SunFounder picar-4wd

www.sunfounder.com

Sep 21, 2022

CONTENTS

CHAPTER

ONE

ABOUT PICAR-4WD KIT

PiCar-4wd is a cost-effectively developmental smart car kit of Raspberry Pi, suitable for the Raspberry Pi 3 model B/B+ and 4 Model B. With the grayscale module, ultrasonic module, Photo-interrupter sensor module, the car can realize obstacle avoidance, line following, object following, cliff detection, velocity measurement, mileage calculation and so on. Besides, the kits comes with operational examples of Web terminal, a rookie-friendly design. We also provide the original python code for in-depth learning.

If you want to learn another projects which we don't have, please feel free to send Email and we will update to our online tutorials as soon as possible, any suggestions are welcomed.

Here is the Email: cs@sunfounder.com.

1.1 Component List and Assembly Instructions

You need to check whether there are missing or damaged components according to the list first. If there are any problems, please contact us and we will solve them as soon as possible.

Please follow the steps on the PDF to assemble.

- `Component List and Assembly Instructions.`

1.2 Preparation

In this chapter, we firstly learn to start up Raspberry Pi. The content includes installing the OS, Raspberry Pi network and how to open terminal.

Note: You can check the complete tutorial on the official website of the Raspberry Pi: raspberry-pi-setting-up.

If your Raspberry Pi is set up, you can skip the part and go into the next chapter.

1.2.1 Installing the OS

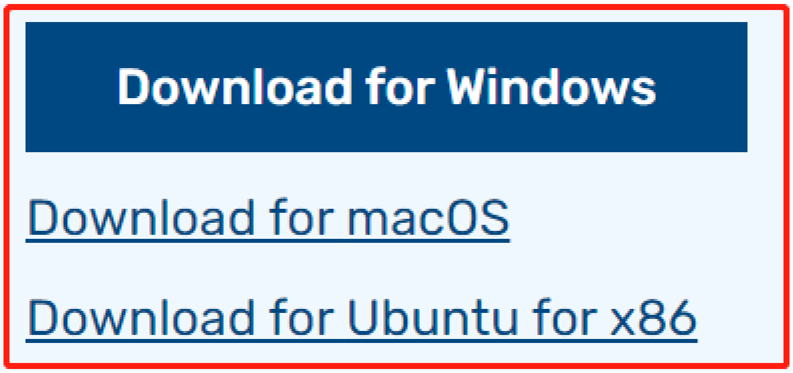
Required Components

Any Raspberry Pi	1 * Personal Computer
1 * Micro SD card	

Step 1

Raspberry Pi have developed a graphical SD card writing tool that works on Mac OS, Ubuntu 18.04 and Windows, and is the easiest option for most users as it will download the image and install it automatically to the SD card.

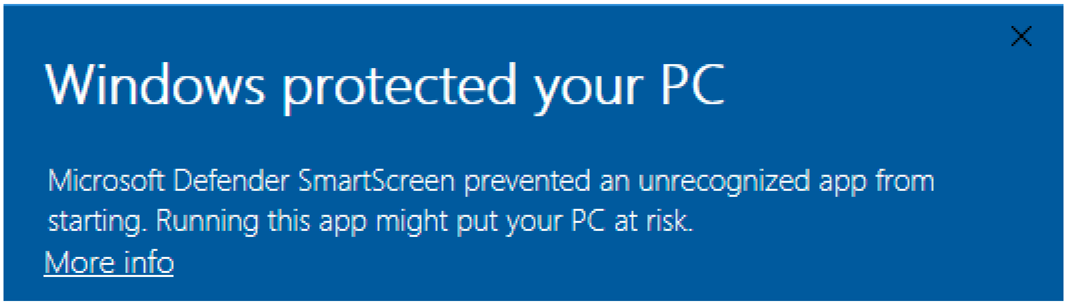
Visit the download page: https://www.raspberrypi.org/software/. Click on the link for the Raspberry Pi Imager that matches your operating system, when the download finishes, click it to launch the installer.



Step 2

When you launch the installer, your operating system may try to block you from running it. For example, on Windows I receive the following message:

If this pops up, click on **More info** and then **Run anyway**, then follow the instructions to install the Raspberry Pi Imager.



Windows protected your PC

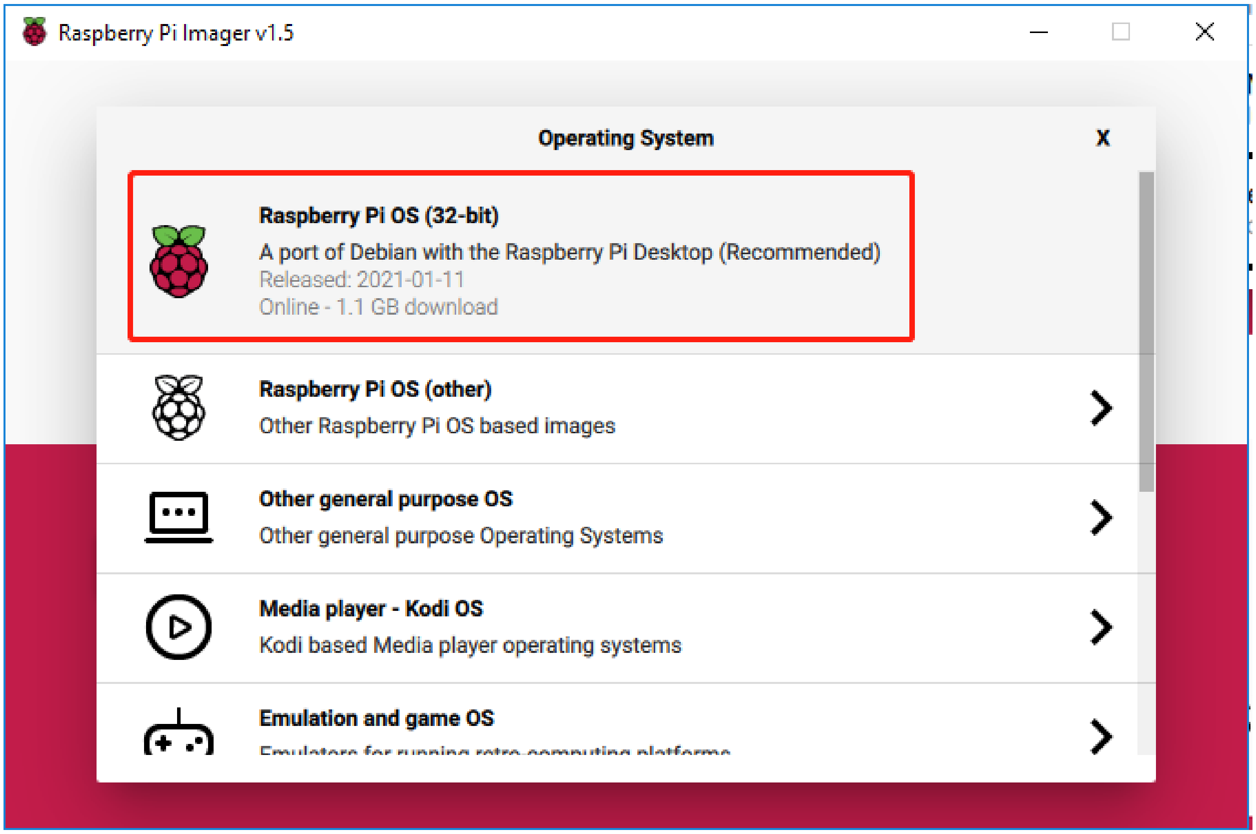
Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

Step 3

Insert your SD card into the computer or laptop SD card slot.

Step 4

In the Raspberry Pi Imager, select the OS that you want to install and the SD card you would like to install it on.



Raspberry Pi OS (32-bit)
A port of Debian with the Raspberry Pi Desktop (Recommended)
Released: 2021-01-11
Online - 1.1 GB download

Raspberry Pi OS (other)
Other Raspberry Pi OS based images

Other general purpose OS
Other general purpose Operating Systems

Media player - Kodi OS
Kodi based Media player operating systems

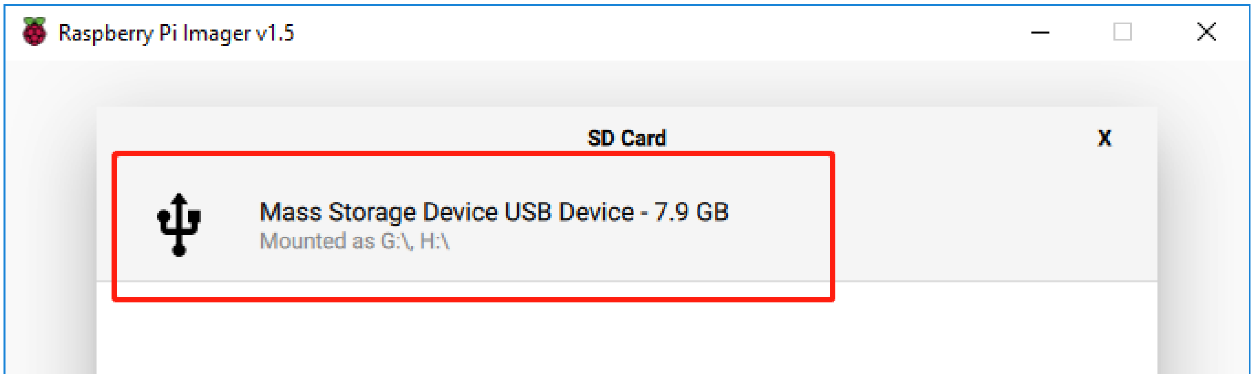
Emulation and game OS
Emulators for running retro-computing platforms

Note:

1) You will need to be connected to the internet the first time.
2) That OS will then be stored for future offline use(lastdownload.cache, C:/Users/yourname/AppData/Local/Raspberry Pi/Imager/cache). So the next time you open the software, it will have the display "Released: date, cached on your computer".

Step 5

Select the SD card you are using.



Mass Storage Device USB Device - 7.9 GB
Mounted as G:\, H:\

Step 6

Press **Ctrl+Shift+X** to open the **Advanced options** page to enable SSH and configure wifi, these 2 items must be set, the others depend on your choice . You can choose to always use this image customization options.

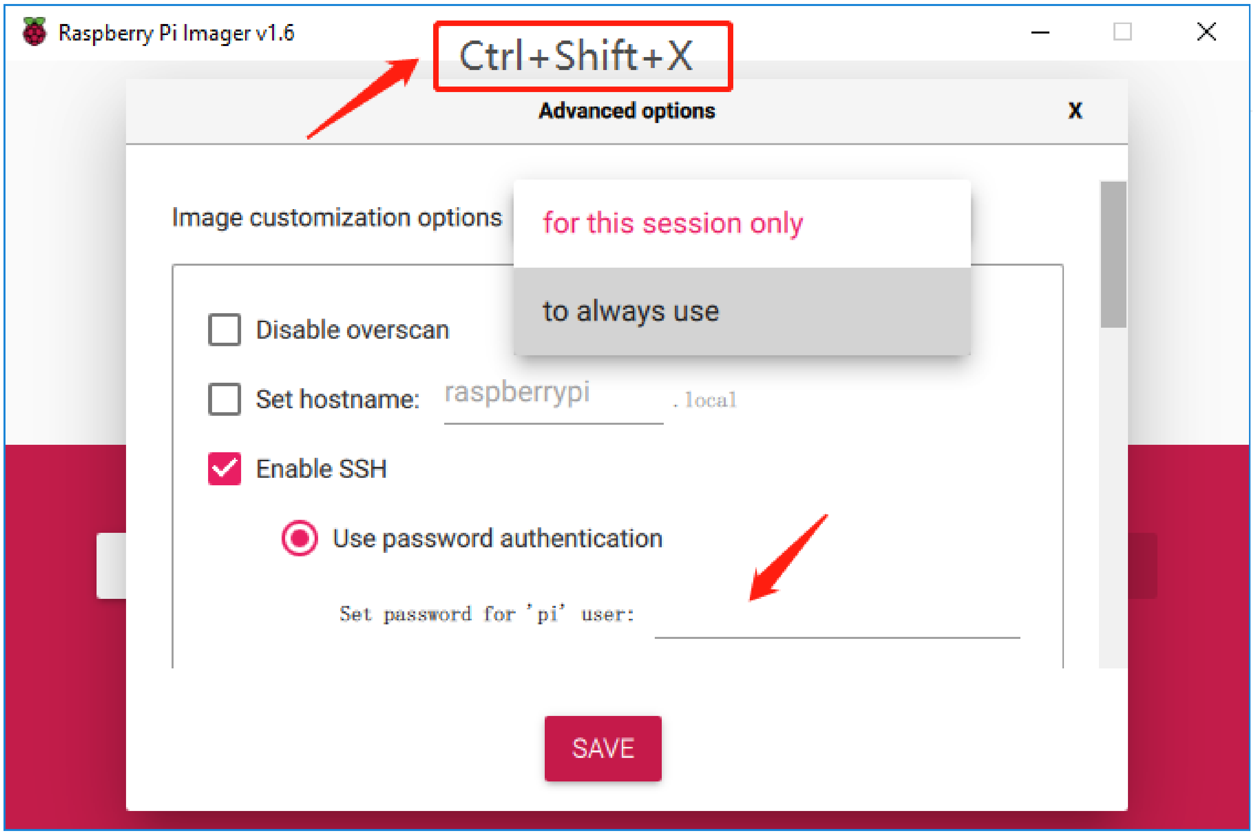


Image customization options

- for this session only
- to always use

□ Disable overscan

□ Set hostname: raspberrypi .local

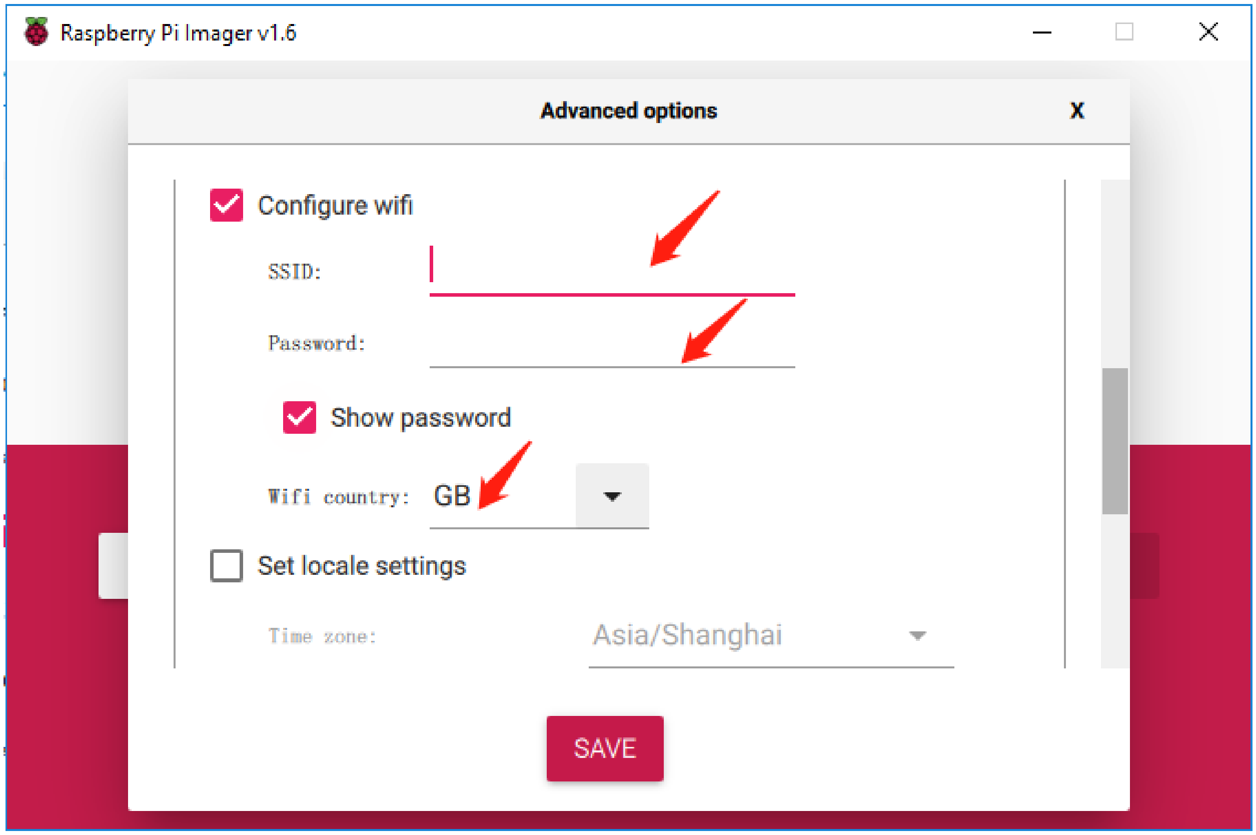
☑ Enable SSH

● Use password authentication

Set password for 'pi' user:

Then scroll down to complete the wifi configuration and click **SAVE**.

Note: **wifi country** should be set the two-letter ISO/IEC alpha2 code for the country in which you are using your Raspberry Pi, please refer to the following link: https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2#Officially_assigned_code_elements.



Raspberry Pi Imager v1.6

Advanced options

☑ Configure wifi

SSID:

Password:

☑ Show password

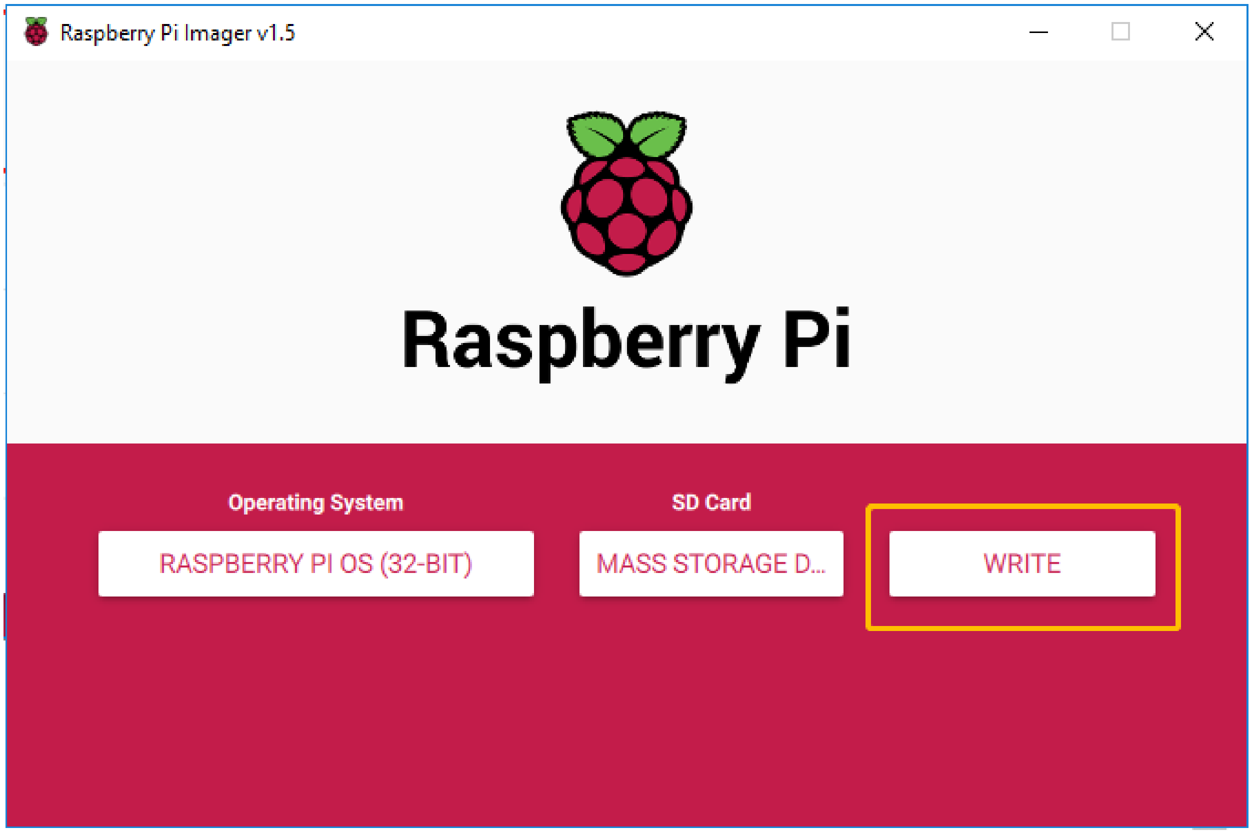
Wifi country: GB

□ Set locale settings

Time zone: Asia/Shanghai

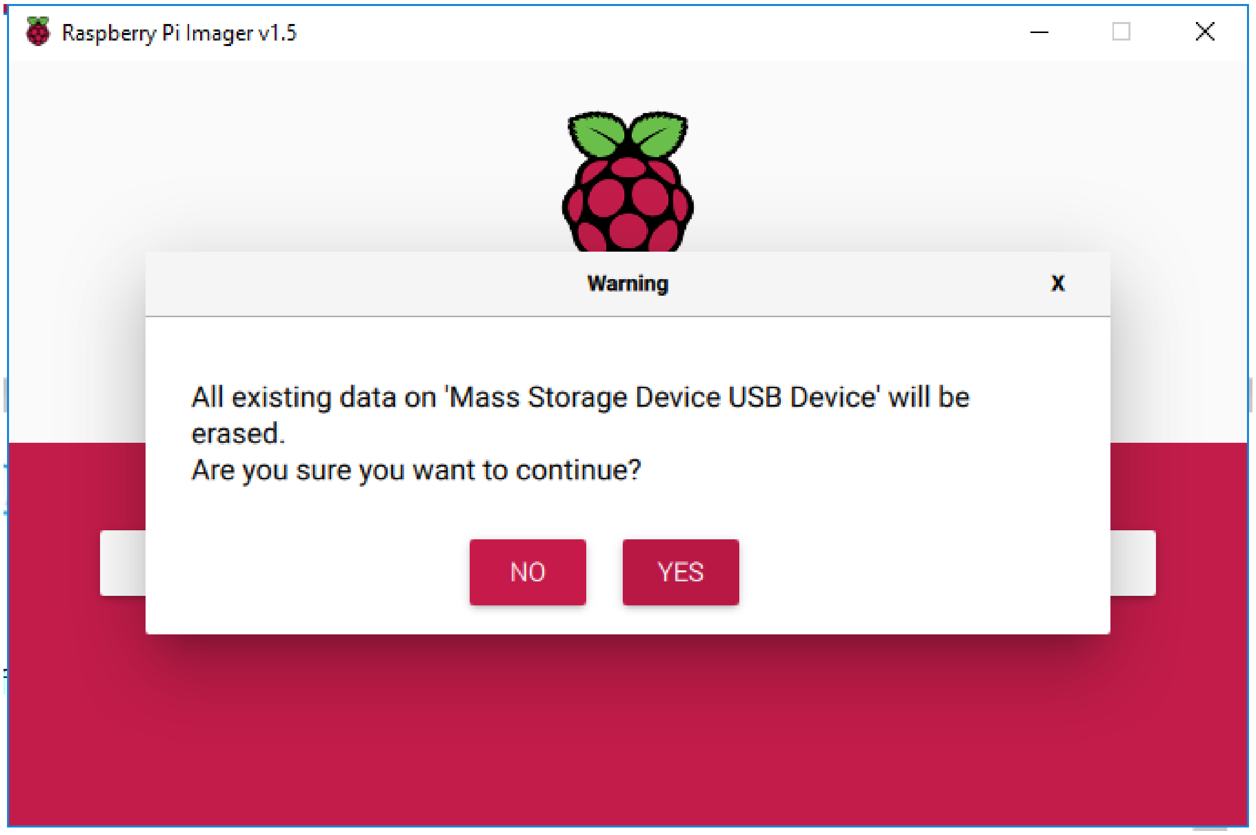
Step 7

Click the **WRITE** button.



Step 8

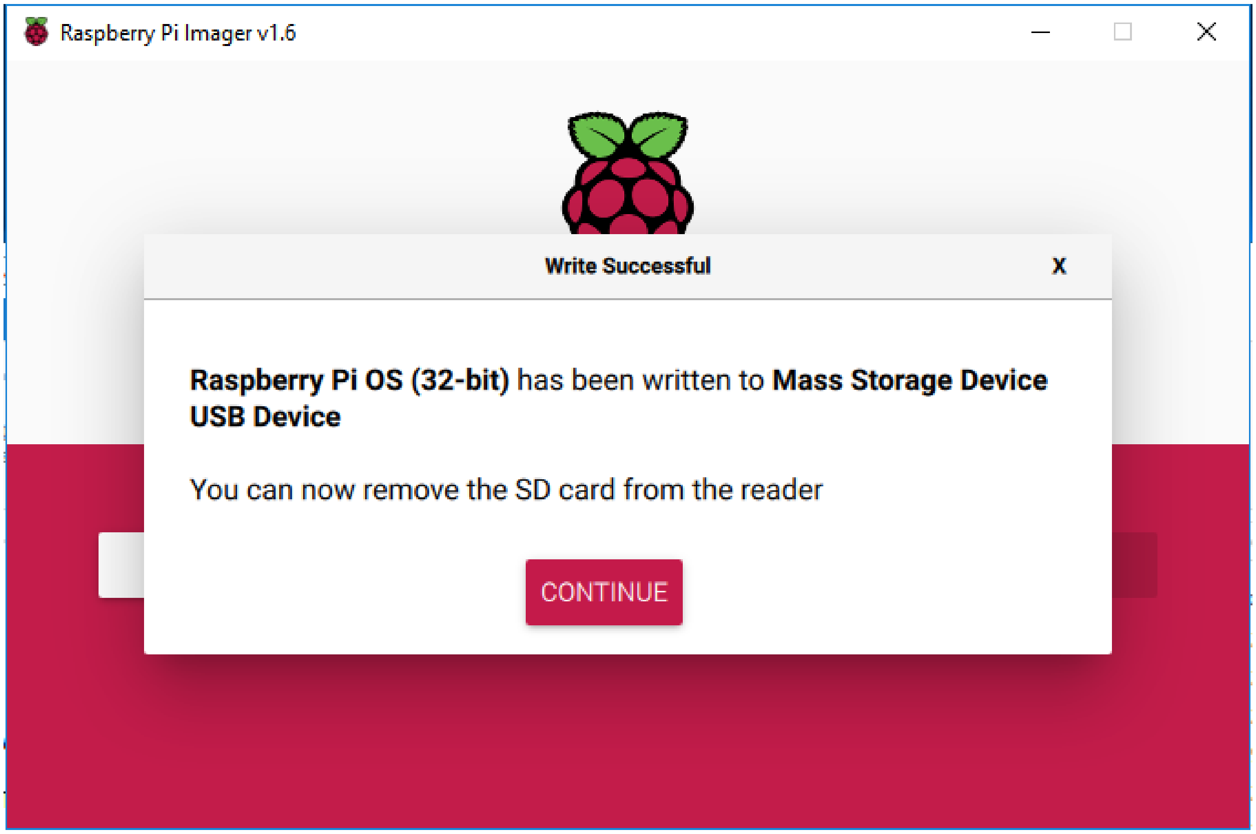
If your SD card currently has any files on it, you may wish to back up these files first to prevent you from permanently losing them. If there is no file to be backed up, click **Yes**.



All existing data on 'Mass Storage Device USB Device' will be erased.
Are you sure you want to continue?

Step 9

After waiting for a period of time, the following window will appear to represent the completion of writing.



Write Successful

Raspberry Pi OS (32-bit) has been written to **Mass Storage Device USB Device**

You can now remove the SD card from the reader

1.2.2 Set up Your Raspberry Pi

If You Have a Screen

If you have a screen, it will be easy for you to operate on the Raspberry Pi.

Required Components

Any Raspberry Pi	1 * Power Adapter
1 * Micro SD card	1 * Screen Power Adapter
1 * HDMI cable	1 * Screen
1 * Mouse	1 * Keyboard

1) Insert the SD card you've set up with Raspberry Pi OS into the micro SD card slot on the underside of your Raspberry Pi.

2) Plug in the Mouse and Keyboard.

3) Connect the screen to Raspberry Pi's HDMI port and make sure your screen is plugged into a wall socket and switched on.

Note: If you use a Raspberry Pi 4, you need to connect the screen to the HDMI0 (nearest the power in port).

4) Use the power adapter to power the Raspberry Pi. After a few seconds, the Raspberry Pi OS desktop will be displayed.

If You Have No Screen

If you don't have a display, you can log in to the Raspberry Pi remotely, but before that, you need to get the IP of the Raspberry Pi.

Get the IP Address

After the Raspberry Pi is connected to WIFI, we need to get the IP address of it. There are many ways to know the IP address, and two of them are listed as follows.

1. Checking via the router

If you have permission to log in the router(such as a home network), you can check the addresses assigned to Raspberry Pi on the admin interface of router.

The default hostname of the Raspberry Pi OS is **raspberrypi**, and you need to find it. (If you are using ArchLinuxARM system, please find alarmpi.)

2. Network Segment Scanning

You can also use network scanning to look up the IP address of Raspberry Pi. You can apply the software, **Advanced IP scanner** and so on.

Scan the IP range set, and the name of all connected devices will be displayed. Similarly, the default hostname of the Raspberry Pi OS is **raspberrypi**, if you haven't modified it.

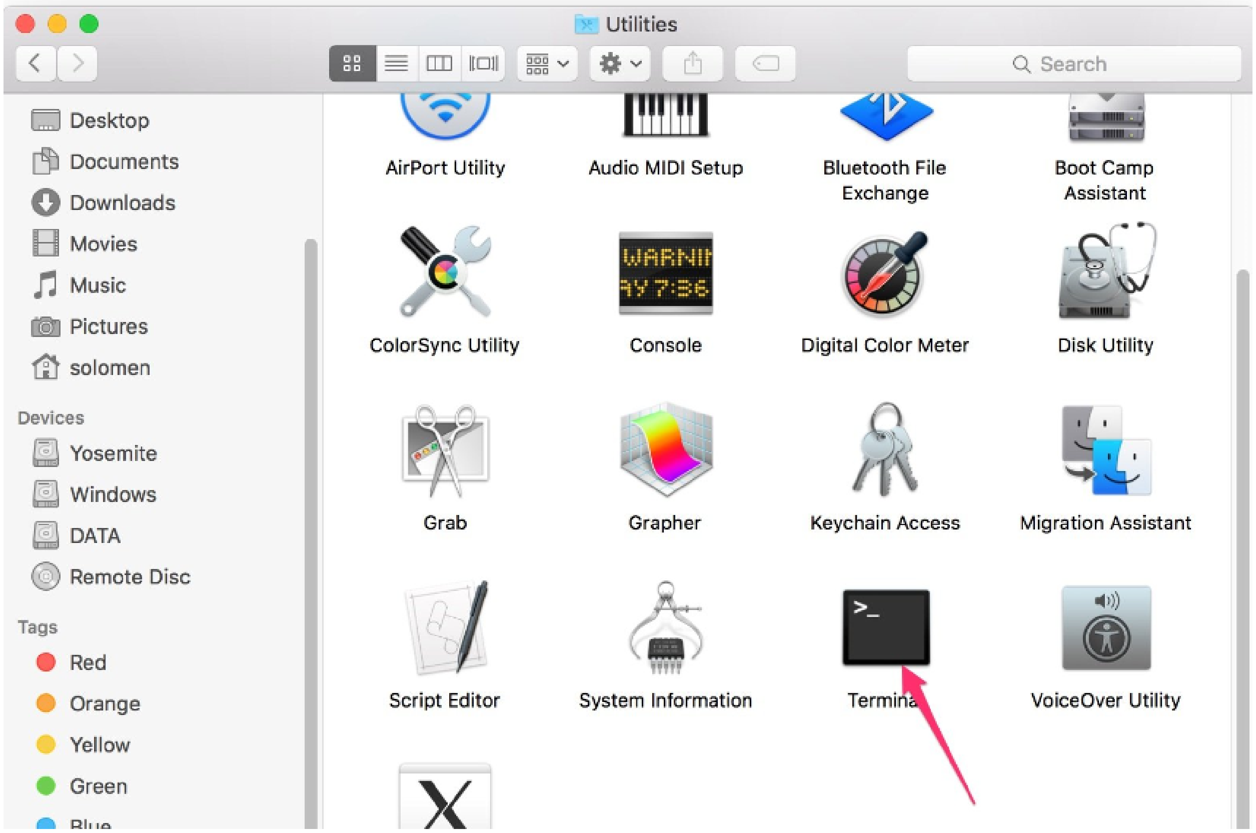
Use the SSH Remote Control

We can open the Bash Shell of Raspberry Pi by applying SSH. Bash is the standard default shell of Linux. The Shell itself is a program written in C that is the bridge linking the customers and Unix/Linux. Moreover, it can help to complete most of the work needed.

For Linux or/Mac OS X Users

Step 1

Go to **Applications->Utilities**, find the **Terminal**, and open it.



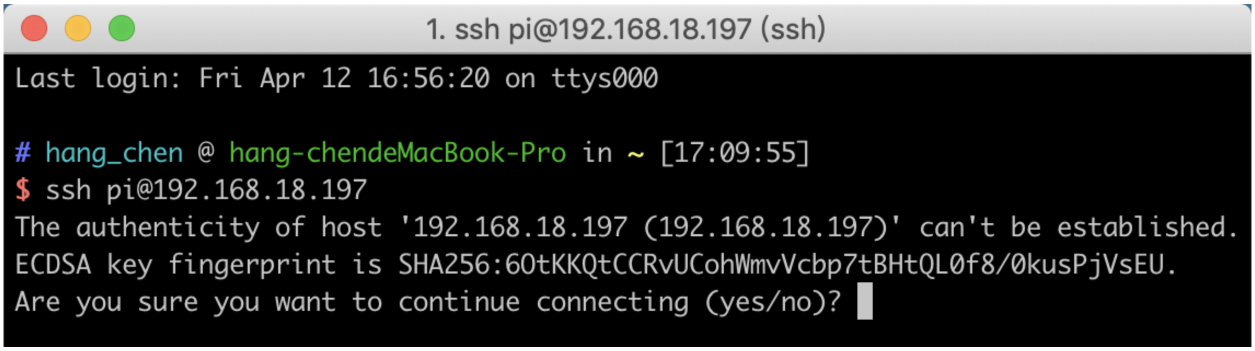
Step 2

Type in **ssh pi@ip_address** . “pi” is your username and “ip_address” is your IP address. For example:

```
ssh pi@192.168.18.197
```

Step 3

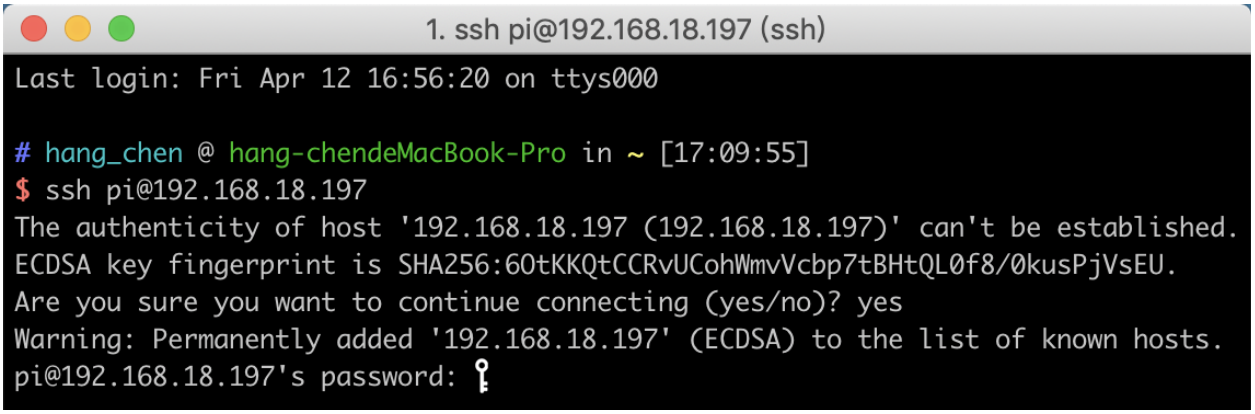
Input “yes”.



```
Last login: Fri Apr 12 16:56:20 on ttys000

# hang_chen @ hang-chendeMacBook-Pro in ~ [17:09:55]
$ ssh pi@192.168.18.197
The authenticity of host '192.168.18.197 (192.168.18.197)' can't be established.
ECDSA key fingerprint is SHA256:6OtKKQtCCRvUCohWmvVcbp7tBHtQL0f8/0kusPjVsEU.
Are you sure you want to continue connecting (yes/no)?
```

Step 4

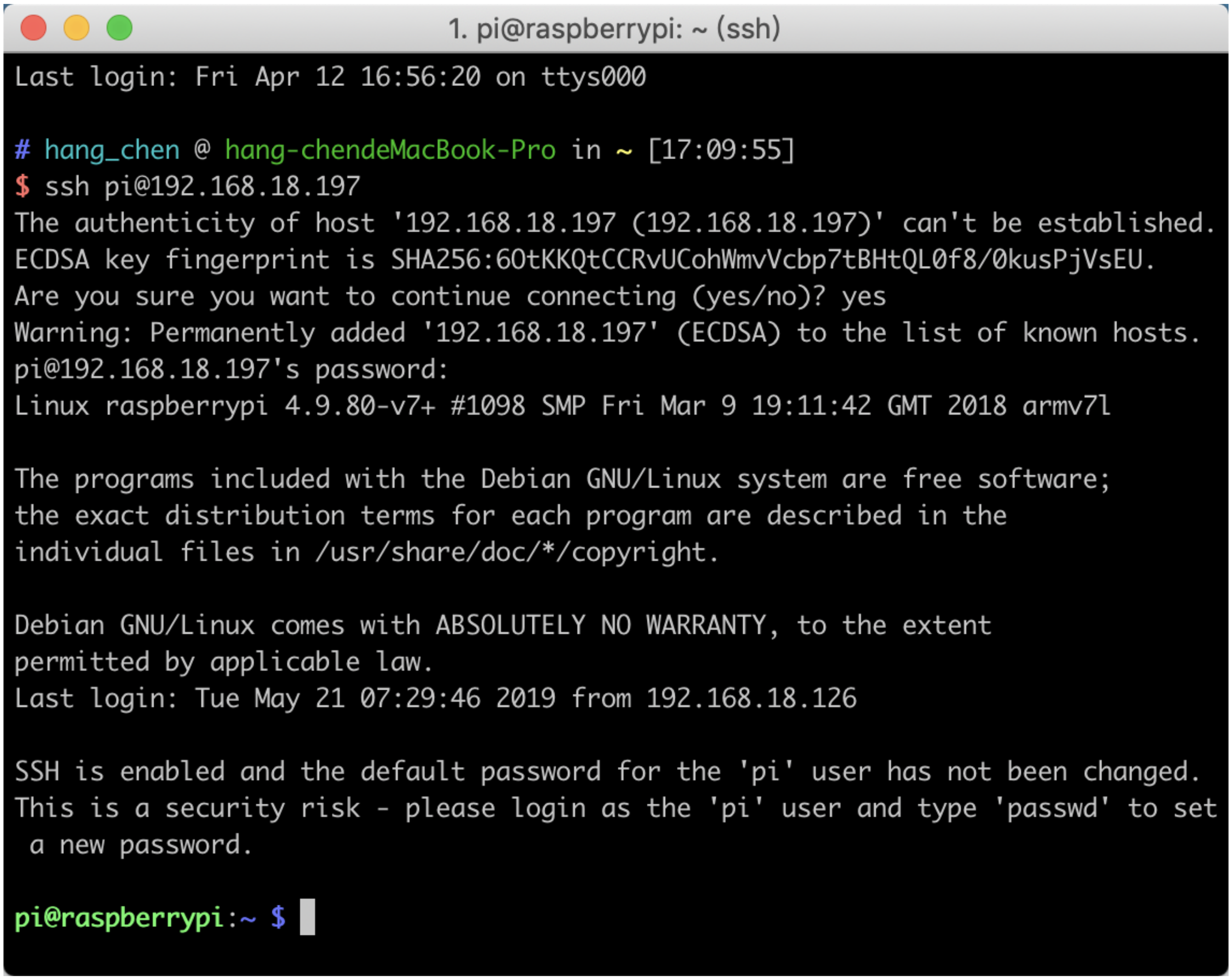
Input the passcode and the default password is **raspberry**.



```
Last login: Fri Apr 12 16:56:20 on ttys000

# hang_chen @ hang-chendeMacBook-Pro in ~ [17:09:55]
$ ssh pi@192.168.18.197
The authenticity of host '192.168.18.197 (192.168.18.197)' can't be established.
ECDSA key fingerprint is SHA256:6OtKKQtCCRvUCohWmvVcbp7tBHtQL0f8/0kusPjVsEU.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.18.197' (ECDSA) to the list of known hosts.
pi@192.168.18.197's password:
```

Step 5

We now get the Raspberry Pi connected and are ready to go to the next step.



```
Last login: Fri Apr 12 16:56:20 on ttys000

# hang_chen @ hang-chendeMacBook-Pro in ~ [17:09:55]
$ ssh pi@192.168.18.197
The authenticity of host '192.168.18.197 (192.168.18.197)' can't be established.
ECDSA key fingerprint is SHA256:6OtKKQtCCRvUCohWmvVcbp7tBHtQL0f8/0kusPjVsEU.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.18.197' (ECDSA) to the list of known hosts.
pi@192.168.18.197's password:
Linux raspberrypi 4.9.80-v7+ #1098 SMP Fri Mar 9 19:11:42 GMT 2018 armv7l

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 May 21 07:29:46 2019 from 192.168.18.126

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:~ $
```

Note: When you input the password, the characters do not display on window accordingly, which is normal. What you need is to input the correct password.

For Windows Users

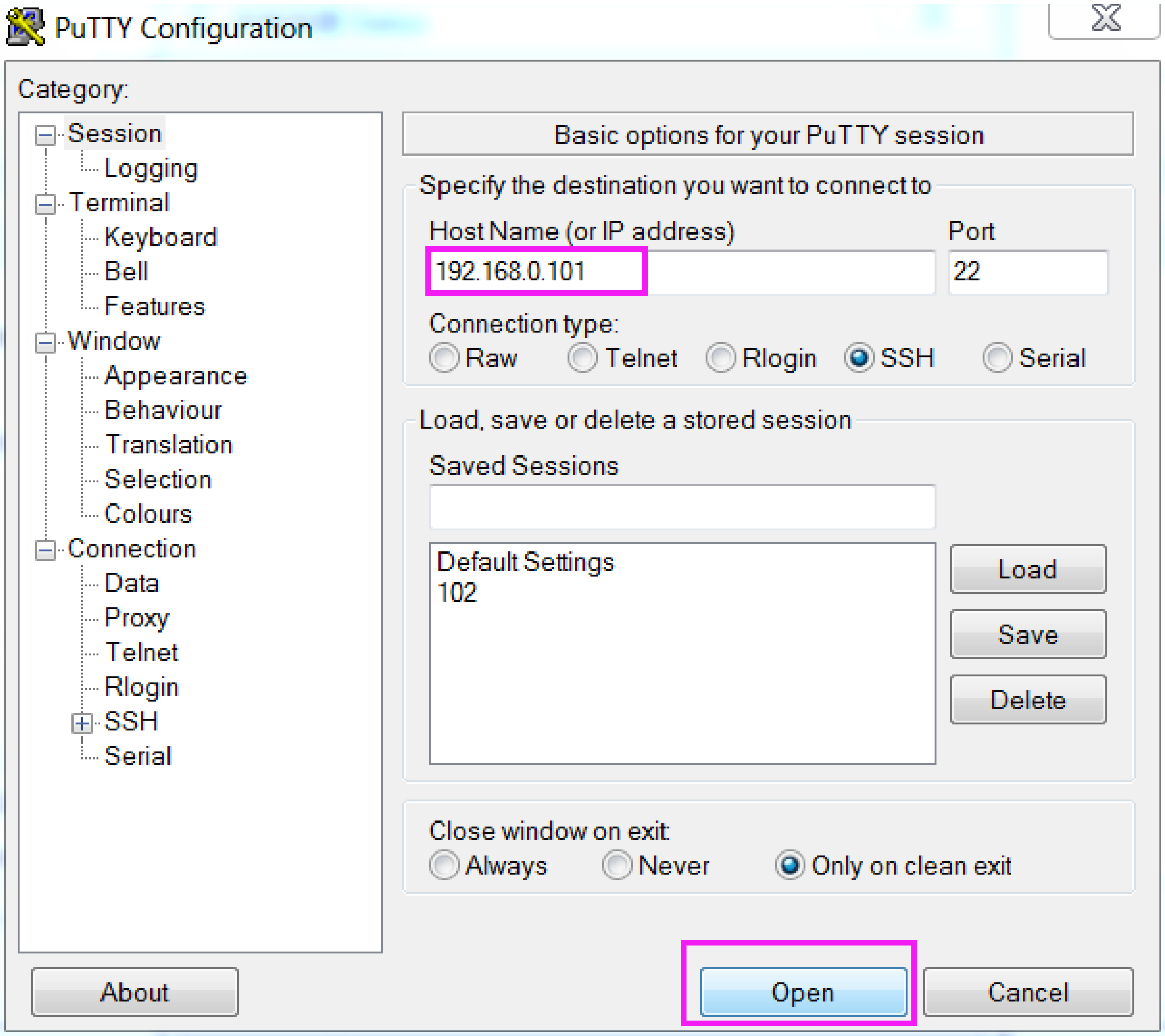
If you're a Windows user, you can use SSH with the application of some software. Here, we recommend **PuTTY**.

Step 1

Download PuTTY.

Step 2

Open PuTTY and click **Session** on the left tree-alike structure. Enter the IP address of the RPi in the text box under **Host Name (or IP address)** and **22** under **Port** (by default it is 22).



PuTTY Configuration

Category:

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 192.168.0.101

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions

Default Settings
102

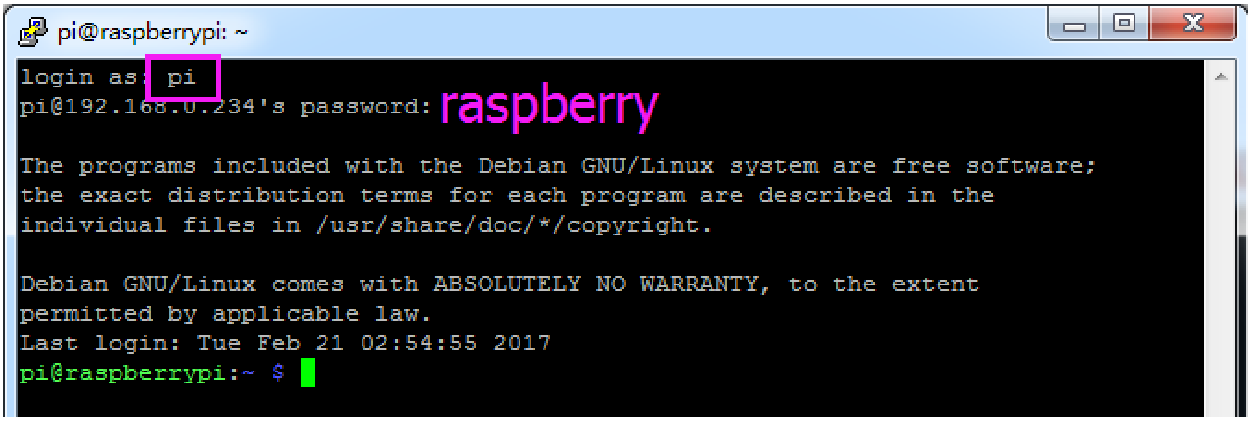
Close window on exit: ○ Always ○ Never ● Only on clean exit

Step 3

Click **Open**. Note that when you first log in to the Raspberry Pi with the IP address, there prompts a security reminder. Just click **Yes**.

Step 4

When the PuTTY window prompts "**login as:**", type in "**pi**" (the user name of the RPi), and **password**: "raspberry" (the default one, if you haven't changed it).



```
login as: pi
pi@192.168.0.234's password: raspberry

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 Feb 21 02:54:55 2017
pi@raspberrypi:~ $
```

Step 5

Here, we get the Raspberry Pi connected and it is time to conduct the next steps.

Note: When you input the password, the characters do not display on window accordingly, which is normal. What you need is to input the correct password.

If you are not satisfied with using the command window to control the Raspberry Pi, you can also use the remote desktop function, which can help us manage the files in the Raspberry Pi easily.

For details on how to do this, please refer to *Remote Desktop*.

1.3 Download the Code

The codes in the disk don't need downloading, only for your fast study. If you need the latest codes, please go to github to download as following methods.

Change directory to /home/pi.

```
cd /home/pi/
```

Note: cd, short for change directory is to change from the current path to the intended directory. Informally, here is to go to the path /home/pi/.

Clone the repository from github.

```
git clone https://github.com/sunfounder/picar-4wd
```

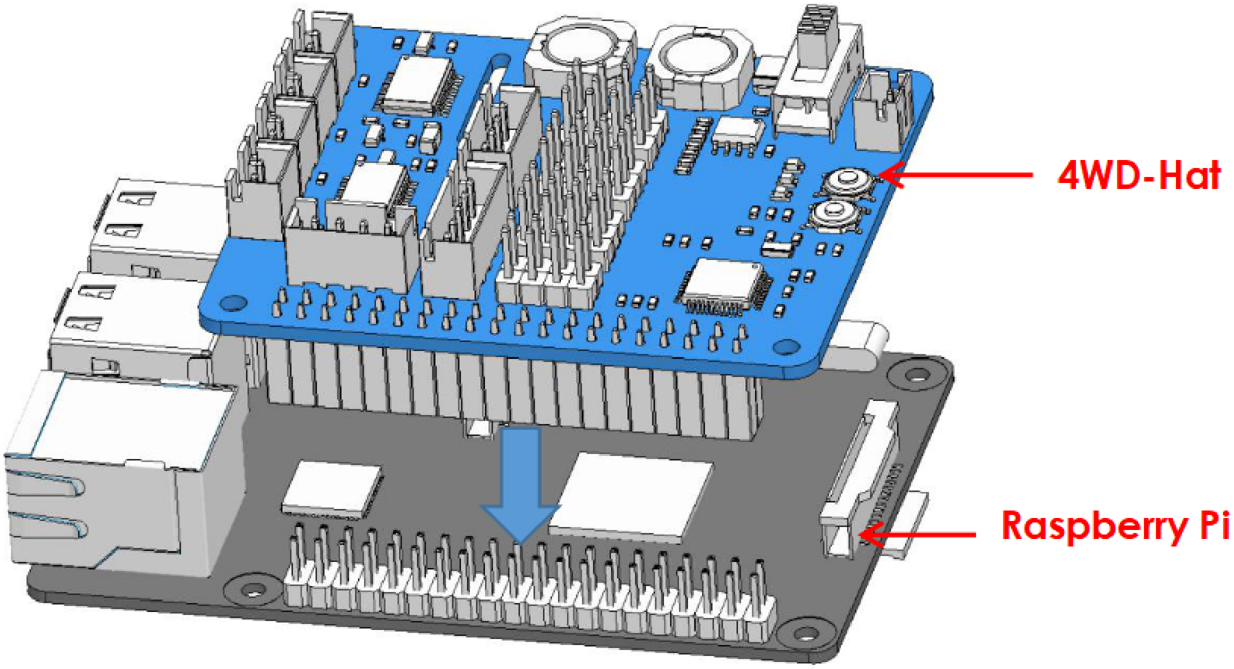
1.4 Test the Modules

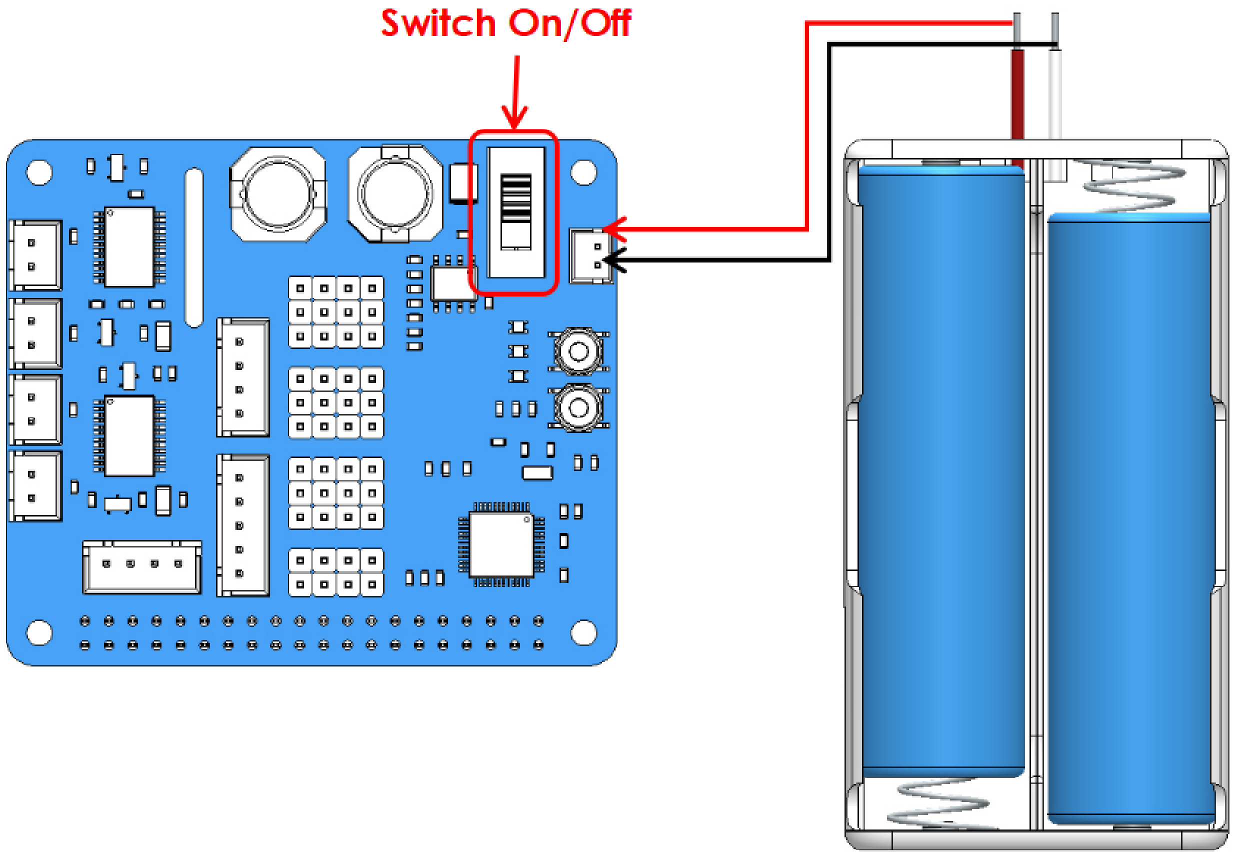
In case of anything wrong with the components, you can firstly debug them, then start your assembling and use.

1.4.1 Initialize the Environment

Initialize the environment before using the product example. The methods are as follows.

 Assemble 4WD-Hat and boot the Raspberry Pi.





Note: Due to the long test time, you can also use an external power supply to power the Raspberry Pi at the same time. But 4WD-Hat can only be powered by battery, you can't just use external power supply without batterries, otherwise the following error will occur when you are running the command: **picar-4wd test motor**.

—- picar-4wd test motor returns OSError: [Errno 121] Remote I/O error —-

Enter the folder picar-4wd.

```
cd /home/pi/picar-4wd/
```

Start up the initialization function.

```
sudo python3 setup.py install
```

The main functions of setup.py may be:

A. Download and configure necessary library files, including pip, sysstat, i2c-tools.

B. Provide entry points calledpicar-4wdto call the function of example code.

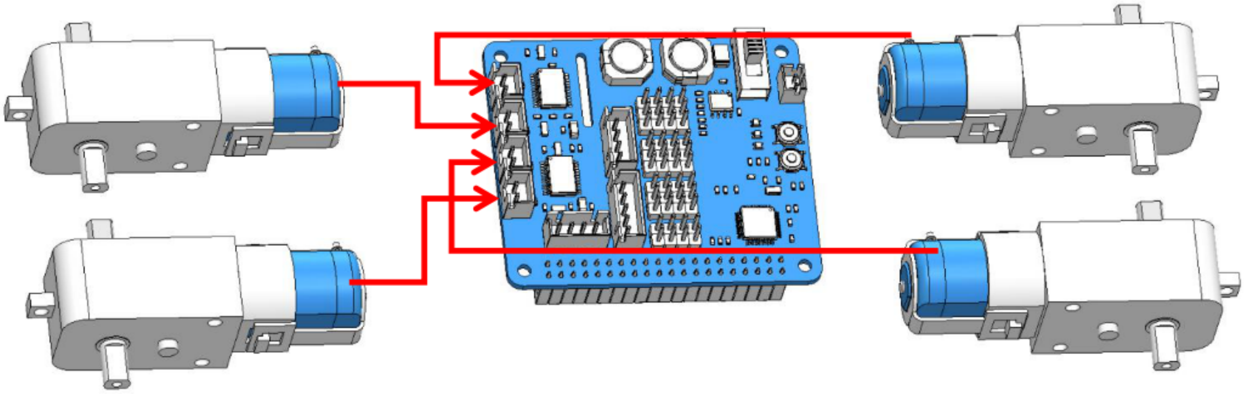
Note: Initialization needs connecting to the internet and spending a lot of time. The final line will appearfinishafter configuration is finished.

1.4.2 Module Testing

To make sure that the module works well, you can test the module by these means.

Testing the motor of the car.

Wire up the motor according to the diagram.



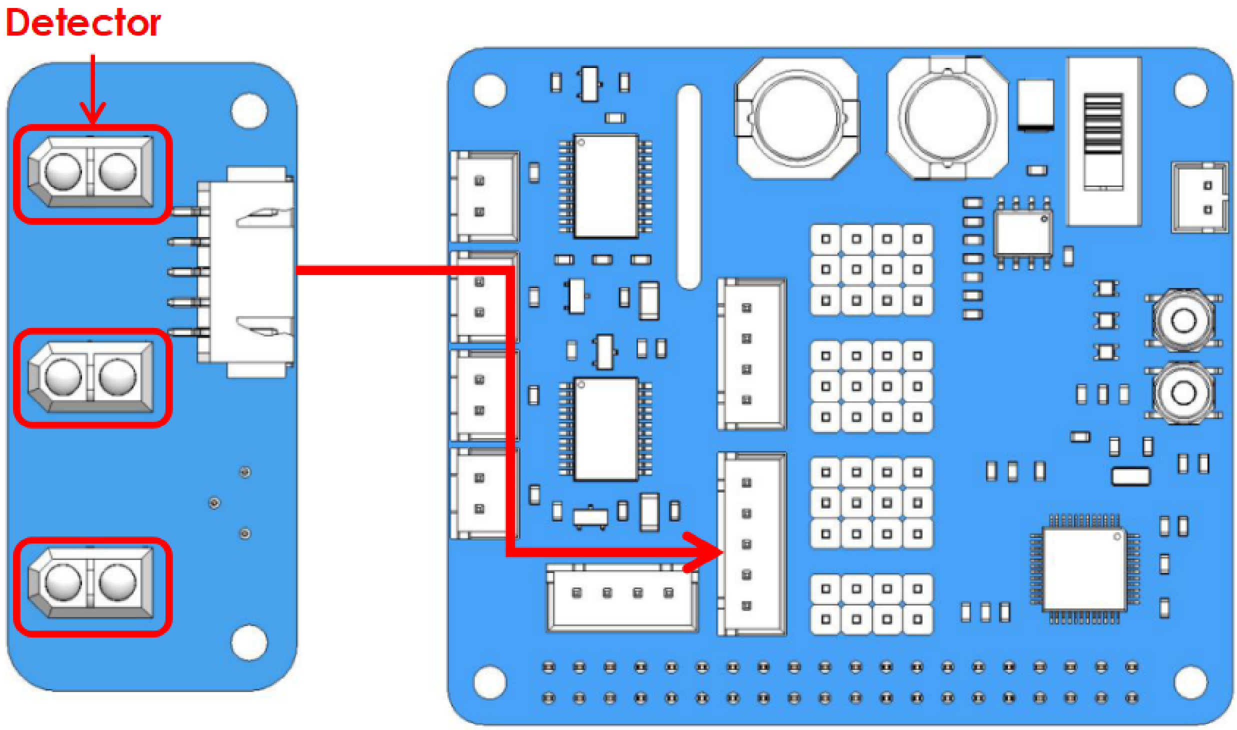
Input the command.

```
picar-4wd test motor
```

After you input the command, the four motors will last working till you press on Ctrl+C. Among the motors, two of them rotate clockwise, and the rest of them rotate anticlockwise.

Testing the grayscale sensor module.

Finishing the wire up according to the diagram.



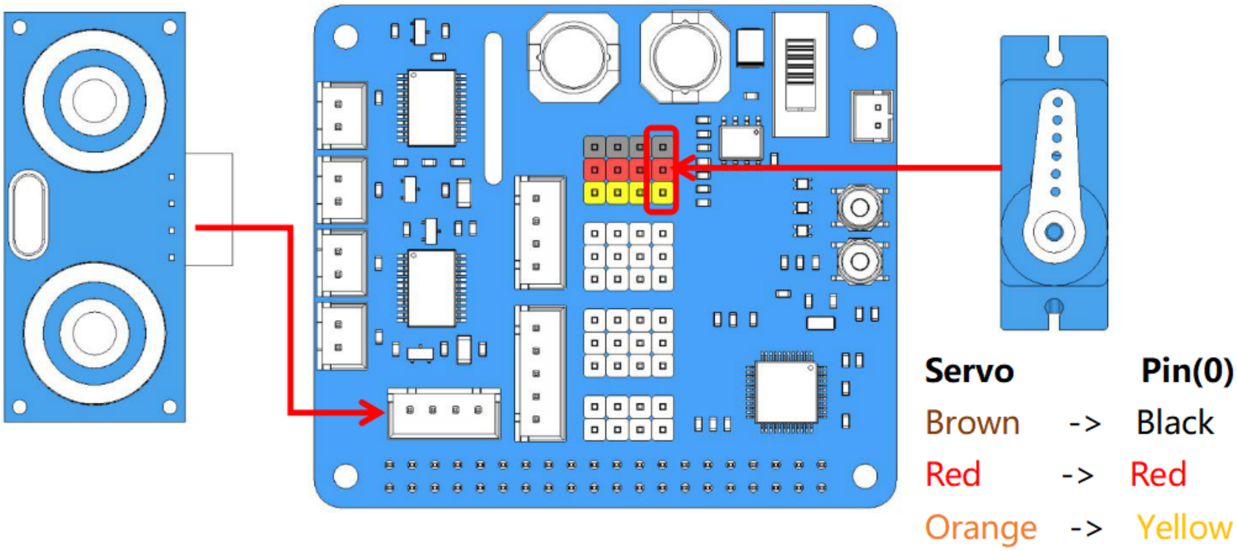
Input the following command.

```
picar-4wd test grayscale
```

After you input the command, the monitor will print the readings of three detectors of line-following module once.

Testing the servo and the ultrasonic module.

Start your wire up:



Input the commands:

```
picar-4wd test servo
```

The arm of servo will rotate to the angle of 0, meanwhile the monitor will print the readings of ultrasonic module once.

Note: You need to boot this function to check the direction of servo's output axis when you assemble the car and the servo.

1.4.3 Power Supply Detection

You can use the following command to get the supply voltage.

```
picar-4wd power-read
```

The supply voltage ought to be 6V~8.5V to ensure that the car works well.

1.4.4 Reset Controller

If you encounter some errors during your use, for example, the car don’t make response to your control command of control terminal, you can try to reset HATS by using the following command.

```
picar-4wd soft-reset
```

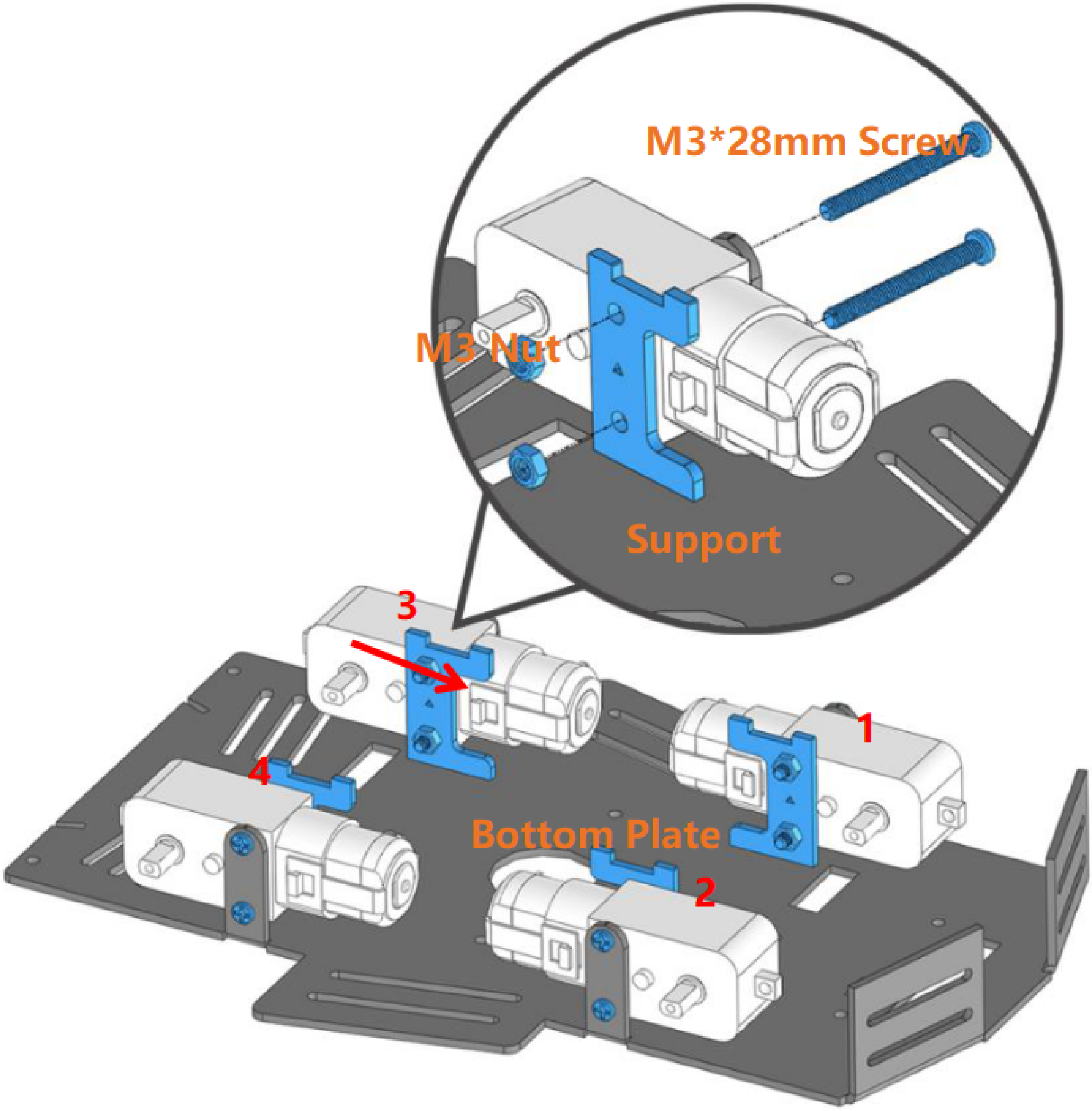
1.5 Hardware Assembling

1.5.1 Assembling Motor

Assemble 4 motor to the bottom plate.

Note: Keep the vertex angle of the triangle sticking upward.

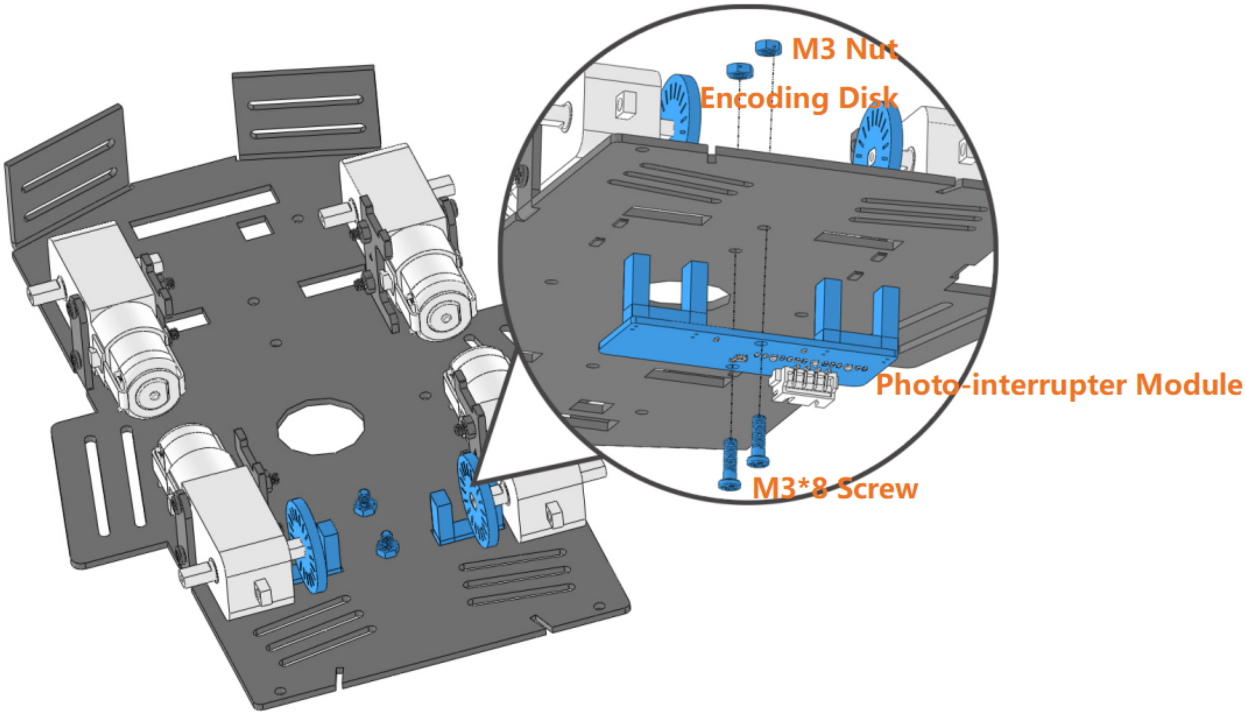
For the convenience of later assembly, please mark the number(1, 2, 3, 4) on the socket of the 4 motors.



1.5.2 Photo-interrupter Module + Encoding Disk

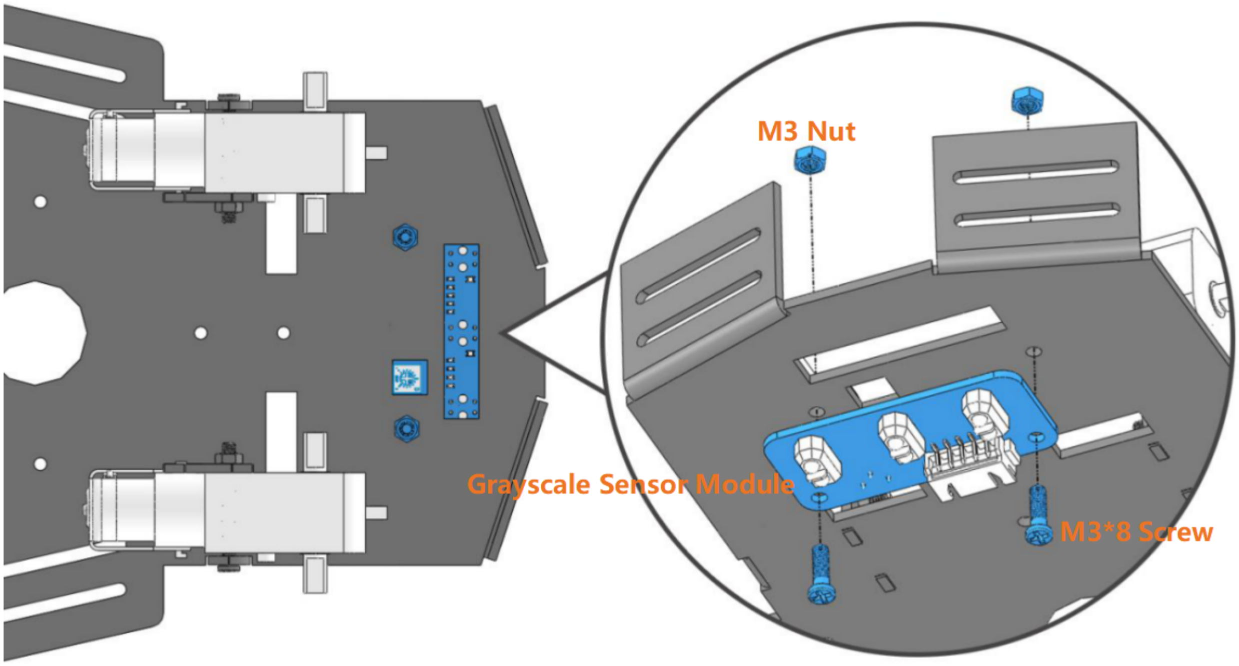
Insert the Encoding Disk into the motor, then assemble the Photo-interrupter module from bottom to top.

Note: Keep the Encoding Disk on the motor away from the Photo-interrupter module to avoid the potential abrasion.



1.5.3 Assembling Grayscale Sensor Module

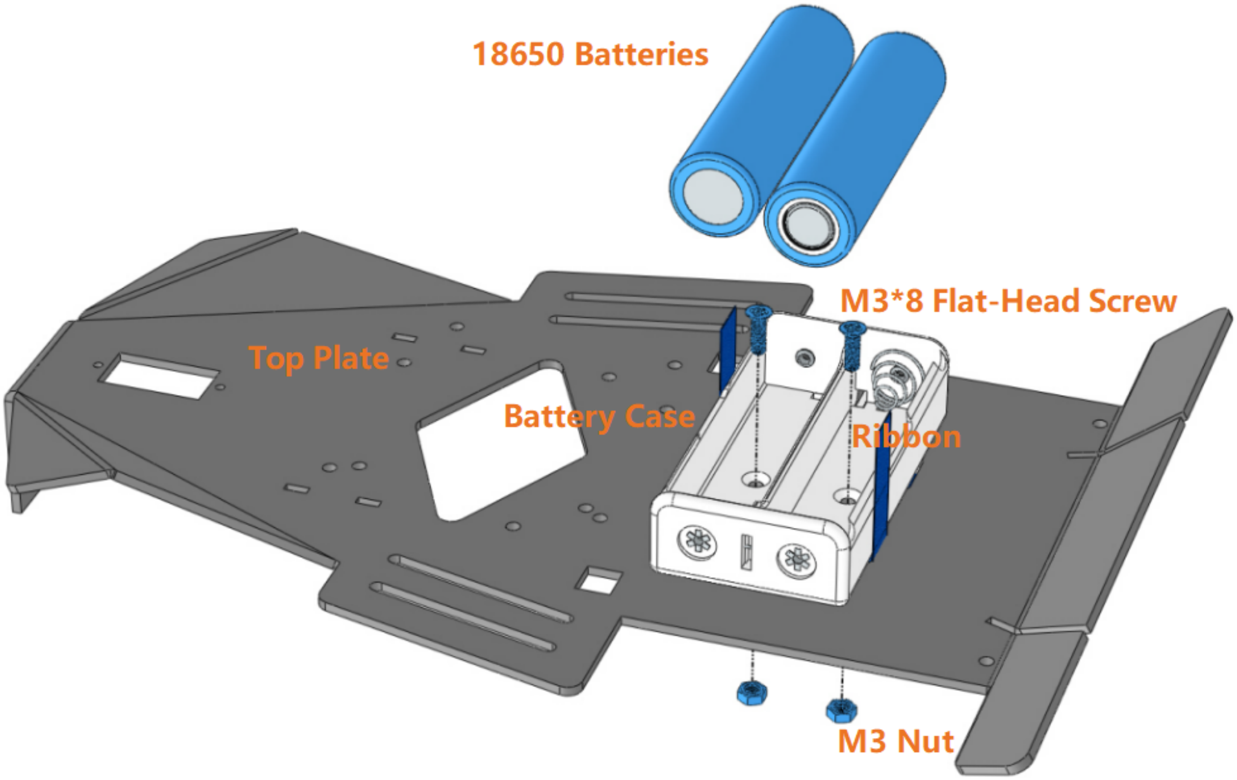
Have the Grayscale sensor module be under the head of the car.



1.5.4 Assembling Battery Holder

Assemble the battery holder to the top plate and put the battery in.

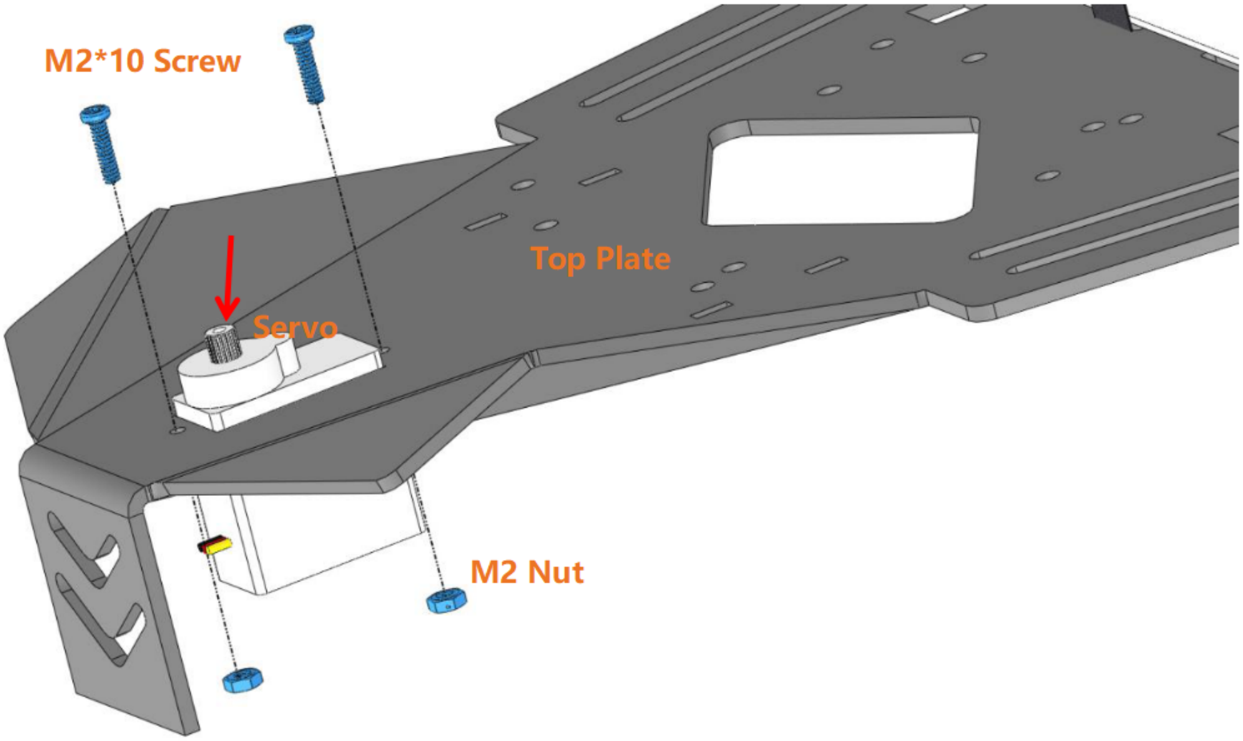
Note: To get the battery out of the holder easily, you can wind the battery with ribbon like this.



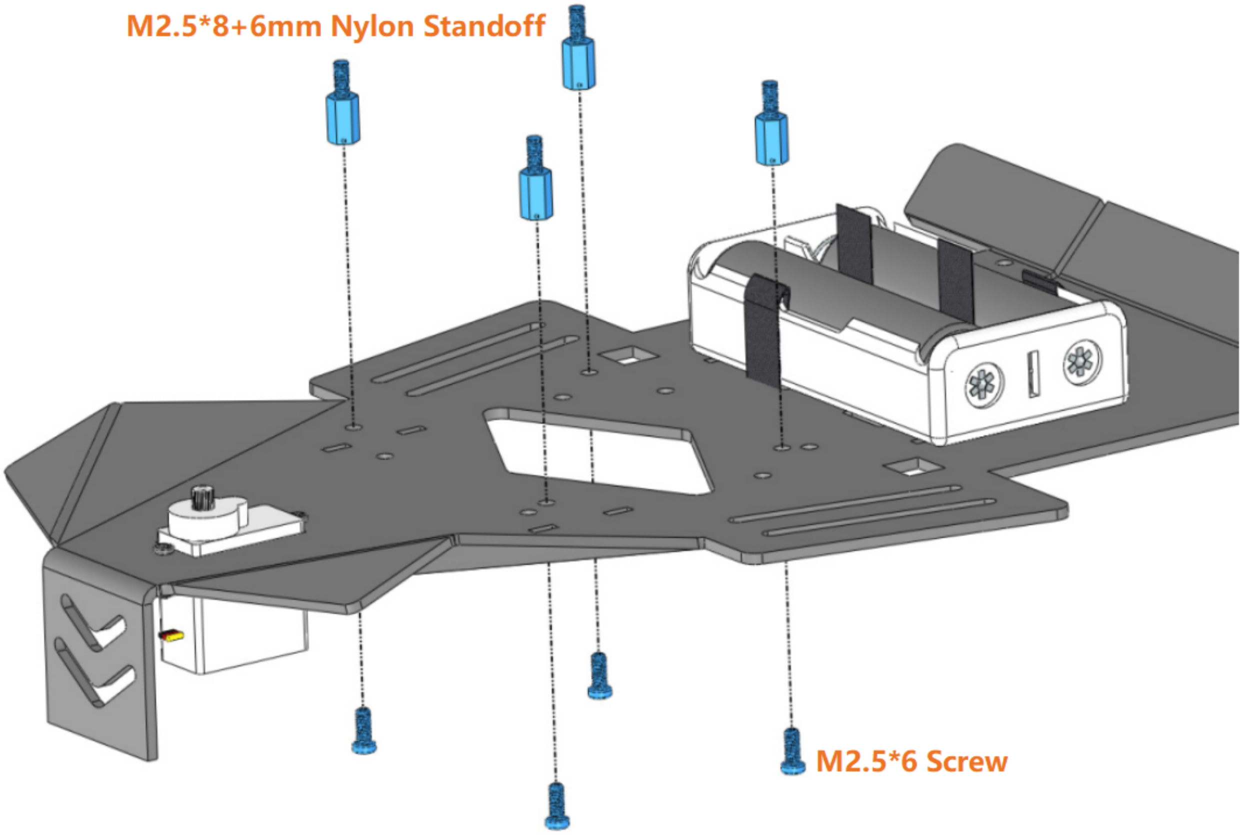
1.5.5 Assembling Servo

Assemble the servo from bottom to top.

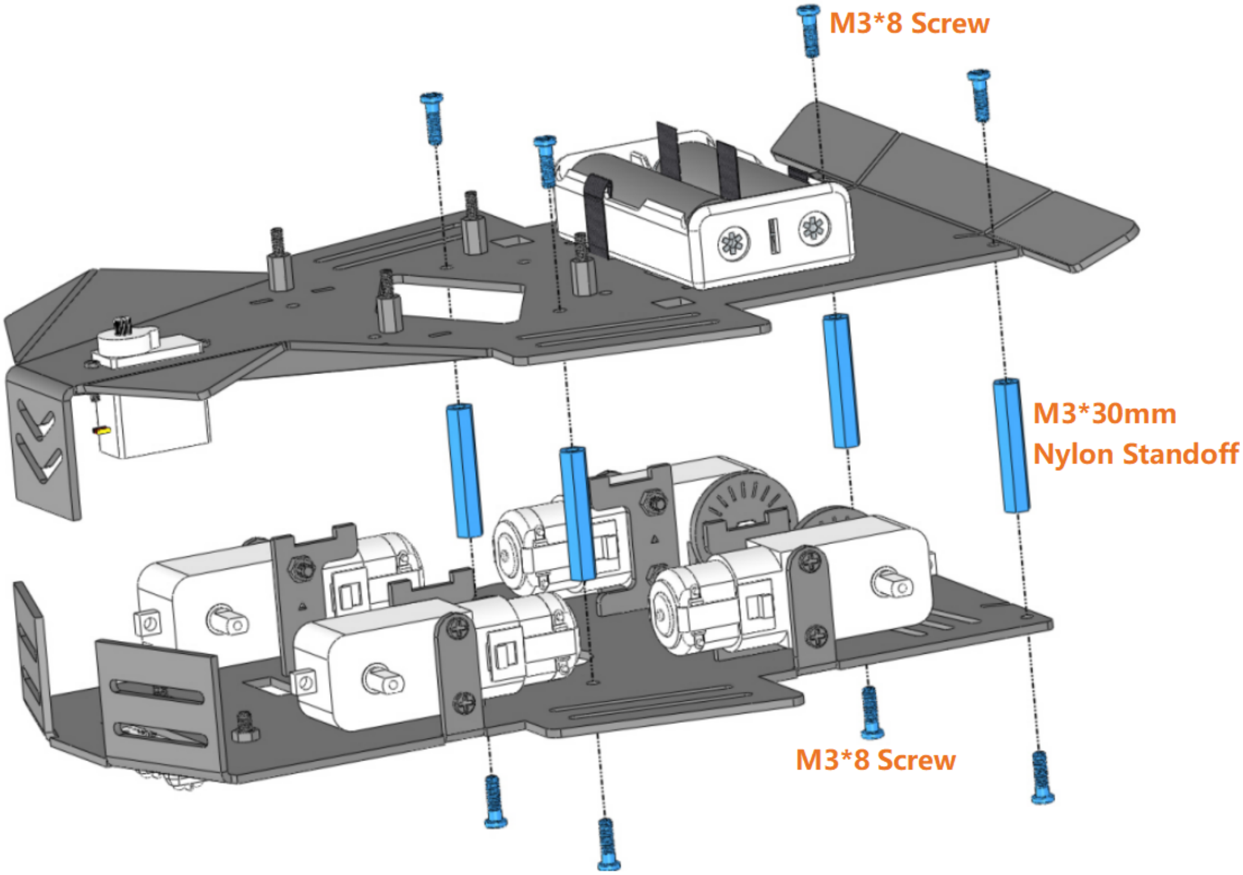
Note: The output axis is to face outwards.



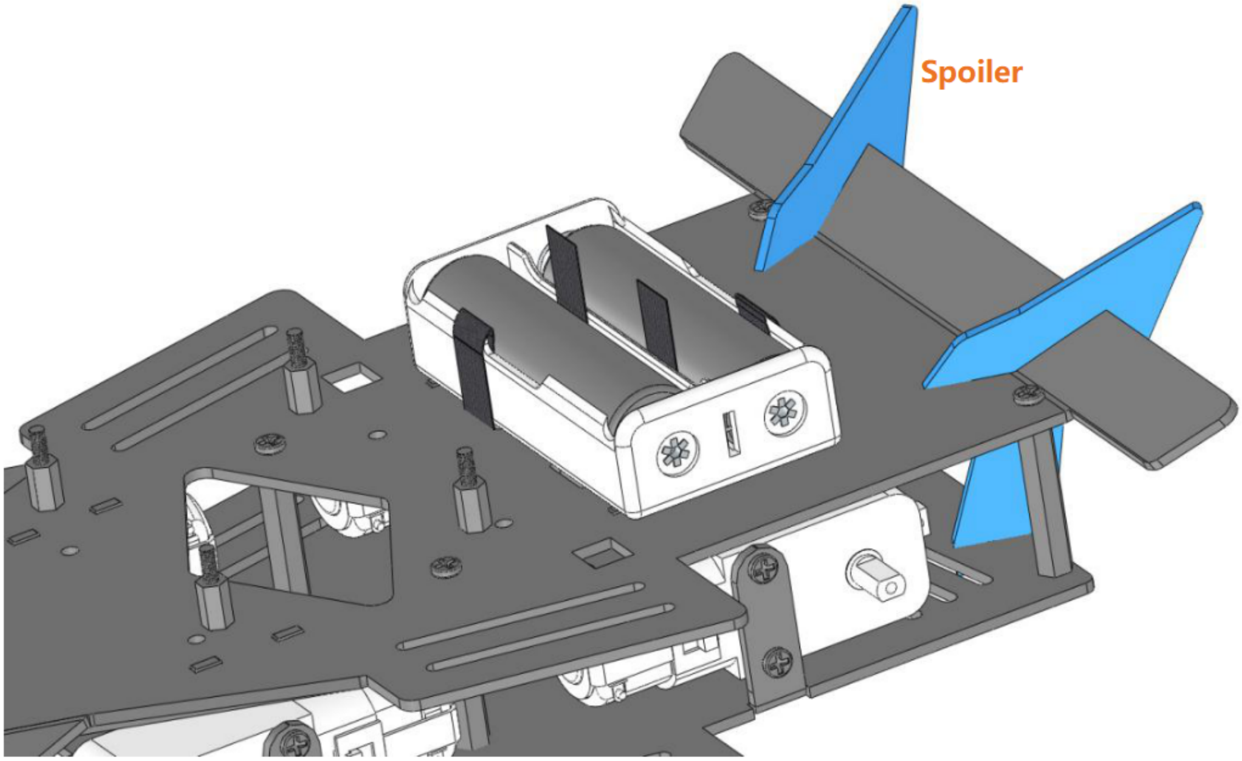
1.5.6 Assembling Nylon Standoffs



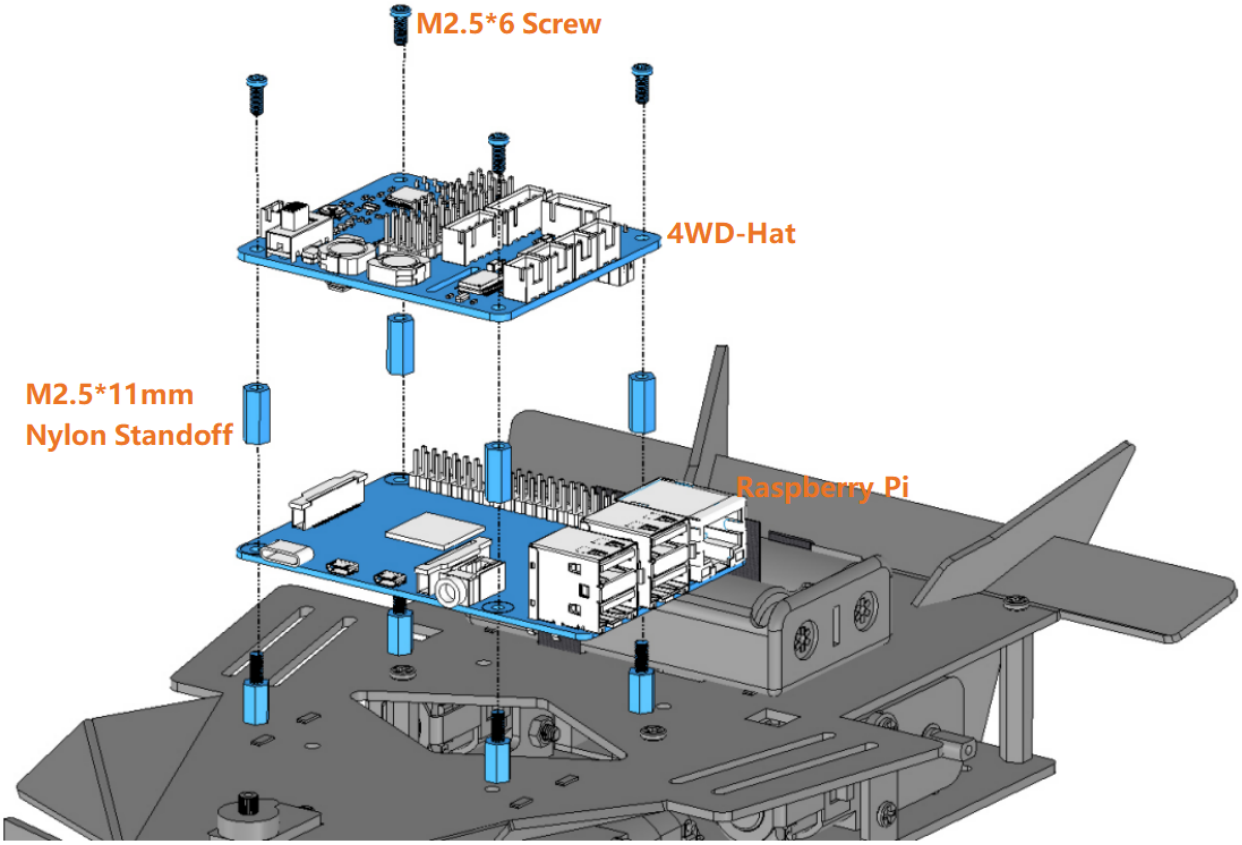
1.5.7 Assembling Top Plate and Bottom Plate



1.5.8 Assembling Spoiler



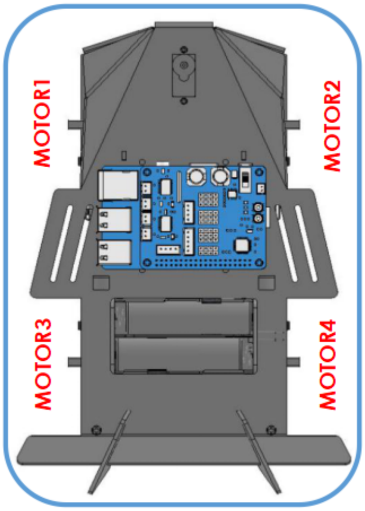
1.5.9 Assembling Control Board

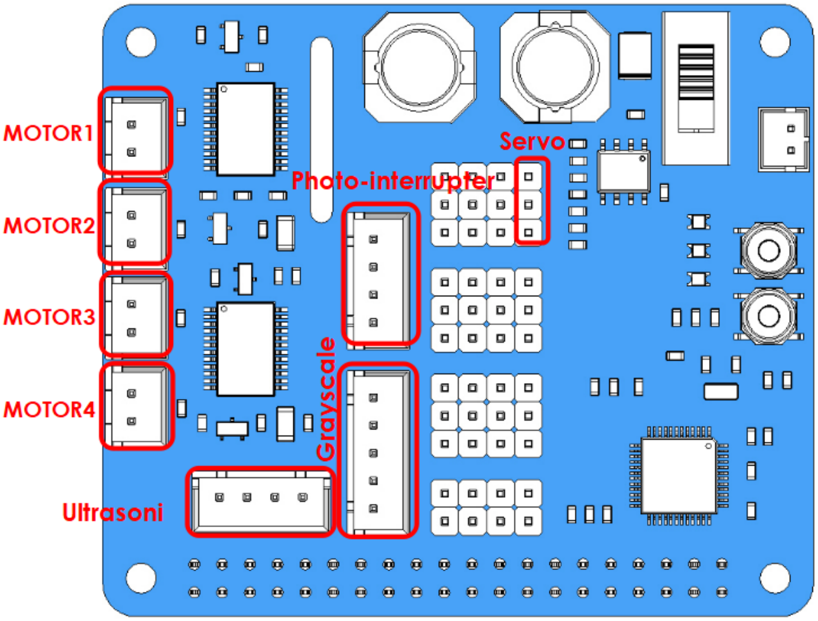


1.5.10 Connect Wires

Connect the modules properly with the guidance of the marks on the 4WD-Hat board.

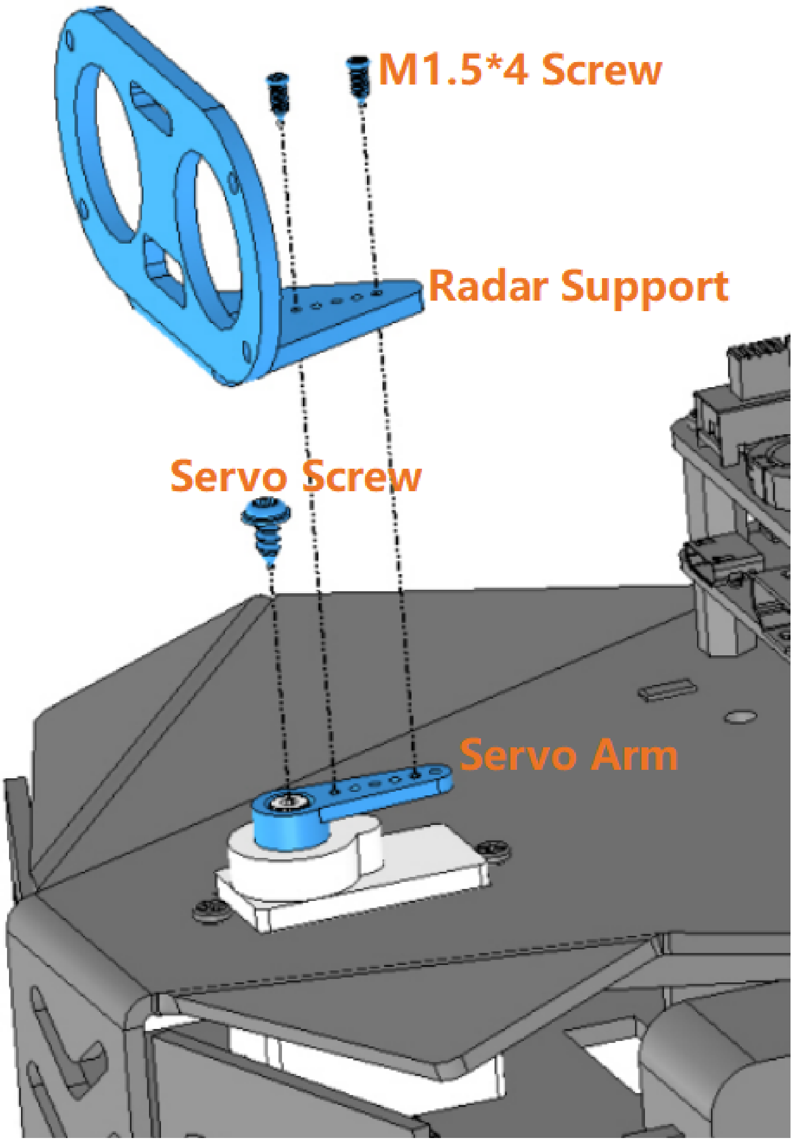
Note: For good appearance, you need to make sure that all the wires go through the bottom plate of the car to the HAT on the upper plate of the car.



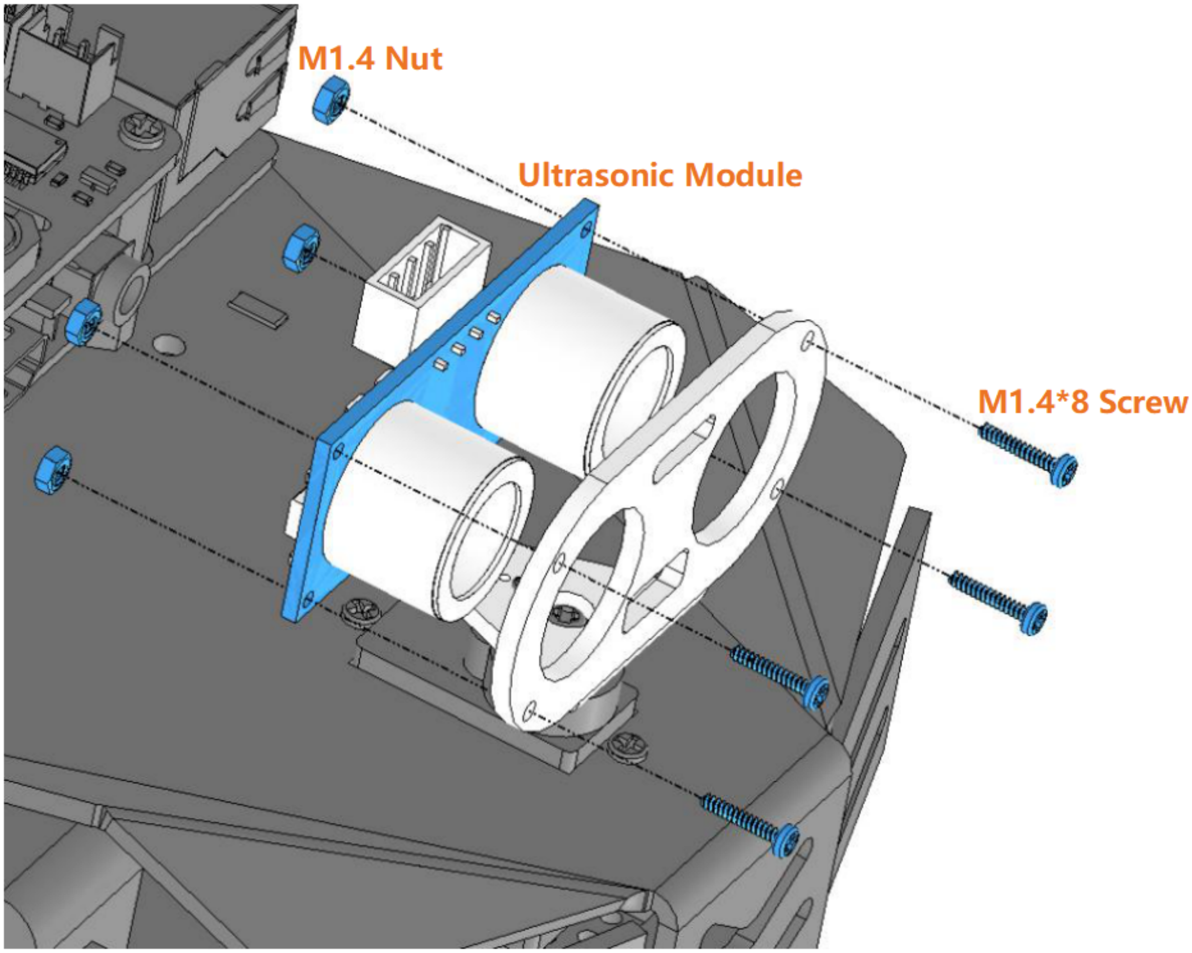


1.5.11 Assembling Radar Support

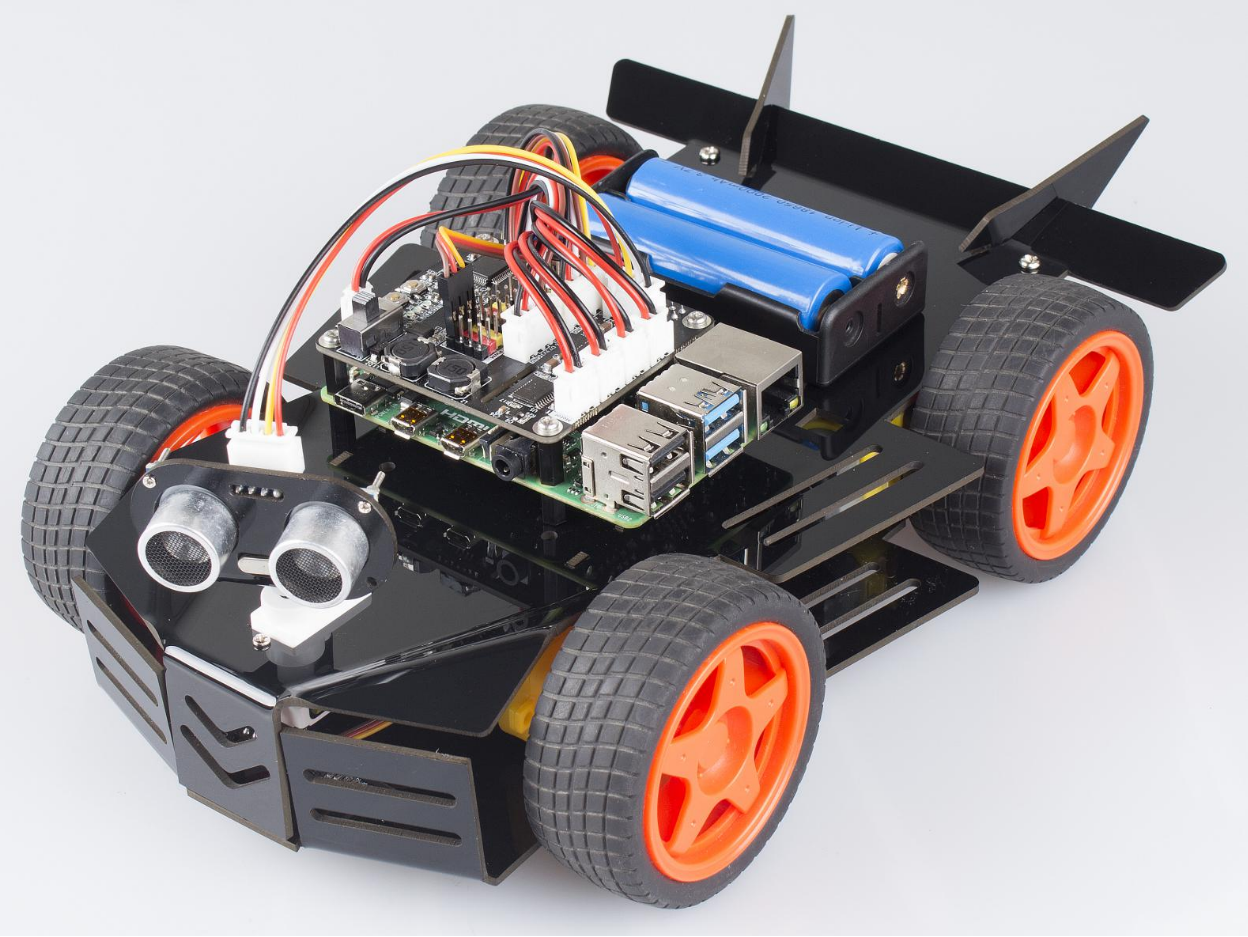
Note: Assemble the servo arm only when the Raspberry Pi is energized and the command picar-4wd test servois running.


M1.5*4 Screw
Radar Support
Servo Screw
Servo Arm

1.5.12 Assembling Ultrasonic Sensor Module



1.5.13 Assembling Wheels(finished)



1.6 Web Control

In the kit, there are two control methods: Web control and Python program control. In this chapter, we are provided with method of web control. So we can open the browser on different devices, including computer, cellphoneRecommended, tablet and so on. You can use most functions of Picar-4wd, such as obstacle avoidance, line-following, object following and voltage check. Note: Check the next chapter if you want to use python to control Picar-4wd.

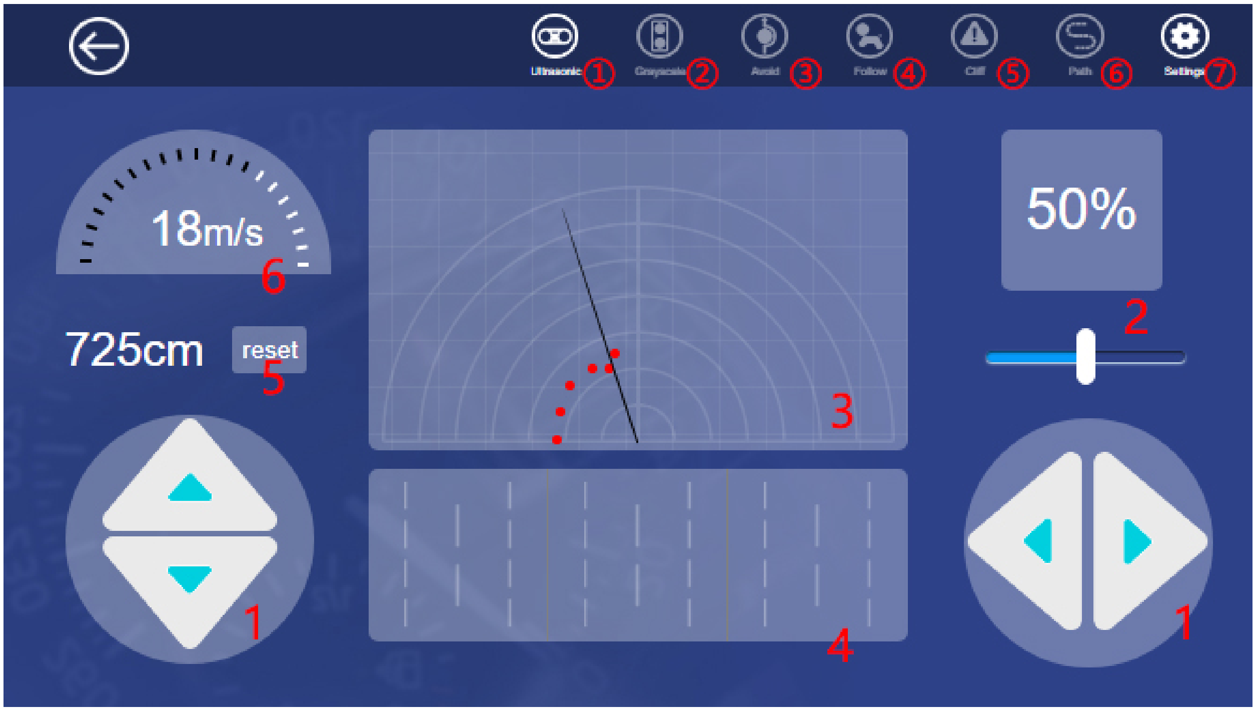
1.6.1 Boot Web Control

Input the command and start up the web control program.

```
cd /home/pi/picar-4wd/
picar-4wd web-example
```

As the program is running, on the browser, type the IP of the Raspberry Pi, and you can enter the Web interface.

Interface Introduction



1. Arrow Key
2. Power Regulator
3. Obstacle Monitor: checking the obstacle ahead within 35cm and 180°. Red dot represents there is an obstacle ahead detected by ultrasonic module.
4. Grayscale Monitor: displaying the color condition of the detected surface. If the module detects the black lines and then the cliff, the different colors will display on the monitor.
5. Mileage
6. Speed Monitor

Ultrasonic: turning on/off the obstacle monitor and the ultrasonic measurement function.

Grayscale: turning on/off the grayscale monitor and the line following function.

Avoid: turning on/off obstacle avoidance function can make the car get around the obstacles automatically or not.

Follow: turning on/off line following function can let the car follow the objects ahead or not.

Cliff: turn on/off edge detection. The car retreats once it detects the cliff or not. **If the car doesn't realize the effect, please adjust the threshold on the setting page**.

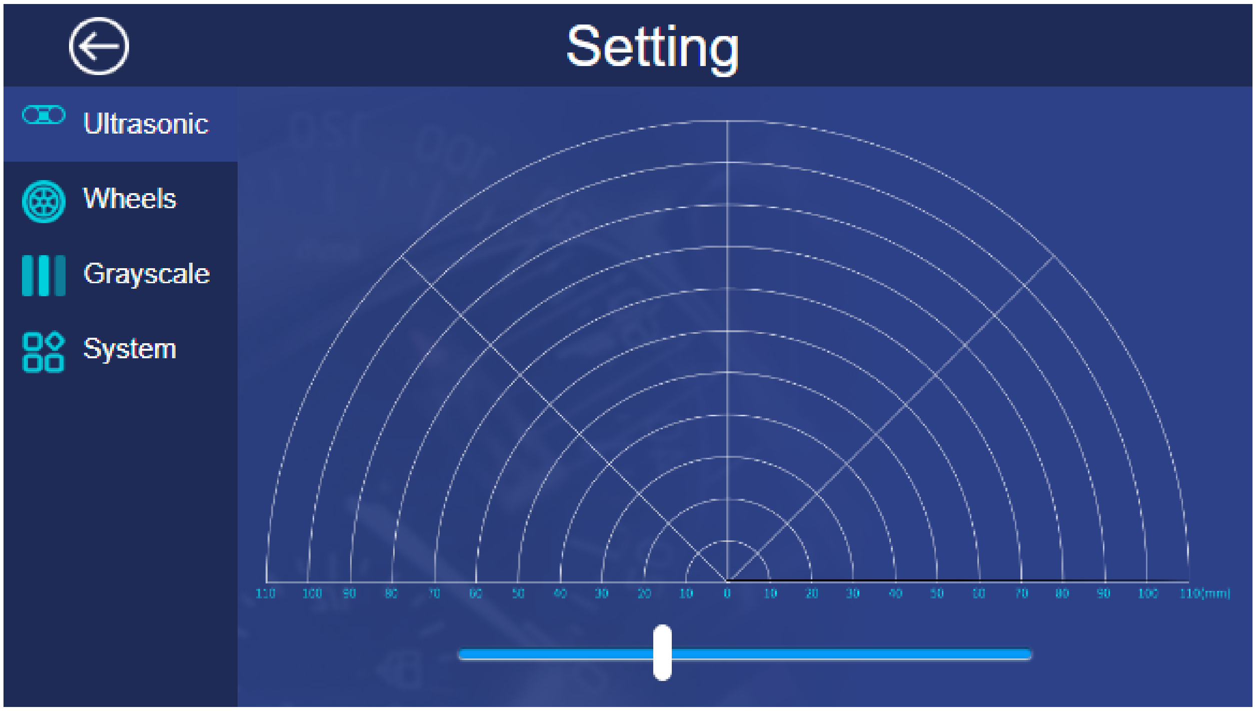
Path: to turn on/off line following function. Once you start up the line following function, the car will drive along the black lines on the ground. **If not, please adjust the threshold on the setting page**.

Setting: You can enter the Setting page by clicking this one.

1.6.2 Setting Page

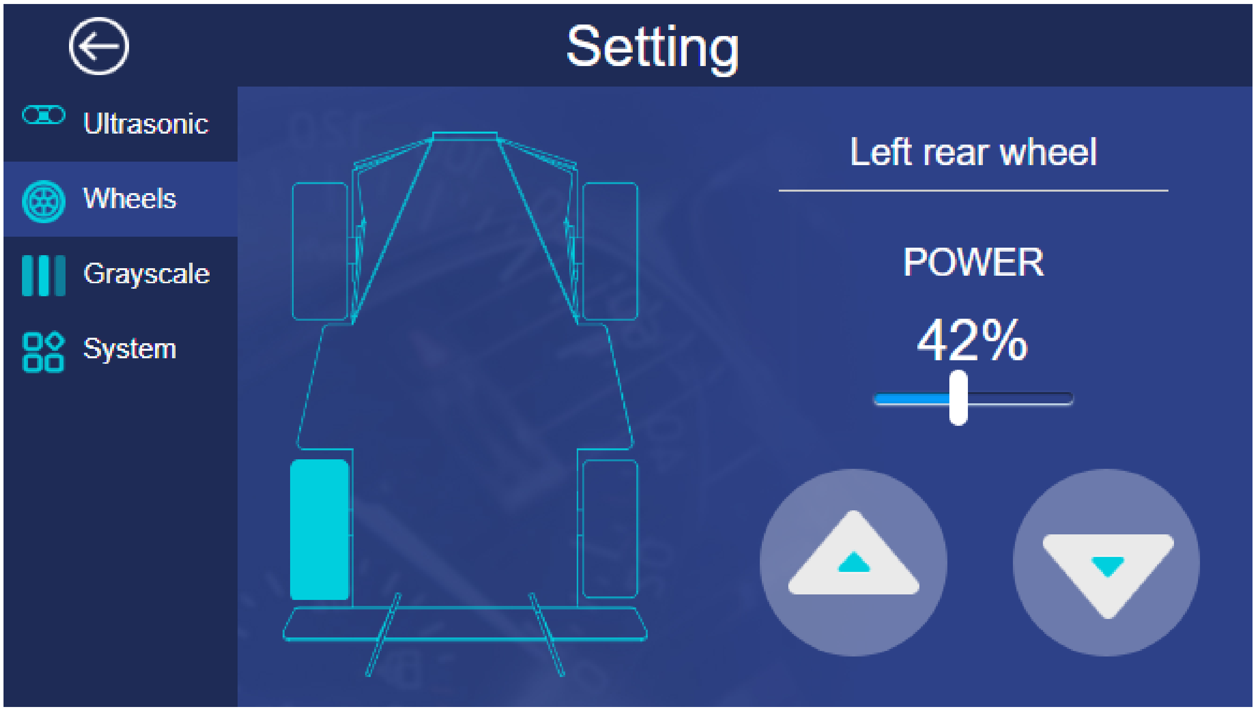
Ultrasonic

This page is used to test whether the ultrasonic sensor works well. The slider below can be used to adjust the current direction and angle of the ultrasonic sensor.


Setting
Ultrasonic
Wheels
Grayscale
System
110 100 90 80 70 60 50 40 30 20 10 0 10 20 30 40 50 60 70 80 90 100 110(mm)

Wheels

You can test the working condition of the wheels on this page.

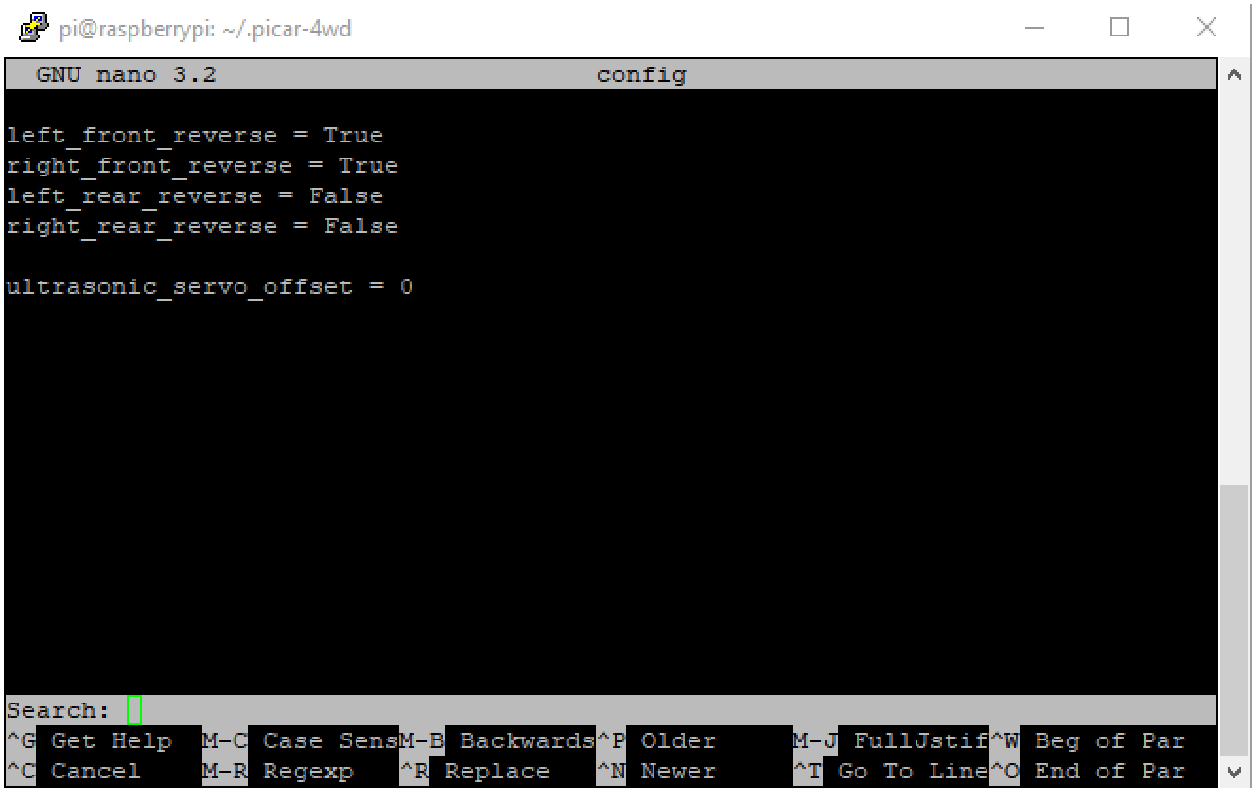


Click a wheel then set the POWER, click the two buttons in bottom right corner. When you press the button at left, the wheel goes forward; press the button on the right, the wheel go backward. If the car doesn't follow your operation like that, you should adjust the direction. The methods are as follows:

1) Input the following command to open the configure file.

```
cd ..
ls -a
cd .picar-4wd/
nano config
```

2) In the configure file, the first 4 lines of codes are used to control the turning of the motor. The last line is used to adjust the angle of the servo. For example, if in the above test, the direction of the motor at bottom left is reverse, so we change the code in the third line to left_rear_reverse = **True**.

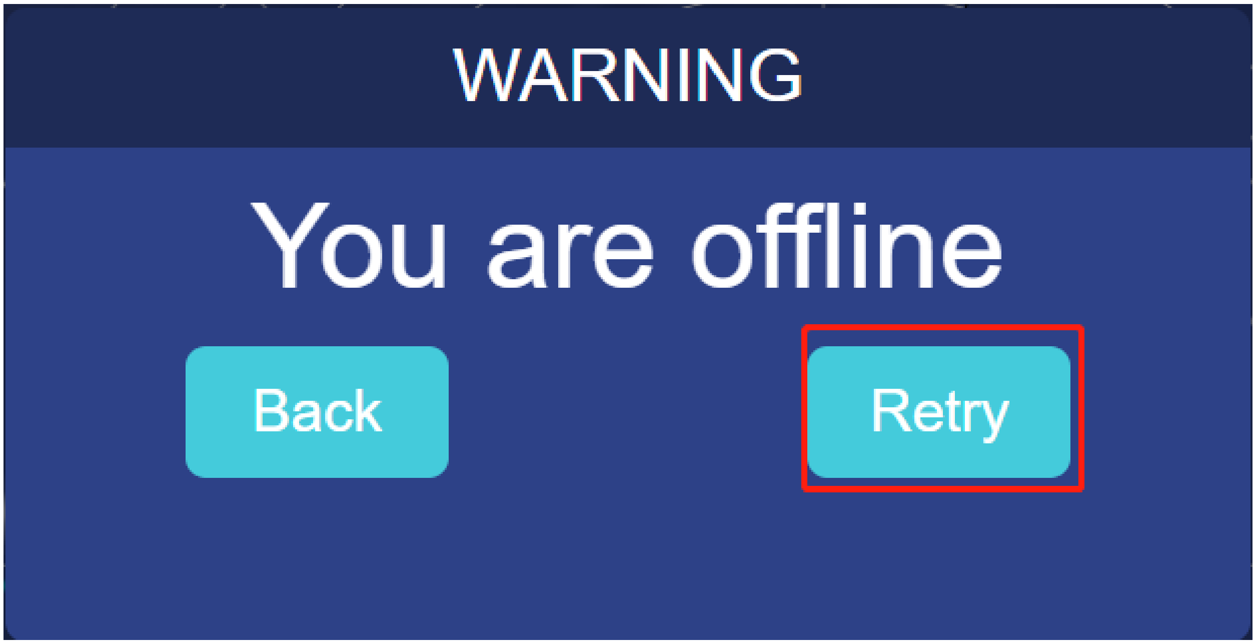


```
  GNU nano 3.2                                      config

left_front_reverse = True
right_front_reverse = True
left_rear_reverse = False
right_rear_reverse = False

ultrasonic_servo_offset = 0









Search: 
^G Get Help  M-C Case SensM-B Backwards^P Older      M-J FullJstif^W Beg of Par
^C Cancel    M-R Regexp   ^R Replace   ^N Newer      ^T Go To Line^O End of Par
```

3) Rerun the following commands.

```
cd /home/pi/picar-4wd/
picar-4wd web-example
```

4) In the browser, click Retry, and you can test the direction of the wheel ready to be used.



WARNING

You are offline

Grayscale

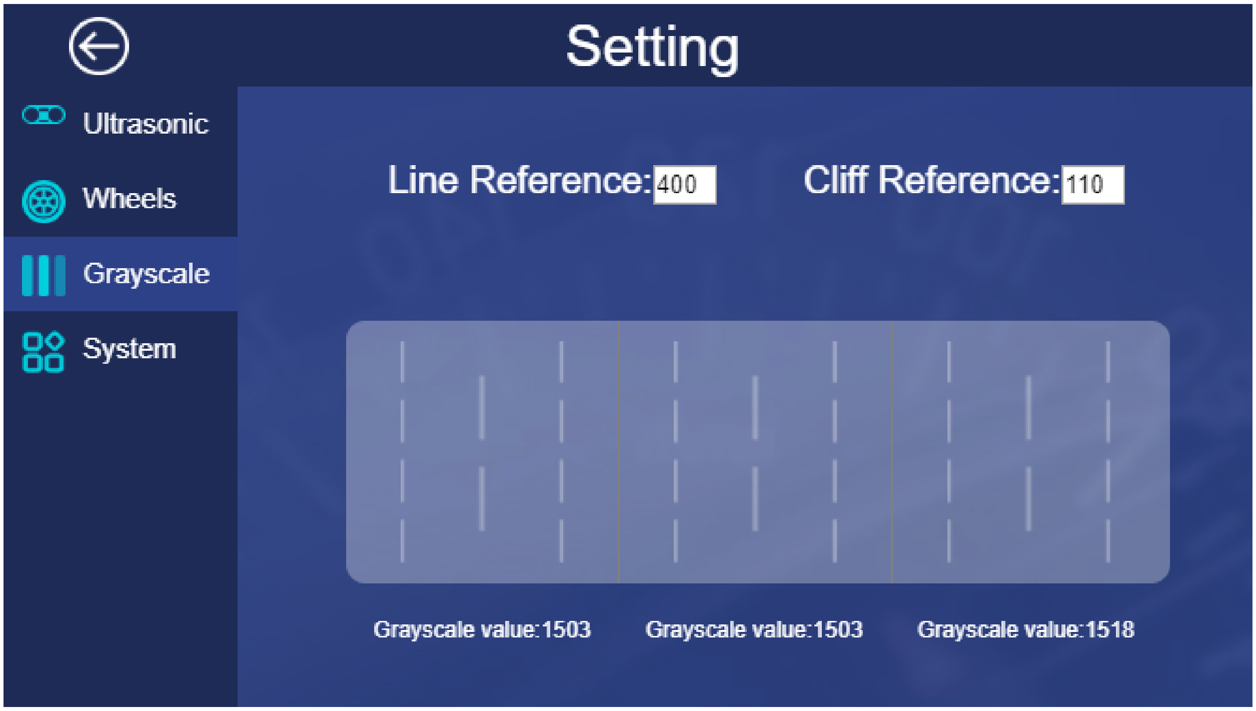
On this page, you can adjust the threshold of the line following and cliff detection.

Line Reference

Put the Picar-4wd on the white surface and record these three values of Grayscale. Now the value of Grayscale is large enough, about 1500. Then put the grayscale sensor module of Picar-4wd above the black surface and record these three values of Grayscale. Now, the value of Grayscale gets relatively small, about 150 as is predicted. So we set the value of Line Reference to 400. Of course, you can change the value. Once you finish doing this, you can see that the corresponding area turning into gray if the value of Grayscale is larger than 400; if not so, turning into black.

Cliff Reference

Put the car at the edge of the desk with the car's head sticking out from the desk. Record three values of Grayscale, and now the value is relatively small, less than 100. Please set the value of Cliff Reference to 110. When you finish doing this, the corresponding area appears red exclamation symbol if the value of Grayscale is less than 110.



Line Reference: 400

Cliff Reference: 110

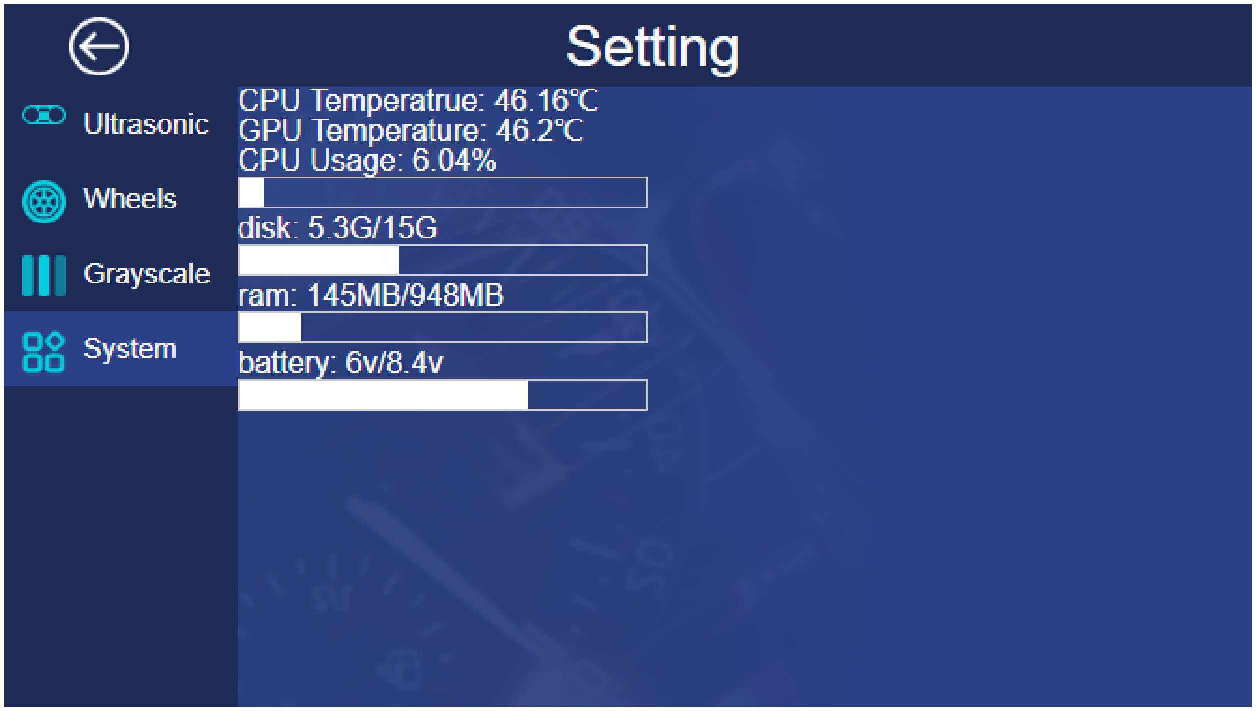
Grayscale value:1503

Grayscale value:1503

Grayscale value:1518

System

You can check the operating condition of the system on this page.



CPU Temperatrue: 46.16℃
GPU Temperature: 46.2℃
CPU Usage: 6.04%
disk: 5.3G/15G
ram: 145MB/948MB
battery: 6v/8.4v

Make the example program run automatically at startup.

```
picar-4wd web-example enable
```

If you want to turn off the Start on Boot, please type in the following command.

```
picar-4wd web-example disable
```

1.7 Python Program Control

1.7.1 Motor Calibration

Because of assembly methods and other reasons, your motor may still need a calibration direction. Otherwise, when the car moves forward, it may turn left, turn right, or even go backwards. Please follow the following steps to complete the calibration.

How to do

1. Enter the examples folder.

```
cd /home/pi/picar-4wd/examples
```

2. Run the `move_forward.py` . This is a very simple example, which will let the car go forward.

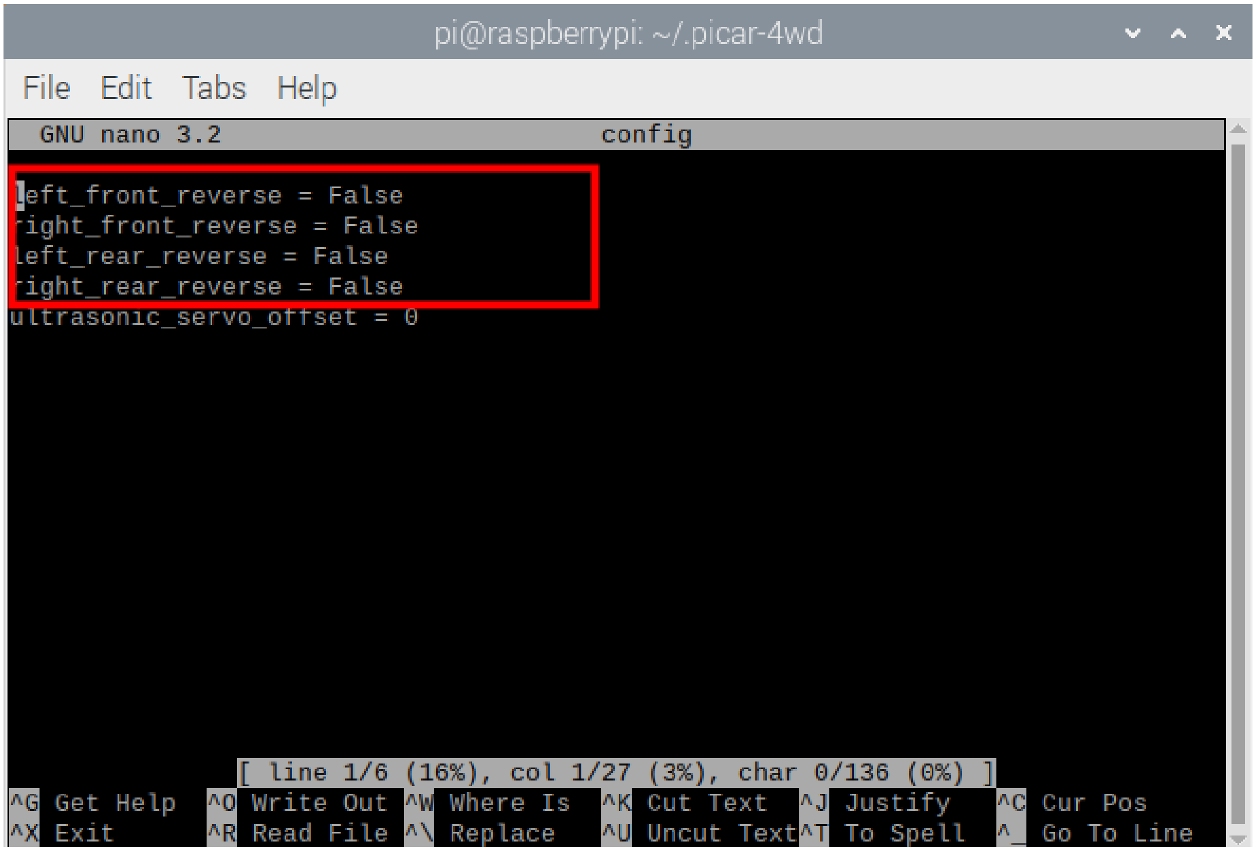
```
python3 move_forward.py
```

3. Observe whether the wheels of the car are driving forward and marked the wrong motor.

4. Open the configuration file.

```
sudo nano /home/pi/.picar-4wd/config
```

5. Change the reverse FLAG of the error motor from `False` to `True`.



```
  GNU nano 3.2                                  config

left_front_reverse = False
right_front_reverse = False
left_rear_reverse = False
right_rear_reverse = False
ultrasonic_servo_offset = 0

               [ line 1/6 (16%), col 1/27 (3%), char 0/136 (0%) ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Spell  ^_ Go To Line
```

6. Run again `move_forward.py`, if the car is driving correctly, the calibration is completed.

```
python3 move_forward.py
```

1.7.2 Python Program Control

Python Example Program is in the examples folder.

```
cd /home/pi/picar-4wd/examples
```

You can run them by using Python3.

keyboard_control.py

After running the example, pressW, A, S, D, and you can get the car going ahead, back, left, right. And the keys4and6can increase or decrease the power of the motor.

Run it by using the following command.

```
python3 keyboard_control.py
```

obstacle_avoidance.py

After running the example, the car goes ahead automatically.

It can also turn right to get around the obstacles if there is anything obstructive ahead.

```
python3 obstacle_avoidance.py
```

track_line.py

If you paste a black lane (width: around 2.5cm) on the white floor, the car will move along the lane.

```
python3 track_line.py
```

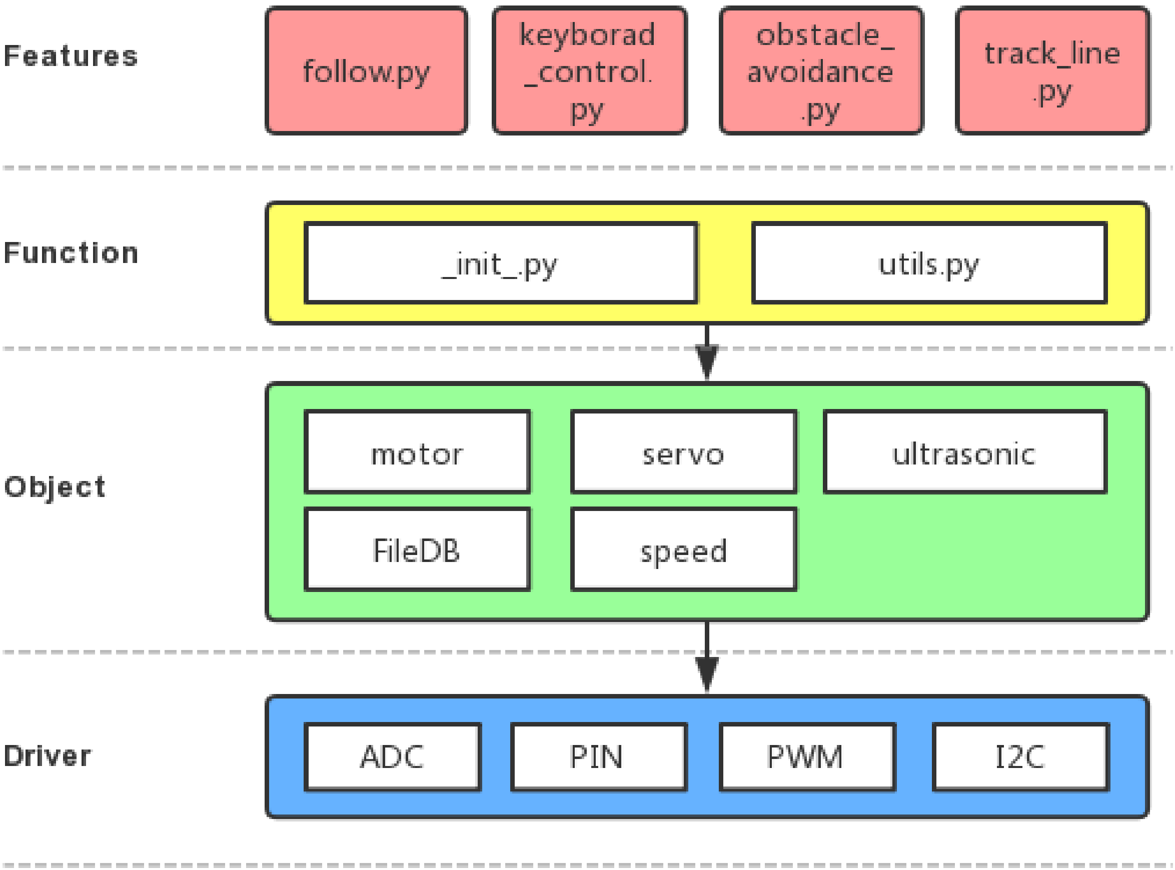
follow.py

When the example runs, the car will follow the object ahead.

```
python3 follow.py
```

1.7.3 Python Example Code Analysis

Program Structure Framework



keyboard_control.py

In the code, we use readchar and readkey to get the input state of the key, then we assign some functions to the keys by using if statement.

Note: You can **Modify/Reset/Copy/Run/Stop** the code below. But before that, you need to go to source code path like `picar-4wd/examples`. After modifying the code, you can run it directly to see the effect.

```
import picar_4wd as fc
import sys
import tty
import termios
import asyncio

power_val = 50
key = 'status'
print("If you want to quit.Please press q")
def readchar():
```

(continues on next page)

(continued from previous page)

```
    fd = sys.stdin.fileno()
    old_settings = termios.tcgetattr(fd)
    try:
        tty.setraw(sys.stdin.fileno())
        ch = sys.stdin.read(1)
    finally:
        termios.tcsetattr(fd, termios.TCSADRAIN, old_settings)
    return ch

def readkey(getchar_fn=None):
    getchar = getchar_fn or readchar
    c1 = getchar()
    if ord(c1) != 0x1b:
        return c1
    c2 = getchar()
    if ord(c2) != 0x5b:
        return c1
    c3 = getchar()
    return chr(0x10 + ord(c3) - 65)

def Keyborad_control():
    while True:
        global power_val
        key=readkey()
        if key=='6':
            if power_val <=90:
                power_val += 10
                print("power_val:",power_val)
        elif key=='4':
            if power_val >=10:
                power_val -= 10
                print("power_val:",power_val)
        if key=='w':
            fc.forward(power_val)
        elif key=='a':
            fc.turn_left(power_val)
        elif key=='s':
            fc.backward(power_val)
        elif key=='d':
            fc.turn_right(power_val)
        else:
            fc.stop()
        if key=='q':
            print("quit")
            break
if __name__ == '__main__':
    Keyborad_control()
```

obstacle_avoidance.py

In the code, the distance measurement radar (ultrasonic + servo) will take sample for 11 times in total (once every 18°). Then it makes judgement on the distance states of the fourth and the seventh sampling. If radar indicates no obstacle(the returned value is 2), the car will continue moving ahead, or else the car turn right to get around the obstacle.

Note: Check "scan_step()", "get_status_at()" in the "_init_.py" for more details about the judgement on radar sampling and distance state.

```
import picar_4wd as fc

speed = 30

def main():
    while True:
        scan_list = fc.scan_step(35)
        if not scan_list:
            continue

        tmp = scan_list[3:7]
        print(tmp)
        if tmp != [2,2,2,2]:
            fc.turn_right(speed)
        else:
            fc.forward(speed)

if __name__ == "__main__":
    try:
        main()
    finally:
        fc.stop()
```

track_line.py

In this code, grayscale sensor module will judge the gray level of the floor under the car's head. The car adjusts the directions according to the detection values of these three detectors to follow the black line.

Note: About **Grayscale judgement**, refer to "get_line_status()" in "_init_.py" for more details.

```
import picar_4wd as fc

Track_line_speed = 20

def Track_line():
    gs_list = fc.get_grayscale_list()
    if fc.get_line_status(400,gs_list) == 0:
        fc.forward(Track_line_speed)
    elif fc.get_line_status(400,gs_list) == -1:
        fc.turn_left(Track_line_speed)
    elif fc.get_line_status(400,gs_list) == 1:
        fc.turn_right(Track_line_speed)
```

(continues on next page)

(continued from previous page)

```
if __name__=='__main__':
    while True:
        Track_line()
```

follow.py

In the code, the ultrasonic module will take sample to form an array containing 11 elements; the module detects once every 18° and puts the distance state (012) of every angle into the array. If there is something obstructive at the right front, the detection value is [22222211122].

The array takes the 2 as a delimiter to divide the array anew. The examples above will be divided into several null elements and a [111]- value element.

The corresponding angular range of the element with a value [111] is 108°~144° (0°+18°*6=108°, 0°+18°*8=144°), and the car will turn its angle into a median 126°144+108/2

Judge the distance, and keep the car away from the object about 10cm.

Note: Refer to "scan_step()" and "get_status_at()" in "_init_.py" to get more details about radar sampling and the distance state judgement.

```
import picar_4wd as fc

speed = 30

def main():
    while True:
        scan_list = fc.scan_step(23)
        # print(scan_list)
        if not scan_list:
            continue

        scan_list = [str(i) for i in scan_list]
        scan_list = "".join(scan_list)
        paths = scan_list.split("2")
        length_list = []
        for path in paths:
            length_list.append(len(path))
        # print(length_list)
        if max(length_list) == 0:
            fc.stop()
        else:
            i = length_list.index(max(length_list))
            pos = scan_list.index(paths[i])
            pos += (len(paths[i]) - 1) / 2
            # pos = int(pos)
            delta = len(scan_list) / 3
            # delta *= us_step/abs(us_step)
            if pos < delta:
                fc.turn_left(speed)
            elif pos > 2 * delta:
                fc.turn_right(speed)
            else:
```

(continues on next page)

(continued from previous page)

```
                if scan_list[int(len(scan_list)/2-1)] == "0":
                    fc.backward(speed)
                else:
                    fc.forward(speed)

if __name__ == "__main__":
    try:
        main()
    finally:
        fc.stop()
```

1.8 Appendix

1.8.1 Remote Desktop

There are two ways to control the desktop of the Raspberry Pi remotely:

VNC and **XRDP**, you can use any of them.

VNC

You can use the function of remote desktop through VNC.

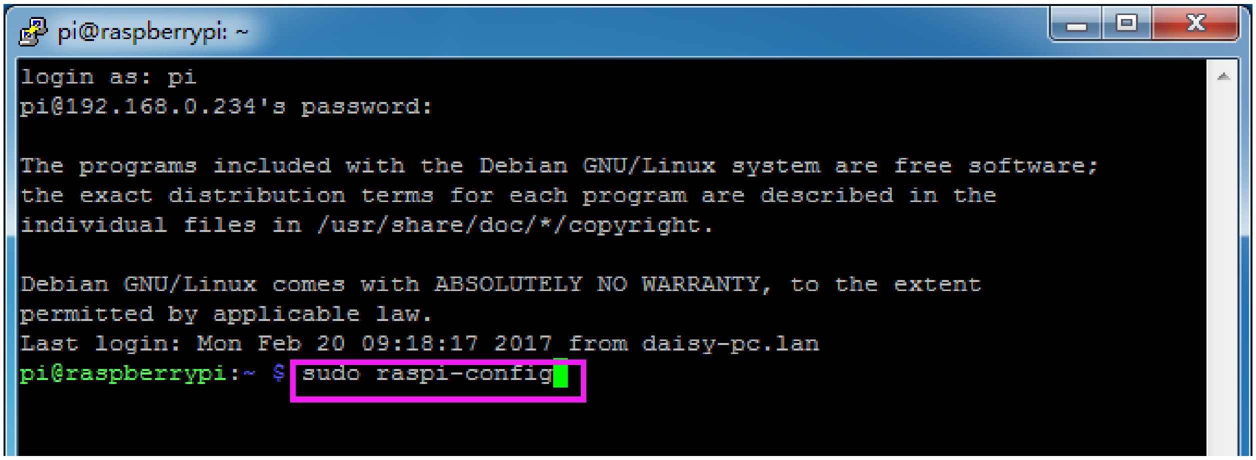
Enable VNC service

The VNC service has been installed in the system. By default, VNC is disabled. You need to enable it in config.

Step 1

Input the following command:

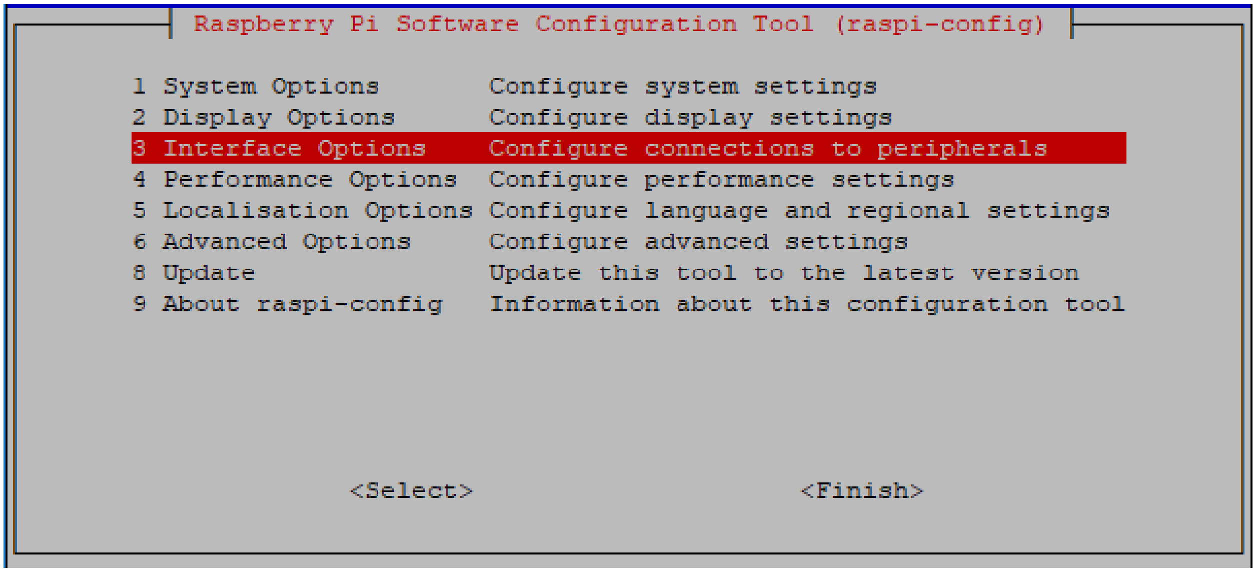
```
sudo raspi-config
```



```
login as: pi
pi@192.168.0.234's password:

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: Mon Feb 20 09:18:17 2017 from daisy-pc.lan
pi@raspberrypi:~ $ sudo raspi-config
```

Step 2

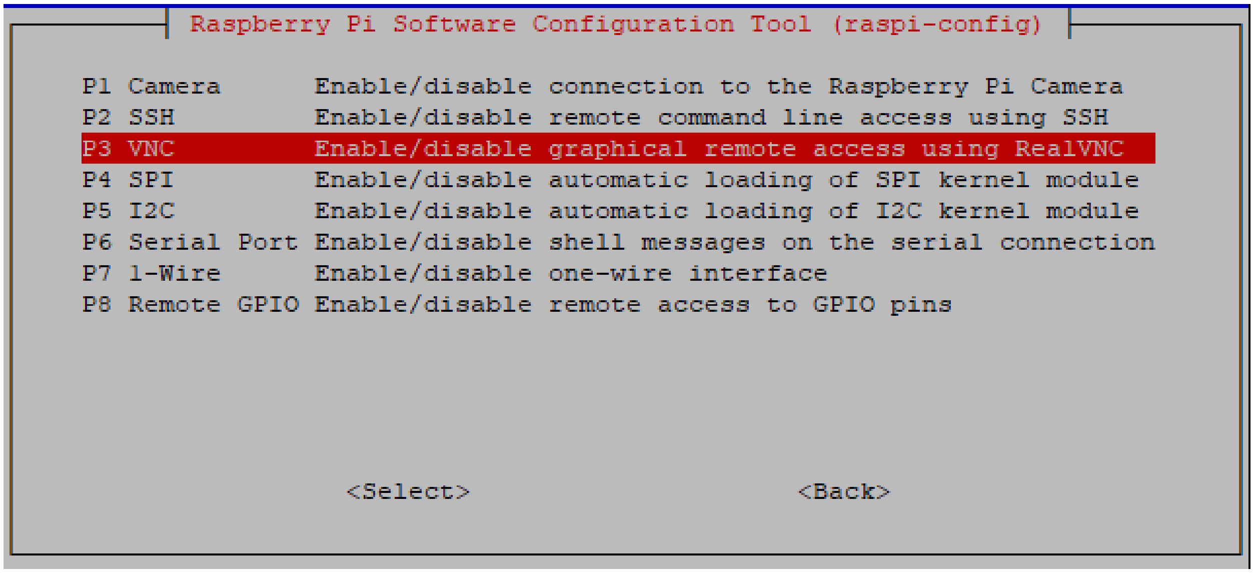
Choose **3 Interfacing Options** by press the down arrow key on your keyboard, then press the **Enter** key.



```
Raspberry Pi Software Configuration Tool (raspi-config)

1 System Options       Configure system settings
2 Display Options      Configure display settings
3 Interface Options    Configure connections to peripherals
4 Performance Options  Configure performance settings
5 Localisation Options Configure language and regional settings
6 Advanced Options     Configure advanced settings
8 Update               Update this tool to the latest version
9 About raspi-config   Information about this configuration tool

              <Select>                    <Finish>
```

Step 3

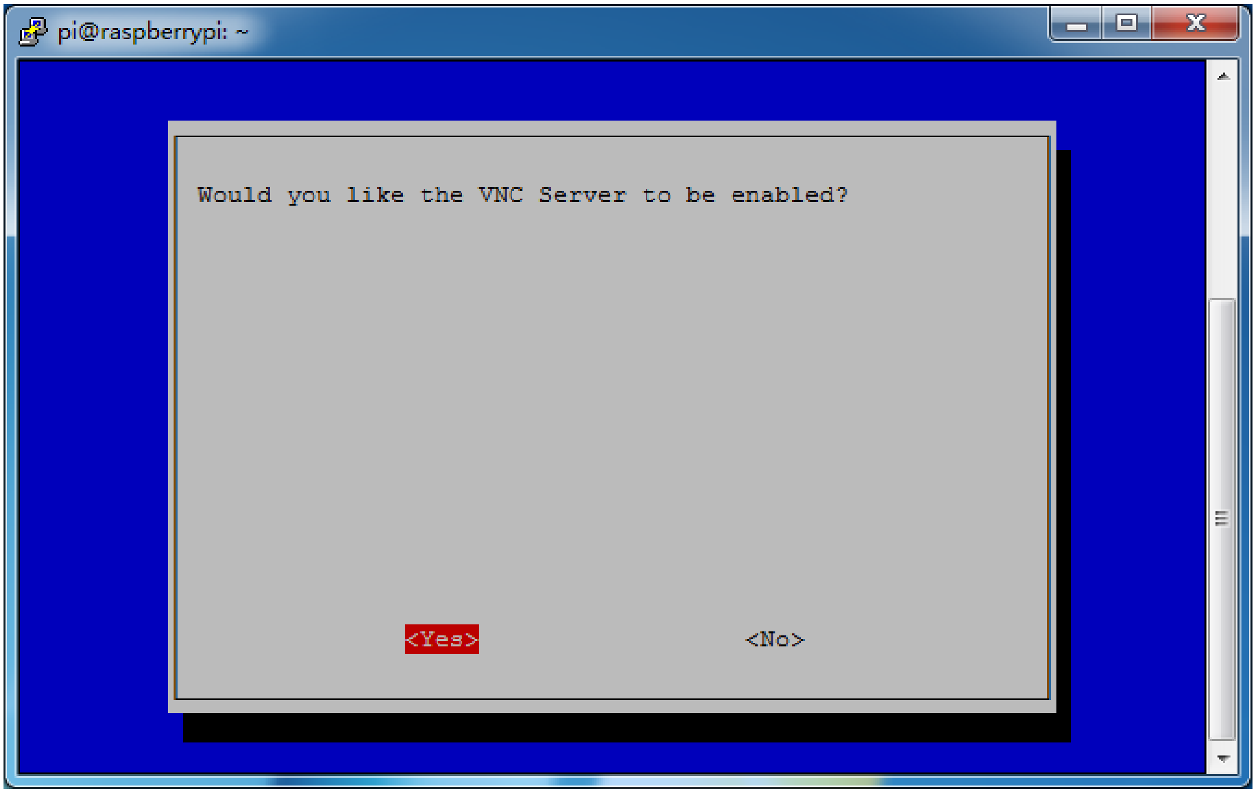
P3 VNC



```
Raspberry Pi Software Configuration Tool (raspi-config)

P1 Camera      Enable/disable connection to the Raspberry Pi Camera
P2 SSH         Enable/disable remote command line access using SSH
P3 VNC         Enable/disable graphical remote access using RealVNC
P4 SPI         Enable/disable automatic loading of SPI kernel module
P5 I2C         Enable/disable automatic loading of I2C kernel module
P6 Serial Port Enable/disable shell messages on the serial connection
P7 1-Wire      Enable/disable one-wire interface
P8 Remote GPIO Enable/disable remote access to GPIO pins

              <Select>                    <Back>
```

Step 4

Select **Yes -> OK -> Finish** to exit the configuration.



```
Would you like the VNC Server to be enabled?

<Yes>                <No>
```

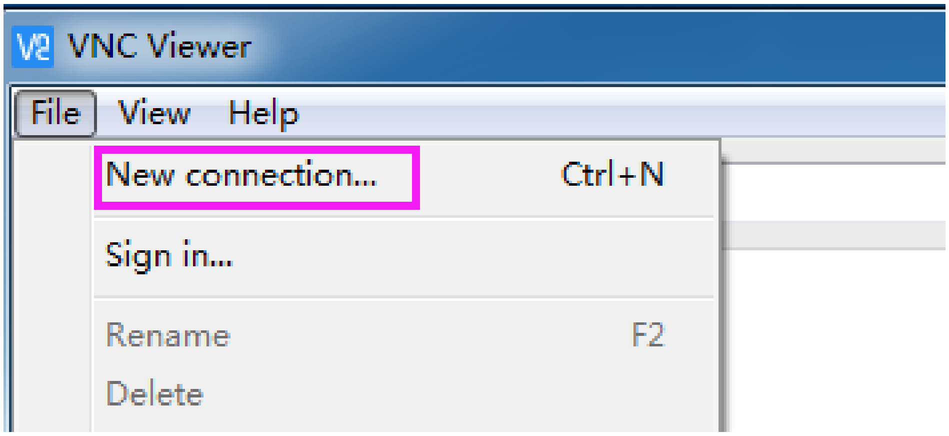
Login to VNC

Step 1

You need to download and install the VNC Viewer on personal computer. After the installation is done, open it.

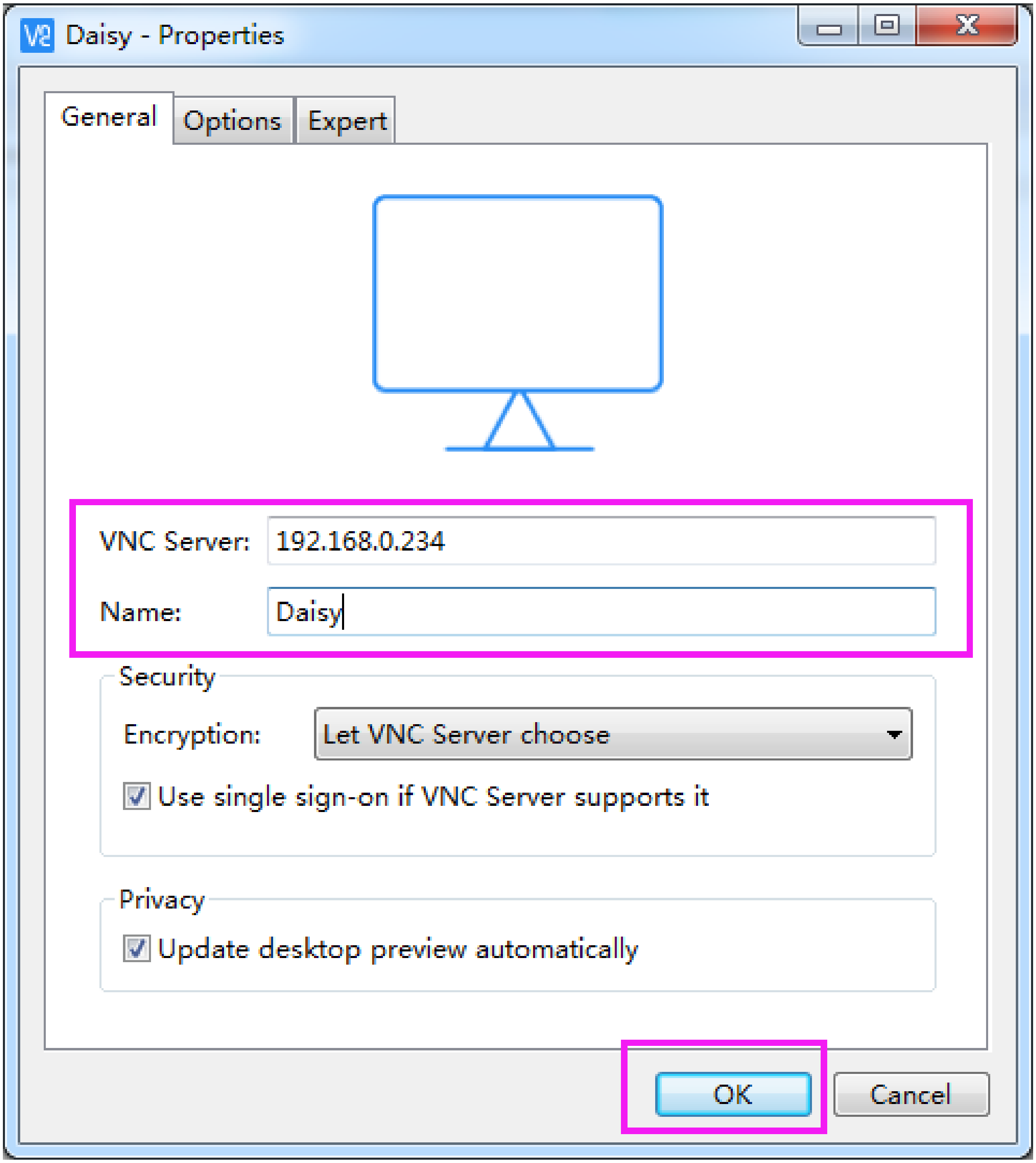
Step 2

Then select "**New connection**".



Step 3

Input IP address of Raspberry Pi and any **Name**.



VNC Server: 192.168.0.234

Name: Daisy

Security

Encryption: Let VNC Server choose

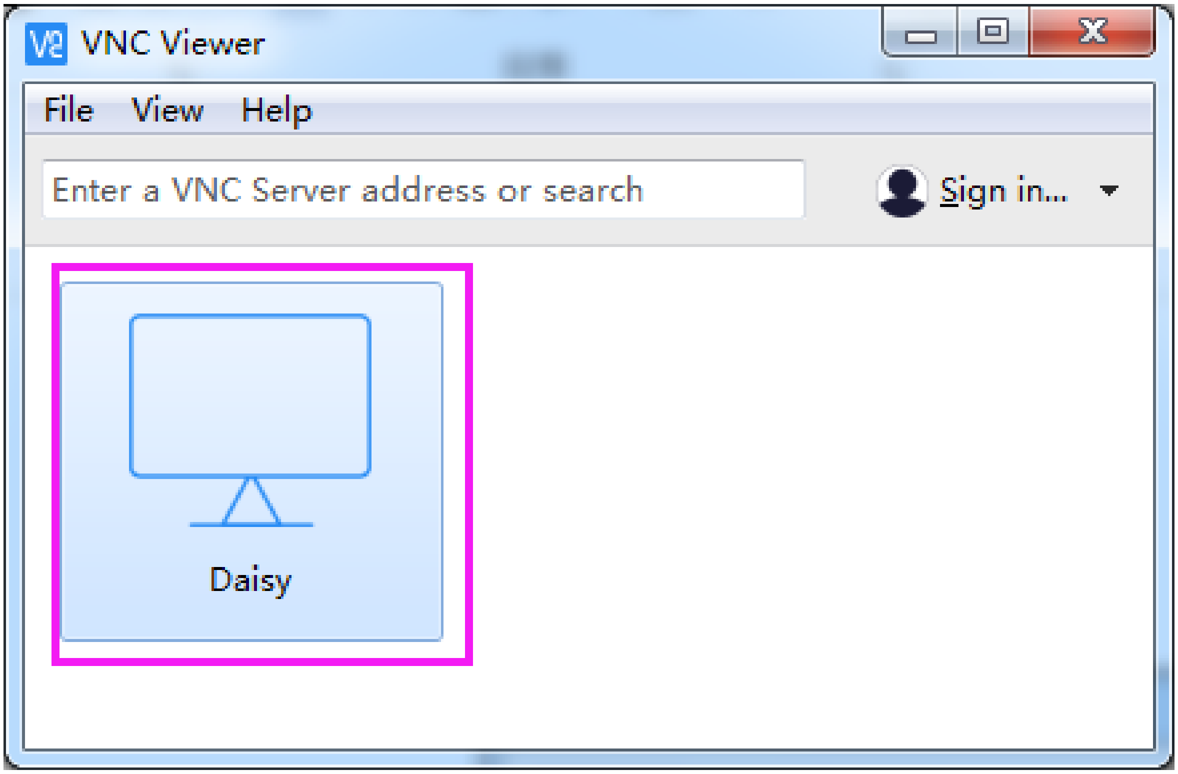
☑ Use single sign-on if VNC Server supports it

Privacy

☑ Update desktop preview automatically

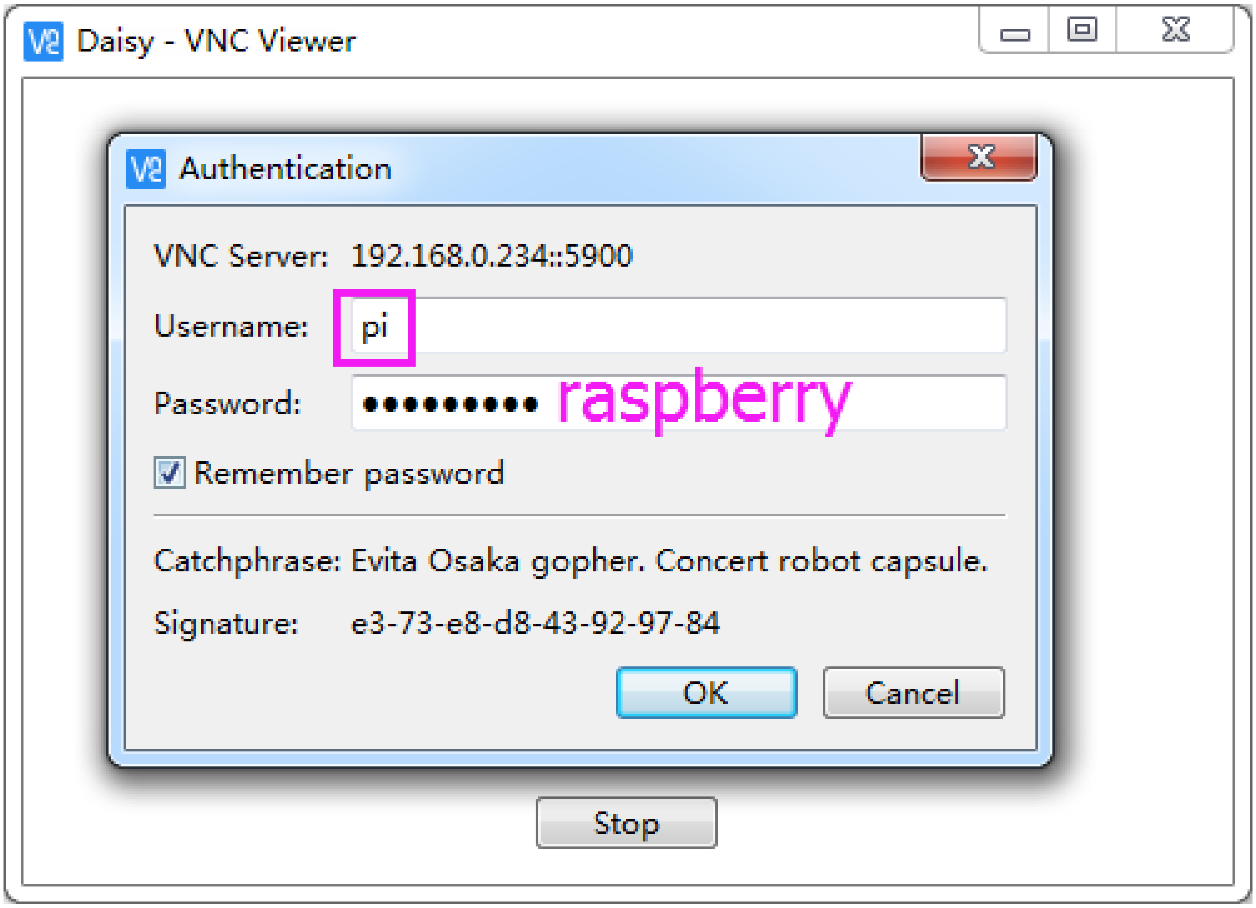
Step 4

Double click the **connection** just created:



Step 5

Enter Username (**pi**) and Password (**raspberry** by default).



Daisy - VNC Viewer

Authentication

VNC Server: 192.168.0.234::5900

Username: pi

Password: ●●●●●●●●●● raspberry

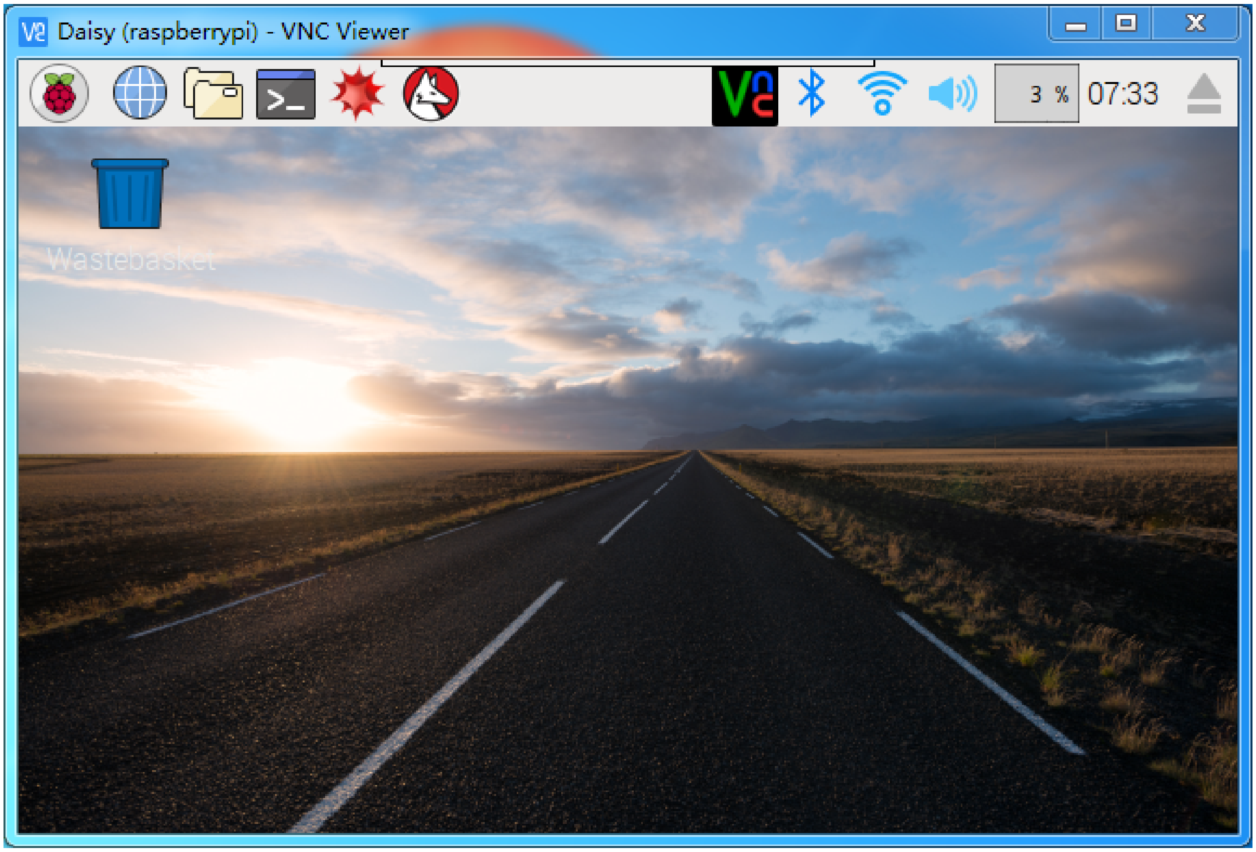
☑ Remember password

Catchphrase: Evita Osaka gopher. Concert robot capsule.

Signature: e3-73-e8-d8-43-92-97-84

Step 6

Now you can see the desktop of the Raspberry Pi:



That's the end of the VNC part.

XRDP

Another method of remote desktop is XRDP, it provides a graphical login to remote machines using RDP (Microsoft Remote Desktop Protocol).

Install XRDP

Step 1

Login to Raspberry Pi by using SSH.

Step 2

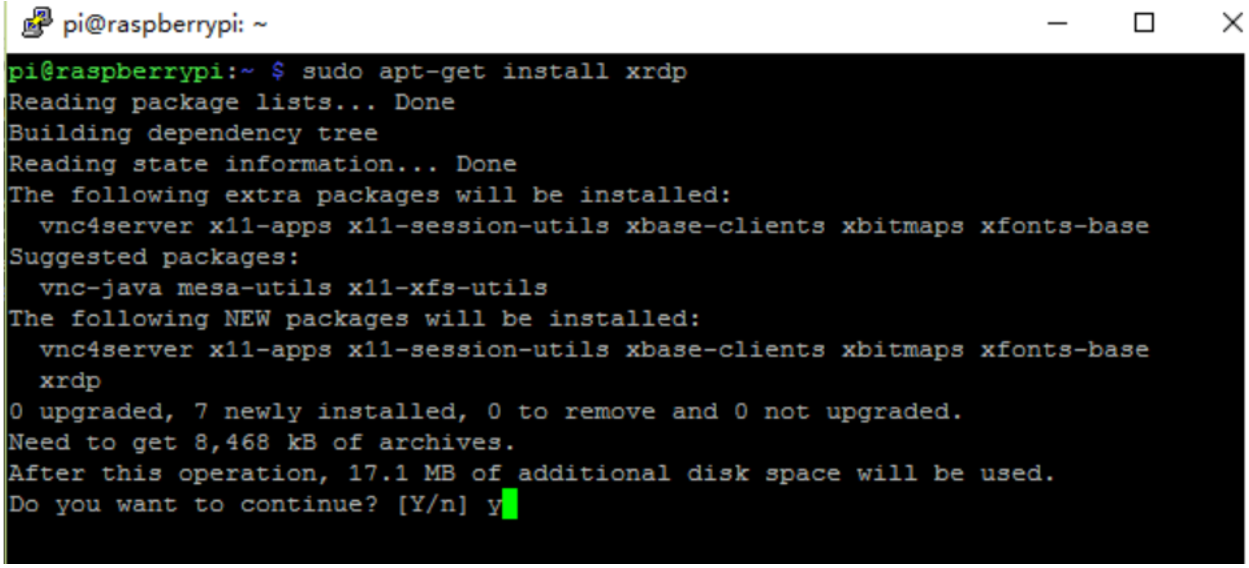
Input the following instructions to install XRDP.

```
sudo apt-get update
sudo apt-get install xrdp
```

Step 3

Later, the installation starts.

Enter "Y", press key "Enter" to confirm.



```
pi@raspberrypi:~ $ sudo apt-get install xrdp
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  vnc4server x11-apps x11-session-utils xbase-clients xbitmaps xfonts-base
Suggested packages:
  vnc-java mesa-utils x11-xfs-utils
The following NEW packages will be installed:
  vnc4server x11-apps x11-session-utils xbase-clients xbitmaps xfonts-base
  xrdp
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 8,468 kB of archives.
After this operation, 17.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Step 4

Finished the installation, you should login to your Raspberry Pi by using Windows remote desktop applications.

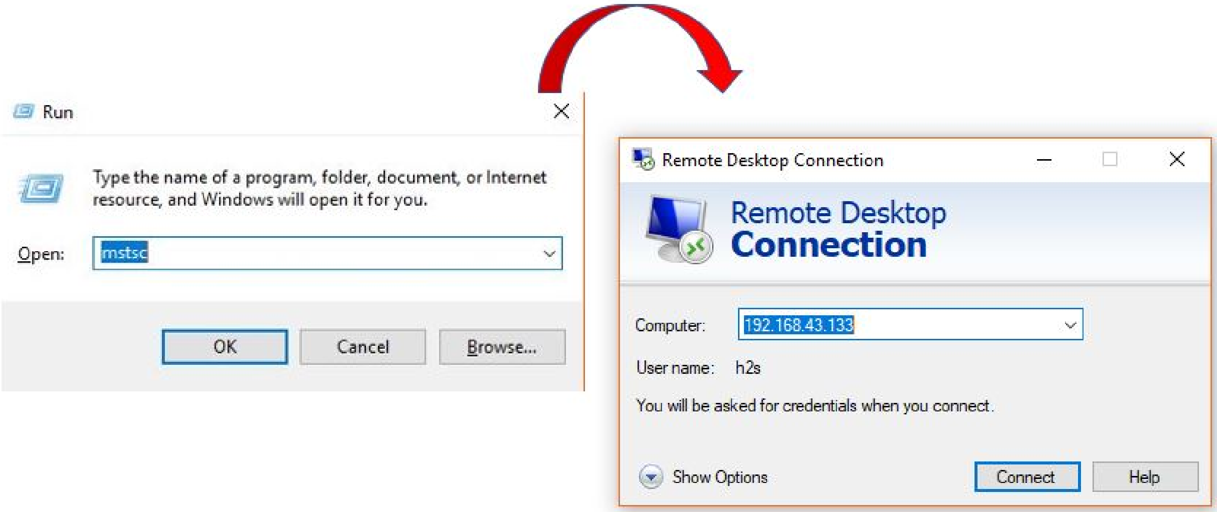
Login to XRDP

Step 1

If you are a Windows user, you can use the Remote Desktop feature that comes with Windows. If you are a Mac user, you can download and use Microsoft Remote Desktop from the APP Store, and there is not much difference between the two. The next example is Windows remote desktop.

Step 2

Type in "**mstsc**" in Run (WIN+R) to open the Remote Desktop Connection, and input the IP address of Raspberry Pi, then click on "Connect".



Run
Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.
OK
Cancel
Browse...
Remote Desktop Connection
Remote Desktop
Connection
Show Options
Connect
Help

Open: mstsc

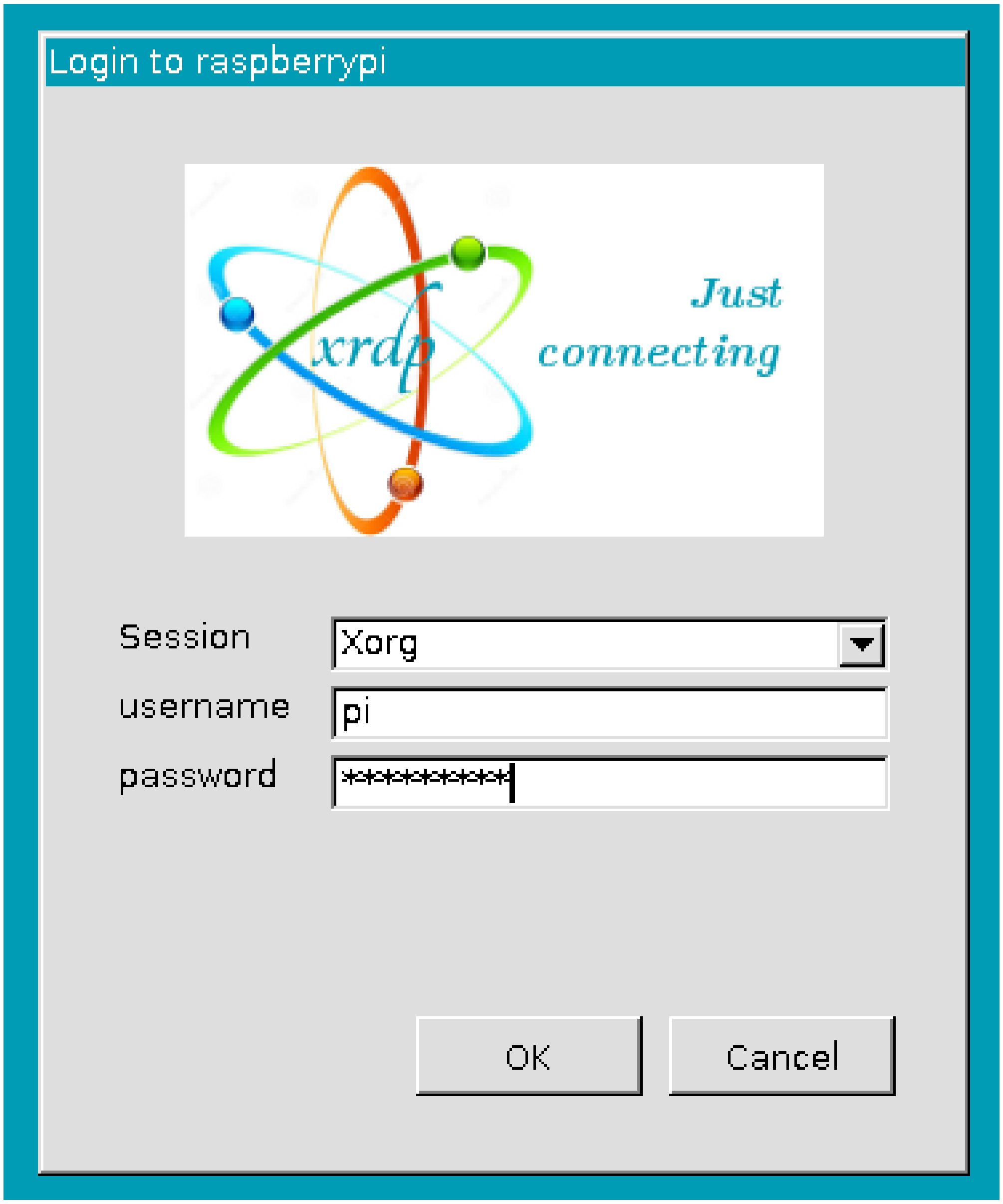
Computer: 192.168.43.133

User name: h2s

You will be asked for credentials when you connect.

Step 3

Then the xrdp login page pops out. Please type in your username and password. After that, please click "OK". At the first time you log in, your username is "pi" and the password is "raspberry".



Login to raspberrypi

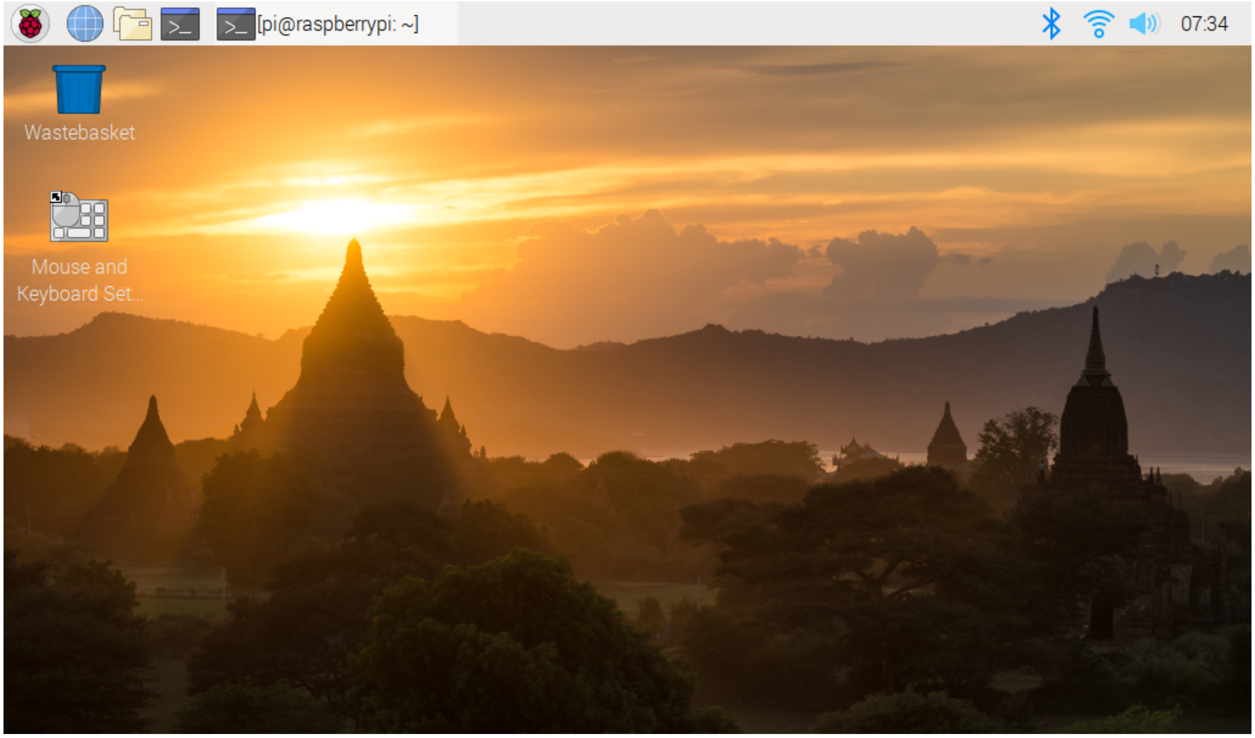
Session: Xorg

username: pi

password: *********

Step 4

Here, you successfully login to RPi by using the remote desktop.



Copyright Notice

1.9 Thank You

Thanks to the evaluators who evaluated our products, the veterans who provided suggestions for the tutorial, and the users who have been following and supporting us. Your valuable suggestions to us are our motivation to provide better products!

Particular Thanks

- Len Davisson
- Kalen Daniel
- Juan Delacosta

Now, could you spare a little time to fill out this questionnaire?

Note: After submitting the questionnaire, please go back to the top to view the results.

CHAPTER

TWO

COPYRIGHT NOTICE