Welcome

Thank you for choosing Freenove products!
Here are some useful information to help you get started.

How to Start

If you are reading this, you should have downloaded the ZIP file for this product.

Unzip it and you will get a folder contains tutorials and related files. Please start with this PDF tutorial first.

! Please unzip the ZIP file instead of opening the file in the ZIP file directly.
! Please do not move, delete or rename files in the folder just unzipped.

Solve a Problem

Encountered problems? Don't panic!

Whether it is packaging damage, quality problems, or questions encountered in use. Just send us an email. We will reply to you within one working day and provide a solution.

support@freenove.com

Safety

Pay attention to safety when using and storing this product:

- Do not expose children under 6 years of age to this product. Put it out of their reach.
- Children lack safety ability should use this product under the guardianship of adults.
- This product contains small and sharp parts. Do not swallow, prick and scratch to avoid injury.
- This product contains conductive parts. Do not hold them to touch power supply and other circuits.
- Some parts will rotate or move when it works. Do not touch them to avoid being bruised or scratched.
- The wrong operation may cause overheat. Do not touch and disconnect the power supply immediately.
- Operate in accordance with the requirements of the tutorial. Otherwise, the parts may be damaged.
- Store the product in a dry place and avoid direct sunlight.
- Turn off the power of the circuit before leaving.

About

Freenove provides open source electronic products and services.

Freenove is committed to helping customers learn programming and electronic knowledge, quickly realize their creative ideas and product prototypes and launching innovative products. Our services include:

- Kits of robots, smart cars and drones
- Kits for learning Arduino, Raspberry Pi and micro:bit
- Electronic components and modules, tools
- Product customization service

You can learn more about us or get our latest information through our website:

http://www.freenove.com

Copyright

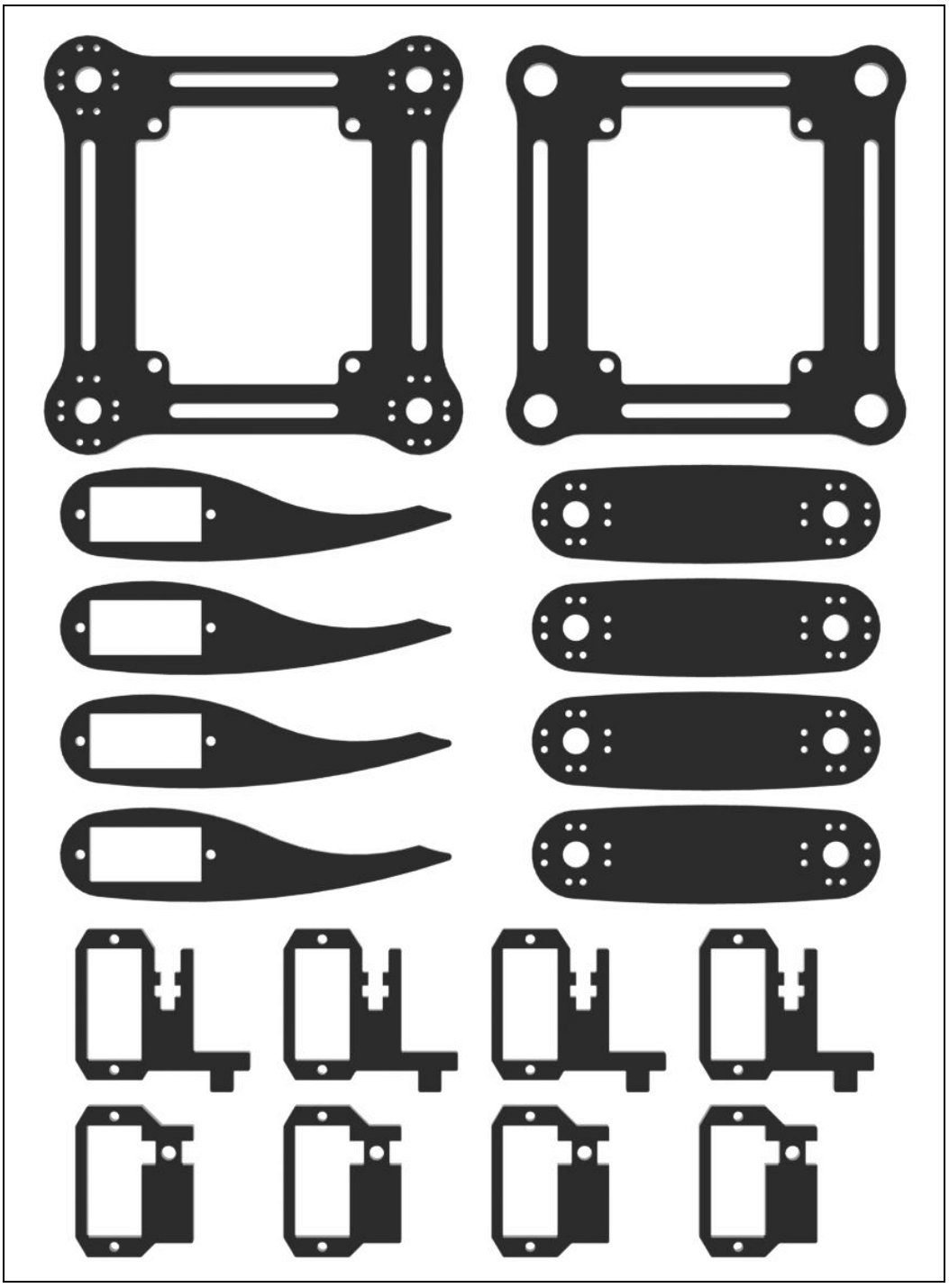
FREENOVE®

FREE YOUR INNOVATION

Contents

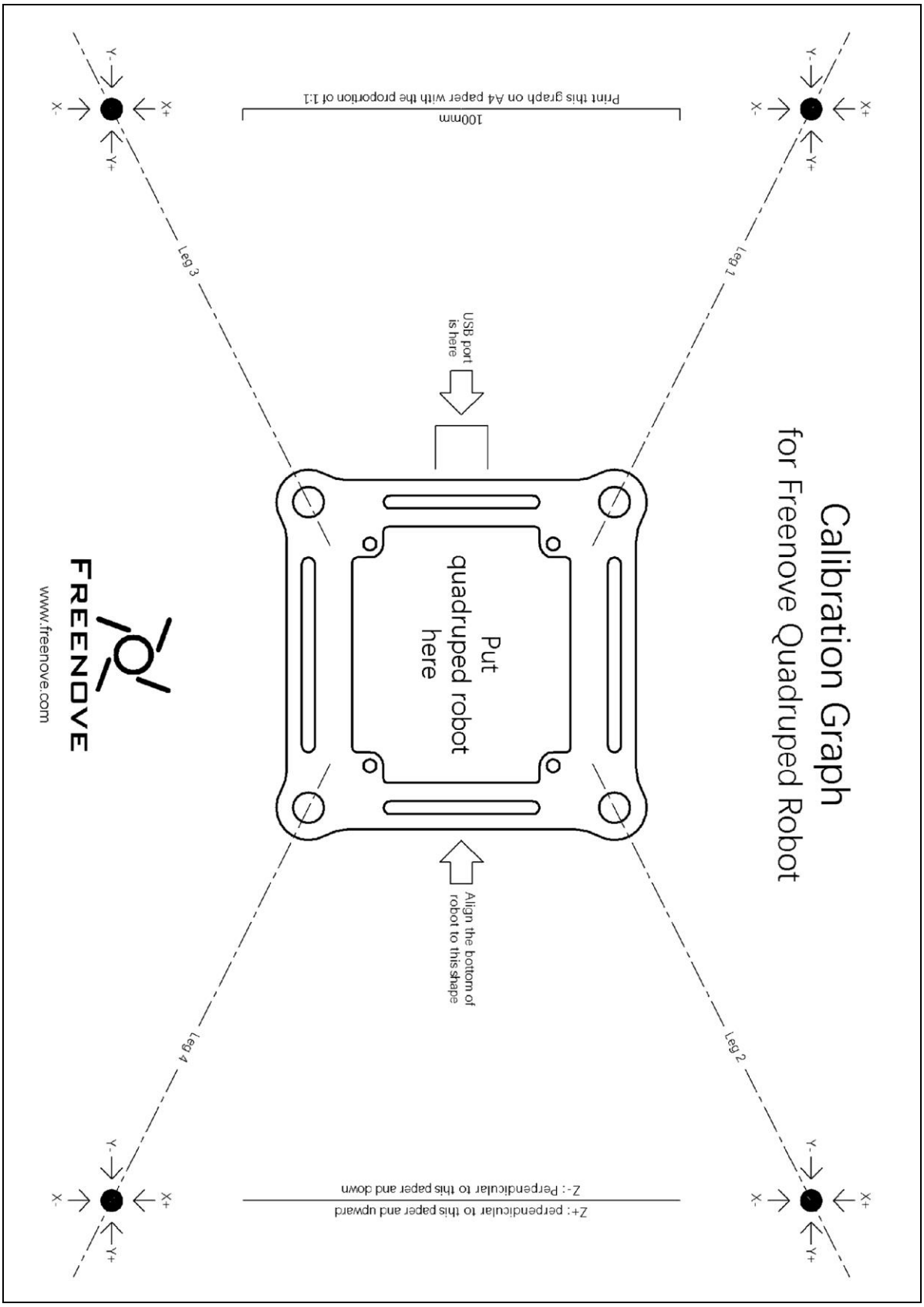
List

Acrylic Parts



The surface of the acrylic parts is covered with a layer of protective film, you need to remove it first. Some holes in the acrylic parts may have residues, you also need to clean them before using.

Calibration Graph

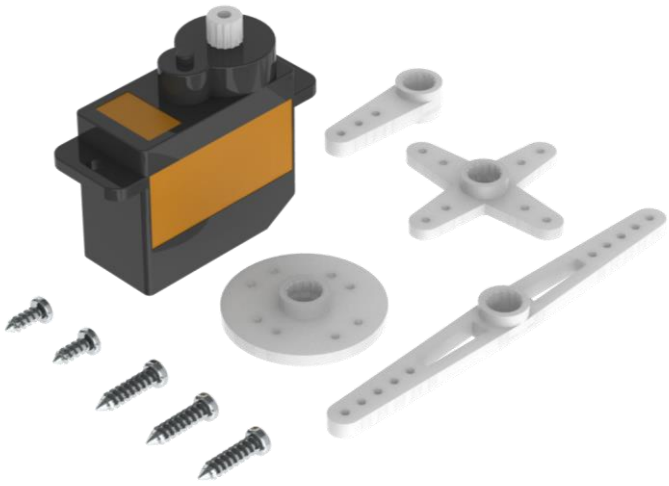


Mechanical Parts

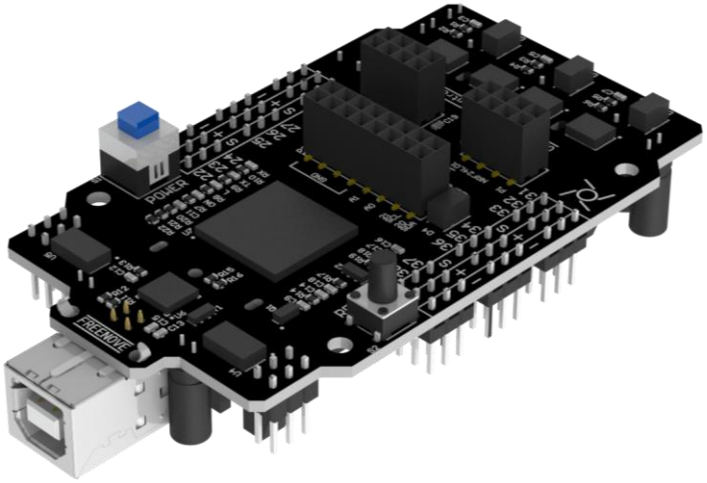
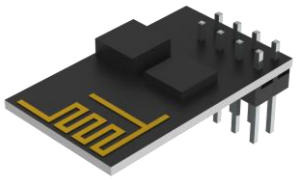
M3*27 Copper Standoff  x6 Freenove	M3*5 Copper Standoff  x6 Freenove	M3*12 Screw  x6 Freenove	M3*8 Screw  x10 Freenove
M2 Nut  x26 Freenove	M3 Nut  x6 Freenove	M2*10 Screw  x26 Freenove	M1.2*7 Self-tapping Screw  x100 Freenove

Dynamic Parts

Servo x12



Electronic Parts

Freenove Crawling Robot Controller x1 	ESP8266 Wi-Fi Module x1 
USB Cable x1 	
Cable Tidy x50cm 	

Tools

Cross Screwdriver x1

Self-prepared Parts

14500 3.7V rechargeable battery x2

! Please prepare the right batteries and fully charge them before assembling.

Assembling without right batteries will cause installation errors, which may damage the servos.

Refer to “AboutBattery.pdf” for detailed information about battery.

Charger for 14500 3.7V rechargeable battery x1

Preface

This is a quadruped robot kit based on Arduino and Processing. Arduino and Processing, which are free and open source, can be runned on Windows, Mac, and Linux platforms.

You can use this kit to assemble a cool quadruped robot, and make it move and act through wireless control. You can also directly control the IO ports on control board. In details, you can use the following devices to control this robot:

- Laptop or desktop computer with Wi-Fi adapter. (Run Windows, Mac or Linux, including Raspberry Pi)
- Android phone or tablet. (Run Android 4.2 or later)
- Remote control. (Freenove Remote Control Kit, FNK0028)

We provide complete code, but also you can easily write code for this robot. By using the Arduino library file we provided, you only need write a few lines of code to control action and movement of the robot. You can also connect sensors and modules to the IO ports and power ports on control board.

The assembled robot is shown below (the wires are not shown).

Arduino

Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects.

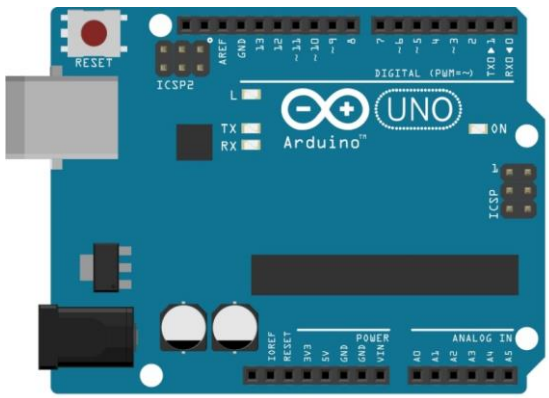
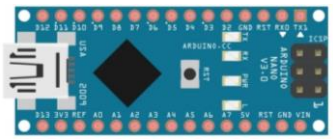
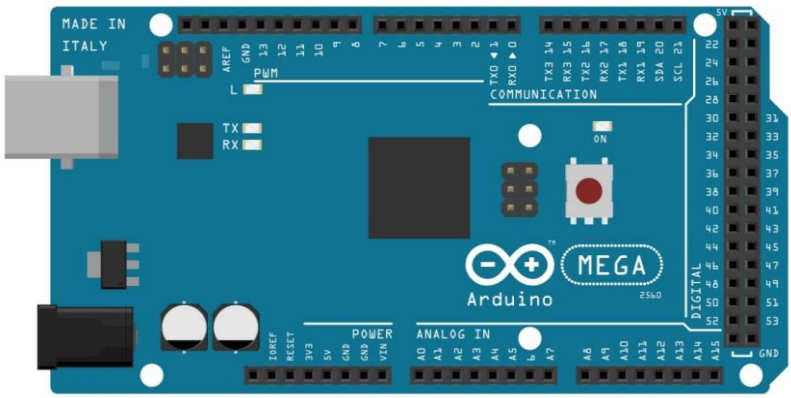
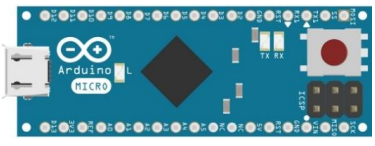
Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on C/C++), and the Arduino software (IDE).

The Arduino software is free and open source, and runs on the Mac, Windows, and GNU/Linux platforms.

Arduino Board

Arduino Board is a circuit board, with integrates micro controller, input, output interface and etc.

Arduino Board has several models. Popular boards include:

Uno 	Nano 
Mega 2560 	Micro 

Arduino Programming Language

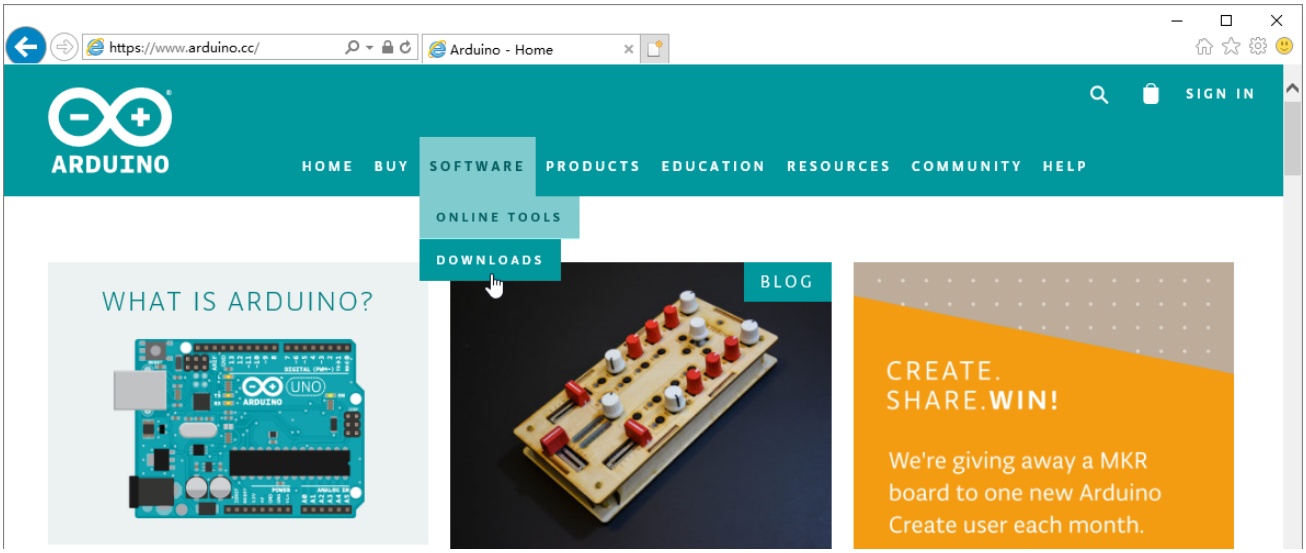
The Arduino programming language is based on C/C++.
Please visit https://www.arduino.cc, click "Learning" > "Reference" for details.

If you want to learn it easily, please visit http://www.freenove.com for Arduino kits designed for beginners.

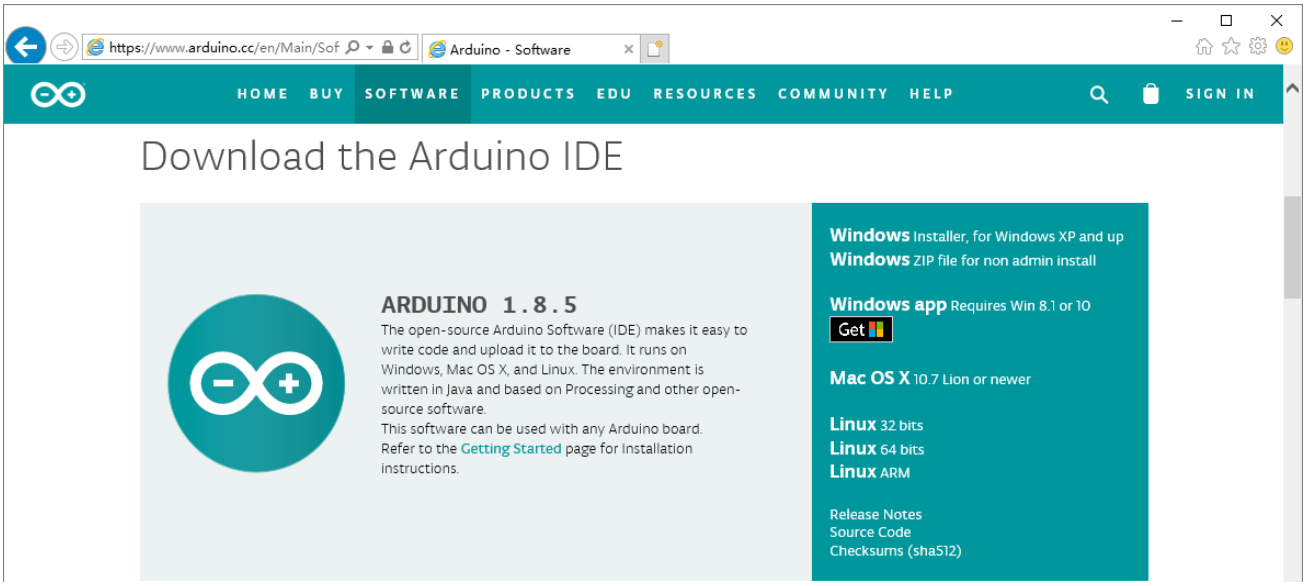
Arduino Software

The Arduino Software (IDE) allows you to write code in the Arduino programming language and upload them to your board.

Please visit https://www.arduino.cc, then click "SOFTWARE" – "DOWNLOADS" to enter the downloads page.

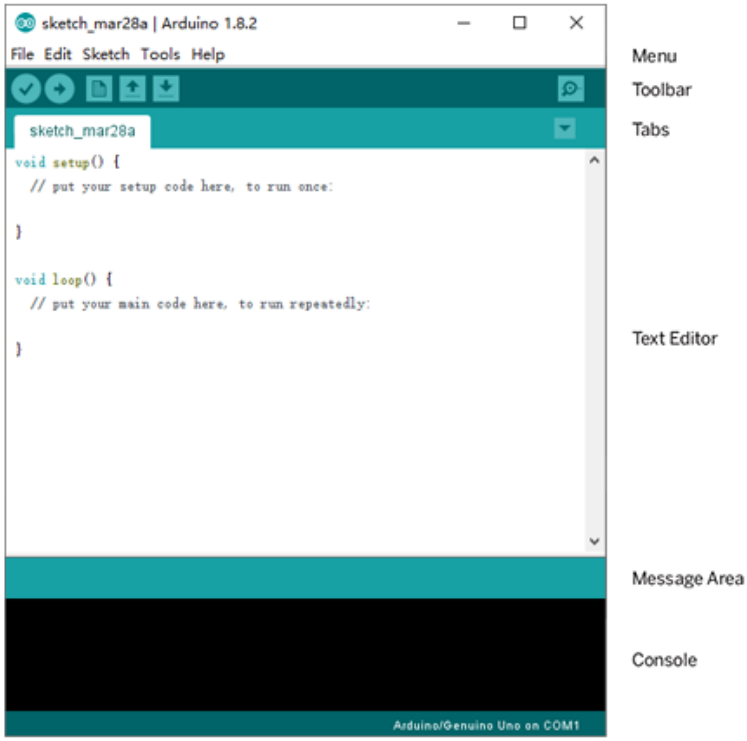


Find "Download the Arduino IDE". Microsoft Windows users please click the "Windows Installer".



After the download is complete, run the installer and complete the installation.

Open the Arduino Software, the interface of Arduino Software is as follows:



Programs written using Arduino Software (IDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension **.ino**. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

Verify
Checks your code for errors compiling it.

Upload
Compiles your code and uploads it to the configured board.

New
Creates a new sketch.

Open
Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.

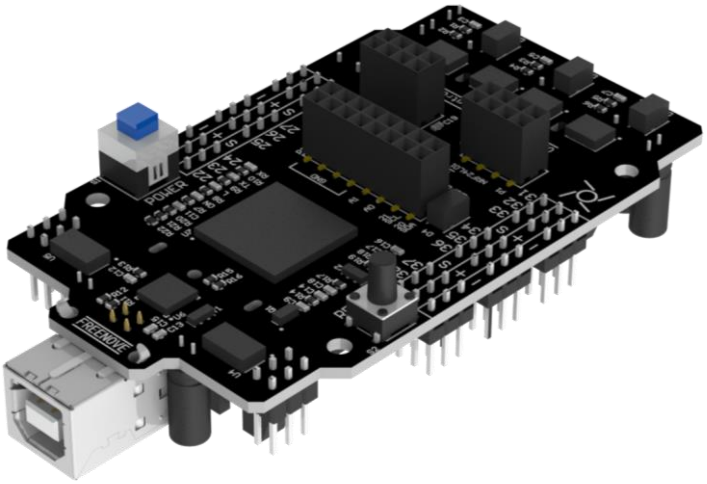
Save
Saves your sketch.

Serial Monitor
Opens the serial monitor.

Additional commands are found within the five menus: File, Edit, Sketch, Tools, Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

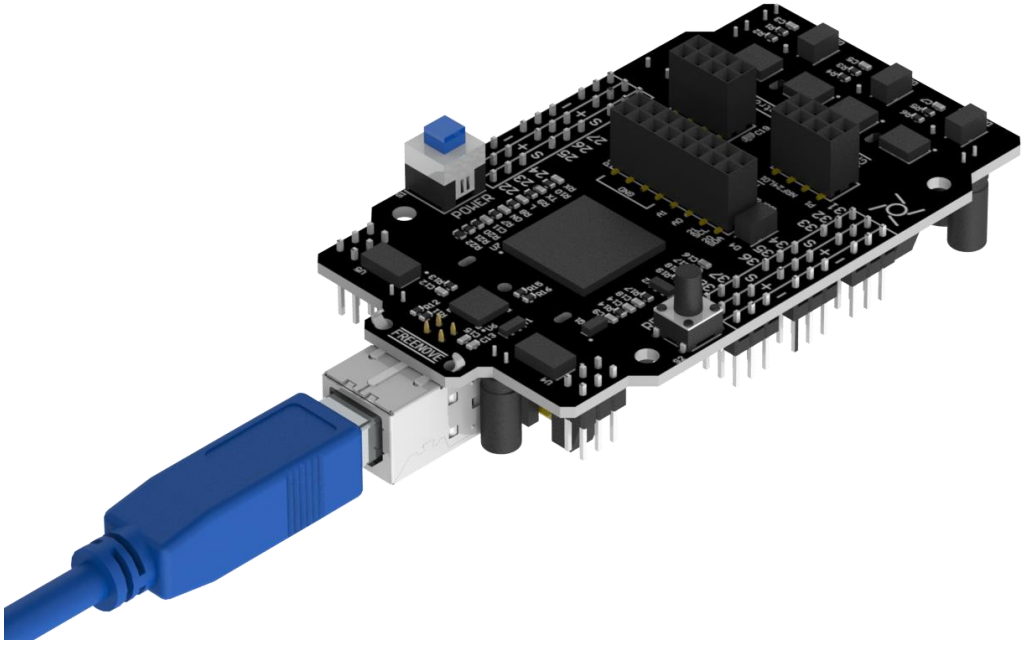
Freenove Crawling Robot Controller

Freenove Crawling Robot Controller is used to control this robot. It is based on Arduino Mega 2560. We changed the circuit, so that it can drive 18 servos and install ESP8266 Wi-Fi module and NRF24L01 wireless module. It also has some IO ports and power ports for your use.

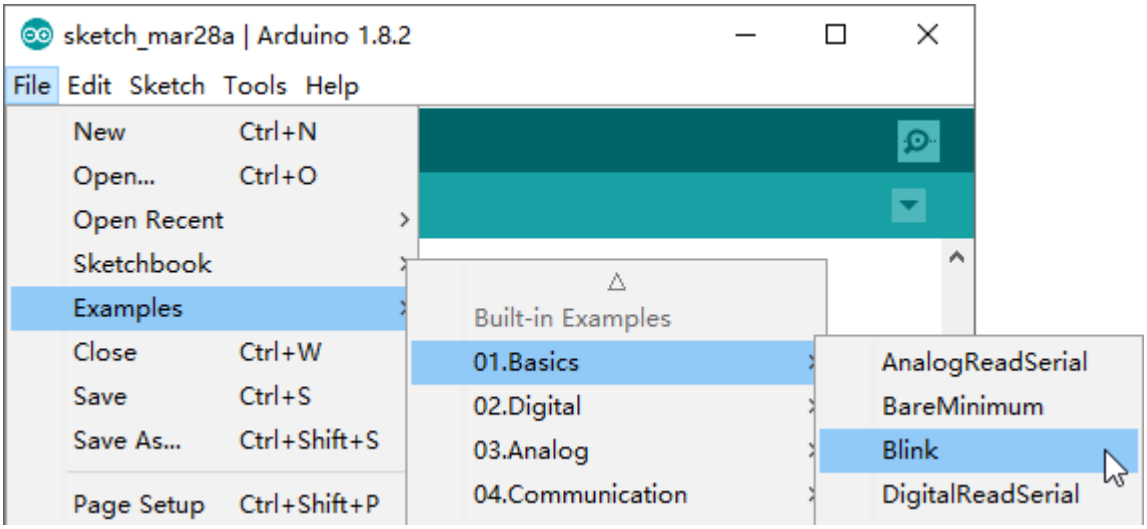


First Use

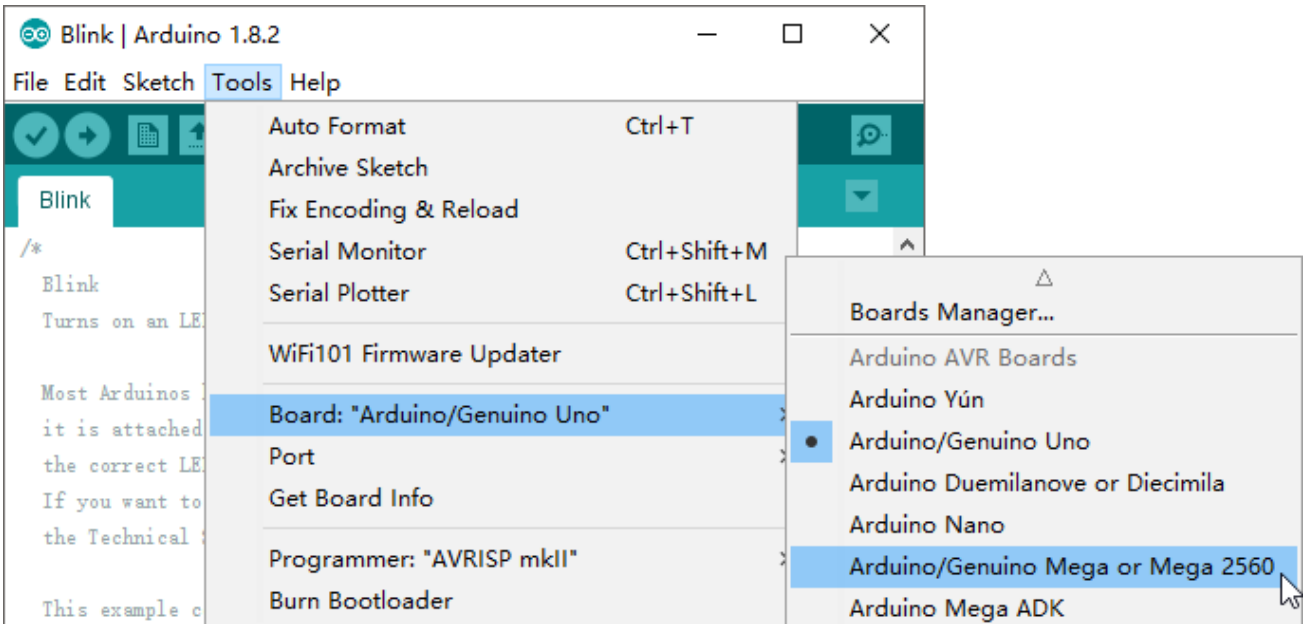
Connect Freenove Crawling Robot Controller to computer with USB cable.



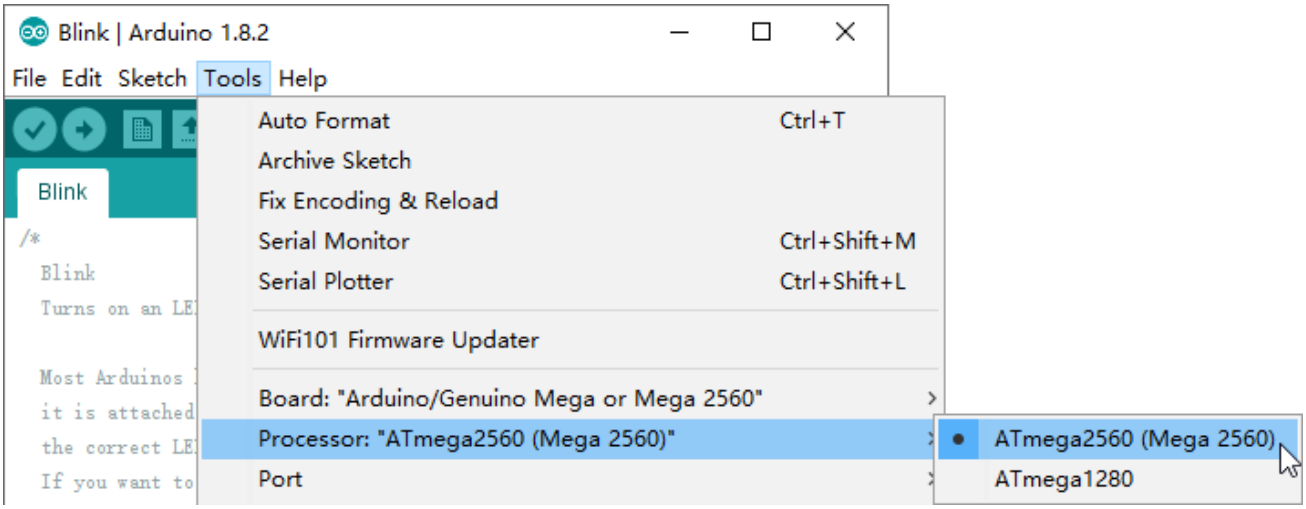
Open the exmple sketch "Blink".



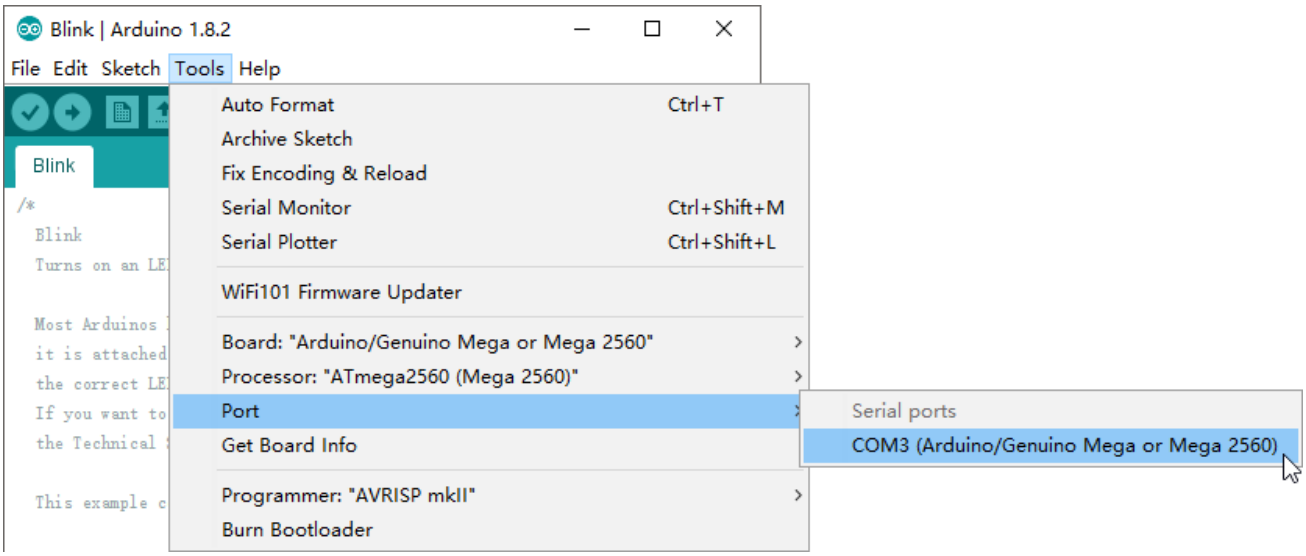
Select "Board" > "Arduino/Genuino Mega or Mega 2560".



Select "Processor" > "ATmega2560 (Mega 2560)".



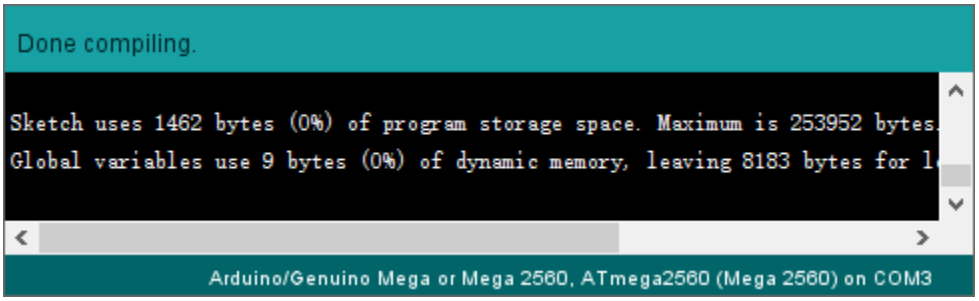
Select "Port". Your port may be different from the following figure. If the board is not detected, please wait for a while, then click "Tools" to check again.



Click "Verify" button.

Figure below shows code are compiling.

Wait a moment for the compiling to be completed. Figure below shows the code size and percentage of space occupation.



Usually, when we write code, if it has a syntax error, the interface will prompt the error message. Then the compiling can't be completed.

Click "Upload" button.

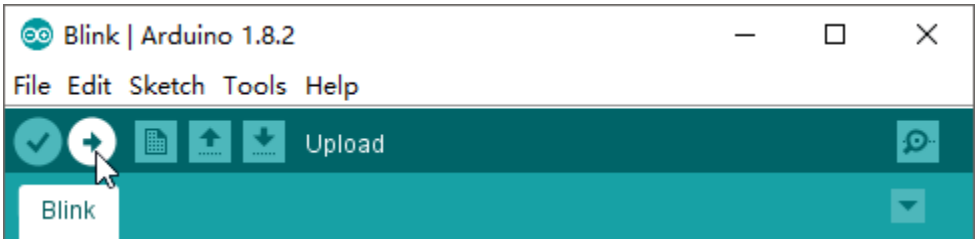
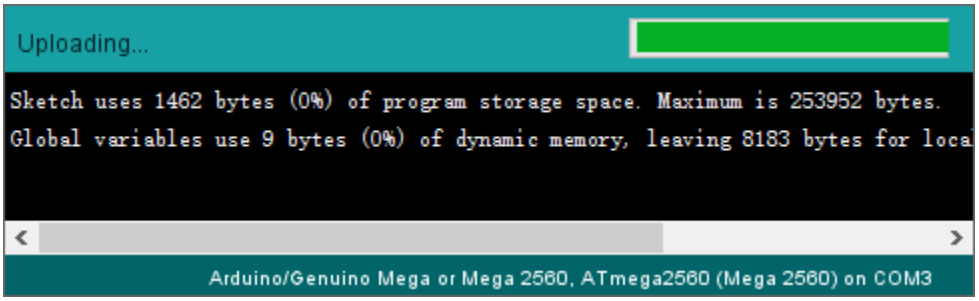
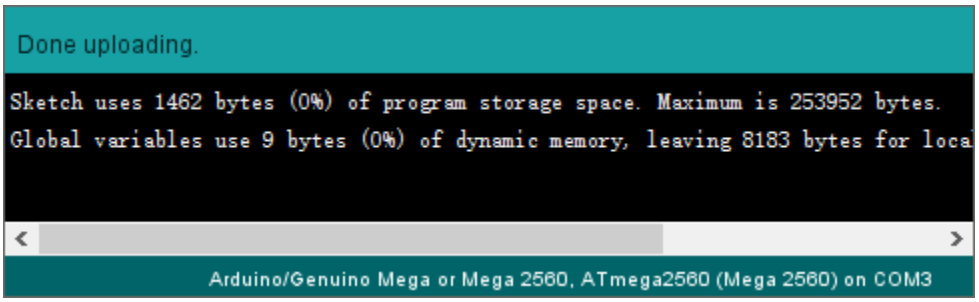


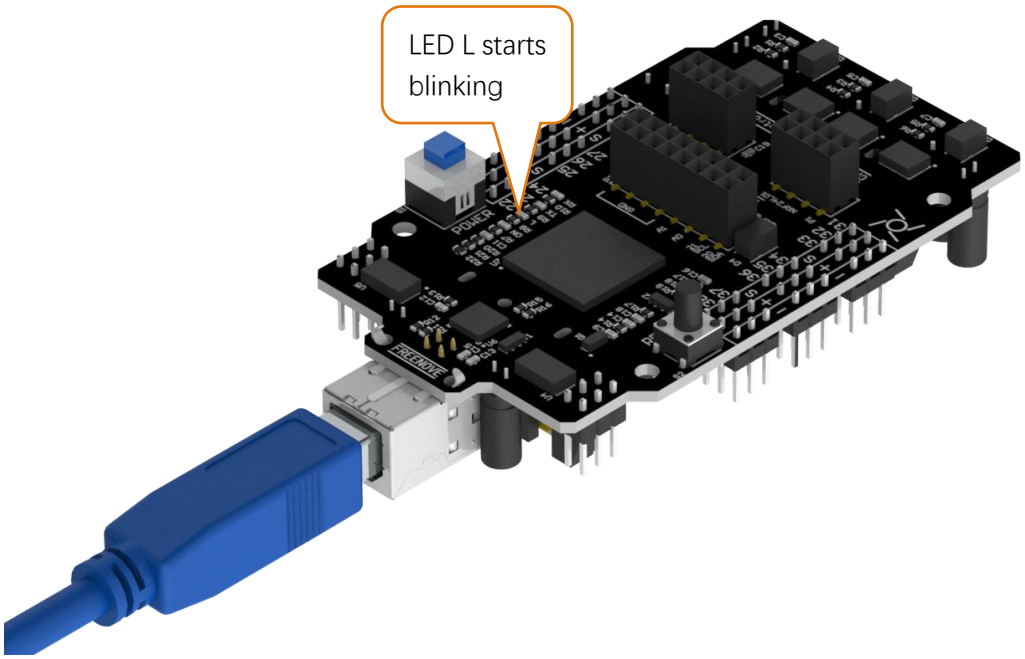
Figure below shows code are uploading.



Wait a moment for the uploading to be completed.



After that, we will see the LED marked with "L" on Freenove Crawling Robot Controller starts blinking.



So far, we have completed the first use. I believe you have felt the joy of Arduino.

Processing

Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts.

The Processing software is free and open source, and runs on the Mac, Windows, and GNU/Linux platforms, which is the same as Arduino software. In fact, Arduino software is based on Processing software. At present, they still have similar interface.

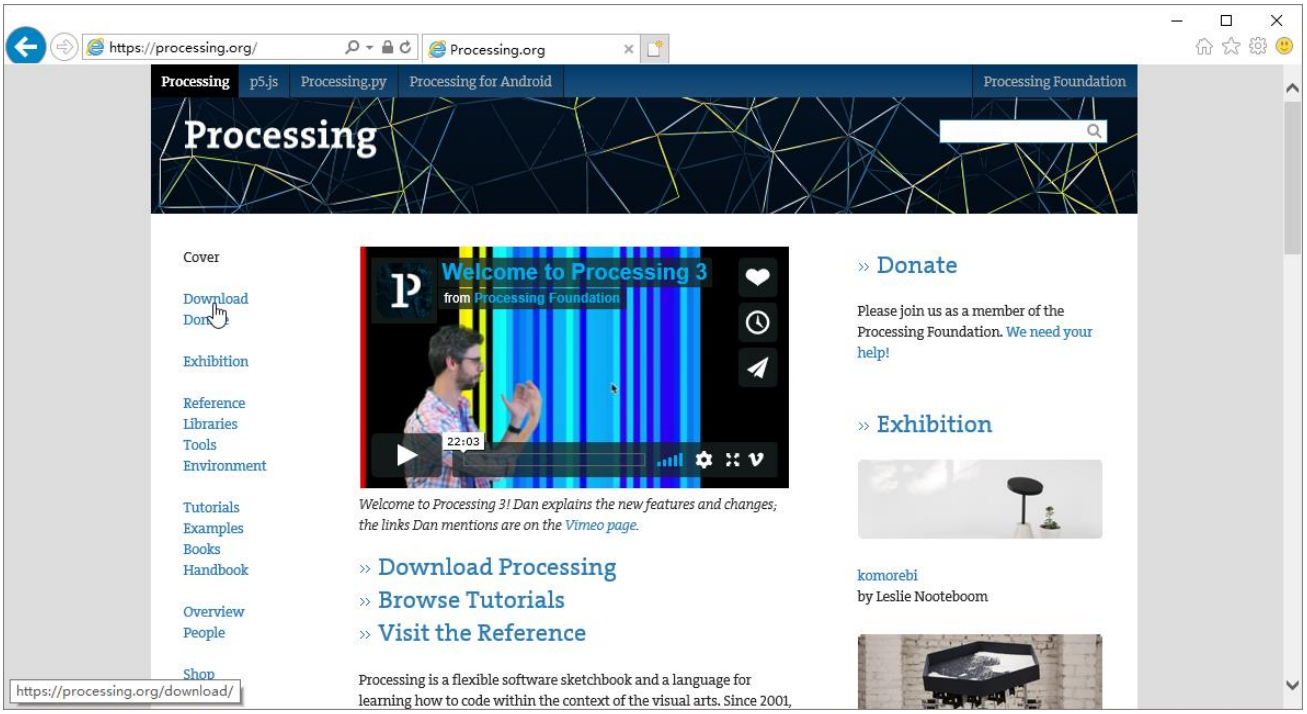
Processing Programming Language

The Processing programming language is based on Java in default.
Please visit https://processing.org/, click "Reference" for details.

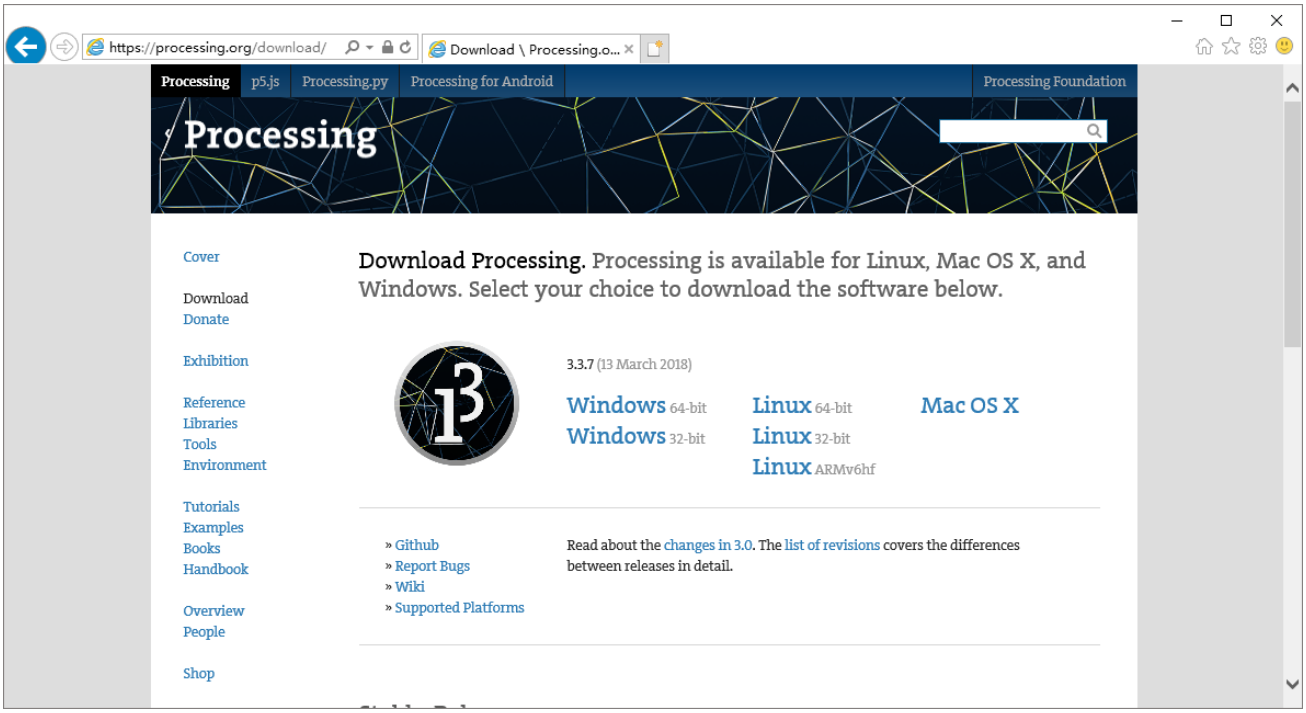
Processing Software

Processing software / Processing Development Environment (PDE) makes it easy to write Processing programs.

Please visit https://processing.org/, click "Download" to enter the download page.



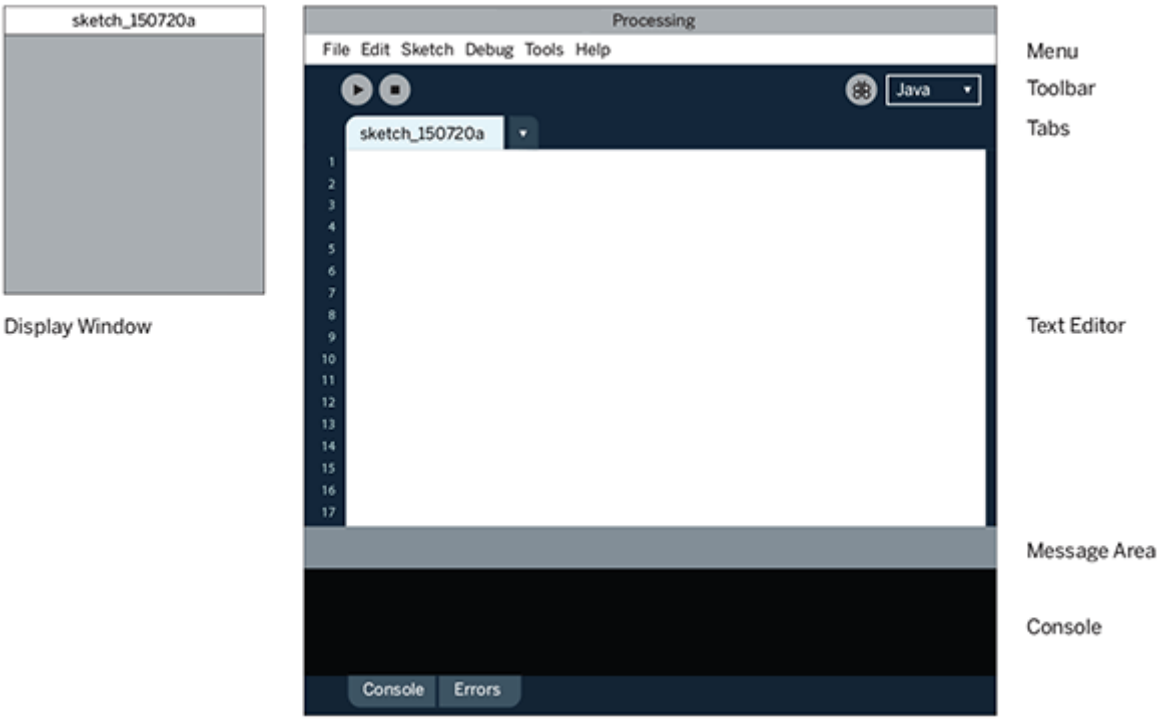
Download Processing for your computer.



Installation on each machine is straightforward:

- On Windows, you'll have a .zip file. Double-click it, and drag the folder inside to a location on your hard disk. It could be Program Files or simply the desktop, but the important thing is for the processing folder to be pulled out of that .zip file. Then double-click processing.exe to start.
- The Mac OS X version is also a .zip file. Double-click it and drag the Processing icon to the Applications folder. If you're using someone else's machine and can't modify the Applications folder, just drag the application to the desktop. Then double-click the Processing icon to start.
- The Linux version is a .tar.gz file, which should be familiar to most Linux users. Download the file to your home directory, then open a terminal window, and type:
  tar xvfz processing-xxxx.tgz
  (Replace xxxx with the rest of the file's name, which is the version number.) This will create a folder named processing-2.0 or something similar. Then change to that directory:
  cd processing-xxxx
  and run it:
  ./processing

With any luck, the main Processing window will now be visible. Everyone's setup is different, so if the program didn't start, or you're otherwise stuck, visit the troubleshooting page for possible solutions.



Programs written using Processing Software (PDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension **.pde**. It has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by Processing sketches including complete error messages and text output from sketches with the print() and println() functions. (Note that the console works well for occasional messages, but is not intended for high-speed, real-time output.)

The buttons on the toolbar can run and stop programs:

Run
Runs the sketch. In Java mode, it compiles the code and opens a new display window.

Stop
Terminates a running sketch.

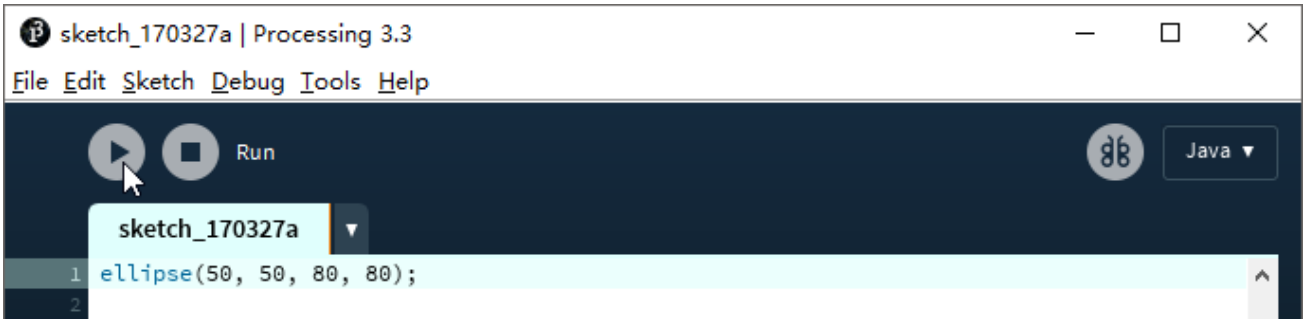
Additional commands are found within the six menus: File, Edit, Sketch, Debug, Tools, Help. The menus are context sensitive which means only those items relevant to the work currently being carried out are available.

First Use

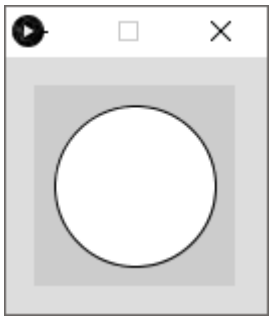
In the editor, type the following:

```
1  ellipse(50, 50, 80, 80);
```

This line of code means "draw an ellipse, with the center 50 pixels over from the left and 50 pixels down from the top, with a width and height of 80 pixels." Click the Run button (the triangle button in the Toolbar).

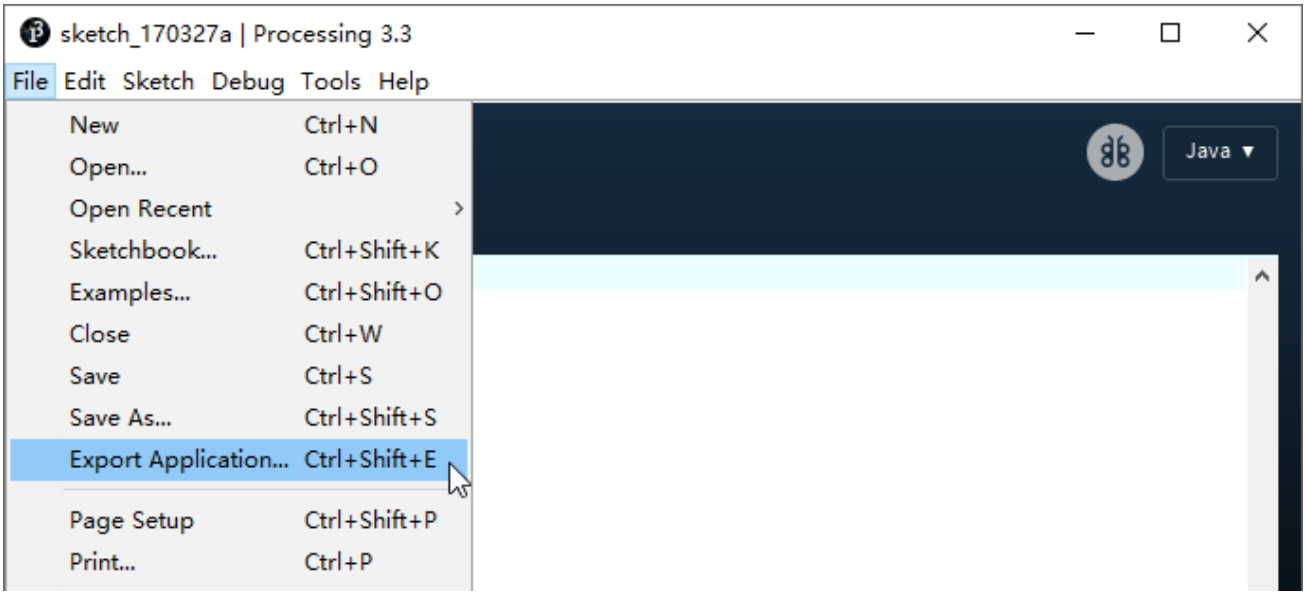


If you've typed everything correctly, you'll see a circle on your screen.



If you didn't type it correctly, the Message Area will turn red and complain about an error. If this happens, make sure that you've copied the example code exactly: the numbers should be contained within parentheses and have commas between each of them, and the line should end with a semicolon.

You can export Processing sketch to an application to run it directly without opening the Processing.
To export the sketch to the application, you must save it first.



So far, we have completed the first use. I believe you have felt the joy of Processing.

Assembly

Now let us start assembling the robot.

! Please prepare the right batteries and fully charge them before assembling.
Assembling without right batteries will cause installation errors, which may damage the servos.
Refer to “AboutBattery.pdf” for detailed information about battery.

! Please follow the tutorial strictly and do not skip any steps.
Ask our support for help if you encounter a problem instead of ignoring it.

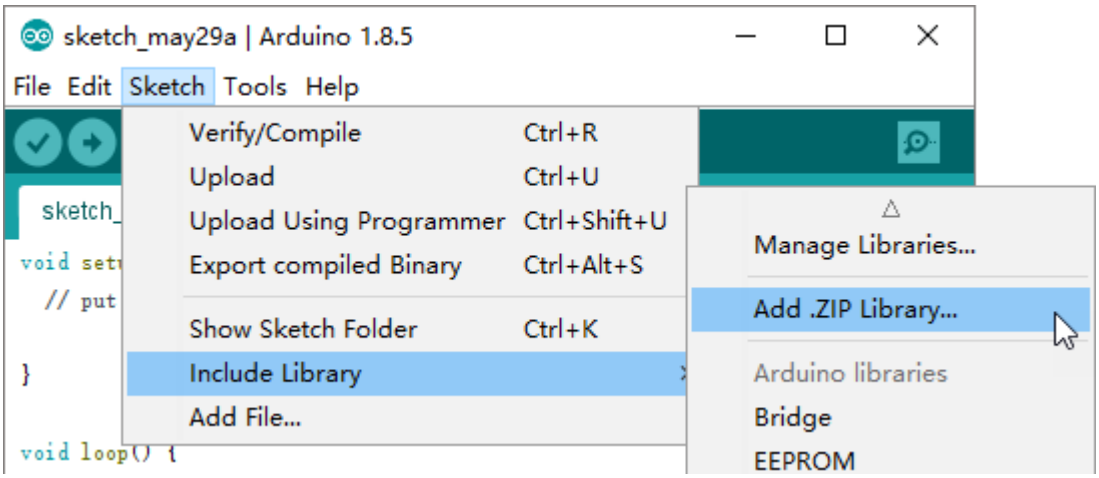
Step 01

First, we need to upload the default sketch to the control board (Freenove Crawling Robot Controller).

We suggest you read the previous “Arduino” chapter if you are not familiar with Arduino board and software.

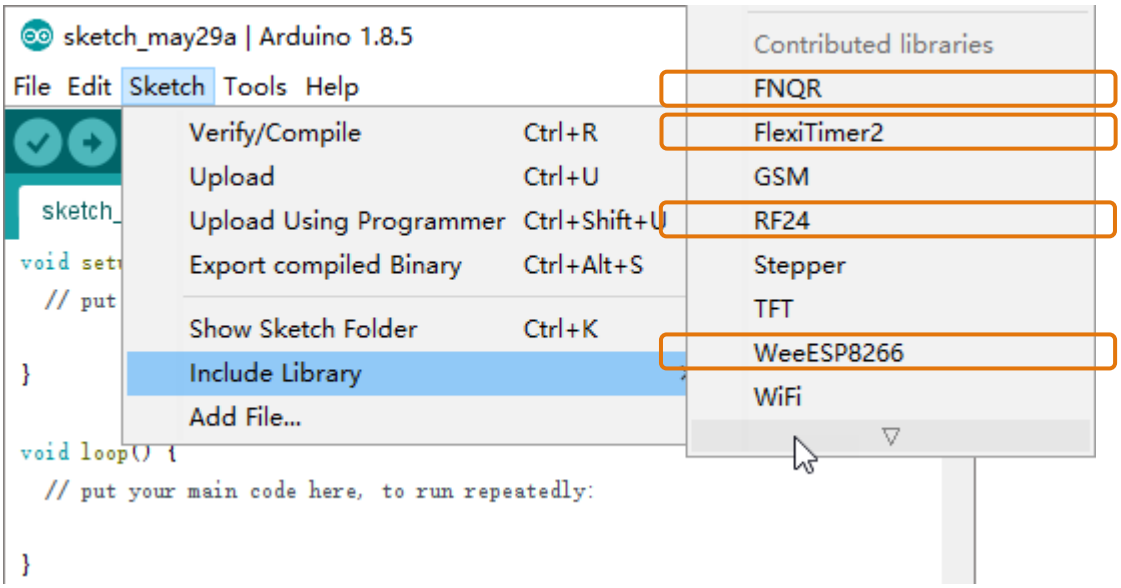
Libraries are collections of code that make programming simple.
The "FNQR" (Freenove Quadruped Robot) library is used to control this robot. We need to add it and other necessary libraries to Arduino software.

Open Arduino software, click “Sketch” > “Include Library” > “Add .ZIP Library...” to add a library file.

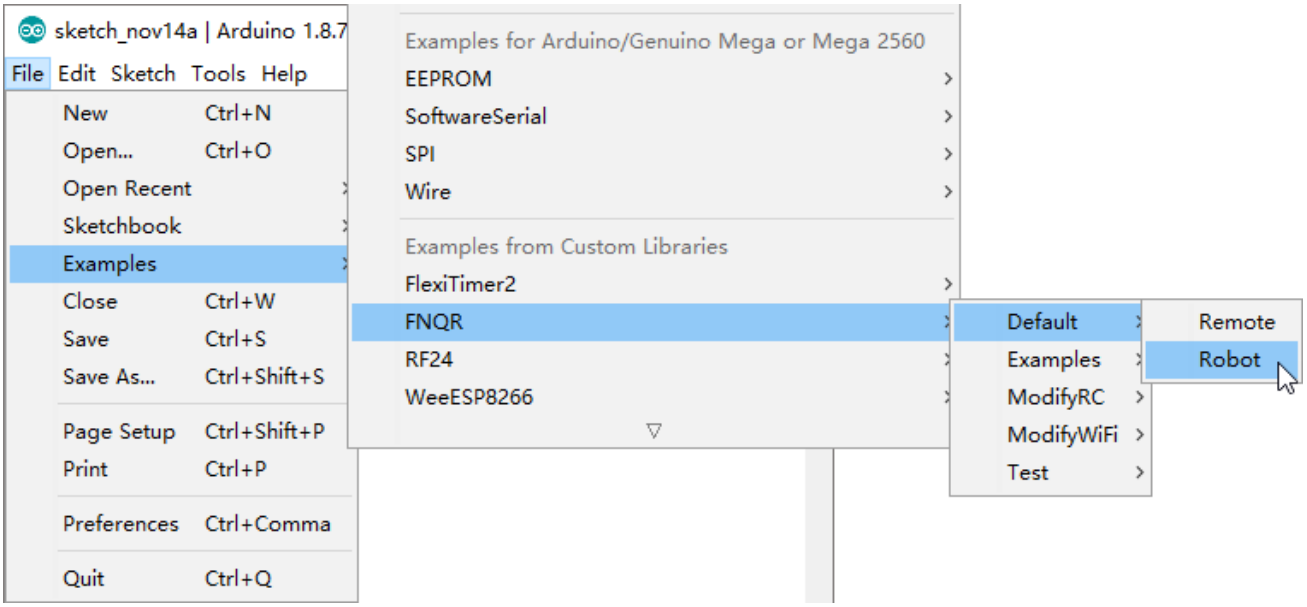


Add all library files under “ArduinoLibraries” folder. This folder is in the same folder contains this PDF tutorial.

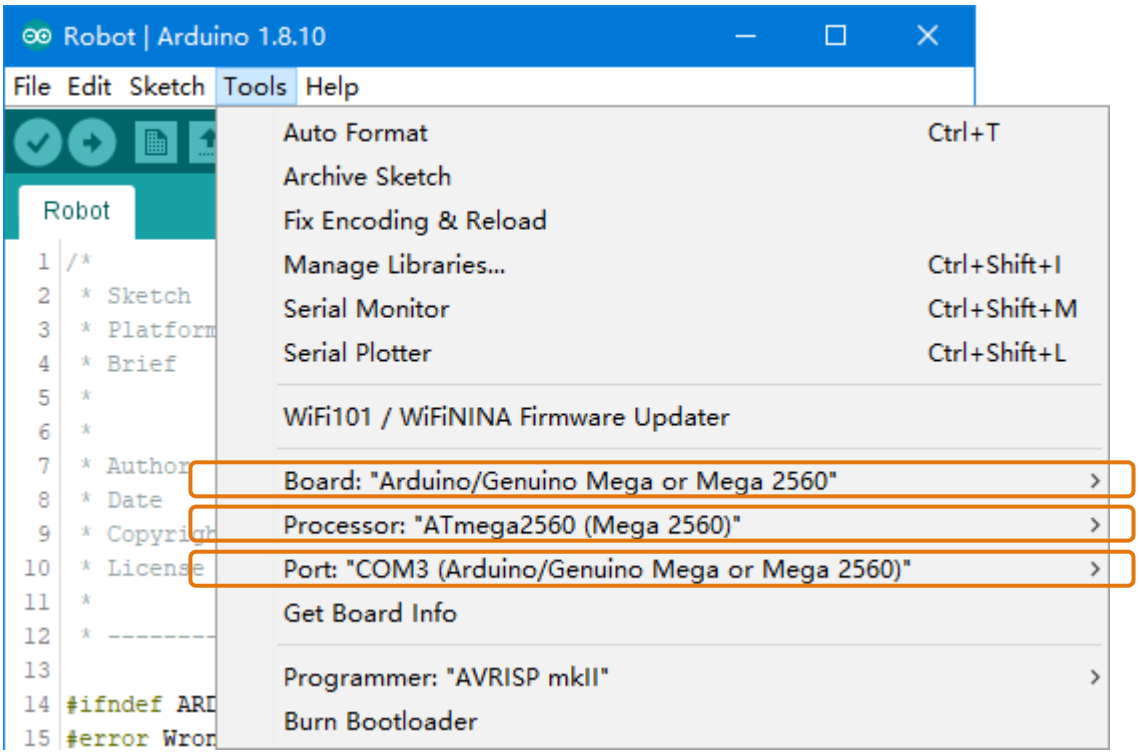
You can find the added libraries in “Sketch” > “Include Library”. Make sure all the following libraries have been added.



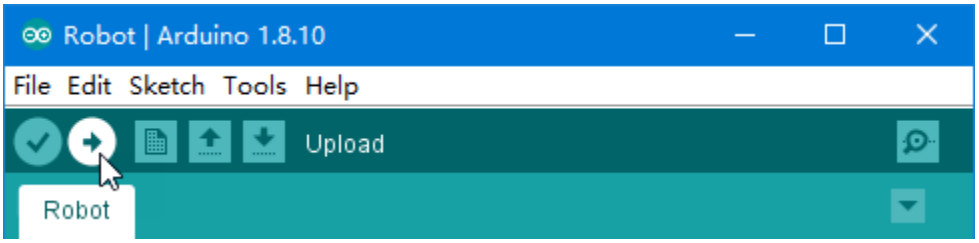
Now open "File" > "Examples" > "FNQR" > "Default" > "Robot".



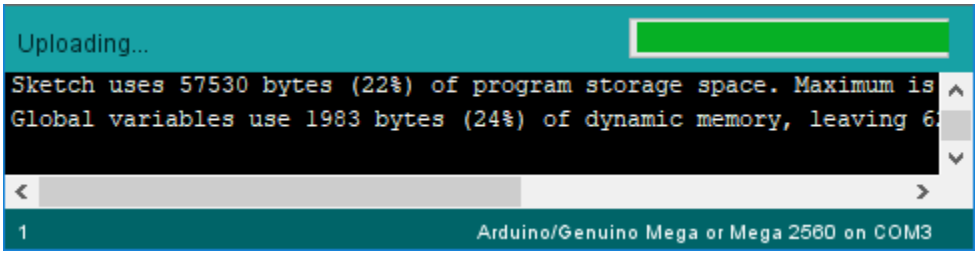
Select the right Board, Processor and Port (Your COM number may be different from the following figure).



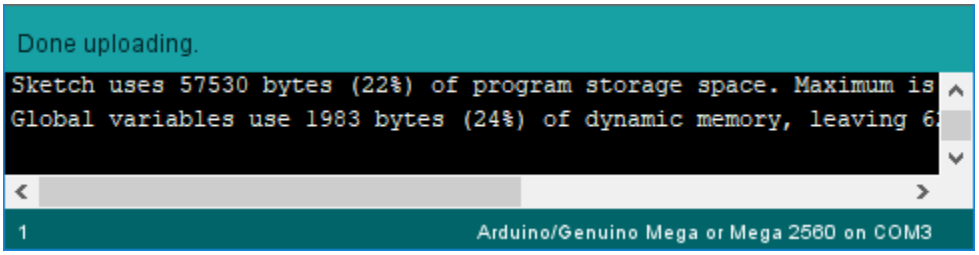
Then click “Upload” button to upload the default sketch to the control board.



Wait for uploading. It will take a few seconds.



Once you see "Done uploading", it means the upload is successful.



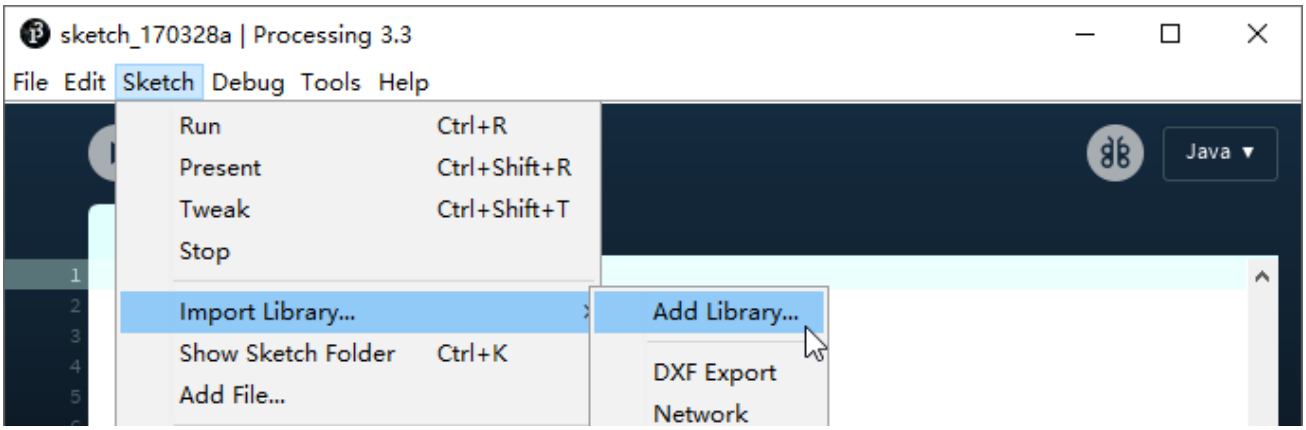
Step 02

Now we need to run a Processing sketch. It will allow us to set the control board.

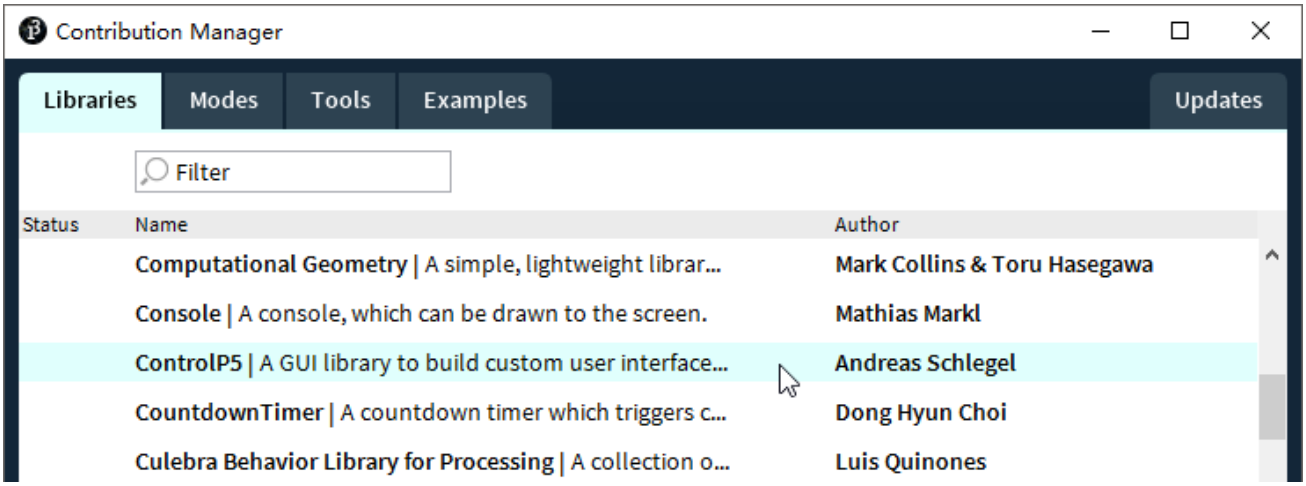
We suggest you read the previous “Processing” chapter if you are not familiar with Processing software.

We also need to add library to Processing software.

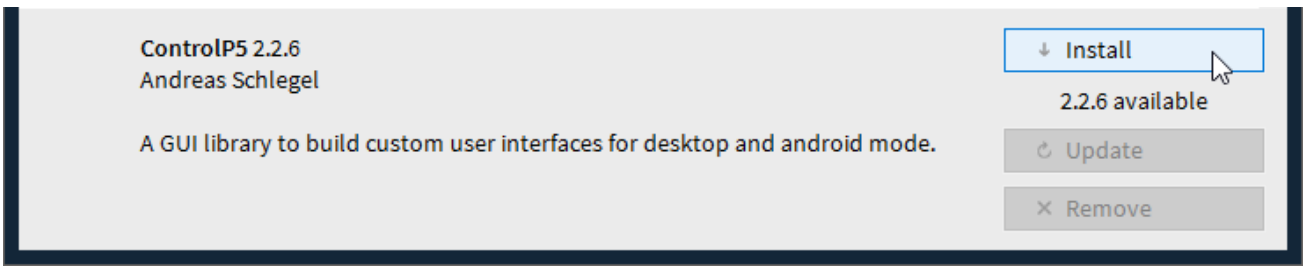
We can add library online. Click "Add Library..." to open "Contribution Manager".



Find "ControlP5" in the "Libraries" tab, and click to select it.



Then click “Install”.

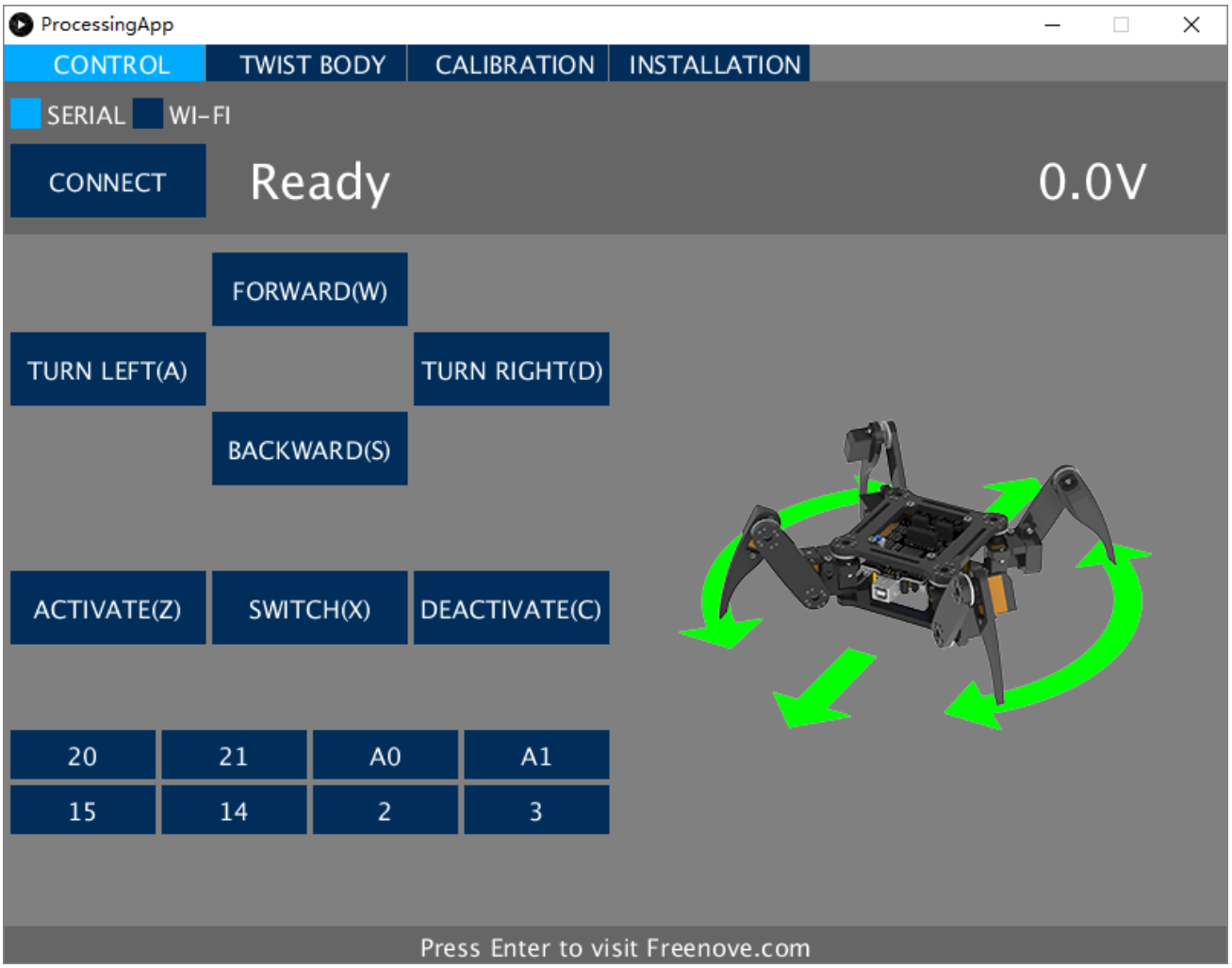


After the installation succeeds, close “Contribution Manager” window.

If you cannot complete the online installation, you can install it offline.
Just unzip “ProcessingLibraries\controlP5.zip” to "libraries" folder under "Sketchbook location" in the “Processing software” > "File" > "Preferences" window. You will need to create the "libraries" folder if this is your first add library. After that, restart Processing software.

After library files are installed, open "ProcessingApp\ProcessingApp.pde" with Processing software.

Click “Run”, then the Processing App window appears.



Now let us try it. Connect the control board to your computer via USB cable.

Select "SERIAL" in Processing App, then click "CONNECT".

Wait for processing. It may take a few seconds. Do not click again when it's processing.

When the "CONNECT" button changes to "DISCONNECT", it means the connection is successful.

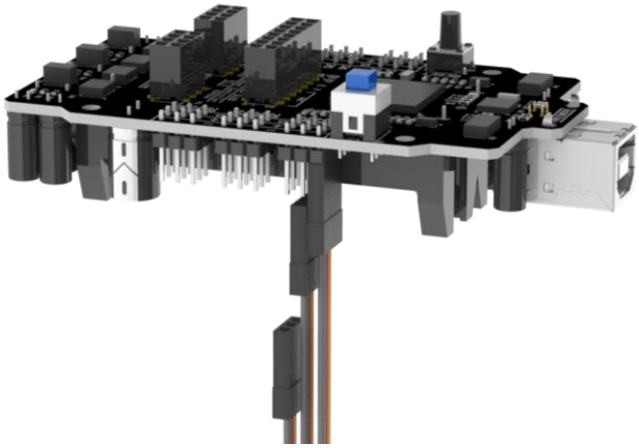
! If the connection is unsuccessful, the default sketch may not have been uploaded successfully.
Upload the default sketch to the control board and then try again.

Click "DISCONNECT" button to disconnect.

Step 03

Now we will make a test and then set the control board to installation mode.

Take out all the servos and connect them to the control board (yellow wire of servo to S, red to +, black to -). Servos can be connected randomly to port 22~27 and 34~39 (do NOT connect to 28~33).
(The name of the port is marked on the front of the control board.)



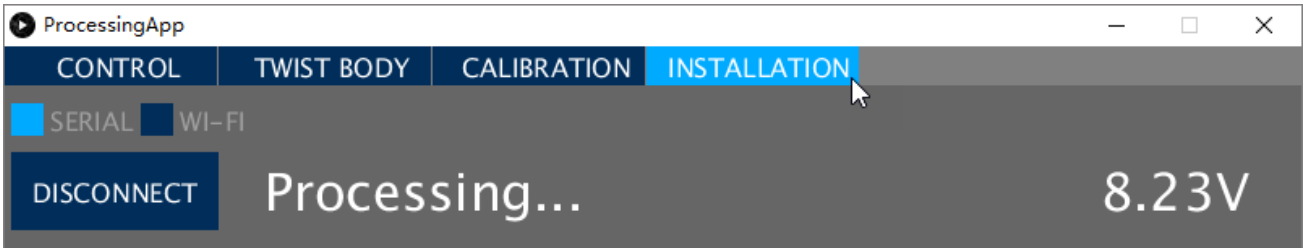
Then install your batteries and keep power off.

Connect the control board to computer via USB cable. Then open Processing App and click “CONNECT”. After the connection succeeds, turn on the power. The servos will rotate and then stop. The battery voltage will be shown on the Processing App.

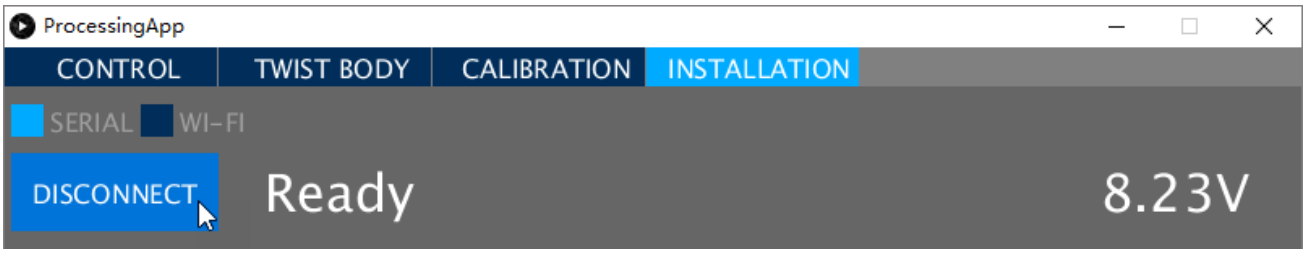
! The battery voltage should be around 8.0~8.4V.

If the voltage is less than 8V, please charge the battery and then try again. If you get the same result, then you may have wrong battery. Please check the type of battery or whether there is a protective board. Refer to “AboutBattery.pdf” for detailed information about battery.

If the voltage is no problem, then we need to set the control board to installation mode.
Click "INSTALLATION" on the top. Wait for processing and the servos will rotate again.



Click “DISCONNECT” to disconnect. The control board will restart and the servos will rotate again.



! The LED "L" on the control board now should flash three times every short time.

It is indicating that the control board is under installation mode.

If it is not, you need to use Processing App to set it again.

You have to do this step right. Otherwise, will cause installation errors and damage the servos.

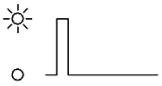
If you meet problems, check your batteries or ask our support team for help.

! Switch the power off. Disconnect the USB cable, remove the batteries and disconnect all the servos.

Now let us know the signal LED (LED “L” on the control board). It can indicate current state of the robot. You should always observe it, which is very important and useful. See “SignalLED.mp4” in “Videos” folder.

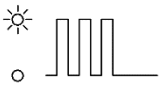
The signal LED will flash several times every few seconds (cycle).

In each cycle, if the LED lights up several times then stays off. It is indicating different working mode:



Blinks once: ready mode.
This mode should be set after completing installation and calibration of the robot.

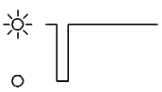
Blinks twice: calibration mode.
This mode should be set when calibrating the robot.



Blinks 3 times: installation mode.
This mode should be set when installing the robot.

The working mode of the robot can be set by Processing App. The robot will remember the working mode you set, even if you restart the power, the robot will still enter the mode you set originally.

In each cycle, if the LED goes off several times then stays on. It is indicating error state:
(Only applicable to V2.0 and later versions control board)



Blinks once: power error.
It indicates the voltage of battery is too low and need to be charged.

The robot will stop all movements and cut off power of all servos when there is an error. You must first solve it to continue usage.

Step 04

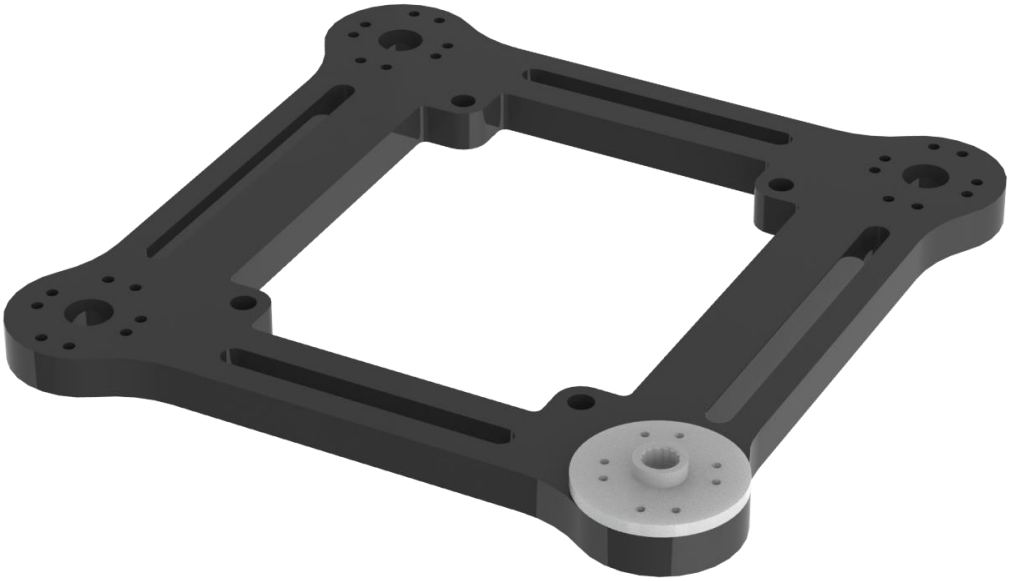
Now let us assemble the robot.

! Make sure the battery is correct and the control board has been set to the installation mode.

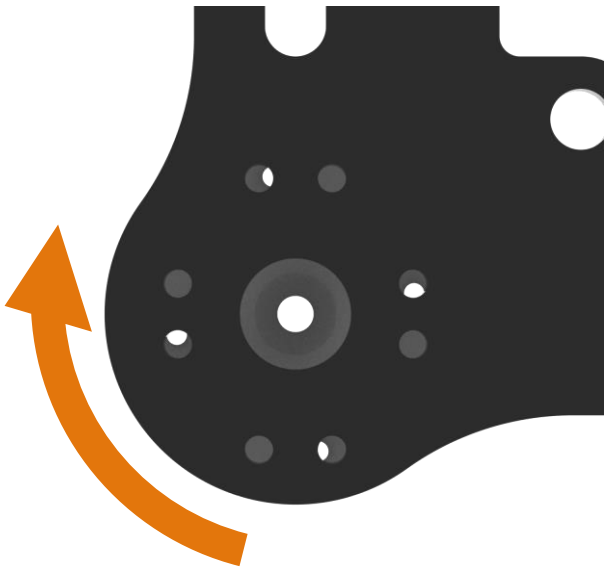
! Please assemble and use the robot on a smooth surface such as desktop.
Assembling and using the robot on rough surfaces such as carpets will damage the servos.

Place disc servo arm on acrylic plate.
(Disk servo arm and servo are packed together.)

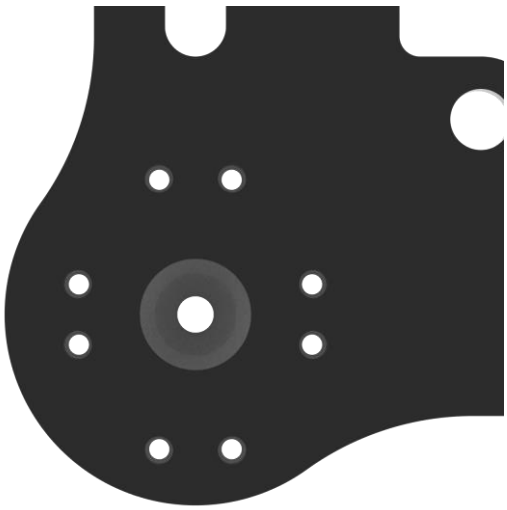
Right location is as below.



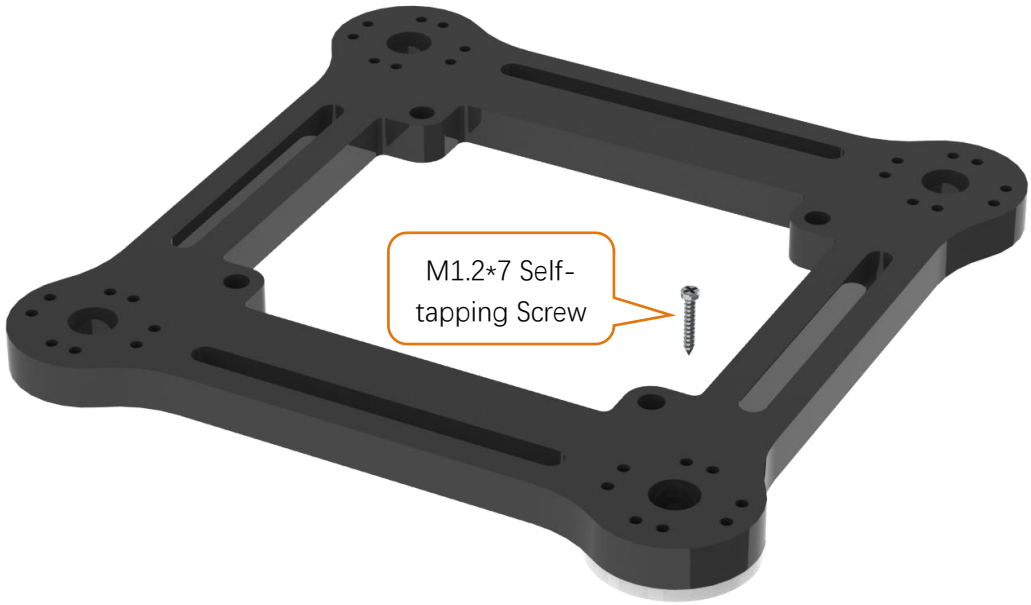
Rotate disc servo arm so that its holes can aligned with the holes in the acrylic plate.



When all holes align:



Use the screw to fix disc servo arm to acrylic plate.



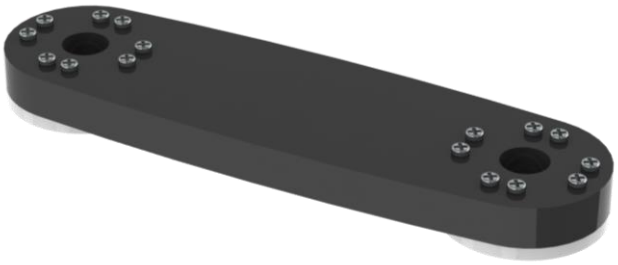
Tighten the screws as below.

Use the same screws to fix other holes of disc servo arm.

Use the same screws to fix 3 other disc servo arms to acrylic plate.

Step 05

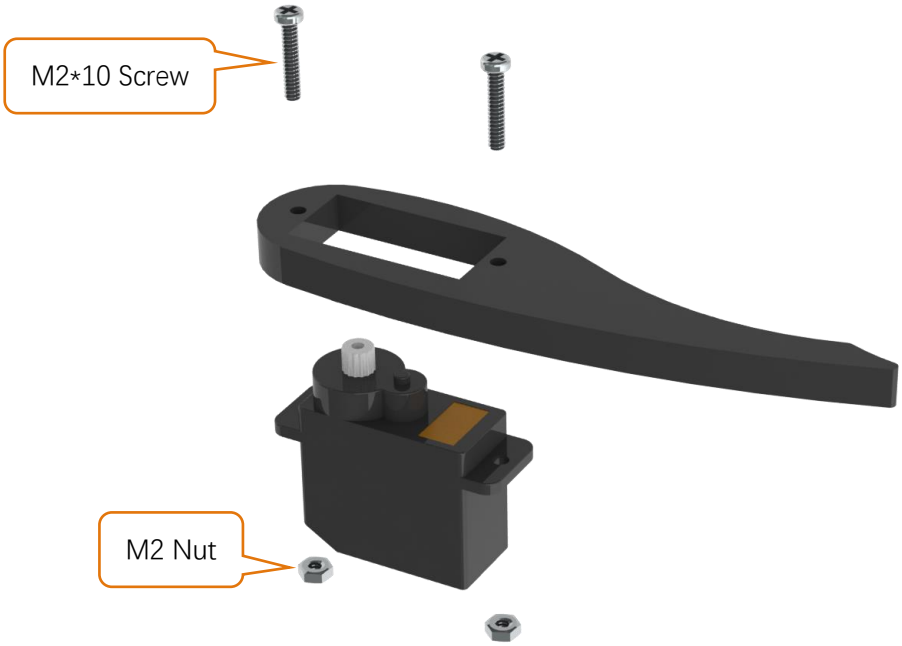
Use the same screws to fix 2 disc servo arms to following acrylic plate.



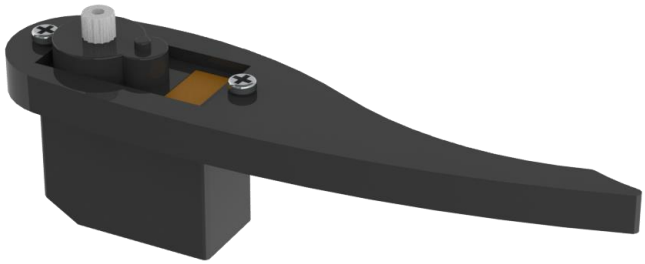
Assemble 3 other acrylic plates as below.

Step 06

Use screws and nuts to fix servo to following acrylic plate.



Fix them as below. Note the position of the servo shaft.

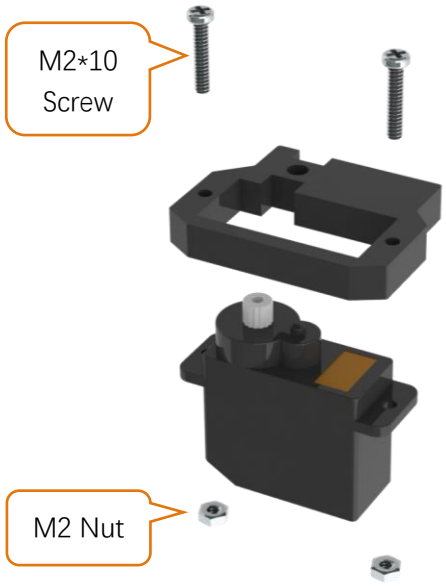


Assemble 3 other acrylic plates.

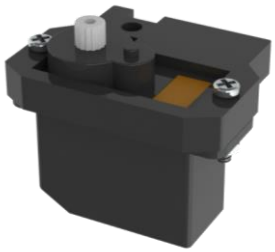
Note the direction of acrylic plates. Two of them are opposite to the other two.

Step 07

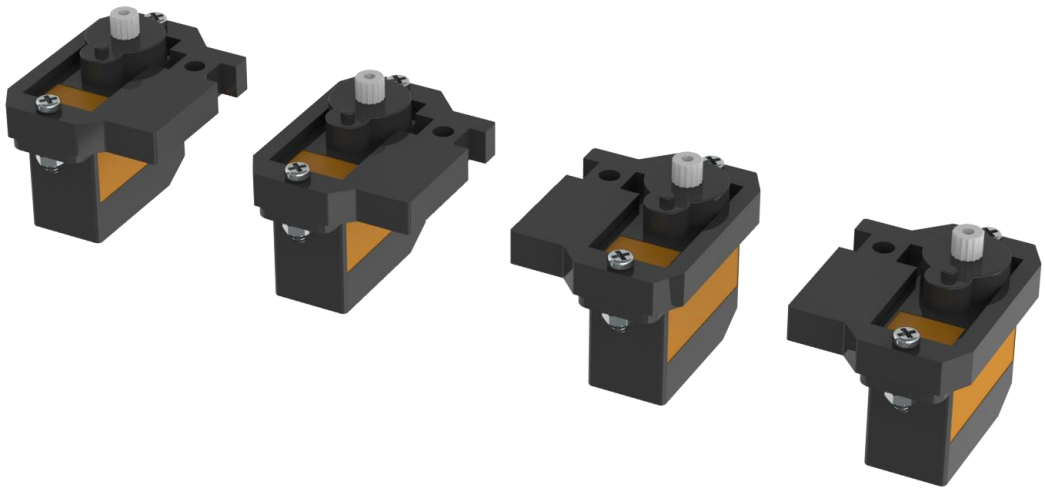
Use screws and nuts to fix servo to following acrylic plate.



Fix them as below. Note the position of the servo shaft.

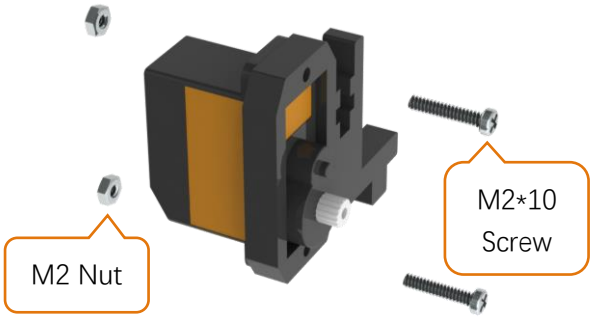


Assemble 3 other acrylic plates.
Note the direction of acrylic plates. Two of them are opposite to the other two.

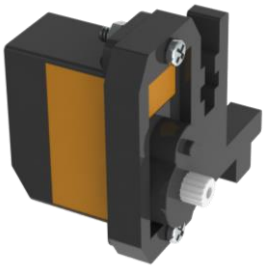


Step 08

Use screws and nuts to fix servo to following acrylic plate.

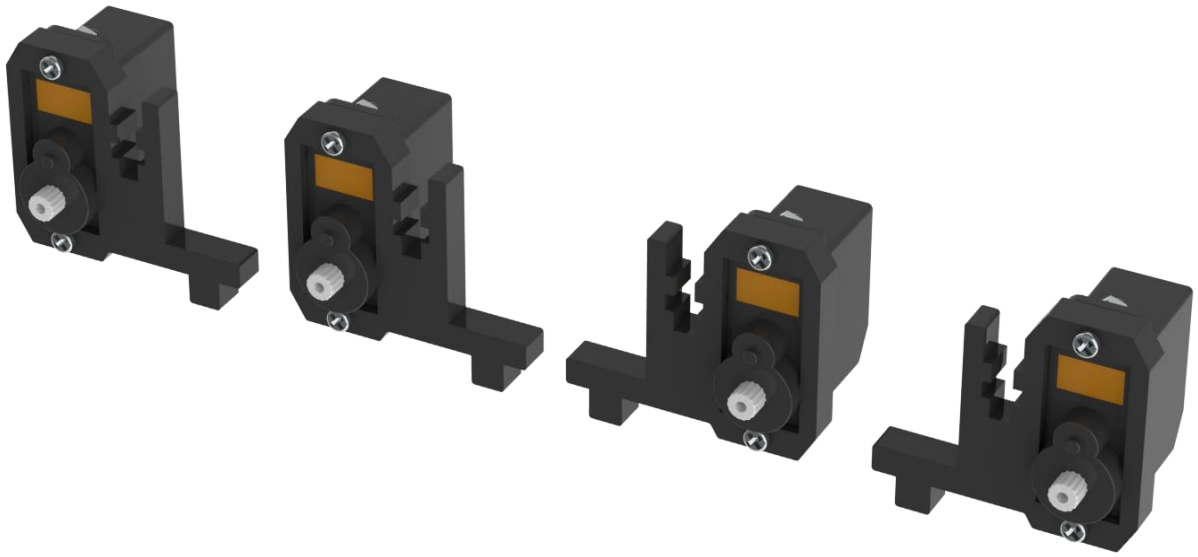


Fix them as below. Note the position of the servo shaft.

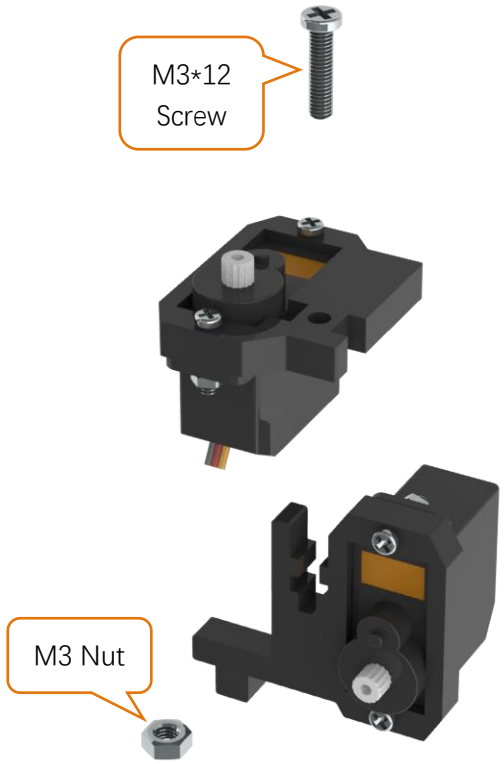


Assemble 3 other acrylic plates.

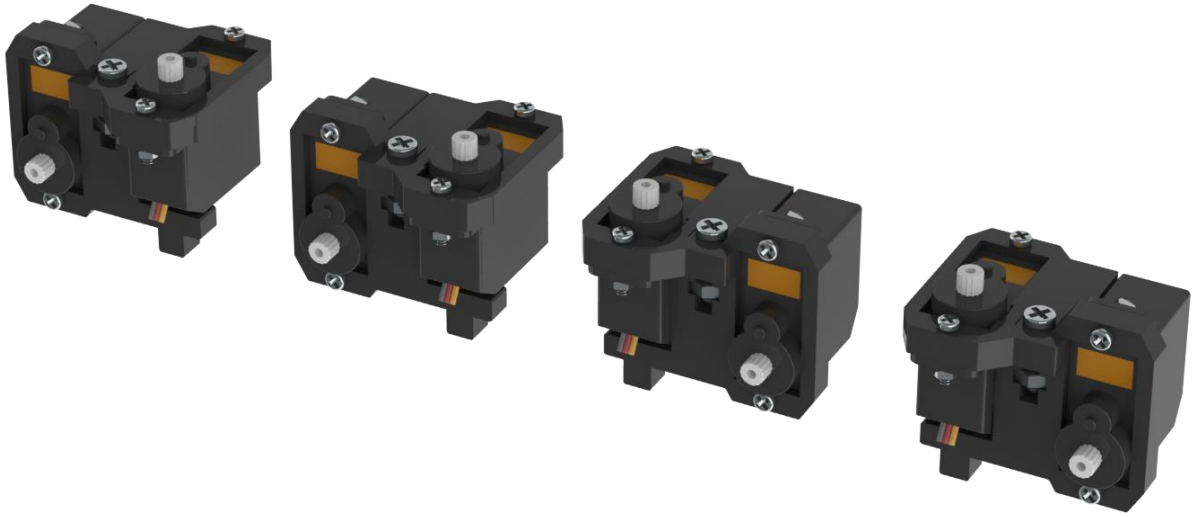
Note the direction of acrylic plates. Two of them are opposite to the other two.



Step 09

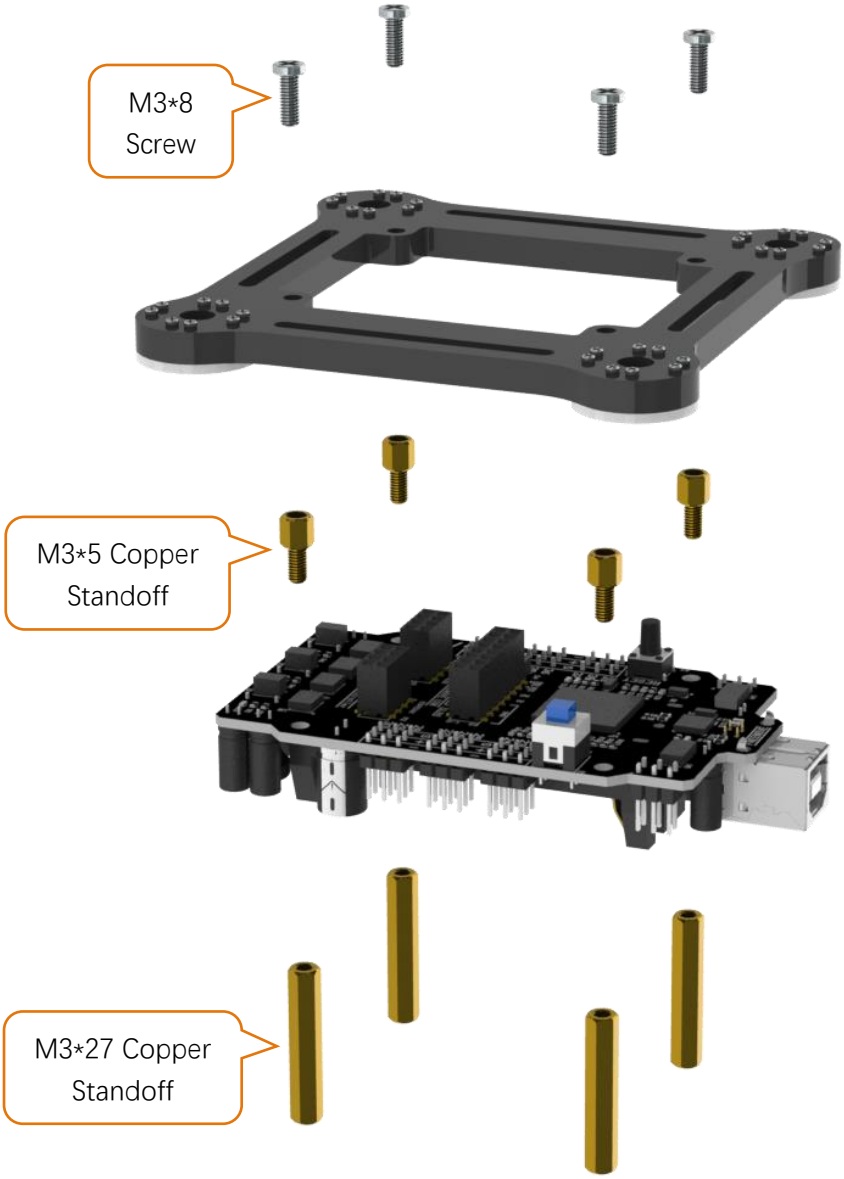
Use screws and nuts to fix two parts assembled before. 	Fix them as below. Two set of them need to be assembled. 
Use screws and nuts to fix two parts assembled before. 	Fix them as below. Two set of them need to be assembled. 

After the assembly is completed, 4 following components are obtained.
Note that two of them are different from the other two.

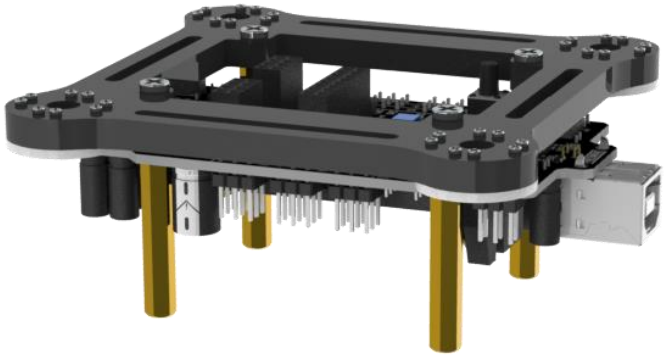


Step 10

Use screws and copper standoffs to fix control board to following part assembled before.



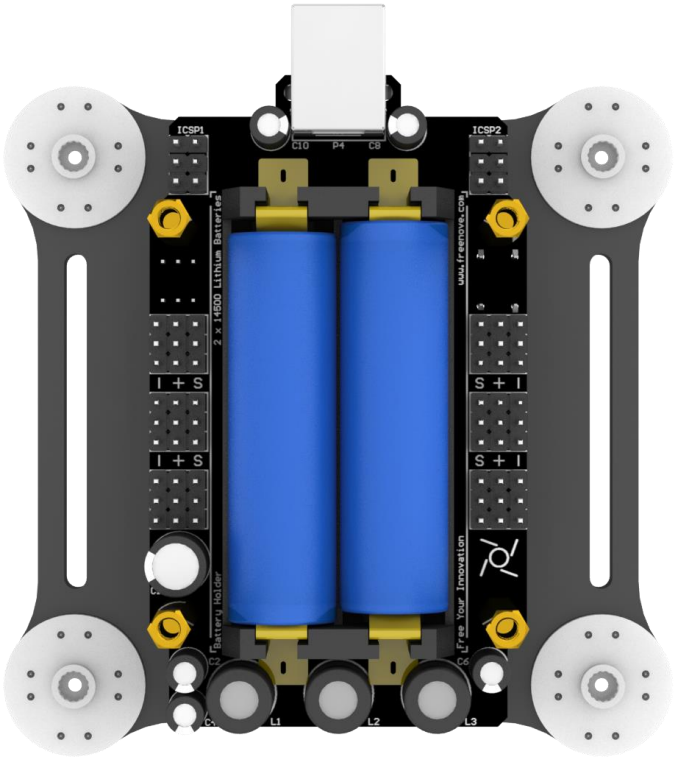
Fix them as below.



Install batteries for the control board.

! You have to install the right batteries that are full charged. (Refer to "AboutBattery.pdf")

Assembling the robot without right batteries will cause installation errors and damage the servos.

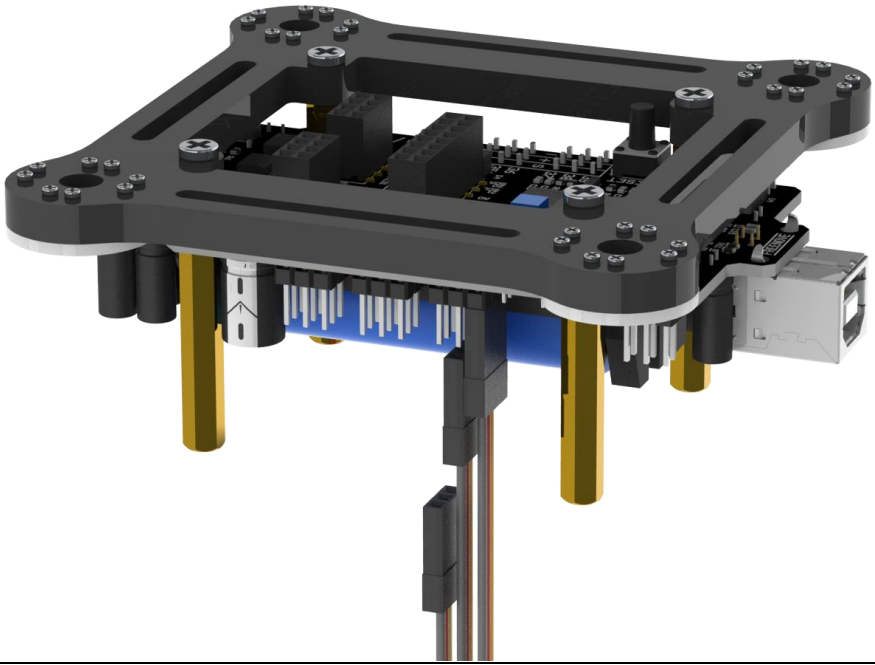


Make sure the power is turned off, and then connect all servos to control board.

(yellow wire of servo to S, red to +, black to -)

(servos can be connected randomly to port 22~27 and 34~39, do NOT connect to 28~33)

(The name of the port is marked on the front of the control board.)



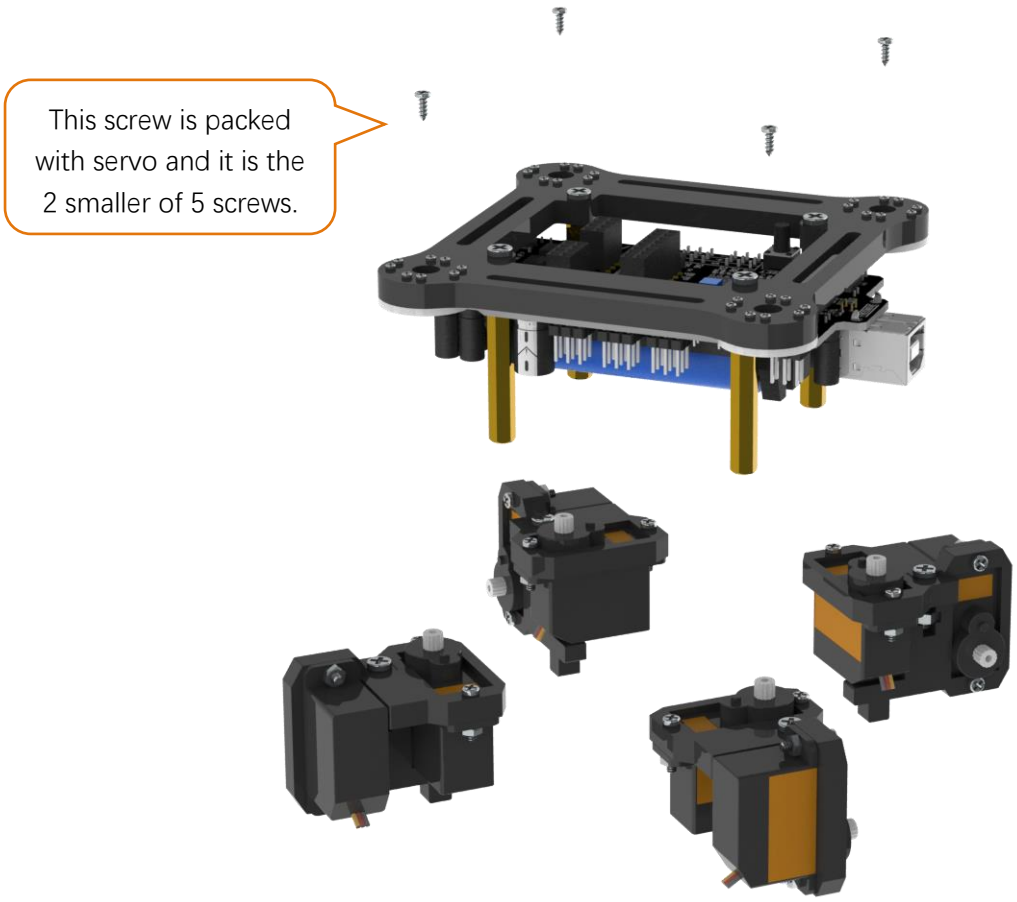
! Keep all servos connect to the control board, unless requested to disconnect.

Step 11

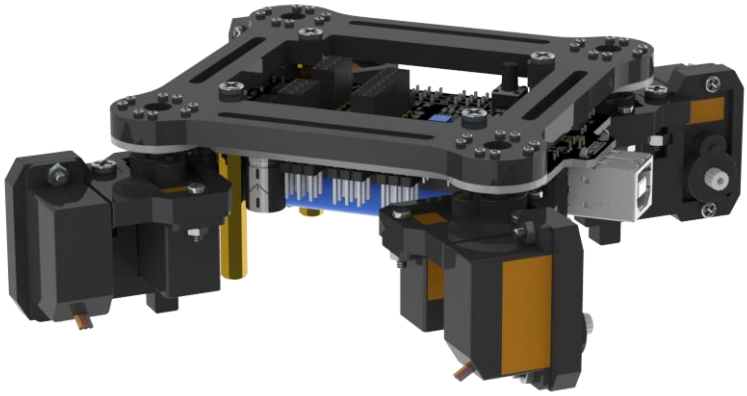
Turn on the power. The servos will rotate and then stop.

! Keep power on unless it is required to turn off.
! The wires of servos are not shown in the later steps.

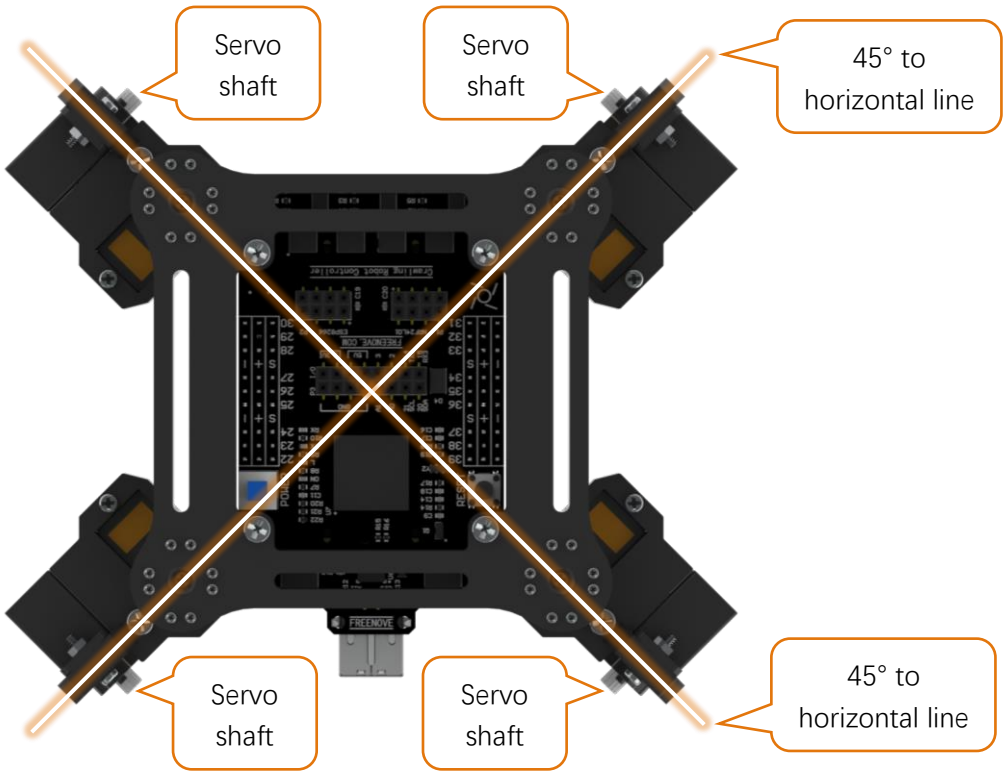
Use screws to fix 4 parts assembled before. Please note the installation angle of 4 parts. (Refer to assembly result later)



Fix them as below.

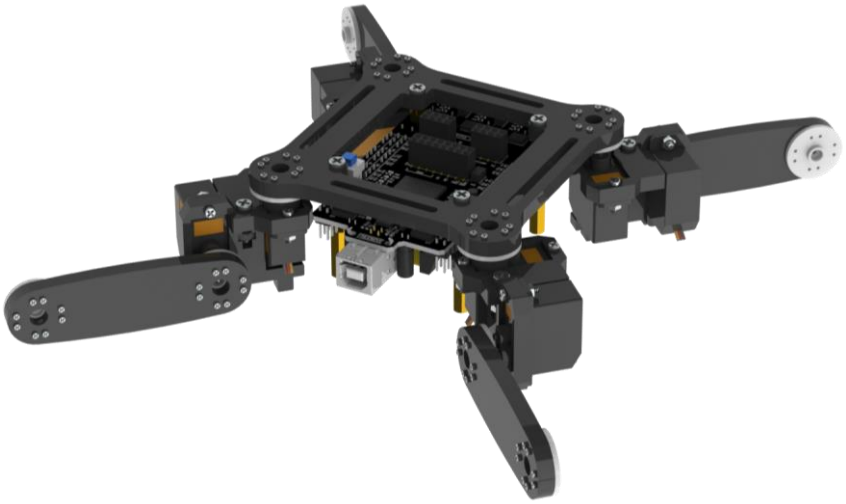


After the installation is completed, the angle of 4 parts should be shown as below. Please try to approach the angle. A small deviation is acceptable. We will correct the deviation in later calibration step.

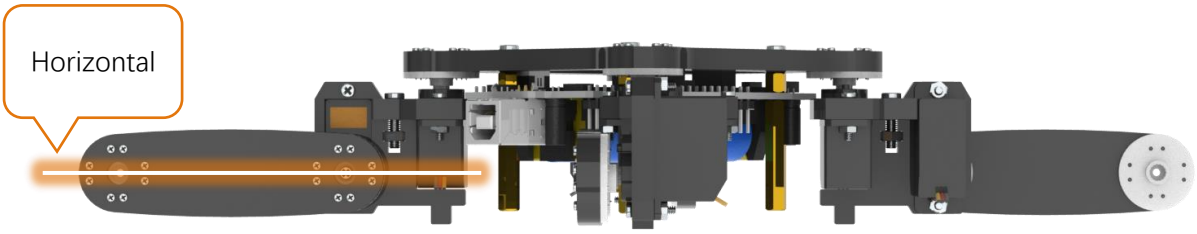


Step 12

Use the same screws to fix 4 parts assembled before. Please note the installation angle of 4 parts. (Refer to assembly result later)



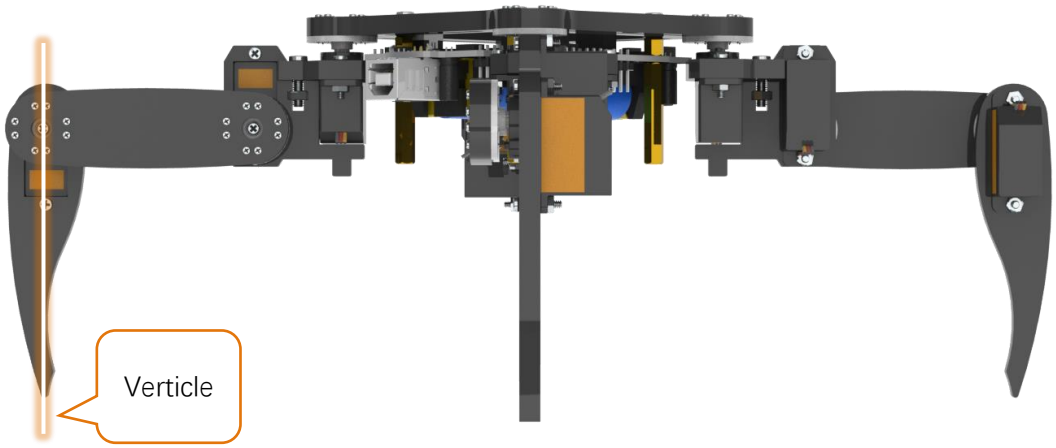
After the installation is completed, the angle of 4 parts should be shown as below. Please try to approach the angle. A small deviation is acceptable. We will correct the deviation in later calibration step.



Step 13

Use the same screws to fix 4 parts assembled before. Please note the installation angle of 4 parts. (Refer to assembly result later).

After the installation is completed, the angle of 4 parts should be shown as below. Please try to approach the angle. A small deviation is acceptable. We will correct the deviation in later calibration step.



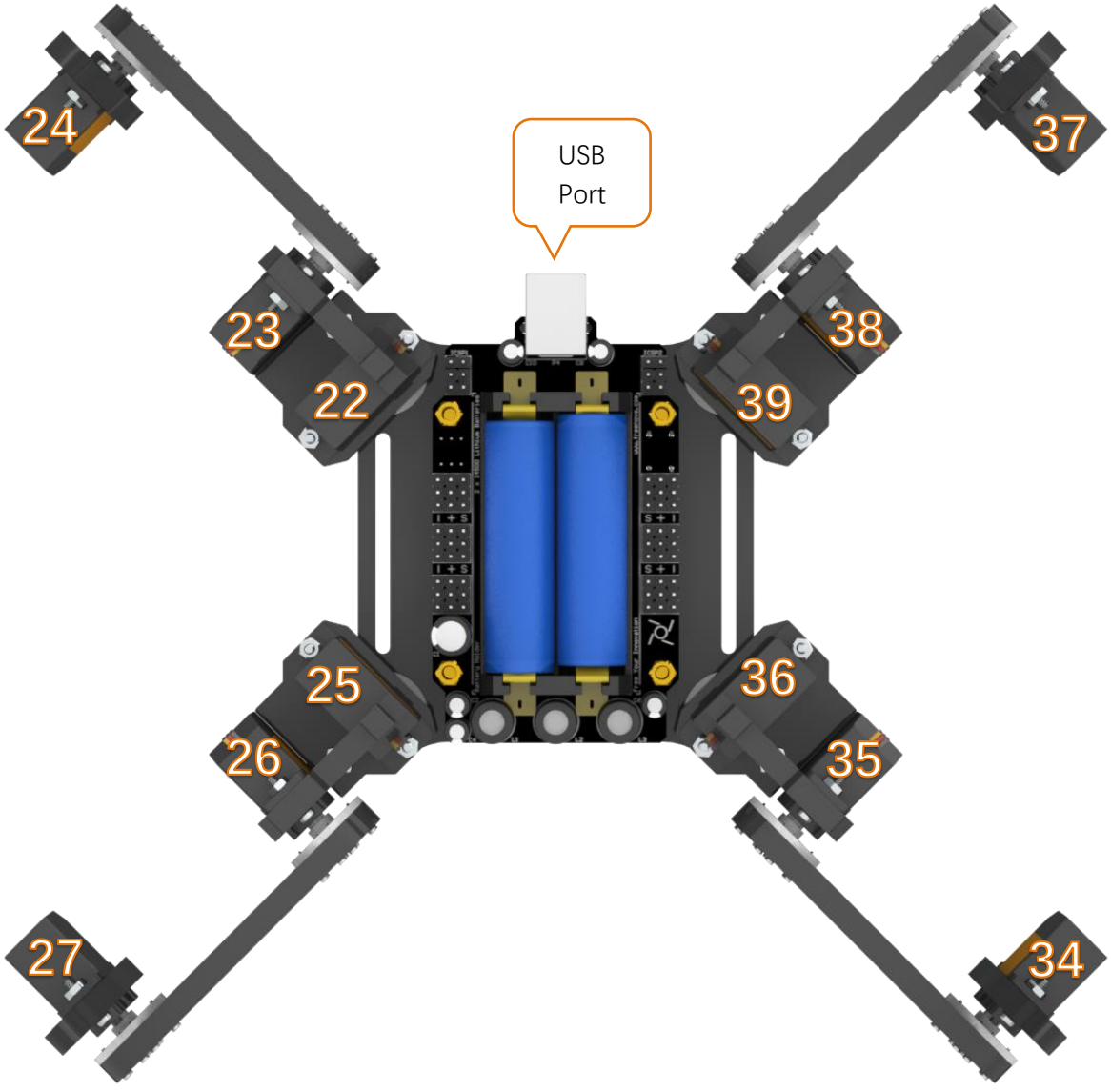
Step 14

! Turn off the power and pull off all the wires of servos.

Then reconnect the servos to control board, the ports they should connect to are shown below.

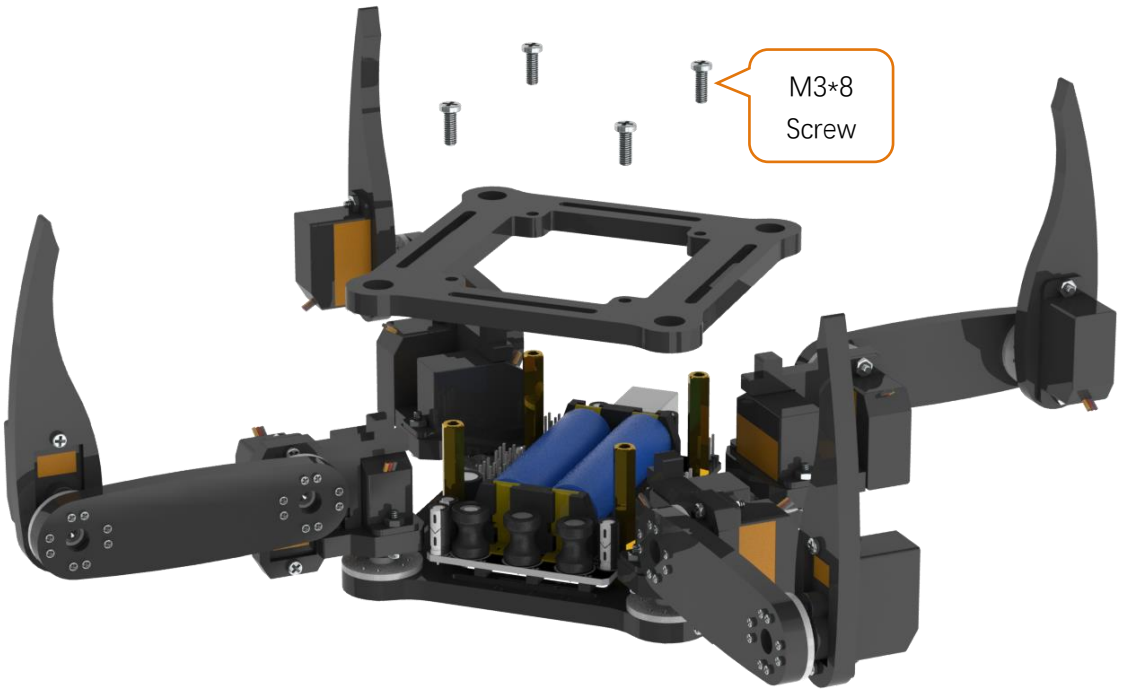
! The view of the picture below is from the bottom of the robot (you can see the battery).

! The name of the port is marked on the front of the control board.

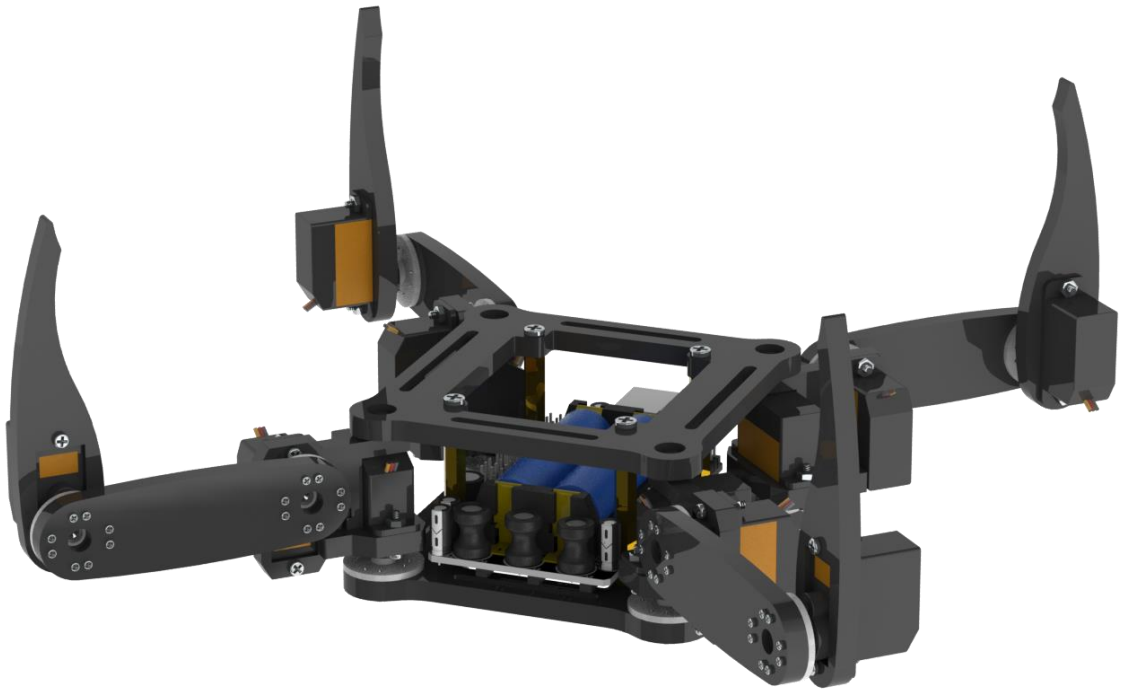


Step 15

Use screws to fix the folowing acrylic plate.

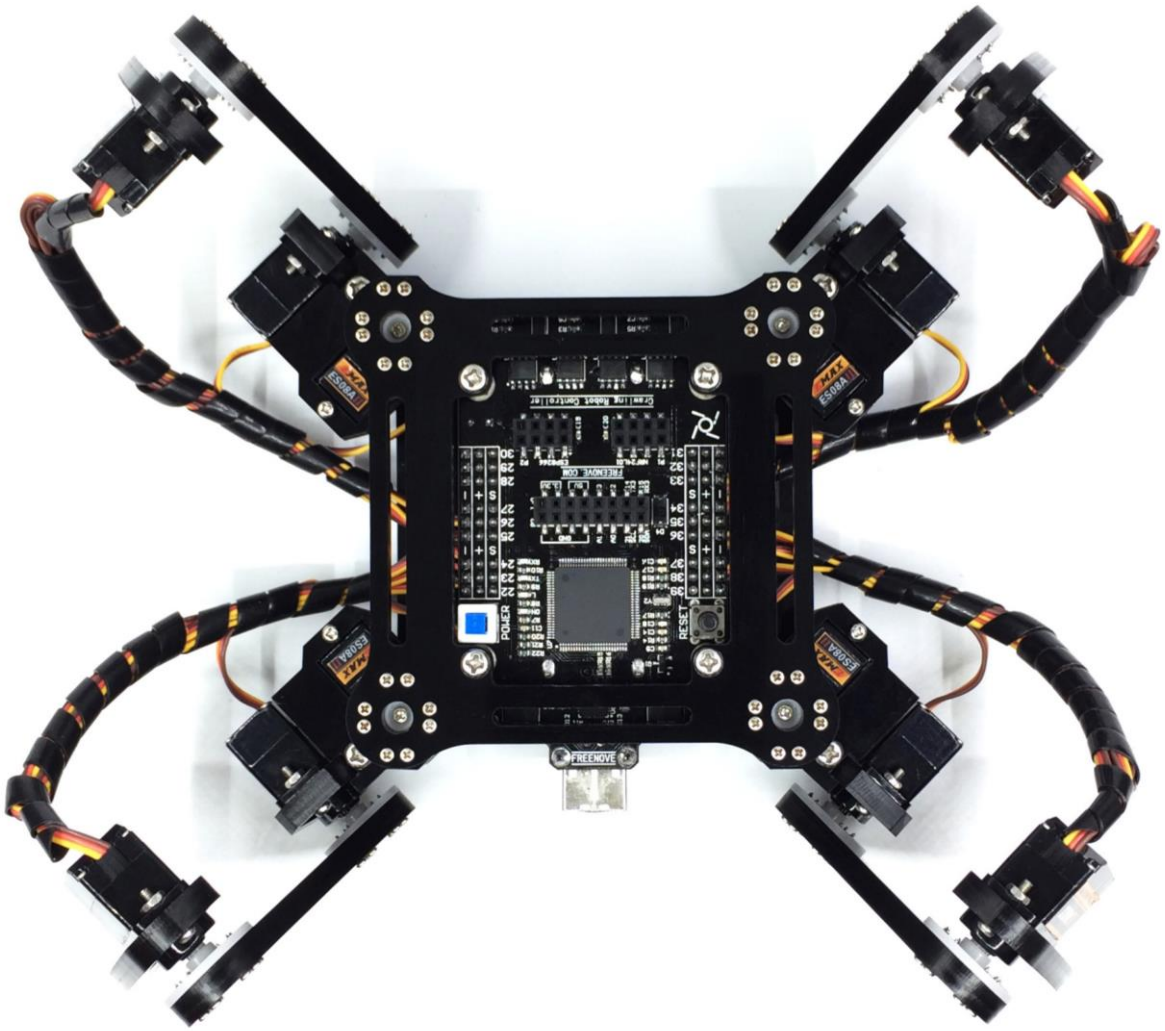


Fix it as below.



Use cable tidy to arrange the wires of servos.

Do not wrap the wires too tight, so that the servos are free to move.



Step 16

Fix ESP8266 module to control board.

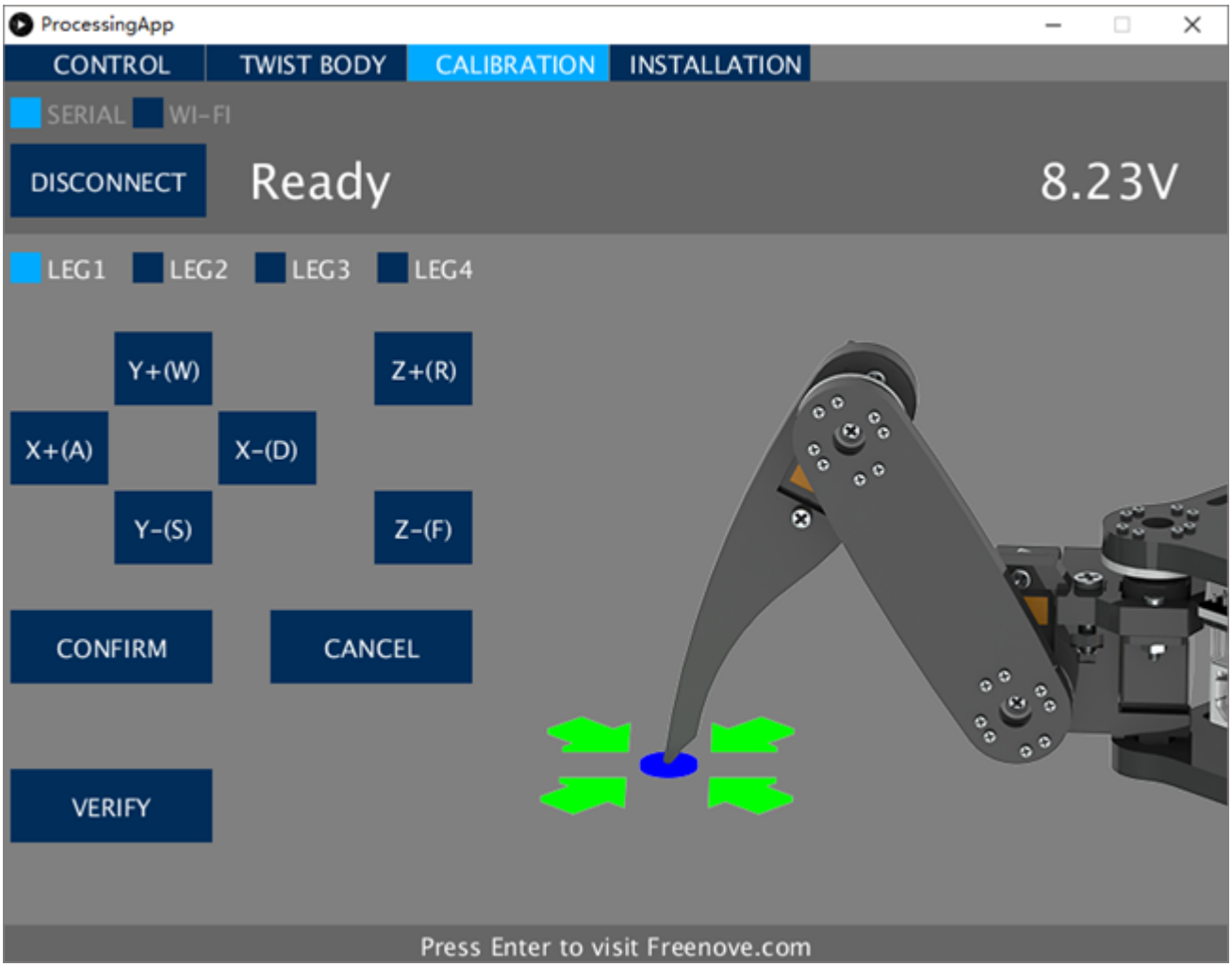
Fix it as below.

Step 17

At the end of the assembly, let us calibrate the servos.

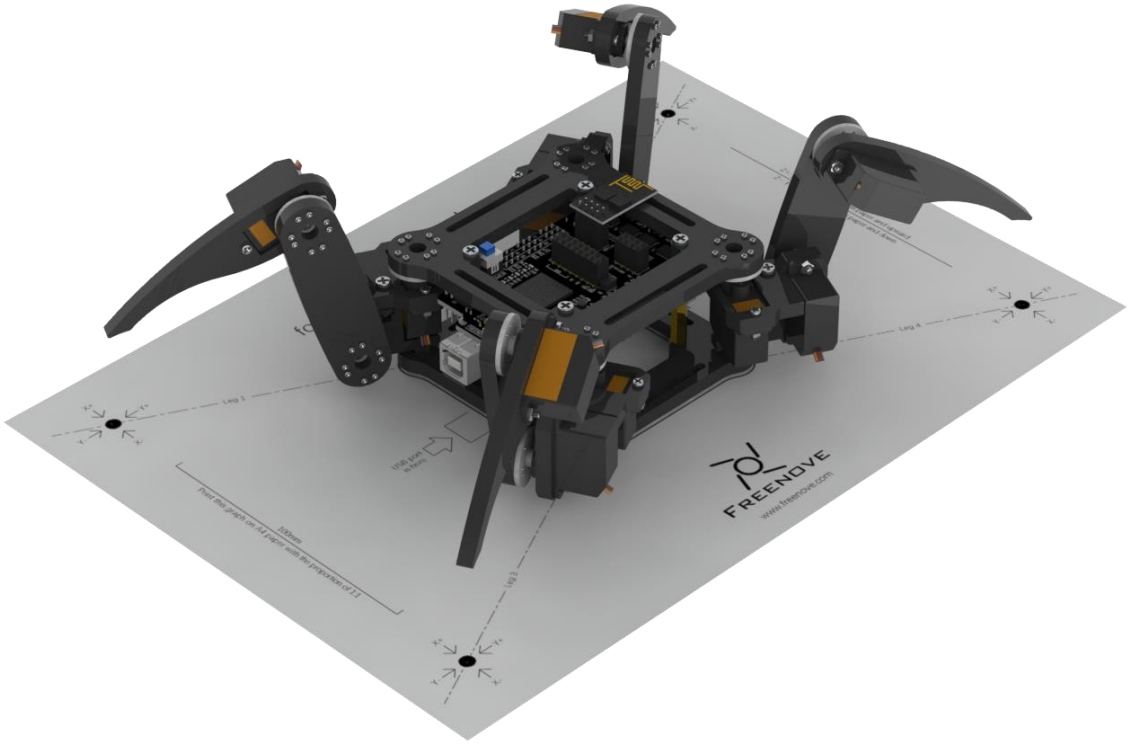
Keep the power off.

Then connect the robot with Processing App. After the connection succeeds, click "CALIBRATION" on the top.



Turn on the power. The robot will move to following posture, indicating it is already in calibration mode.

Put the robot on the calibration graph. The bottom of the robot should coincide with the specific outline in the graph. USB interface should also coincide with the mark in the graph.

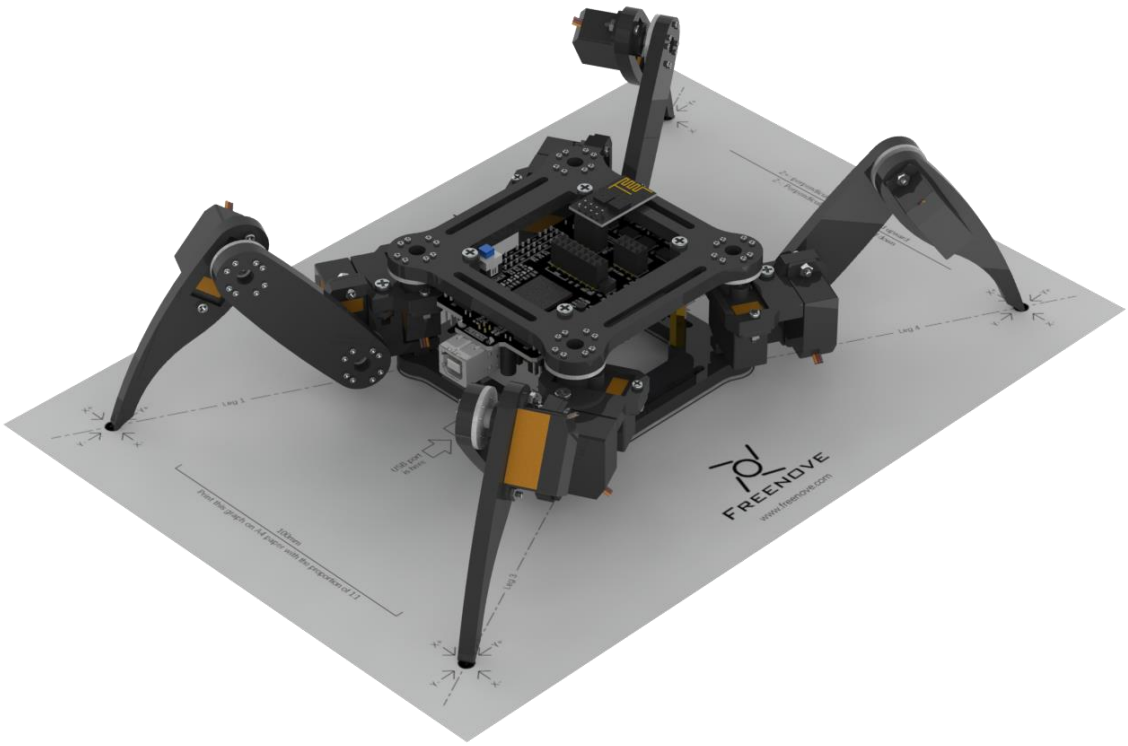


If the calibration graph is missing or damaged, you can print a copy. Print the "CalibrationGraph.pdf" by 1:1 (100%) on A4 paper.

Then start calibrate. Select "LEG1", and then click "X+", "X-", "Y+", "Y-" and "Z+", "Z-", so that the end of Leg1 is aligned with the black dots in the graph.

! Please note that each click will only move the leg by 1mm, so you may need to click many times.
You can use keyboard to move the leg. The key is marked in brackets on the button.

Then select the "LEG2", "LEG3" and “LEG4" to move other legs to corresponding dots as shown below.



Click "CONFIRM" and the calibration data will be stored in the robot.

Click "VERIFY", and then the robot will restore to the state before calibration. Then converted to the state after calibration, which indicates that the calibration is completed. If the end of legs is not aligned with the dots, click RESET and restart from moving the legs to try again.

The calibration needs to be excuted only once. Its data is stored in EEPROM. It will not be changed if you upload the default sketch again. If you disassemble the robot, replace the servo or control board, you need to calibrate the robot again. If you are not satisfied with the results of last calibration, you can also choose to calibrate again.

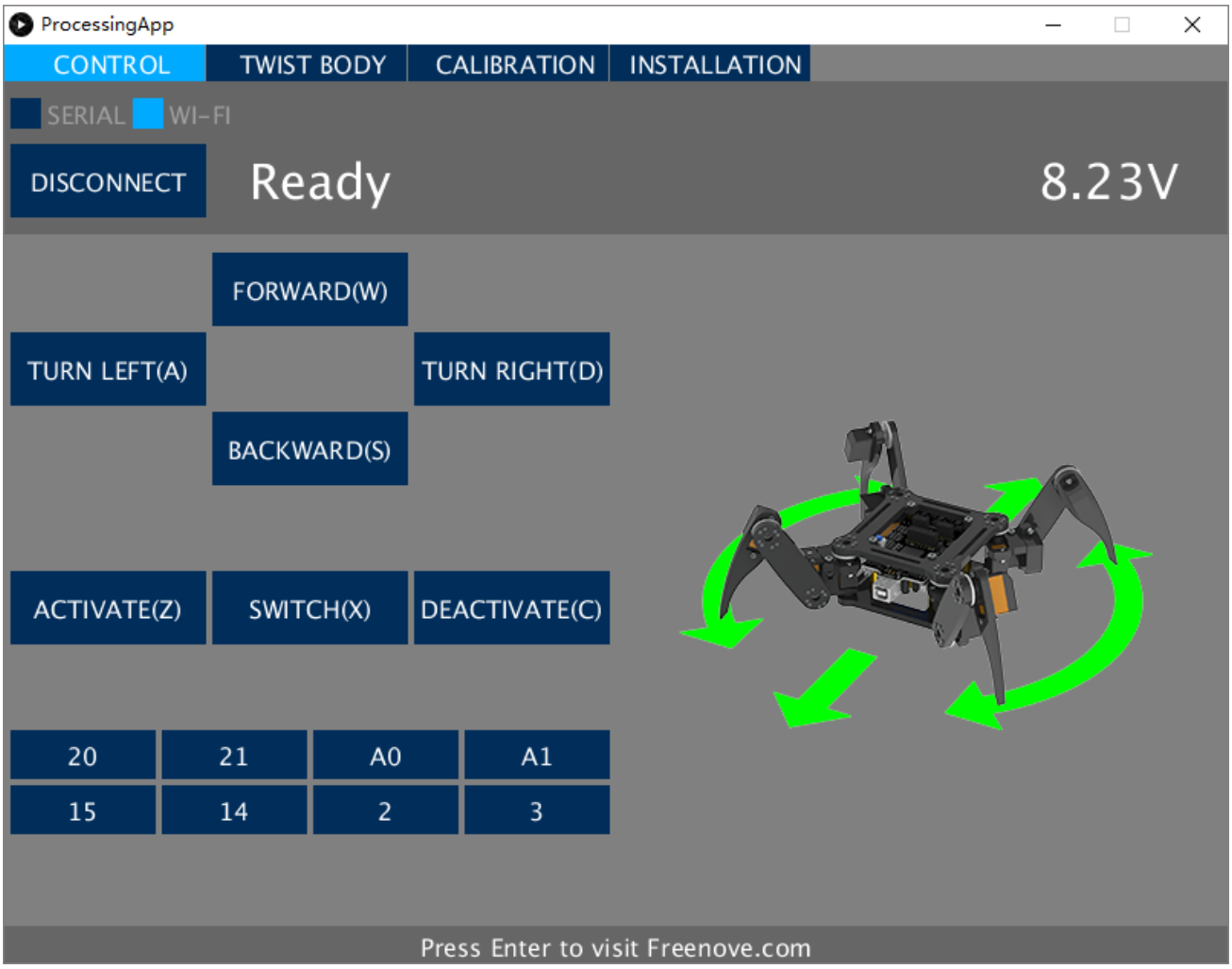
Default Functions

Now we can control the robot.

Use Computer

Now you can use Wi-Fi to connect the control board. Turn on the switch, the robot will create a Wi-Fi hotspot named "Freenove Quadruped Robot" with password "Freenove" (case sensitive). Connect the computer to this hotspot, select "WI-FI" in Processing App, and then click "CONNECT" button.

In the "CONTROL" page, you can control the basic functions of the robot.

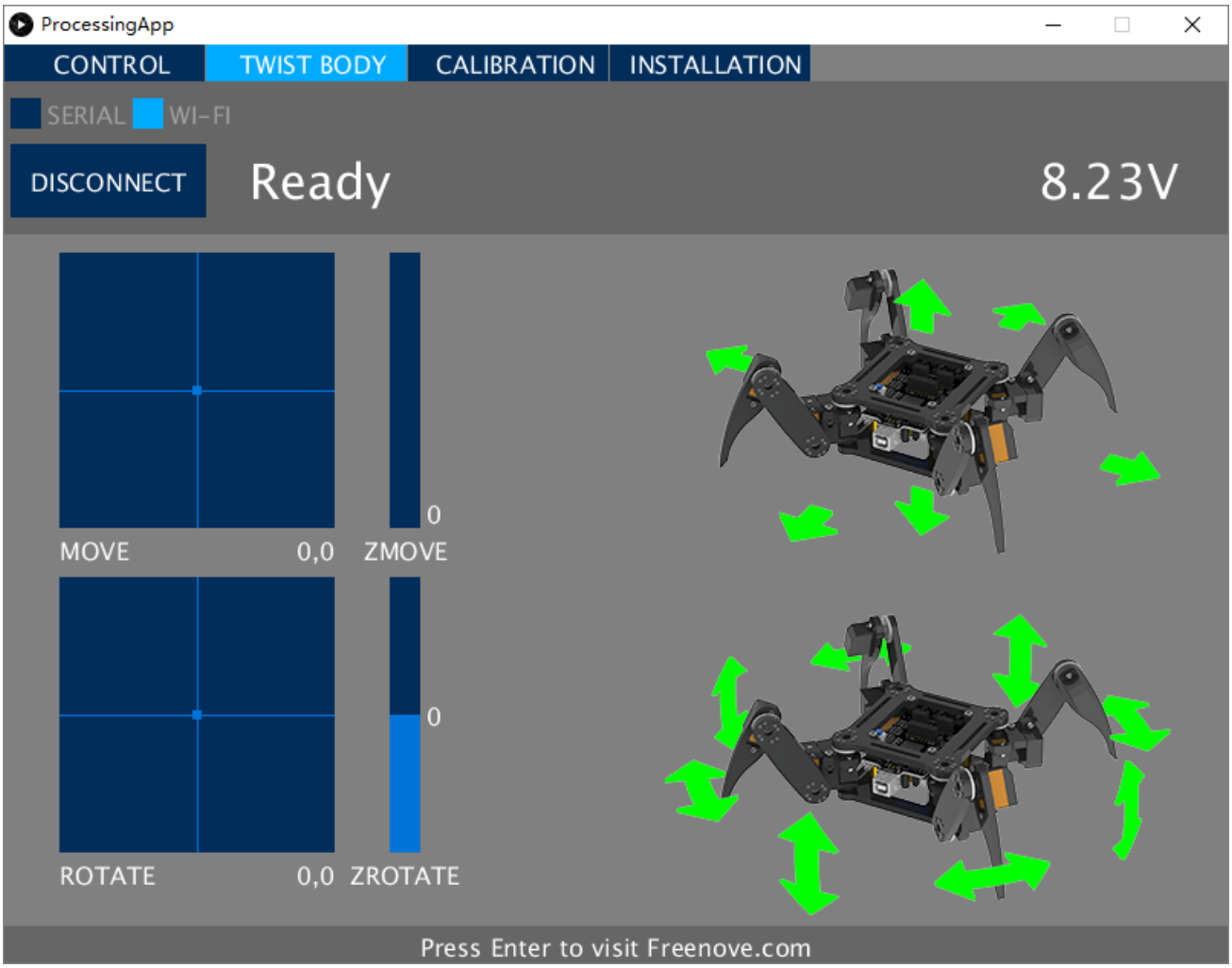


You can execute the following operation:

- Forward, backward, turn left, turn right.
- Switch active mode and sleep mode. Sleep mode can help to save the power of batteries.
- Switch output state of IO ports. You can connect and control some output modules.

You can use keyboard to move the leg. The key is marked in brackets on the button.

In the "TWIST BODY" page, you can control the robot to move and rotate body in the place where it stays.



You can execute the following operation:

- Use mouse to click on the "MOVE" box and "ZMOVE" slider, the robot body will move to the corresponding location.
- Use mouse to click on the “ROTATE" box and "ZROTATE" slider, the robot body will rotate to the corresponding posture.

If there is no action after about 10 secends, the robot will switch into sleep mode to save power automatically. Any command will activate the robot again.

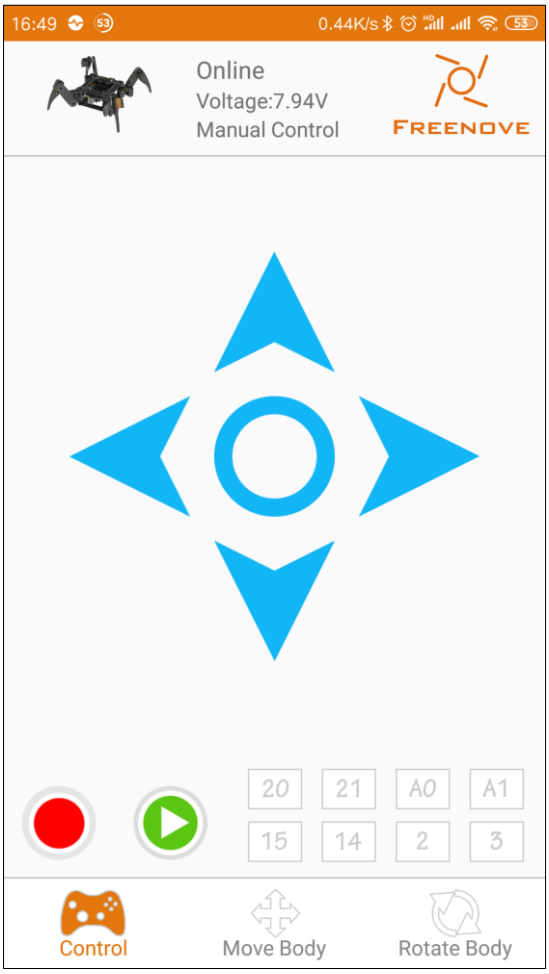
Use Android Device

You can use Android phone or tablet to control the robot.

First, please install Freenove APP for your Android device:

- View or download on Google Play:
  https://play.google.com/store/apps/details?id=com.freenove.suhayl.Freenove
- Download APK file directly:
  https://github.com/Freenove/Freenove_App_for_Android/raw/master/freenove.apk

After the installation is completed, connect the Android device to Wi-Fi hotspot of the robot.
The name of Wi-Fi hotspot is " Freenove Quadruped Robot" and the password is "Freenove" (case sensitive).
Then open the Freenove APP and select “Freenove Quadruped Robot Kit”. As shown below:



The APP will connect the robot automatically. The online state will appear on the top. If the offline state appears, you can touch the robot logo on the top left corner to reconnect.

This APP is similar to the Processing App. You can explore it by yourself, or refer to its tutorial:
https://github.com/Freenove/Freenove_App_for_Android/raw/master/Tutorial.pdf

Use Remote Control

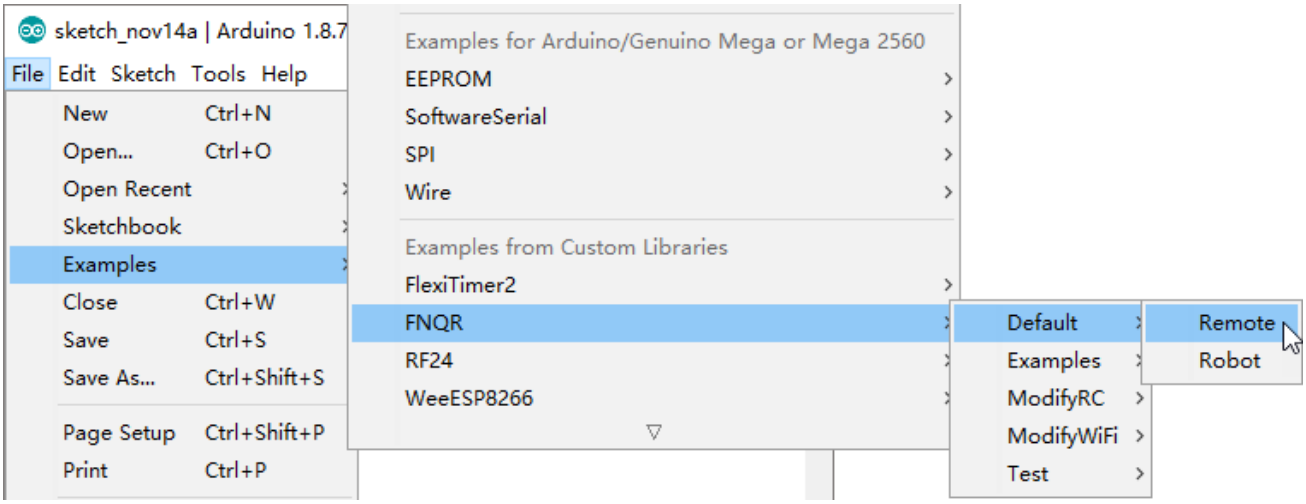
You can also use the remote control (Freenove Remote Control Kit, FNK0028) to control the robot.

Please refer to tutorial(https://github.com/Freenove/Freenove_Remote_Control_Kit/raw/master/Tutorial.pdf) to assemble the remote control first. The Freenove Remote Control Kit contains two NRF24L01 modules, fix one of them to the robot. Please turn off the power when assembling.

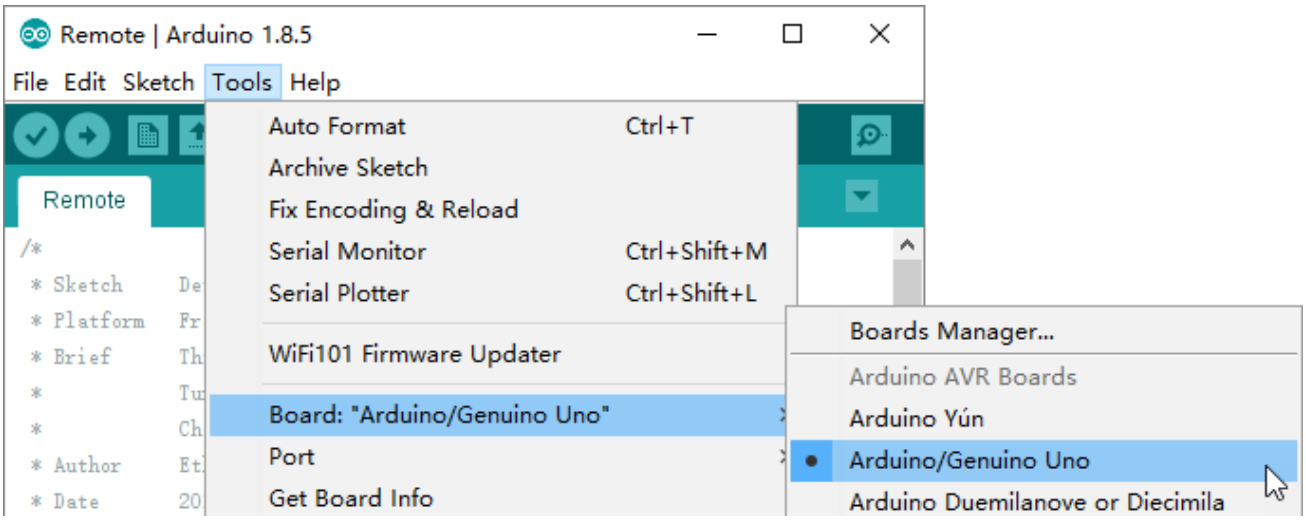
Fix NRF24L01 module to the control board.

Fix it as below.

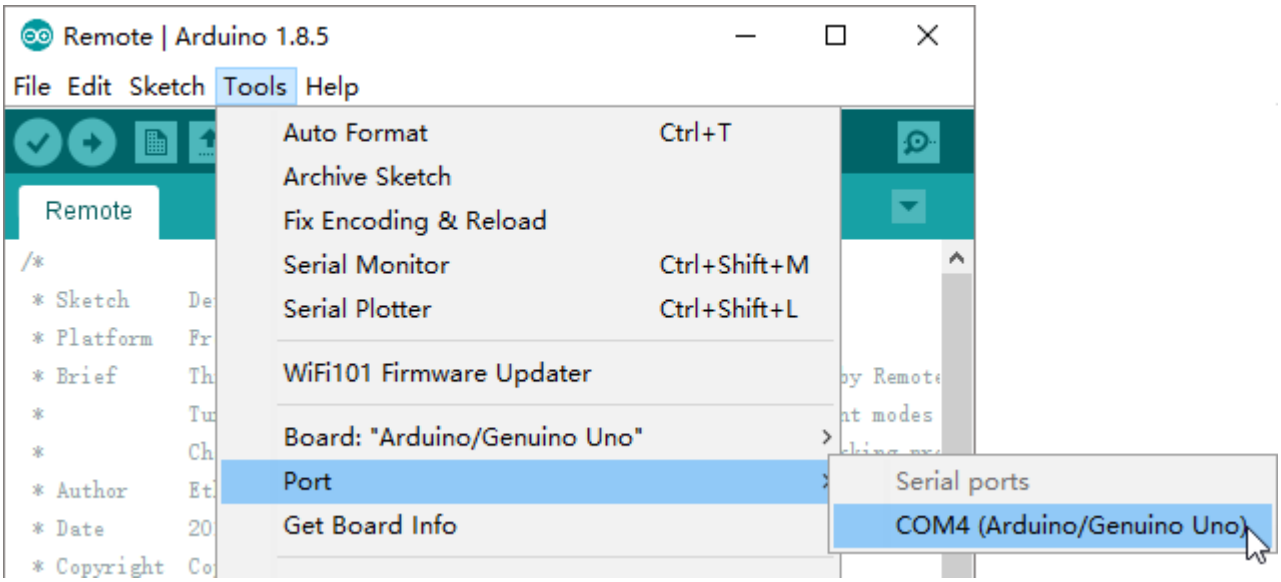
Now, upload sketch for remote control. Connect the remote control to computer.
Open "File" > "Examples" > "FNQR" > "Default" > "Remote".



Select "Board" > "Arduino/Genuino Uno".



Select "Port" (Your COM number may be different from the following figure).



Then upload it to the remote control.

Install a 9V battery for the remote control or connect it to any available USB port. Turn on the power switch of remote control and robot. If a wireless connection is established between the remote control and the robot, the LED3 on remote control will light up or flash. Then you can use the remote control to control the robot.

The remote controller has 3 toggle switches and 3 buttons. The closer toggle switche and button are connected. Turn on or turn off these switches or buttons to control the robot under different mode:

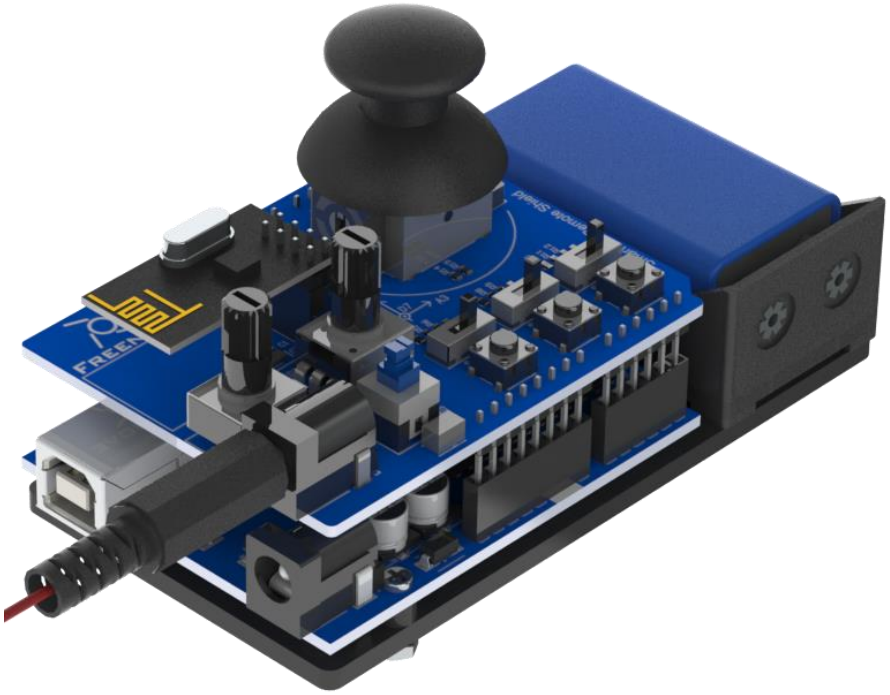
- Only turn on S1:
  You can use the joystick to control the robot to move and turn.
  Press the joystick to swich between active mode and sleep mode.

- Only turn on S2:
  You can use the joystick to control the robot to move body in place.

- Only turn on S3:
  You can use the joystick and POT1 to control the robot to rotate body in place.

- Turn on S2 and S3:
  Move body when only turn on S2, and then turn S3 to rotate body based on the move body position.

When turn on S2 or S3, you can rotate POT2 to adjust the height of robot body.



Programming

It is easy to reprogram this robot.

Modify Default Sketch

If you just want to modify the parameters of the default functions, it is very simple.

! In this case, all previous functions will not be affected except the parameters you want to modify.

Just add the corresponding functions to the default sketch, under the following line.

```
1   robot.Start(true);
```

! Only the functions mentioned in this section can be added.
! Other codes of default sketch cannot be modified or deleted.

You can modify the Wi-Fi hotspot name and password by calling the following function.

```
1   void FNQR::SetWiFi(String name, String password)
```

Please open "File" > "Examples" > "FNQR" > "ModifyWiFi" > "Robot" to see how to use.

When there are many Wi-Fi signals around, you may not be able to connect to the robot or the signal is not good. Then you can try to modify the channel by calling the following function.

```
1   void FNQR::SetWiFiChannel(byte channel)
```

Please open "File" > "Examples" > "FNQR" > "ModifyWiFiChannel" > "Robot" to see how to use.

You can modify the wireless communication address between robot and remote.
To modify the robot, call the following function.

```
1   void FNQR::SetRemote(byte byte0, byte byte1, byte byte2, byte byte3, byte byte4)
```

To modify the remote, call the following function.

```
1   void FNQRRemote::Set(byte byte0, byte byte1, byte byte2, byte byte3, byte byte4)
```

You must set same channel to be able to control robot by romote.
Please open "File" > "Examples" > "FNHR" > "ModifyRemote" to see how to use.

You can also modify the wireless communication channel between robot and remote.
To modify the robot, call the following function.

```
1   void FNQR::SetRemoteChannel(byte byte0, byte byte1, byte byte2, byte byte3, byte byte4)
```

To modify the remote, call the following function.

```
1   void FNQRRemote::SetChannel(byte byte0, byte byte1, byte byte2, byte byte3, byte byte4)
```

You must set same channel to be able to control robot by romote.
Please open "File" > "Examples" > "FNHR" > "ModifyRemoteChannel" to see how to use.

Custom Programming

You can also write a new sketch to control the robot by include FNHR library.

! In this case, the robot is controlled only by your code.
Processing App, Android App and remote will no longer work.

! You can add any code you need, not just the functions mentioned in this section.
! You can add sensors and other modules, and the P3 I/O port on the control board can also be used.

Create a new blank Arduino sketch, include FNHR library at the begnning.

```
1  #include <FNQR.h>
```

Then define a robot object.

```
2  FNQR robot;
```

In function setup(), start the robot.

```
3  void setup() {
4    robot.Start();
5  }
```

Now, you can directly use the following code in function loop() to control the robot.

```
6    robot.ActiveMode();
7    robot.SleepMode();
8    robot.SwitchMode();
9    robot.CrawlForward();
10   robot.CrawlBackward();
11   robot.TurnLeft();
12   robot.TurnRight();
13   robot.MoveBody(float x, float y, float z);
14   robot.RotateBody(float x, float y, float z);
15   robot.TwistBody(float xMove, float yMove, float zMove, float xRotate, float yRotate, float zRotate);
```

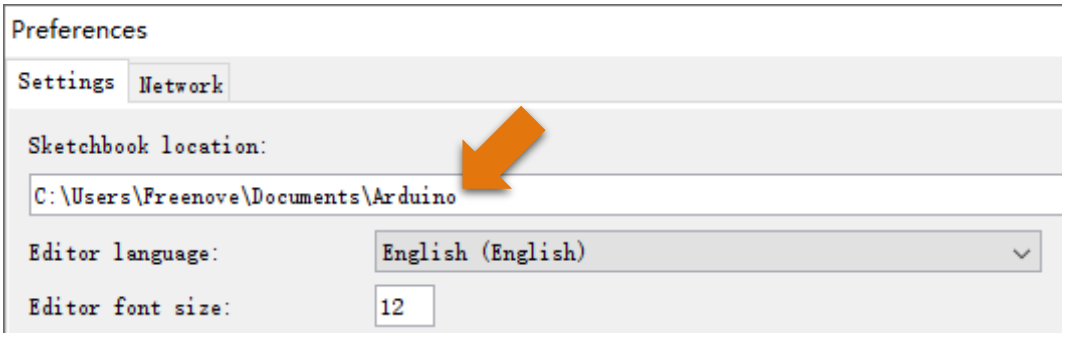
There are some examples in"File" > "Examples" > "FNQR" > "Examples ". You can open and upload them to learn how to use FNQR library to control the robot.

Complete Reprogramming

If you want to use your own code to control every servo, we do not recommend it.
This is more difficult and easy to damage the servos.

If you want to know the details of the code, you can view the FNQR library source code.

All the library files already added to Arduino software are in the "libraries" folder under "Sketchbook location" in the “File” > "Preferences” window.



And if you want to know the details about the control board, please find the schematic in the folder.

If you have further questions, please contact our support for help.

Hardware

Please find the circuit diagrams of the boards in the “Hardware” folder.

If FNQR library is used, the reference voltage of analog input may be switched to an external.

- On V2 board: the reference voltage is 2.094V.
- On other versions of boards: the reference voltage uses the default 5V.

If use a port as an analog input, the voltage range that can be measured is 0V to reference voltage.
If the voltage to be measured is higher than the reference voltage, use two resistors to divide the voltage.

What's Next?

Thanks for your reading.

This tutorial is all over here. If you find any mistakes, missions or you have other ideas and questions about contents of this tutorial or the kit and ect, please feel free to contact us, and we will check and correct it as soon as possible.

After completing the projects of this tutorial, you can try to remodify this robot, including purchasing and installing other electronic modules of Freenove, or improving the code to achieve different functions you want.

If you want to learn more about Arduino, Raspberry Pi, smart cars, robots and orther interesting products in science and technology, please continue to focus on our website. We will continue to launch cost-effective, innovative and exciting products.

Thank you again for choosing Freenove products.