting driver continued

Check if the old drivers were removed in the paths by using the ls command with your respective OS version. You will notice that the *.kext file is removed from the respective paths. In this case, the usbserial.kext was removed from Mac OSX High Sierra.

```
ls
```

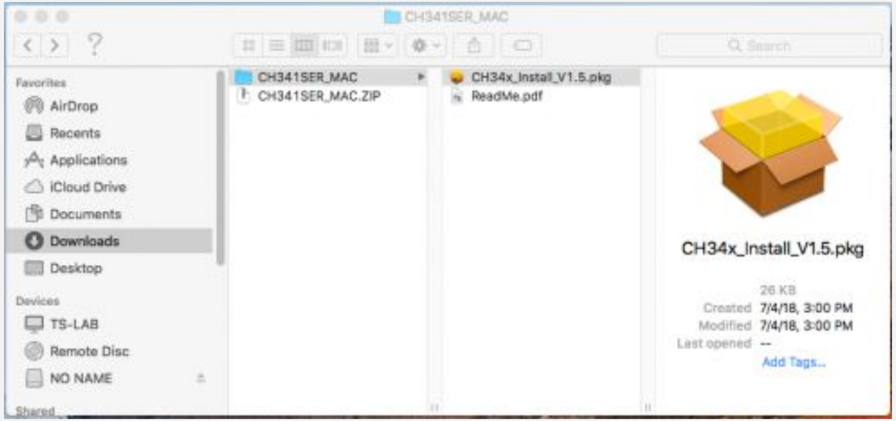


```
Last login: Tue Jul 30 13:09:12 on ttys000
PDEV-LAB:~ productdev$ cd /library/extensions
PDEV-LAB:extensions productdev$ ls
ACS6x.kext                      HighPointIOP.kext
ATTOCelerityFC8.kext            HighPointRR.kext
ATTOExpressSASHBA2.kext         PromiseSTEX.kext
ATTOExpressSASRAID2.kext        SoftRAID.kext
ArcMSR.kext
CalDigitHDProDrv.kext
PDEV-LAB:extensions productdev$ 
```

Troubleshooting driver continued

Next, download and extract the folder here.

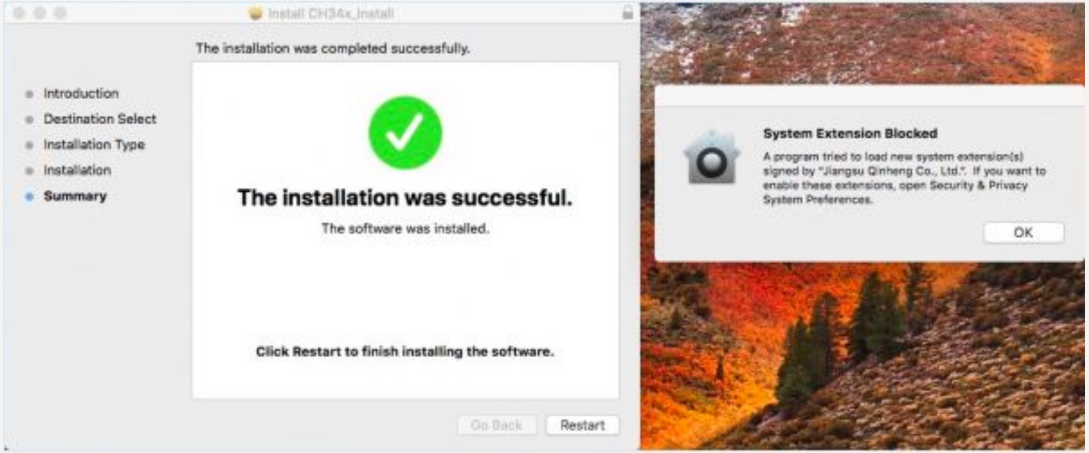
Then, open the "***.pkg**" file from the unzipped folder and follow the instructions. You'll need to restart your computer for the changes to take effect.



26 KB
Created 7/4/18, 3:00 PM
Modified 7/4/18, 3:00 PM
Last opened --

Troubleshooting driver continued

Heads up! Depending on your settings, you may need to adjust your **Security & Privacy** settings to allow the CH340 drivers to function. A window may pop up indicating that the drivers have been block as shown in the image below



The installation was completed successfully.

The installation was successful.

The software was installed.

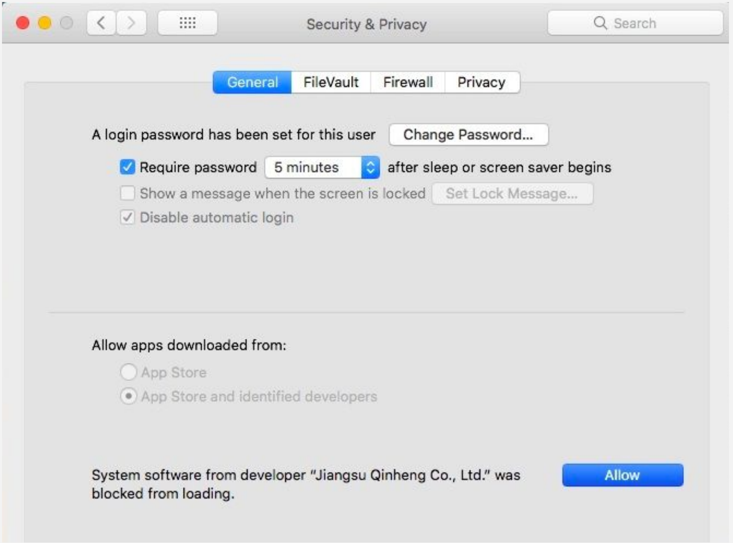
Click Restart to finish installing the software.

System Extension Blocked

A program tried to load new system extension(s) signed by "Jiangsu Qinheng Co., Ltd.". If you want to enable these extensions, open Security & Privacy System Preferences.

Troubleshooting driver

If you receive a window that indicates that the system extension is blocked, you'll need to open a search with Spotlight by pressing ⌘ (**Command**) + space bar (**Space Bar**). Type **Security & Privacy** and click on the "**Allow**" button to enable the CH340 drivers.



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

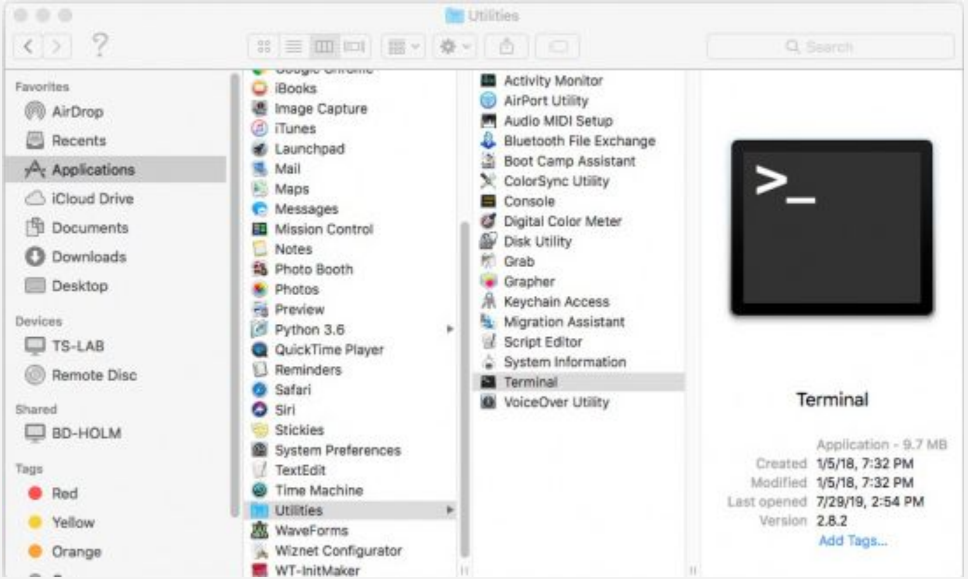
System software from developer "Jiangsu Qinheng Co., Ltd." was blocked from loading.

Driver Verification

To verify that your driver is working, you should see a difference in the following pictures after plugging the CH340 to a USB port.

Command Line

Open the Terminal by heading to **Applications** > **Utilities** > **Terminal** if the program is not open yet.



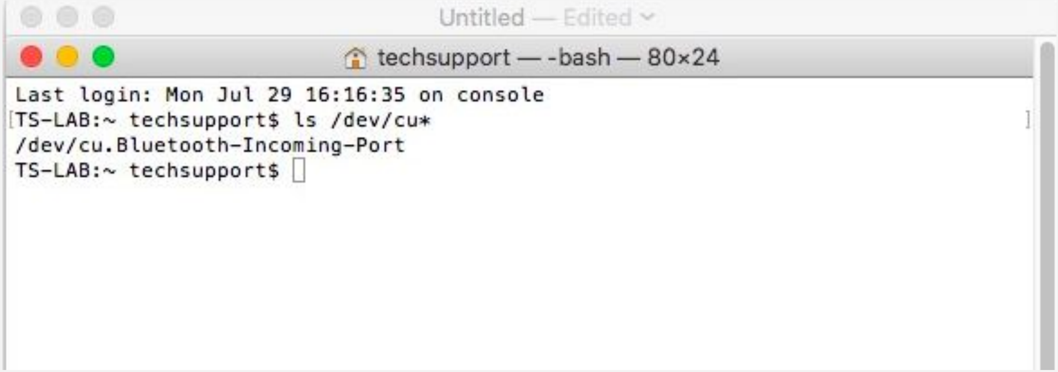
Application - 9.7 MB
Created 1/5/18, 7:32 PM
Modified 1/5/18, 7:32 PM
Last opened 7/29/19, 2:54 PM
Version 2.8.2

Driver Verification Continued

Next, run the following command:

```
ls /dev/cu*
```

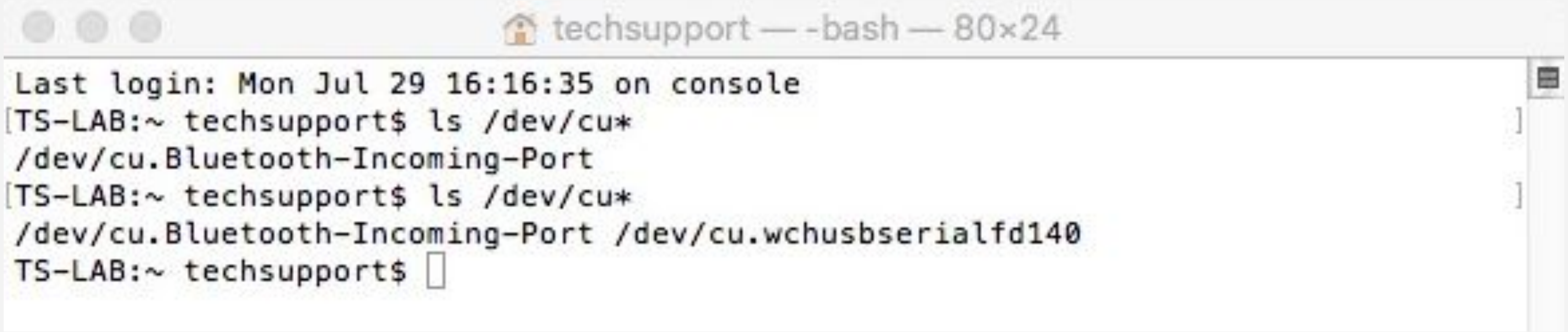
A list of devices connected to your Mac's COM ports will be displayed as a result. Assuming that the CH340 is not connected to your computer, you should see something similar to the image.



```
Last login: Mon Jul 29 16:16:35 on console
TS-LAB:~ techsupport$ ls /dev/cu*
/dev/cu.Bluetooth-Incoming-Port
TS-LAB:~ techsupport$ 
```

Driver Verification Continued

Connect the CH340 to one of your Mac's COM ports. Check for the following changes (your board may show up under a different device name). The CH340 should show up as **/dev/cu.wchusbserial*****. Depending on your computer, the COM port may show up as a different number.



```
Last login: Mon Jul 29 16:16:35 on console
TS-LAB:~ techsupport$ ls /dev/cu*
/dev/cu.Bluetooth-Incoming-Port
TS-LAB:~ techsupport$ ls /dev/cu*
/dev/cu.Bluetooth-Incoming-Port /dev/cu.wchusbserialfd140
TS-LAB:~ techsupport$
```