Welcome

Thank you for choosing Freenove products!

How to Start

When reading this, you should have downloaded the ZIP file for this product.
Unzip it and you will get a folder containing tutorials and related files. Please start with this PDF tutorial.

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

Get Support

Encounter problems? Don't worry! Refer to “TroubleShooting.pdf” or contact us.

When there are packaging damage, quality problems, questions encountering in use, etc., just send us an email. We will reply to you within one working day and provide a solution.

support@freenove.com

Attention

Pay attention to safety when using and storing this product:

- This product is not suitable for children under 12 years of age because of small parts and sharp parts.
- Minors should use this product under the supervision and guidance 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.
- To avoid personal injury, do not touch parts rotating or moving while working.
- The wrong operation may cause overheat. Do not touch and disconnect the power supply immediately.
- Operate in accordance with the requirements of the tutorial. Fail to do so may damage the parts.
- Store this product in a dry and dark environment. Keep away from children.
- 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 implement product prototypes, realize their creativity and launch innovative products. Our services include:

- Kits for learning programming and electronics
- Kits compatible with Arduino®, Raspberry Pi®, micro:bit®, ESP32®, etc.
- Kits for robots, smart cars, drones, etc.
- Components, modules and tools
- Design and customization

To learn more about us or get our latest information, please visit our website:

http://www.freenove.com

Copyright

It means you can use these files on your own derived works, in part or completely. But not for commercial use.

Freenove® brand and logo are trademarks of Freenove Creative Technology Co., Ltd. Must not be used without permission.

Other registered trademarks and their owners appearing in this document:

Arduino® is a trademark of Arduino LLC (https://www.arduino.cc/).
Raspberry Pi® is a trademark of Raspberry Pi Foundation (https://www.raspberrypi.org/).
micro:bit® is a trademark of Micro:bit Educational Foundation (https://www.microbit.org/).
ESPRESSIF® and ESP32® are trademarks of ESPRESSIF Systems (Shanghai) Co., Ltd (https://www.espressif.com/).

Contents

Preface

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

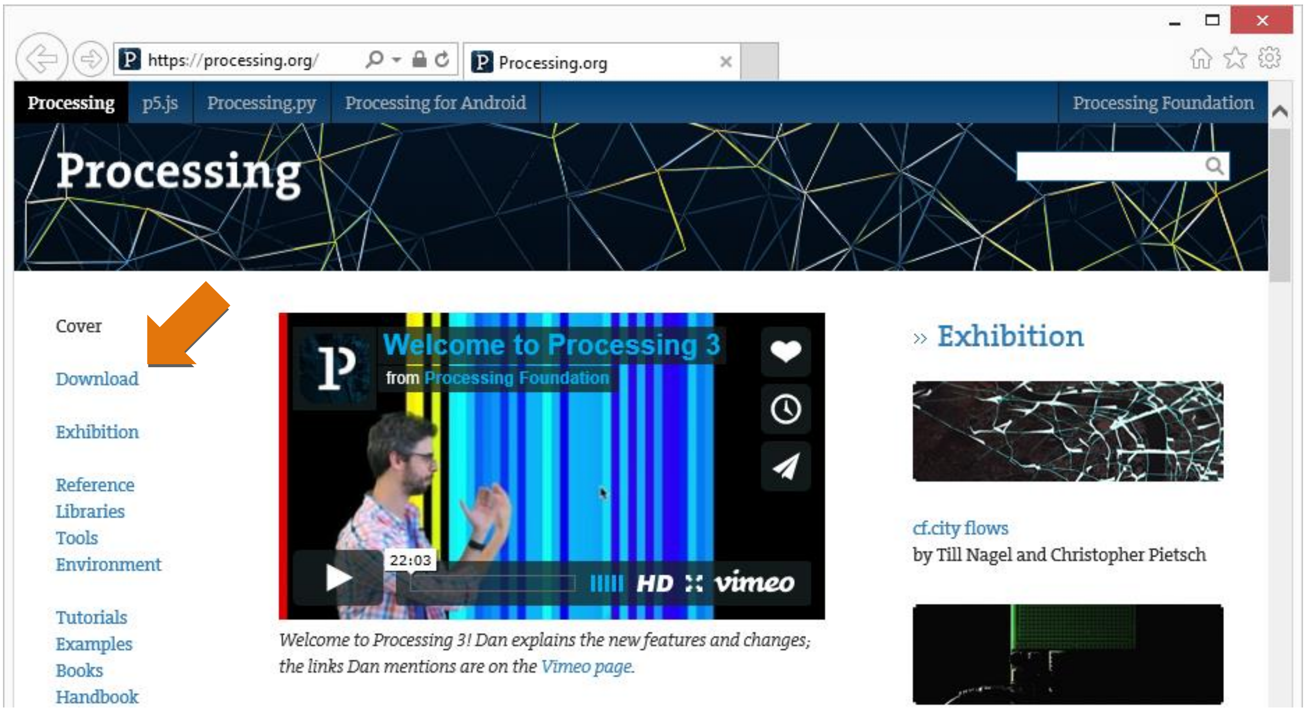
This document will show you how to use Processing to write programs to communicate with the control board. By doing this, we can make virtual instruments, game consoles and other projects.

Processing Software

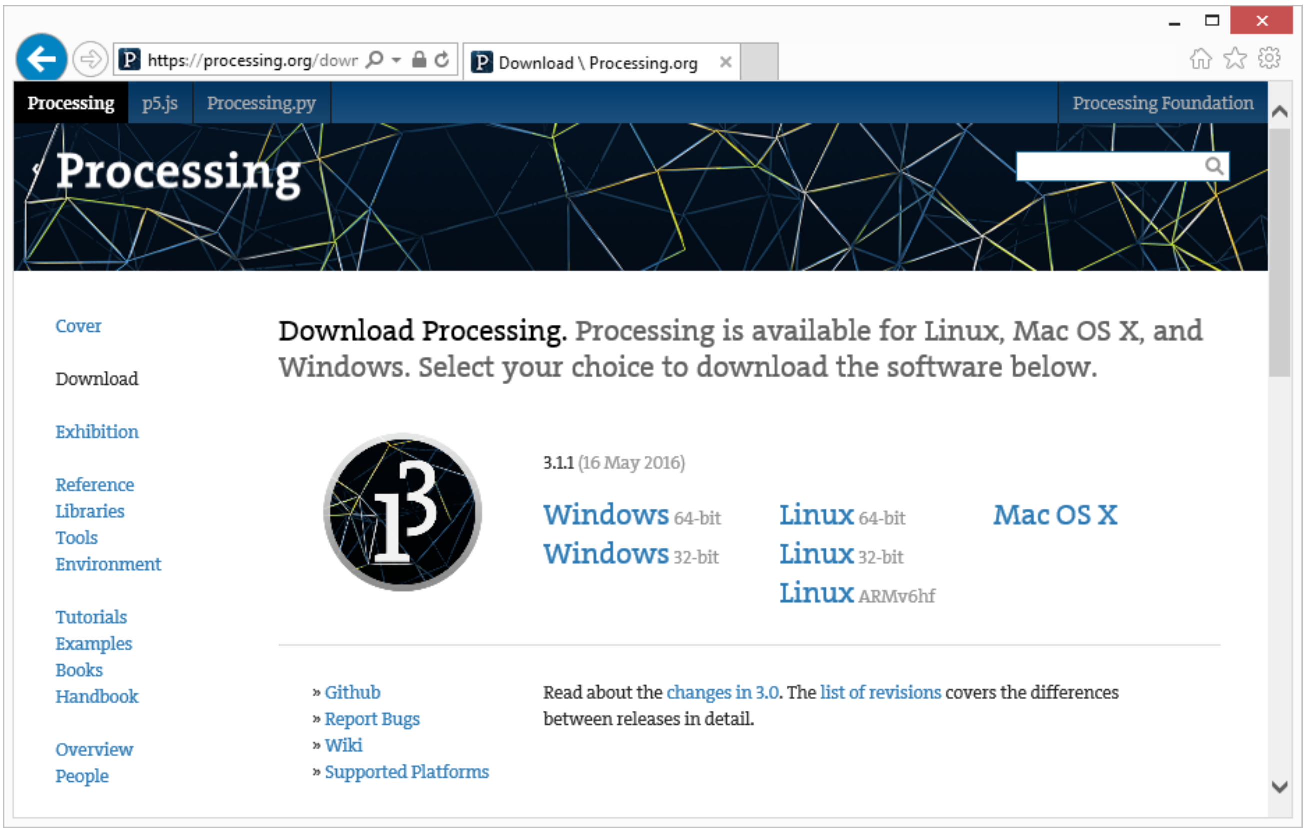
Processing software / Processing Development Environment (PDE) makes it easy to write programs.

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.



Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below.

Read about the changes in 3.0. The list of revisions covers the differences between releases in detail.

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
  ```

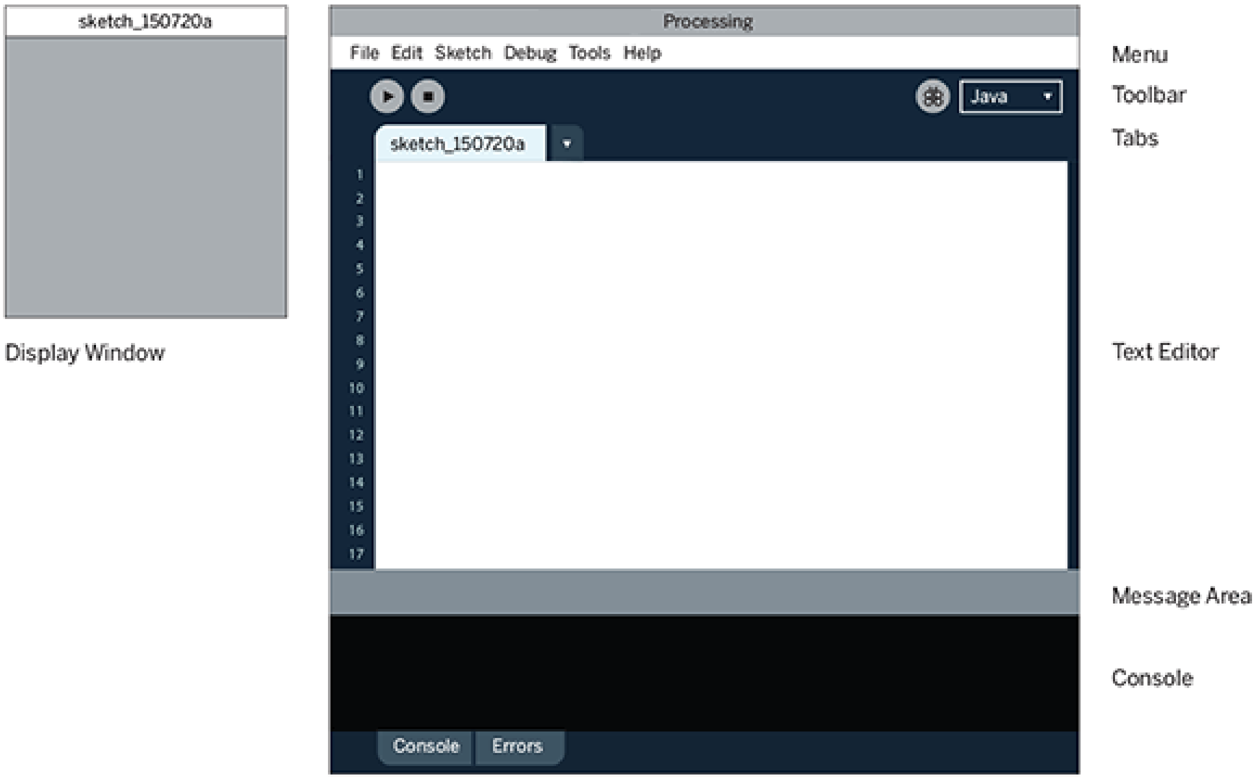
  (Replace xxxx with the rest of the file's name, which is the version number.) This will create a folder named processing-2.0 or something similar. Then change to that directory:

  ```
  cd processing-xxxx
  ```

  and run it:

  ```
  ./processing
  ```

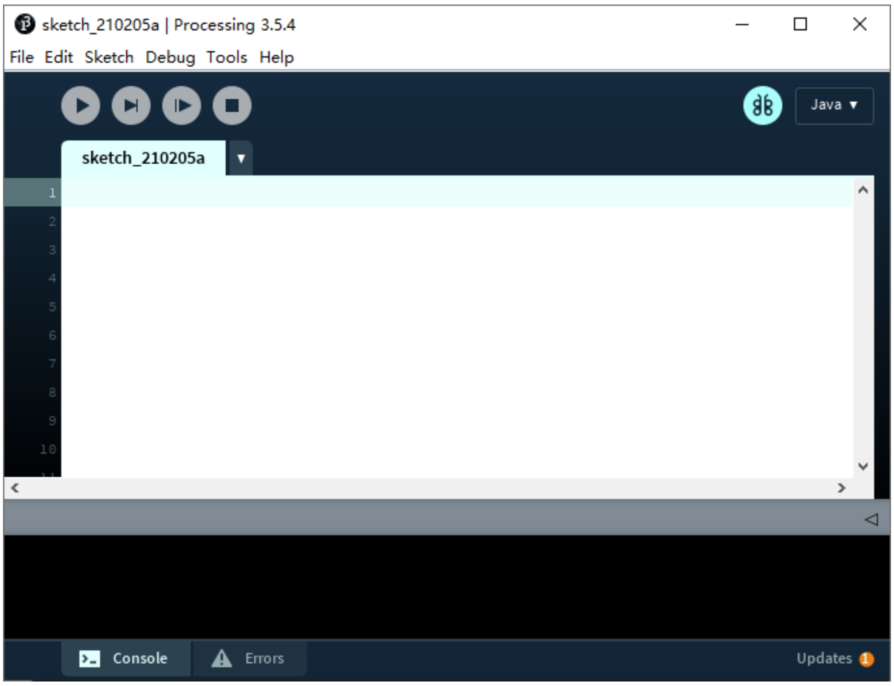
With any luck, the main Processing window will now be visible. Everyone's setup is different, so if the program didn't start, or you're otherwise stuck, visit the troubleshooting page for possible solutions.



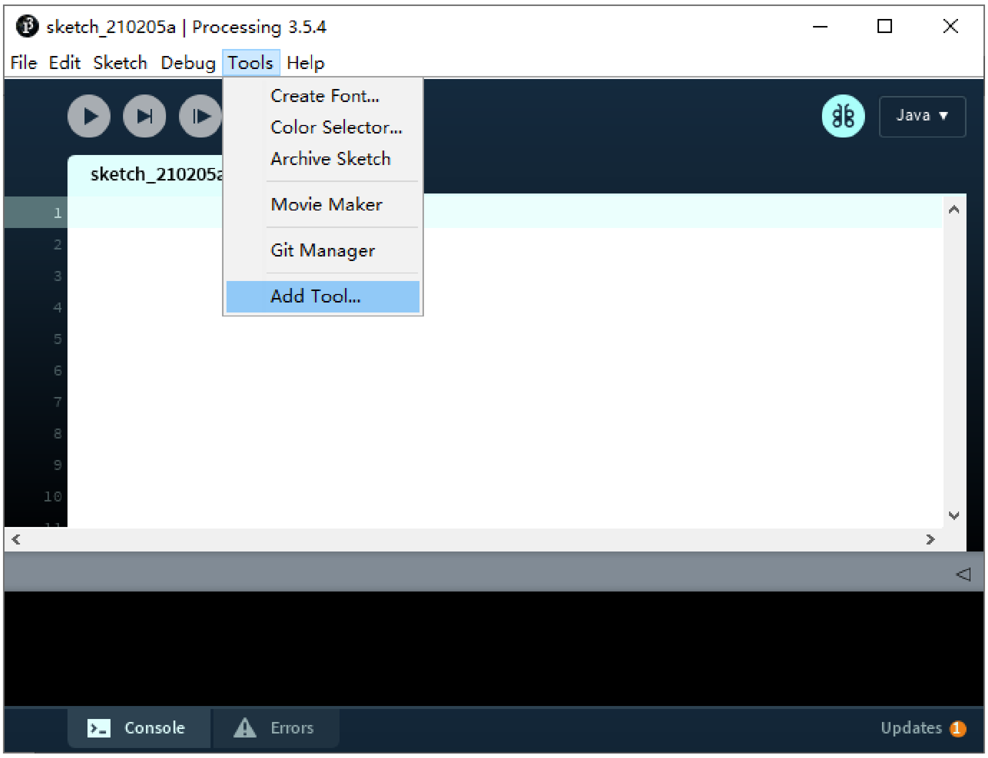
You're now running the Processing Development Environment (or PDE). There's not much to it; the large area is the Text Editor, and there's a row of buttons across the top; this is the toolbar. Below the editor is the Message Area, and below that is the Console. The Message Area is used for one line messages,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

Install ControlP5

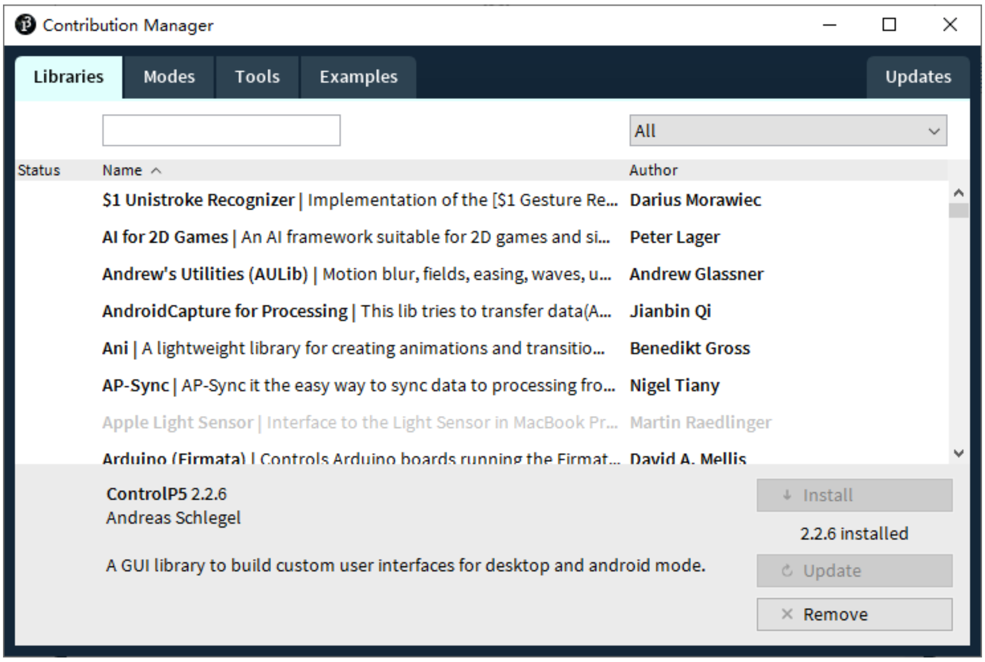
Open Processing.


sketch_210205a | Processing 3.5.4
File Edit Sketch Debug Tools Help
Java
sketch_210205a
Console
Errors
Updates 1

Click Add Tool under Tools.



Select Libraries in the pop-up window.



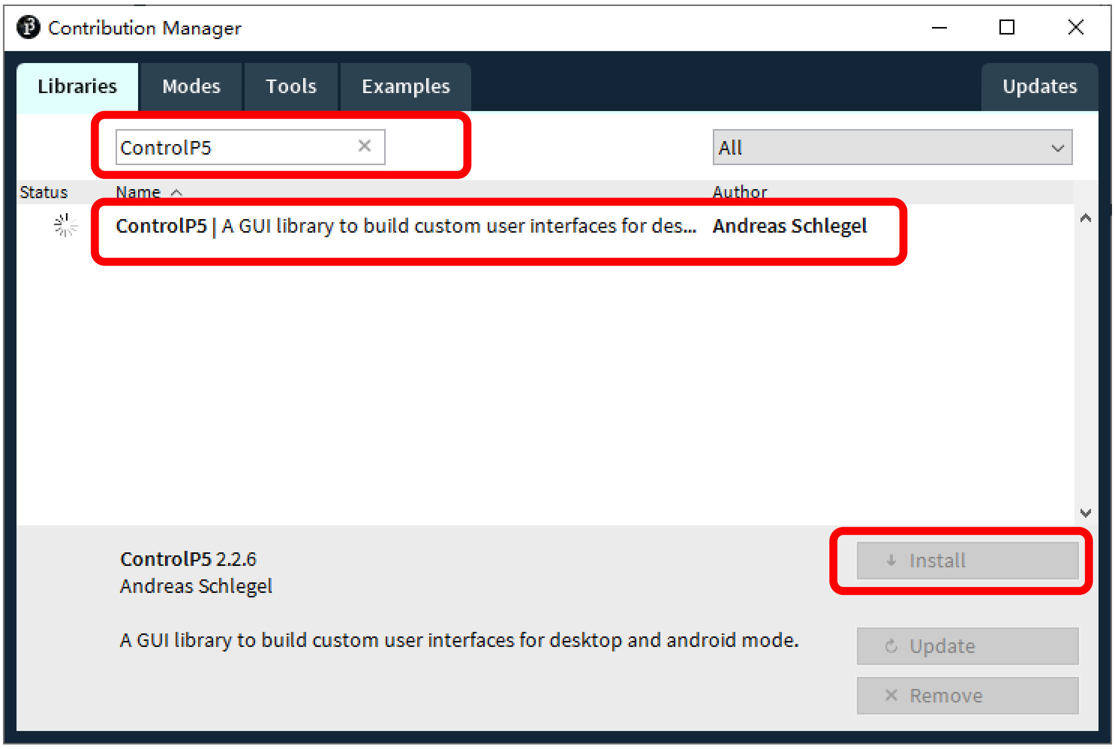
Status	Name	Author
	$1 Unistroke Recognizer | Implementation of the [$1 Gesture Re...	**Darius Morawiec**
	AI for 2D Games | An AI framework suitable for 2D games and si...	**Peter Lager**
	Andrew's Utilities (AULib) | Motion blur, fields, easing, waves, u...	**Andrew Glassner**
	AndroidCapture for Processing | This lib tries to transfer data(A...	**Jianbin Qi**
	Ani | A lightweight library for creating animations and transitio...	**Benedikt Gross**
	AP-Sync | AP-Sync it the easy way to sync data to processing fro...	**Nigel Tiany**
	Apple Light Sensor | Interface to the Light Sensor in MacBook Pr...	**Martin Raedlinger**
	Arduino (Firmata) | Controls Arduino boards running the Firmat...	**David A. Mellis**

ControlP5 2.2.6
Andreas Schlegel

A GUI library to build custom user interfaces for desktop and android mode.

2.2.6 installed

Input “ControlP5” in the searching box, and then select the option as below. Click “Install” and wait for the installation to finish.



ControlP5 | A GUI library to build custom user interfaces for des... Andreas Schlegel

ControlP5 2.2.6
Andreas Schlegel

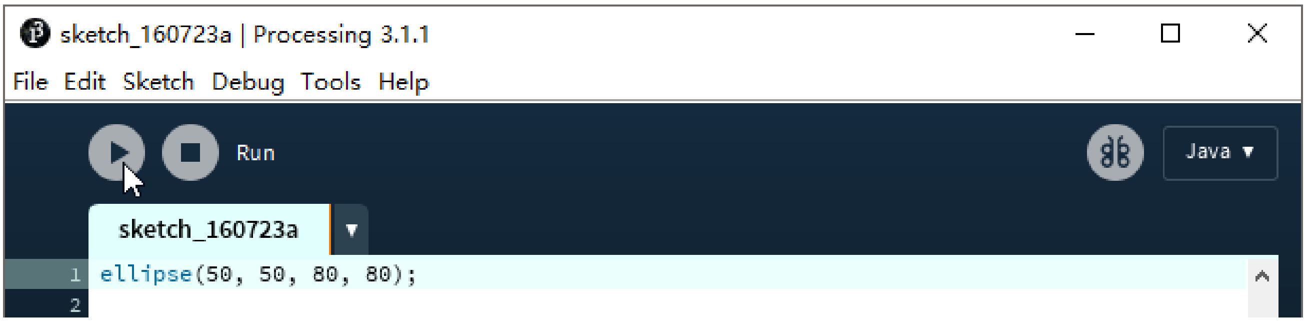
A GUI library to build custom user interfaces for desktop and android mode.

First Use

In the editor, type the following:

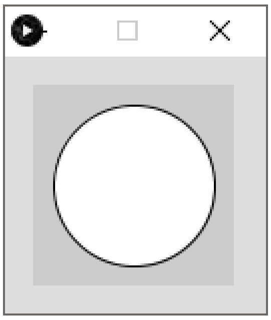
```
ellipse(50, 50, 80, 80);
```

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).

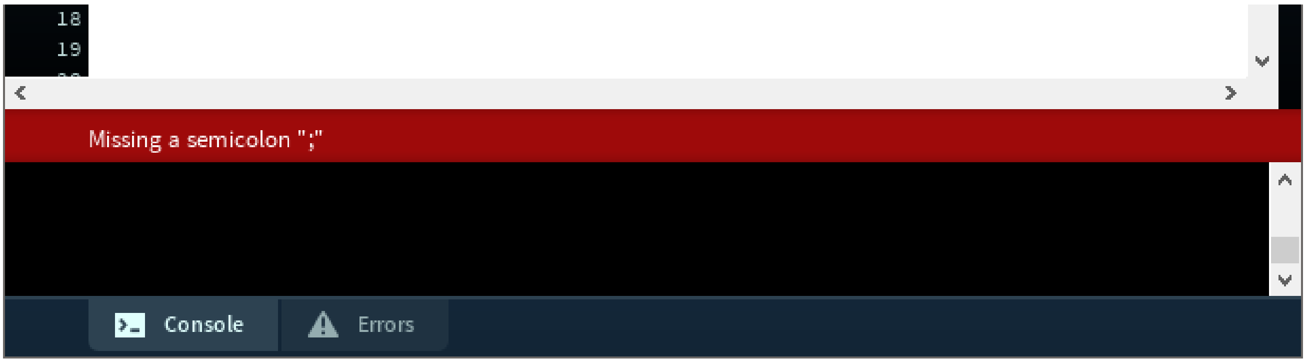


```
ellipse(50, 50, 80, 80);
```

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

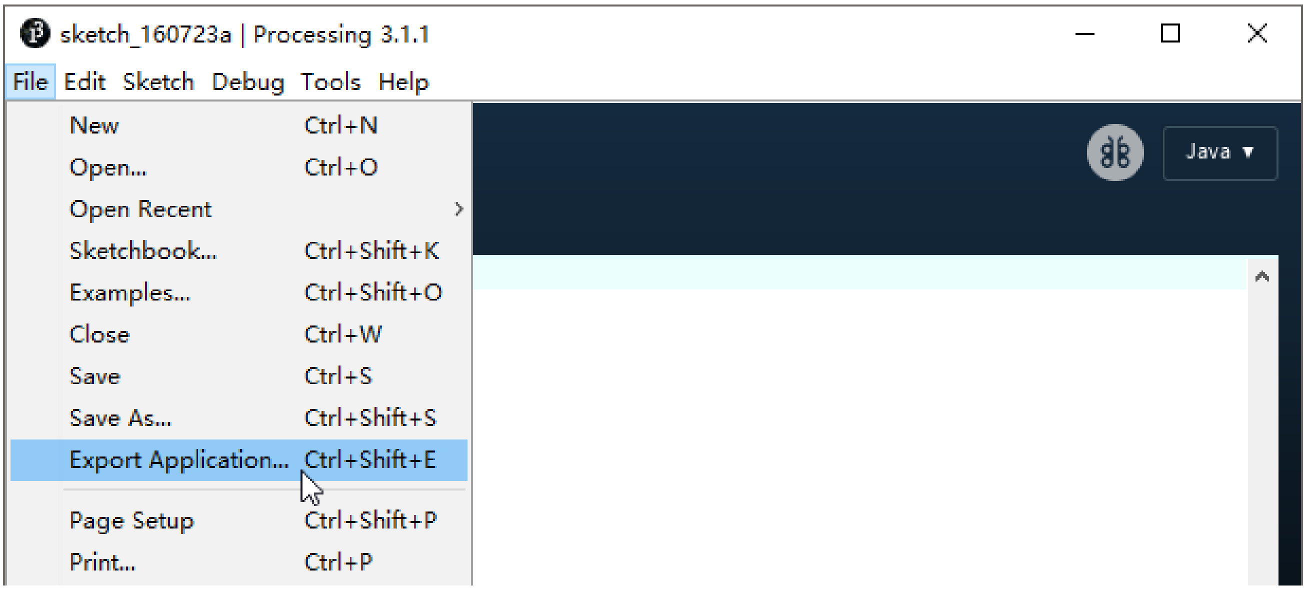


If you didn't type it correctly, the Message Area will turn red and complain about an error. If this happens, make sure that you've copied the example code exactly: the numbers should be contained within parentheses and have commas between each of them, and the line should end with a semicolon.



Missing 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.

Communication protocol

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

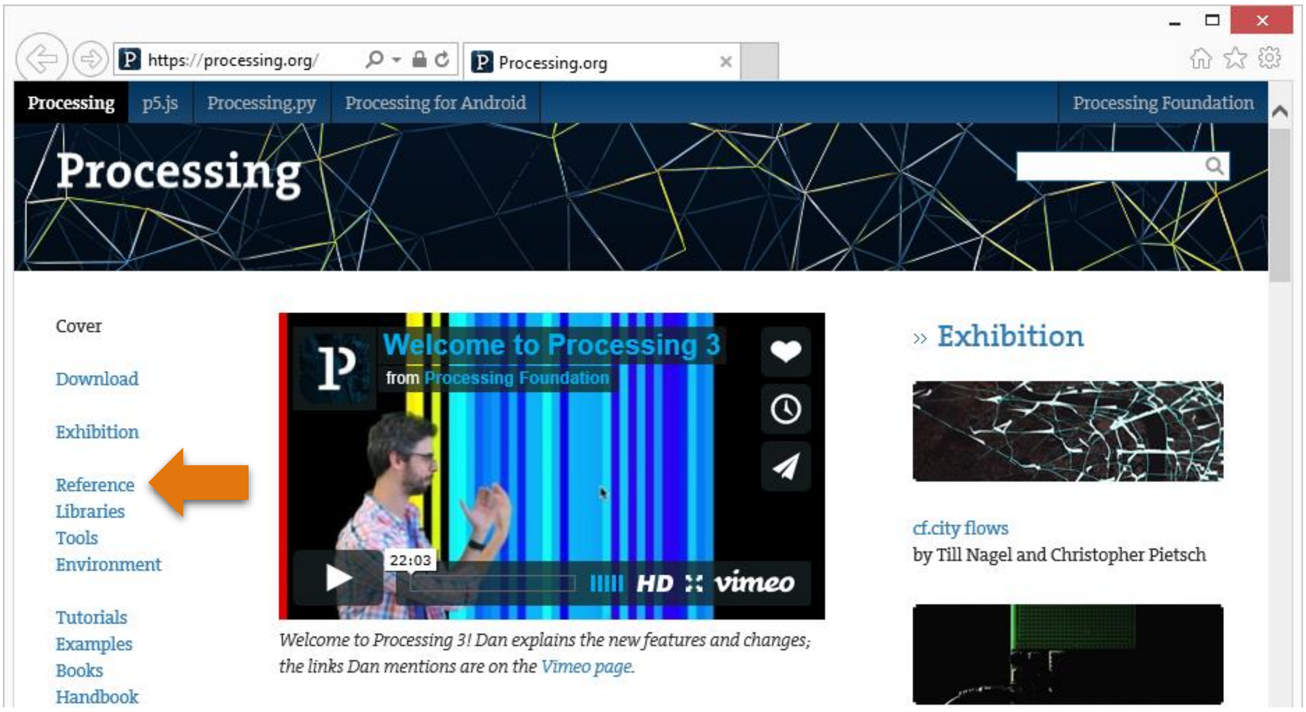
In order to simplify and facilitate the operation, we prepared a SerialDevice class for Processing to communicate with the control board. To use this class, we need to upload the following sketch to the control board:
Freenove Ultimate Starter Kit for ESP32\C\Processing\SerialDevice\SerialDevice.ino.
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.

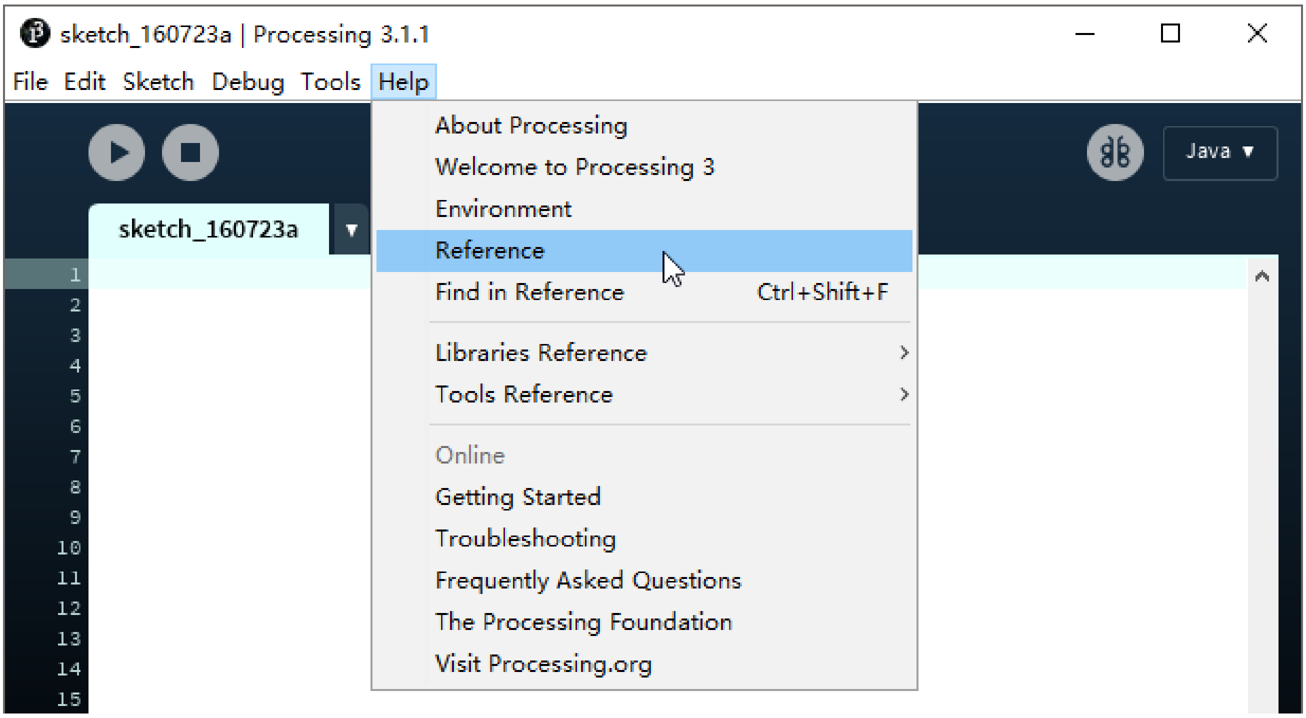
Here, Processing sketch code will not be introduced in detail. Interested readers can learn it by yourself.

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



Welcome to Processing 3! Dan explains the new features and changes; the links Dan mentions are on the Vimeo page.

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



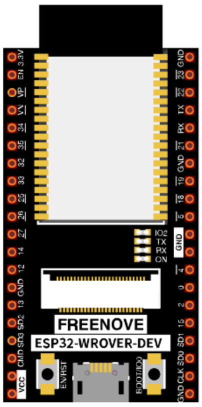
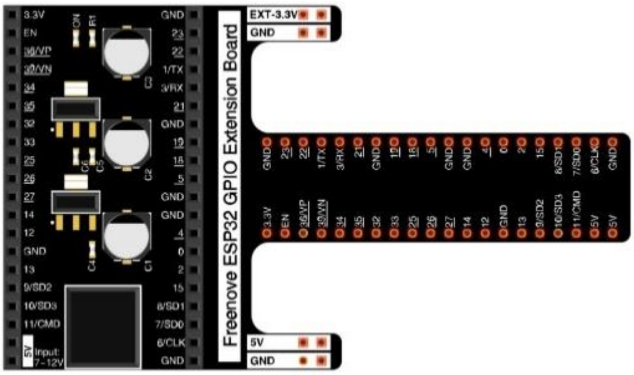
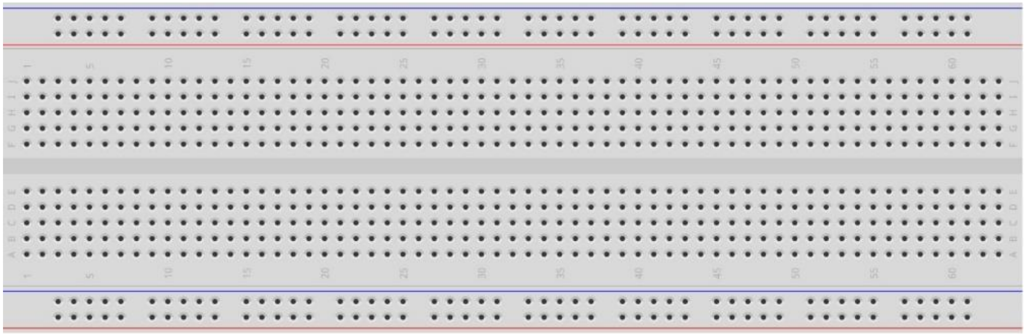
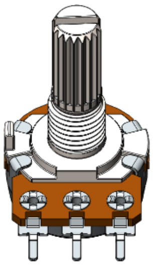
Chapter 1 Oscilloscope

In this chapter, we will make a 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 1.1 Oscilloscope

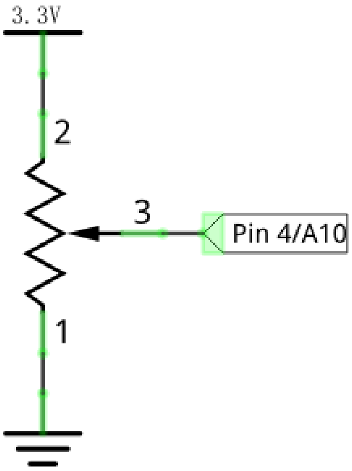
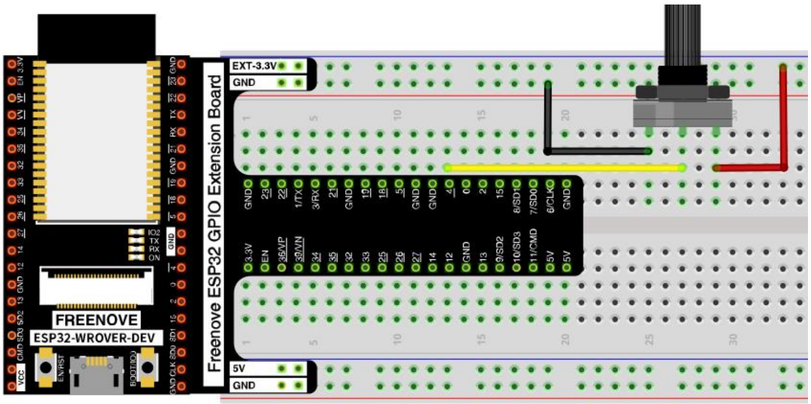
Now, let's use Processing and ESP32 board to create an oscilloscope.

Component list

ESP32 board x1  	GPIO Extension Board x1  Freenove ESP32 GPIO Extension Board EXT-3.3V GND 5V GND
BreadBoard x1 	
Jumper M/M x3 	Rotary potentiometer x1 

Circuit

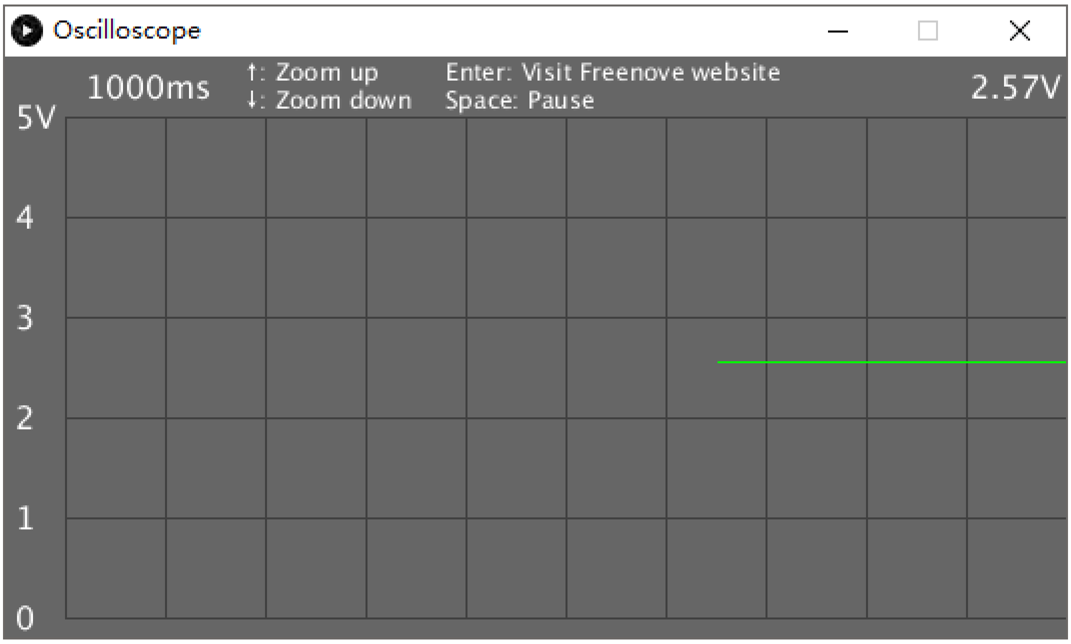
Use Pin4 (A10) port to detect the voltage of rotary potentiometer.

Schematic diagram	Hardware connection
	

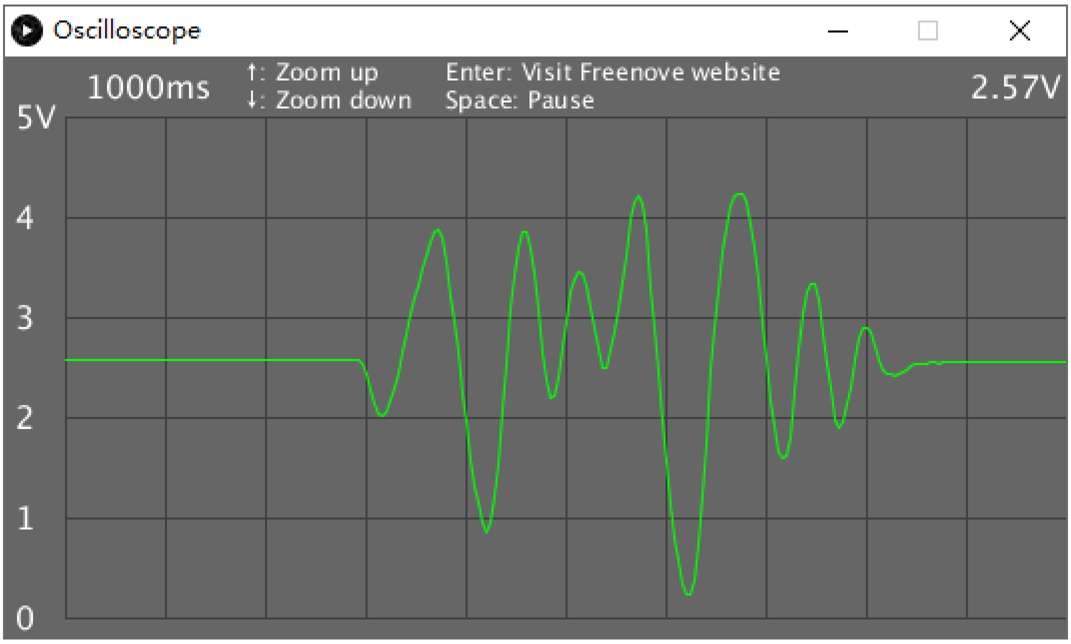
Sketch

Sketch Oscilloscope

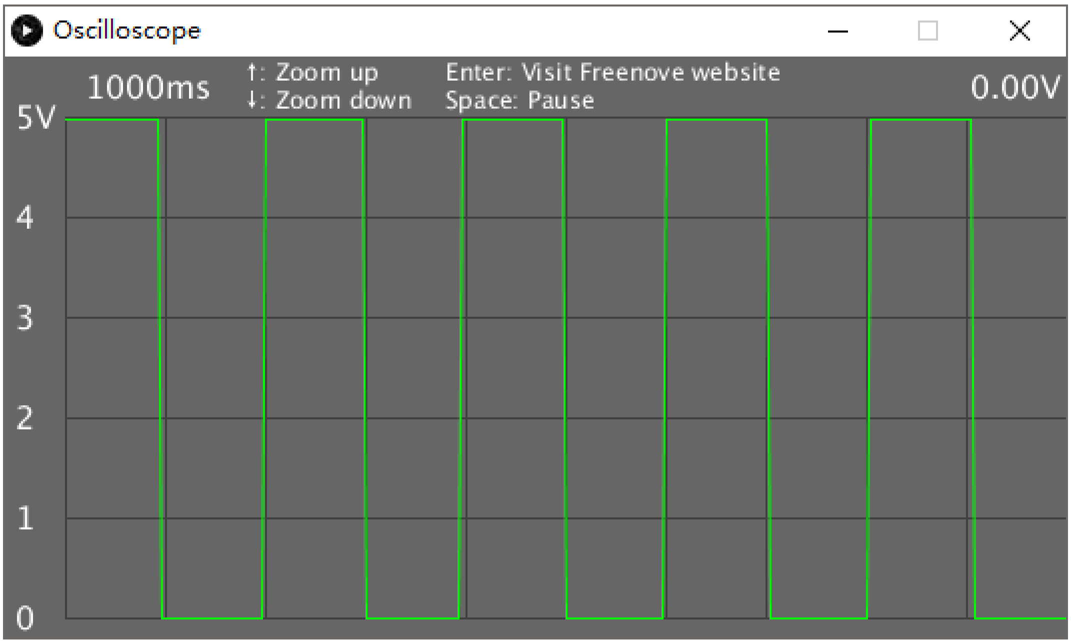
Use Processing to open **Freenove Ultimate Starter Kit for ESP32\C\Processing\Processing\Oscilloscope\ Oscilloscope.pde** and click Run. If the connection succeeds, the follow will be shown:



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



Disconnect the A10 port from the potentiometer and connect it to the Pin 2 port. Pin 2 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.

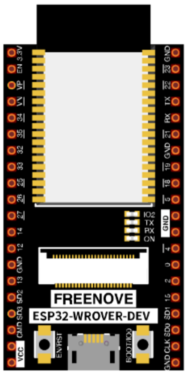
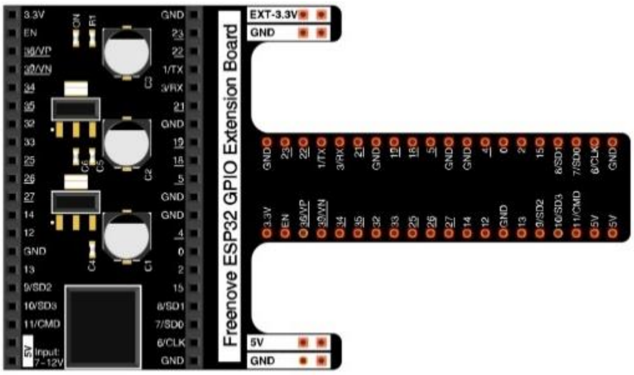
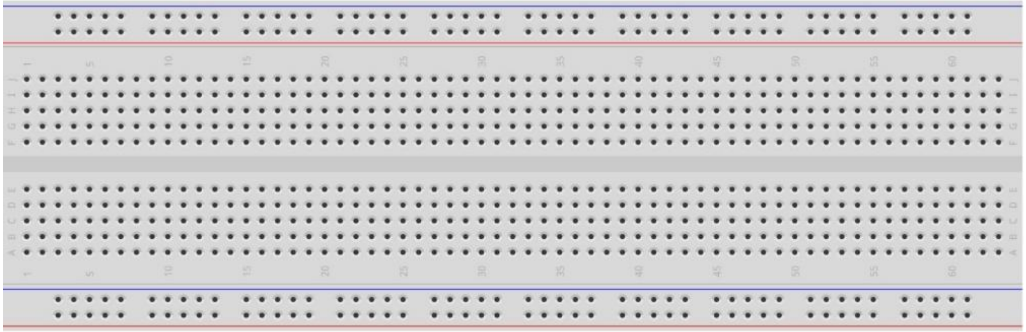
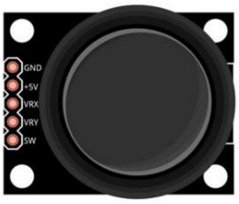
Chapter 2 Control 2D and 3D Figures

In this chapter, we will use ESP32 board to make Processing program control figure changes. And we will control 2D and 3D figures, respectively.

Project 2.1 Ellipse

First, control a 2D figure.

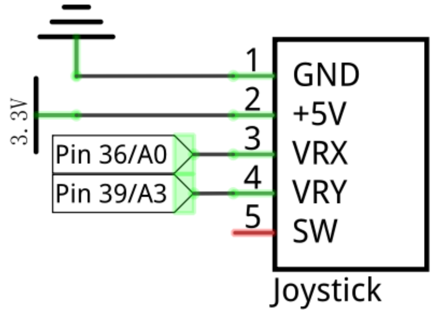
Component list

<table>
<tr><td>ESP32 board x1
 </td><td>GPIO Extension Board x1
 </td></tr>
<tr><td colspan="2">BreadBoard x1
</td></tr>
<tr><td>Jumper M/M x5
 </td><td>Joystick x1
 </td></tr>
</table>

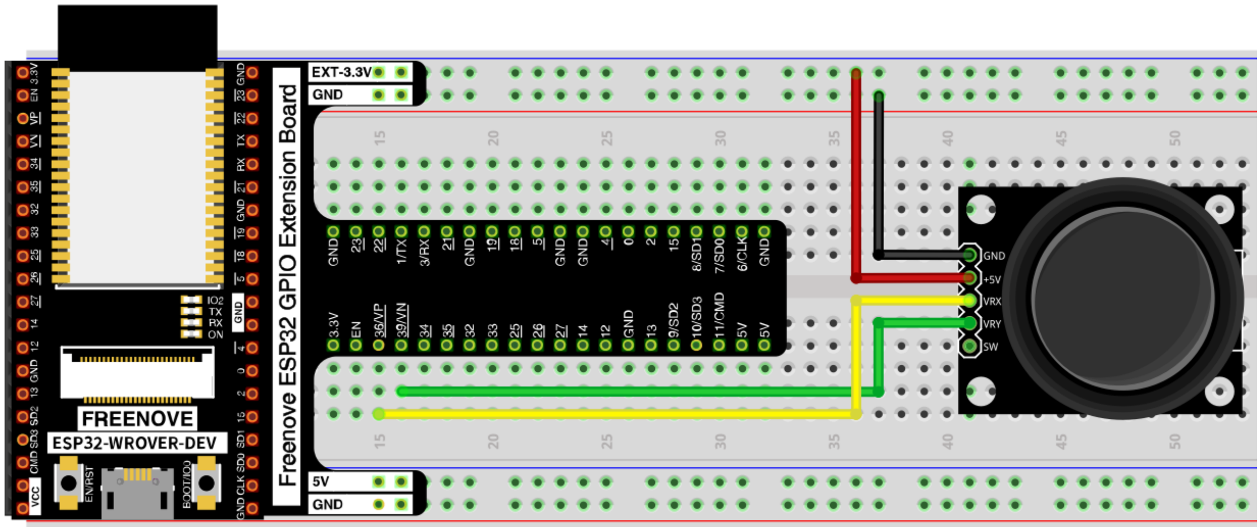
Circuit

Use Pin36 and Pin39 (A0 and A3) ports to detect the voltage value of two rotary potentiometers inside joystick.

Schematic diagram



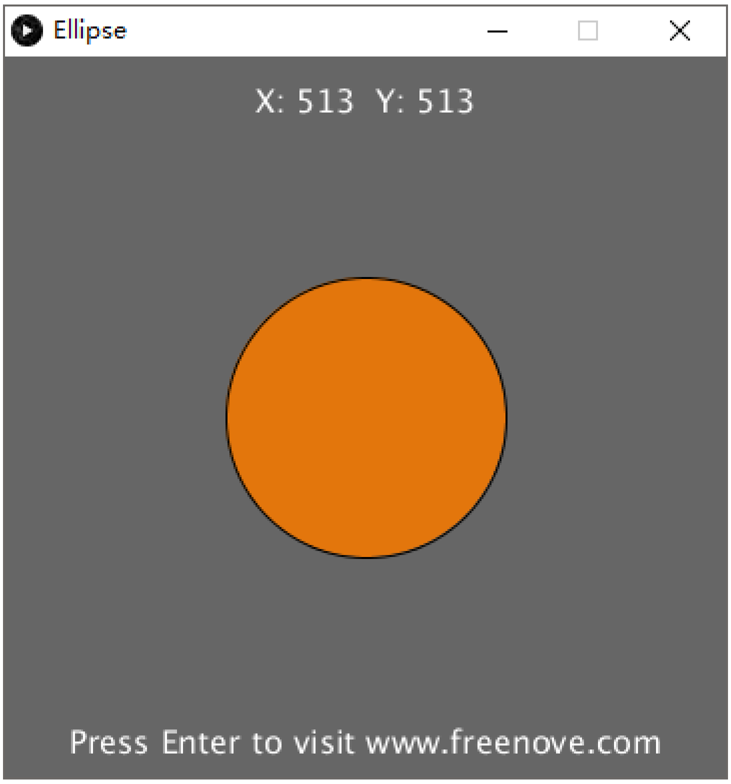
Hardware connection



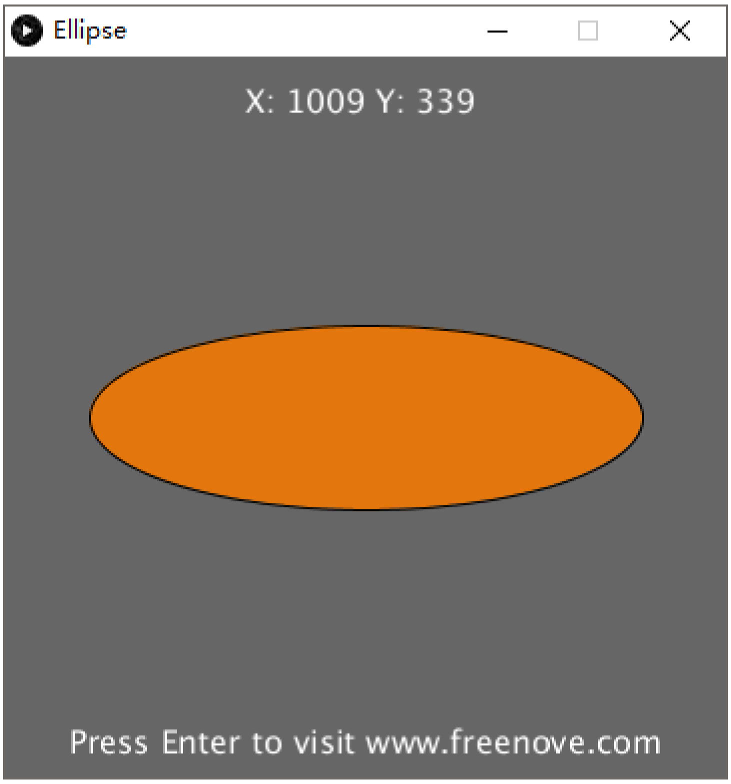
Sketch

Sketch Ellipse

Use Processing to open **Freenove Ultimate Starter Kit for ESP32\C\Processing\Processing\Ellipse\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 2.2 Box 3D

Now control a 3D figure.

Component list

The same as last section.

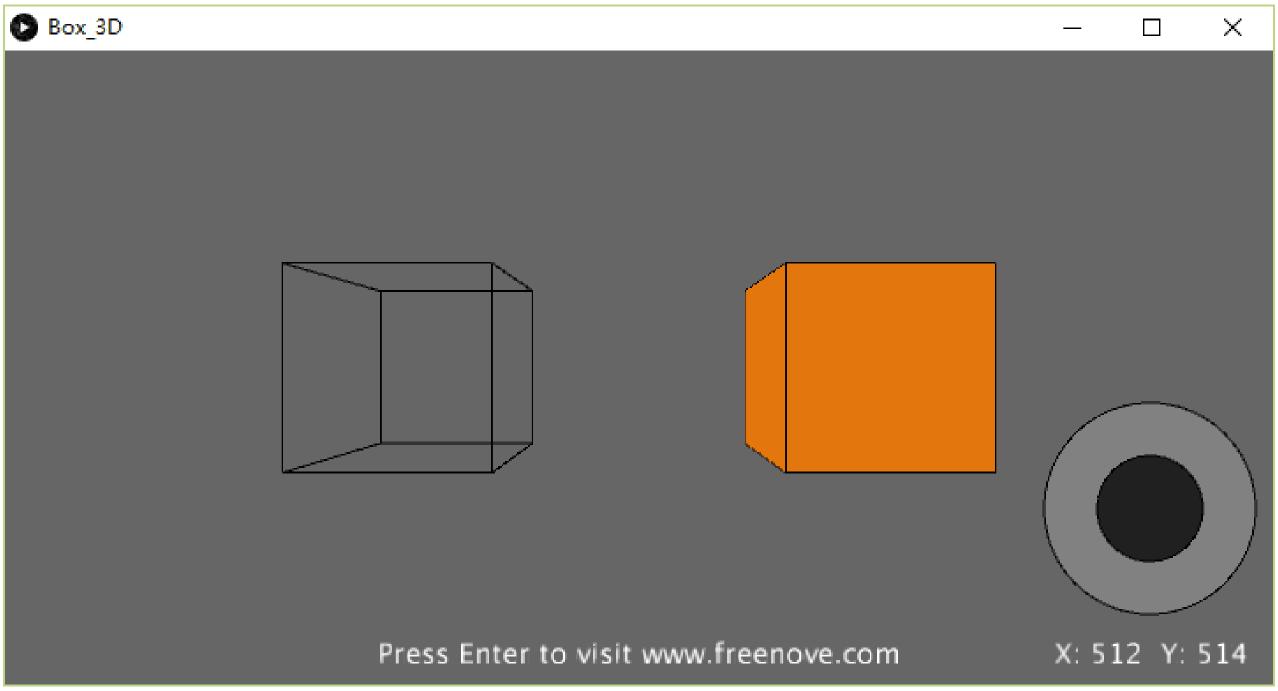
Circuit

The same as last section.

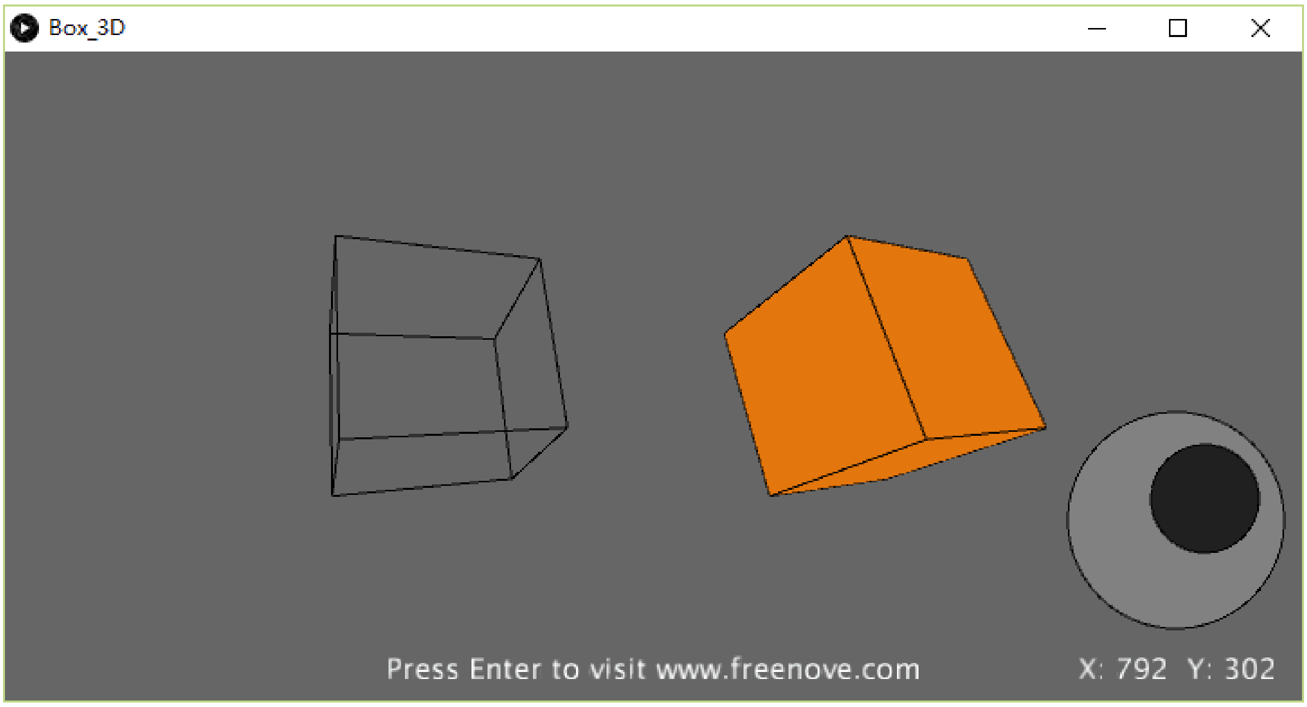
Sketch

Sketch Box_3D

Use Processing to open **Freenove Ultimate Starter Kit for ESP32\C\Processing\Processing\Box_3D\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:



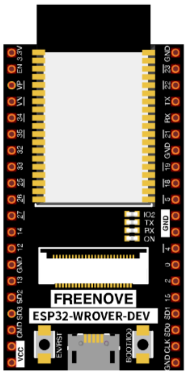
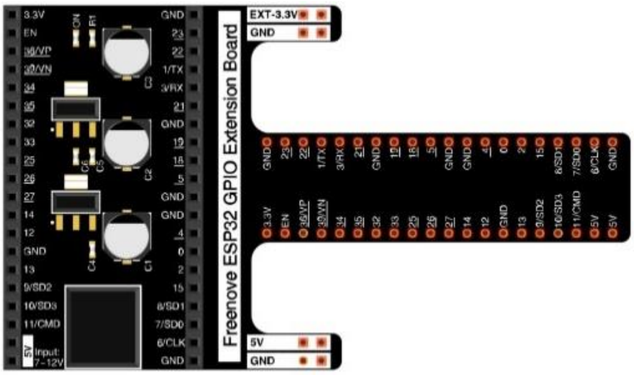
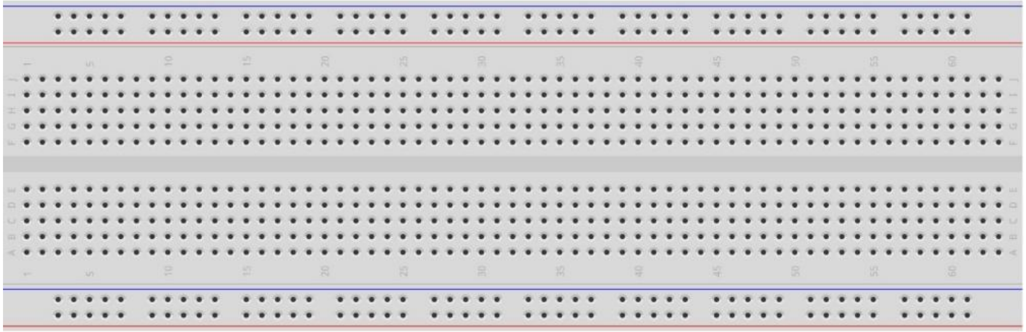
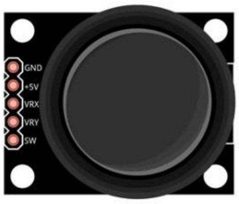
Chapter 3 Snake Game

We have learned how control 2D and 3D figures. Now, we will use ESP32 board to play the classic snake game. There are both 2D and 3D version.

Project 3.1 Snake Game

First, let’s play a 2D snake game.

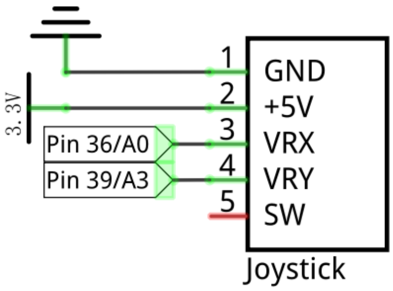
Component list

ESP32 board x1  	GPIO Extension Board x1  Freenove ESP32 GPIO Extension Board EXT-3.3V GND 5V GND
BreadBoard x1 	
Jumper M/M x5 	Joystick x1 

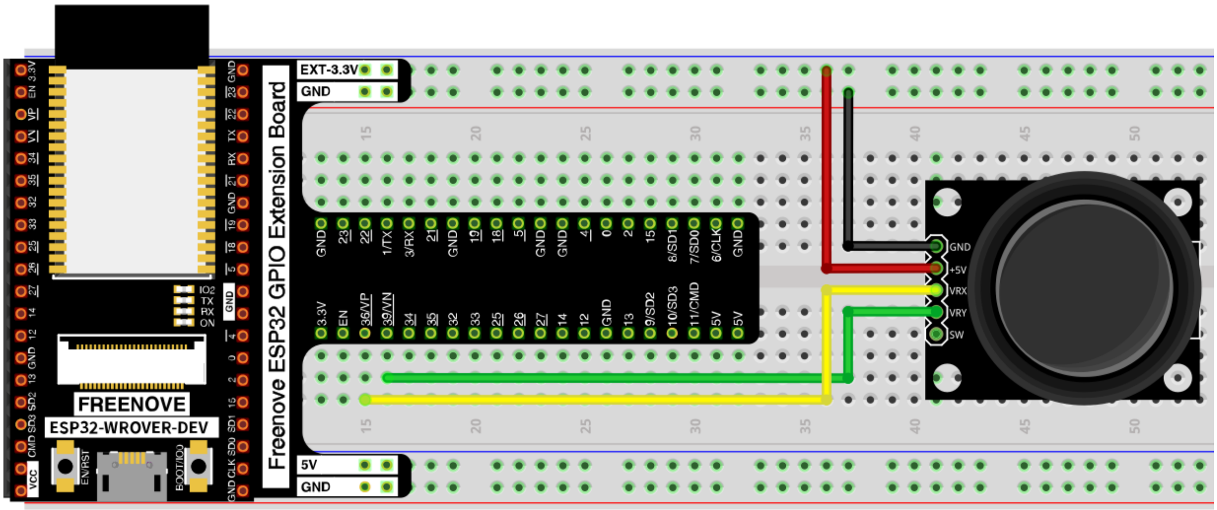
Circuit

Use Pin36 and Pin39 (A0 and A3) ports to detect the voltage value of two rotary potentiometers inside joystick.

Schematic diagram



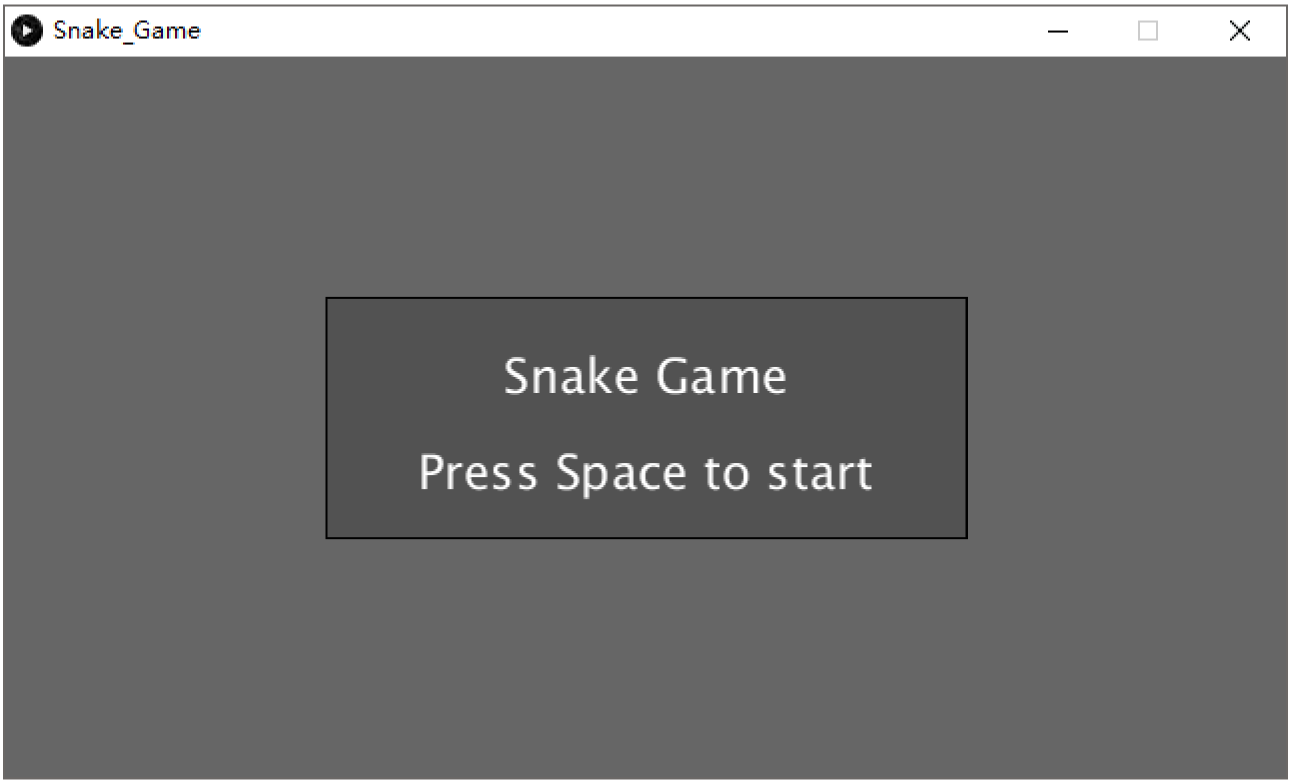
Hardware connection



Sketch

Sketch Snake_Game

Use Processing to open **Freenove Ultimate Starter Kit for ESP32\C\Processing\Processing\Box_3D\ Box_3D.pde**, and click Run.Snake_Game.pde and click Run. If the connection succeeds, the follow will be shown:

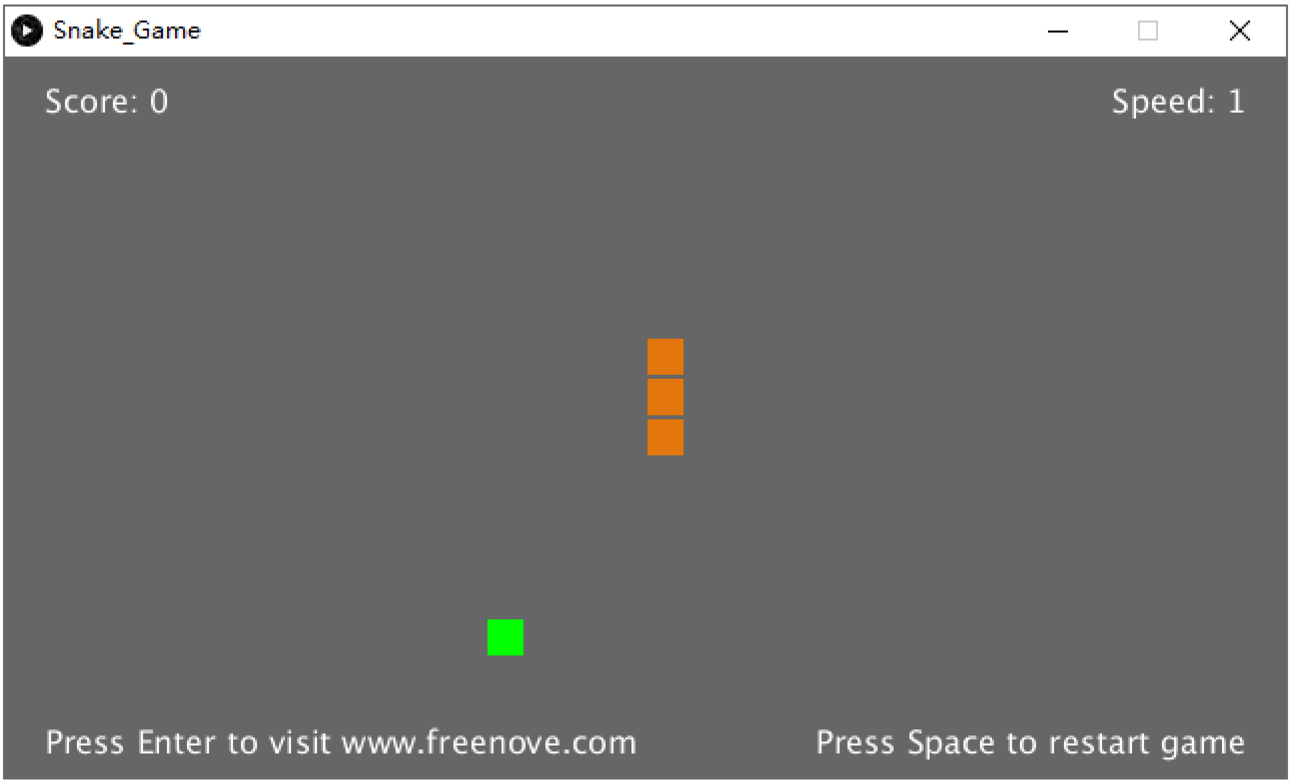


Snake_Game

Snake Game

Press Space to start

Press the space bar on keyboard to start the game:



Snake_Game

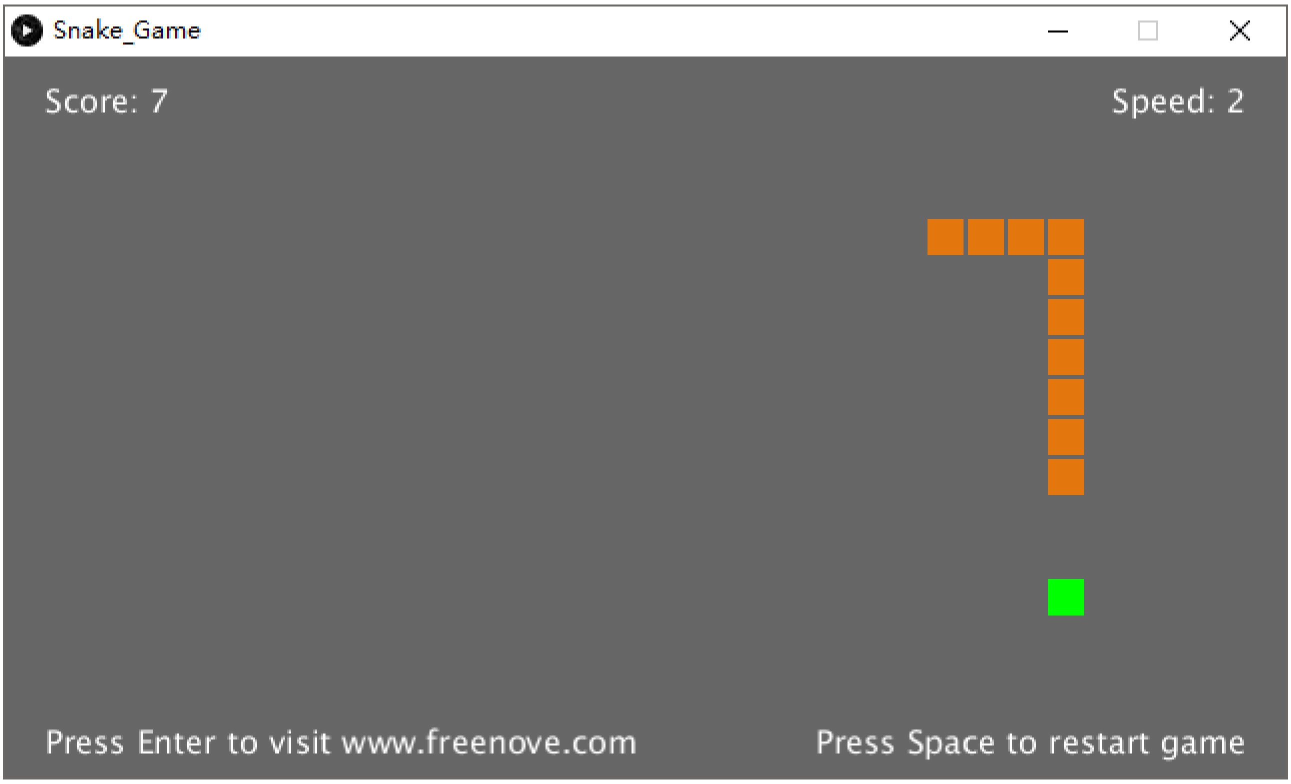
Score: 0

Speed: 1

Press Enter to visit www.freenove.com

Press Space to restart game

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



Snake_Game

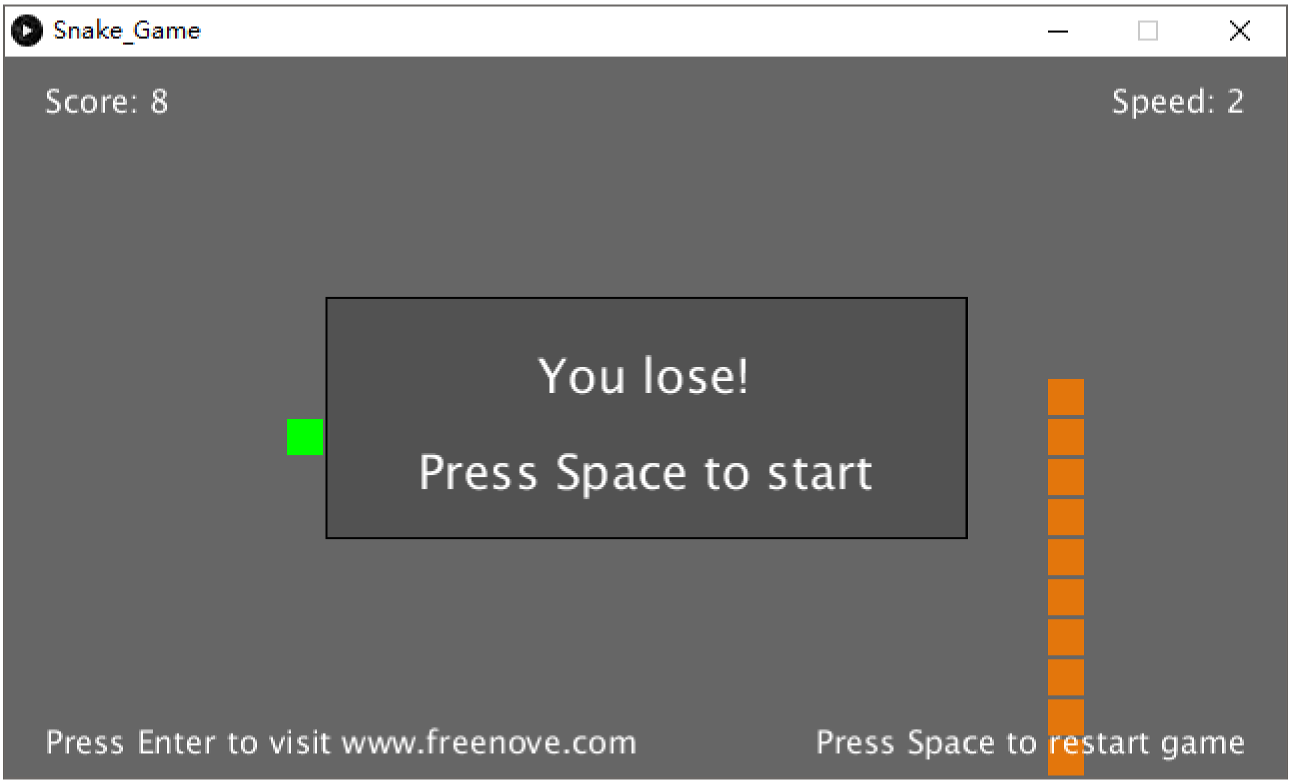
Score: 7

Speed: 2

Press Enter to visit www.freenove.com

Press Space to restart game

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


Snake_Game
Score: 8
Speed: 2
You lose!
Press Space to start
Press Enter to visit www.freenove.com
Press Space to restart game

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

Project 3.2 Snake Game 3D

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

Component list

The same as last section.

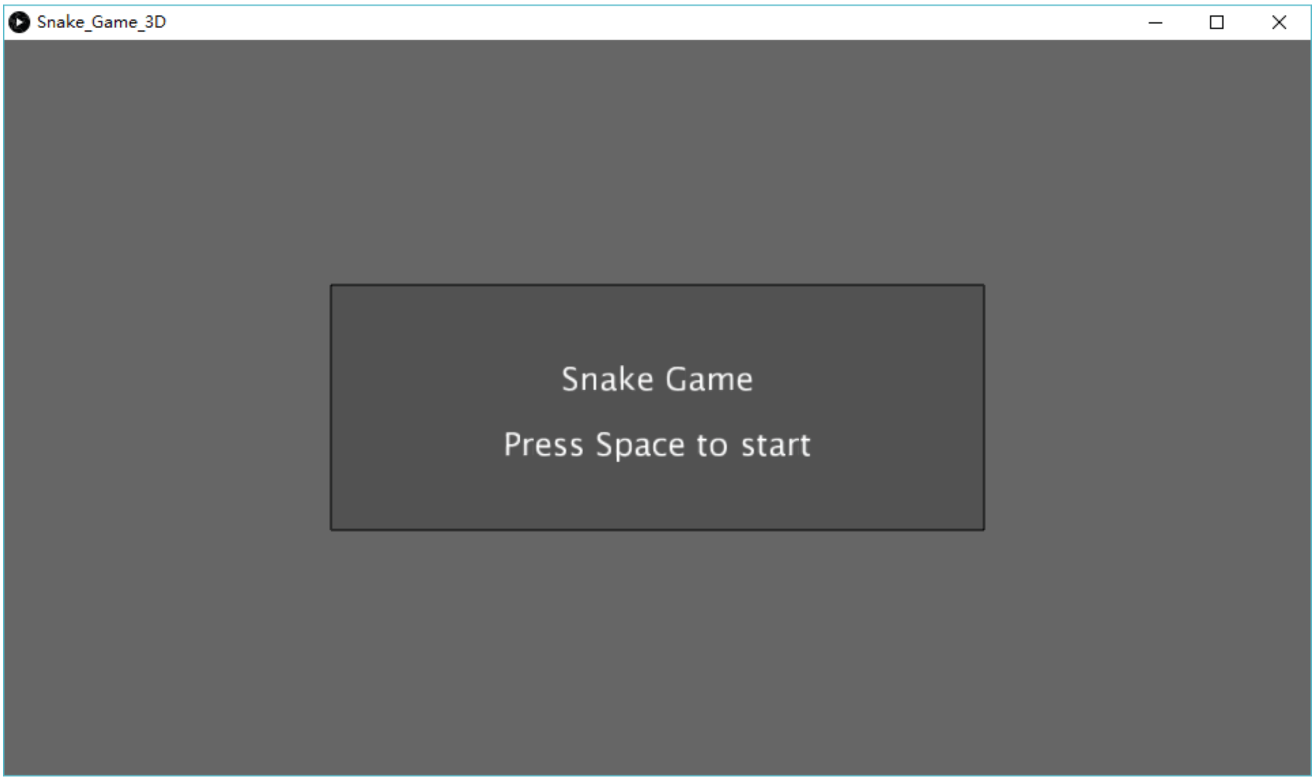
Circuit

The same as last section.

Sketch

Sketch Snake_Game_3D

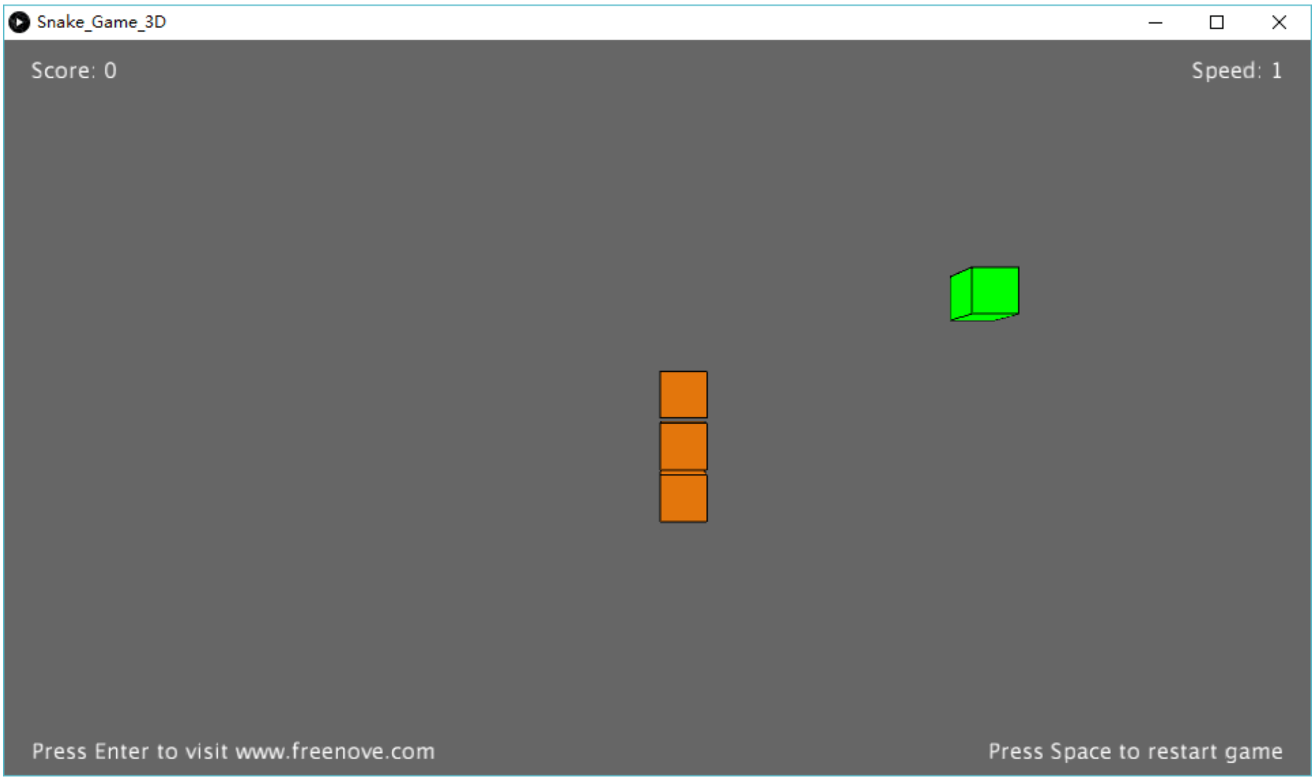
Use Processing to open **Freenove Ultimate Starter Kit for ESP32\C\Processing\Processing\Snake_Game_3D\Snake_Game_3D.pde** Snake_Game_3D.pde and click Run. If the connection succeeds, the follow will be shown:



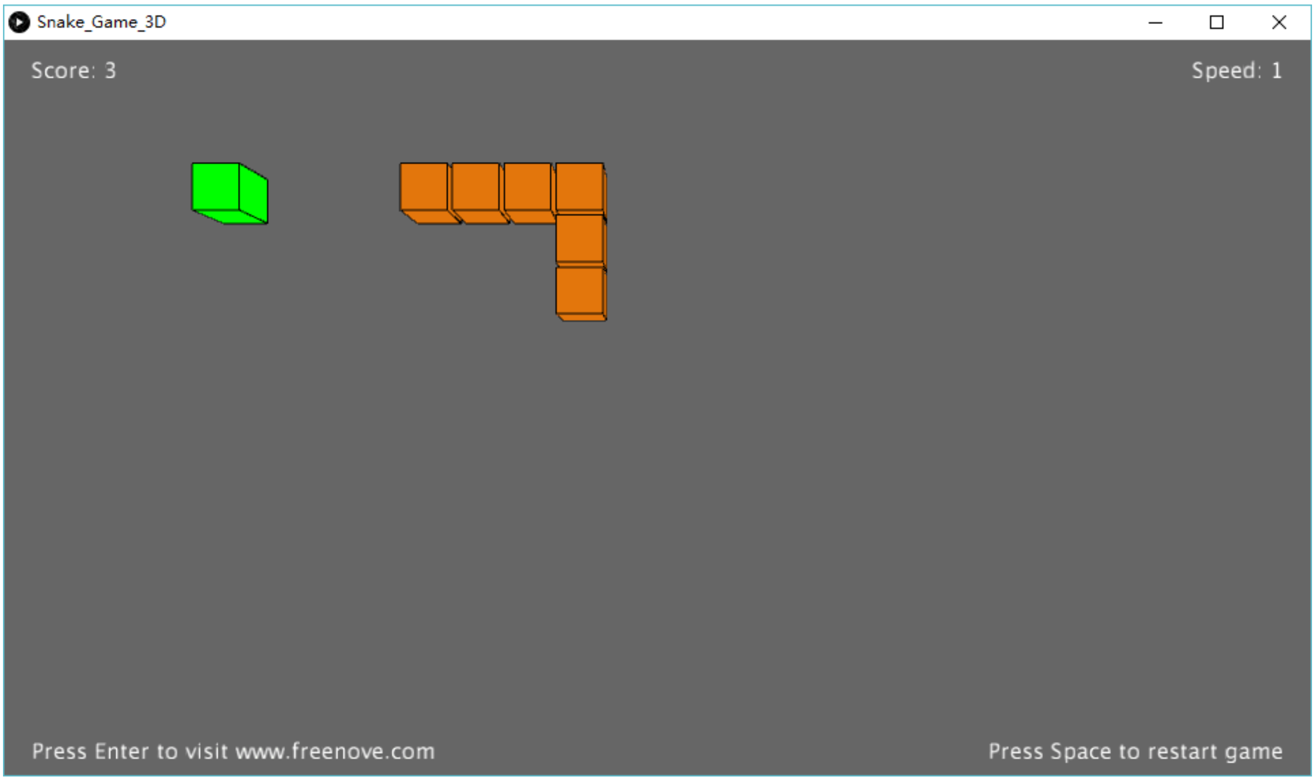
Snake Game

Press Space to start

Press the space bar on keyboard to start the game:


Snake_Game_3D
Score: 0
Speed: 1
Press Enter to visit www.freenove.com
Press Space to restart game

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


Snake_Game_3D
Score: 3
Speed: 1
Press Enter to visit www.freenove.com
Press Space to restart game

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

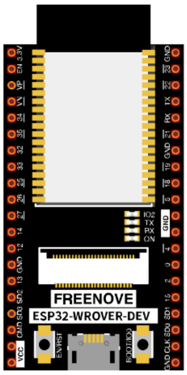
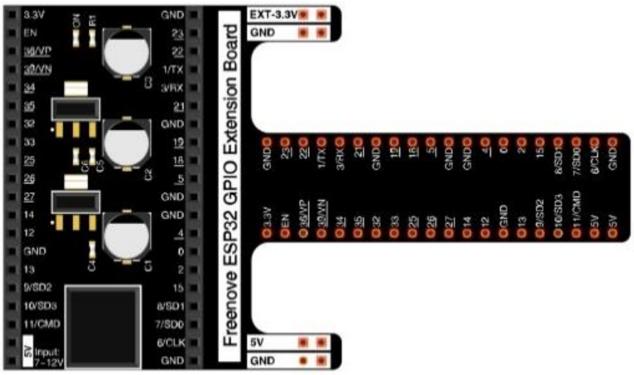
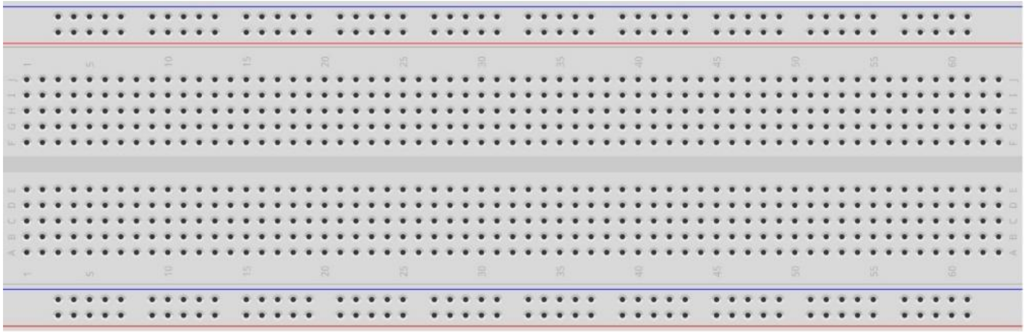
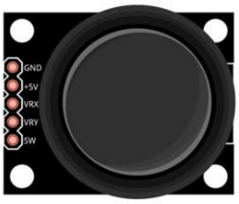
Chapter 4 Pick Apples

In this chapter, we prepare pick apples game. You can use the joystick to pick apples.

Project 4.1 Pick Apples

Now, let’s use Processing and ESP32 board to achieve the game.

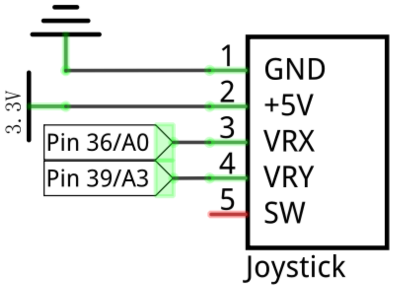
Component list

ESP32 board x1 	GPIO Extension Board x1 
BreadBoard x1 	
Jumper M/M x5 	Joystick x1 

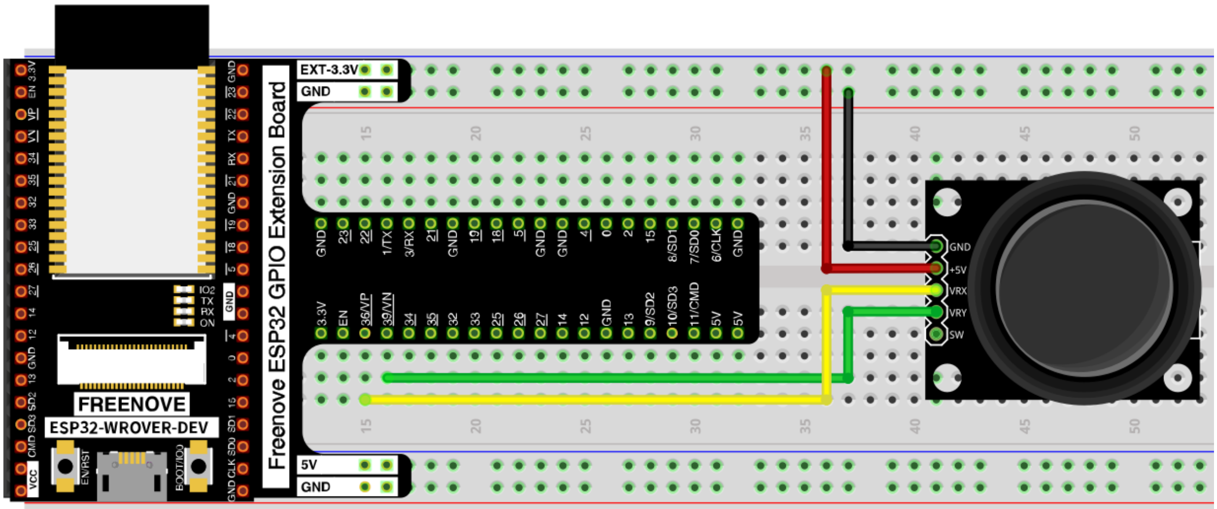
Circuit

Use Pin36 and Pin39 (A0 and A3) ports to detect the voltage value of two rotary potentiometers inside joystick.

Schematic diagram



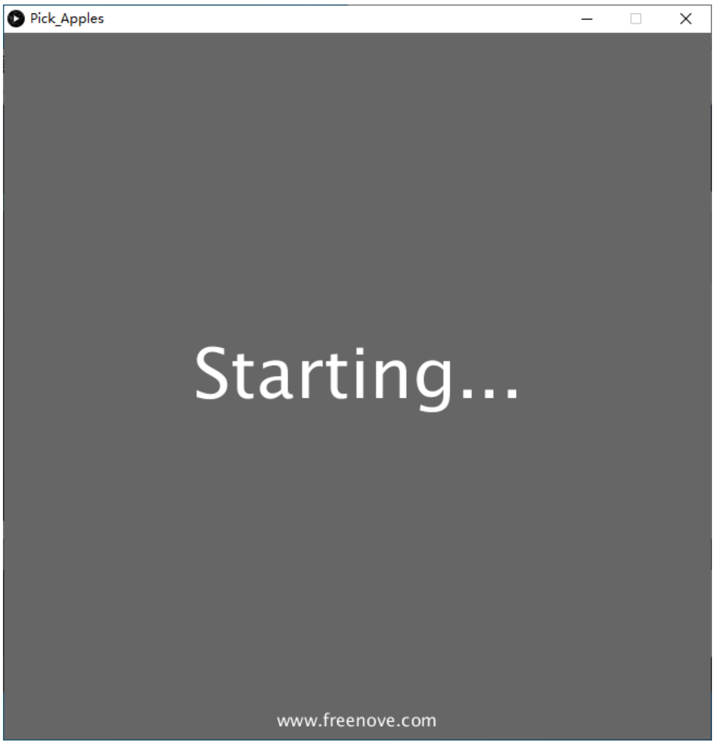
Hardware connection



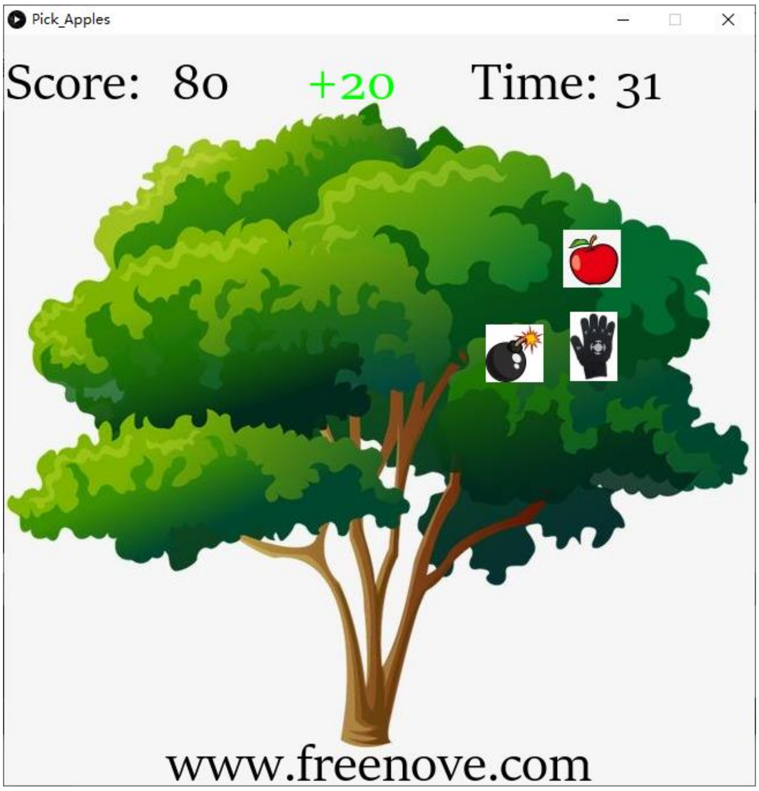
Sketch

Sketch Pick Apples

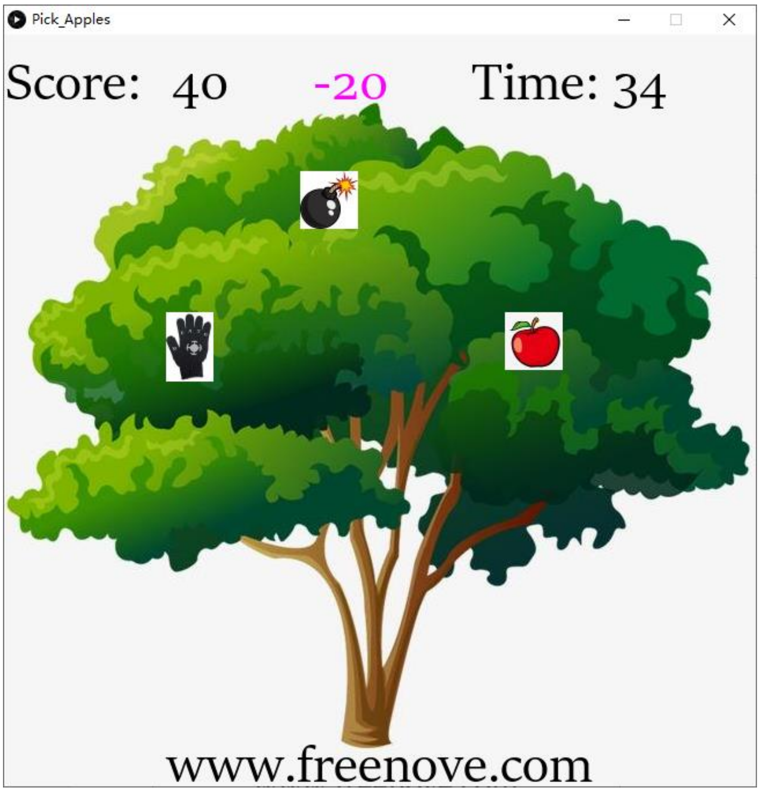
Use Processing to open Pick_Apples.pde and click Run. If the connection succeeds, the follow will be shown:



You can use the joystick to control the game. When the palm picked the apple, score +20.

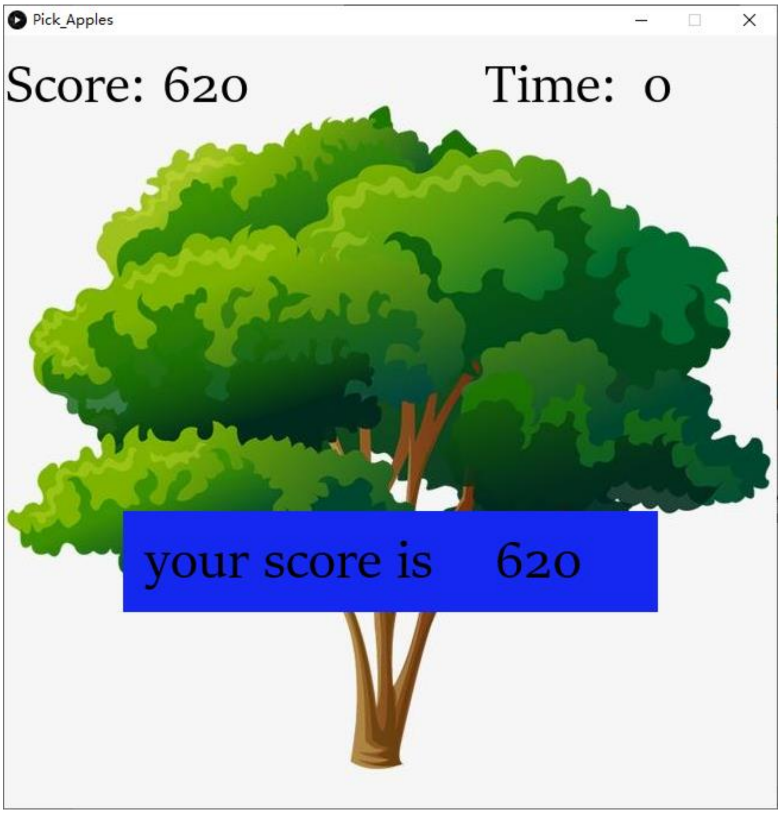


If the palm touches the bomb, the score is -20.



At the same time, you need to pick as many apples as possible before the countdown is over.

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



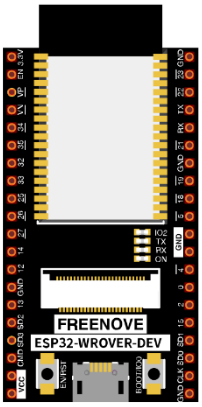
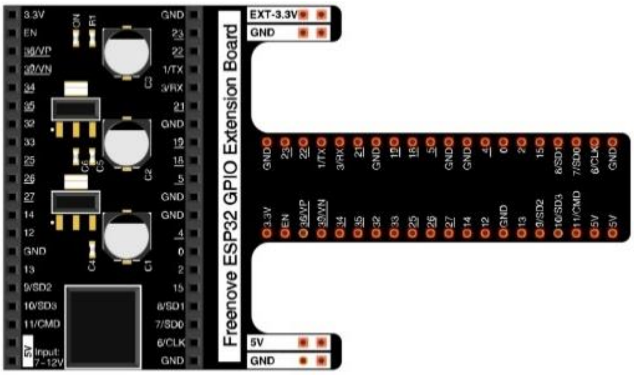
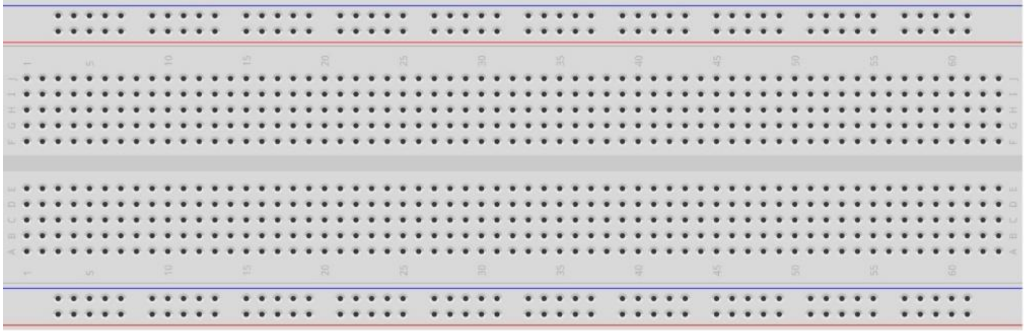
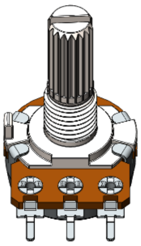
Chapter 5 Pong Game

We have experienced single-player game snake before. Now, let's use ESP32 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.

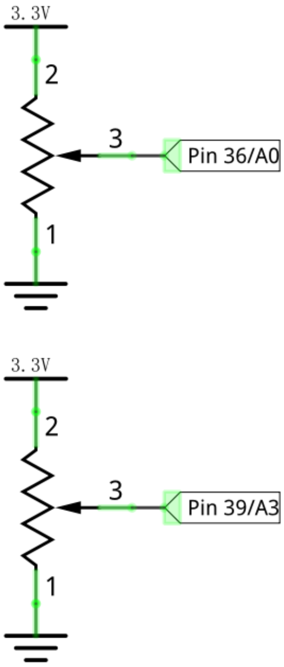
Component list

ESP32 board x1 	GPIO Extension Board x1 
BreadBoard x1 	
Jumper M/M x6 	Rotary potentiometer x2 

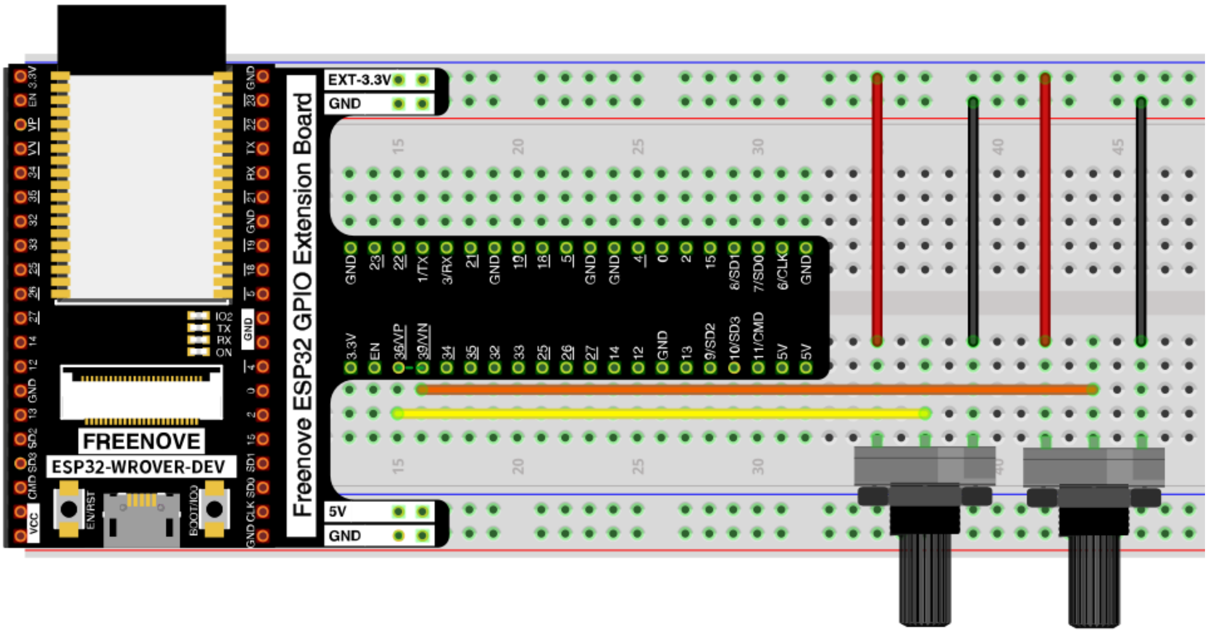
Circuit

Use A0, A3 ports on ESP32 board to detect the voltage of rotary potentiometers.

Schematic diagram



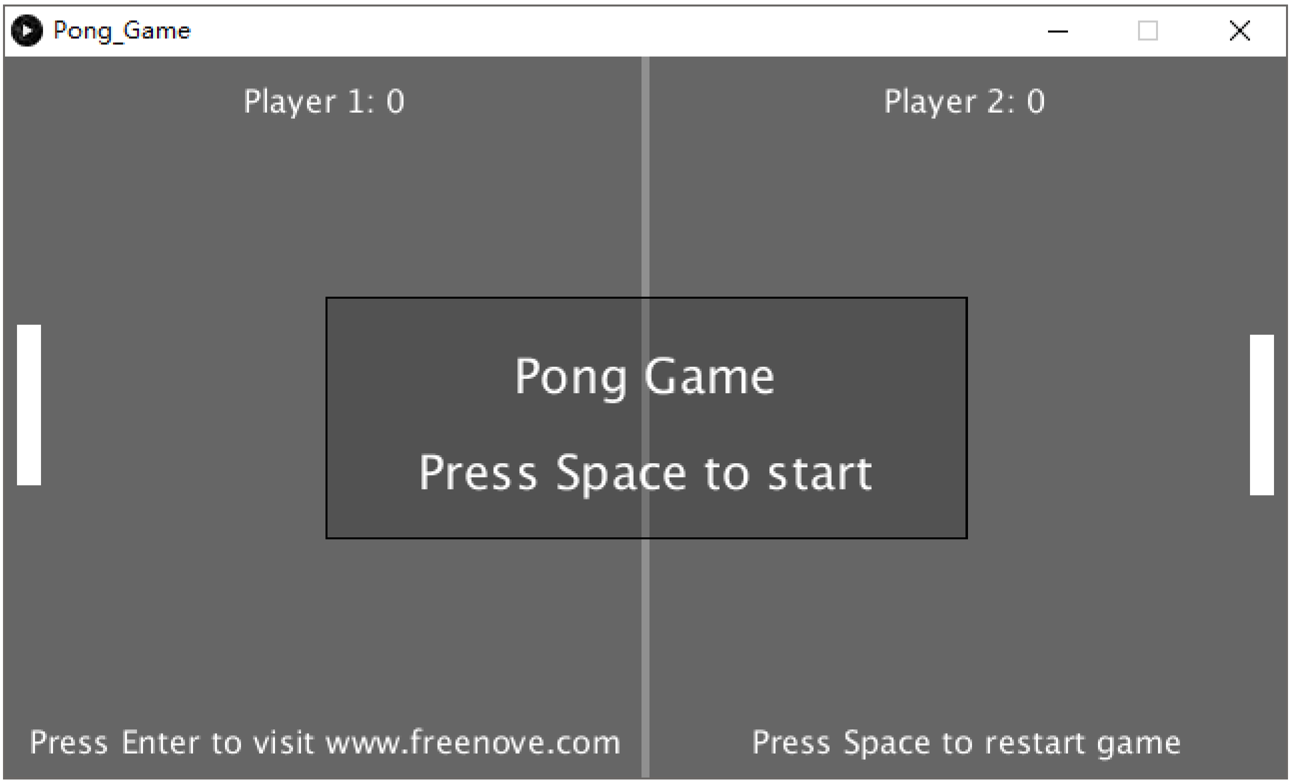
Hardware connection



Sketch

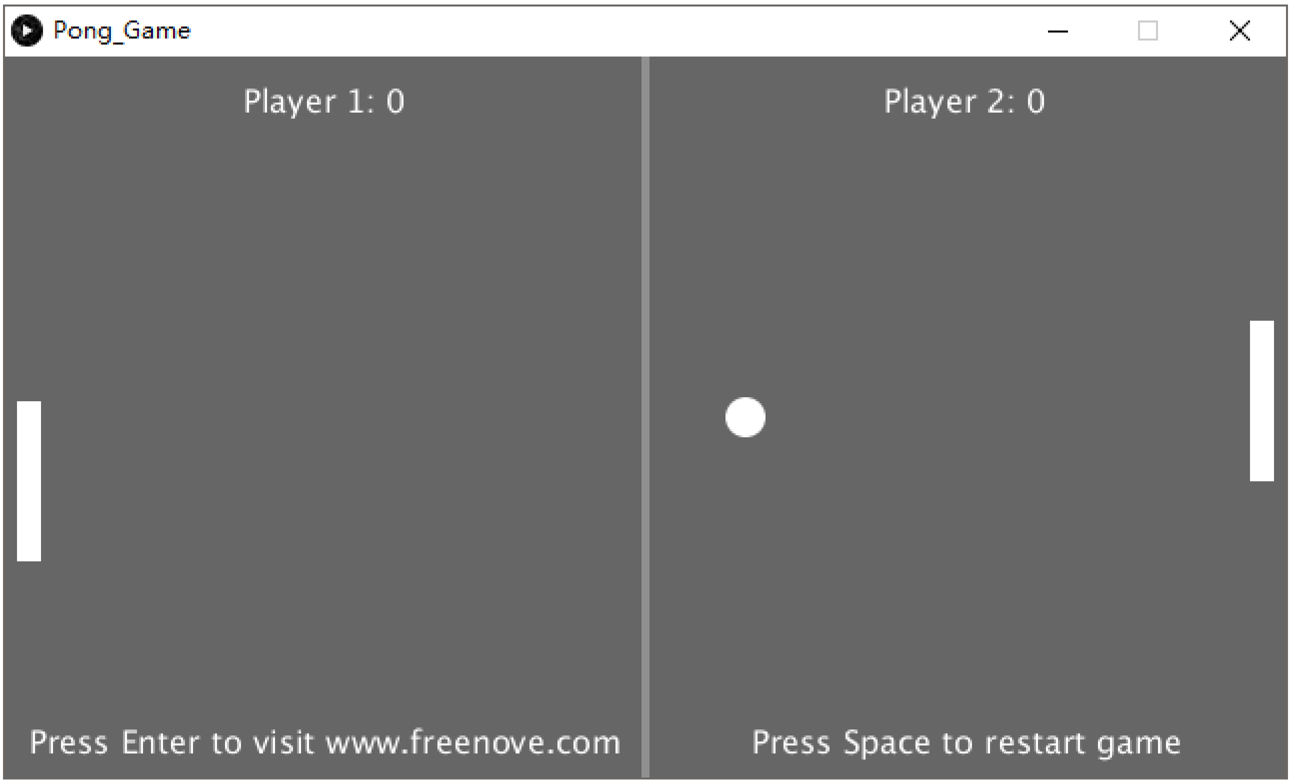
Sketch Pong_Game

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



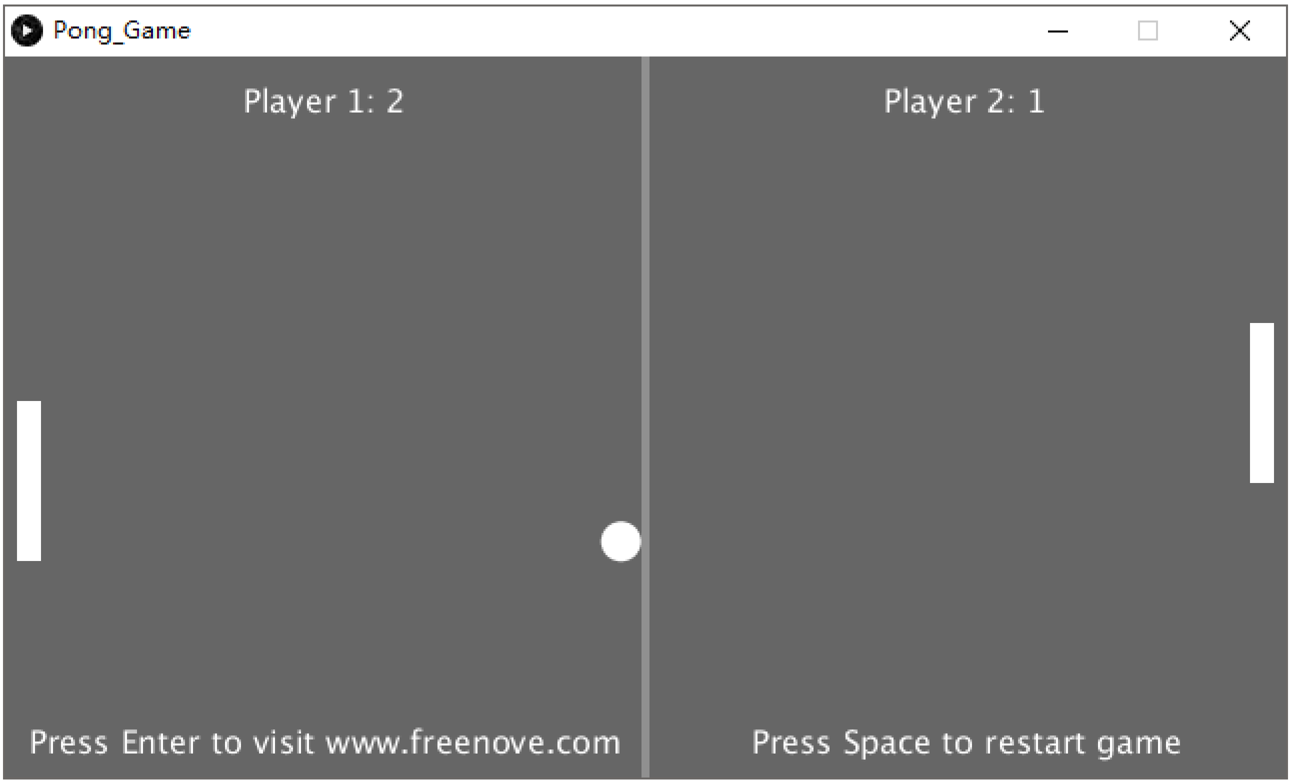
Pong_Game
Player 1: 0
Player 2: 0
Pong Game
Press Space to start
Press Enter to visit www.freenove.com
Press Space to restart game

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



Pong_Game
Player 1: 0
Player 2: 0
Press Enter to visit www.freenove.com
Press Space to restart game

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



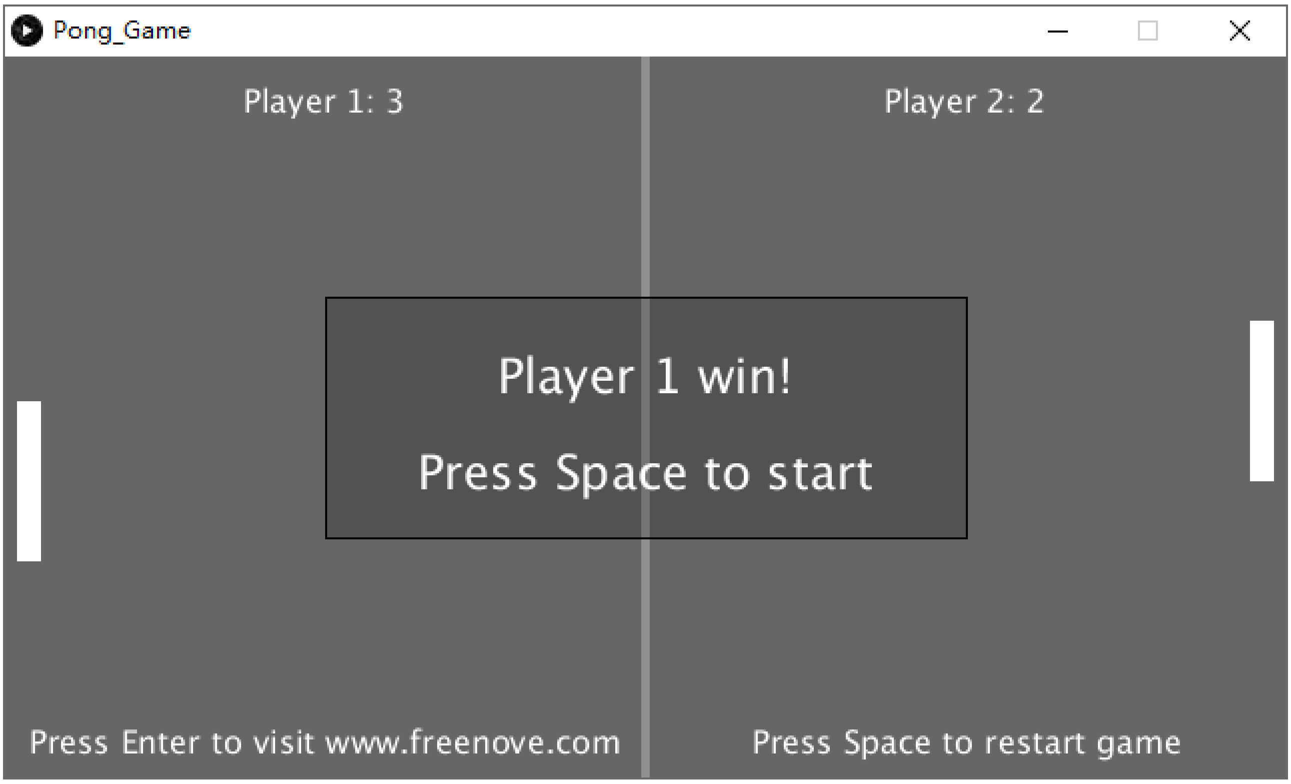
Player 1: 2

Player 2: 1

Press Enter to visit www.freenove.com

Press Space to restart game

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



Player 1: 3

Player 2: 2

Player 1 win!

Press Space to start

Press Enter to visit www.freenove.com

Press Space to restart 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.

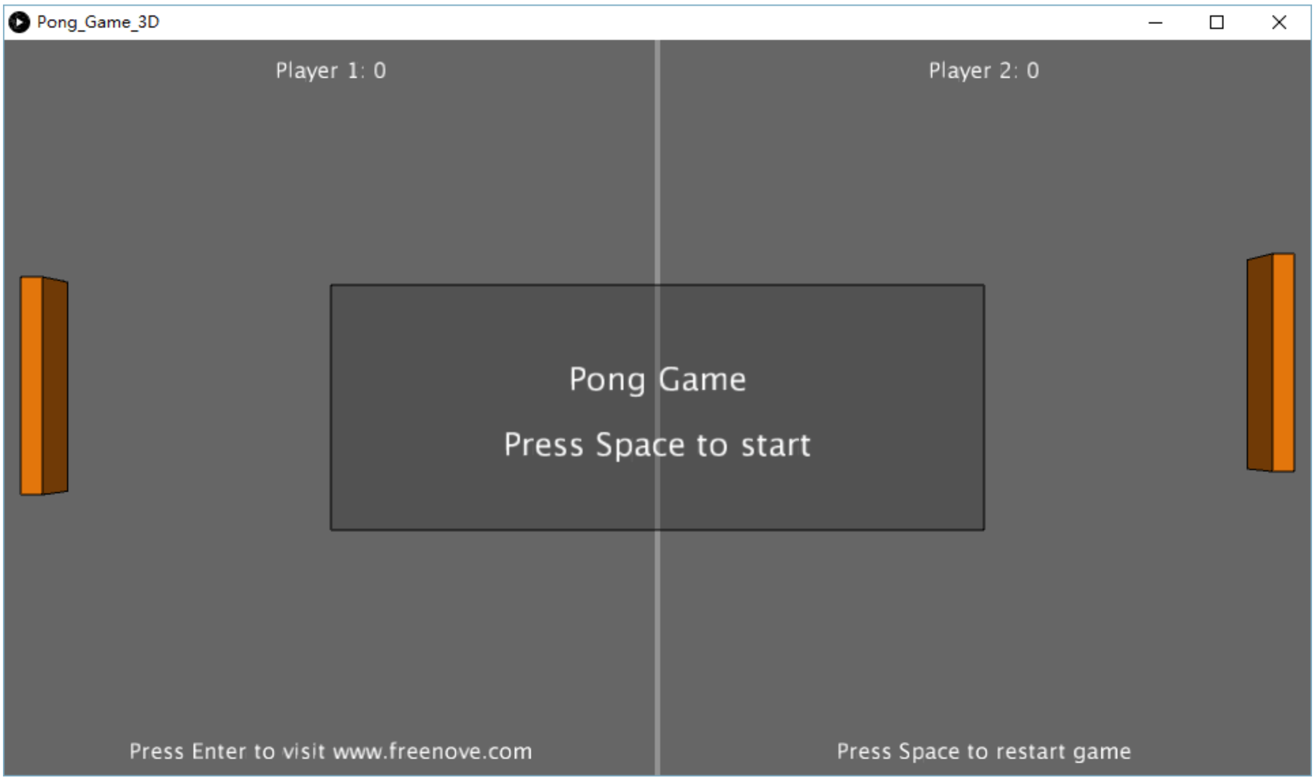
Circuit

The same as last section.

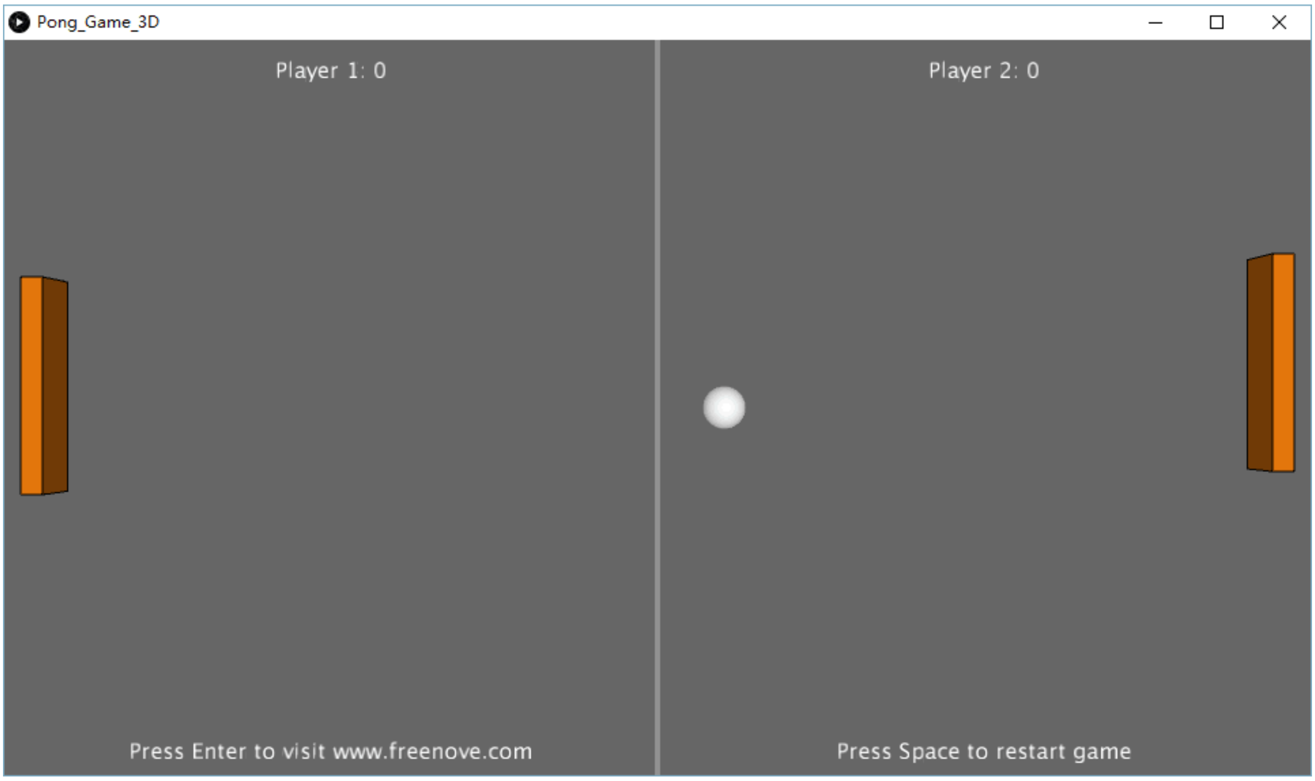
Sketch

Sketch Pong_Game_3D

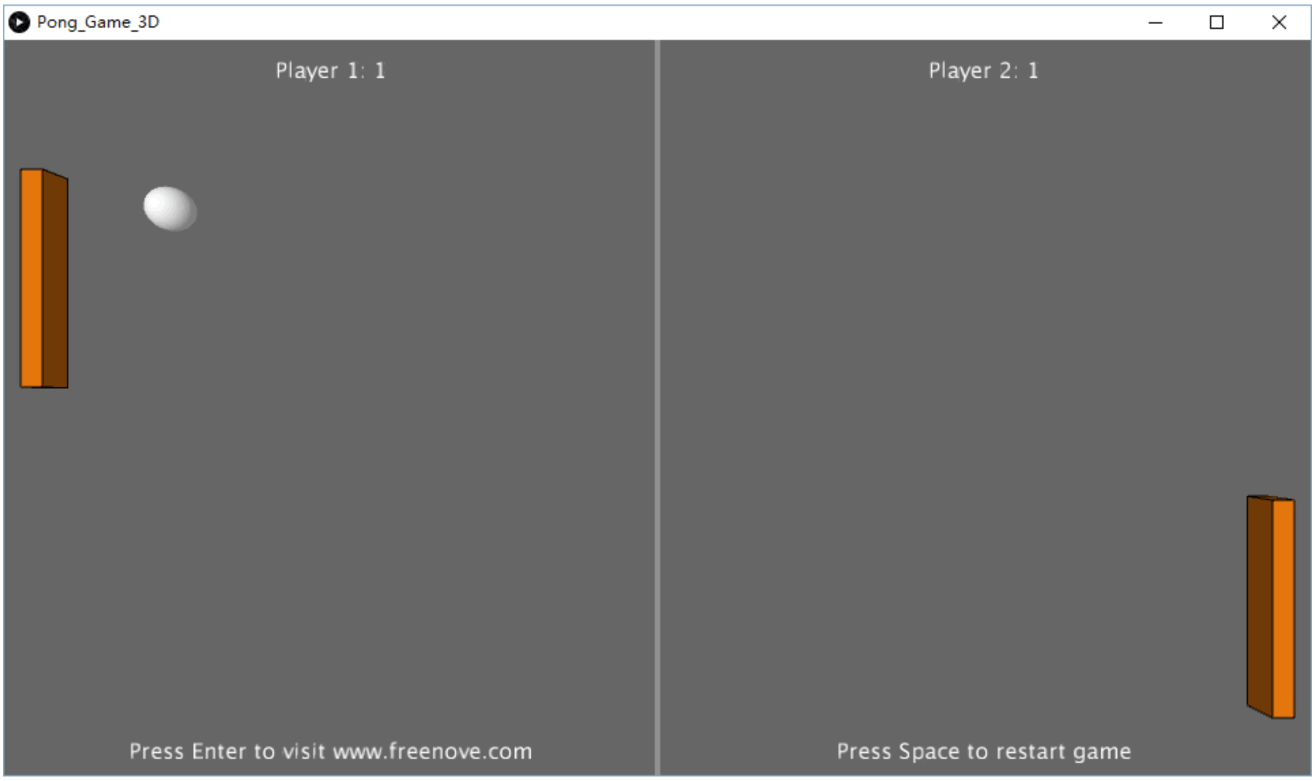
Use Processing to open Pong_Game_3D.pde and click Run. If the connection succeeds, the follow will be shown:


Pong_Game_3D
Player 1: 0
Player 2: 0
Pong Game
Press Space to start
Press Enter to visit www.freenove.com
Press Space to restart game

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


Pong_Game_3D
Player 1: 0
Player 2: 0
Press Enter to visit www.freenove.com
Press Space to restart game

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


Pong_Game_3D
Player 1: 1
Player 2: 1
Press Enter to visit www.freenove.com
Press Space to restart game

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

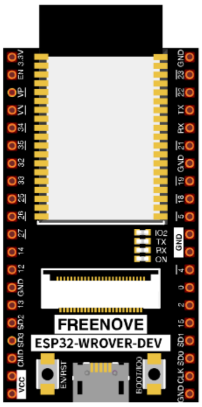
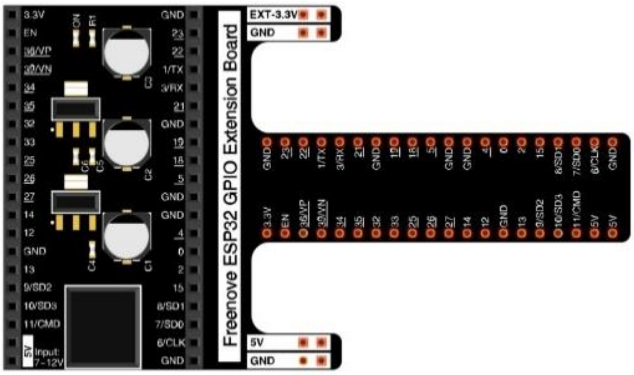
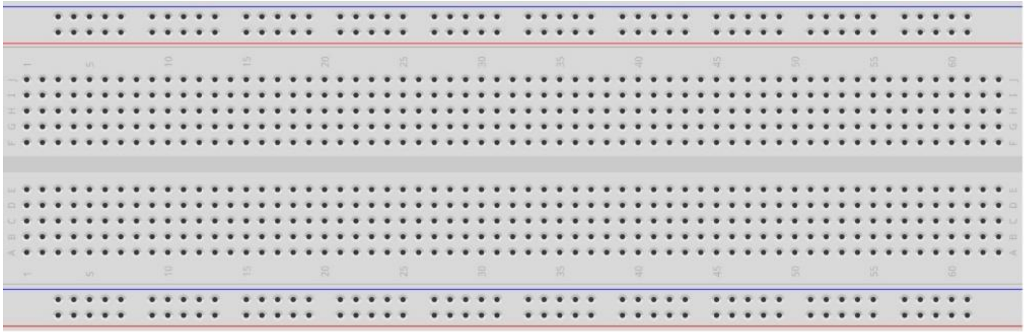
Chapter 6 Tetris

In this chapter, we prepare a tetris game for you. You can play the game by pressing the buttons.

Project 6.1 Tetris

Now, let’s use Processing and ESP32 board to achieve the tetris game.

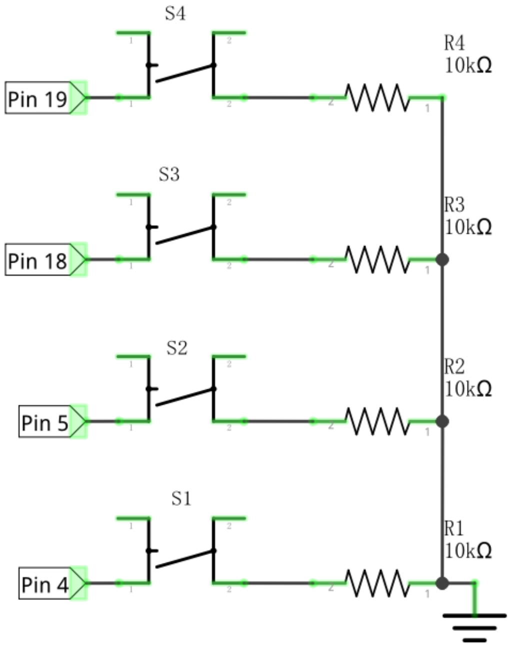
Component list

<table>
<tr><td>ESP32 board x1

 </td><td colspan="2">GPIO Extension Board x1

 </td></tr>
<tr><td colspan="3">BreadBoard x1
</td></tr>
<tr><td>Jumper M/M x4
 </td><td>Push button x4
 </td><td>Resistor 10kΩ x4
 </td></tr>
</table>

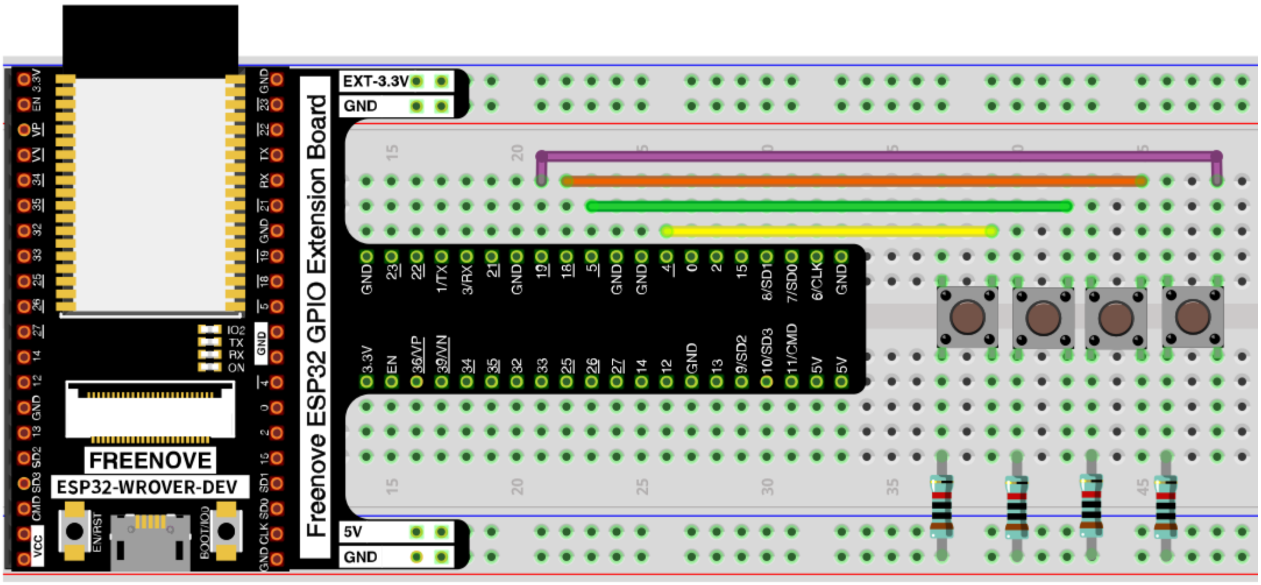
Circuit

Use Pin4, Pin5, Pin18, Pin19 port on ESP32 board to control the movement of the block.

Schematic diagram



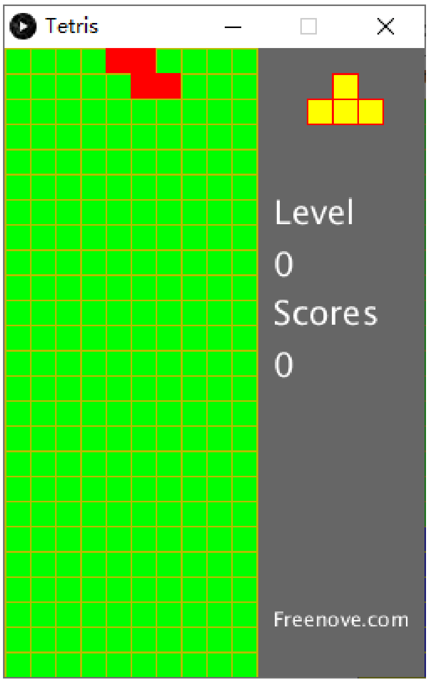
Hardware connection



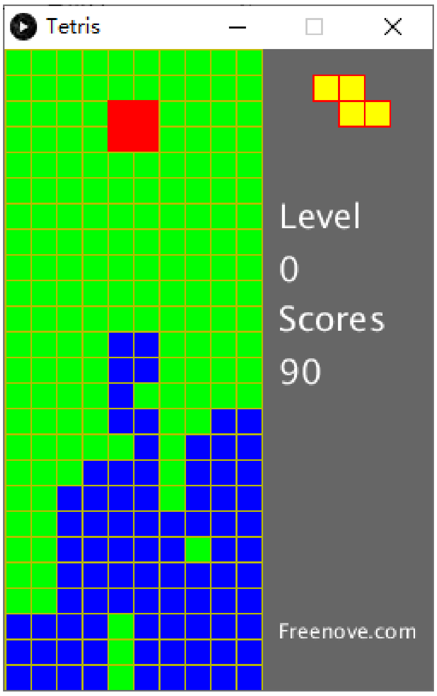
Sketch

Sketch Tetris

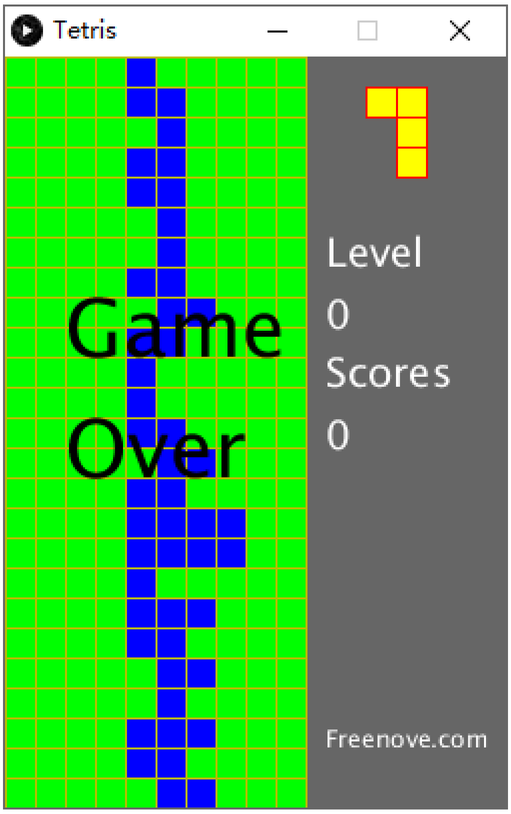
Use Processing to open Tetris.pde and click Run. If the connection succeeds, the follow will be shown:



Now you can try using the buttons to control the falling block. Press the space bar to pause the game.



When the game is over, you can press the space bar to play the game again. Press ESC to exit the game.



What's next?

Thanks for your reading! This document is all over here.
If you find any mistakes, please feel free to contact us at support@freenove.com. 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.
www.freenove.com

Thank you again for choosing Freenove products.