

Lesson 3 Getting Started with Raspberry Pi

In this lesson, we will learn how to install and remotely log in to the Raspberry Pi OS under Windows. We will download the code program to control the robot.

3.1 Preparation

- When studying this lesson, you need to prepare the following components first:
One SD card that has been formatted (we recommend using an SD card with memory above 16G), one card reader, a Raspberry Pi development board.
- You need to insert the SD card into the card reader first, and then connect the card reader to the computer.



3.2 Downloading the Raspberry Pi OS Raspbian

Raspbian is the official OS of the Raspberry Pi Foundation. It is customized based on Debian GNU/Linux and can run on all versions of the Raspberry Pi motherboard. According to experience, Raspbian works best with Raspberry Pi. It is stable, powerful, and easy to use, and can basically meet the needs of various applications. This course uses Raspbian as the preferred OS for the Raspberry Pi. Next, we will teach you how to download the Raspberry Pi OS Raspbian.

This OS is recommended for Raspberry Pi 5, Raspberry Pi 3B, 3B+, and 4. This system can also be used.

1. visit the official website of the Raspberry Pi through a browser to download Raspbian: <https://www.raspberrypi.com/software/operating-systems/>

After entering the Raspberry Pi official website page, click "[Raspberry Pi OS](#)" to jump to the corresponding location, or you can find the location of Raspberry Pi OS by sliding the page.

Operating system images

Many operating systems are available for Raspberry Pi, including Raspberry Pi OS, our official supported operating system, and operating systems from other organisations.

[Raspberry Pi Imager](#) is the quick and easy way to install an operating system to a microSD card ready to use with your Raspberry Pi. Alternatively, choose from the operating systems below, available to download and install manually.

Download:

[Raspberry Pi OS](#)

[Raspberry Pi OS \(64-bit\)](#)

[Raspberry Pi OS \(Legacy\)](#)

[Raspberry Pi OS \(Legacy, 64-bit\)](#)

[Raspberry Pi Desktop](#)

2. Select the Raspberry Pi OS with desktop version, which contains a complete desktop system and recommended software packages.

Raspberry Pi OS with desktop

Release date: March 15th 2024

System: 32-bit

Kernel version: 6.6

Debian version: 12 (bookworm)

Size: 1,158MB

[Show SHA256 file integrity hash:](#)

[Release notes](#)

Download

[Download torrent](#)

[Archive](#)

Raspberry Pi OS with desktop and recommended software

Release date: March 15th 2024

System: 32-bit

Kernel version: 6.6

Debian version: 12 (bookworm)

Size: 2,678MB

[Show SHA256 file integrity hash:](#)

[Release notes](#)

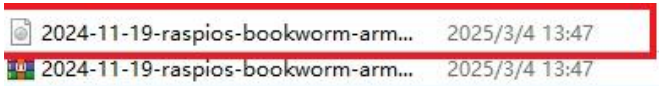
Download

[Download torrent](#)

[Archive](#)

3. Click "**Download**", to get the ".img.xz" file(or compressed files in other formats), and wait for the download to complete. (remember where the file is saved).

4. Find the ".img.xz" file you just downloaded, and extract it. The uncompressed file format is ".img". Pay attention that the path of the uncompressed ".img" file should be named with only English letters and contain no special characters.



3.3 Burning the downloaded Raspberry Pi OS to the SD card

We recommend using the Raspberry Pi Imager tool officially provided by the Raspberry Pi. Raspberry Pi Imager is a new image burning tool launched by the Raspberry Pi Foundation. Users can download and run this tool on Windows, Mac OS and Ubuntu to burn the OS image for the Raspberry Pi. Its usage is similar to Etcher and win32diskimager.

3.3.1 Downloading Raspberry Pi Imager

1. Visit the official website of Raspberry Pi to download through a browser:

<https://www.raspberrypi.org/downloads/>

- For Windows OS, click "**Download for windows**" to download.
- For Mac OS, click "**Download for macOS**" to download.
- For Linux OS, Click "**Download for Ubuntu**" to download.

Install Raspberry Pi OS using Raspberry Pi Imager

Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi.

Download and install Raspberry Pi Imager to a computer with an SD card reader. Put the SD card you'll use with your Raspberry Pi into the reader and run Raspberry Pi Imager.

[Download for Windows](#)

[Download for macOS](#)

[Download for Ubuntu for x86](#)

To install on **Raspberry Pi OS**, type
`sudo apt install rpi-imager`
in a Terminal window.

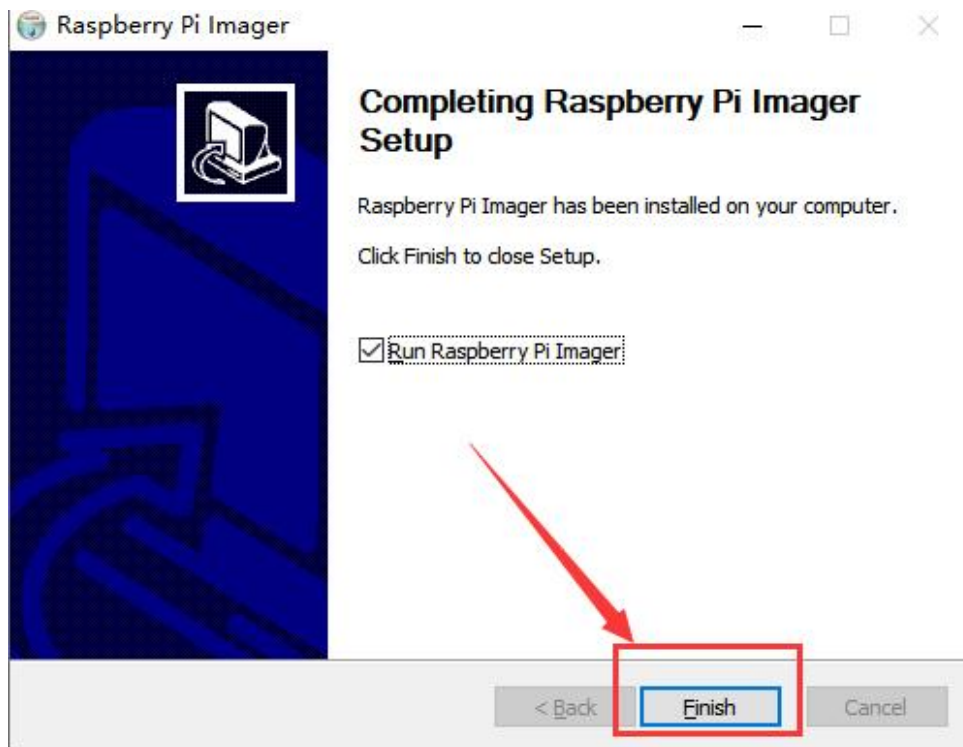


After the download is complete, install the software and burn the Raspberry Pi OS .

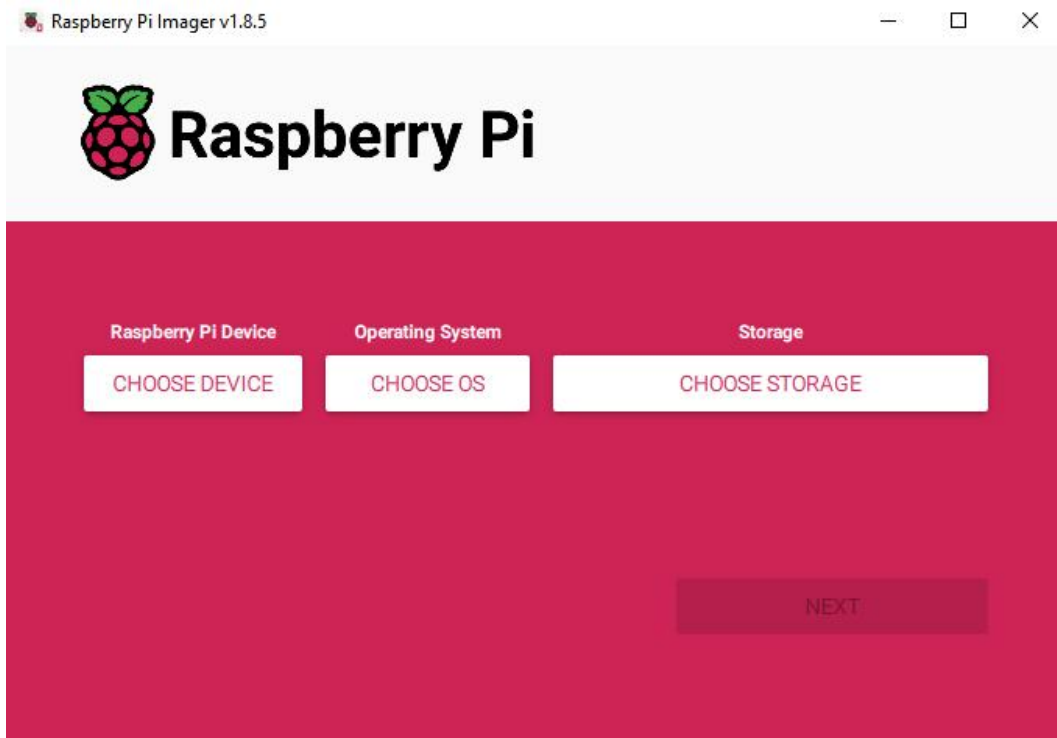
2. Open the downloaded file "**imager.exe**" and click "**Install**".



3. Then click "**Finish**".



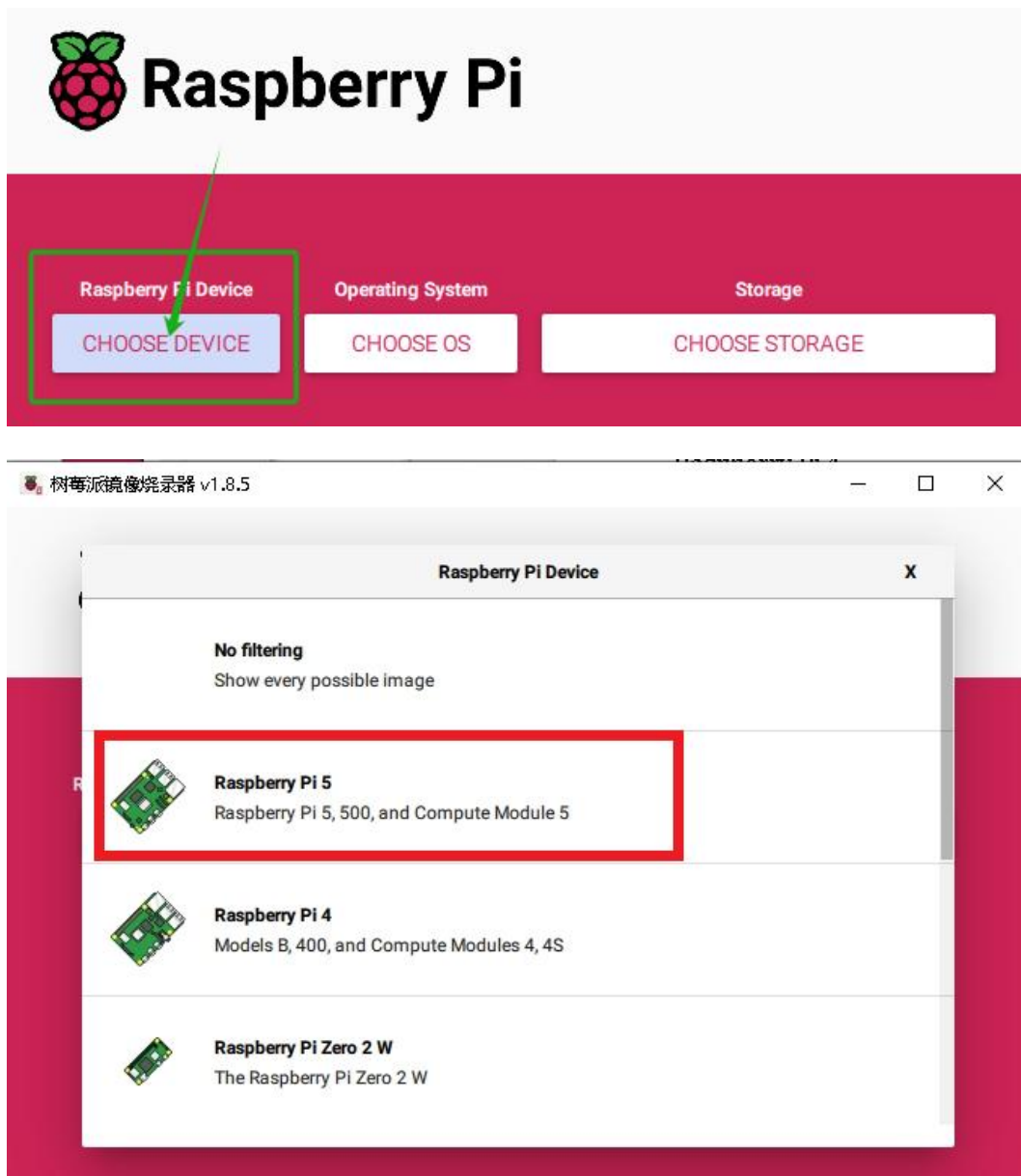
4.The software interface after opening is as shown below:



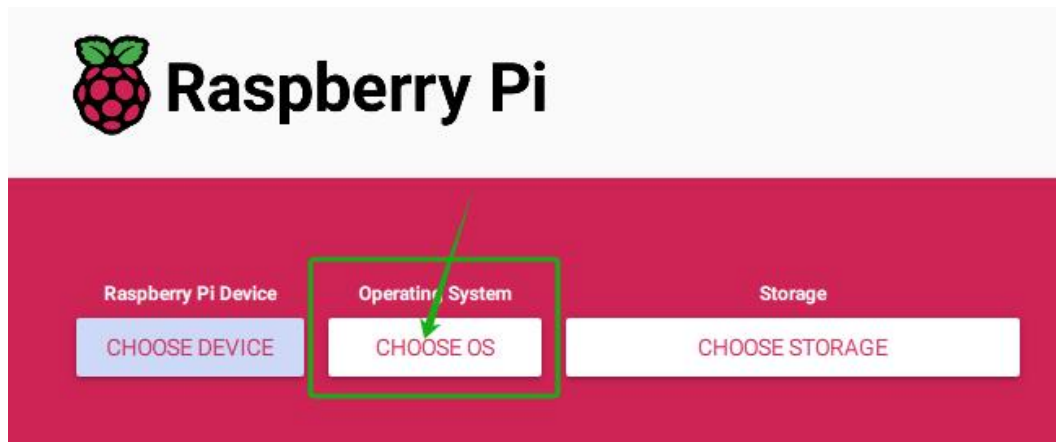
3.3.2 Burning Raspberry Pi OS to SD card with Raspberry Pi Imager

Install the operating system on RPi for programming and running programs.

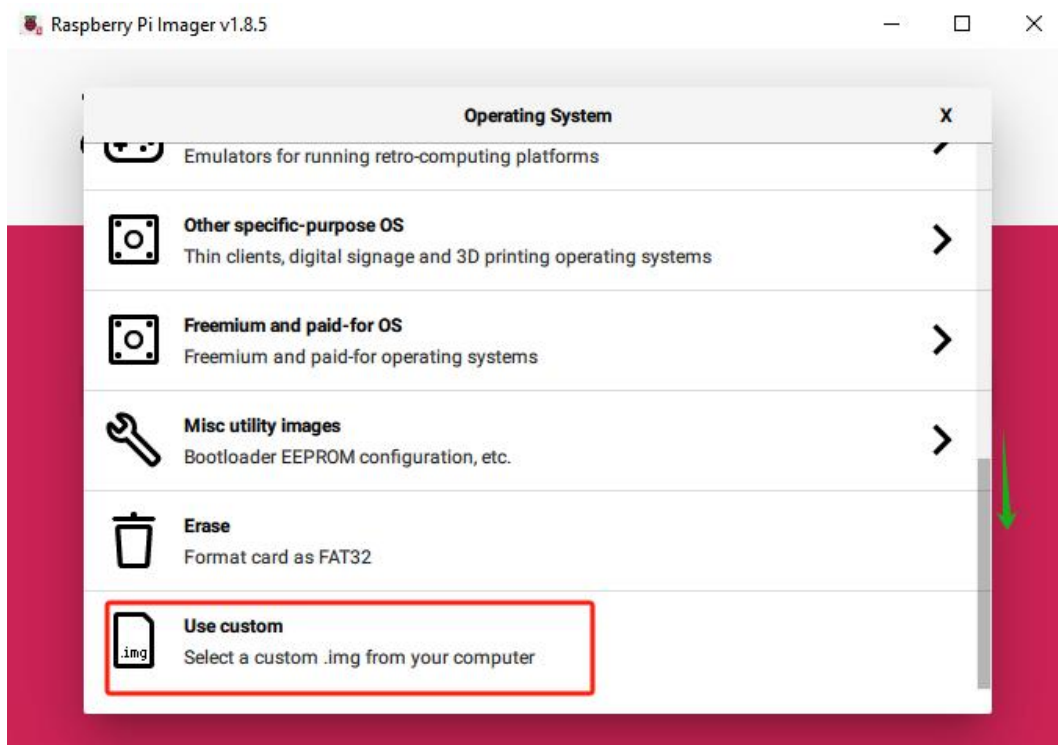
1. Click "**CHOOSE DEVICE**" to select the corresponding Raspberry Pi model, such as Raspberry Pi 5.



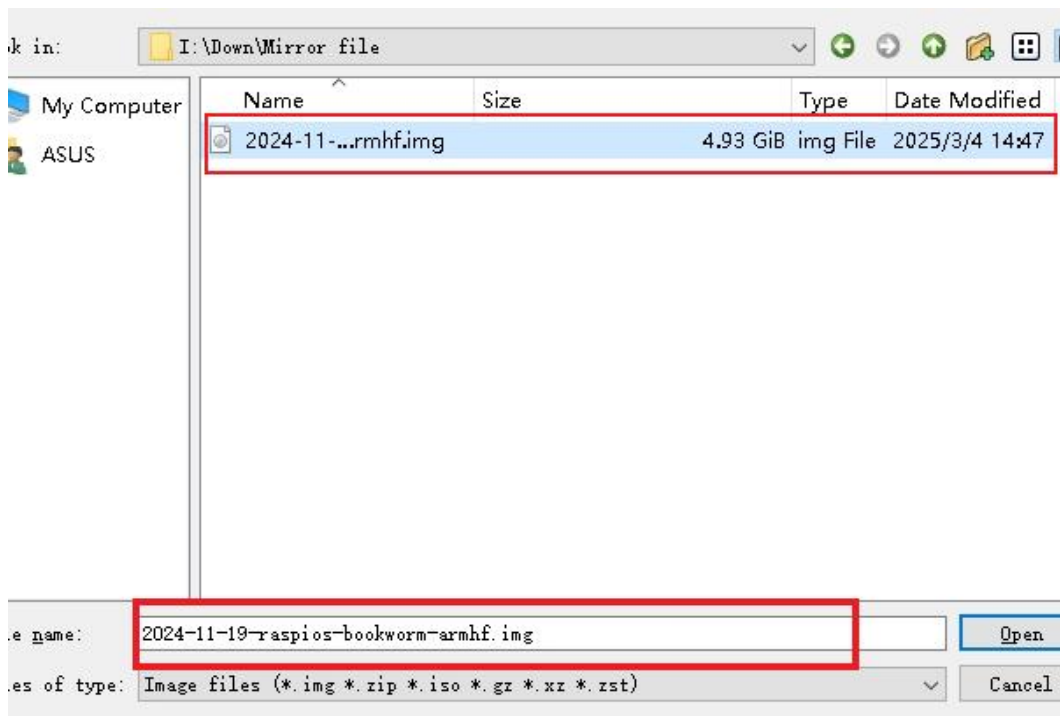
2. Click "**CHOOSE OS**" on the opened Raspberry Pi Imager software interface.



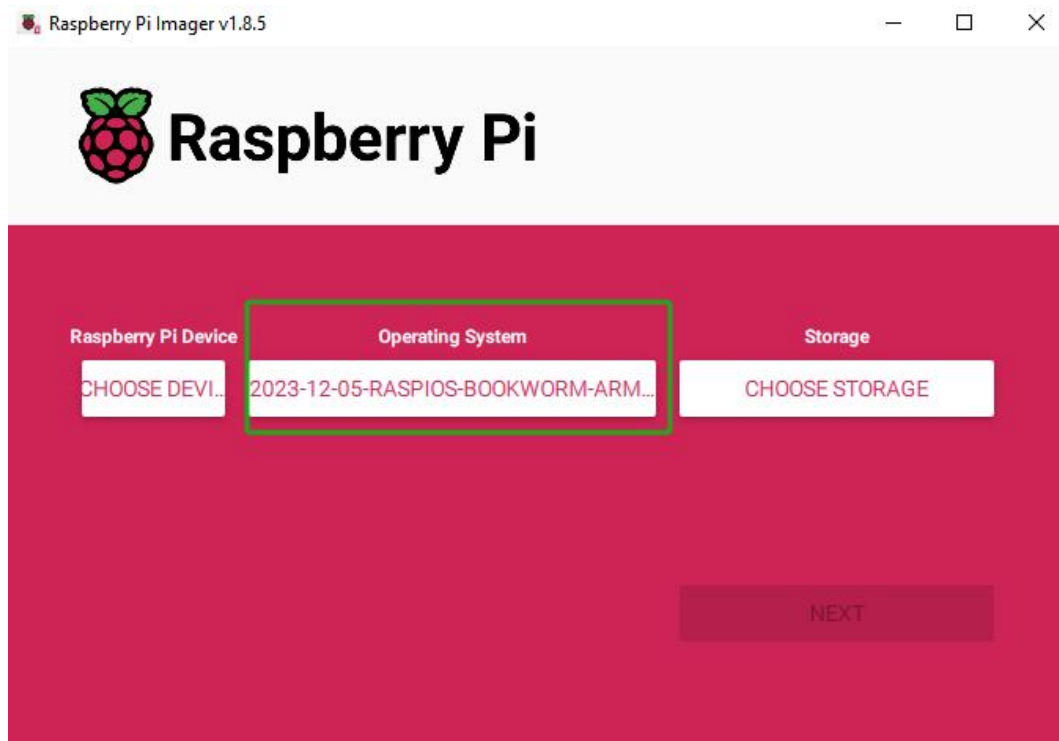
3. Click "**Use custom**" and select a custom ".img" or ".img.xz" file from your computer, which is the ".img" or ".img.xz" file of the Raspberry Pi OS that we downloaded and decompressed before.



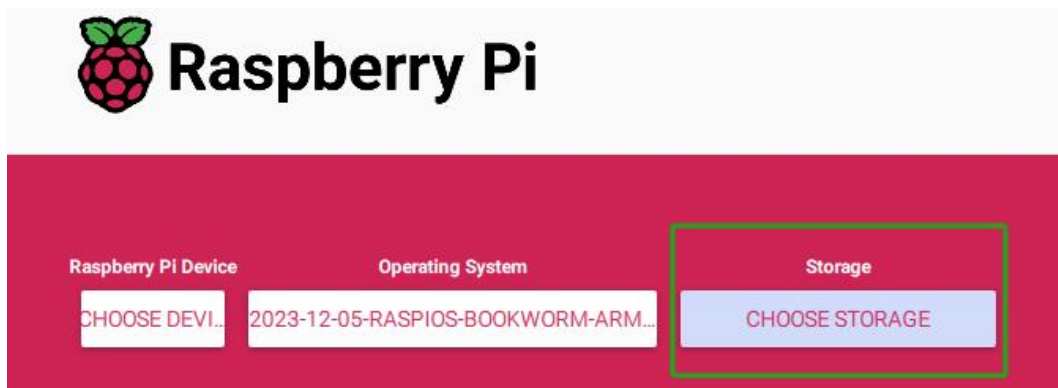
4. Find the ".img" or ".img.xz" file of the Raspberry Pi OS that we downloaded and decompressed before. Click "**Open**".



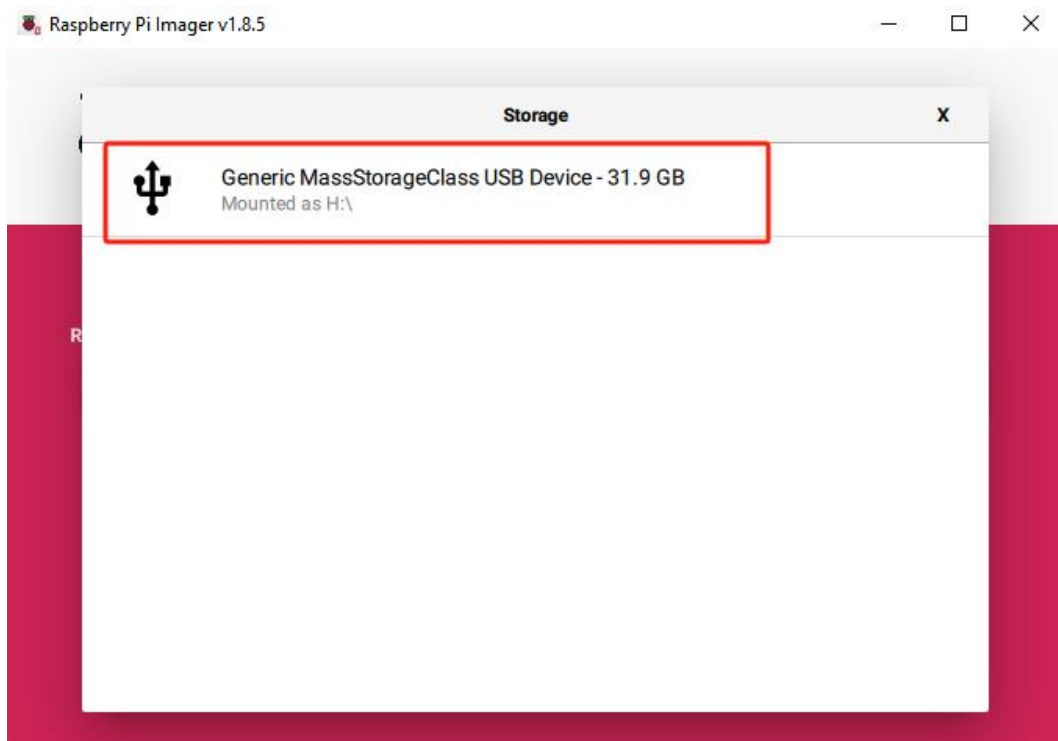
5. Then on the interface of Raspberry Pi Imager, the ".img" file of our selected Raspberry Pi OS will appear.



6. Click "**CHOOSE STORAGE**".



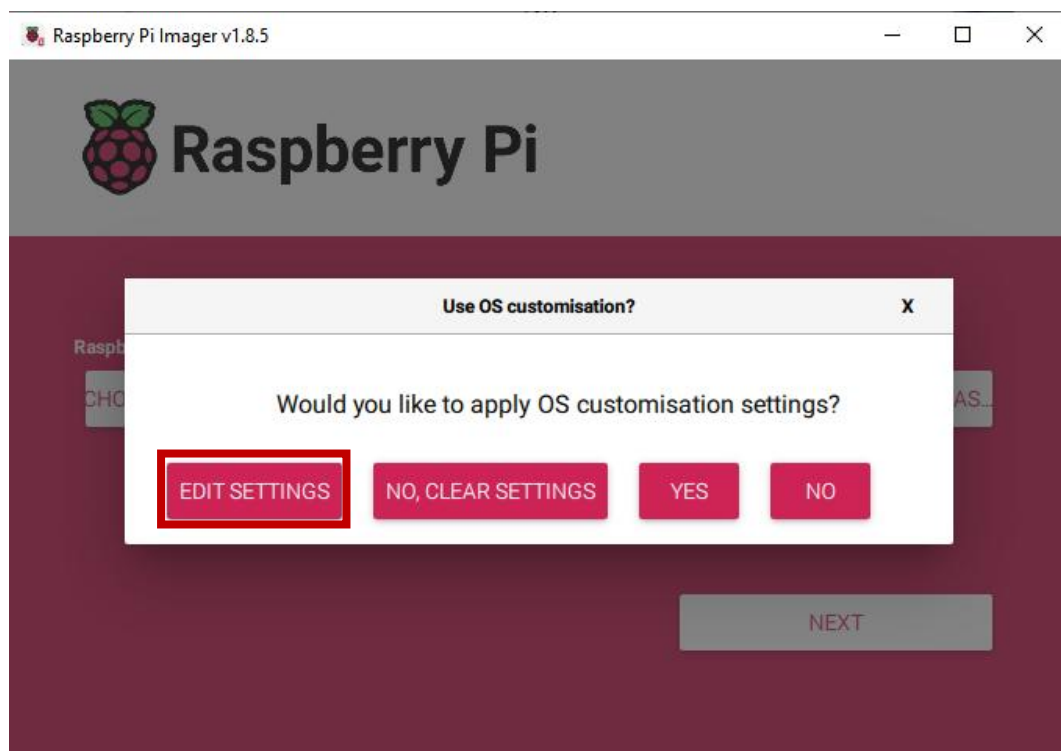
7. Then select the SD card we need to burn.



8. Click "**NEXT**" and use OS customisation.



9. Click "**EDIT SETTINGS**" to set your Raspberry Pi login account and password. Then let the Raspberry Pi connect to your WiFi. This WiFi is in the same LAN as your computer.

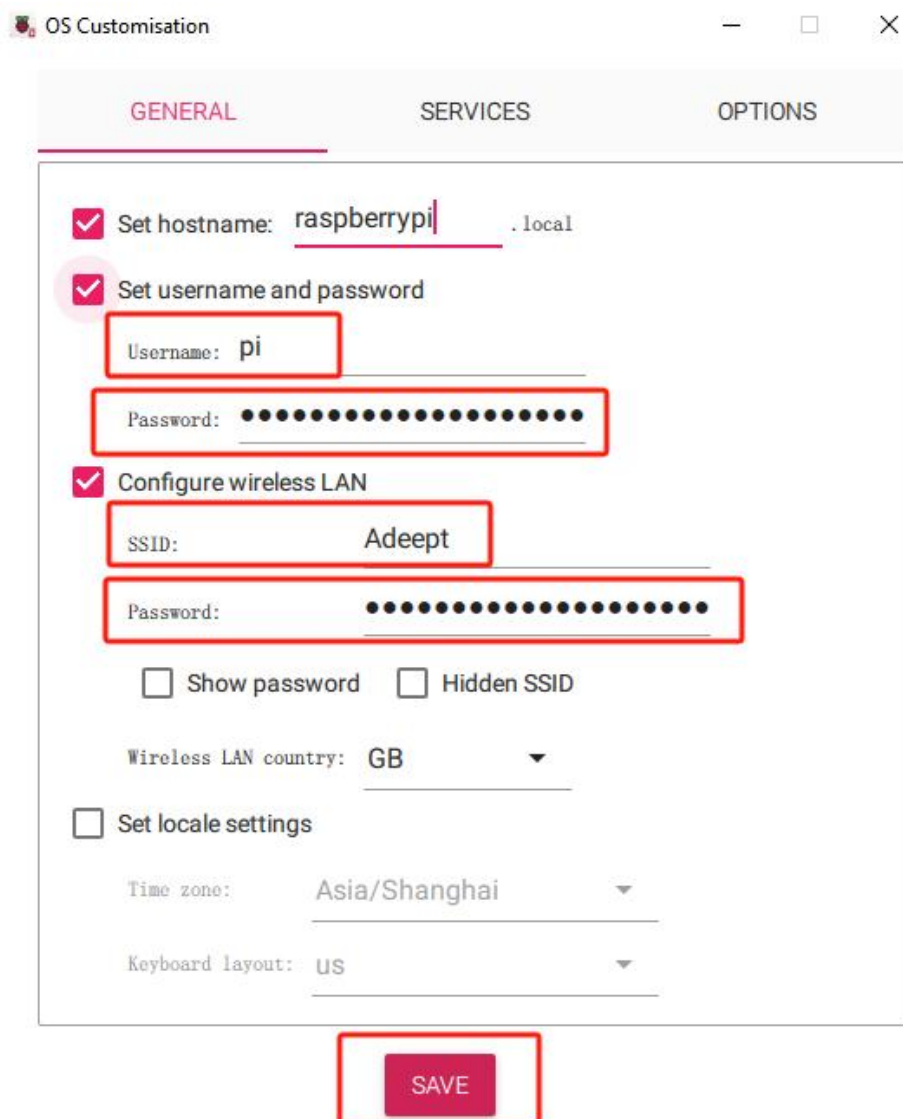


Note: The default account of Raspberry Pi is "pi" and the password is "raspberrypi". Please note that it is case sensitive.

Set the account name you want and a password you can remember. You can also use the Raspberry Pi's default account "pi" and password "raspberrypi".

If you are a beginner, it is recommended to use the default account and password.

Then fill in your WiFi name and password. Our WiFi account and password are written in the picture, you need to modify them according to the actual situation.

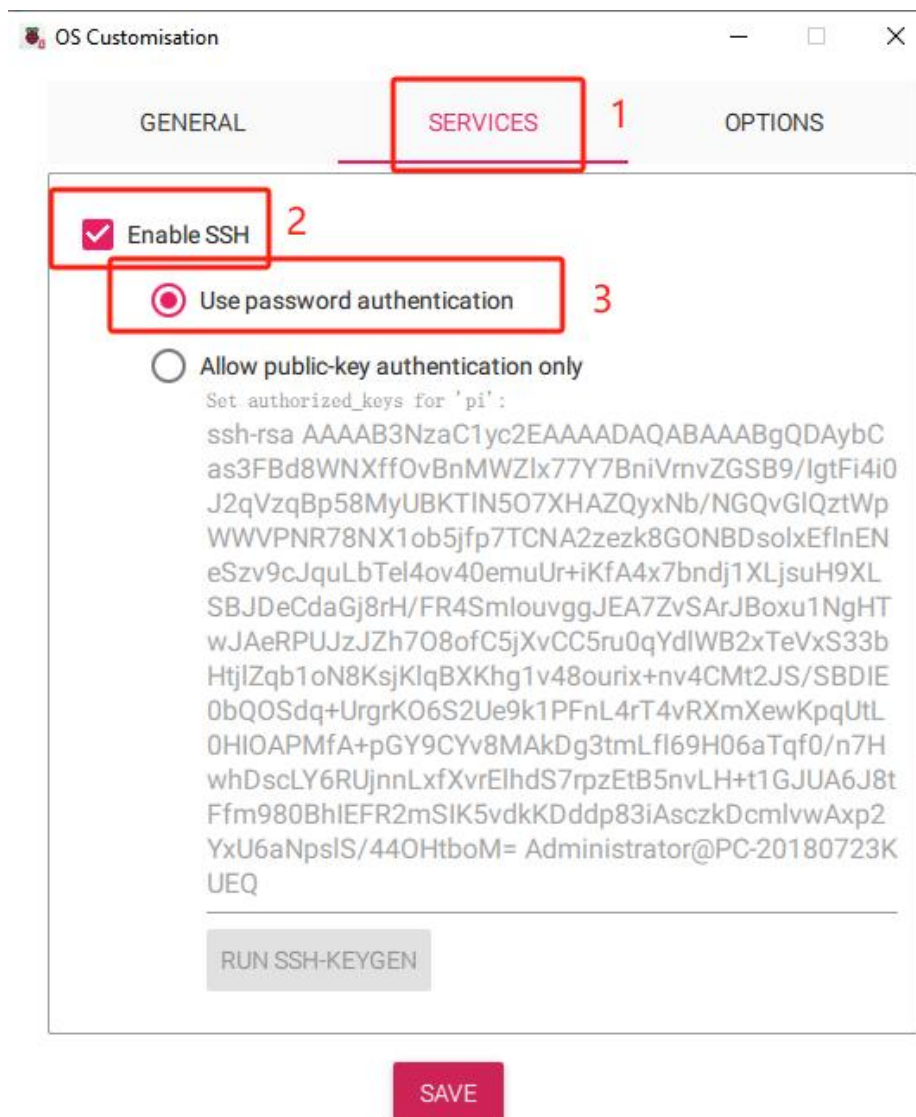


The screenshot shows the 'OS Customisation' window with the 'GENERAL' tab selected. The window has three tabs: 'GENERAL', 'SERVICES', and 'OPTIONS'. The 'GENERAL' tab contains the following settings:

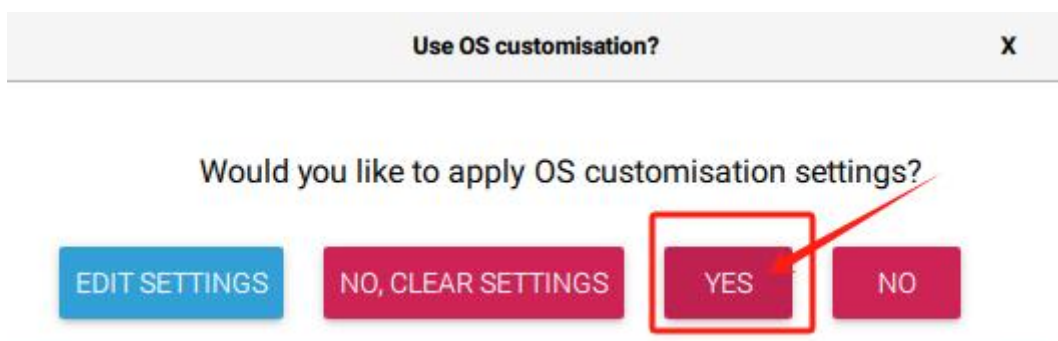
- ☒ Set hostname: raspberrypi.local
- ☒ Set username and password
 - Username: pi
 - Password: [masked]
- ☒ Configure wireless LAN
 - SSID: Adeept
 - Password: [masked]
 - ☐ Show password ☐ Hidden SSID
 - Wireless LAN country: GB
- ☐ Set locale settings
 - Time zone: Asia/Shanghai
 - Keyboard layout: US

A red box highlights the 'SAVE' button at the bottom of the window.

10. Open "SSH".



11. Click "YES".



12. After the burning is completed, a prompt message will appear to indicate that the burning was successful.

【Pay Attention】

If burning the image file is unsuccessful, please try burning it again or replace the SD card and burn the image file again.

3.4 Start the Raspberry Pi OS

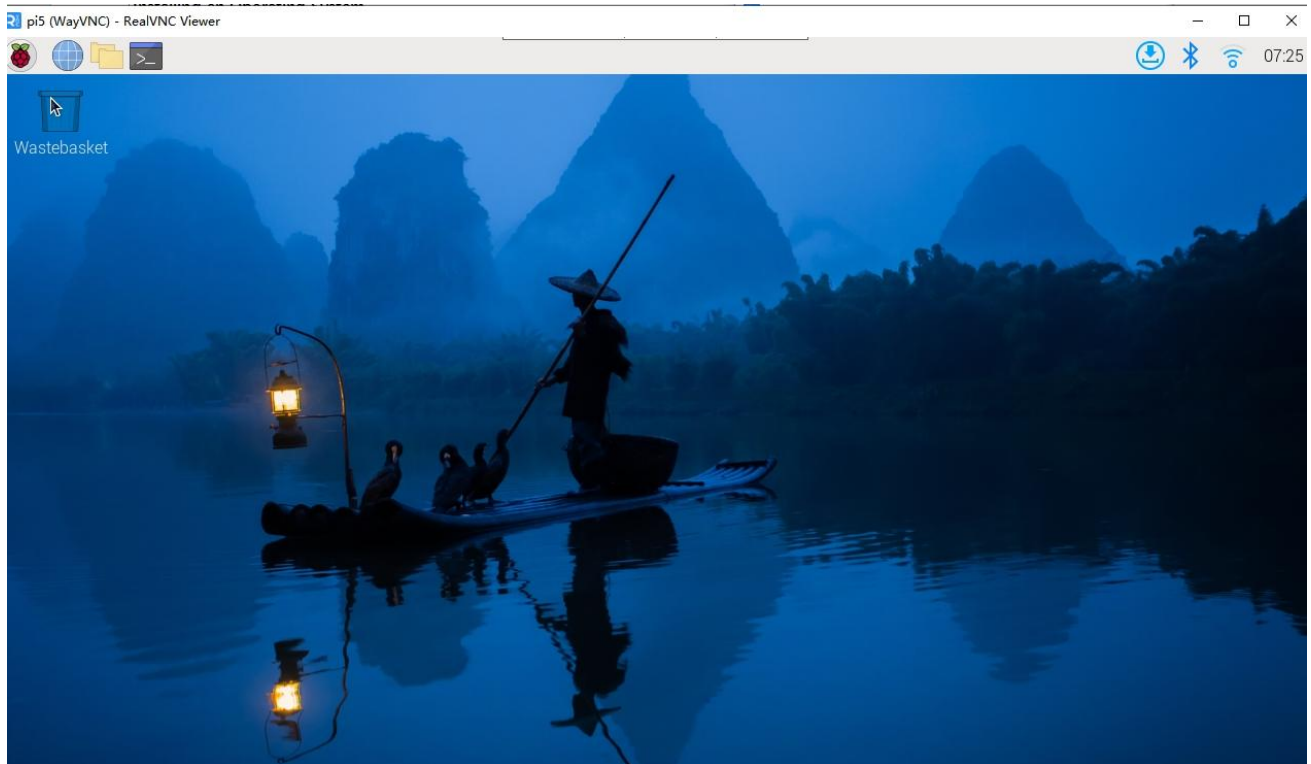
Now, we can take out the SD card and insert it into the "MICRO SD CARD" slot on the Raspberry Pi development board. After that, use a Type-C data cable to supply power to the Raspberry Pi. Then, the Raspberry Pi will start up and run.



If you do not have a spare monitor, please skip to next section [“3.5 Remotely log in to the Raspberry Pi OS”](#) . If you have a spare monitor, please follow the steps in this section.

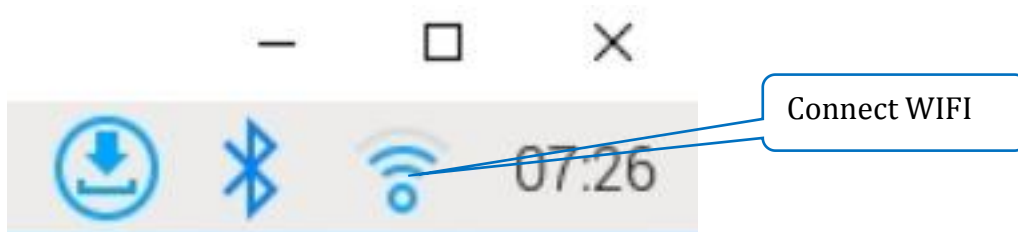
After the system is written successfully, take out Micro SD Card and put it into the SD card slot of RPi. Then connect your RPi to the monitor through the HDMI port, attach your mouse and keyboard through the USB ports, attach a network cable to the network port and finally, connect

your power supply (making sure that it meets the specifications required by your RPi Module Version. Your RPi should start (power up). Later, after setup, you will need to enter your user name and password to login. The default user name: **pi**, password: **raspberrypi**. After login, you should see the following screen.



Congratulations! You have successfully installed the Raspberry Pi OS operating system on your RPi.

Raspberry Pi 5, 4B, 3B+/3B integrates a Wi-Fi adaptor. You can use it to connect to your Wi-Fi. Then you can use the wireless remote desktop to control your RPi. This will be helpful for the following work. Raspberry Pi of other models can use wireless remote desktop through accessing an external USB wireless card.



3.5 Remotely log in to the Raspberry Pi OS

Before using SSH to connect to the Raspberry Pi, you need to know the IP address of the Raspberry Pi and use software that supports the SSH protocol.

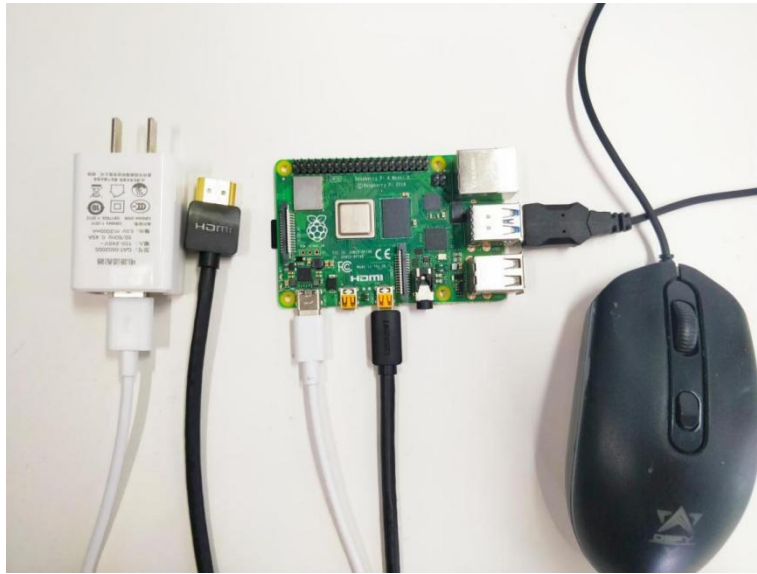
Remote login to the Raspberry Pi is achieved via the SSH protocol. You can use software with SSH support for remote login. For example, Putty is a common choice. For Mac users, if they want to use Putty, they may need to set up a suitable environment (such as using a virtual machine to simulate a Windows environment where Putty can run). Linux systems and Mac OS both come with built-in SSH functionality. Therefore, users of these systems can also log in to the Raspberry Pi remotely through the terminal.

3.5.1 Obtaining the IP address of the Raspberry Pi

Method 1: Obtaining an IP address with an external display


We provide a simple and fast way to get the Raspberry Pi IP address. You need to prepare the following components:

- One Type-C data cable: used to supply power to the Raspberry Pi.
- One HDMI cable: used to connect the monitor.
- One mouse: used to operate.
- One monitor
- One Raspberry Pi



1. Connect the HDMI cable to the HDMI port of the monitor:

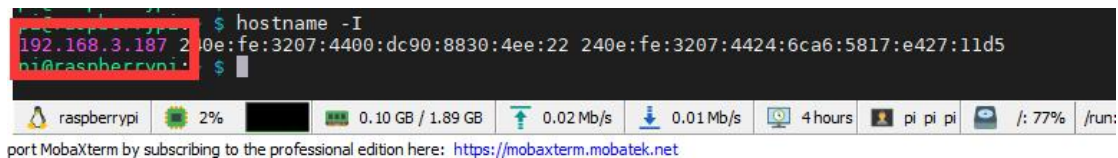


2. Turn on the monitor switch. Connect the mouse to a USB port of the Raspberry Pi and supply power to the Raspberry Pi with the Type - C data cable. Then the Raspberry Pi starts. After entering the system interface, we move the mouse cursor to the "  03:35 " in the upper right corner, and it will display the IP address of the Raspberry Pi: 192.168.3.157 (the IP address of each Raspberry Pi is different). It is necessary for you to record this IP address as it is needed to log in to the Raspberry Pi OS later.



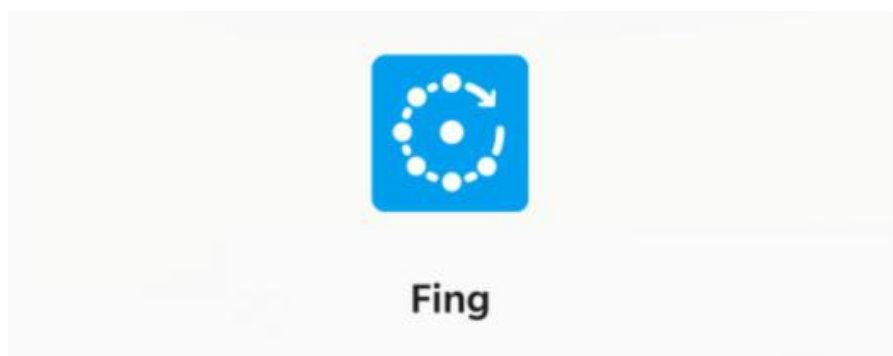
3. You can also check the IP address by opening the command window of the Raspberry Pi and entering the following command. Make sure to write it down.

hostname -I

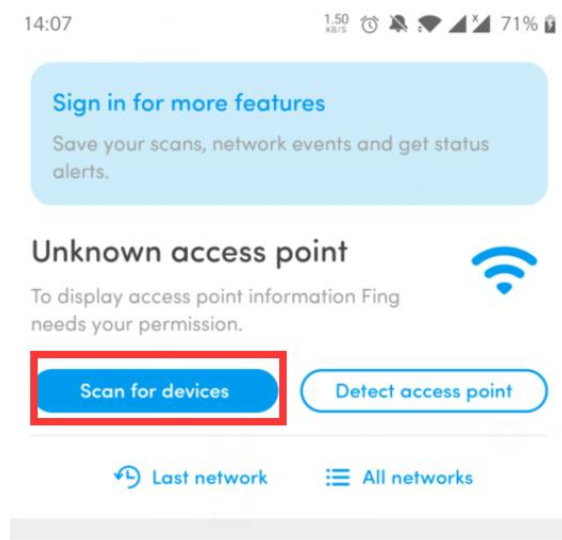


Method 2: Obtaining an IP address with a mobile phone

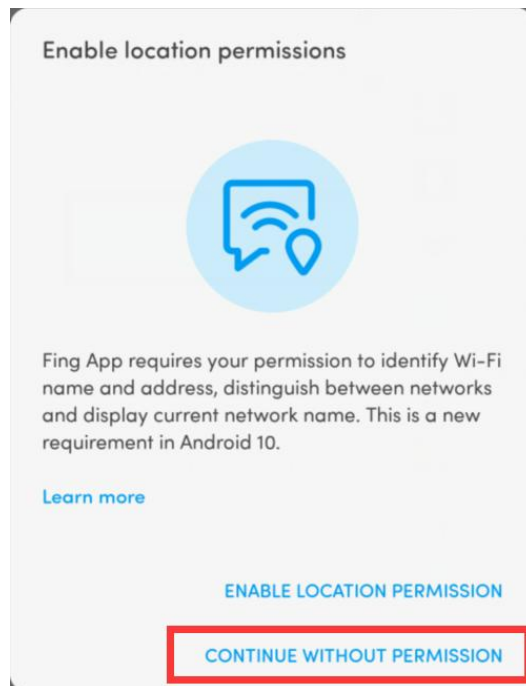
1. You need to download an APP called "**Fing**" on your phone, as shown below:



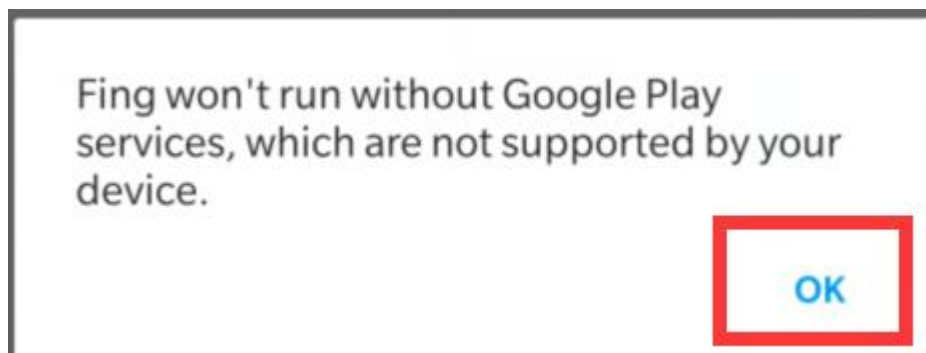
2. After you have completed downloading the "**Fing**" app, your phone and the Raspberry Pi should be on the same local area network, that is, both of them are connected to the same Wi-Fi network. Then, open the "**Fing**" app and click on "**Scan for devices**".



3. Click "**CONTINUE WITHOUT PERMISSION**" :



4. Click "OK" :



5.Wait for the scan to finish. In the list of scanned devices, locate the device named "**Raspberry Pi**". You'll be able to see its IP address, [192.168.3.157](#), in the lower left corner of the relevant interface. Make sure to write down this IP address.



3.5.2 Remotely log in to the Raspberry Pi OS

This course will demonstrate how to log in to the Raspberry Pi using PuTTY and VNC. Both Linux systems and Mac OS systems come with the SSH function. You can remotely log in to the Raspberry Pi using the terminal without the need to download any software.

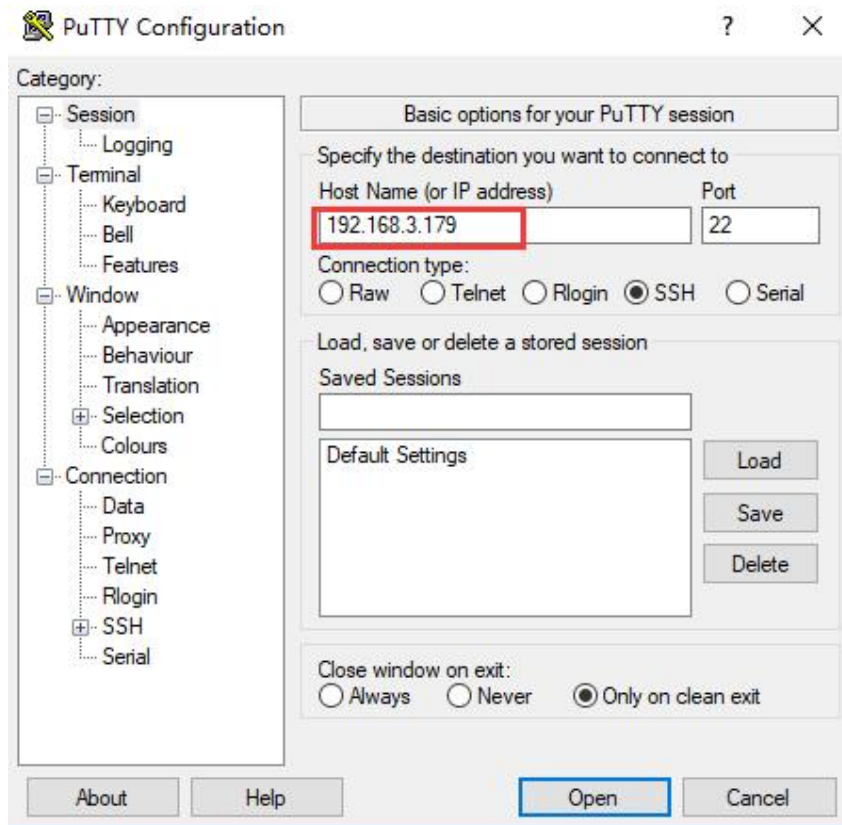
1.PuTTY

You need to download and install **PuTTY** corresponding to your computer OS version, and use it to log in to the Raspberry Pi. **PuTTY** download address:

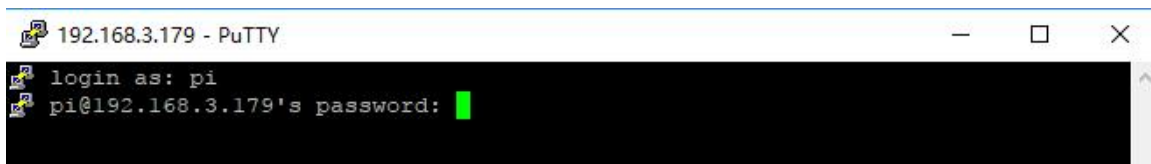
<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

1. Run PuTTY, enter the IP address of the Raspberry Pi into the Host Name, and click **"Open"** .

For example: 192.168.3.179

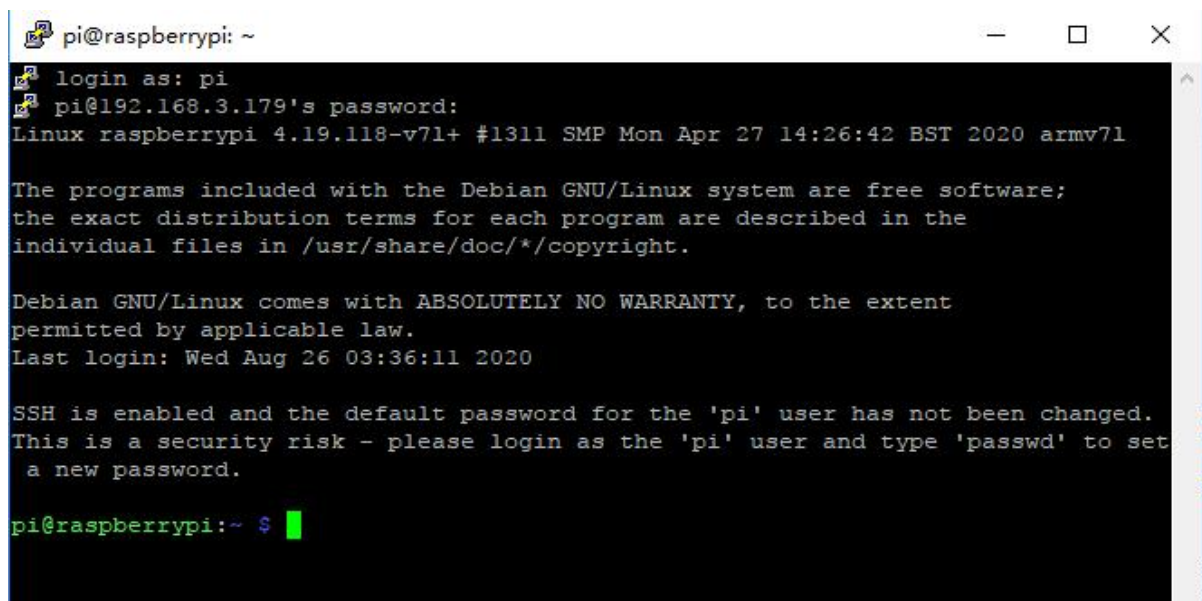


2. If it prompts Network error: Connection timed out, it means you probably entered the wrong IP address.
3. When the connection is normal, you will see a security warning. You can safely ignore it and click the **"Yes"** button. You will see this warning when PuTTY connects to a Raspberry Pi that has not been connected before.
4. You will see the usual login prompt now. Log in with the same username and password as those of the Raspberry Pi system. The default login name for Raspbian is **"pi"** and the default password is **"raspberry"**. When entering the password, nothing will be shown on the screen.



Note: If you have changed the Raspberry Pi account and password (for example, using a non-default account and password when flashing the Raspberry Pi system), you need to accurately enter the username and password that you set during this process.

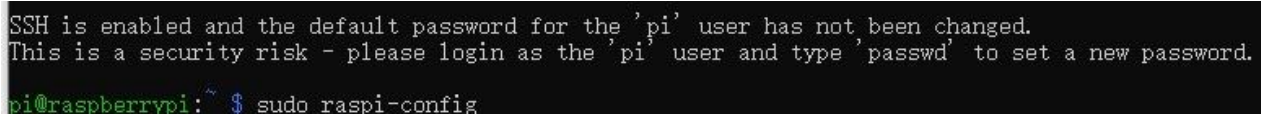
5. You should now have the Raspberry Pi prompt, which will be the same as the prompt on the Raspberry Pi itself.

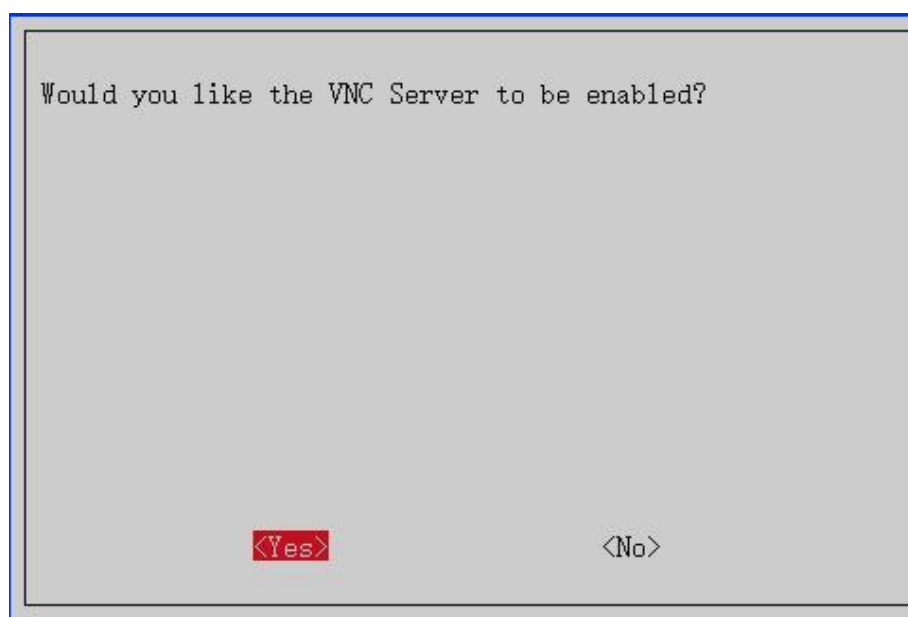
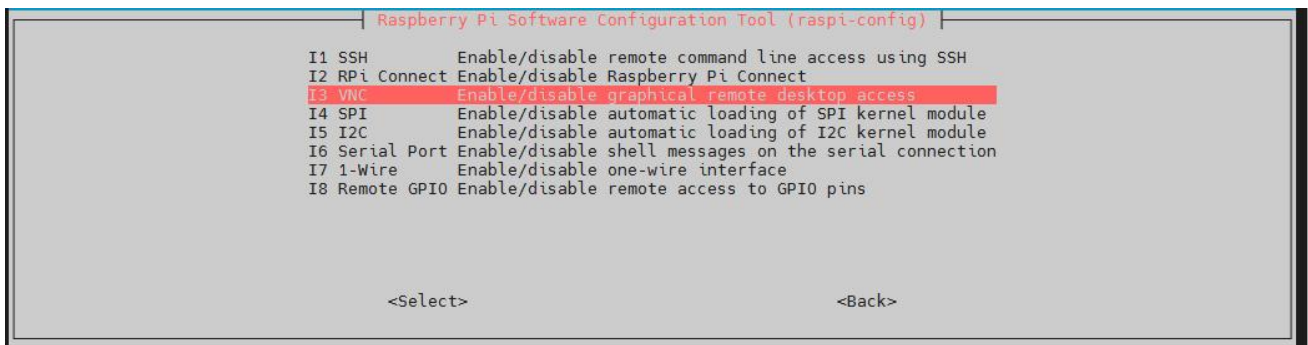


2. VNC

Type the following command. Then, select "**Interface Options**", and among the options, find and select the item related to enabling VNC (for example, the option labeled as something like "**I3 VNC**" which is related to the VNC interface setting). Press **Enter**, then select "**Yes**" and click "**OK**". At this stage, the Raspberry Pi may need to be restarted. Confirm the restart by selecting "**OK**". After that, open the VNC interface.

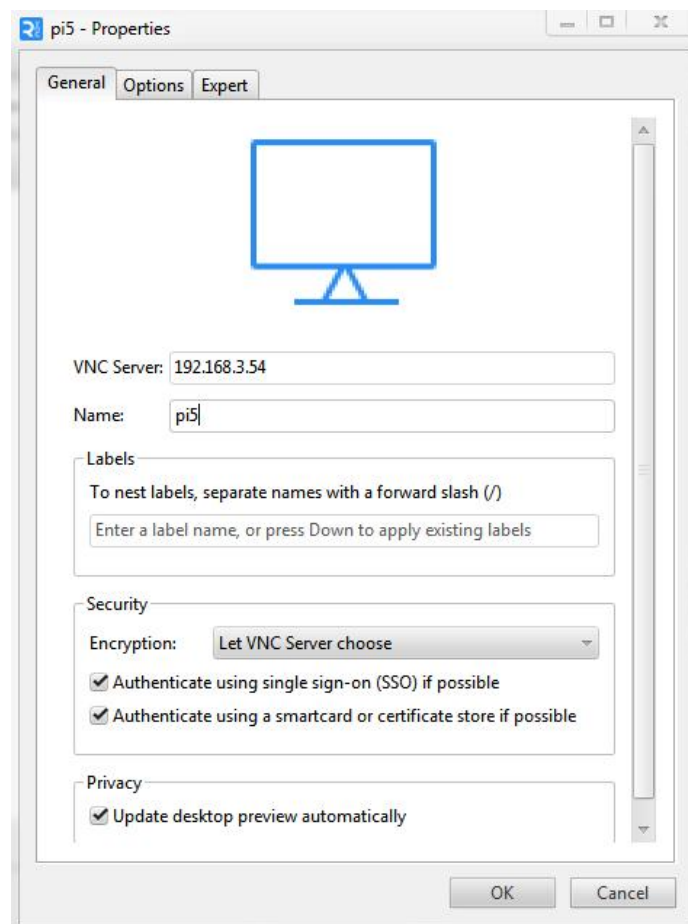
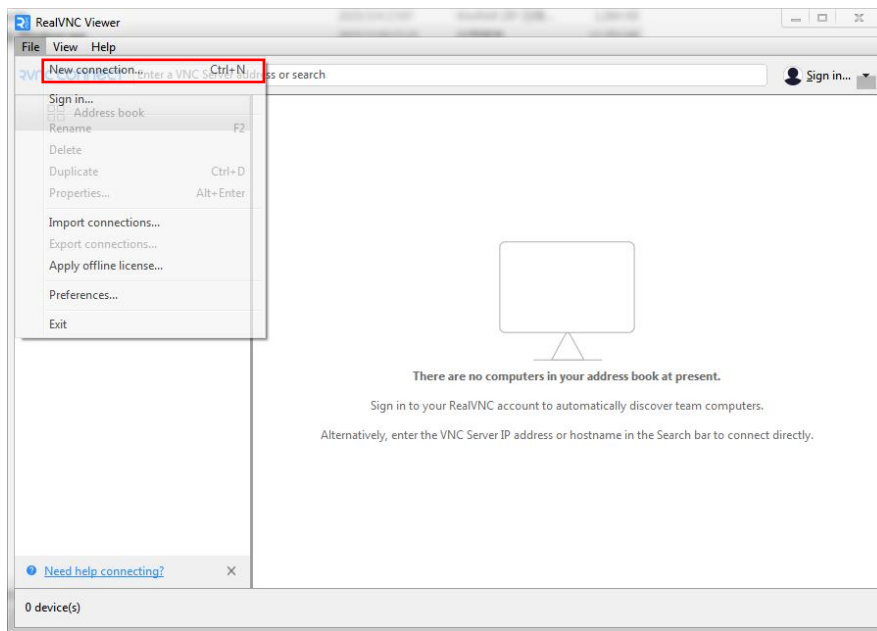
```
sudo raspi-config
```





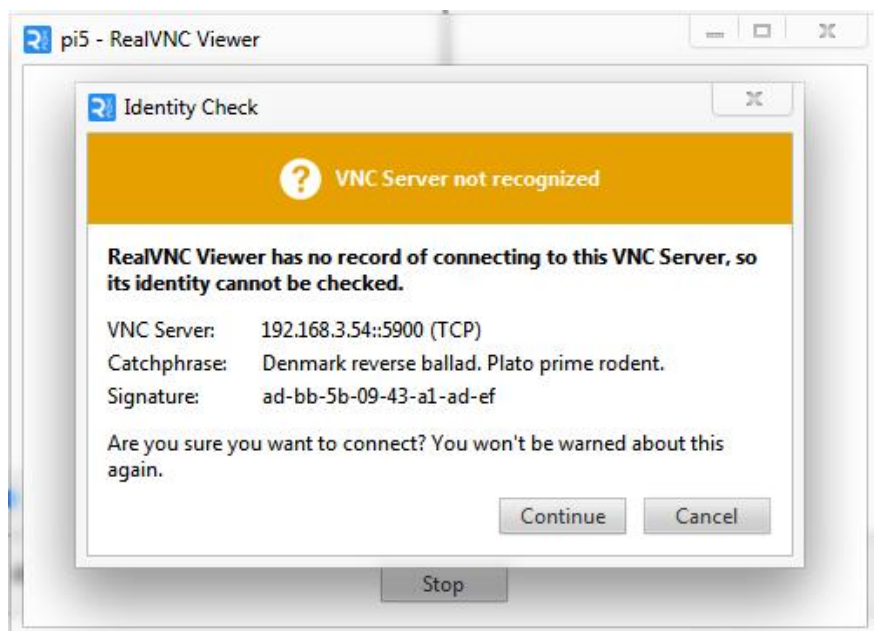
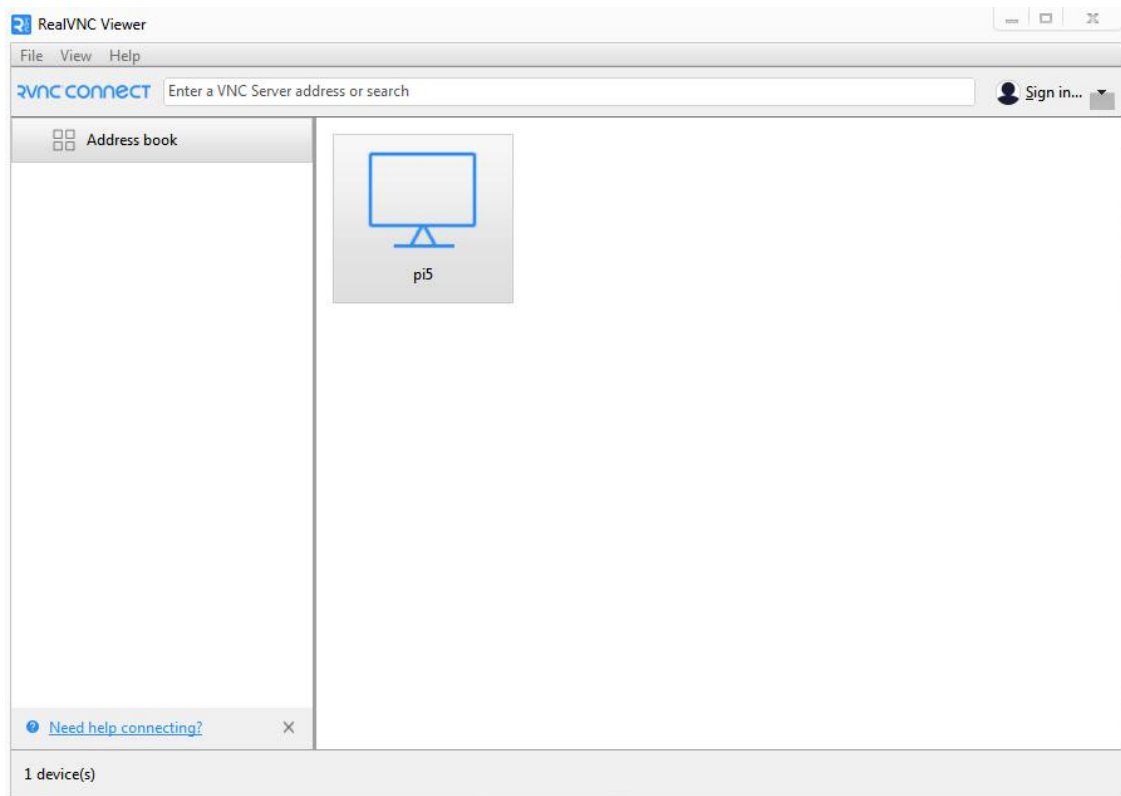
Then download and install VNC Viewer according to your computer system by click following link: <https://www.realvnc.com/en/connect/download/viewer/>

After installation is completed, open VNC Viewer. And click File New Connection. Then the interface is shown below.

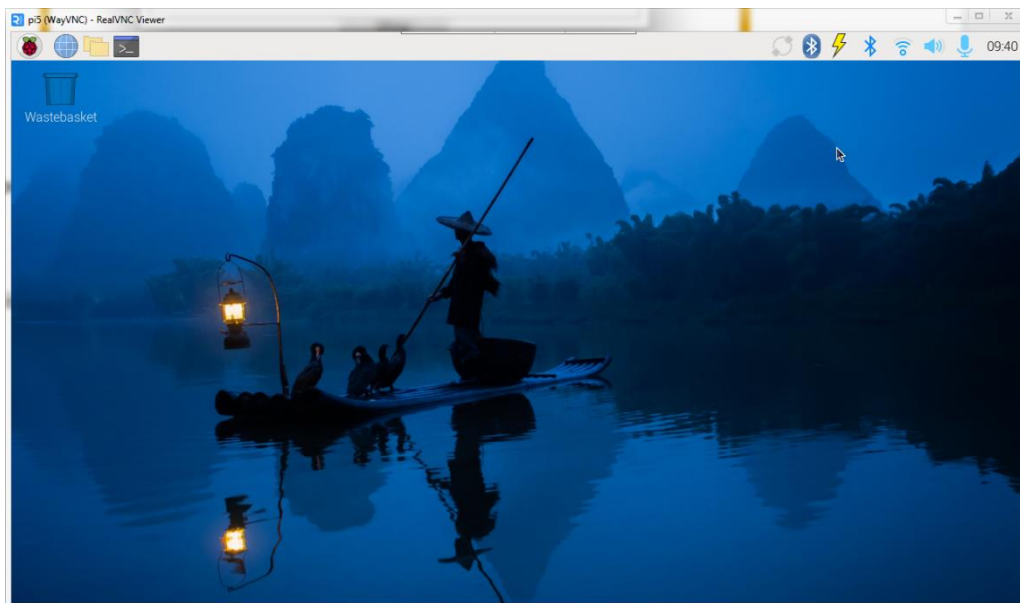
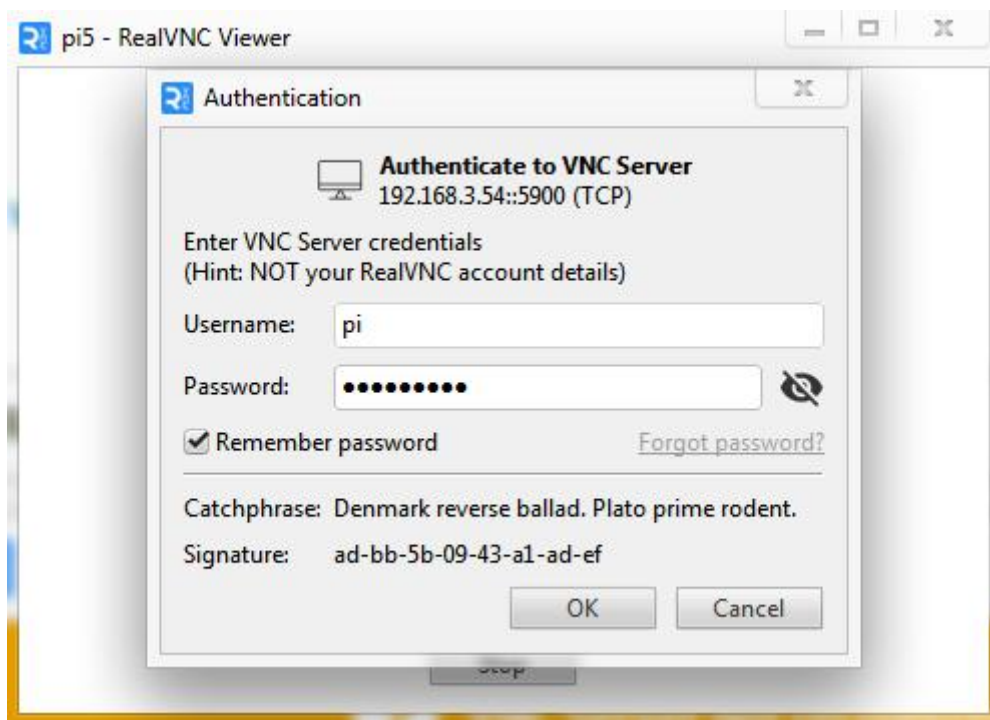


Enter the **IP** address of your Raspberry Pi and fill in a connection name. Then click **OK**.

Then on the VNC Viewer panel, double-click the new connection you just created, and the following dialog box pops up.

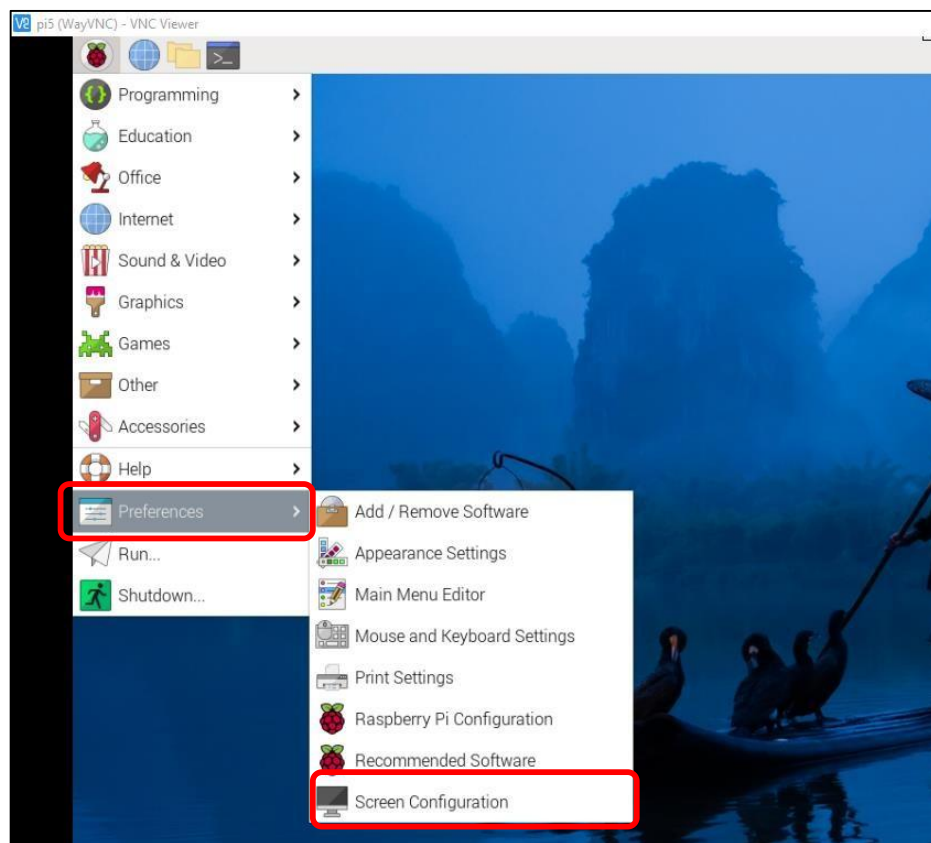
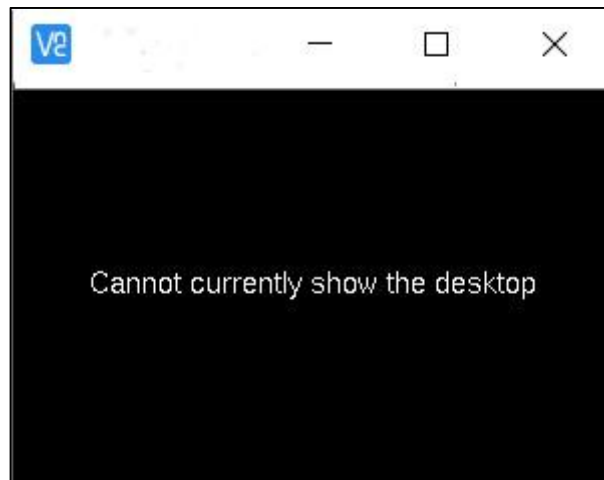


Enter username: **pi** and Password: **raspberry**. And click **OK**.

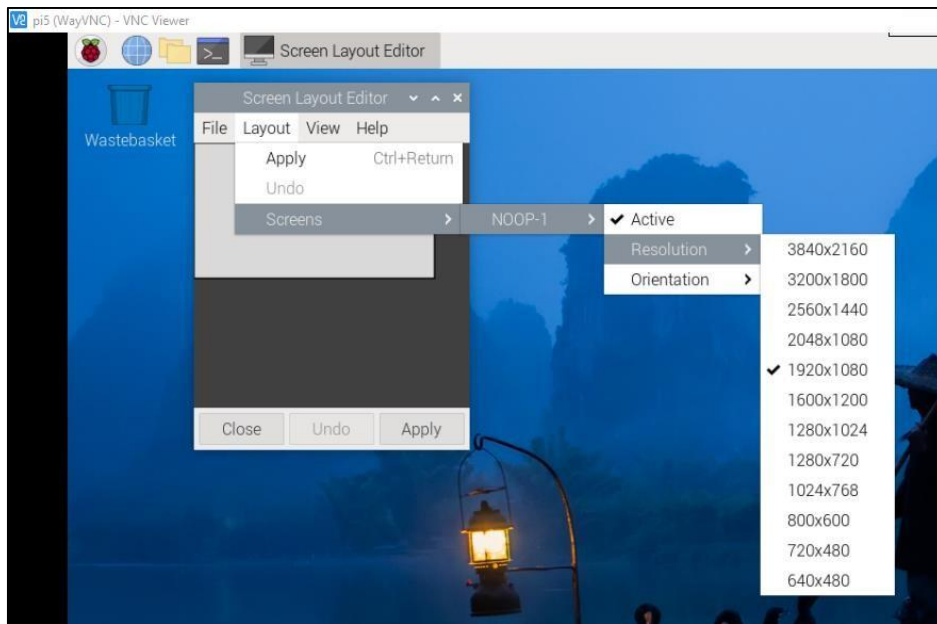


Here, you have logged in to Raspberry Pi successfully by using VNC Viewer

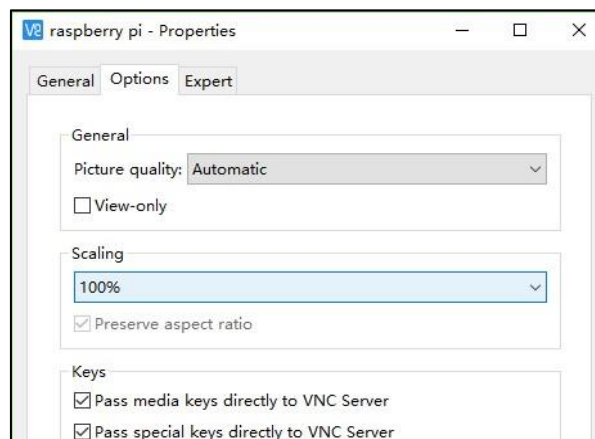
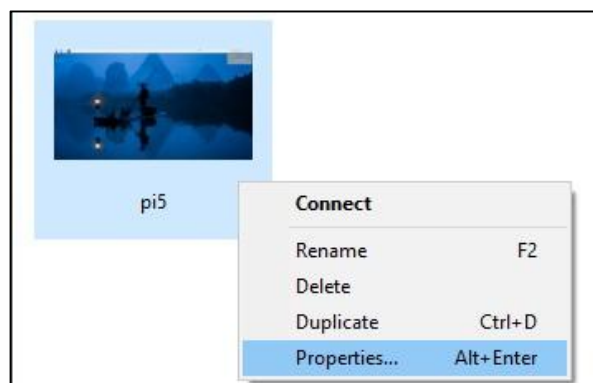
If there is a black window, please [set the resolution](#).



If you don't know what resolution to set properly, you can try 1920x1080.



In addition, the VNC Viewer window may scale the desktop of your Raspberry Pi. You can adjust this setting. On the VNC Viewer control panel, right-click (click the right mouse button). Then, select "Properties" -> "Options tab" -> "Scaling". After that, set an appropriate scaling value.



The Raspberry Pi 5, 4B, 3B+, and 3B all integrate a WiFi adaptor. So, if you haven't connected your Raspberry Pi to a WiFi network, you can connect it to a WiFi network wirelessly to control the robot.

3. Windows10, Linux and Mac OS come with SSH function

Steps to connect to Raspberry Pi via SSH:

1. Open a console terminal window.
2. The initial user name of the Raspberry Pi is **"pi"** and the initial password is **"raspberrypi"**.
3. Enter **ssh pi<IP>** in the command line and replace **<IP>** with your Raspberry Pi IP address, as shown in the following example:

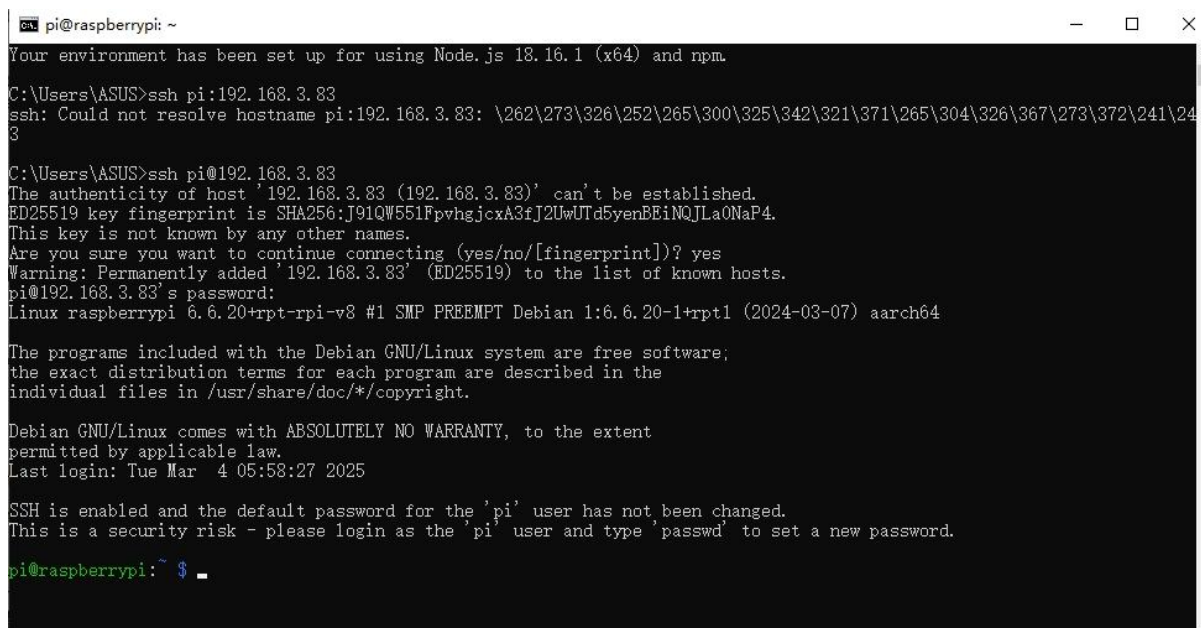
```
ssh pi@192.168.3.83
```

4. Press **Enter**, and the prompt "Are you sure you want to continue connecting (yes/no)?" will appear.

5. Enter **'yes'** and press **Enter**. Then the prompt **'pi@192.168.3.83's password:'** will appear. Enter the initial password **"raspberrypi"** of the Raspberry Pi and pay attention to the case sensitivity. During the password input process, nothing will change on the screen, but this doesn't indicate that the input was unsuccessful. Press **Enter** after you have completed the input.

6. Now you have logged in to the Raspberry Pi.

Windows10



```

C:\Users\ASUS>ssh pi@192.168.3.83
ssh: Could not resolve hostname pi:192.168.3.83: \262\273\326\252\265\300\325\342\321\371\265\304\326\367\273\372\241\243
C:\Users\ASUS>ssh pi@192.168.3.83
The authenticity of host '192.168.3.83 (192.168.3.83)' can't be established.
ED25519 key fingerprint is SHA256:J91QW551FpvhgjcxA3tJ2UwUTd5yenBEiNQJLa0NaP4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.3.83' (ED25519) to the list of known hosts.
pi@192.168.3.83's password:
Linux raspberrypi 6.6.20+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.20-1+rpt1 (2024-03-07) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar  4 05:58:27 2025

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.
pi@raspberrypi:~ $

```

Linux and Mac OS

```
adeept — pi@raspberrypi: ~ — ssh pi@192.168.3.54 — 139x43
Last login: Wed Mar 19 14:44:19 on ttys000
[adeept@AdeeptdeMacBook-Pro ~ % ssh pi@192.168.3.54
pi@192.168.3.54's password:
Linux raspberrypi 6.6.31+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-05-29) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Mar 19 06:25:52 2025 from 192.168.3.88

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ ls
ADR019          Downloads      Speech.py      adeept_darkpaw  output.wav      test.wav
Adeept_PiCar-Pro ExampleOfAI.py Speech_Recognition adeept_rasptank2 response_audio.wav testVoice
Adeept_Rasptank_Metal Mpu6050.py    Speech_Recognition.py create_ap        sherpa-ncnn     testVosk
Arecored.py     Music          Templates      demo            startup.sh       tts_http_demo.py
Bookshelf       Pictures       Text.py        flask-video-streaming startup.sh.Metal.bak
Desktop         Public         TimeOled.py    image.jpg        startup.sh.rasptank2.bak
Documents      SnowOled.py   Videos        output.txt       temp.mp3
```

3.6 Download the Code Program to Control the Robot

- For the power supply of Raspberry Pi, please refer to this official document:

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md>

- Our Raspberry Pi robot driver board, Adeept Robot HAT V3.2, can directly supply power to the Raspberry Pi through the GPIO pins. However, because the software installation time on the Raspberry Pi is relatively long, it is not recommended to use battery power when installing software on the Raspberry Pi. The Raspberry Pi robot driver board, Adeept Robot HAT V3.2, or the camera do not need to be installed when installing software on the Raspberry Pi. This does not affect the software installation. But when you run the installed program, you must connect the driver board and the Raspberry Pi camera. Otherwise, it will cause the program to report an error.
- All the code of our product has been open sourced on GitHub. You need to download it to the Raspberry Pi and install the relevant dependencies before it can run normally.

github address: https://github.com/adeept/Adeept_4WD_Smart_Car_for_RPi

- Enter the following commands in the console:

```
sudo git clone https://github.com/adeept/Adeept_4WD_Smart_Car_for_RPi.git
```

After the input is complete, Press **Enter** to start downloading the robot program from GitHub. This process will continue for a period of time. Wait patiently for the download to complete.

```
pi@raspberrypi:~ $ sudo git clone https://github.com/adeept/Adeept_4WD_Smart_Car_for_RPi.git
Cloning into 'Adeept_4WD_Smart_Car_for_RPi'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
```

After the download is complete, a new folder '**Adeept_4WD_Smart_Car_for_RPi**' will appear, in which the product code is stored. You can check the content of the folder through the Linux command '**ls**' .

```
ls
```

```
pi@raspberrypi:~ $ ls
Adeept_4WD_Smart_Car_for_RPi  FlowingLights.py  Public
```

3.7 Install the Dependency Library of the Program

- When installing dependent libraries, the installation of some dependent libraries may fail due to network problems or dependent library version issues. For dependent libraries that fail to install, the program will automatically install multiple times.
- After the installation is complete, the Raspberry Pi will automatically restart. At this time, the remote login program of the Raspberry Pi via SSH will be terminated. Please log in again.
- When installing dependent libraries, the installation may fail due to network or regional reasons. If you encounter any problems, please contact us: support@adeept.com

We have prepared a script to install all the dependent libraries that need to be used and set up operations such as turning on the camera and automatically running on startup.

Before that, you need Open I2C and SPI.

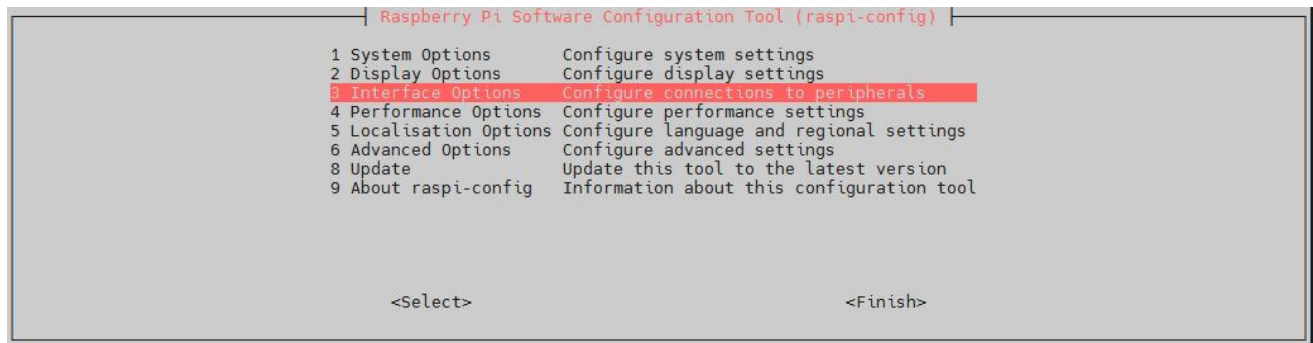
Open I2C:

Enter the following command in the Raspberry Pi:

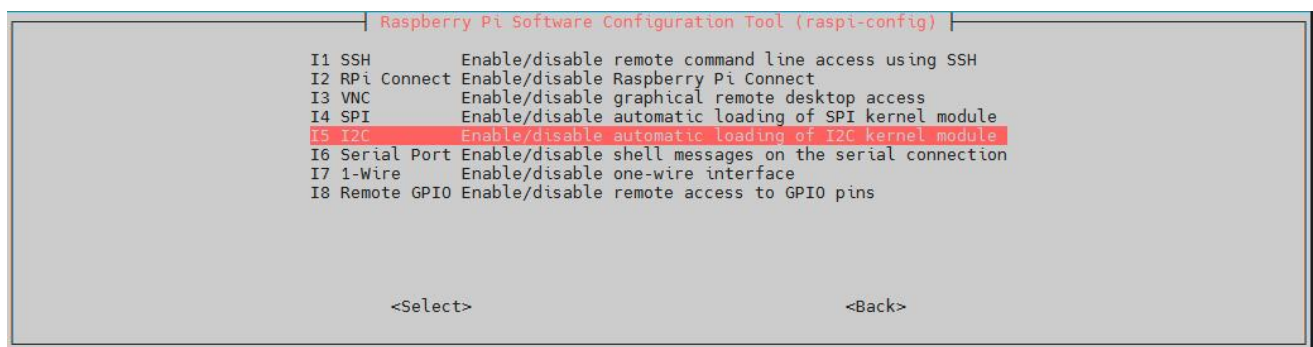
```
sudo raspi-config
```

```
pi@raspberrypi:~ $ sudo raspi-config
```

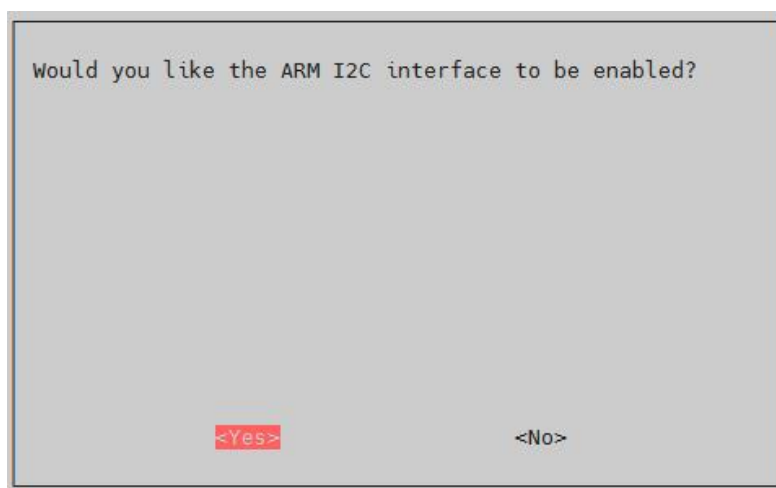
Use the keyboard arrow keys to select "**3 Interface options**" and press "**Enter**".



Select "**I5 I2C**" and press "**Enter**".



Select "**Yes**" and press "**Enter**" to open i2c.



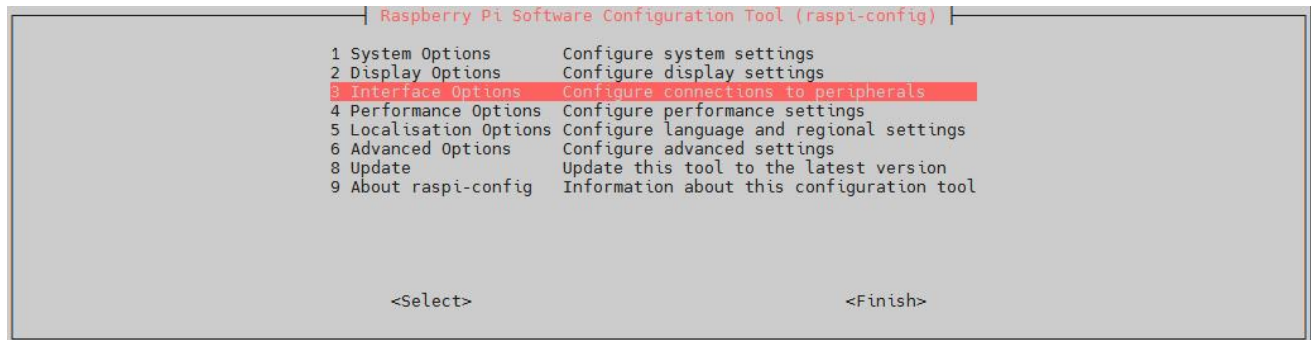
Open SPI:

Enter the following command in the Raspberry Pi:

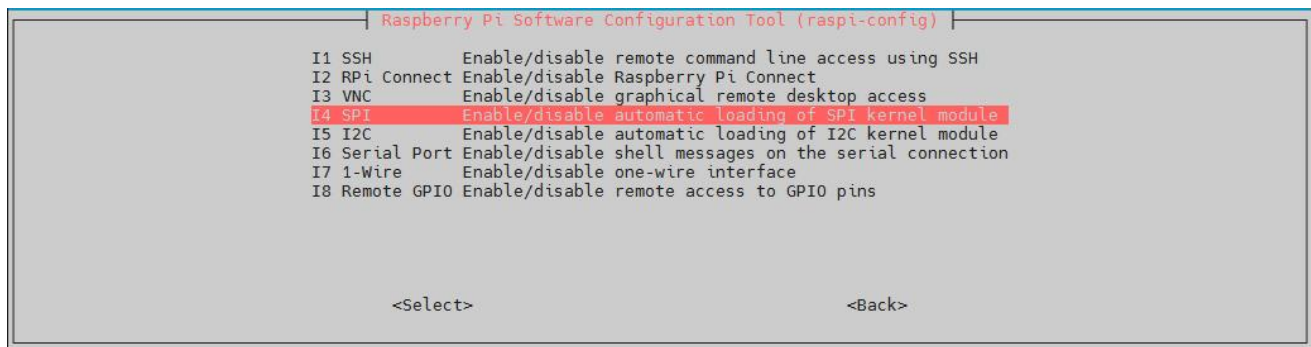
```
sudo raspi-config
```

```
pi@raspberrypi:~$ sudo raspi-config
```

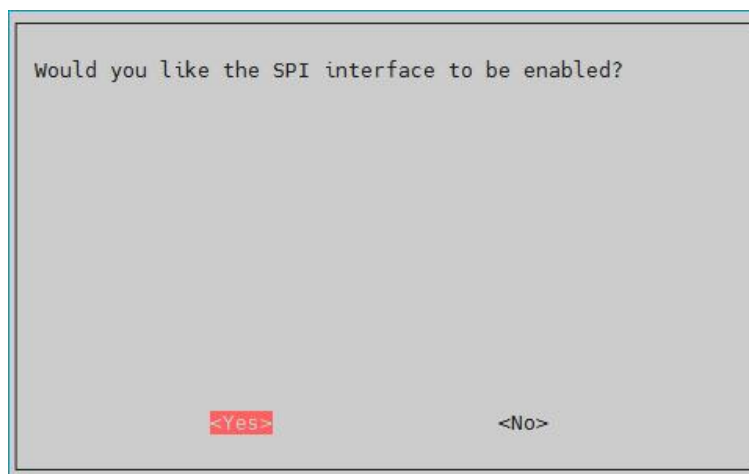
Use the keyboard arrow keys to select "**3 Interface options**" and press "**Enter**".



Select "**I4 SPI**" and press "**Enter**".



Select "**Yes**" and press "**Enter**" to open SPI.



1. Enter the following code in the console and run the script to install the required dependent libraries. The following operations are automatically completed by the script program. Depending on the network environment, this process may last for tens of minutes or several hours, just wait patiently.

Note: This product has two versions: the Mecanum wheel version and the Ordinary wheel version. Please select the corresponding course according to the version.

Below is an example using Mecanum wheel version.

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_MecanumWheels.py
```

```
pi@raspberrypi:~$ sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_MecanumWheels.py
curpath://home/pi/Adeept_4WD_Smart_Car_for_RPi
Hit:1 http://deb.debian.org/debian bookworm InRelease
Get:2 http://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:3 http://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:4 http://archive.raspberrypi.com/debian bookworm InRelease [55.0 kB]
```

If it is Ordinary wheel version, please execute the following command:

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_OrdinaryWheels.py
```

```
pi@raspberrypi:~$ sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_OrdinaryWheels.py
curpath://home/pi/Adeept_4WD_Smart_Car_for_RPi
Hit:1 http://raspbian.raspberrypi.com/raspbian bookworm InRelease
Hit:2 http://archive.raspberrypi.com/debian bookworm InRelease
```

2. After the installation is complete, the console will show text:

```
The program in Raspberry Pi has been installed, disconnected and restarted.
You can now power off the Raspberry Pi to install the camera and driver board (Robot HAT).
After turning on again, the Raspberry Pi will automatically run the program to set the servos port signal to turn the servos to the
middle position, which is convenient for mechanical assembly.
restarting...

Broadcast message from root@raspberrypi on pts/2 (Tue 2025-05-13 06:50:32 BST):
The system will reboot now!

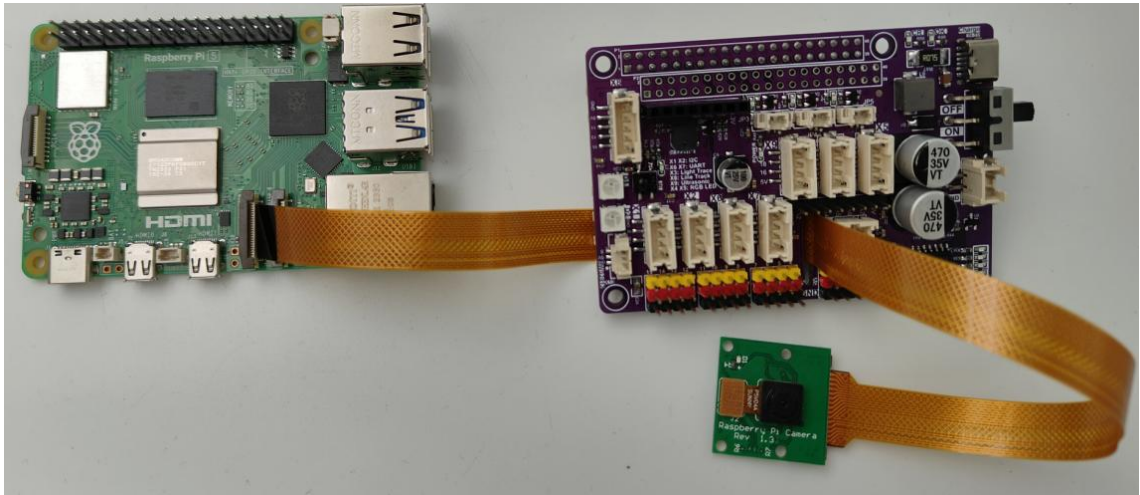
Broadcast message from root@raspberrypi on pts/2 (Tue 2025-05-13 06:50:32 BST):
The system will reboot now!
```

After the installation is complete, the Raspberry Pi will automatically disconnect the SSH connection and restart. At this time, if you are using a Raspberry Pi connected by software such as Putty, there will be an error message such as Network error: Software caused connection abort, which is normal, just close the window, Then reconnect to the Raspberry Pi via SSH.

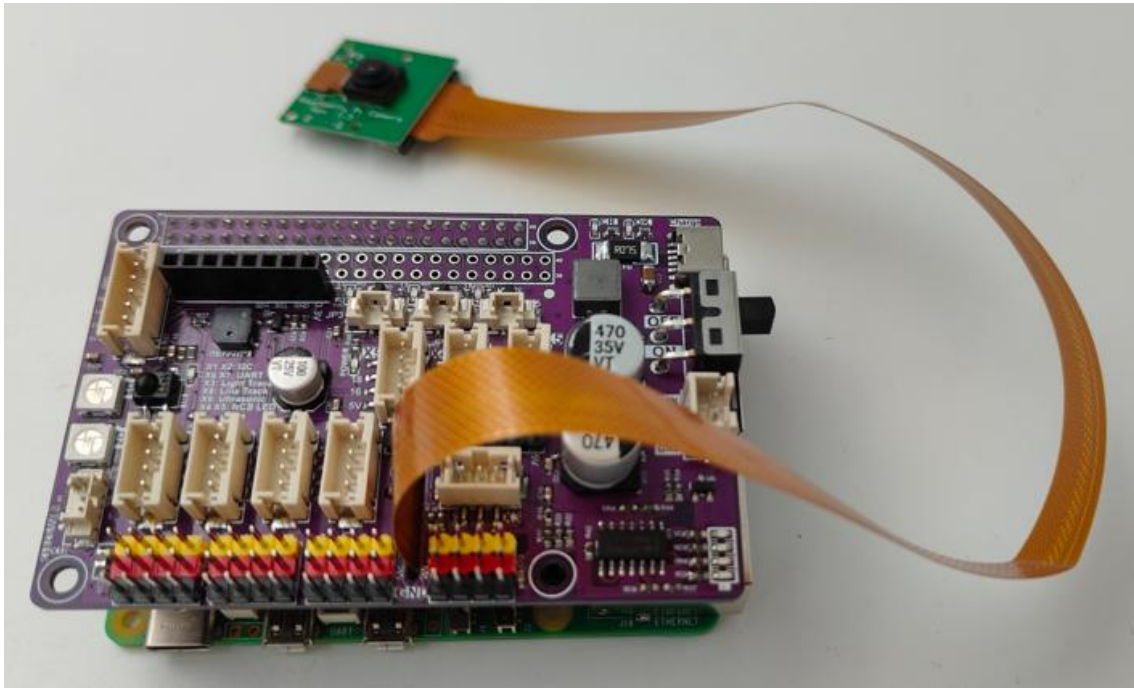
3.8 Check whether the dependent library is installed successfully

The product in this example is the Raspberry Pi 5, which is equipped with a Raspberry Pi camera. The connection status is shown in the following picture.

1. Connect the camera to the camera port of the Raspberry Pi.



2. Install Adept Robot HAT V3.2 and camera, and connect the Raspberry Pi power supply. If you do not connect too many sensors to the HAT expansion board, you can connect it to the Raspberry Pi power interface or the HAT power interface through a USB cable. If you connect a lot of sensors to the HAT, there may be insufficient power when using the USB power supply.



3. After the Raspberry Pi is turned on (about 30-50s), you can access the Raspberry Pi with a browser.

- When no error message appears when the “ [WebServer.py](#) ” program is running, the dependent library is installed successfully.
- If the program does not run when the robot boots up, try connecting to the Raspberry Pi via SSH and then manually run [WebServer.py](#) and check for errors. Refer to the **Q&A** below or email us for help at support@adeept.com.

Before manually running “[WebServer.py](#)” , you need to end the program possibly auto run in the back end to release resources.

```
sudo killall python3
```

```
pi@raspberrypi:~ $ sudo killall python3
```

Use the following command to run “[WebServer.py](#)” .If it is Mecanum wheels, run [Adeept_4WD_Smart_Car_for_RPi/Server/Server_MecanumWheels/WebServer.py](#),if it is Ordinary wheels, run [Adeept_4WD_Smart_Car_for_RPi/Server/Server_OrdinaryWheels/WebServer.py](#).

Below is an example using Mecanum wheels.

```
sudo python3
```

```
Adeept_4WD_Smart_Car_for_RPi/Server/Server_MecanumWheels/WebServer.py
```

```
pi@raspberrypi:~$ sudo python3 Adeept_4WD_Smart_Car_for_RPi/Server/Server_MecanumWheels/WebServer.py
/usr/lib/python3/dist-packages/gpiozero/input_devices.py:852: PWMSoftwareFallback: For more accurate readings, use the
e pigpio pin factory. See https://gpiozero.readthedocs.io/en/stable/api_input.html#distancesensor-hc-sr04 for more info
o
  warnings.warn(PWMSoftwareFallback(
[0:08:45.640118169] [1995] INFO Camera camera_manager.cpp:326 libcamera v0.5.0+59-d83ff0a4
[0:08:45.671048323] [1996] WARN RPISdn sdn.cpp:40 Using legacy SDN tuning - please consider moving SDN inside rpi.de
noise
[0:08:45.673610358] [1996] INFO RPI vc4.cpp:447 Registered camera /base/soc/i2c0mux/i2c@1/ov5647@36 to Unicam device
/dev/media2 and ISP device /dev/media0
[0:08:45.673821506] [1996] INFO RPI pipeline_base.cpp:1121 Using configuration file '/usr/share/libcamera/pipeline/rp
pi/vc4/rpi_apps.yaml'
[0:08:45.681278128] [1995] INFO Camera camera.cpp:1205 configuring streams: (0) 640x480-RGB888 (1) 640x480-SGBRG10_C
SI2P
[0:08:45.681862720] [1996] INFO RPI vc4.cpp:622 Sensor: /base/soc/i2c0mux/i2c@1/ov5647@36 - Selected sensor format:
640x480-SGBRG10_1X10 - Selected unicam format: 640x480-pGAA
loop
loop
192.168.3.130
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead
.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.3.130:5000
Press CTRL+C to quit
waiting for connection...
```

If it is Ordinary wheel version, please execute the following command:

```
sudo python3
```

```
Adeept_4WD_Smart_Car_for_RPi/Server/Server_OrdinaryWheels/WebServer.py
```

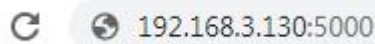
```
pi@raspberrypi:~$ sudo python3 Adeept_4WD_Smart_Car_for_RPi/Server/Server_OrdinaryWheels/WebServer.py
/usr/lib/python3/dist-packages/gpiozero/input_devices.py:852: PWMSoftwareFallback: For more accurate readings, use the
e https://gpiozero.readthedocs.io/en/stable/api_input.html#distancesensor-hc-sr04 for more info
  warnings.warn(PWMSoftwareFallback(
[0:08:31.790354941] [4468] INFO Camera camera_manager.cpp:313 libcamera v0.3.0+65-6ddd79b5
[0:08:31.827279911] [4469] WARN RPISdn sdn.cpp:40 Using legacy SDN tuning - please consider moving SDN inside rpi.den
[0:08:31.830384414] [4469] INFO RPI vc4.cpp:446 Registered camera /base/soc/i2c0mux/i2c@1/ov5647@36 to Unicam device
vice /dev/media3
[0:08:31.830491415] [4469] INFO RPI pipeline_base.cpp:1104 Using configuration file '/usr/share/libcamera/pipeline/rp
[0:08:31.839243878] [4468] INFO Camera camera.cpp:1183 configuring streams: (0) 640x480-RGB888 (1) 640x480-SGBRG10_CS
[0:08:31.839747718] [4469] INFO RPI vc4.cpp:621 Sensor: /base/soc/i2c0mux/i2c@1/ov5647@36 - Selected sensor format: 6
Selected unicam format: 640x480-pGAA
Matrix show error
loop
loop
192.168.3.130
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.3.130:5000
Press CTRL+C to quit
waiting for connection...
```

After running the command, more warning messages will appear, which is normal. **"waiting for connection..."** appears at the end of the running result, indicating that the program ran successfully.

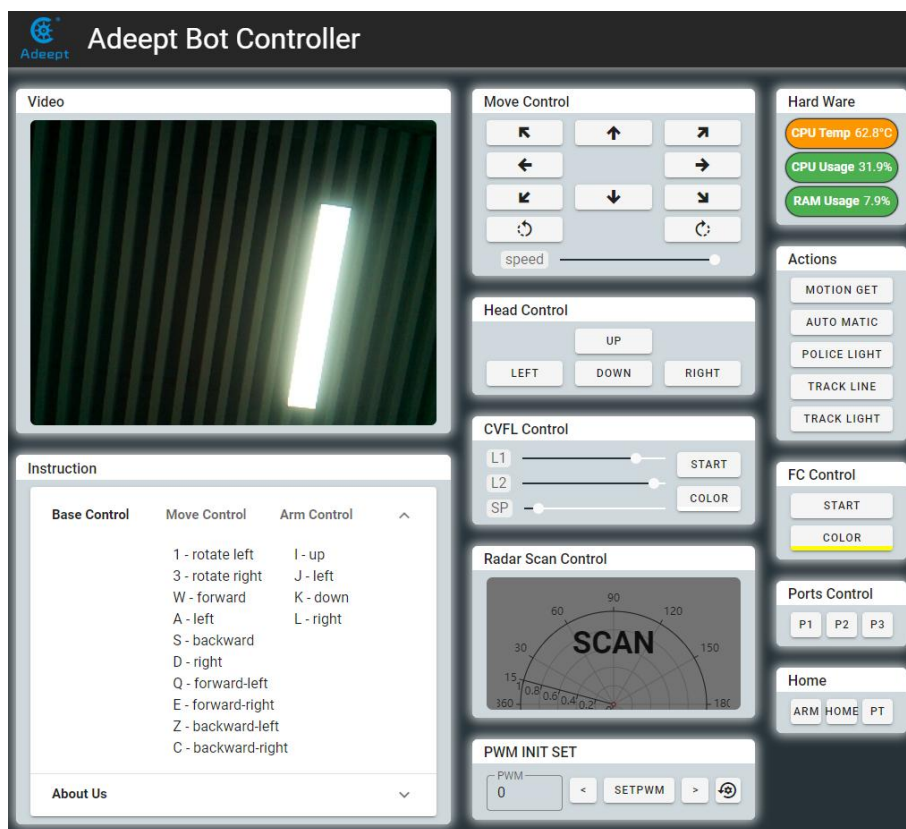
Open the web control interface.

1. Make sure your device is in the same local area network as the Raspberry Pi.

2. Open the browser on the device (chrome browser is recommended to avoid possible browser compatibility issues), enter the IP address of your Raspberry Pi in the address bar, and visit port 5000, for example: 192.168.3.130:5000



3. The web controller will then be loaded into the browser.



Depending on the product, the modules on the web controller are also different.

4. Click on the SSH command line interface and press "**Ctrl + C**" on the keyboard multiple times to stop the program from running.

3.9 Creating a WiFi Hotspot on Raspberry Pi(Less used)

The dependent library is successfully installed, and after "[WebServer.py](#)" runs automatically, our product will automatically turn on the hotspot when the wifi is not connected.

The method of turning on the WIFI hotspot in our robot product uses a project from GitHub [create_ap](#). Our installation script will automatically install this program and related dependent libraries. If you have not run our installation script, you can use the following command to install [create_ap](#):

```
sudo git clone https://github.com/oblique/create_ap.git
cd create_ap
sudo make install
```

1. Install related dependent libraries:

```
sudo apt-get install -y util-linux procps hostapd iproute2 iw haveged dnsmasq iptables iptables-persistent
```

2. Before turning on the hotspot, your Raspberry Pi cannot be connected to WiFi, and the WiFi module cannot be turned off, so when you test the hotspot function, you need to connect the necessary peripherals to the Raspberry Pi.

3.If your Raspberry Pi is connected to peripherals, and you want to test the Raspberry Pi ' s ability to turn on hotspot, you can click the WiFi icon in the upper right corner of the Raspberry Pi ' s desktop, click the name of the connected WiFi, click forget, and never turn Off WiFi, if it is already in the off state, you need to turn it on.

4.When the WiFi module of the Raspberry Pi is turned on and is not connected to any known network, you can enter the following command on the console to turn on the WiFi:

```
sudo killall python3
sudo create_ap wlan0 eth0 Adeept_Robot 12345678
```

"**Adeept_Robot** " is the name of the WIFI hotspot, " **12345678** " is the password of the WIFI hotspot.

Q&A

- Run "WebServer.py" error: **Could not start camera.**

The Raspberry Pi will automatically run the WebServer.py program. The WebServer.py program will occupy the camera. If you run the camera-related program again, the program will report an error. When you need to run the program manually, you need to stop the automatic running program.

1. Enter in the Raspberry Pi command line: (stop the automatic running program)

```
sudo killall python3
```

2. Enter the Raspberry Pi camera command, Under normal circumstances, the Raspberry Pi will take a photo named image.jpg, which can be viewed by entering "ls". If an error is reported, please check whether the camera is installed correctly until this command can run successfully. Or the camera may be damaged.

```
libcamera-jpeg -o image.jpg -n
```

Note: You need to power off the Raspberry Pi when plugging or unplugging the USB camera, otherwise the camera may be damaged.

- Run "WebServer.py" error: **remote I/O error**.

It might be due to the fact that the I2C (Inter-Integrated Circuit) interface is not enabled. Please enter the following command on the Raspberry Pi:

```
sudo raspi-config
```

Select **3 Interface options** -> **I5 I2C**, then enable the I2C interface.

- Run "WebServer.py" error: **ImportError: xxxxx failed to import**.

It may be caused by the unsuccessful installation of the dependent library. Please try to reinstall the dependent libraries.

If it is Mecanum wheels, enter in the Raspberry Pi command line:

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_MecanumWheels.py
```

If it is Ordinary wheels, enter in the Raspberry Pi command line:

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/setup_OrdinaryWheels.py
```

- Errors occur with “**permission denied**” prompt when I manually run “**WebServer.py**” .

You need to add “**sudo**” to the beginning of or “**WebServer.py**” to run the program. [PATH] is the product folder path.

If it is Mecanum wheels, enter in the Raspberry Pi command line:

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/Server/Server_MecanumWheels/WebServer.py
```

If it is Ordinary wheels, enter in the Raspberry Pi command line:

```
sudo python3 Adeept_4WD_Smart_Car_for_RPi/Server/Server_OrdinaryWheels/WebServer.py
```

- Raspberry Pi can't boot.

Remove all parts on the driver board. Only connect the board to the Raspberry Pi and the power supply, and then reboot the Raspberry Pi. Observe whether the green light of the Raspberry Pi is blinking. If the indicator does not flash or it flashes at the same interval, it means that the image file has not been burned successfully. Please re-burn the system.

- “**Remote side unexpectedly closed network connection**” shows on a popup window.

There can be error prompts during installation because the Raspberry Pi will auto reboot after the installation, which will disconnect the board.

- Raspberry Pi failed to connect a WiFi.

Please connect to the Raspberry Pi through the mouse, keyboard, monitor, and then click the WiFi icon in the upper right corner of the Raspberry Pi interface to enter the wifi information.

- I can not connect to the Raspberry Pi terminal via SSH

Please connect the mouse, keyboard, and monitor to the Raspberry Pi. Then, click the main menu in the upper-left corner of the Raspberry Pi interface. Next, select “**preferences**”, then select “**Raspberry Pi Configuration**”, and select “**Interfaces**”. Set the SSH bar to “**Enabled**”, and then click “**OK**”.

- SSH can't connect, error: **WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!**

Enter the following content in the CMD and press **Enter**

ssh-keygen -R <the Raspberry Pi's IP address>

For example:

```
ssh-keygen -R 192.168.3.157
```

Then you can SSH to the Raspberry Pi again

- [Can I power the Robot HAT and Raspberry Pi via USB?](#)

For the Raspberry Pi 3B, the output current required for its operation is 2A. While for the Raspberry Pi 4, at least 3A of output current is needed to run stably. As for the latest Raspberry Pi 5, given its performance improvement and potentially more complex computing requirements, the power supply requirements have also increased accordingly. It is recommended to use a power adapter that can provide at least 3.5A of output current. During the software installation and normal testing phases, using USB power supply is feasible. However, when it comes to connecting and debugging high - power modules such as servos and motors, USB power supply is not sufficient. Because these modules consume a large amount of current during operation. If USB power supply is used, it is very likely to cause a voltage drop, making the power supply of the Raspberry Pi unstable, thus affecting the normal operation of the entire system and even potentially damaging the device. Therefore, when operating high - power modules, it is strongly recommended to adopt a battery - powered solution. Battery power supply can not only provide stable and sufficient power, but also effectively reduce potential risks caused by voltage fluctuations, ensuring that the Raspberry Pi and external high - power modules can work stably and efficiently in coordination.