Note

Before you start reading this document, please make sure you have read “Tutorial.pdf”.

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

Contents

Preface

Processing is an easy-to-use, free and open source software for writing graphical programs to run on a computer.

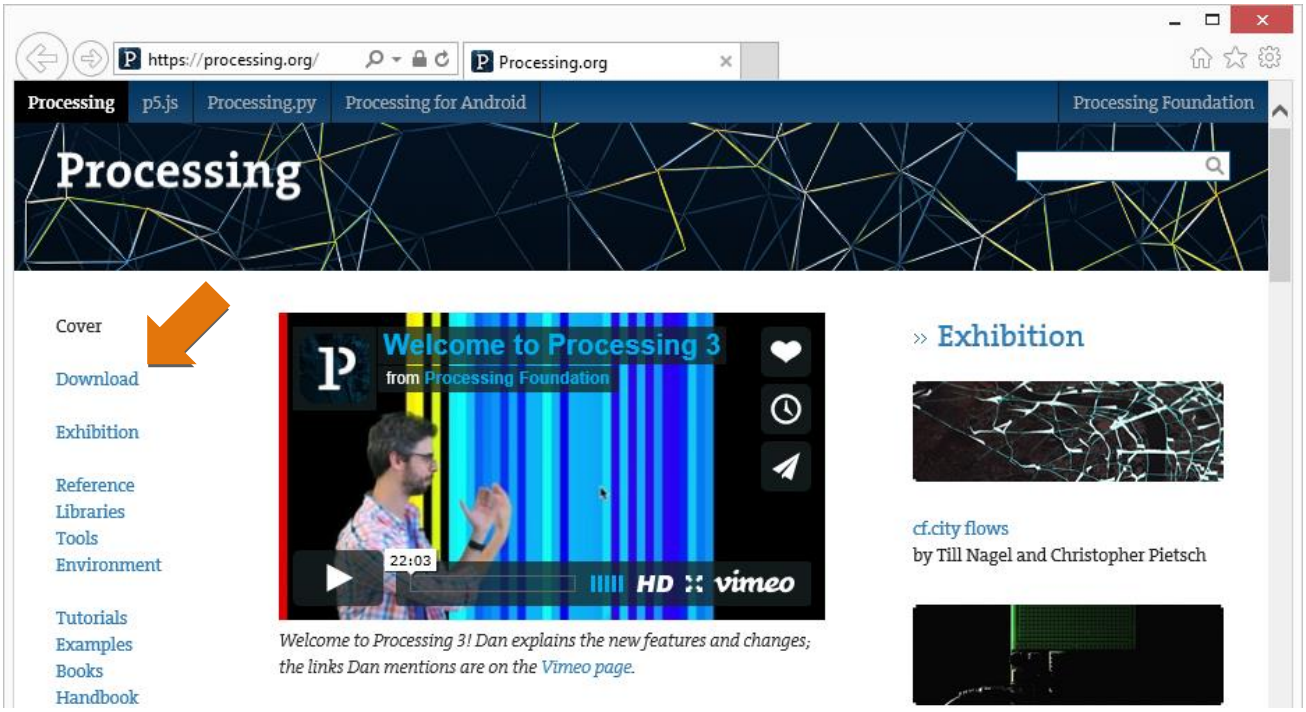
This document will show you how to use Processing to write programs to communicate with the control board. By doing this, we can make virtual instruments, game consoles and other projects.

Processing Software

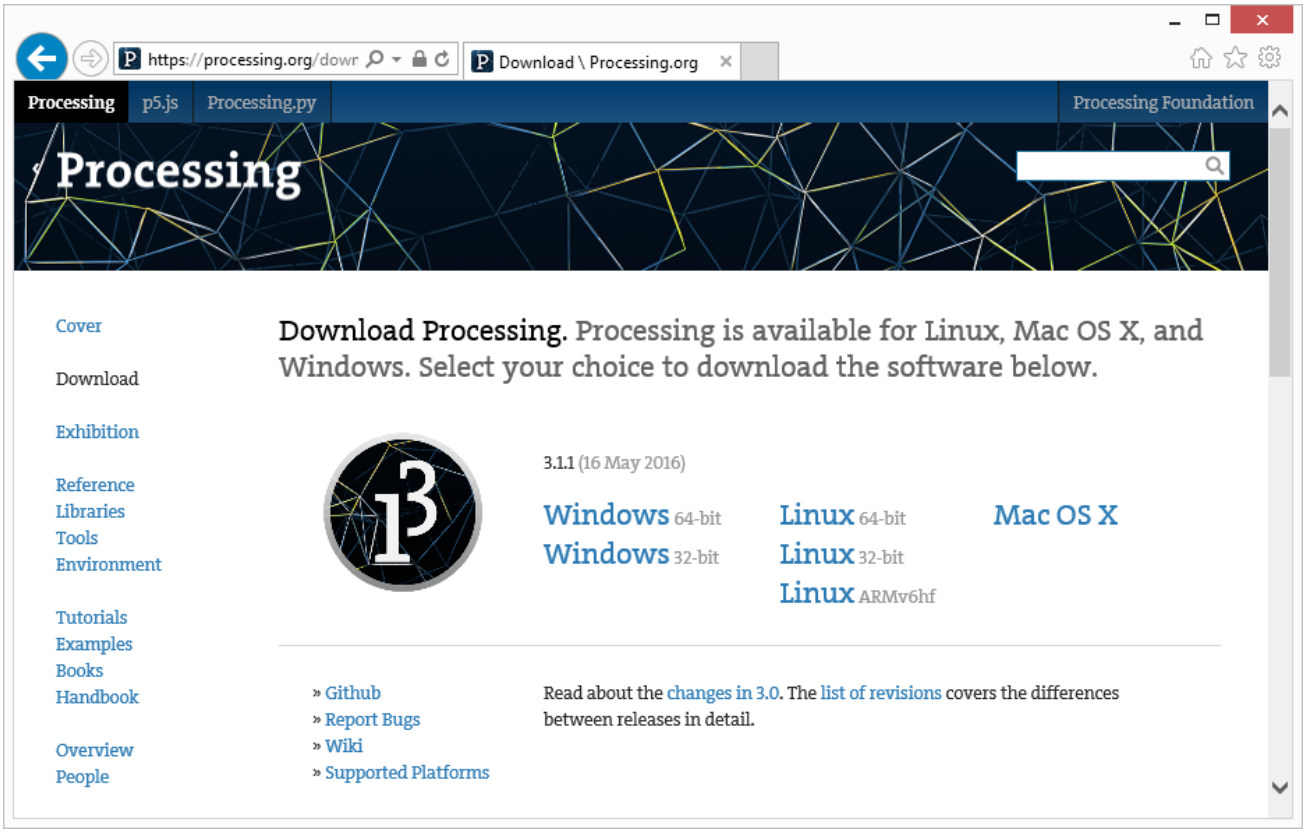
Processing software / Processing Development Environment (PDE) makes it easy to write programs.

Processing software uses Java programming language by default. Do not worry if you don't know Java, because we provide complete code. You can learn Java later if you are interested in it.

First, install Processing software. Visit https://processing.org/, click "Download" to enter the download page.



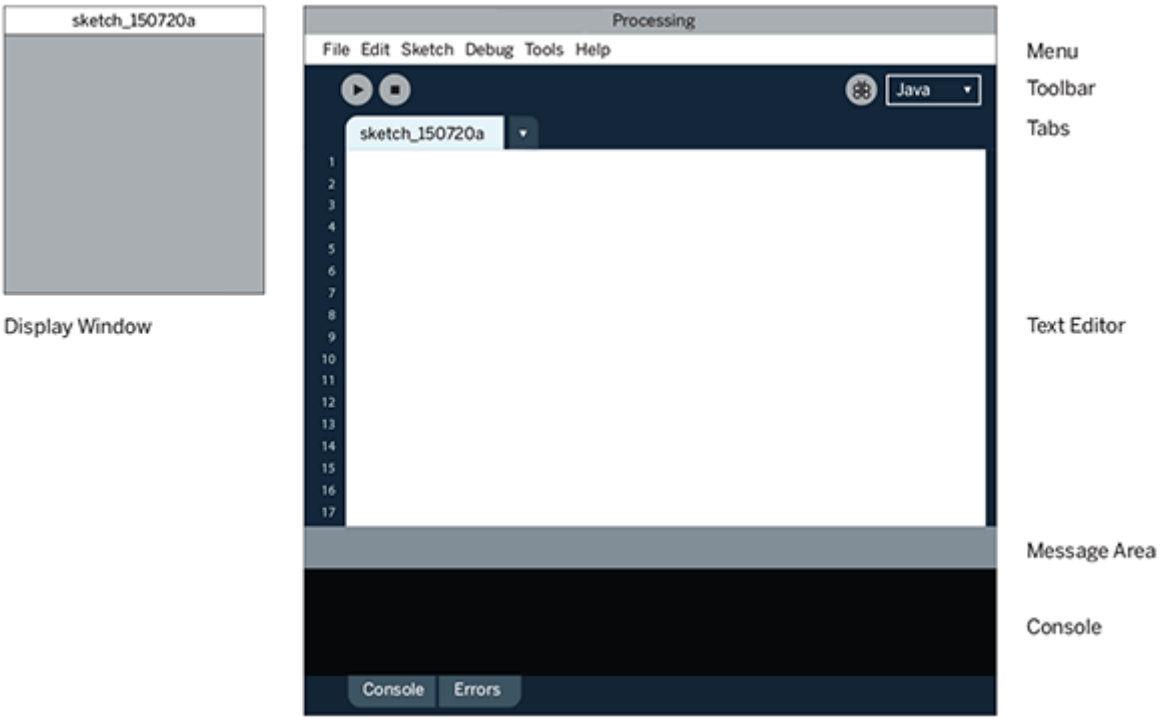
Select the Mac, Windows, or Linux version, depending on what machine you have.



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.



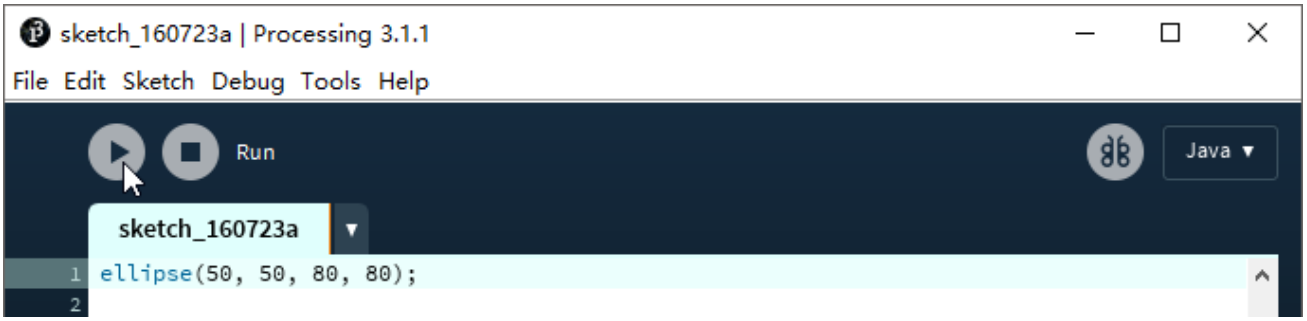
You're now running the Processing Development Environment (or PDE). There's not much to it; the large area is the Text Editor, and there's a row of buttons across the top; this is the toolbar. Below the editor is the Message Area, and below that is the Console. The Message Area is used for one line messages, and the Console is used for more technical details.

First Use

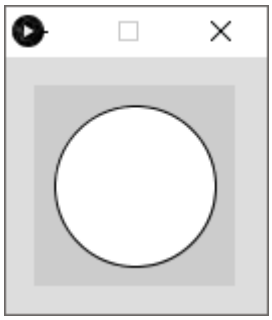
In the editor, type the following:

```
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).

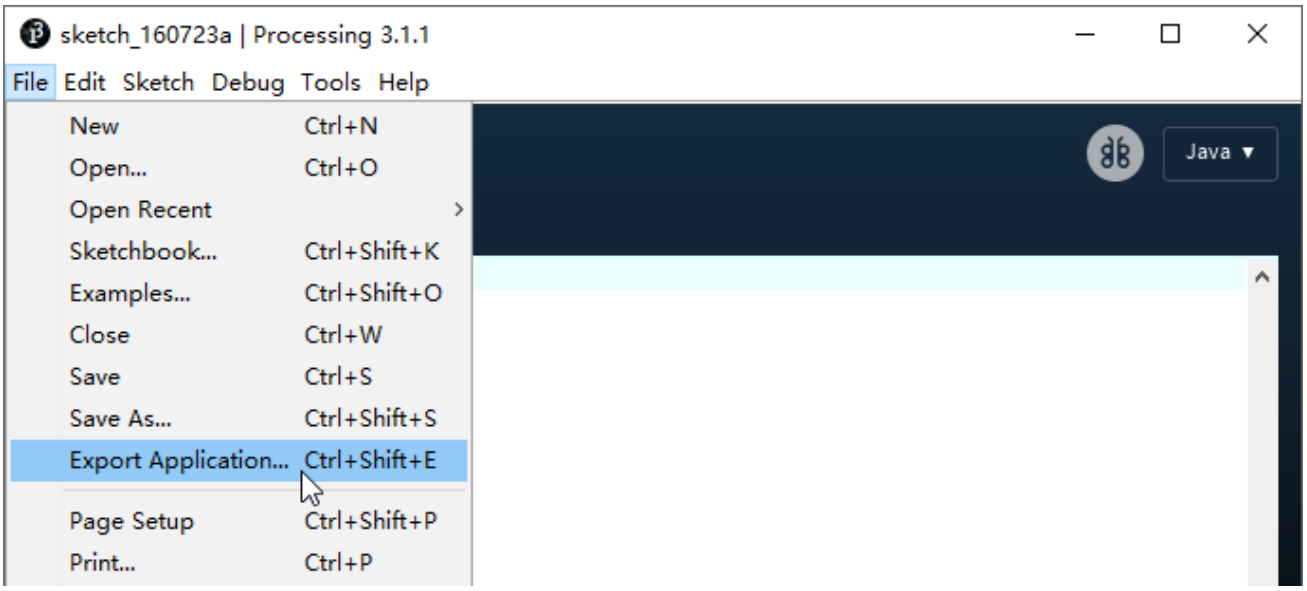


If you've typed everything correctly, you'll see a circle on your screen.



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



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

Chapter 1 Voltmeter

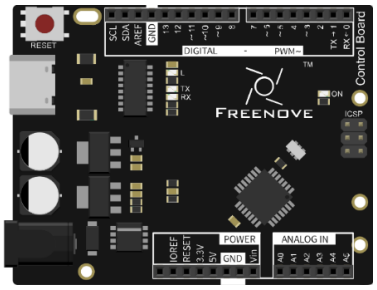
In this chapter, we will use the control board and Processing to make a simple voltmeter to understand the mutual communication between them.

Project 1.1 Voltmeter

First, make a simple voltmeter.

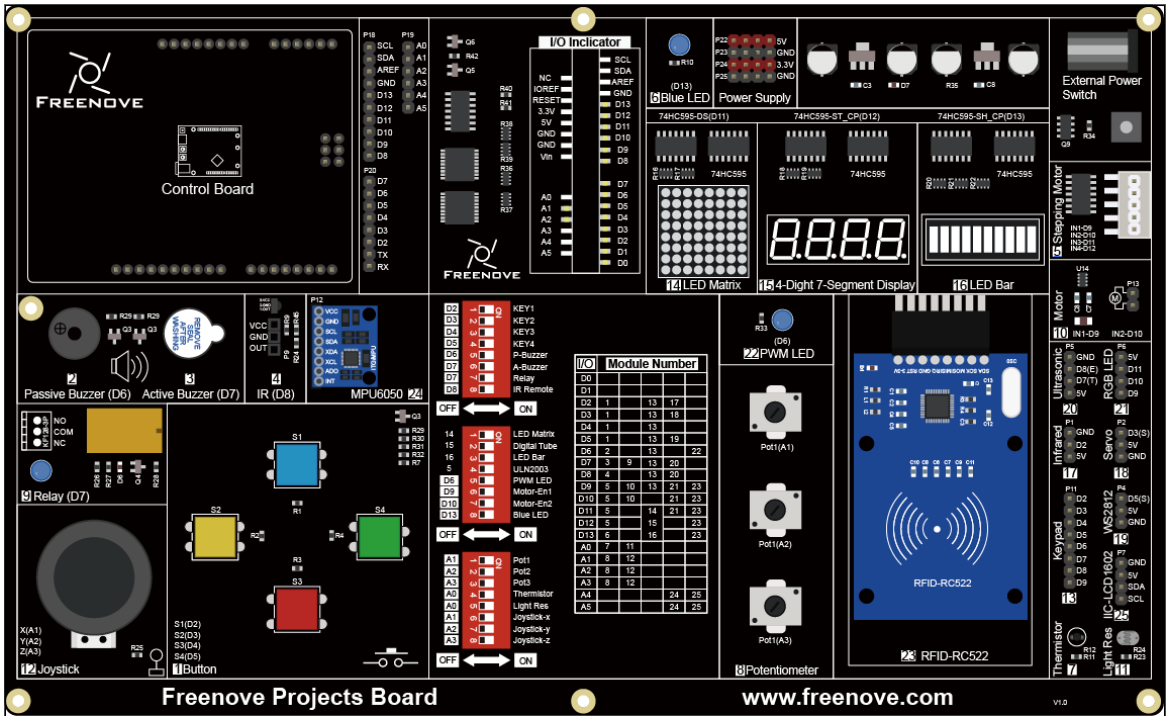
Component list

Control board x1



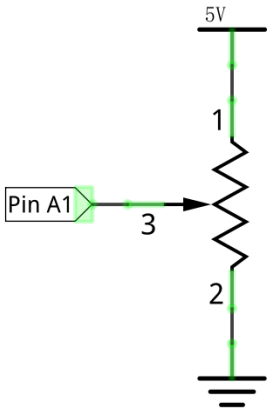
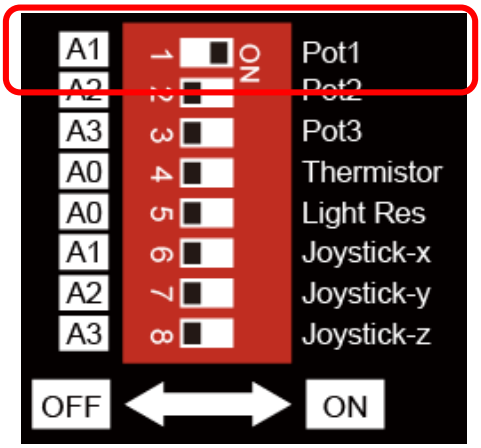
USB cable x1

Freenove Projects Board



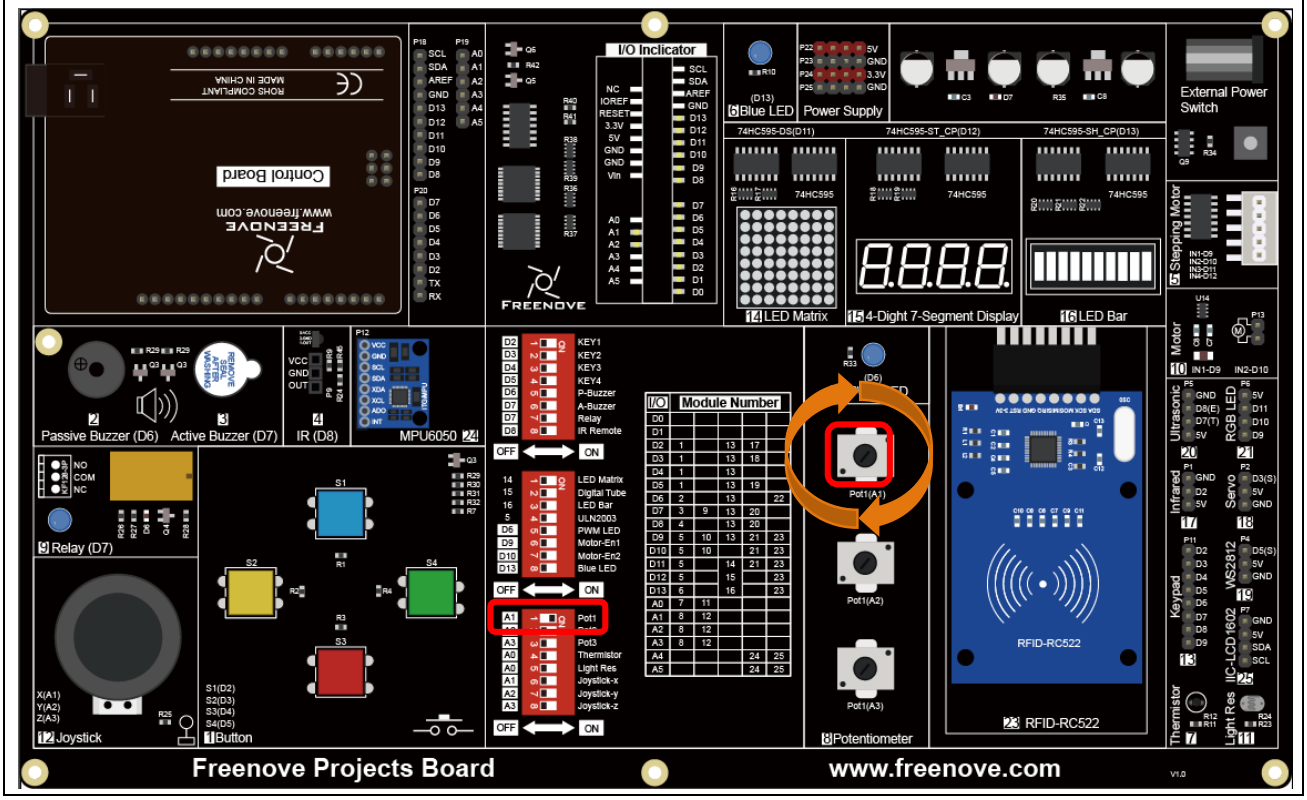
Circuit

Use pin A1 on the control board to detect the voltage of rotary potentiometer.

Schematic diagram	Hardware connection
	

Hardware connection

Insert the Control Board to Freenove Projects Board, and then turn the corresponding switch to the right(ON).



Communication protocol

We need to write code for control board and Processing to complete the interaction project of them, respectively.

In order to simplify and facilitate the operation, we prepared a SerialDevice class for Processing to communicate with the control board. To use this class, we need to upload the following sketch to the control board:
Processing\ControlBoard\SerialDevice\SerialDevice.ino.
This sketch only need to be uploaded once, so the latter projects of this tutorial does not need to upload again.

SerialDevice class and SerialDevice.ino defined the communication protocol between them. The futures include:

- Recognize the control board uploaded SerialDevice.ino and establish connection with it, automatically. No need to view and set the serial number of the control board connected to the computer, even if there are a number of control board, it can be connected automatically.
- If control board uploaded SerialDevice.ino is not connected to computer, the Processing code will not be executed until the connection is done. The Processing sketch does not need to be run again after the connection is done.
- Send data to control board and receive data from it.

Sketch

Before running Processing sketch, make sure that SerialDevice.ino is uploaded to the control board.
Processing sketches is stored under the Processing\Processing folder.

Sketch Voltmeter

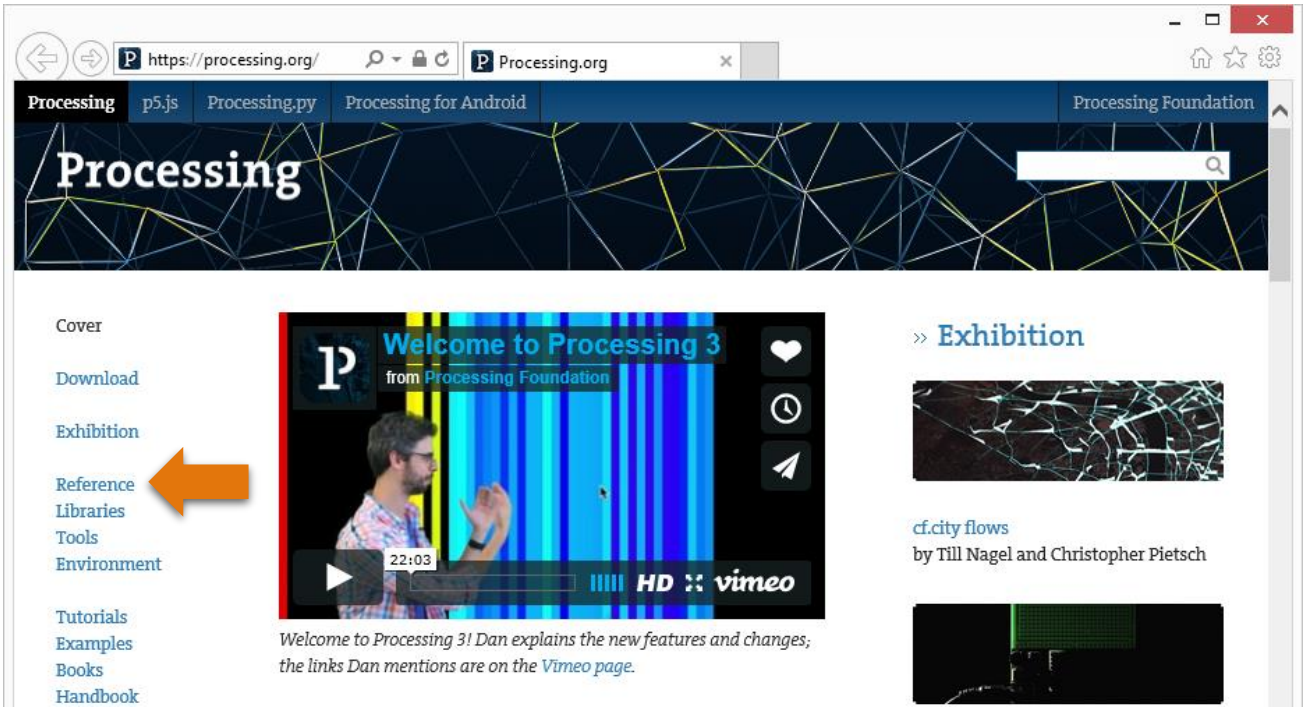
Use Processing to open Voltmeter.pde and click Run. Then, the following window will pop up and its connection to the control board will be started.

If the control board has not been connected to computer, please connect the control board to your computer. If the connection succeeds, the follow will be shown:

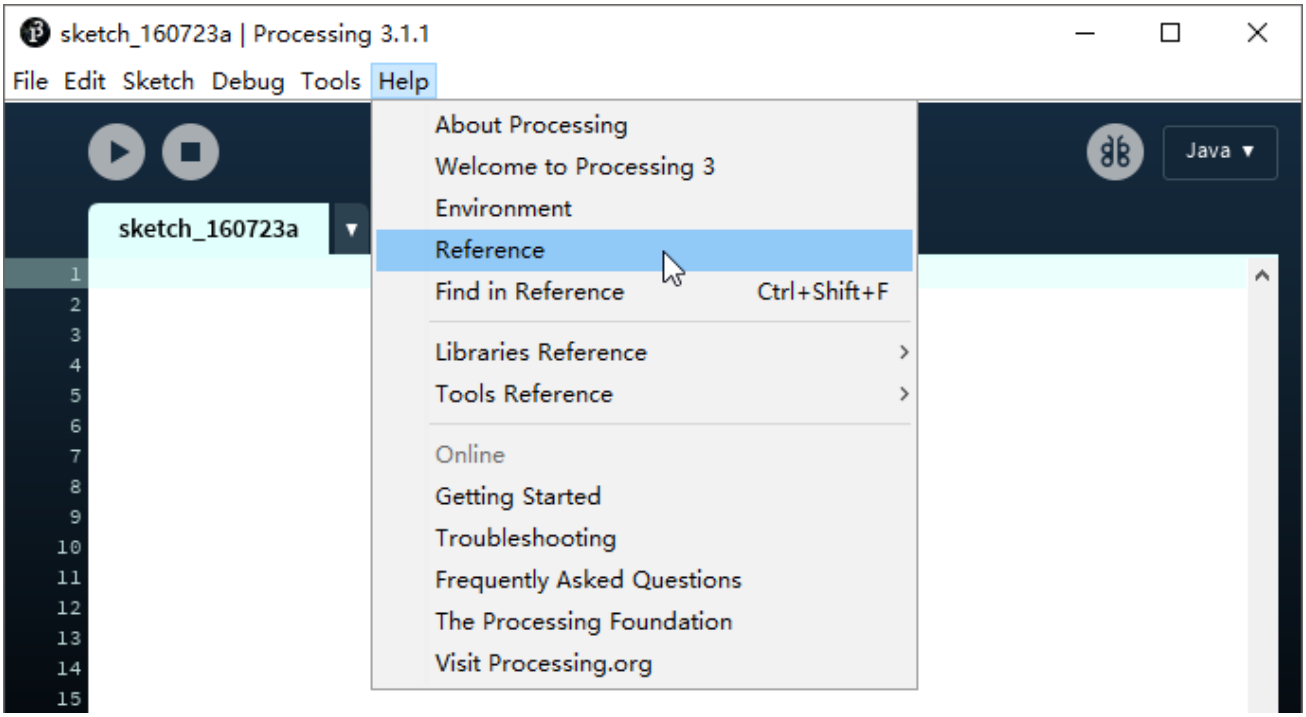
This sketch will obtain analog value from A1 port of control board, and convert it to voltage value to display. You can adujust the potentiometer to observe the change of value, and you can also use the A1 port to measure voltage value of other circuits. Note that the measurement voltage can not exceed 5V, which will do damage to the control board.

Here, Processing sketch code will not be introduced in detail. Interested readers can learn it by yourself.

And as for syntax and standard functions of Processing, you can visit https://processing.org/ and click Reference to view.



Or in the Processing software menu bar, click Help-Reference to view offline documents.

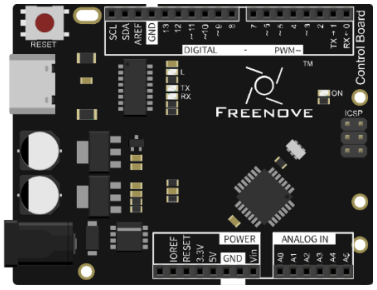


Project 1.2 Voltmeter Dual Channel

Now, let's make a dual channel voltmeter.

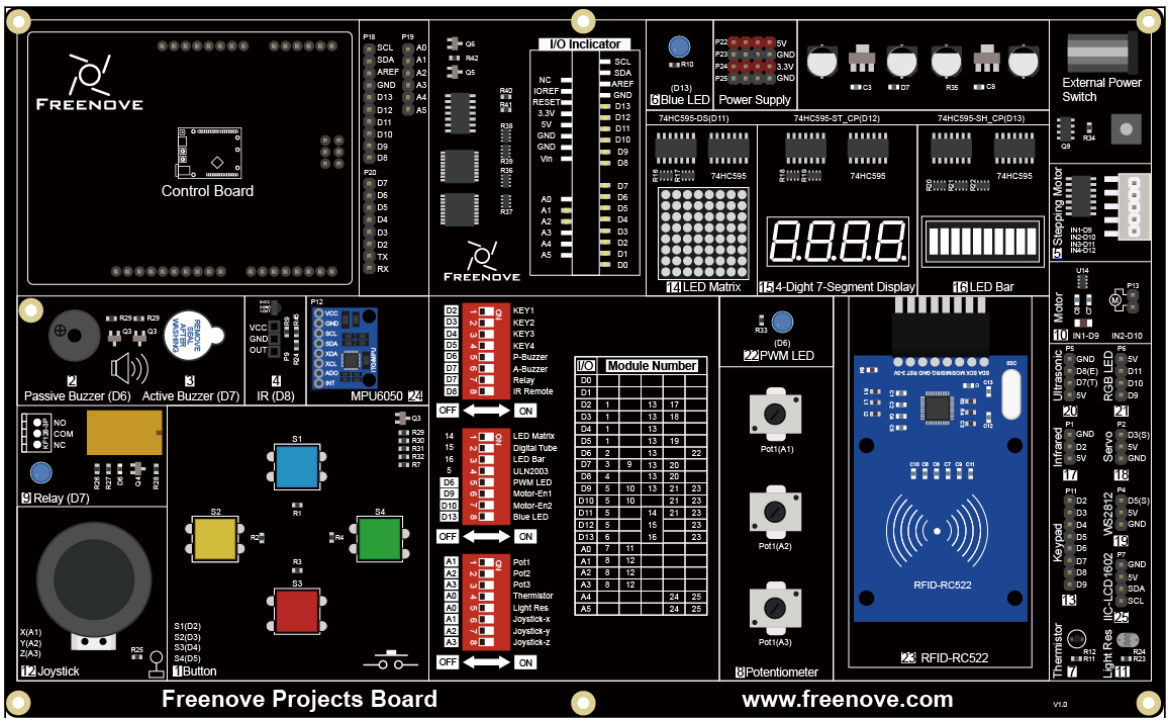
Component list

Control board x1



USB cable x1

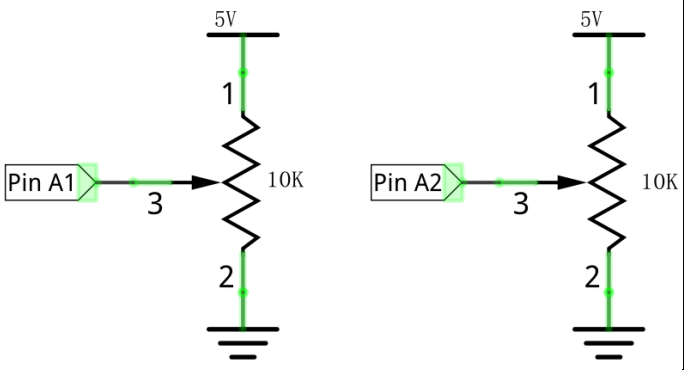
Freenove Projects Board



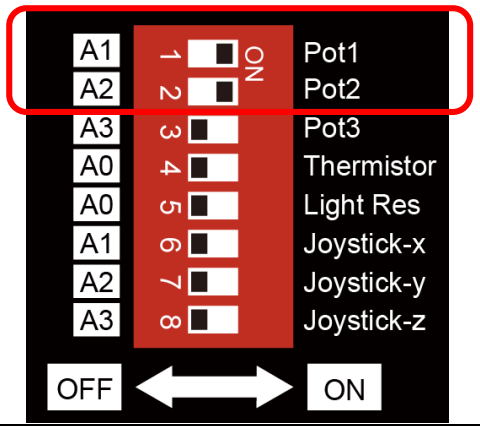
Circuit

Use A1, A2 ports on control board to detect the voltage of rotary potentiometers.

Schematic diagram

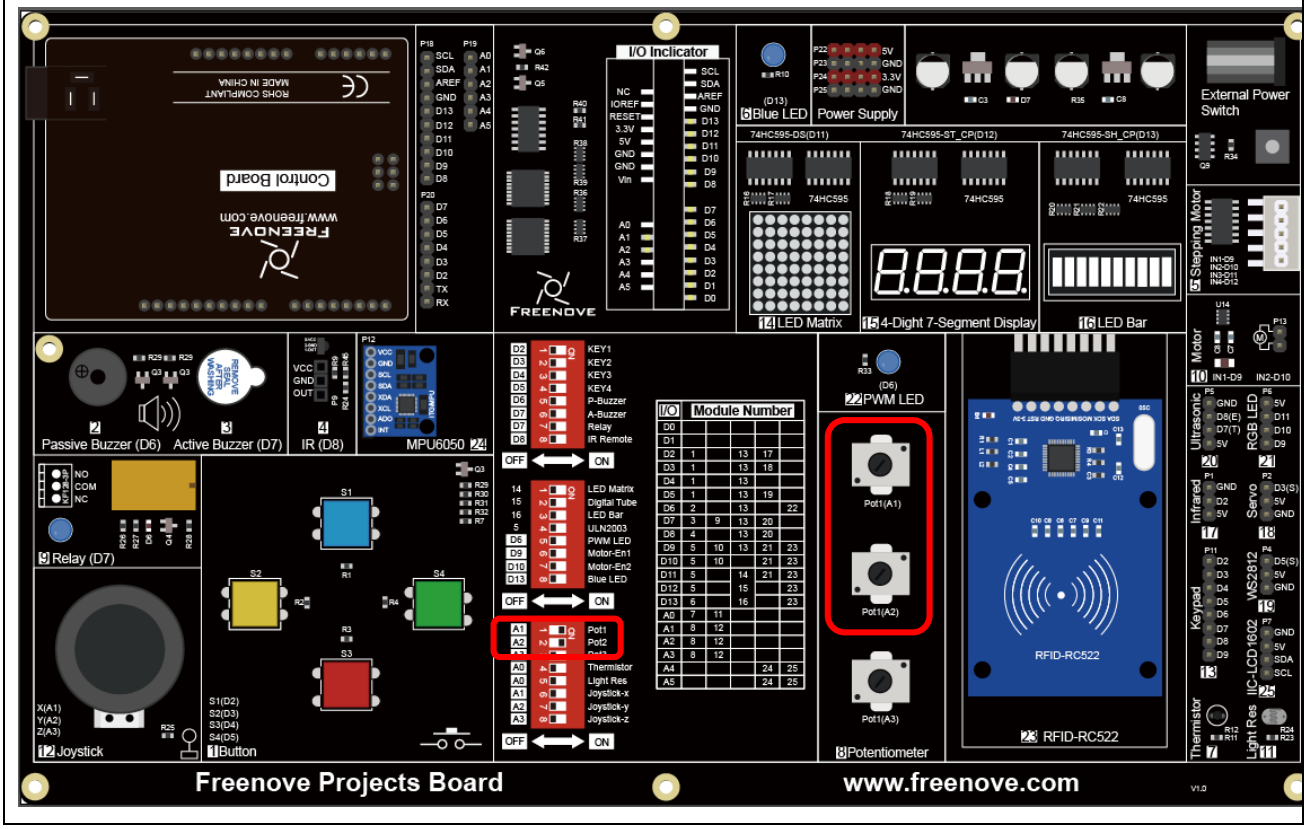


Hardware connection



Hardware connection

Insert the Control Board to Freenove Projects Board, and then turn the corresponding switch to the right(ON).



Sketch

Sketch Voltmeter_Dual_Channel

Use Processing to open Voltmeter_Dual_Channel.pde and click Run. Then, the following window will pop up and its connection to control board will be started.

If the control board has not been connected to computer, please connect the control board to your computer. If the connection succeeds, the follow will be shown:

This sketch will obtain analog value from A1 and A2 ports of control board, and convert them to voltage value to display. You can adujust the potentiometers to observe the change of value, and you can also use the A1 and A2 ports to measure voltage value of other circuits. Note that the measurement voltage can not exceed 5V, which will do damage to the control board.

You can export the two Processing sketches in this chapter to the application as common tools.

Chapter 2 Oscilloscope

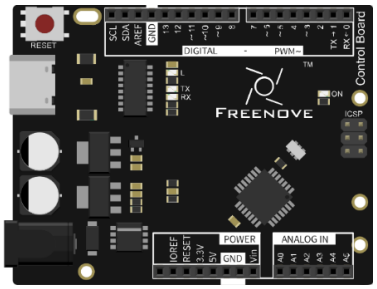
We have implemented a simple virtual instrument voltmeter, before. In this chapter, we will make a more complex virtual instrument, oscilloscope. Oscilloscope is a widely used electronic measuring instrument. It can get the electrical signals not directly observed into visible image to facilitate the analysis and study of various electrical signals change process.

Project 2.1 Oscilloscope

Now, let's use Processing and control board to achieve an oscilloscope.

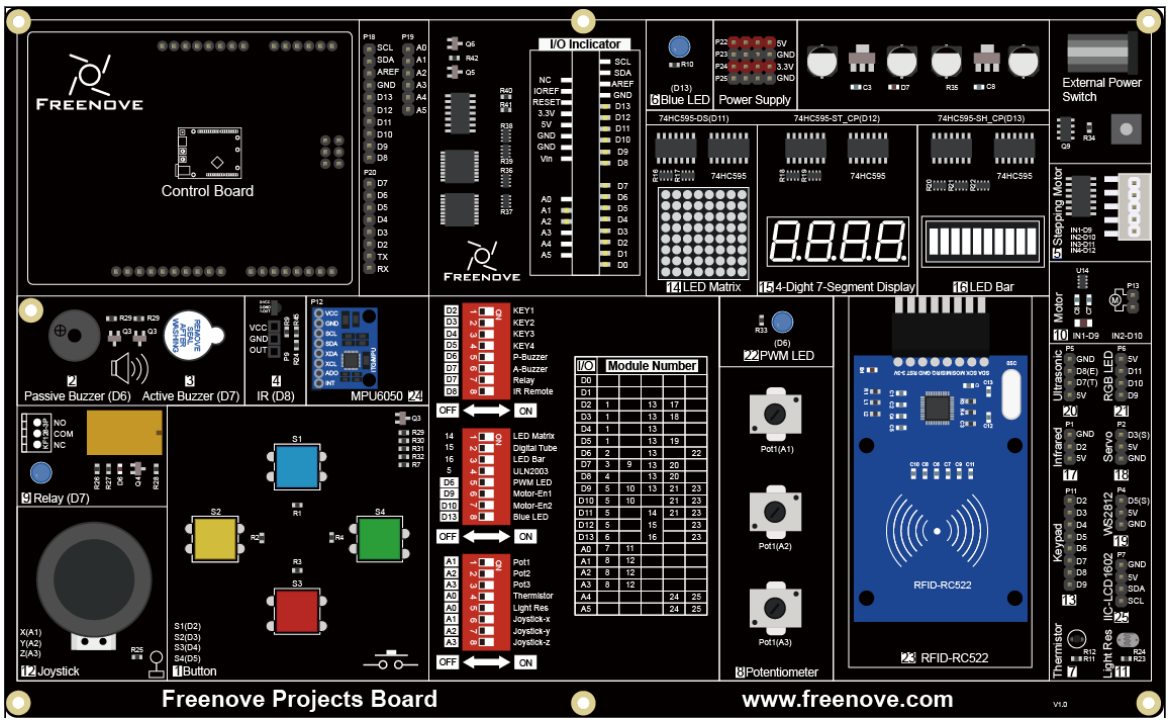
Component list

Control board x1



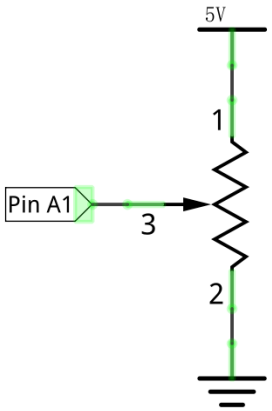
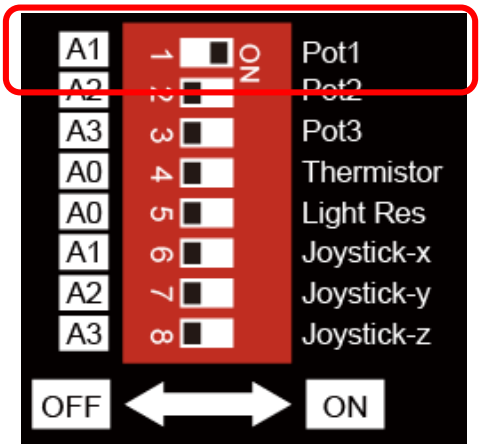
USB cable x1

Freenove Projects Board



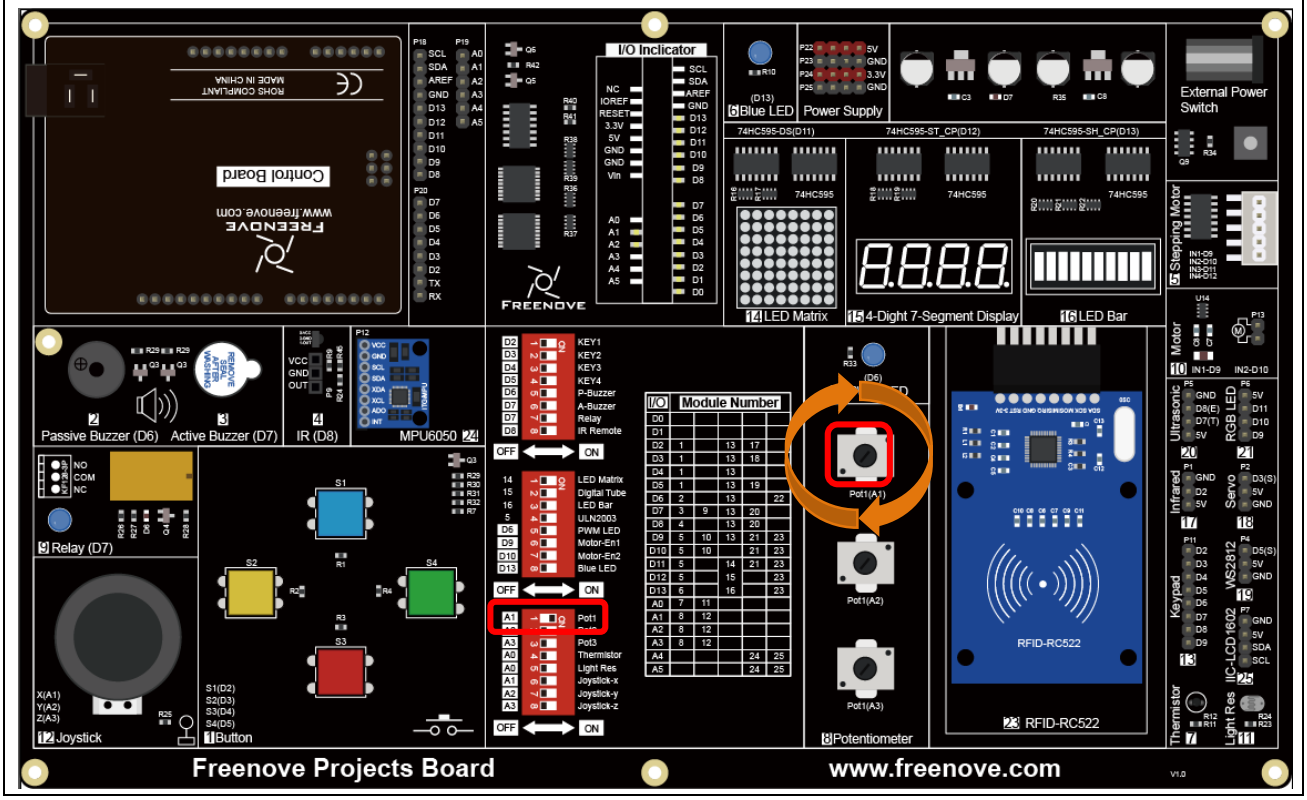
Circuit

Use pin A1 on the control board to detect the voltage of rotary potentiometer.

Schematic diagram	Hardware connection
	

Hardware connection

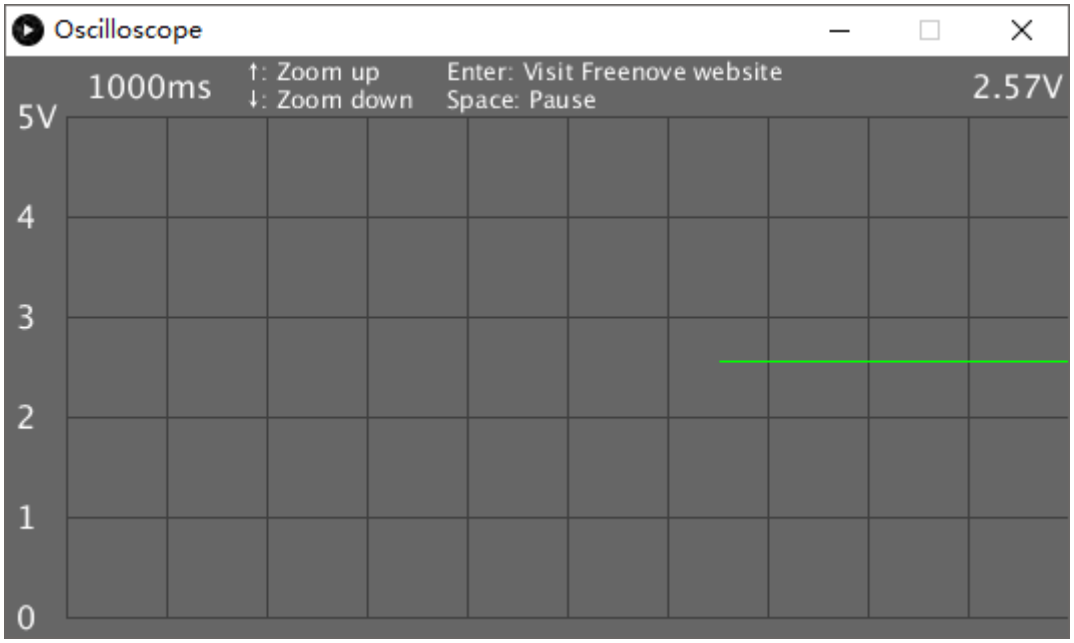
Insert the Control Board to Freenove Projects Board, and then turn the corresponding switch to the right(ON).



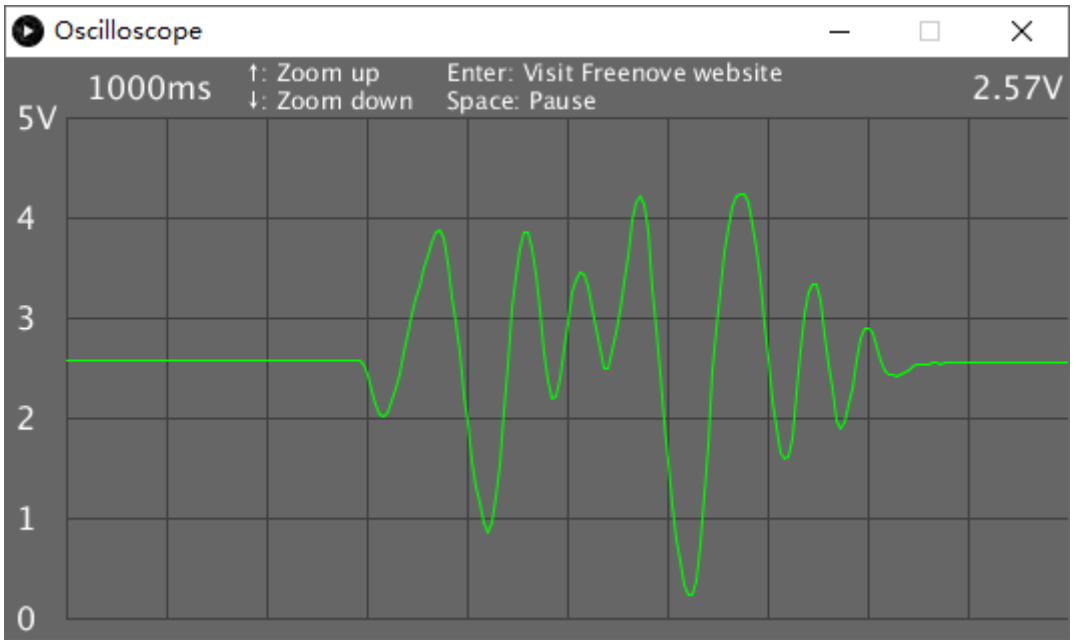
Sketch

Sketch Oscilloscope

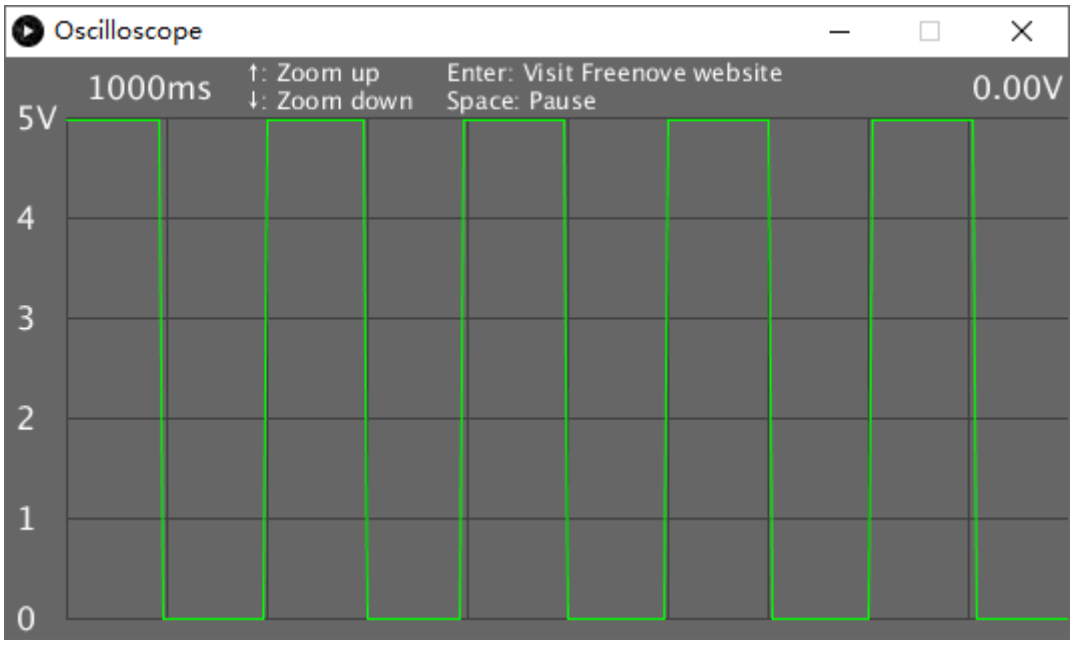
Use Processing to open Oscilloscope.pde and click Run. If the connection succeeds, the follow will be shown:



The green line is the waveform acquisited. Rotate the potentiometer, then you can see changes of the waveform:



Disconnect the A1 port from the potentiometer and connect it to the Pin 13 port. Pin 13 port output is 0.5Hz square wave. As is shown below:



The left side of the software interface is a voltage scale, which is used to indicate the voltage of the waveform. The "1000ms" on top left corner is the time of a square, and you can press "↑" and "↓" key on keyboard to adjust it.

The "0.00V" on top right corner is the voltage value of current signal.

You can press the space bar on keyboard to pause the display waveform, which is easy to view and analysis.

We believe that with the help of this oscilloscope, you can obtain more intuitive understanding of the actual work of some electronic circuits. It will help you complete the project and eliminate the trouble. You can export this sketch to an application used as a tool.

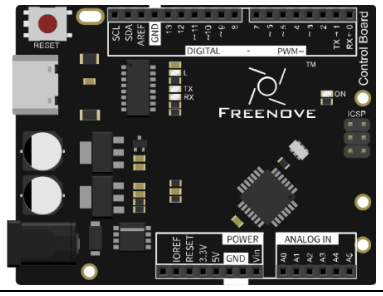
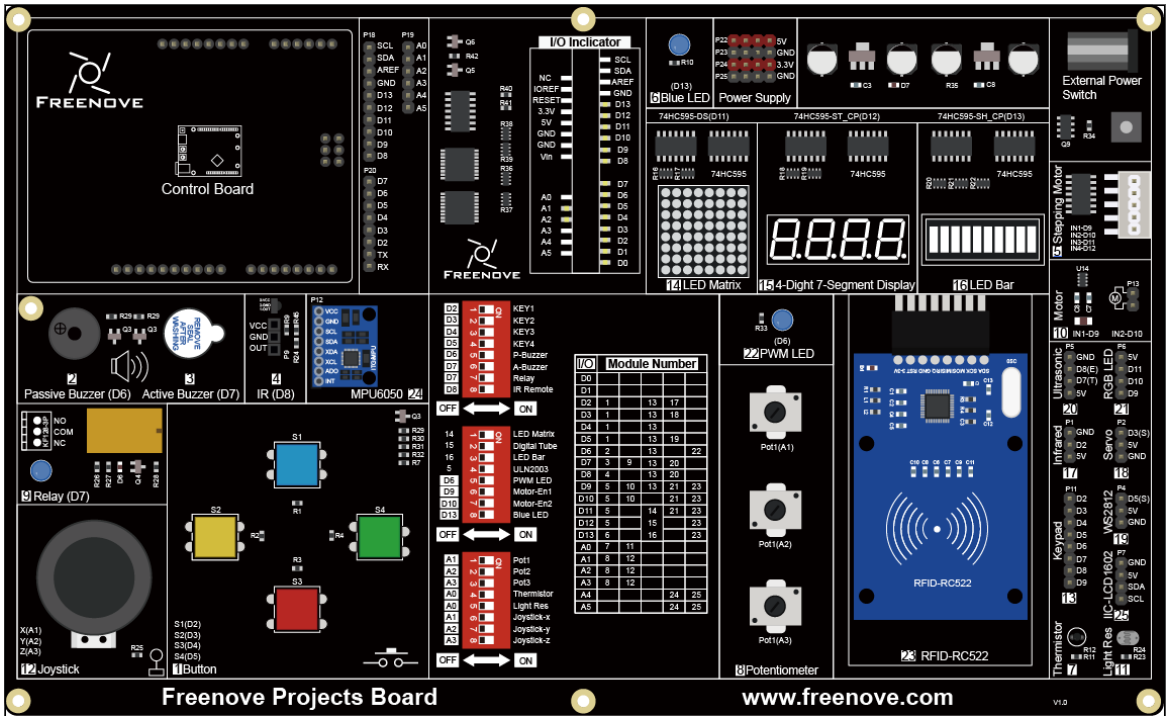
Chapter 3 Control 2D and 3D Figures

In this chapter, we will use connect board to make Processing program display figure changes. And we will control 2D and 3D figures, respectively.

Project 3.1 Ellipse

First, control a 2D figure.

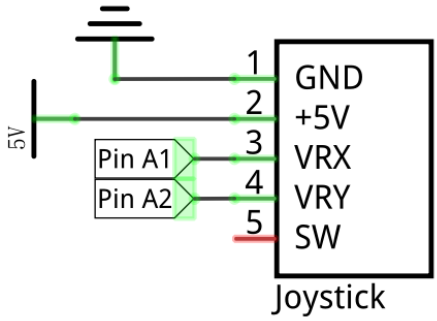
Component list

Control board x1 	USB cable x1 
Freenove Projects Board 	

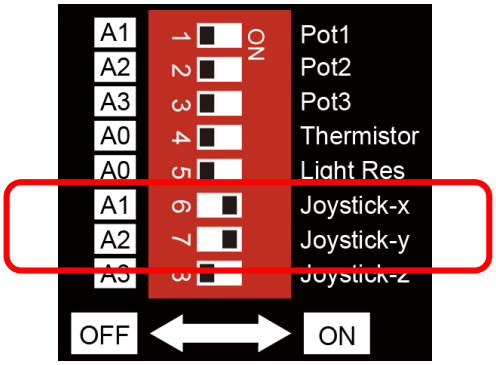
Circuit

Use A1 and A2 ports on connect board to detect the voltage value of two rotary potentiometers inside joystick.

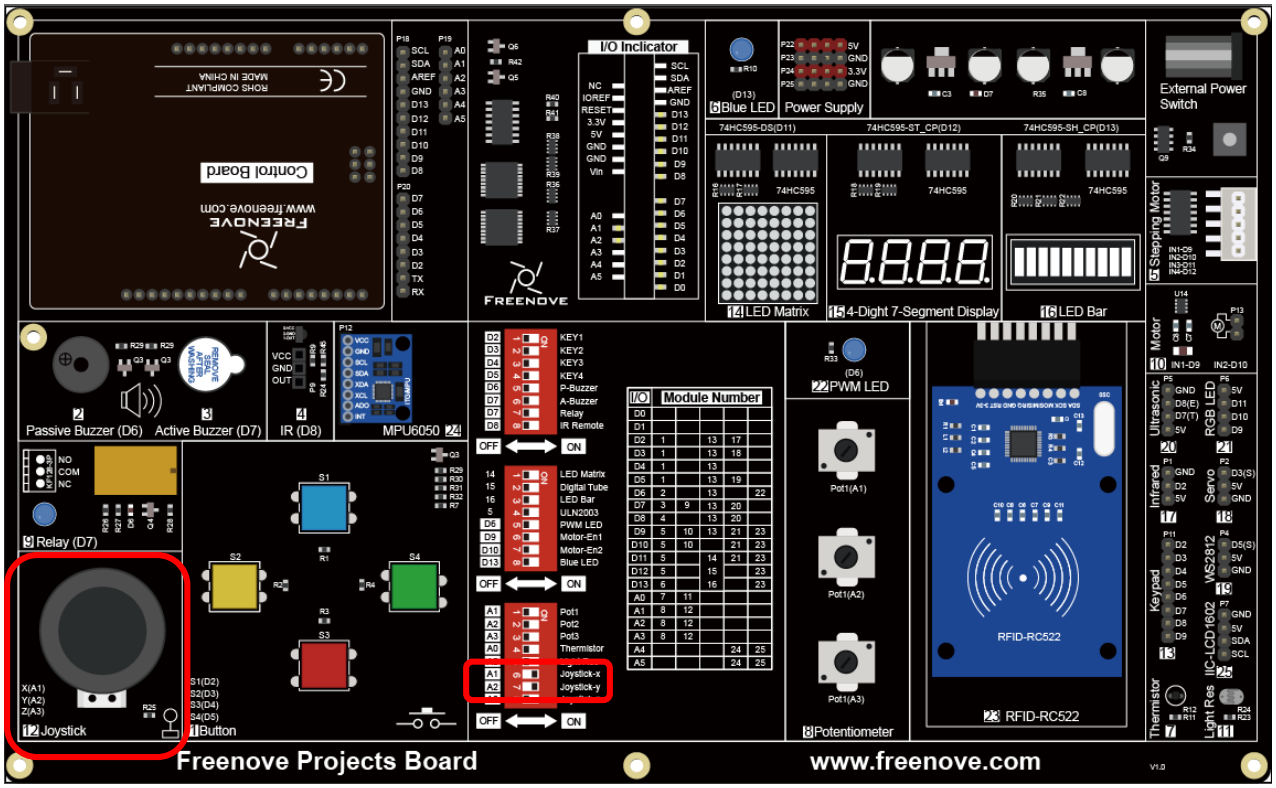
Schematic diagram



Hardware connection



Hardware connection



Sketch

Sketch Ellipse

Use Processing to open Ellipse.pde, then click Run. If the connection succeeds, the following will be shown:

Then you can change the ellipse shape by shifting the joystick:

Project 3.2 Box 3D

Now control 3D figures.

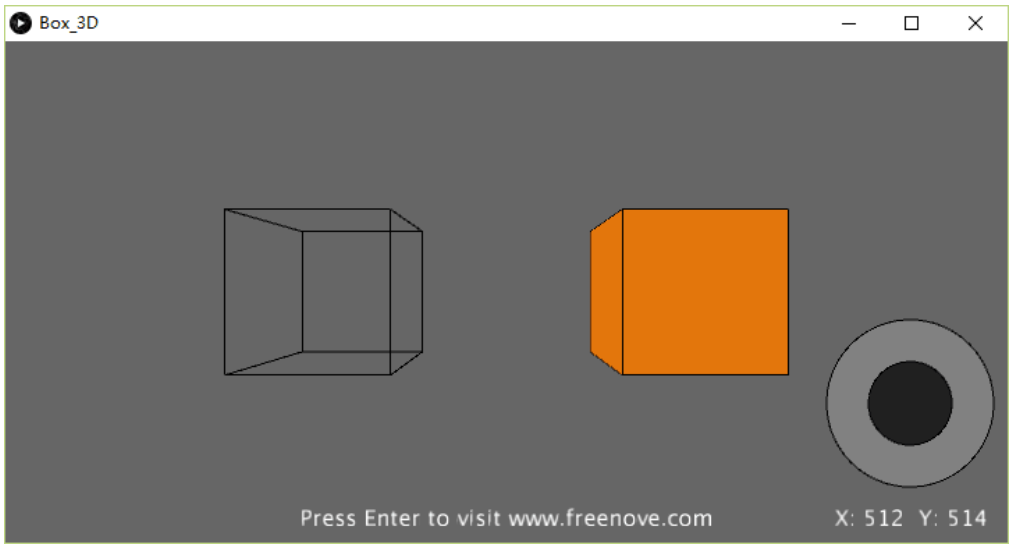
Component list & Circuit

The same as last section.

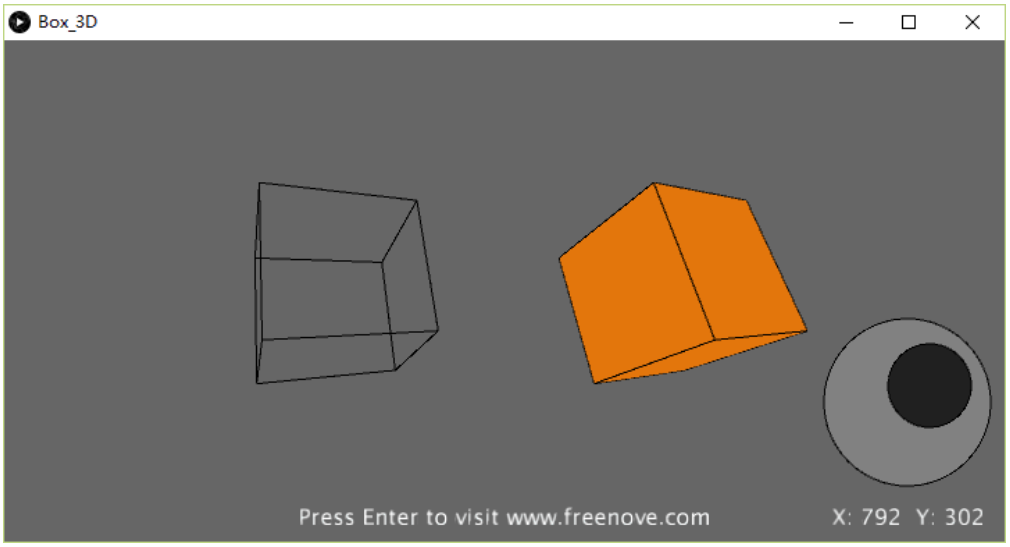
Sketch

Sketch Box_3D

Use Processing to open Box_3D.pde, and click Run. If the connection succeeds, the following will be shown. The left is a 3D box presented by line and the right is a 3D box entity.



Then you can change the space angle of two 3D box by shifting the joystick:



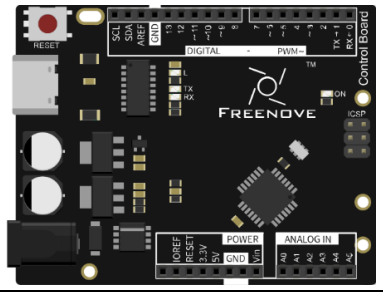
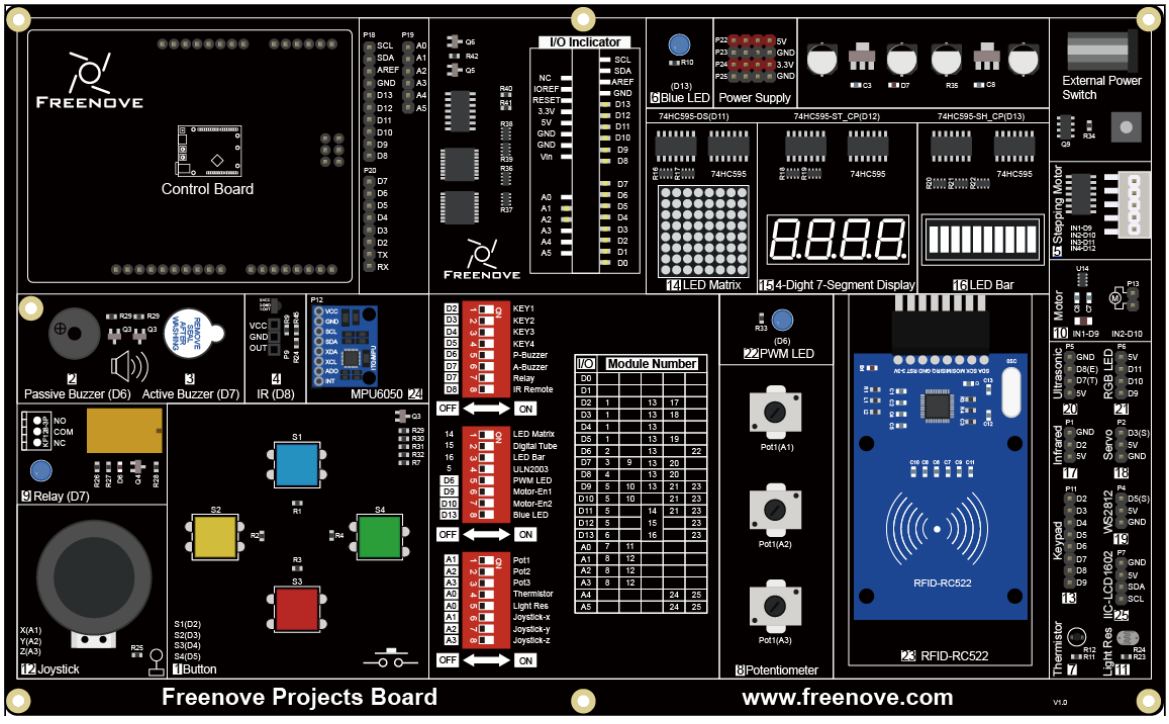
Chapter 4 Snake Game

We have experienced controlling 2D and 3D figures before. Now, we use connect board to play the classic snake game. You will experience both 2D and 3D version.

Project 4.1 Snake Game

First, let's experience the 2D version game.

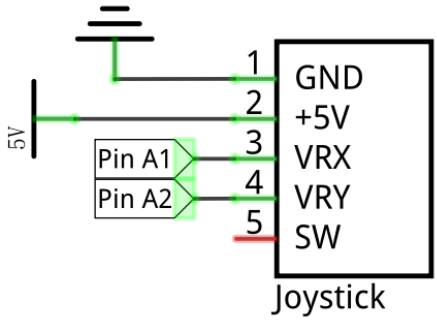
Component list

Control board x1 	USB cable x1 
Freenove Projects Board 	

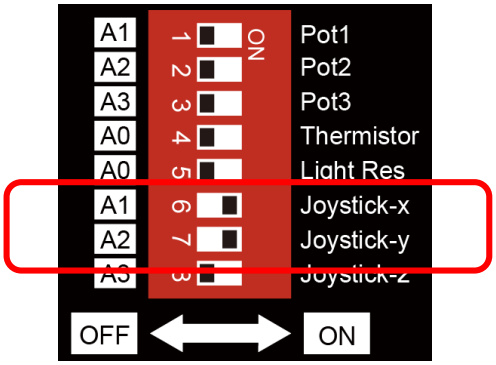
Circuit

Use A1 and A2 ports on connect board to detect the voltage value of two rotary potentiometers inside joystick.

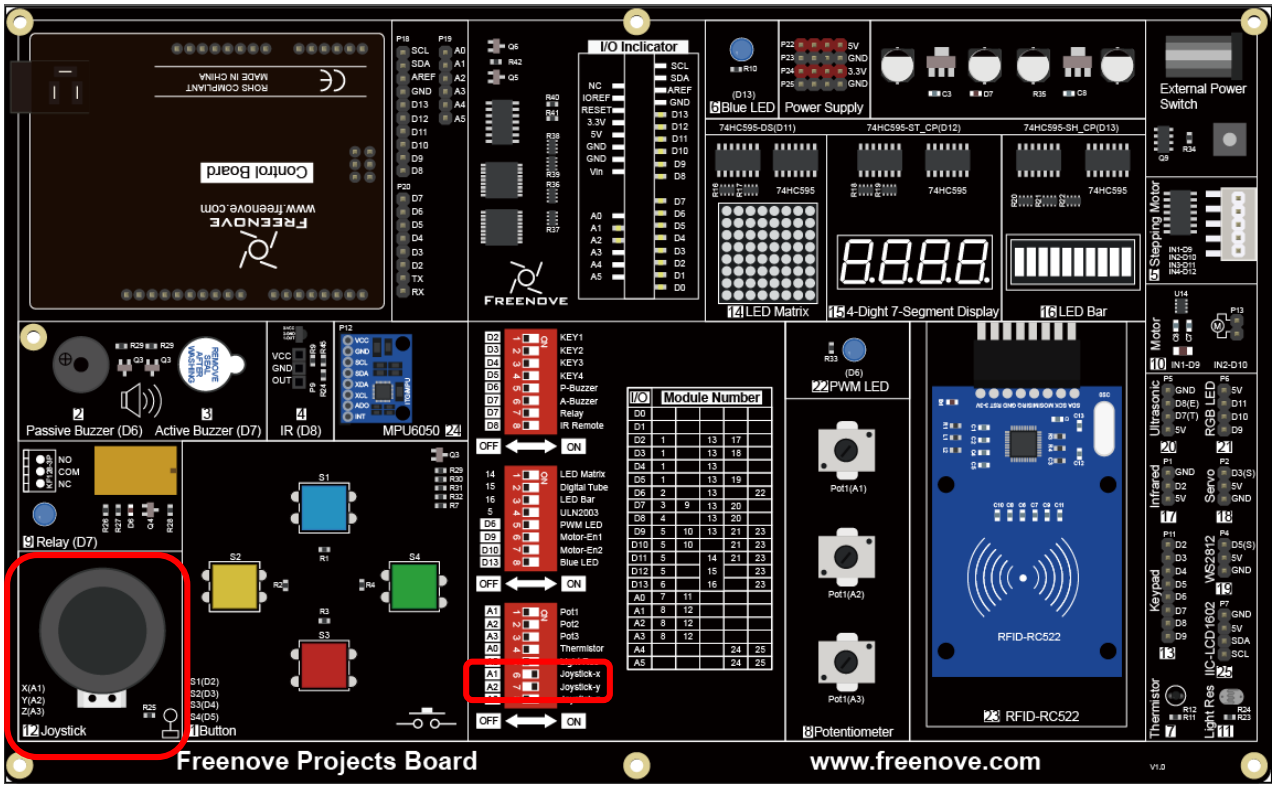
Schematic diagram



Hardware connection



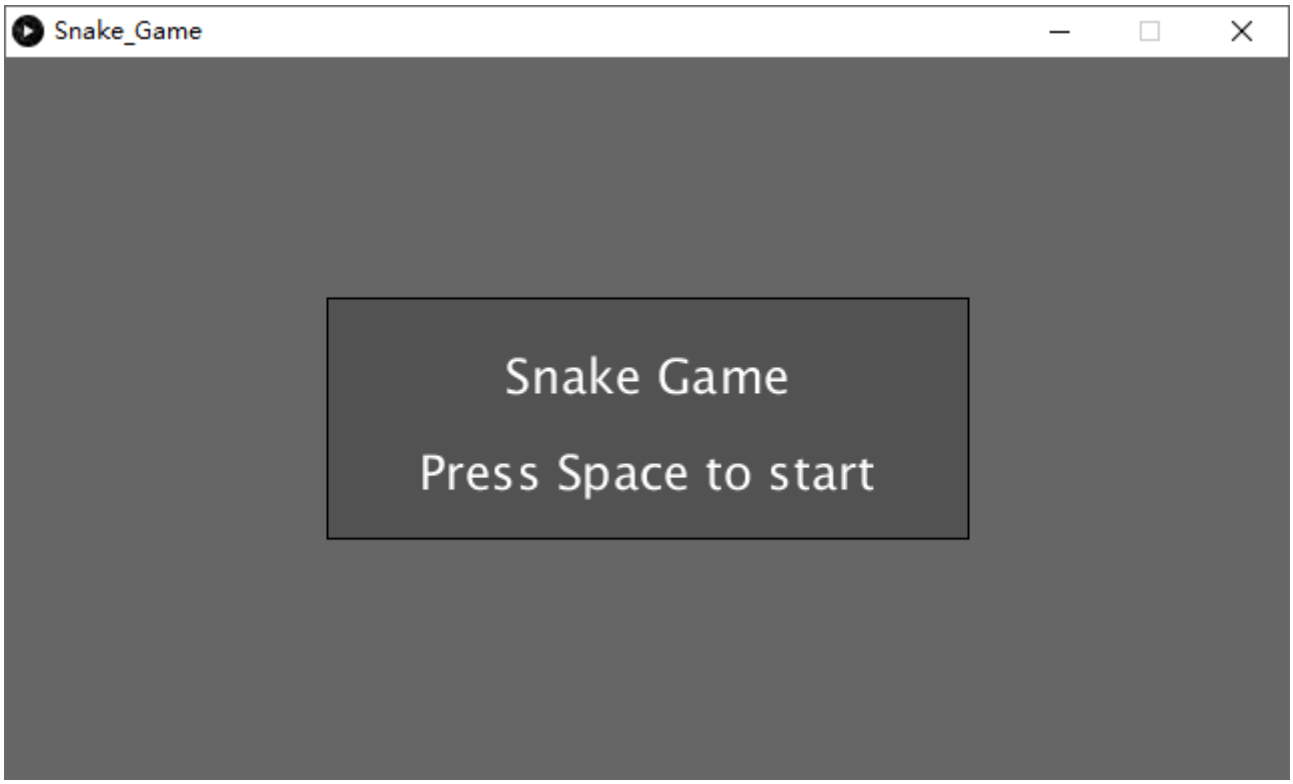
Hardware connection



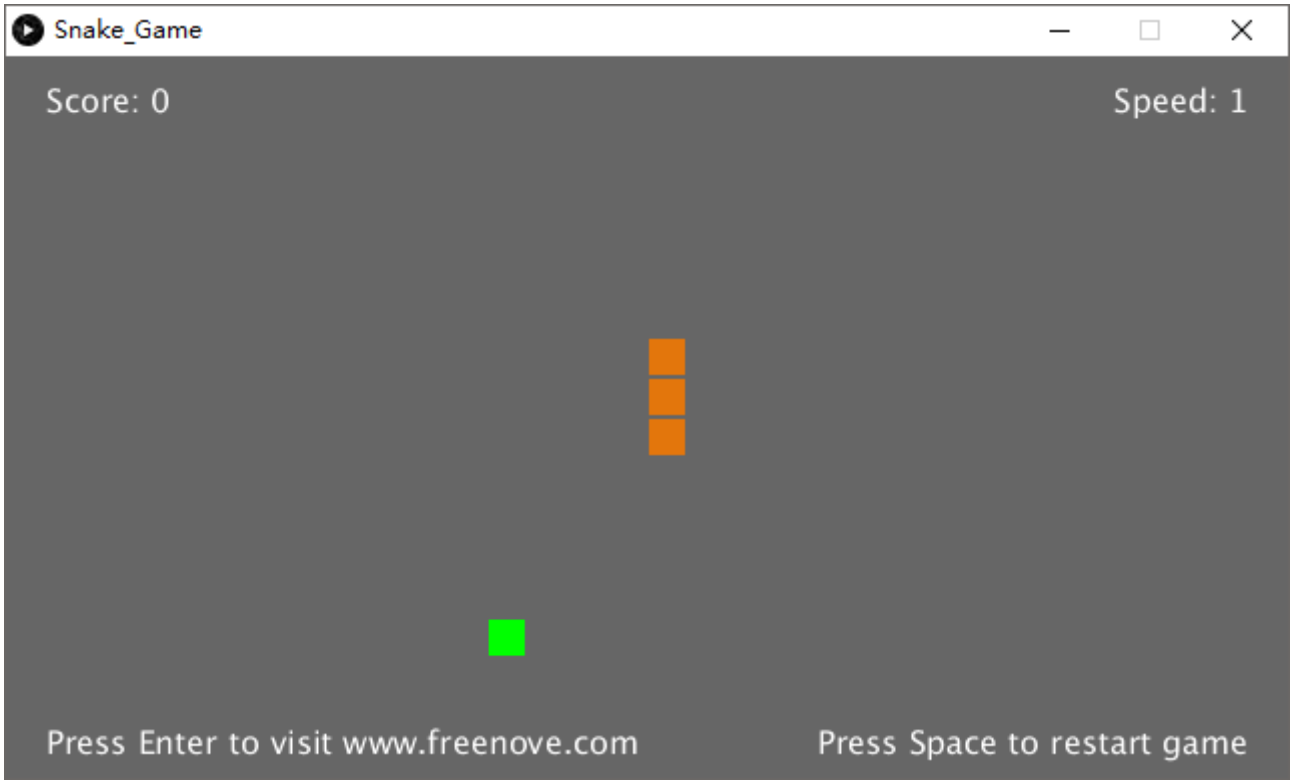
Sketch

Sketch Snake_Game

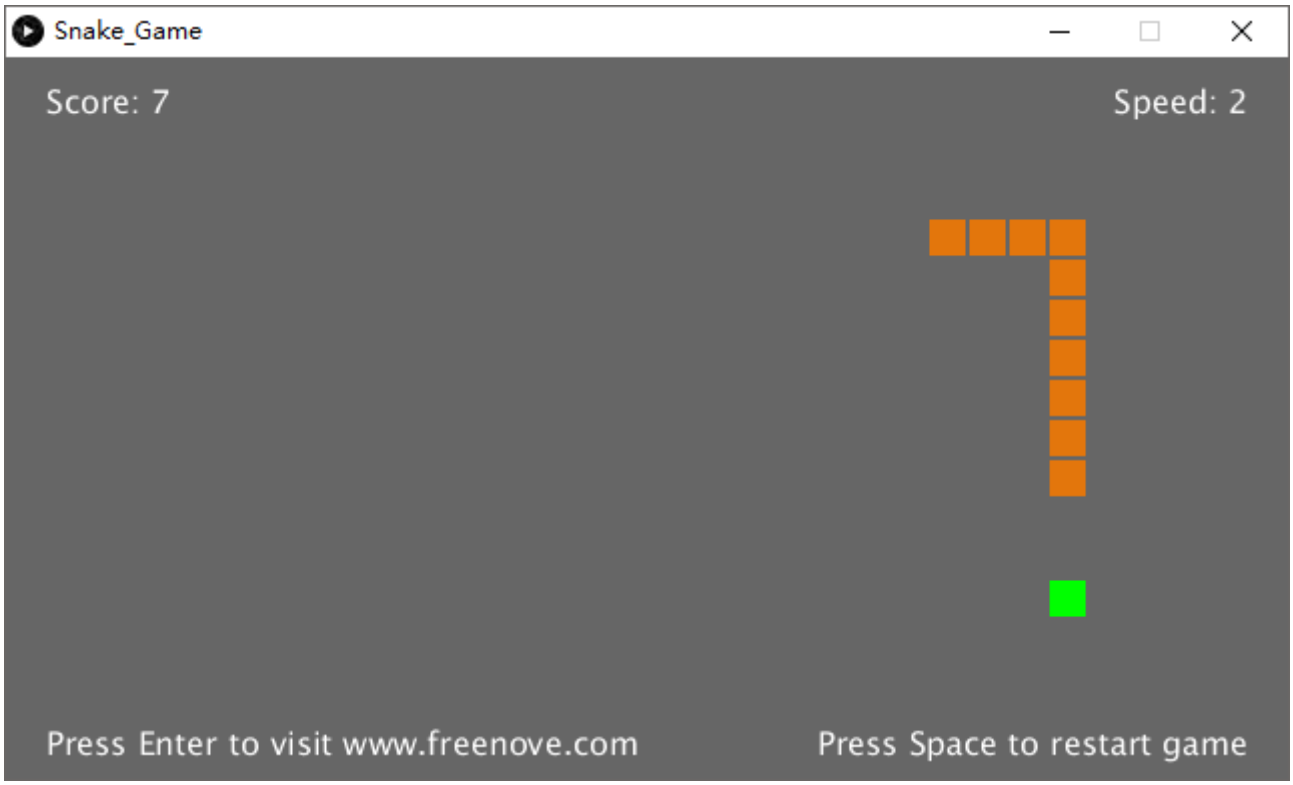
Use Processing to open Snake_Game.pde and click Run. If the connection succeeds, the follow will be shown:



Press the space bar on keyboard to start the game:



Shift the joystick to control the snake action. The game rules are the same as the classic snake game:



When the game fails, press space bar to restart the game:

Additionally, you can restart the game by pressing the space bar at any time.

Project 4.2 Snake Game 3D

Now, let's experience the 3D version game.

Component list

The same as last section.

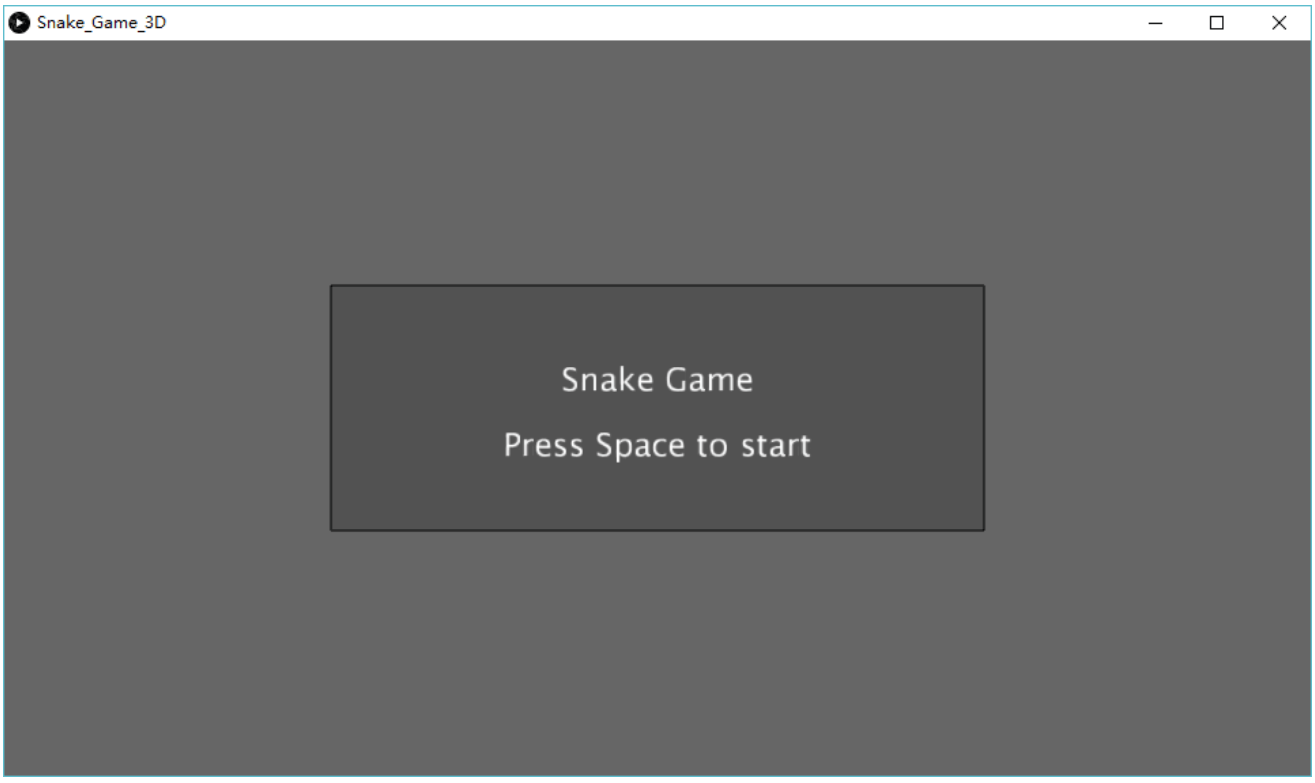
Circuit

The same as last section.

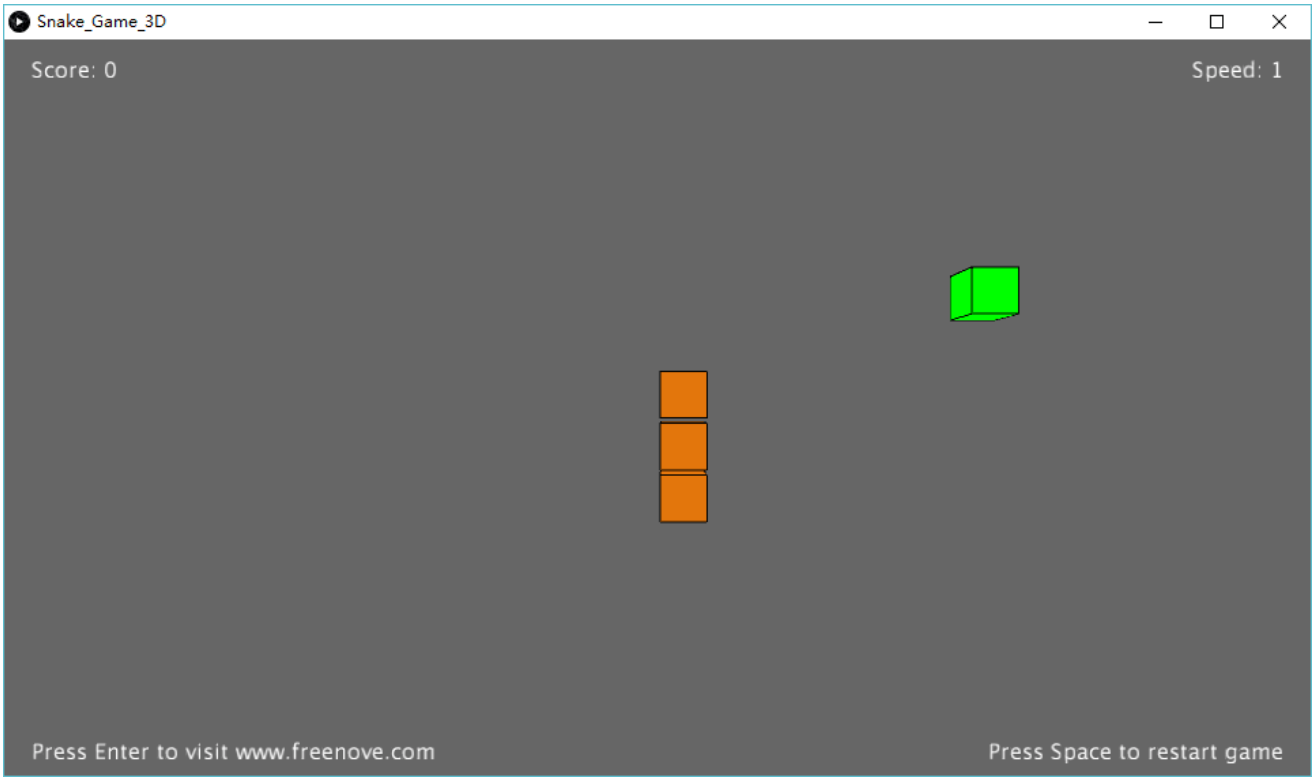
Sketch

Sketch Snake_Game_3D

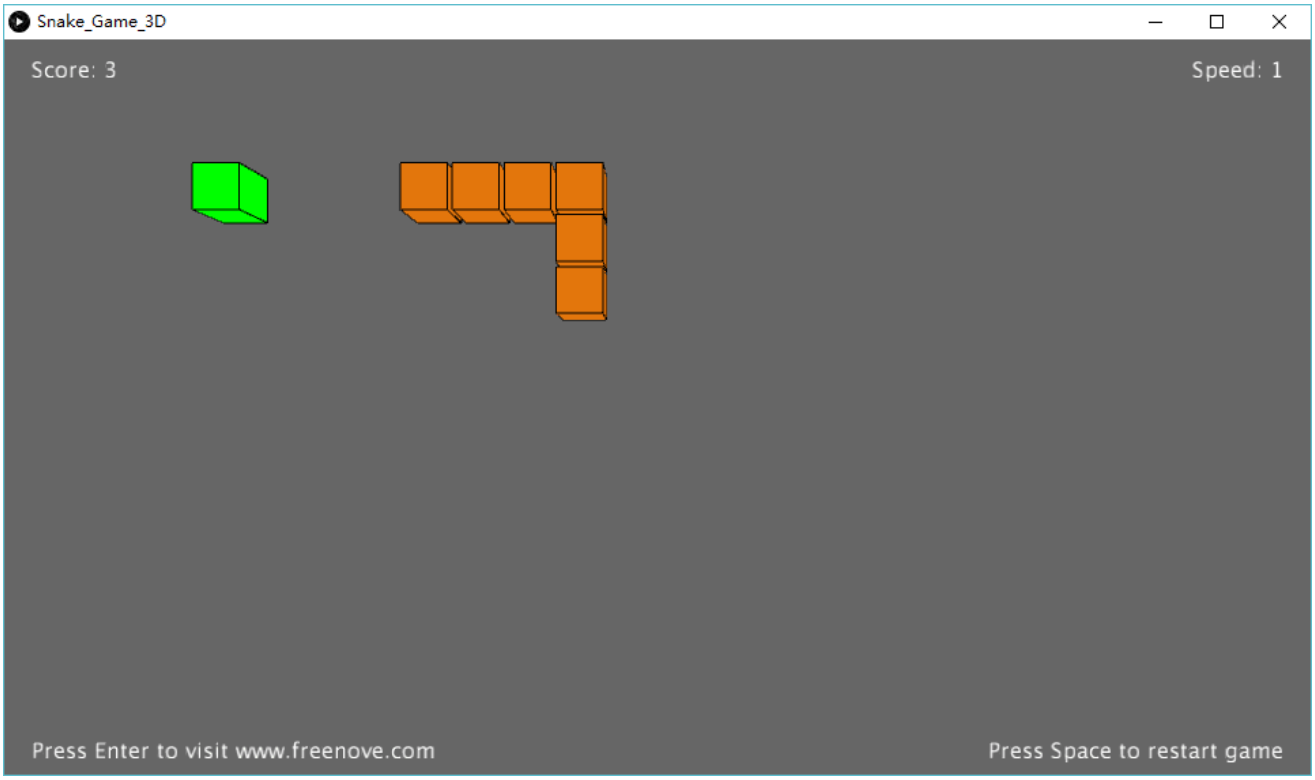
Use Processing to open Snake_Game_3D.pde and click Run. If the connection succeeds, the follow will be shown:



Press the space bar on keyboard to start the game:



Shift the joystick to control the snake action. The game rules are the same as the classic snake game:



The rest operation is the same as the 2D version.

Chapter 5 Pong Game

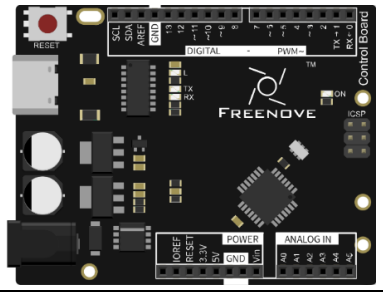
We have experienced single-player game snake before. Now, let's use connect board to play classic two-player pong game. You will experience both 2D and 3D version.

Project 5.1 Pong Game

First, let's experience the 2D version game.

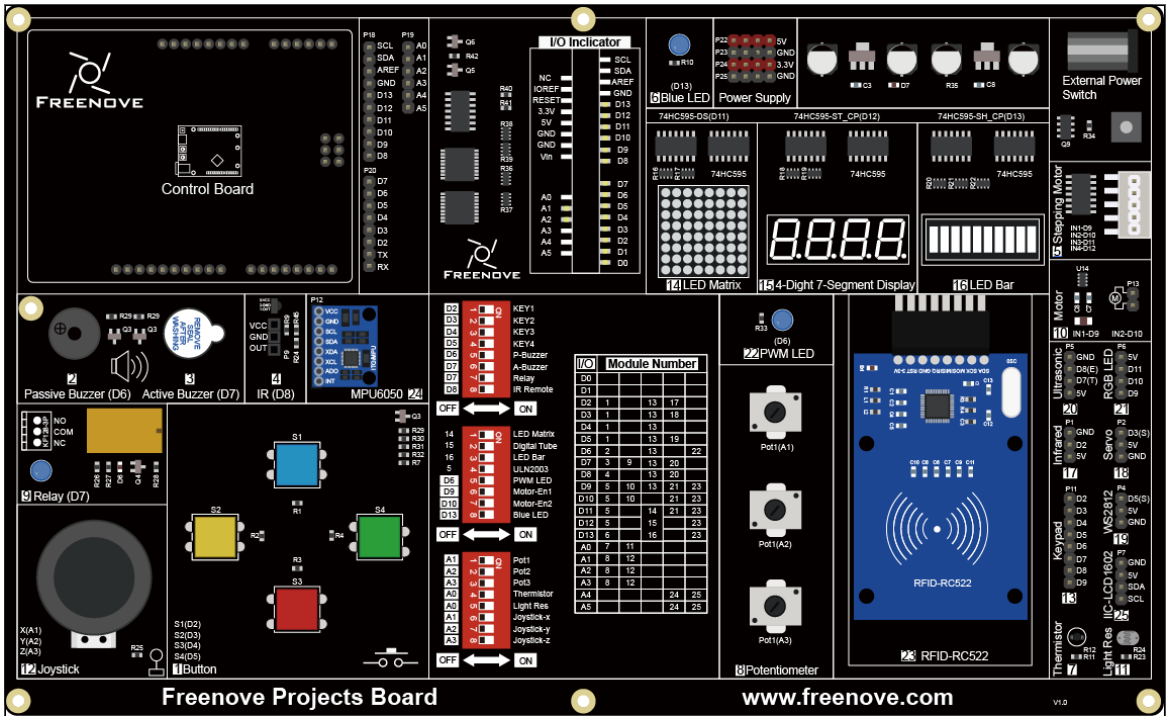
Component list

Control board x1



USB cable x1

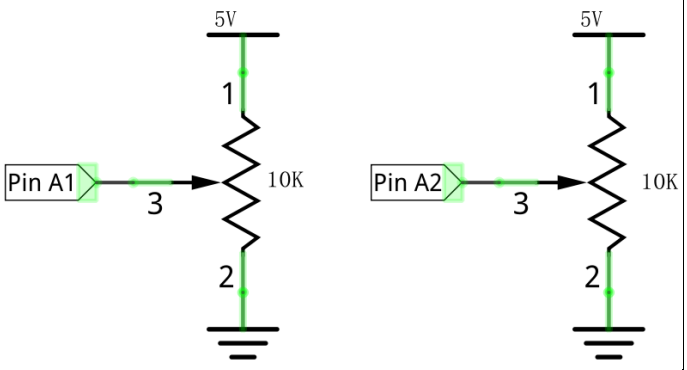
Freenove Projects Board



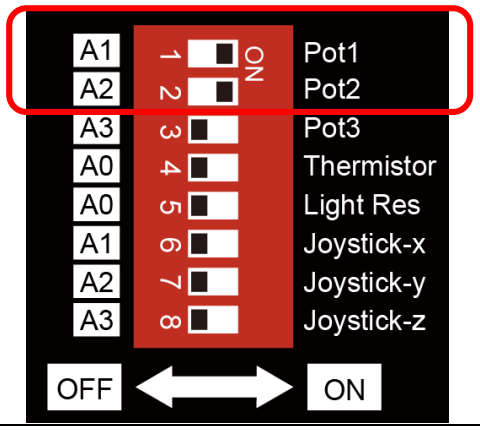
Circuit

Use A1, A2 ports on control board to detect the voltage of rotary potentiometers.

Schematic diagram

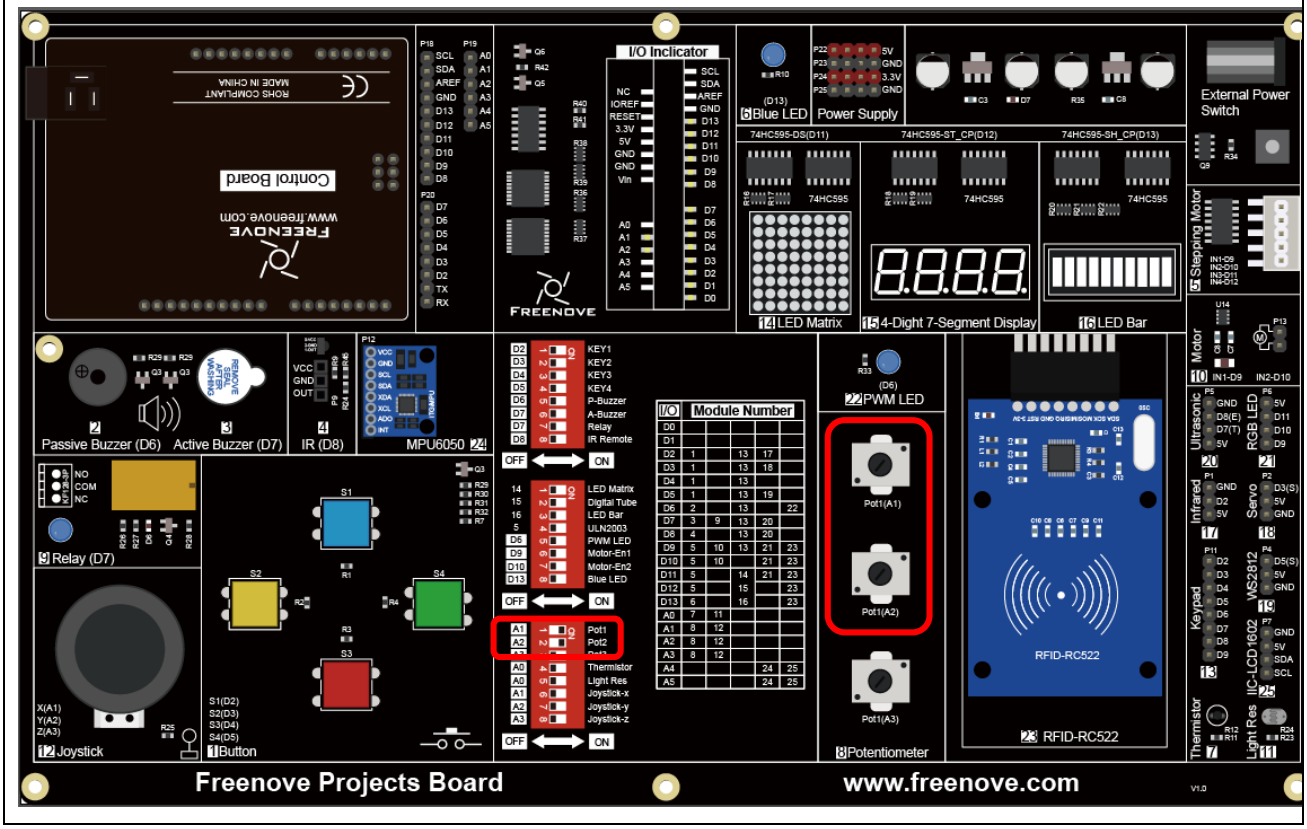


Hardware connection



Hardware connection

Insert the Control Board to Freenove Projects Board, and then turn the corresponding switch to the right(ON).

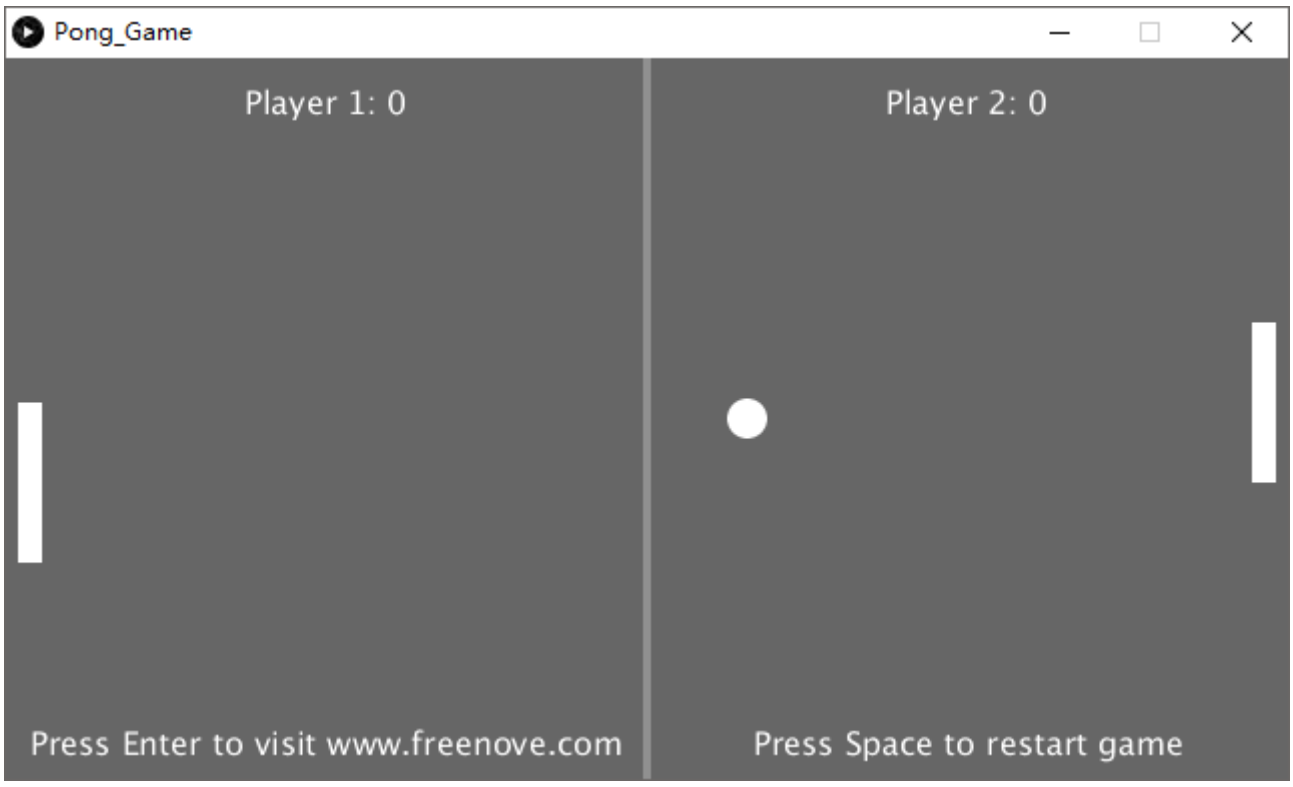


Sketch

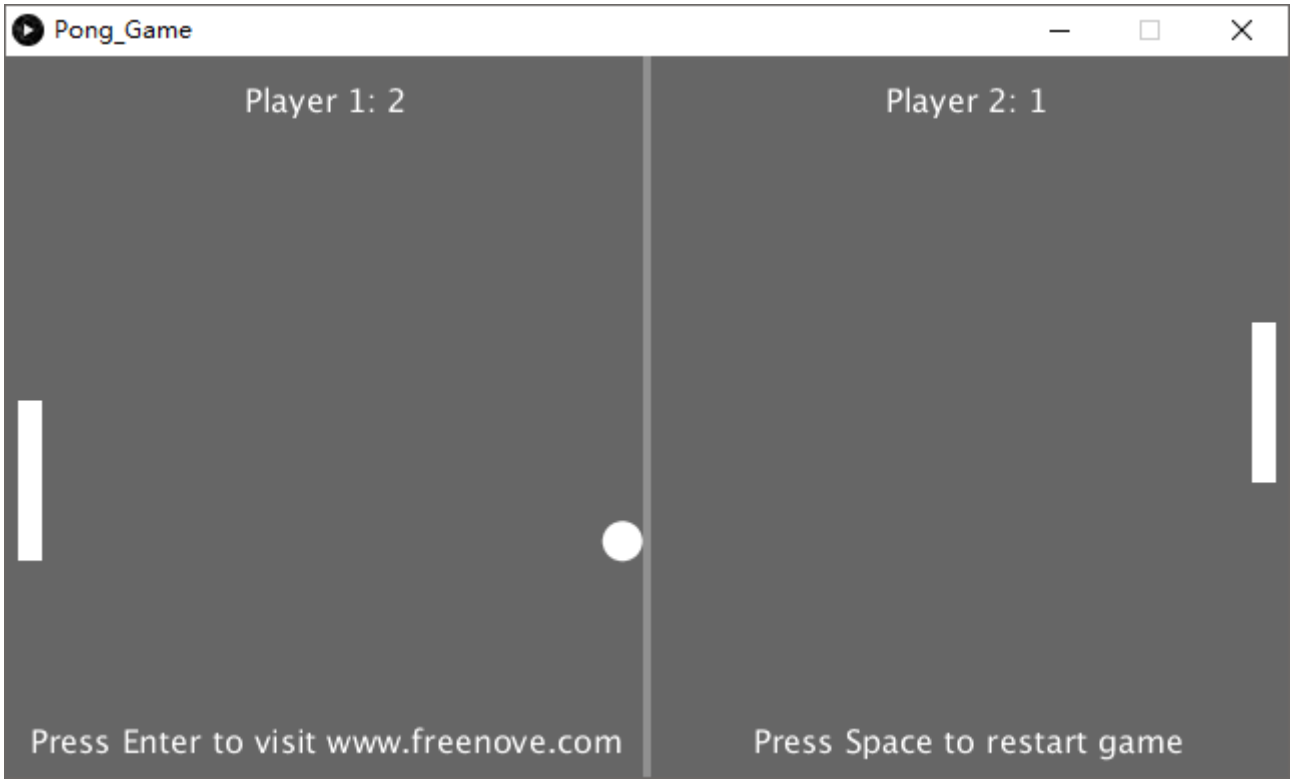
Sketch Pong_Game

Use Processing to open Pong_Game and click Run. If the connection succeeds, the follow will be shown:

Now you can try to turn the potentiometer to control the movement of paddle without ball. Press space bar to start the game:



Use potentiometer to control the movement of paddle to block the ball back. The game rules are the same as classic pong game:



The game will be over when one side reachs three points. Pressing the space bar can restart the game:

Additionally, you can restart the game by pressing the space bar at any time.

Project 5.2 Pong Game 3D

Now, let's experience the 3D version game.

Component list

The same as last section.

Circuit

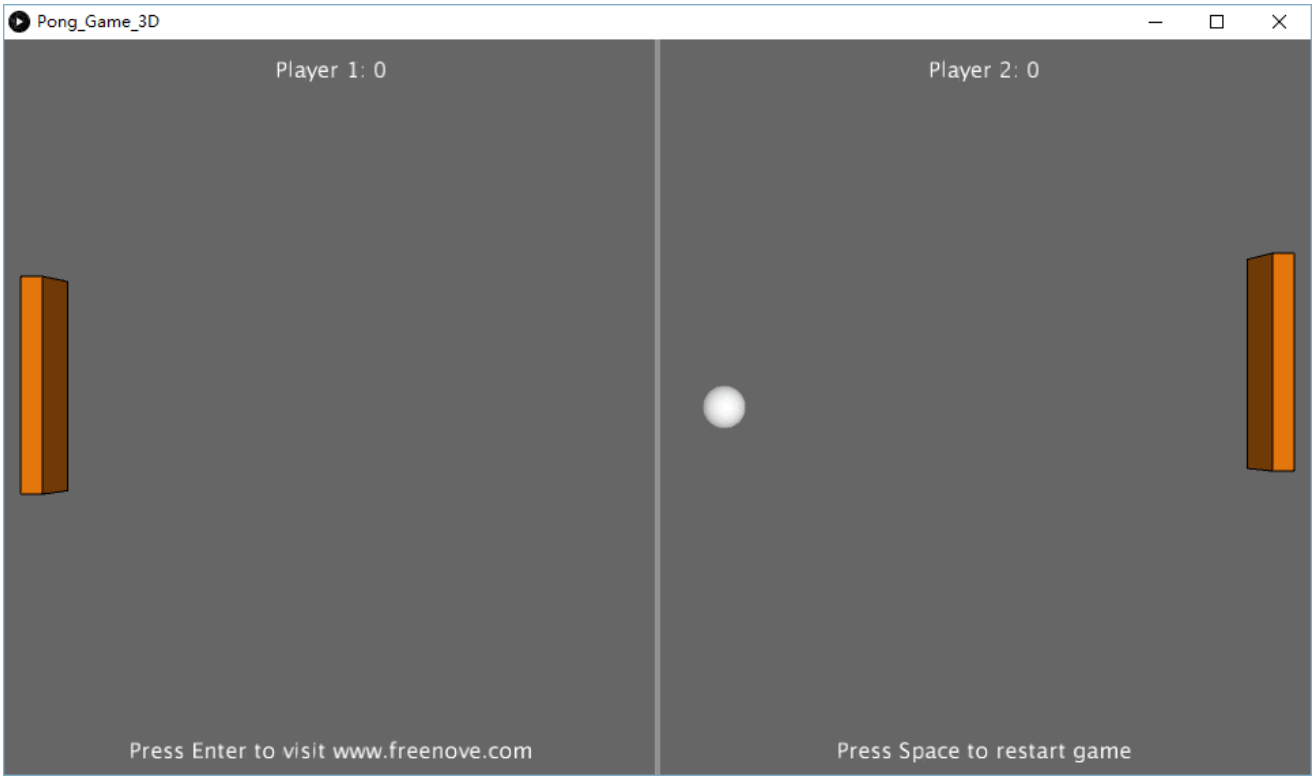
The same as last section.

Sketch

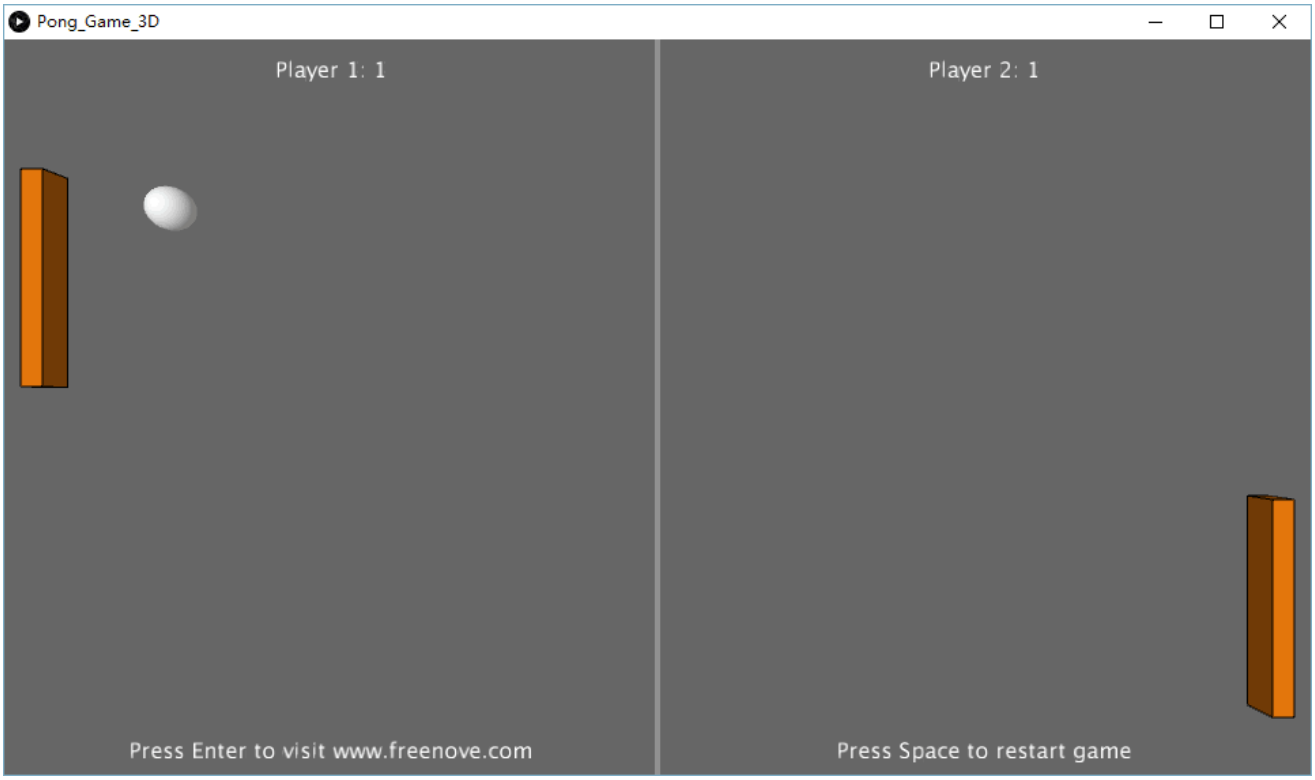
Sketch Pong_Game_3D

Use Processing to open Pong_Game_3D.pde and click Run. If the connection succeeds, the follow will be shown:

Now you can try to turn the potentiometer to control the movement of paddle without ball. Press space bar to start the game:



Use potentiometer to control the movement of paddle to block the ball back. The game rules are the same as classic pong game:



The rest operation is the same as the 2D version.

What's next?

Thanks for your reading!

This document is all over here. If you find any mistakes, omissions or you have other ideas or questions, please feel free to contact us. We would love to hear from you.

If you want to learn more about electronics and programming, interesting robots and projects, please continue to follow our website. We will continue to launch cost-effective, innovative and exciting products.

Thank you again for choosing Freenove products.