Welcome

Thank you for choosing Freenove products!

Get Support and Offer Input

Freenove provides free and responsive product and technical support, including but not limited to:

- Product quality issues
- Product use and build issues
- Questions regarding the technology employed in our products for learning and education
- Your input and opinions are always welcome
- We also encourage your ideas and suggestions for new products and product improvements

For any of the above, you may send us an email to:

support@freenove.com

Safety and Precautions

Please follow the following safety precautions when using or storing this product:

- Keep this product out of the reach of children under 6 years old.
- This product should be used only when there is adult supervision present as young children lack necessary judgment regarding safety and the consequences of product misuse.
- This product contains small parts and parts, which are sharp. This product contains electrically conductive parts. Use caution with electrically conductive parts near or around power supplies, batteries and powered (live) circuits.
- When the product is turned ON, activated or tested, some parts will move or rotate. To avoid injuries to hands and fingers, keep them away from any moving parts!
- It is possible that an improperly connected or shorted circuit may cause overheating. Should this happen, immediately disconnect the power supply or remove the batteries and do not touch anything until it cools down! When everything is safe and cool, review the product tutorial to identify the cause.
- Only operate the product in accordance with the instructions and guidelines of this tutorial, otherwise parts may be damaged or you could be injured.
- Store the product in a cool dry place and avoid exposing the product to direct sunlight.
- After use, always turn the power OFF and remove or unplug the batteries before storing.

About Freenove

Freenove provides open source electronic products and services worldwide.

Freenove is committed to assist customers in their education of robotics, programming and electronic circuits so that they may transform their creative ideas into prototypes and new and innovative products. To this end, our services include but are not limited to:

- Educational and Entertaining Project Kits for Robots, Smart Cars and Drones
- Educational Kits to Learn Robotic Software Systems for Arduino, Raspberry Pi and micro: bit
- Electronic Component Assortments, Electronic Modules and Specialized Tools
- **Product Development and Customization Services**

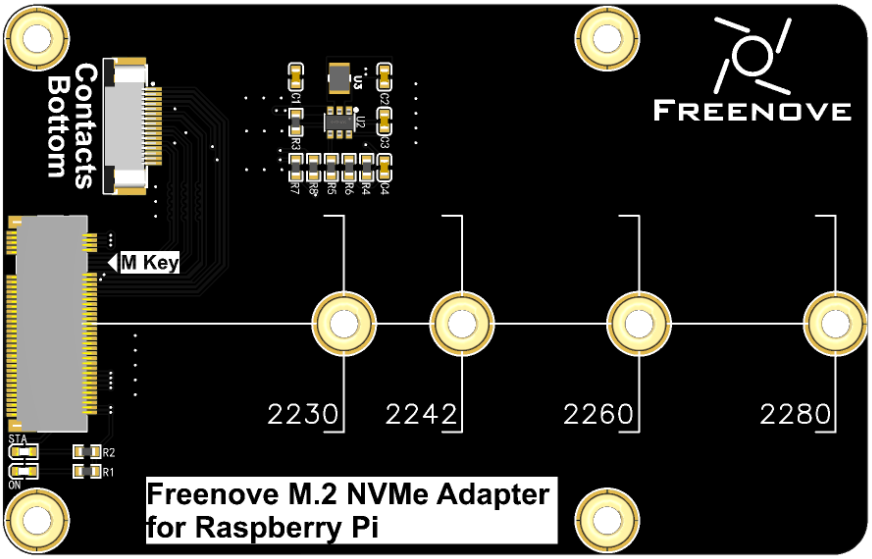
You can find more about Freenove and get our latest news and updates through our website:

http://www.freenove.com

Copyright

Contents

Preface



The Freenove M.2 NVMe Adapter for Raspberry Pi is a solid-state drives adapter designed specifically for the Raspberry Pi 5. Here are its key features:

- **Interface Type**: M.2 with M-Key
- **Supported Protocol**: NVMe
- **Compatible Sizes**: 2230, 2242, 2260, 2280
- **Power Supply**: 3.3V, up to 3A (maximum)
- **Indicator Lights**: Includes both power and SSD status LEDs.

The Raspberry Pi 5 includes a PCIe x1 slot that is certified for PCIe Gen 2.0, providng a theoretical maximum throughput of 5GT/sec, whichroughly translates to 500MB/sec for read and write operations. Although this slot is not officially certified for PCIe Gen 3.0, it is possible to force the use of Gen 3.0 for potentially higher speeds.

The PCIe consortium states that the speed of PCIe Gen 3.0 x1 is up to 8GT/sec, which translates to approximately 985MB/sec; however, Raspberry Pi claims that their implementation can achieve a speed of 10GT/sec, equivalent to around 1231MB/sec.
https://en.wikipedia.org/wiki/PCI_Express#Comparison_table
https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#pcie-gen-3-0

SSDs generally provide significantly faster read and write speeds compared to SD cards and USB drives, which can notably elevate the user experience when operating the Raspberry Pi 5.

Welcome to the Freenove M.2 NVMe Adapter for Raspberry Pi. This guide will walk you through the steps to effectively integrate and utilize this adapter on your Raspberry Pi 5.
Additionally, if you encounter any issues or have questions about this tutorial or the contents of kit, you can

always contact us for free technical support at:

support@freenove.com

Overview

To boot Raspberry Pi 5 from an NVMe SSD, two core tasks need to be accomplished:

1. **Flash the Raspberry Pi's operating system image onto the NVMe SSD.**
2. Configure the boot order of Raspberry Pi to give priority to SSD. (Optional)

Analysis

1. First of all, make sure you can enter the Raspberry Pi os via SD card or U drive.
2. After booting the Raspberry Pi, you can use it to flash the OS image directly onto the NVMe SSD. Alternatively, you can purchase an NVMe SSD to USB adapter and flash the image using USB on Windows or macOS, much like you would for an SD card or USB drive.
3. With this analysis in mind, we can systematically carry out the necessary steps.

Raspberry Pi models manufactured at different times might not boot up in the same way as described, but that's okay; just follow our guide to proceed.

There are various ways to burn the Raspberry Pi OS to SSD, each requiring different hardware tools.

Ways	Ways of burning	Requirements
1	Use Raspberry Pi to burn the OS. This requires that you can successfully boot up the Raspberry Pi via SD card or U disk. (**Recommended, described in this tutorial**)	An SD card or a U disk that can access the Raspberry Pi OS.
2	Purchase an NVMe SSD to USB adapter and flash the image just like you would with an SD card or USB drive.	NVME SSD to USB adapter (need to be bought separately)
3	If there are spare M.2 NVME interface on the motherboard of your PC, you can insert the SSD to it to flash the OS.	PC with M.2 NVME interface

Caution: Incompatible SSDs

The recognition and reading/writing of the NVME SSD are handled by the drivers of the Raspberry Pi 5. The Freenove M.2 NVMe Adapter for Raspberry Pi (hereinafter referred to as the Adapter) serves to connect and power the SSD. If you find that your SSD is not recognized or readable/writeable by the Pi 5, please try to find the drivers suitable for your SSD and install them on the Raspberry Pi, or replace the SSD, or purchase our Adapter kit that comes with an SSD.

Currently, some SSDs are found incompatible, as shown below:
https://docs.pineberrypi.com/nvme-compatibility-list

Not compatible drives

WD SN530 has been problematic for some users when it comes to NVMe boot, especially in the 2242 variant. Our test drive was a 2280 SKU and it worked OK but we generally recommend staying away from Western Digital drives completely.

- WD SN740
- WD SN350
- WD SN770
- WD SN850 possibly
- Transcend TSXXXGMTE300S (submitted by our community)

However, with the update of Raspberry Pi drivers, the incompatible SSDs listed above may become compatible one day.

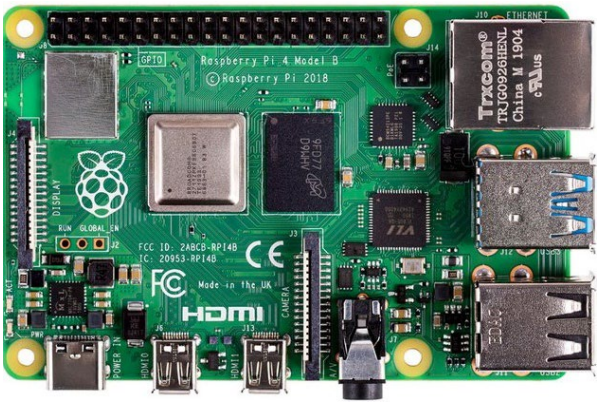
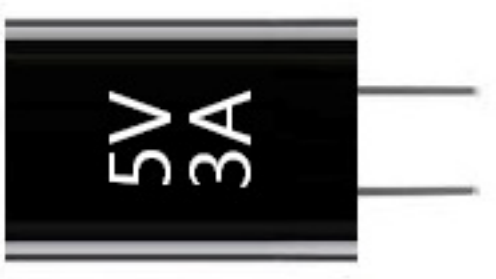
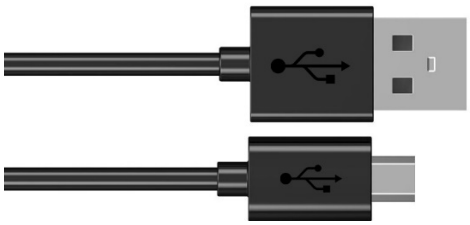
Video Tutorial: https://youtu.be/JTJghC_3c04

1.Flashing OS to SD Card or USB Drive

Based on the analysis above, our first step should be to install the Raspberry Pi operating system onto an SD card or USB drive, with a capacity of at least 16GB. If you are already able to boot the Raspberry Pi using an SD card or USB drive, you can skip this section and move on to the next chapter.

Component List

Required Components (self-prepared)

Raspberry Pi 5 	5V/3A Power Adapter. Note: Different versions of Raspberry Pi have different power requirements (please check the power requirements for yours on the chart in the following page.) 
Type-C USB Cable x1 	Micro SD Card (TF Card) x1, Card Reader x1 

Power requirements of various versions of Raspberry Pi are shown in following table:

Product	Recommended PSU current capacity	Maximum total USB peripheral current draw	Typical bare-board active current consumption
Raspberry Pi Model A	700mA	500mA	200mA
Raspberry Pi Model B	1.2A	500mA	500mA
Raspberry Pi Model A+	700mA	500mA	180mA
Raspberry Pi Model B+	1.8A	600mA/1.2A (switchable)	330mA
Raspberry Pi 2 Model B	1.8A	600mA/1.2A (switchable)	350mA
Raspberry Pi 3 Model B	2.5A	1.2A	400mA
Raspberry Pi 3 Model A+	2.5A	Limited by PSU, board, and connector ratings only.	350mA
Raspberry Pi 3 Model B+	2.5A	1.2A	500mA
Raspberry Pi 4 Model B	3.0A	1.2A	600mA
Raspberry Pi 5 Model B	5.0A	1.6A (600mA if using a 3A power supply)	800mA
Raspberry Pi Zero W	1.2A	Limited by PSU, board, and connector ratings only.	150mA
Raspberry Pi Zero	1.2A	Limited by PSU, board, and connector ratings only	100mA

For more details, please refer to
https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#power-supply

In addition, RPi also needs an Ethernet network cable used to connect it to a WAN (Wide Area Network).

All these components are necessary for any of your projects to work. Among them, the power supply of at least 5V/2.5A, because a lack of a sufficient power supply may lead to many functional issues and even damage your RPi, we STRONGLY RECOMMEND a 5V/2.5A power supply. We also recommend using a SD Micro Card with a capacity of 16GB or more (which, functions as the RPI's "hard drive") and is used to store the operating system and necessary operational files.

Optional Components

Under normal circumstances, there are two ways to login to Raspberry Pi: 1) Using a stand-alone monitor. 2) Using a remote desktop or laptop computer monitor "sharing" the PC monitor with your RPi.

Required Accessories for Monitor

If you choose to use an independent monitor, mouse and keyboard, you also need the following accessories:
1. A display with a HDMI interface
2. A Mouse and a Keyboard with an USB interface

As to Pi Zero and Pi Zero W, you also need the following accessories:
1. A Mini-HDMI to HDMI Adapter and Cable.
2. A Micro-USB to USB-A Adapter and Cable (Micro USB OTG Cable).
3. A USB HUB.
4. USB to Ethernet Interface or USB Wi-Fi receiver.

For different Raspberry Pi Modules, the optional items may vary slightly but they all aim to convert the interfaces to Raspberry Pi standards.

<table>
<tr><th></th><th>Pi Zero</th><th>Pi A+</th><th>Pi Zero W</th><th>Pi 3A+</th><th>Pi B+/2B</th><th>Pi 3B/3B+</th><th>Pi 4B/5B</th></tr>
<tr><td>Monitor</td><td colspan="7">Yes (All)</td></tr>
<tr><td>Mouse</td><td colspan="7">Yes (All)</td></tr>
<tr><td>Keyboard</td><td colspan="7">Yes (All)</td></tr>
<tr><td>Micro-HDMI to HDMI Adapter & Cable</td><td>Yes</td><td>No</td><td>Yes</td><td>No</td><td>No</td><td>No</td><td>No</td></tr>
<tr><td>Micro-HDMI to HDMI Adapter & Cable</td><td colspan="6">No</td><td>Yes</td></tr>
<tr><td>Micro-USB to USB-A Adapter & Cable (Micro USB OTG Cable)</td><td>Yes</td><td>No</td><td>Yes</td><td colspan="4">No</td></tr>
<tr><td>USB HUB</td><td>Yes</td><td>Yes</td><td>Yes</td><td>Yes</td><td>No</td><td>No</td><td></td></tr>
<tr><td>USB to Ethernet Interface</td><td colspan="2" rowspan="2">select one from two or select two from two</td><td colspan="2" rowspan="2">optional</td><td rowspan="2">Internal Integration</td><td colspan="2" rowspan="2">Internal Integration</td></tr>
<tr><td>USB Wi-Fi Receiver</td></tr>
<tr><td></td><td colspan="2"></td><td colspan="2">Internal Integration</td><td>optional</td><td colspan="2"></td></tr>
</table>

Required Accessories for Remote Desktop

If you do not have an independent monitor, or if you want to use a remote desktop, you first need to login to Raspberry Pi through SSH, and then open the VNC or RDP service. This requires the following accessories.

	Pi Zero	Pi Zero W	Pi A+	Pi 3A+	Pi B+/2B	Pi 3B/3B+/4B/5B
Micro-USB to USB-A Adapter & Cable (Micro USB OTG Cable)	Yes	Yes	No	NO		
USB to Ethernet interface	Yes	Yes	Yes			

Raspberry Pi OS

Without Screen - Use Raspberry Pi - under Windows PC: https://youtu.be/XpiT_ezb_7c
With Screen - Use Raspberry Pi - under Windows PC: https://youtu.be/HEywFsFrj3I

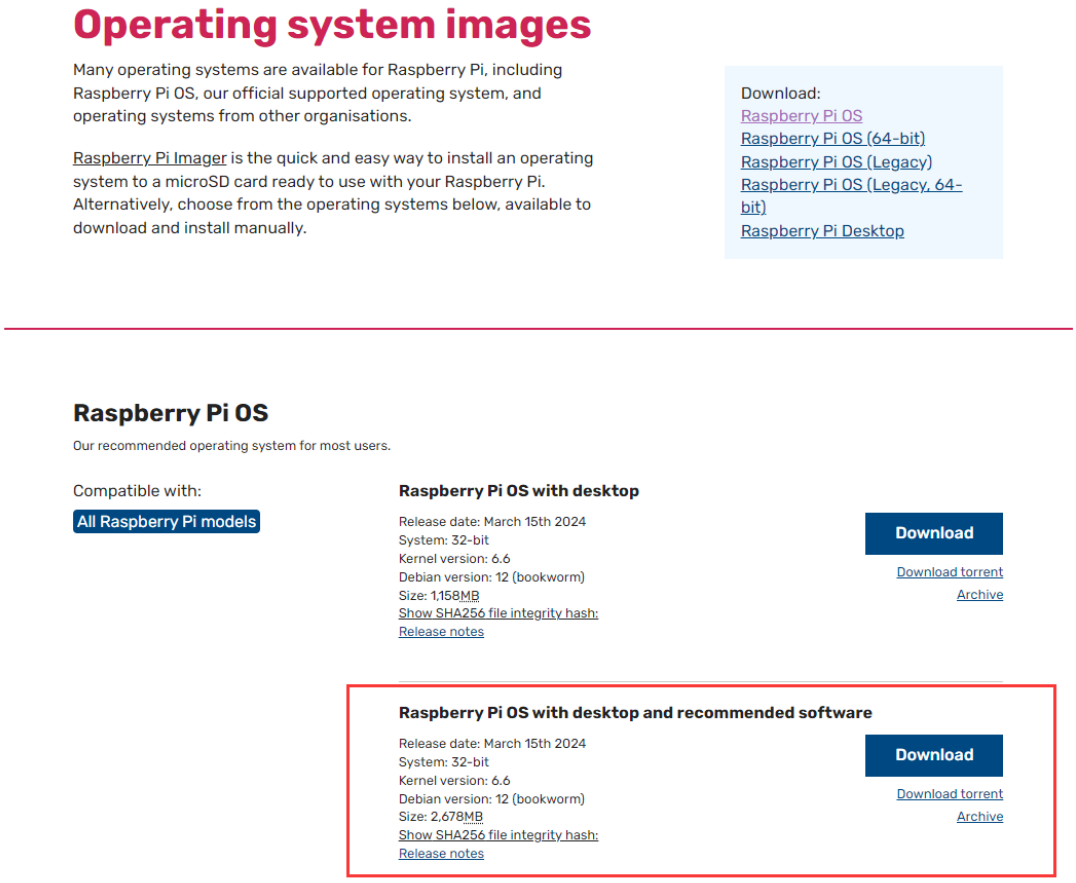
Automatically Method

You can follow the official method to install the system for raspberry pi via visiting link below:
https://projects.raspberrypi.org/en/projects/raspberry-pi-setting-up/2
In this way, the system will be downloaded **automatically** via the application.

Manually Method

After installing the Imager Tool in the **link above**. You can **also** download the system **manually** first.

Visit https://www.raspberrypi.com/software/operating-systems/



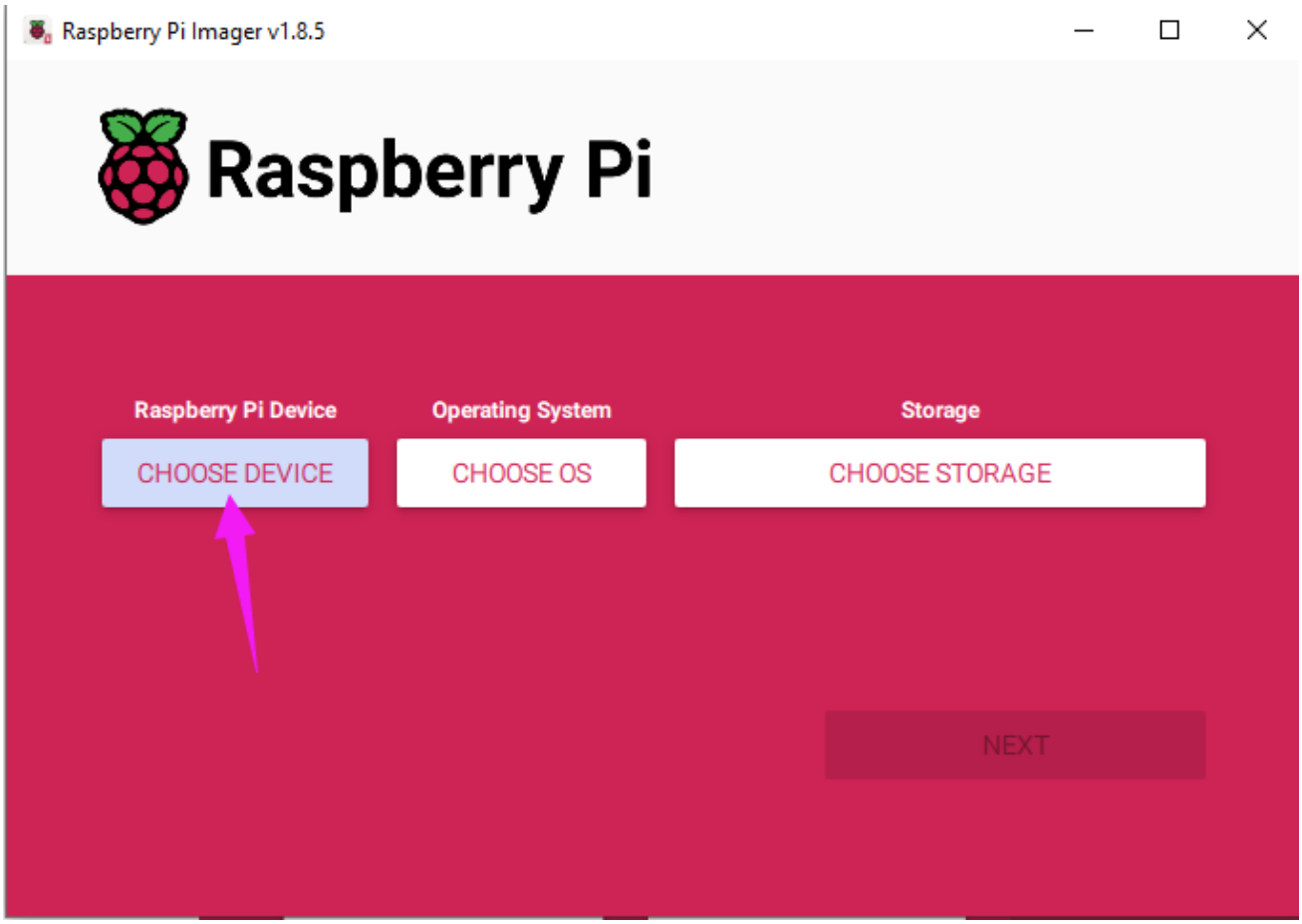
And then the zip file is downloaded.

Write System to Micro SD Card

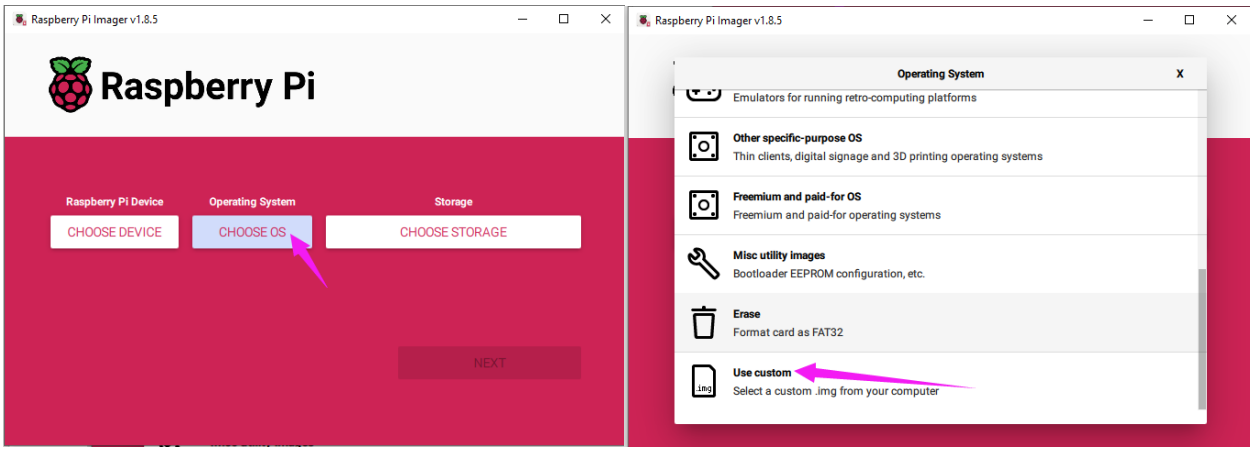
First, put your Micro **SD card** into card reader and connect it to USB port of PC.

Open Raspberry Pi Imager.

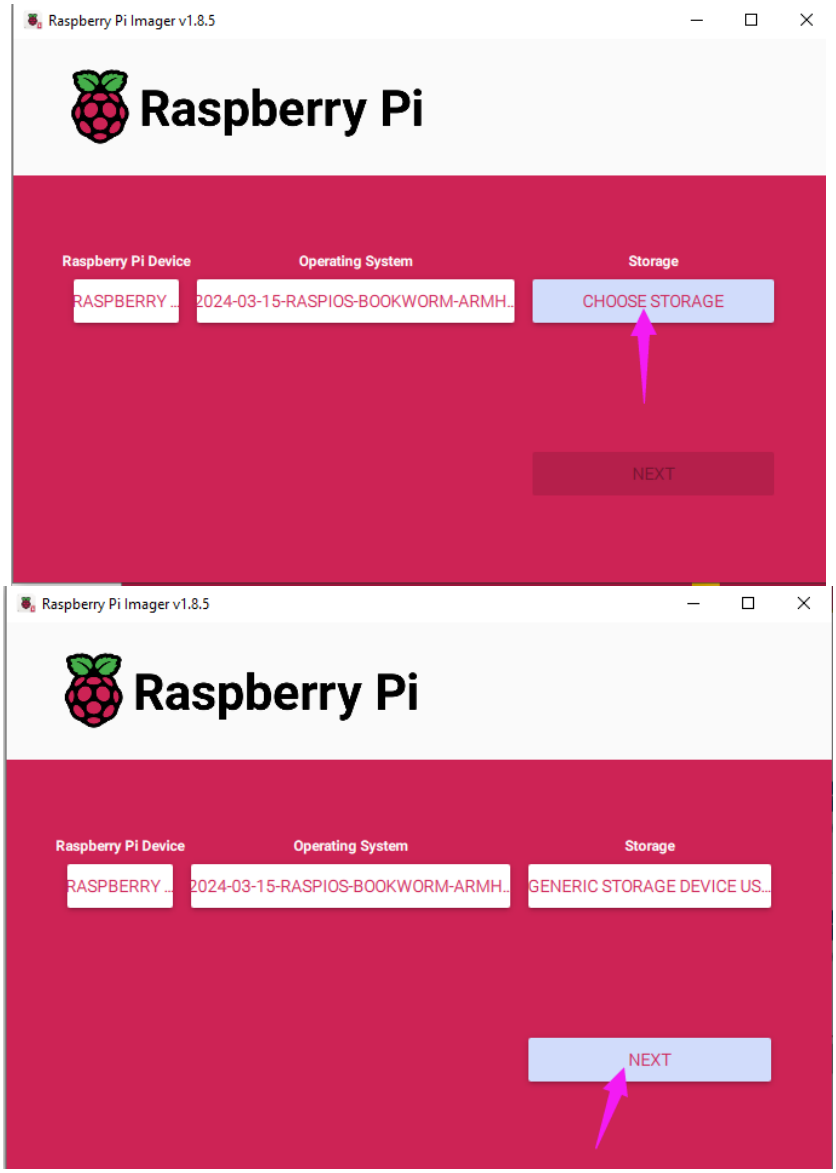
Choose Raspberry Pi 5 as the device.



Choose the system that you just downloaded in Use custom.

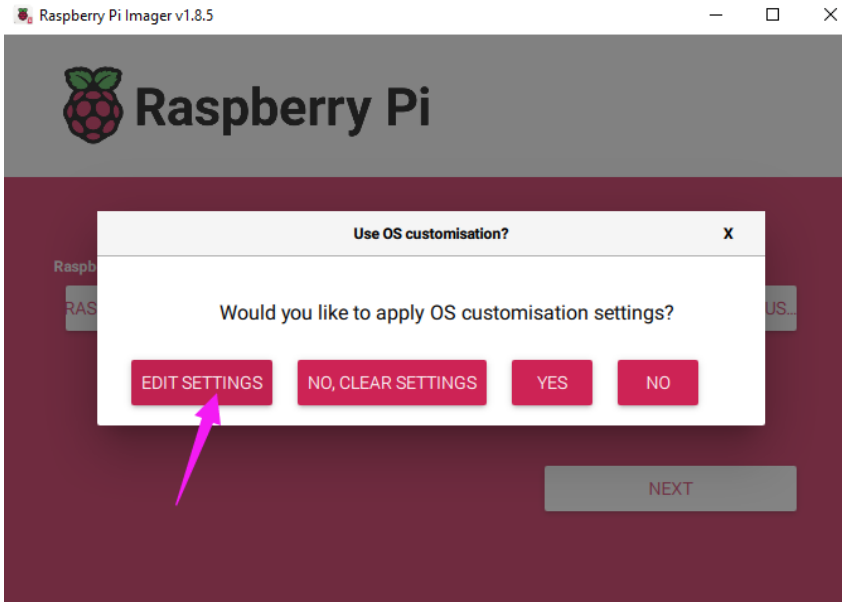


Choose the SD card and click on Next.

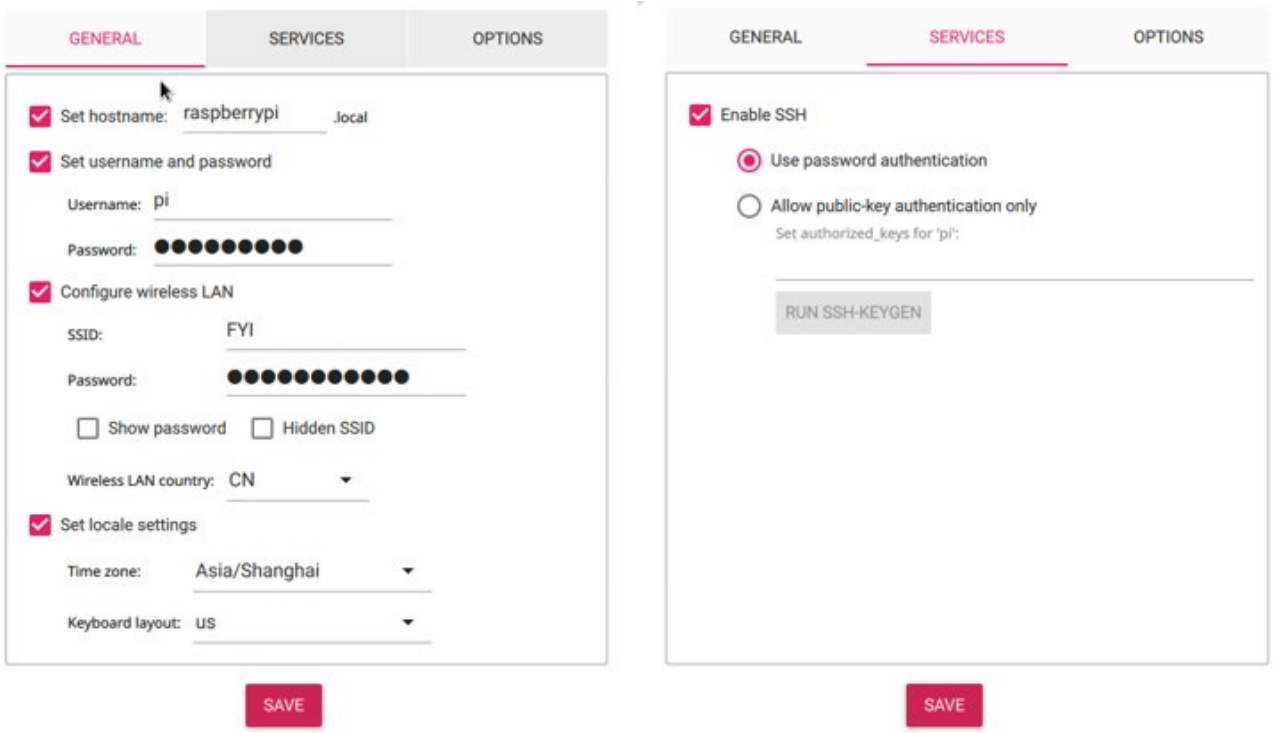


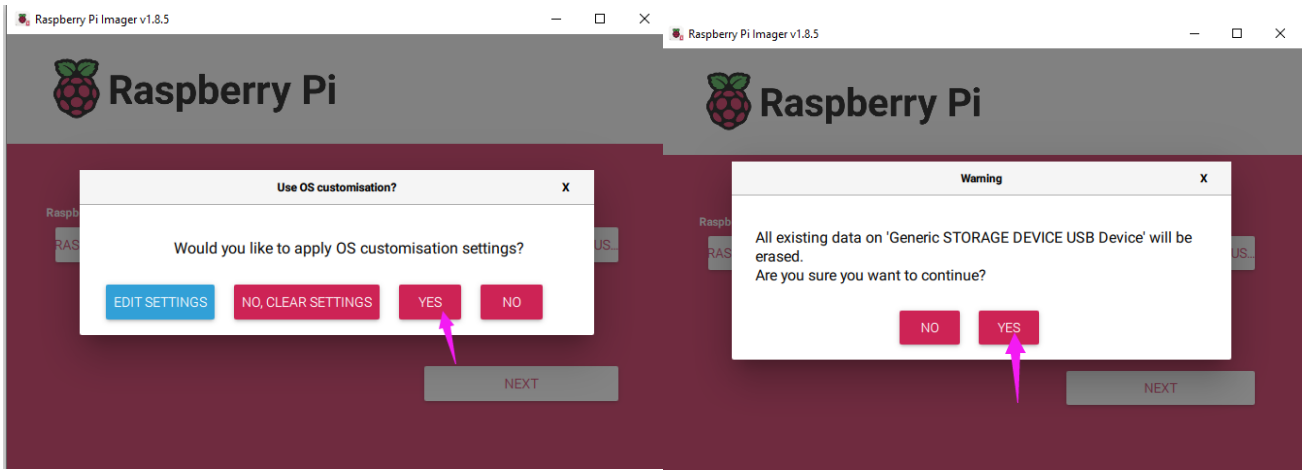
Enable ssh and configure WiFi

Click EDIT SETTINGS.



Configure wireless LAN, enable SSH and click Save.

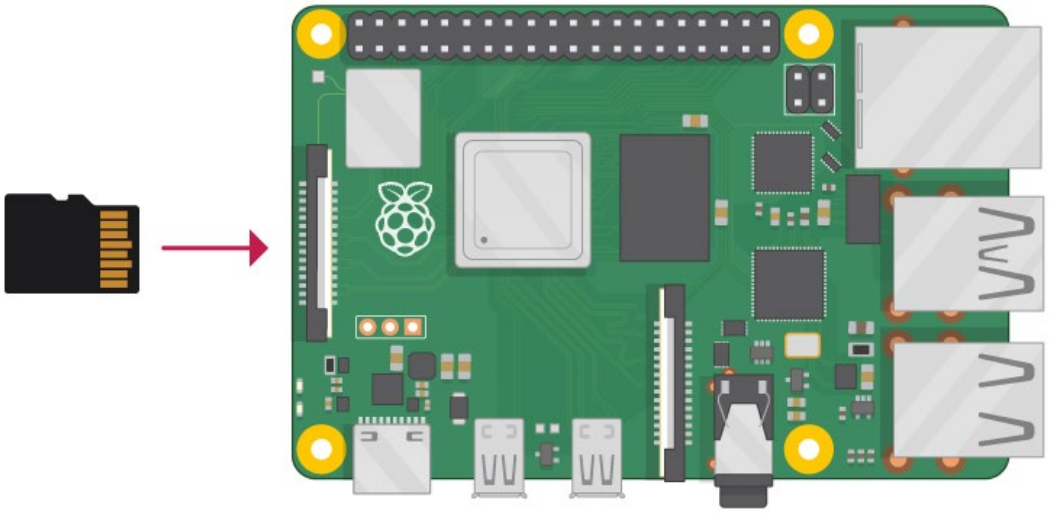




Wait for it to finish writing and verifying.

Insert SD card

Then remove SD card from card reader and insert it into Raspberry Pi.



Getting Started with Raspberry Pi

Monitor desktop

If you do not have a spare monitor, please skip to next section Remote desktop & VNC. 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: raspberry. After login, you should see the following screen.

Congratulations! You have successfully installed the RASPBERRY PI OS operating system on your RPi.

Raspberry Pi 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.

Remote desktop & VNC

If you have logged in Raspberry Pi via display, you can skip to VNC Viewer.

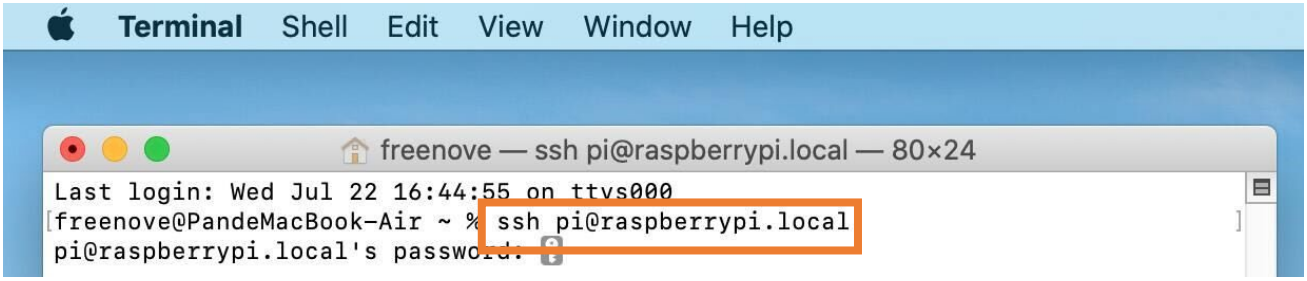
If you don't have a spare display, mouse and keyboard for your RPi, you can use a remote desktop to share a display, keyboard, and mouse with your PC. Below is how to use:
MAC OS remote desktop and Windows OS remote desktop.

MAC OS Remote Desktop

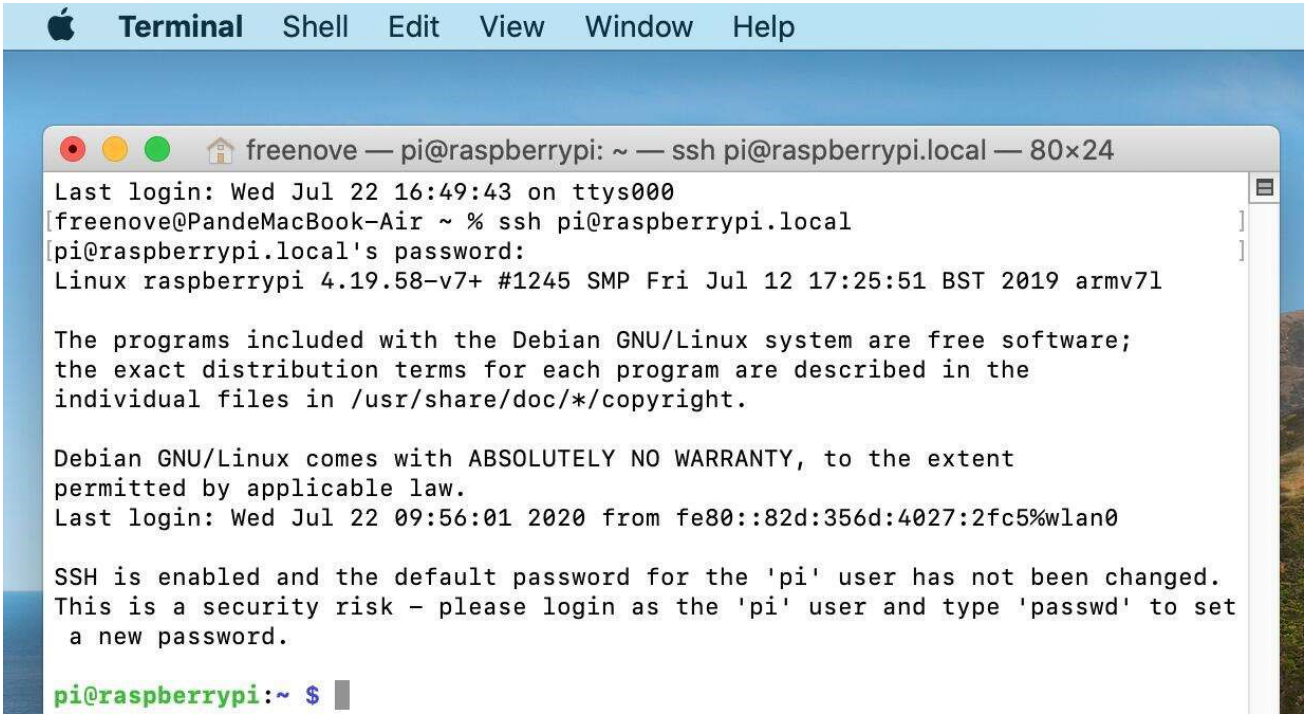
Open the terminal and type following command. **If this command doesn't work, please move to next page.**

```
ssh pi@raspberrypi.local
```

The password is **raspberry** by default, case sensitive.



You may need to type **yes** during the process.



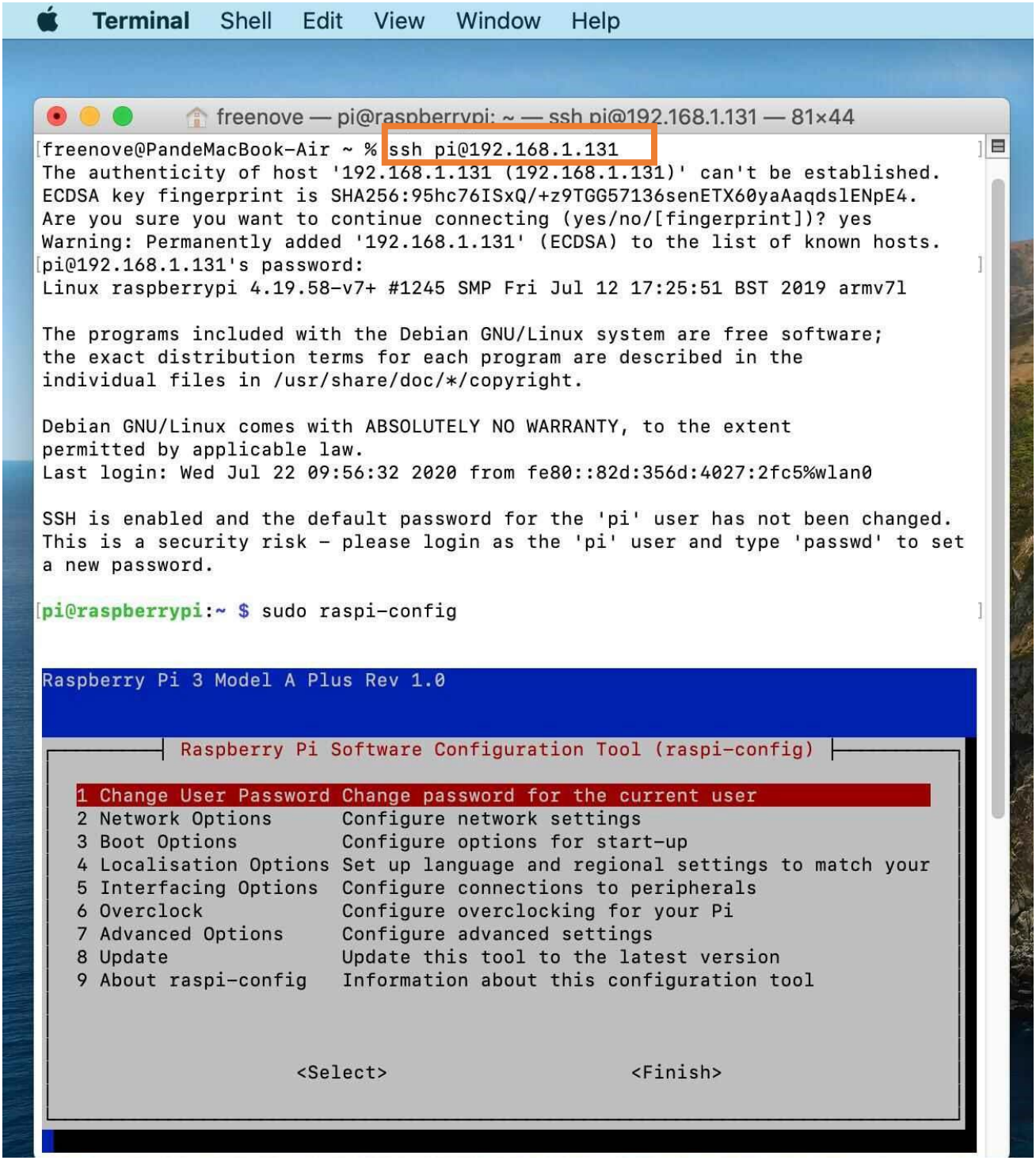
You can also use the IP address to log in Pi.

Enter **router** client to **inquiry IP address** named "raspberry pi". For example, I have inquired to **my RPi IP address, and it is "192.168.1.131".**

Open the terminal and type following command.

```
ssh pi@192.168.1.131
```

When you see pi@raspberrypi:~ $, you have logged in Pi successfully. Then you can skip to next section.



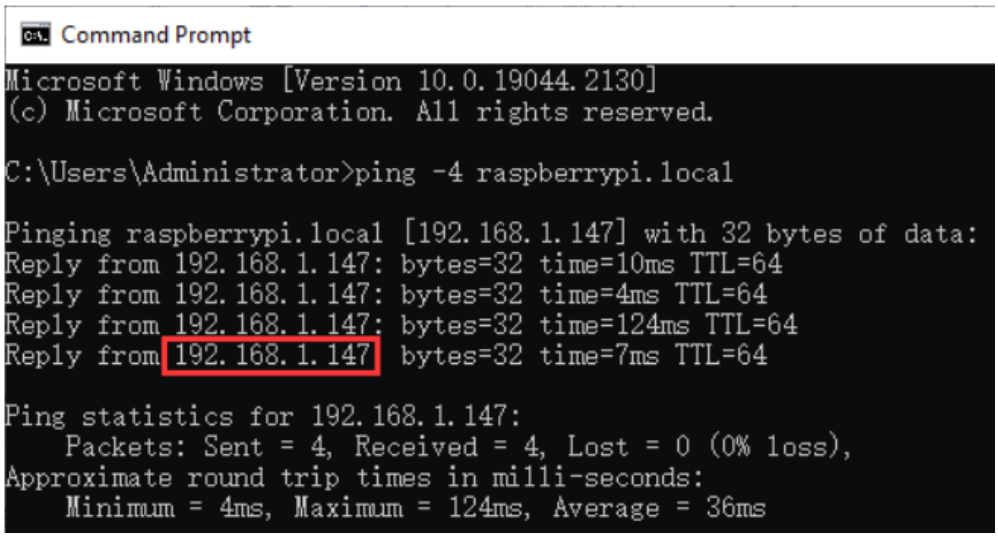
Then you can skip to VNC Viewer.

Windows OS Remote Desktop

If you are using win10, you can use follow way to login Raspberry Pi without desktop.

Press **Win+R**. Enter **cmd**. Then use this command to check IP:

```
ping -4 raspberrypi.local
```



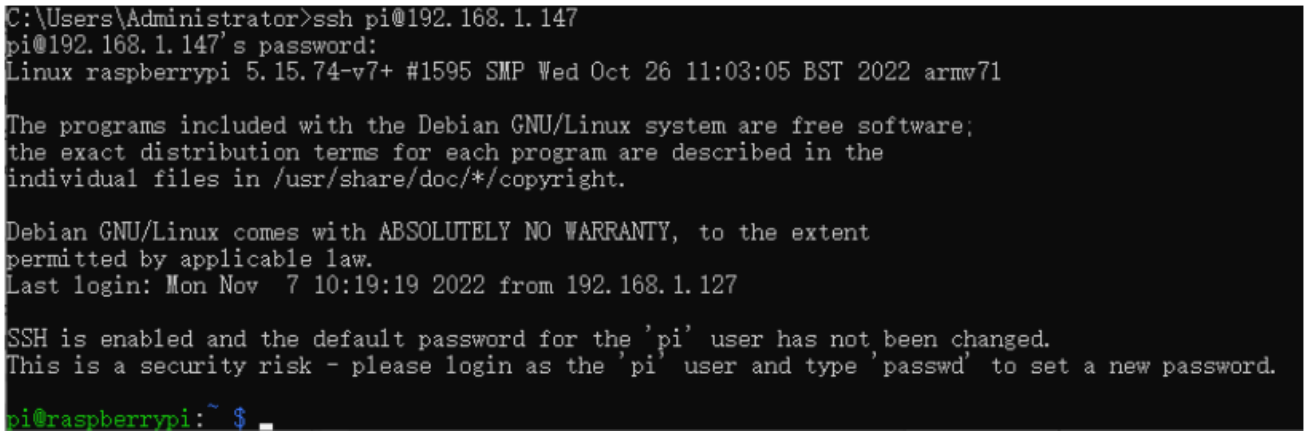
Then 192.168.1.147 is my Raspberry Pi IP.

Or enter **router** client to **inquiry IP address** named "**raspberrypi**". For example, I have inquired to **my RPi IP address, and it is "192.168.1.147".**

```
ssh pi@xxxxxxxxxxx(IP address)
```

Enter the following command:

```
ssh pi@192.168.1.147
```

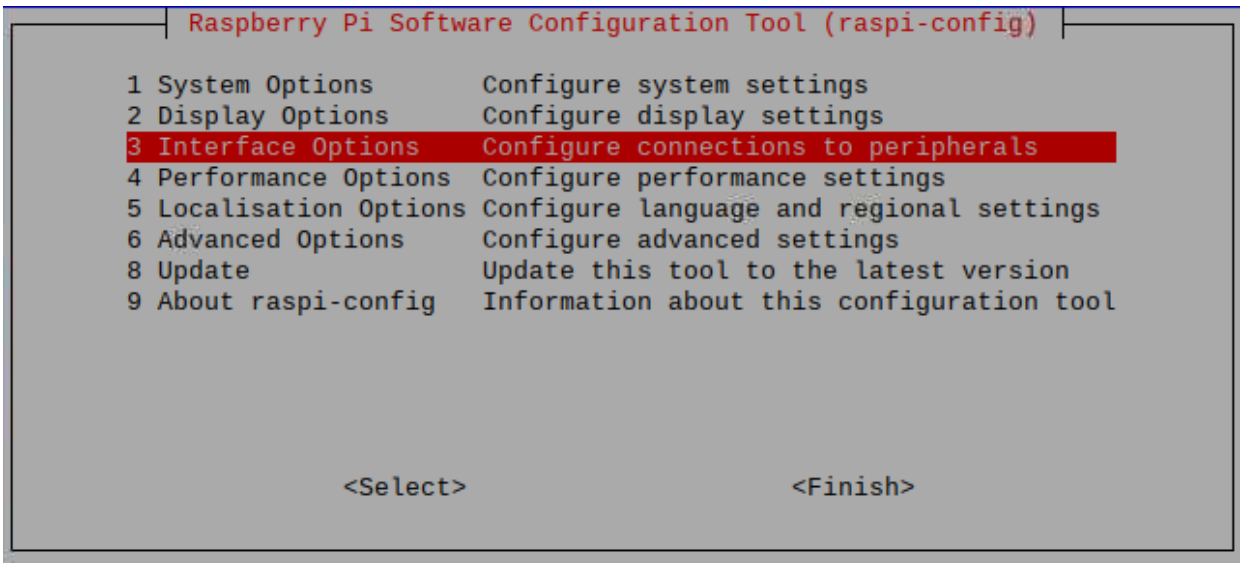


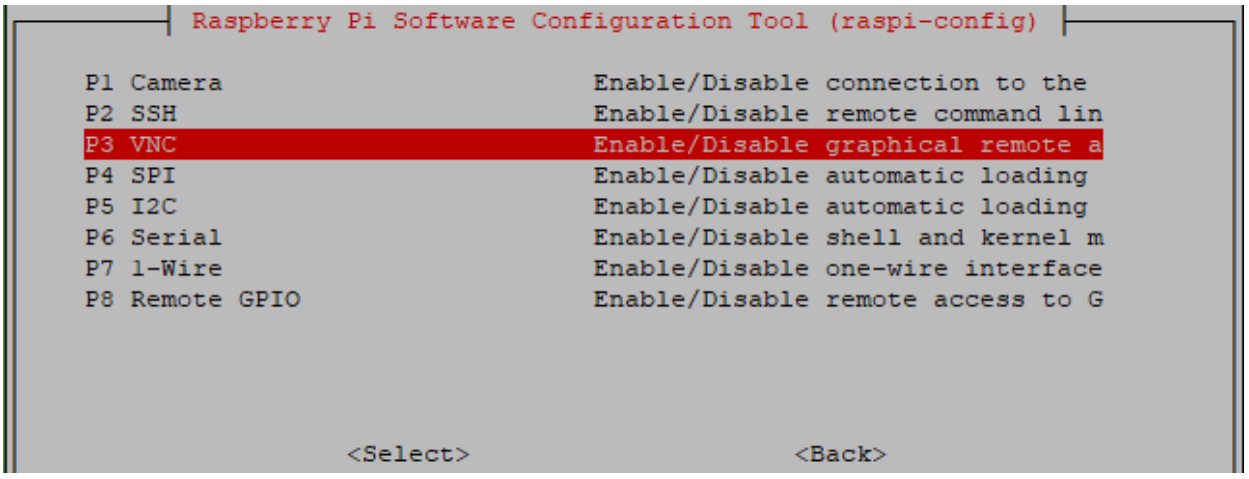
VNC Viewer & VNC

Enable VNC

Type the following command. And select Interface Options→P3 VNC → Enter→Yes→OK. Here Raspberry Pi may need be restarted, and choose ok. Then open VNC interface.

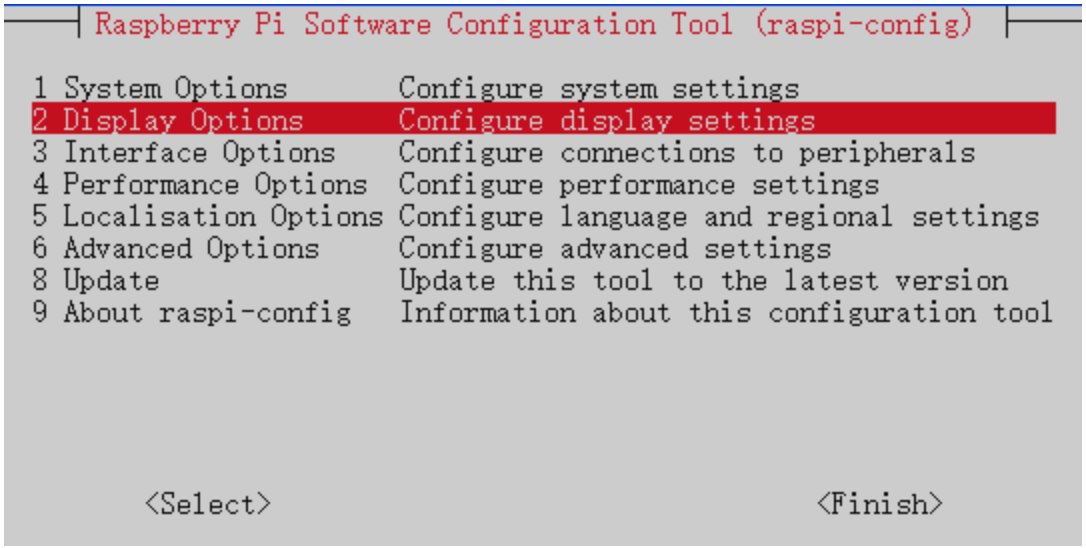
```
sudo raspi-config
```

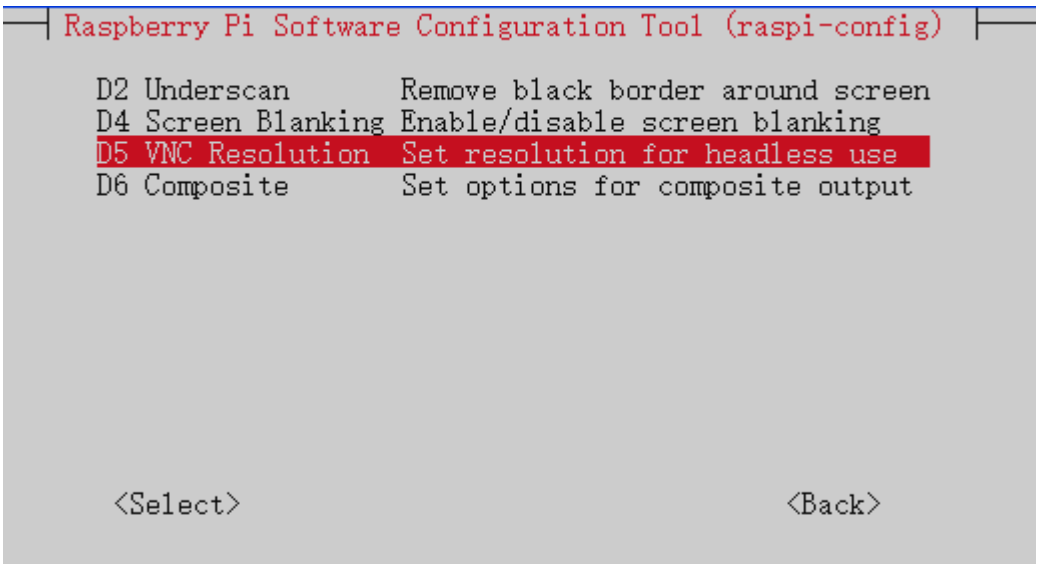


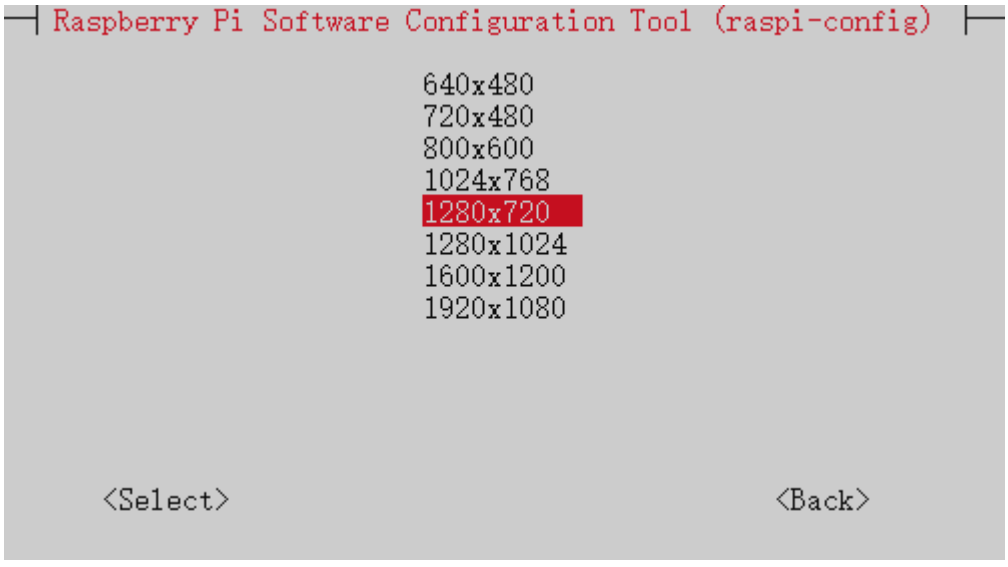


Set Resolution

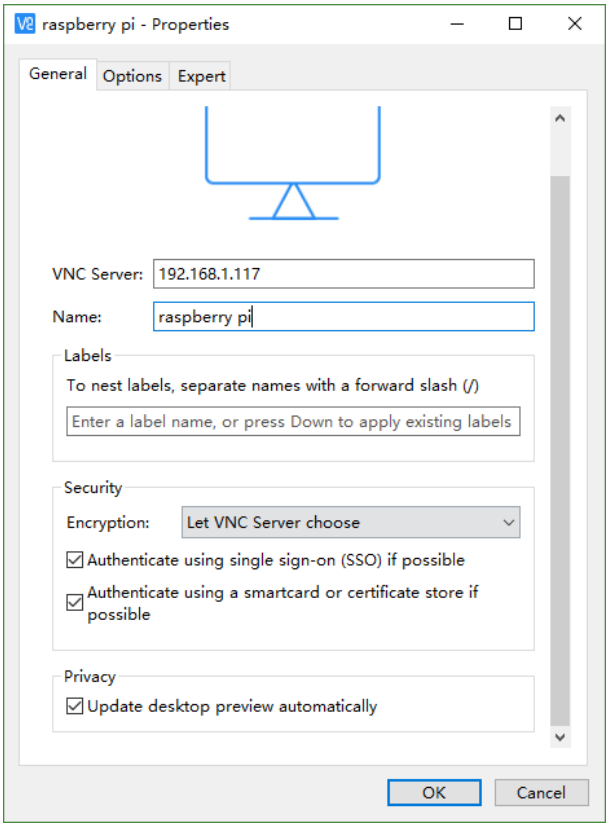
You can also set other resolutions. If you don't know what to set, you can set it as 1280x720 first.





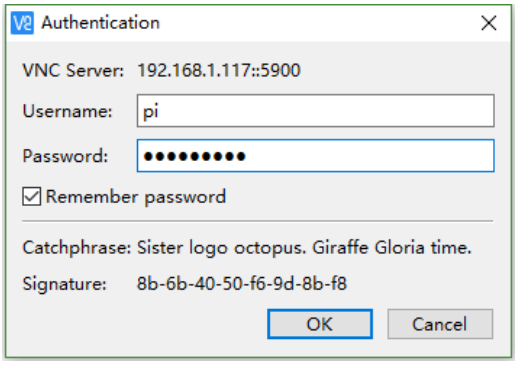


Then download and install VNC Viewer according to your computer system by click following link:
https://www.realvnc.com/en/connect/download/viewer/
After installation is completed, open VNC Viewer. And click File → New Connection. Then the interface is shown below.

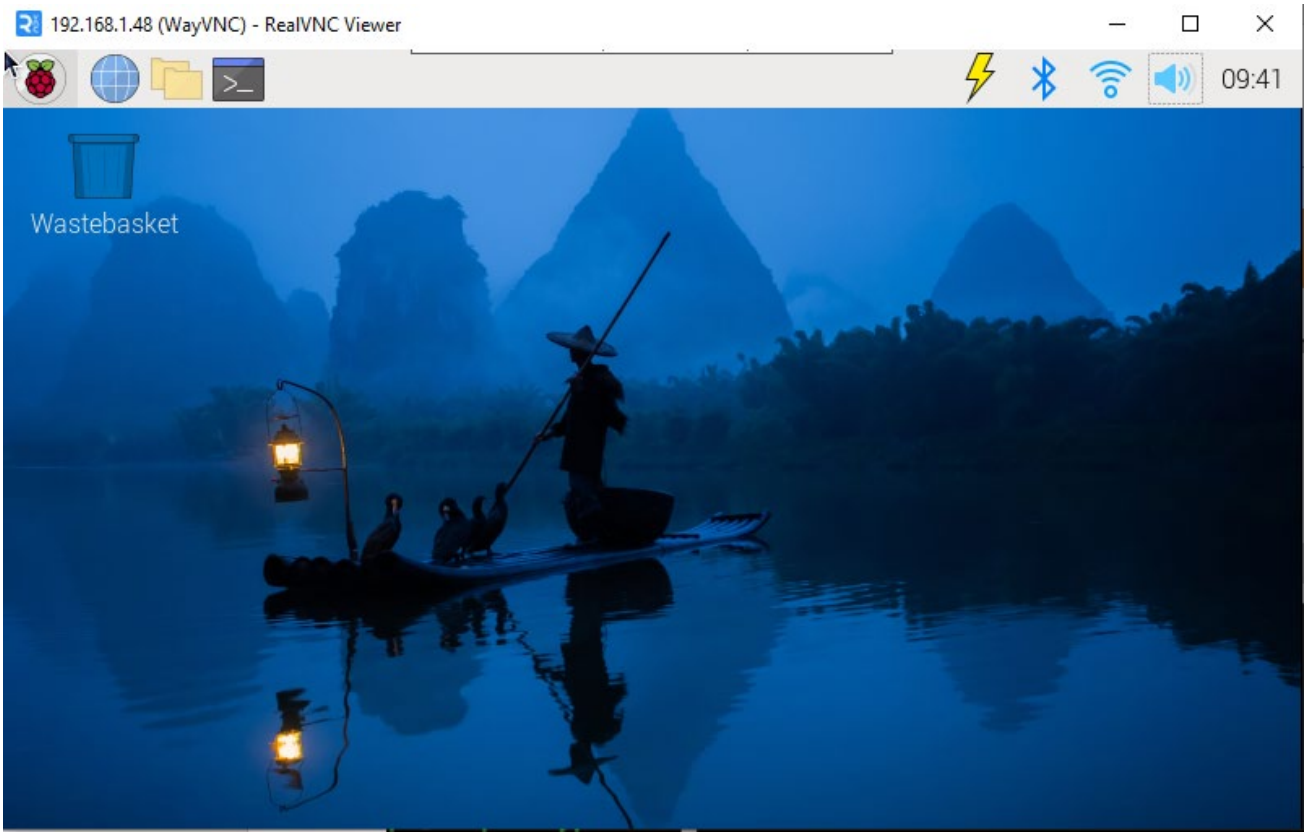


Enter ip address of your Raspberry Pi and fill in a name. Then click OK.
Then on the VNC Viewer panel, double-click new connection you just created,

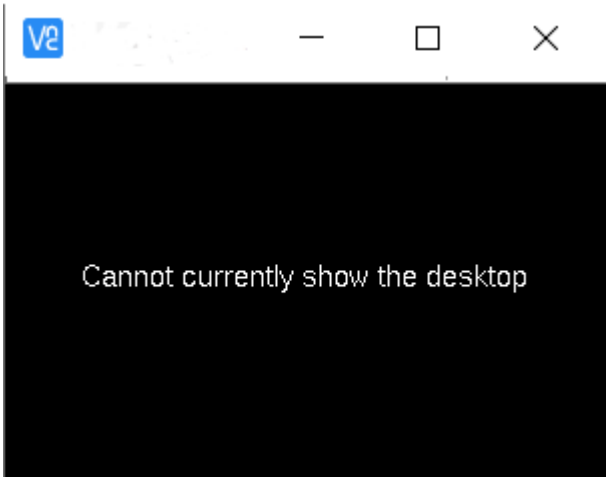
and the following dialog box pops up.



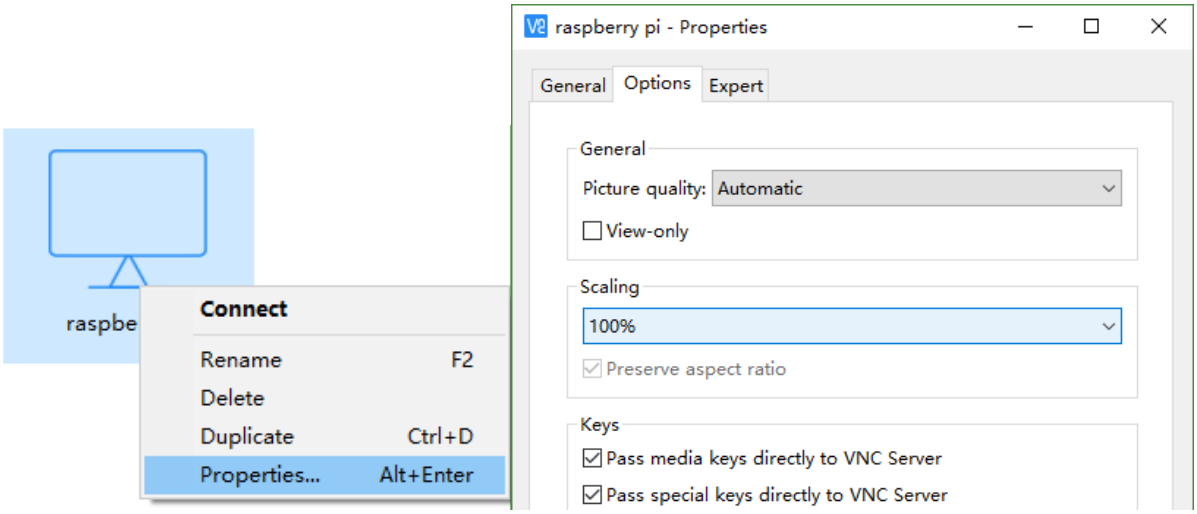
Enter username: **pi** and Password: **raspberry**. And click OK.



Here, you have logged in to Raspberry Pi successfully by using VNC Viewer.
If there is black window, please [set another resolution].



In addition, your VNC Viewer window may zoom your Raspberry Pi desktop. You can change it. On your VNC View control panel, click right key. And select Properties->Options label->Scaling. Then set proper scaling.



Here, you have logged in to Raspberry Pi successfully by using VNC Viewer and operated proper setting.

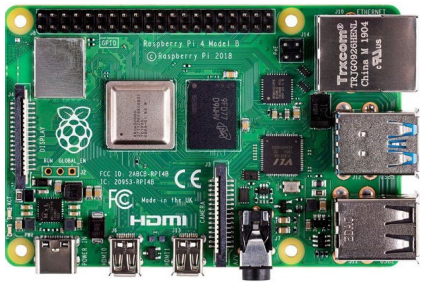
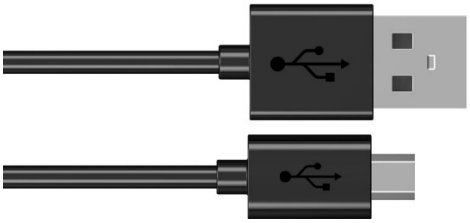
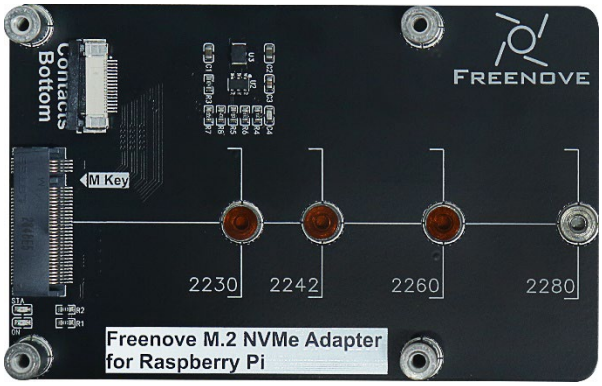
Raspberry Pi 4B/3B+/3B integrates a Wi-Fi adaptor.If you did not connect Pi to WiFi. You can connect it to wirelessly control the robot.

2.Flashing OS to NVME SSD

2.1 Assembly and Wiring

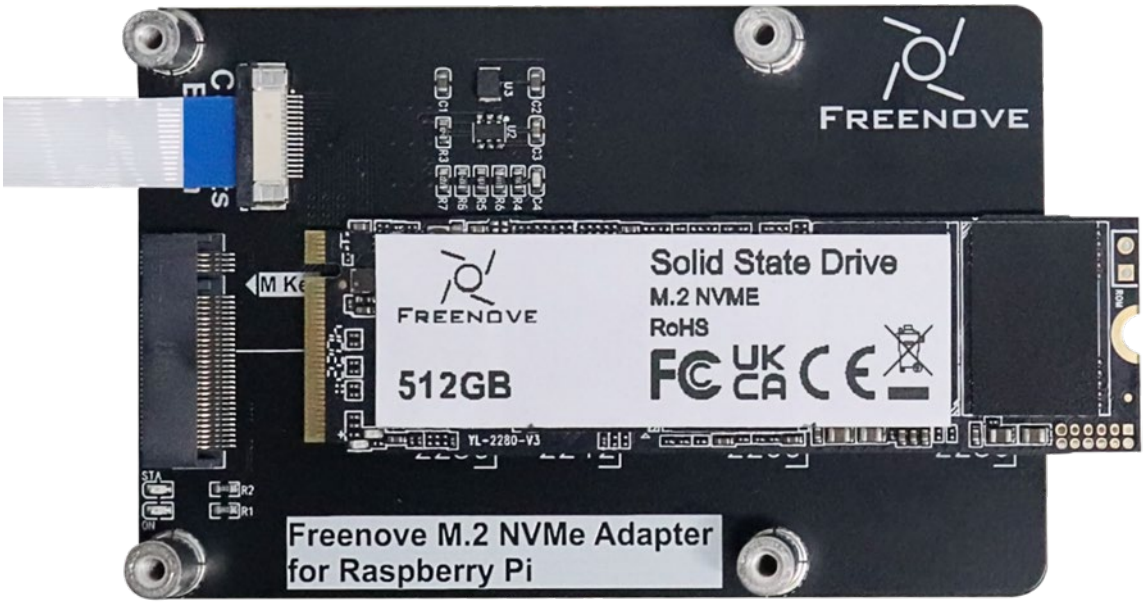
Assemble the Freenove M.2 Adapter for Raspberry Pi and NVME SSD to your Raspberry Pi.

Required Components

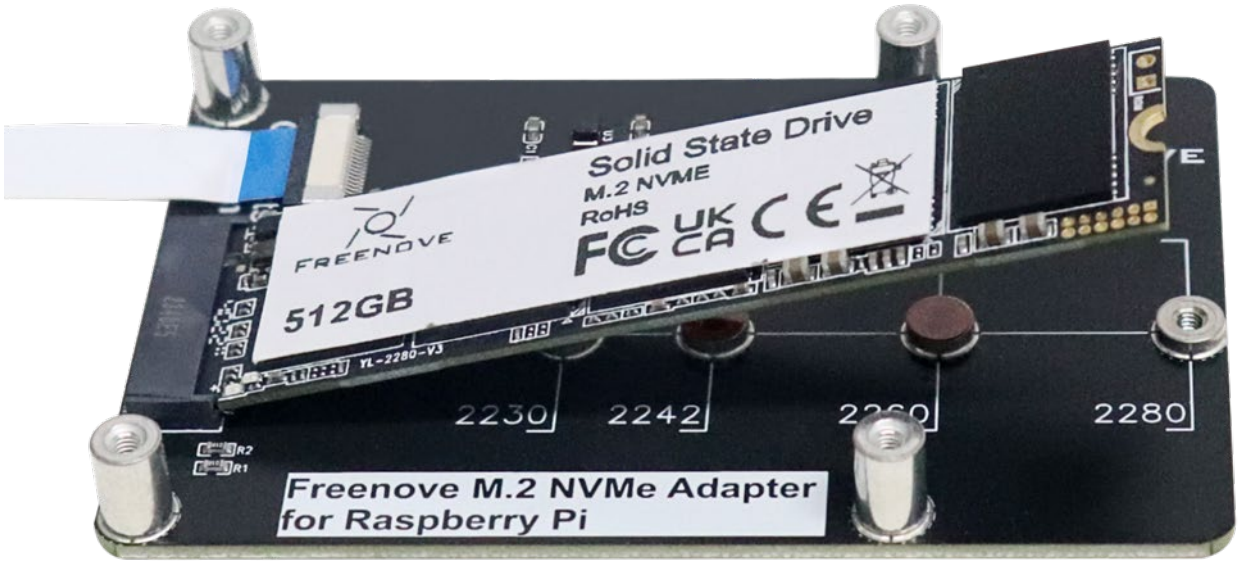
Raspberry Pi 5 	Power Adapter. 
Type-C USB Cable x1 	Micro SD Card (TF Card) x1, Card Reader x1 
Freenove M.2 Adapter for Raspberry Pi 	NVME SSD 
M2.5x3 Screws x5 	Cable 

Assembly and Wiring

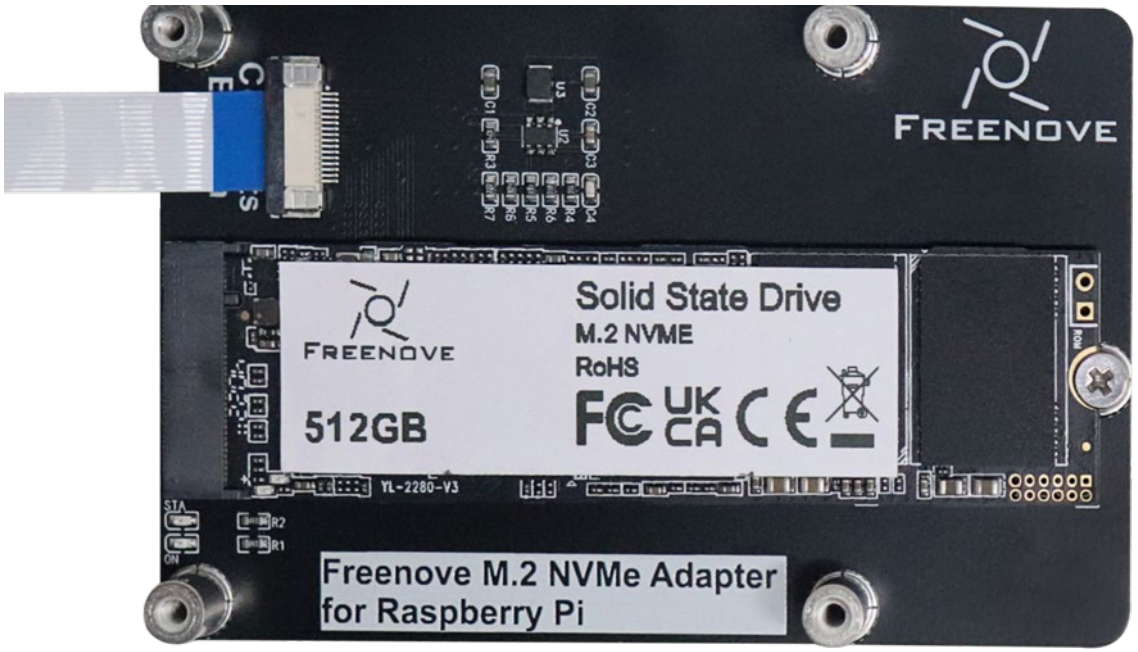
1. Connect the FPC cable to the FPC socket. Please note that the metal contacts of the ribbon cable should face downward, as shown in the figure below.



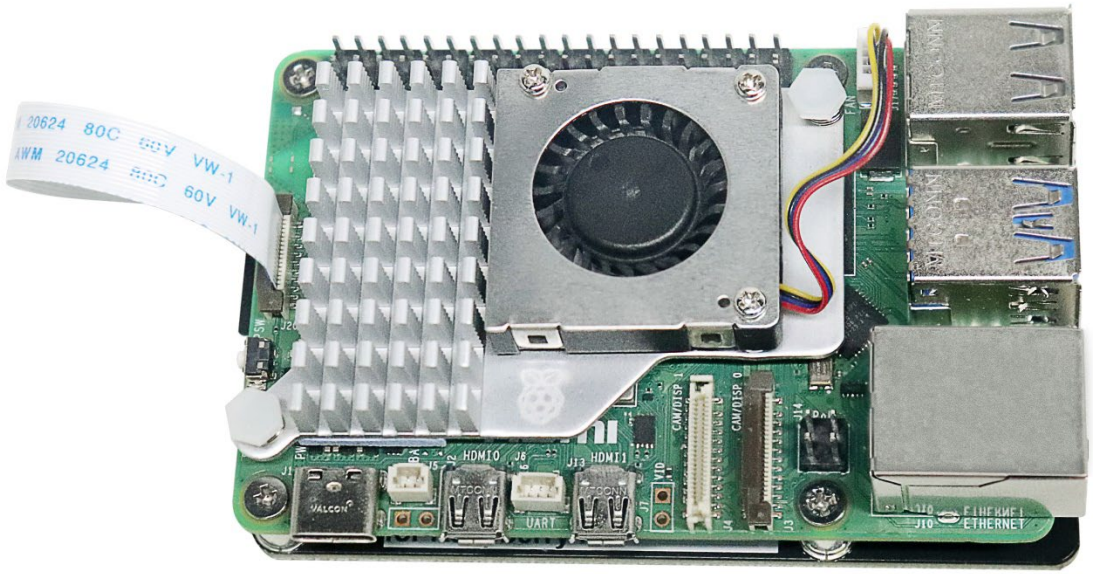
2. Insert the SSD to the adapter board. **Please tilt it to insert**.



3. Fix the SSD with a M2.5 screw..



4. Connect the other end of the cable to Raspberry Pi and fix the adapter to RPi with four M2.5 screws.



2.2 Flashing the RPi OS to NVME SSD

Once everything is set up, power on the Raspberry Pi and boot into the system. (In this case, we are using a brand new SSD with a 512GBits capacity that has not been partitioned yet.)

2.2.1 SSD Dectection

(Note: Not all SSDs are supported by Pi5.)

Run the following command in the Terminal to check whether SSD is detected.

Note that different SSDs display different content.

```
lspci
```

```
pi@raspberrypi:~ $ lspci
0000:00:00.0 PCI bridge: Broadcom Inc. and subsidiaries Device 2712 (rev 21)
0000:01:00.0 Non-Volatile memory controller: Silicon Motion, Inc. SM2263EN/SM2263XT SSD Controller (rev 03)
0001:00:00.0 PCI bridge: Broadcom Inc. and subsidiaries Device 2712 (rev 21)
0001:01:00.0 Ethernet controller: Device 1de4:0001
```

```
lsblk
```

```
pi@raspberrypi:~ $ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
mmcblk0     179:0    0  29.8G  0 disk
├─mmcblk0p1 179:1    0   512M  0 part /boot/firmware
└─mmcblk0p2 179:2    0  29.3G  0 part /
nvme0n1     259:0    0 476.9G  0 disk
pi@raspberrypi:~ $
```

As shown in the figure above, the device ‘nvme0n1’ with a capacity of 476.9GBytes shows up, indicating that the SSD has been correctly recognized. The detected capacity will depend on the size of your SSD. If your drive has been previously partitioned, you may also see some partition information displayed.

Please note: Installing the system will format the SSD, erasing all data. If necessary, please back up any data on your SSD before proceeding.

2.2.2 Enable PCIE3.0 (on OS written into SD card)

If the SSD you received is with **Phison** controller, you may need to enable PCIE 3.0 .(This step is strongly recommended; without this step, the later process may fail.)

If it is not with Phison controller, you do not need to enable PCIE 3.0. **You may skip this section**.

Run the command `lspci` to check the controller.

```
pi@raspberrypi:~ $ lspci
0000:00:00.0 PCI bridge: Broadcom Inc. and subsidiaries BCM2712 PCIe Bridge (rev 21)
0000:01:00.0 Non-Volatile memory controller: Phison Electronics Corporation PS5021-E21
 PCIe4 NVMe Controller (DRAM-less) (rev 01)
0001:00:00.0 PCI bridge: Broadcom Inc. and subsidiaries BCM2712 PCIe Bridge (rev 21)
0001:01:00.0 Ethernet controller: Raspberry Pi Ltd RP1 PCIe 2.0 South Bridge
```

(The above screenshot is the result of an 128GB SSD with Phison as main controller.)

Enable PCIe Gen3.0

Add the line `dtparam=pciex1_gen=3` to /boot/firmware/config.txt to enable PCIe Gen3.0.

As shown below, enter the command to open the file.

```
sudo nano /boot/firmware/config.txt
```

Add the line `dtparam=pciex1_gen=3` to the end of the file, as shown below:

```
  GNU nano 7.2                 /boot/firmware/config.txt *
otg_mode=1

[all]

dtparam=pciex1_gen=3
```

Press Ctrl-O to save the file, Enter to confirm, and Ctrl-X to exit.

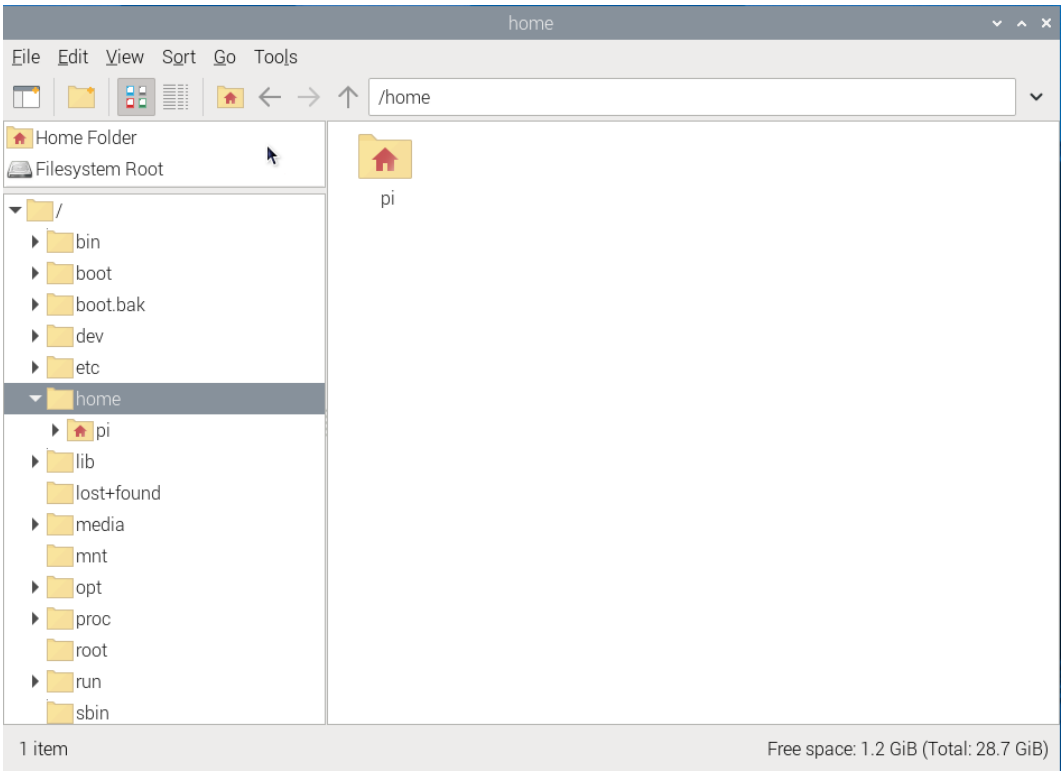
Reboot your Raspberry Pi.

```
sudo reboot
```

2.2.3 SSD Partitioning and Formatting

This step is not a must-do, but it can further test whether the SSD perform normally on Raspberry Pi to ensure smooth performance in later steps.

At this point, the hard drive cannot be seen in the file manager as the disk has not been partitioned yet.



Install a disk management tool with the following command:

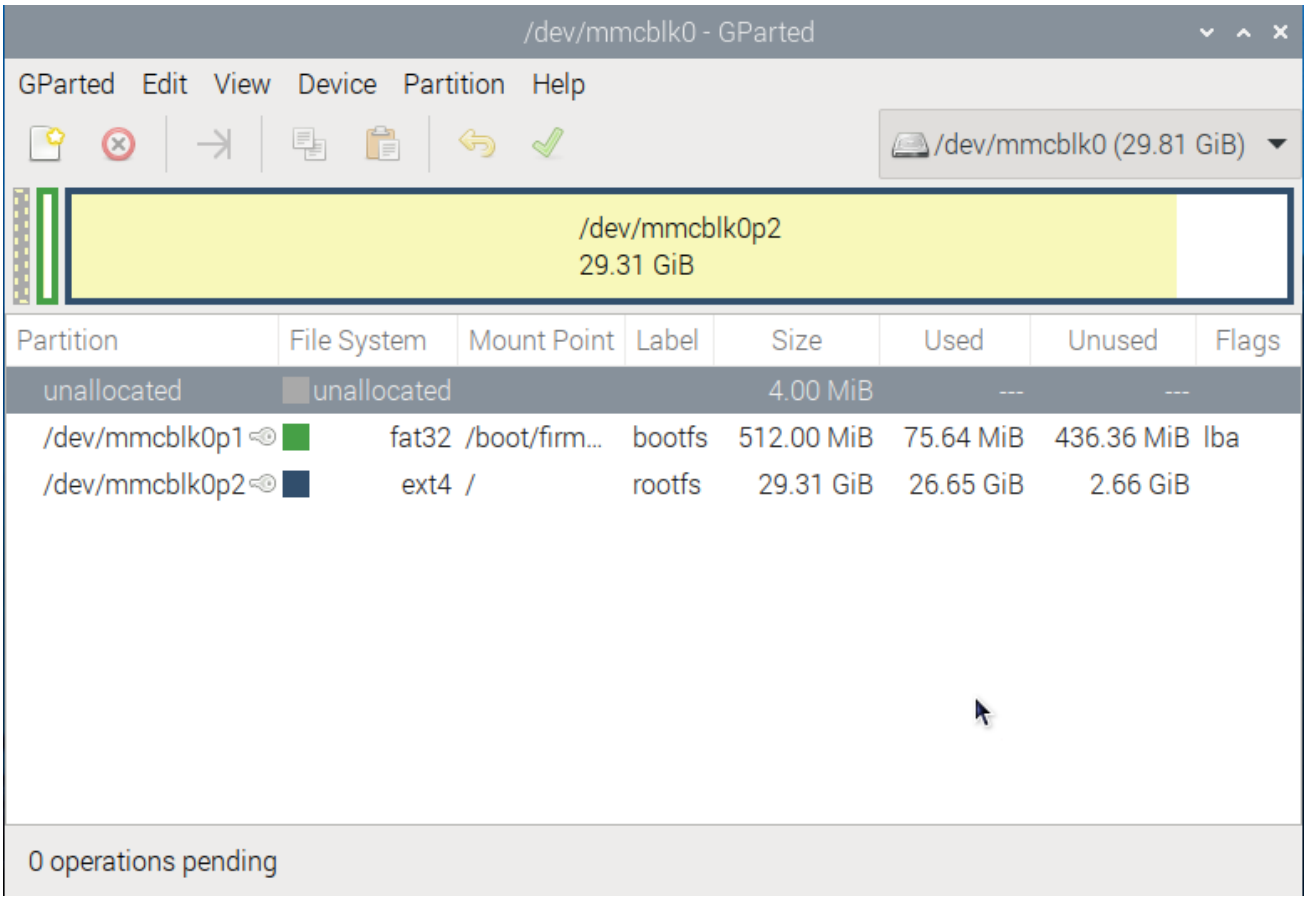
```
sudo apt-get install gparted
```

```
The following NEW packages will be installed:
  gparted gparted-common
0 upgraded, 2 newly installed, 0 to remove and 164 not upgraded.
Need to get 772 kB/2,483 kB of archives.
After this operation, 8,638 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian bookworm/main arm64 gparted arm64 1.3.1-1 [77
2 kB]
Fetched 772 kB in 0s (2,579 kB/s)
Selecting previously unselected package gparted-common.
(Reading database ... 222757 files and directories currently installed.)
Preparing to unpack .../gparted-common_1.3.1-1_all.deb ...
Unpacking gparted-common (1.3.1-1) ...
Selecting previously unselected package gparted.
Preparing to unpack .../gparted_1.3.1-1_arm64.deb ...
Unpacking gparted (1.3.1-1) ...
Setting up gparted-common (1.3.1-1) ...
Setting up gparted (1.3.1-1) ...
Processing triggers for mailcap (3.70+nmu1) ...
Processing triggers for desktop-file-utils (0.26-1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1.1) ...
Processing triggers for man-db (2.11.2-2) ...
pi@raspberrypi:~ $
```

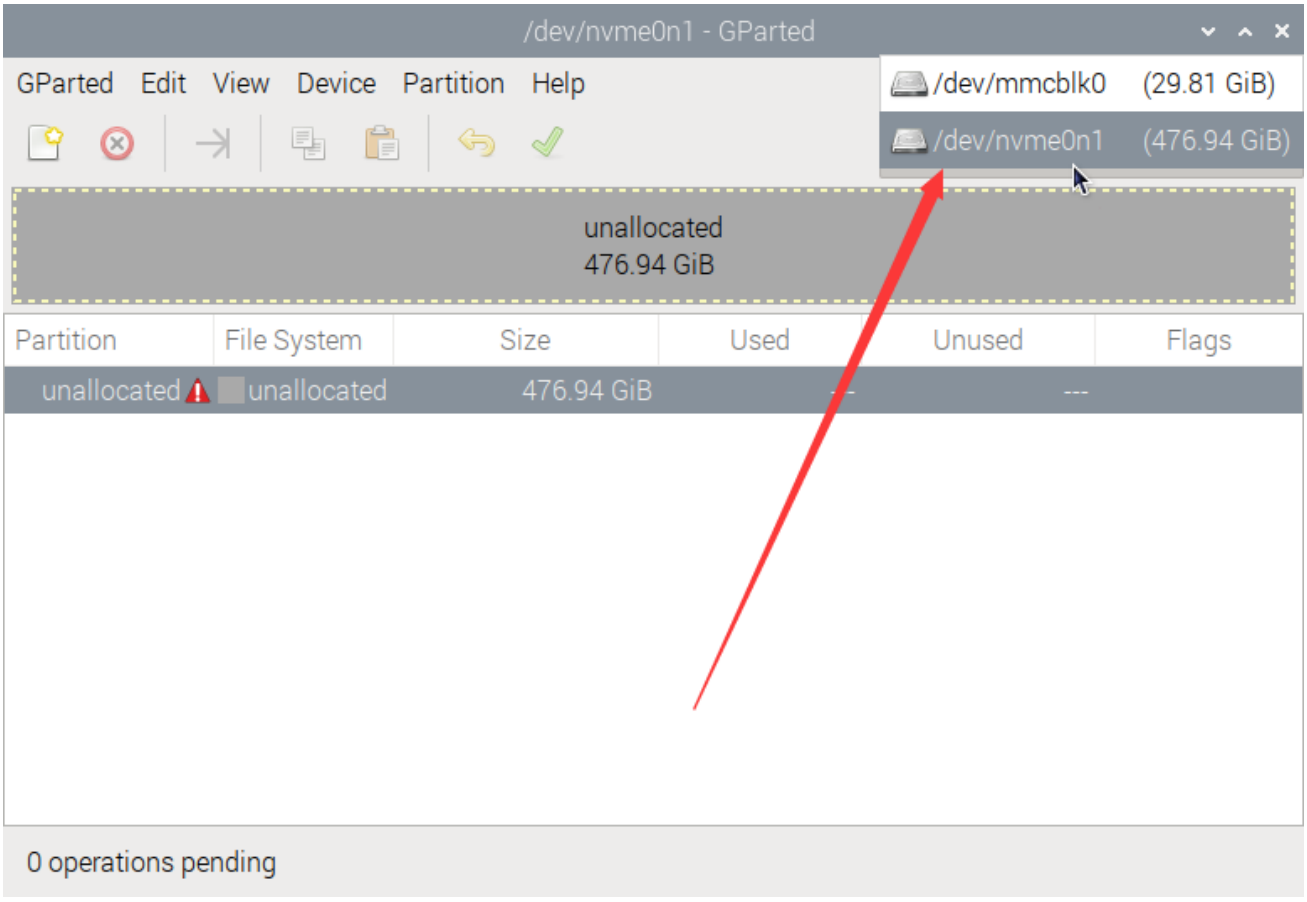
Open gparted with the command:

```
sudo gparted
```

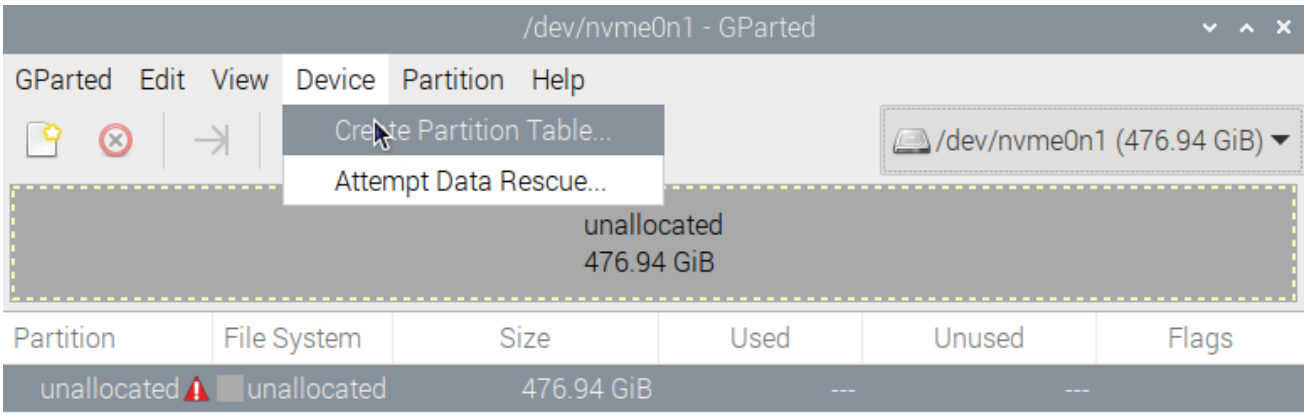
```
pi@raspberrypi:~ $ sudo gparted
error: XDG_RUNTIME_DIR is invalid or not set in the environment.
GParted 1.3.1
configuration --enable-libparted-dmraid --enable-online-resize
libparted 3.5
/dev/nvme0n1: unrecognised disk label
```



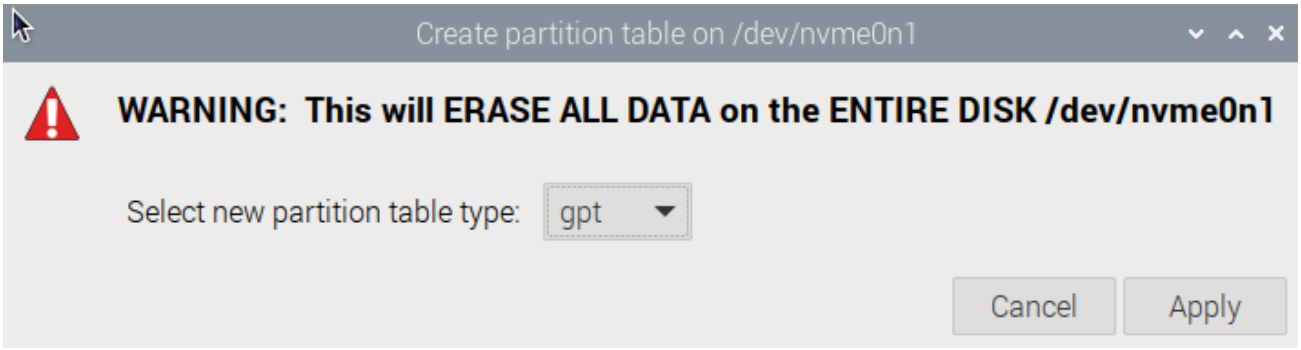
Click on the dropdown menu in the upper right corner and switch to NVME SSD.



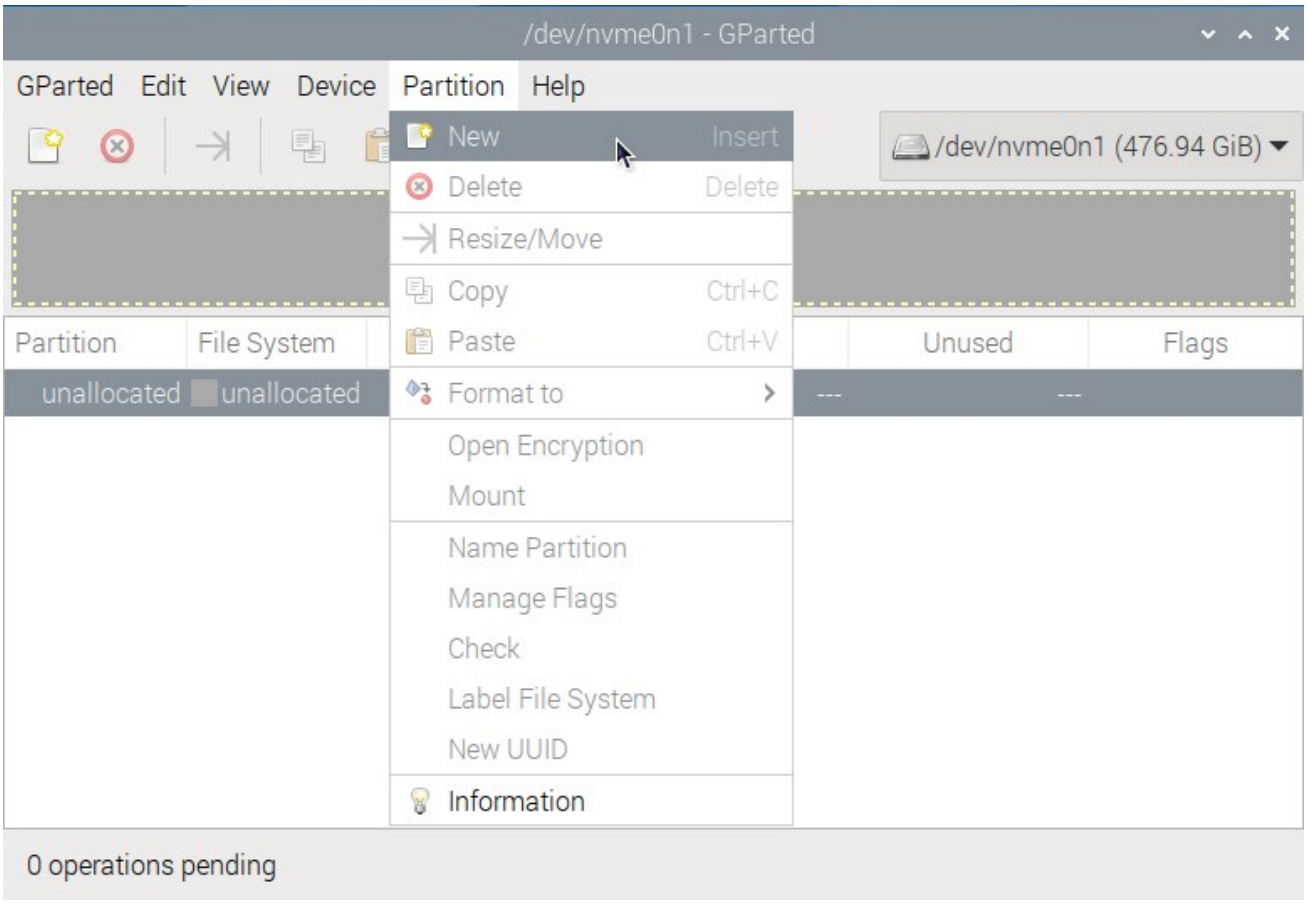
Click Device on the menu bar and select Create Partition Table.



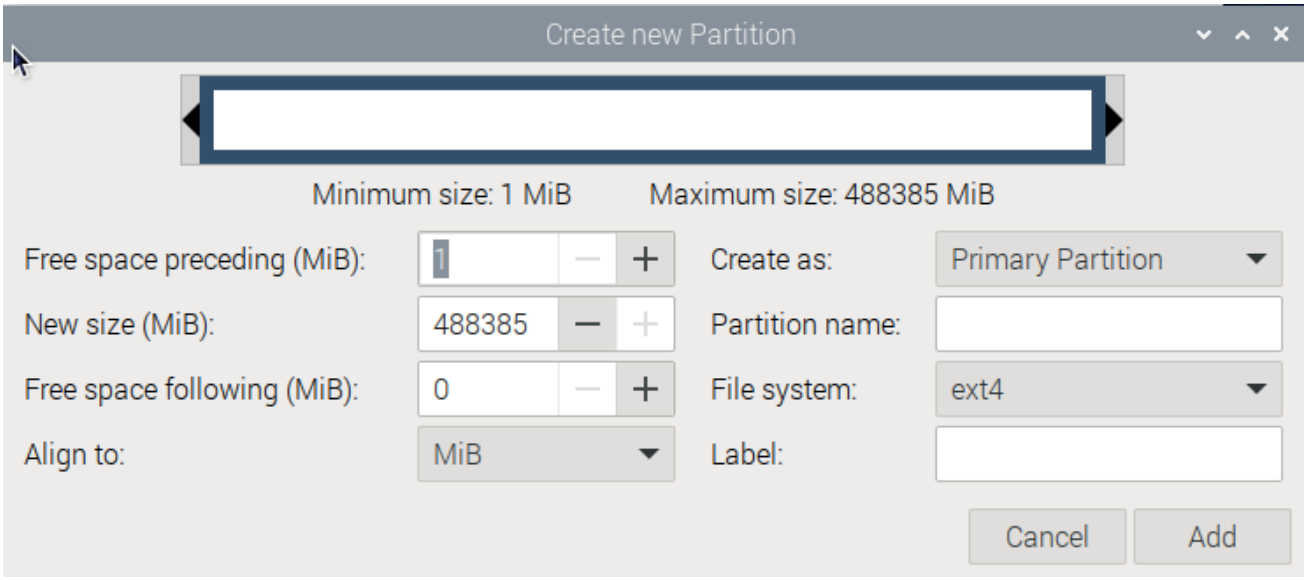
You will see the prompt that data will be erased. It is recommended to select gpt for partition table type. Click Apply.



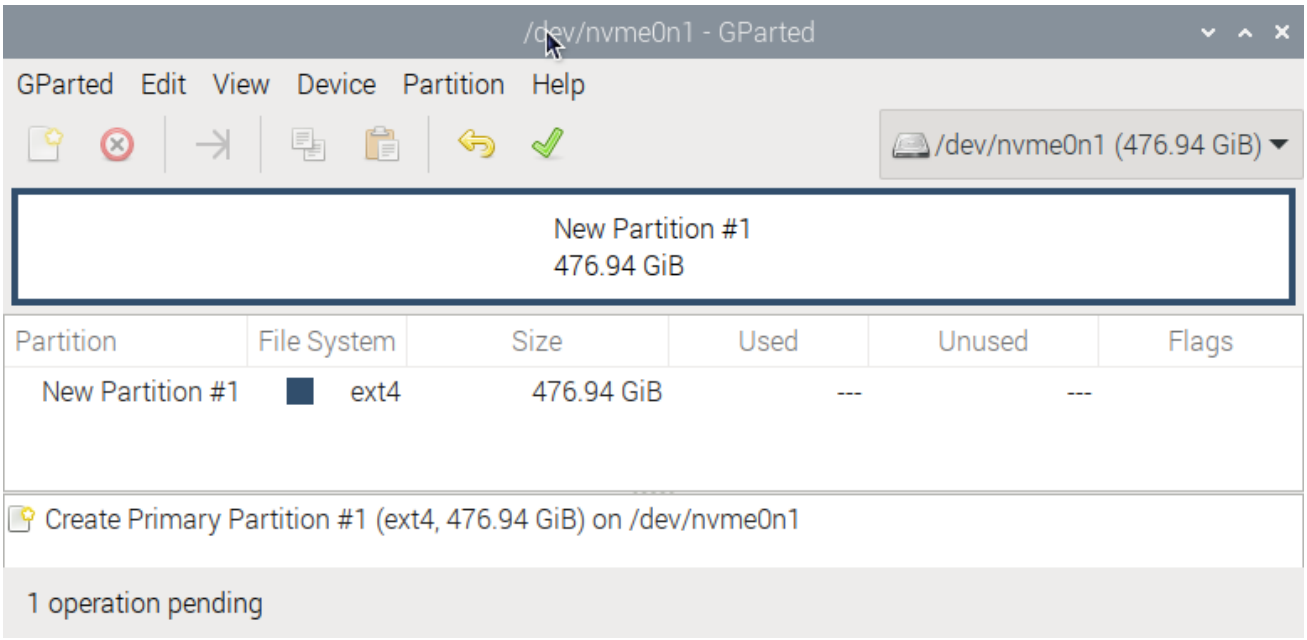
Click Partition on the menu bar, choose New.



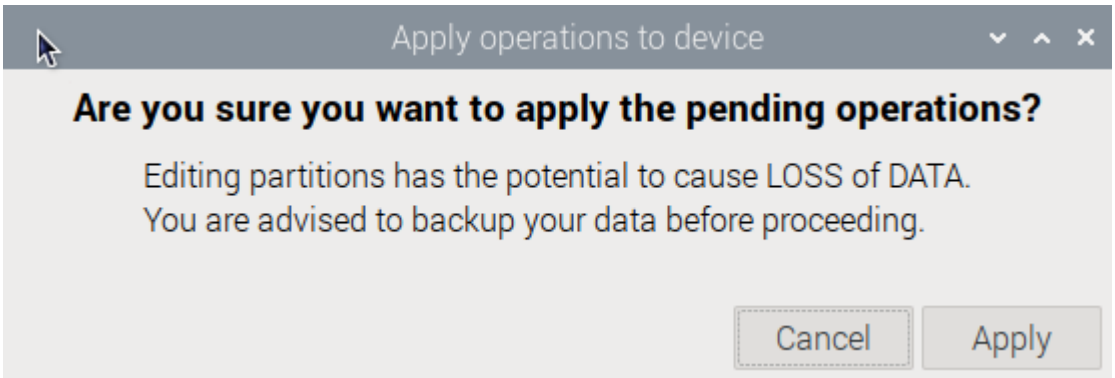
As shown in the figure below, the size of partition can be adjusted by dragging the mouse left and right, or by entering the size directly. The other options can be left as default setting. Here, we allocate all the capacity to a single partition. Click on Add.



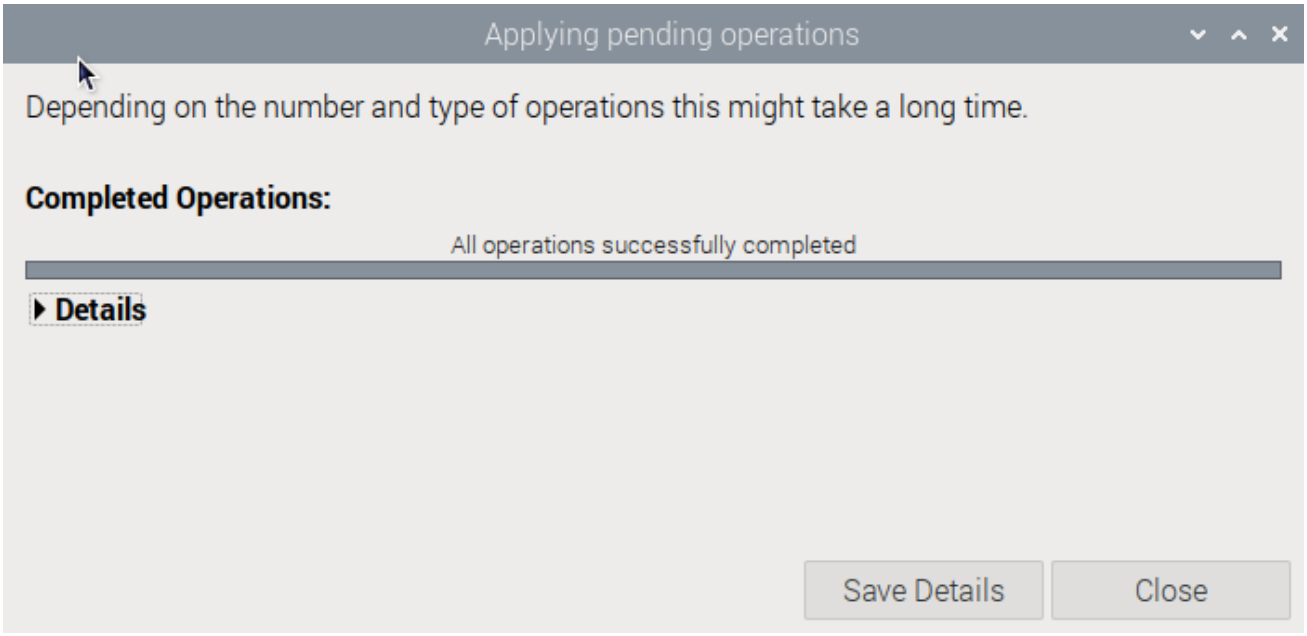
Click the check icon  to save the partition just built, as illustrated below.



Click on Apply.



Wait for it to complete and click on Close.

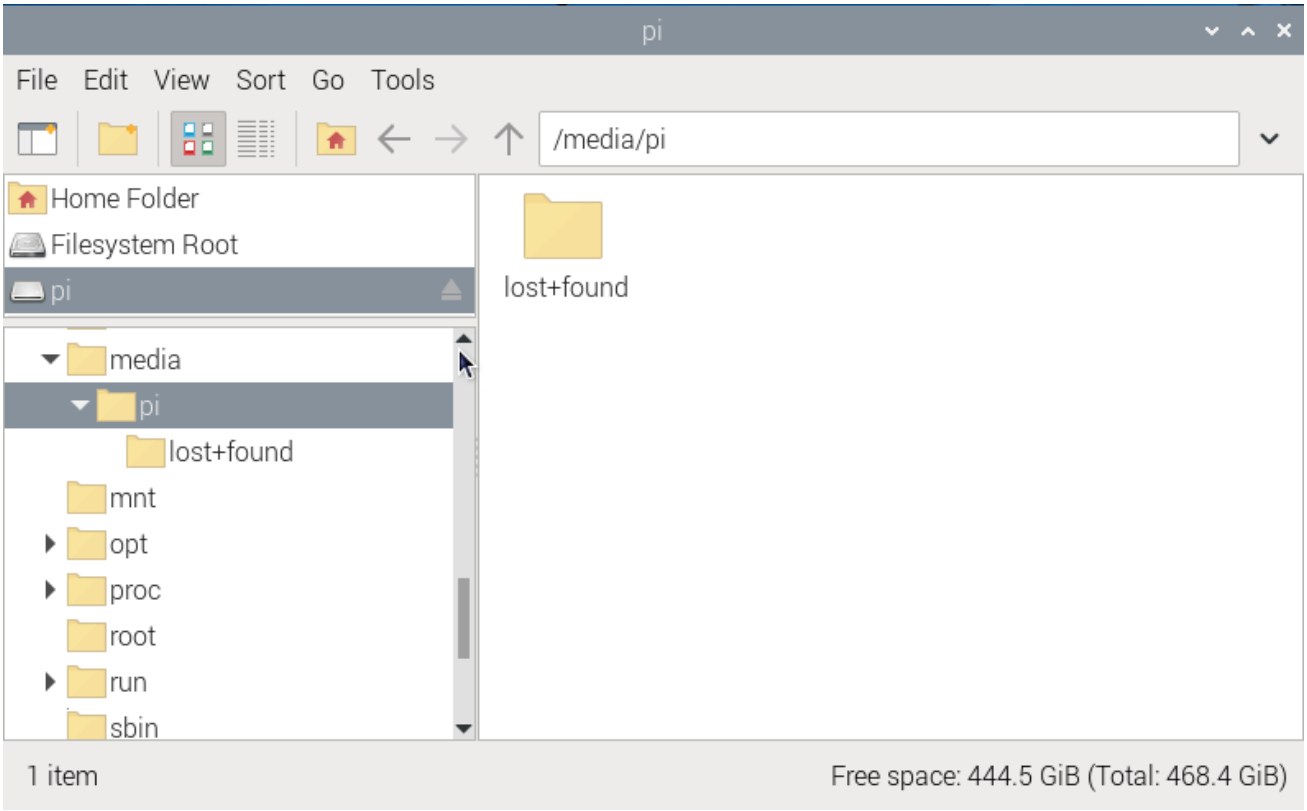

Applying pending operations
Depending on the number and type of operations this might take a long time.
Completed Operations:
All operations successfully completed
Details
Save Details
Close

At this point, you can mount the disk using the mount command and then access the disk space through the file manager. Use the following command to mount the SSD:

```
mkdir pi
sudo mount /dev/nvme0n1p1 /media/pi
```

```
pi@raspberrypi:~ $ mkdir pi
pi@raspberrypi:~ $ sudo mount /dev/nvme0n1p1 /media/pi
```

Open the file manager, as shown below.



If you plan to use the SSD as a standard storage device, you can conclude the process here. However, if you want to further proceed with installing an operating system on the SSD, please read on.

2.2.4 Flashing the OS

Install the OS to SSD with the method similar to that in the previous section on installing a system onto an SD card. This time, operate on the Raspberry Pi.

Install rpi-imager with the following command:

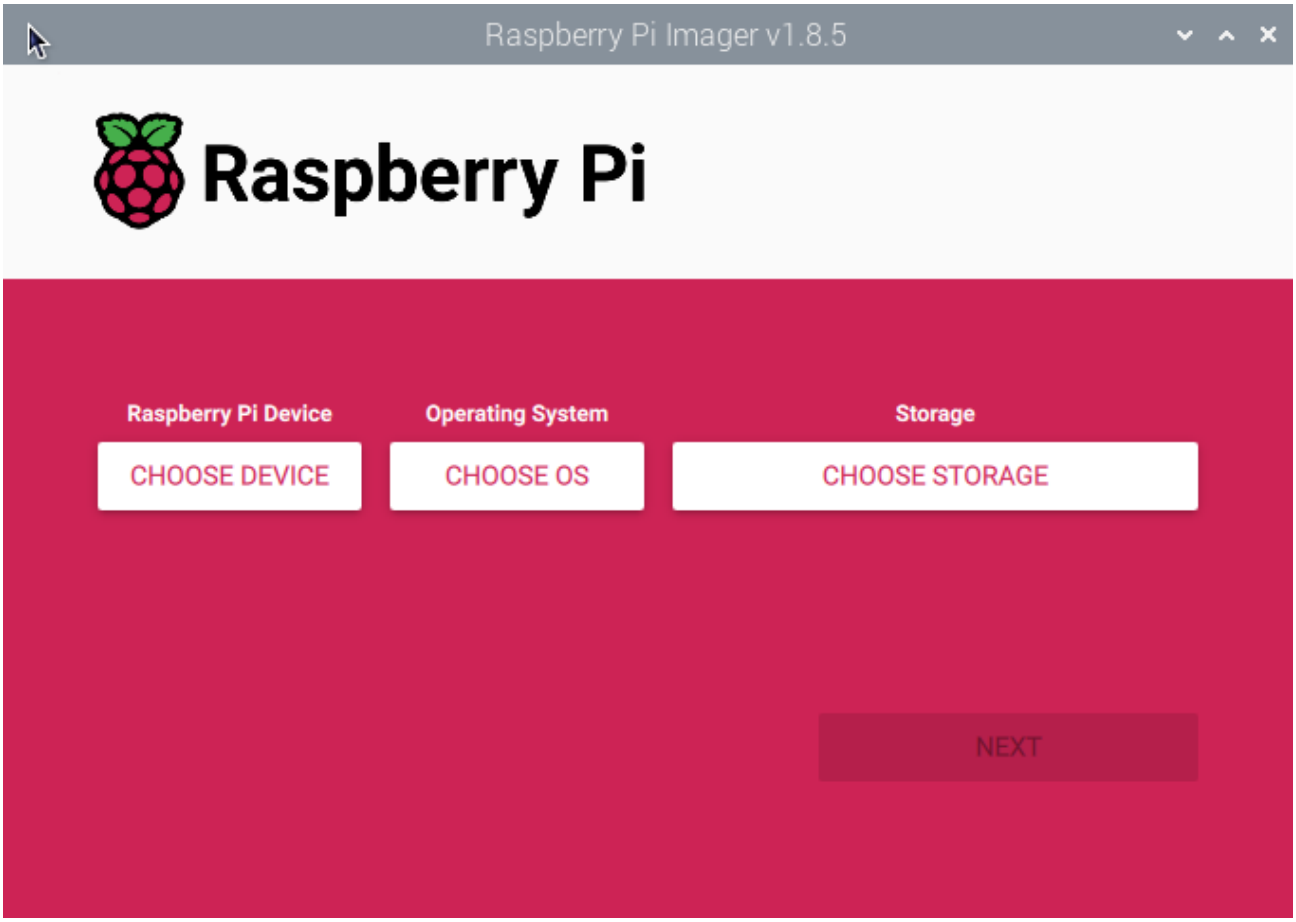
```
sudo apt install rpi-imager
```

```
pi@raspberrypi:~ $ sudo apt install rpi-imager
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
rpi-imager is already the newest version (1.8.5+rpt1).
0 upgraded, 0 newly installed, 0 to remove and 164 not upgraded.
pi@raspberrypi:~ $
```

Open rpi-imager:

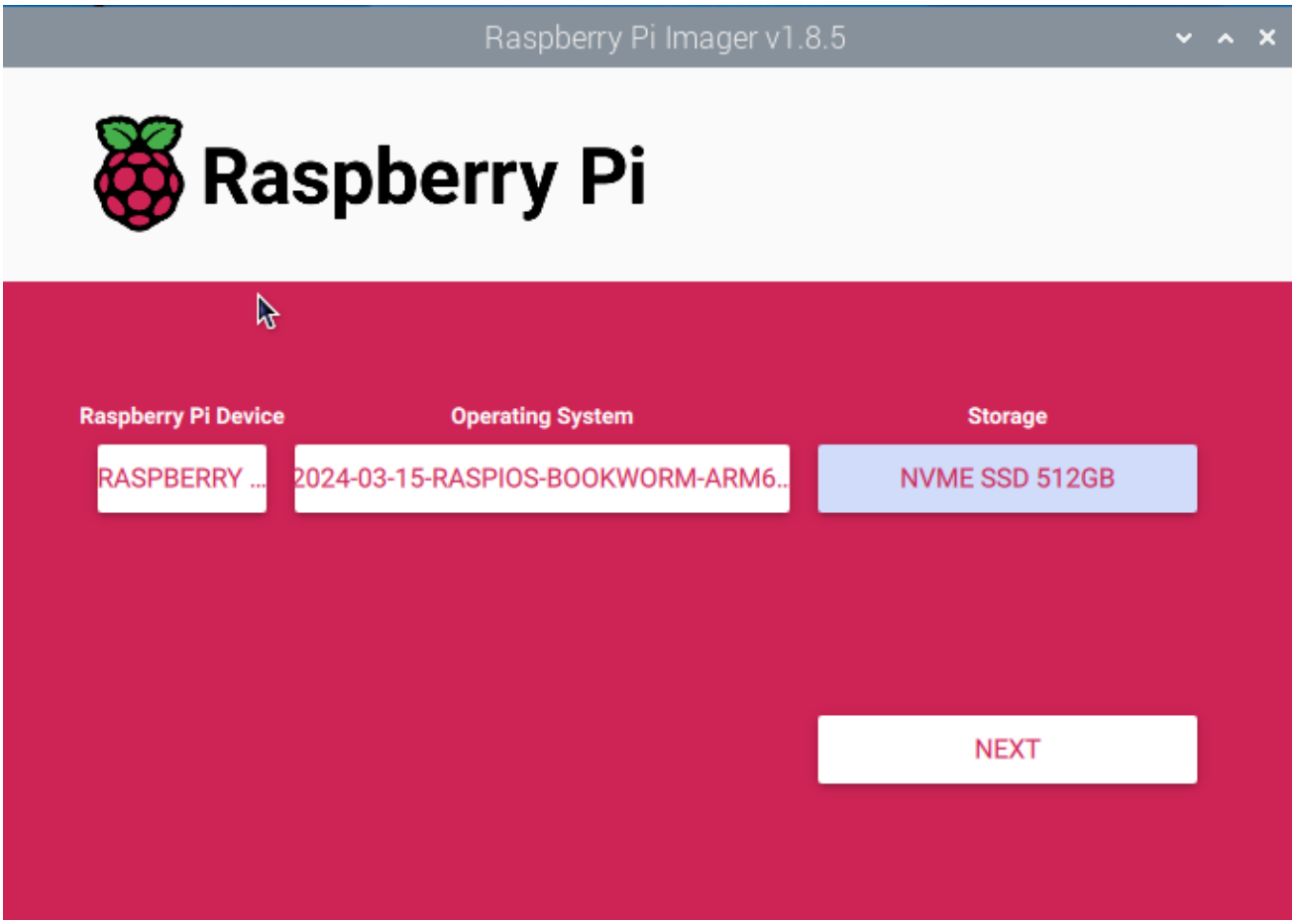
```
sudo rpi-imager
```

```
pi@raspberrypi:~ $ sudo rpi-imager
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
error: XDG_RUNTIME_DIR is invalid or not set in the environment.
```

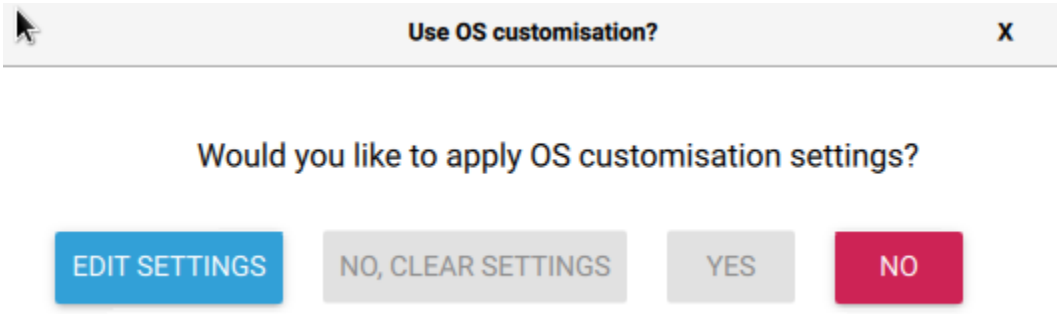


By this point, you should be quite familiar with the process.

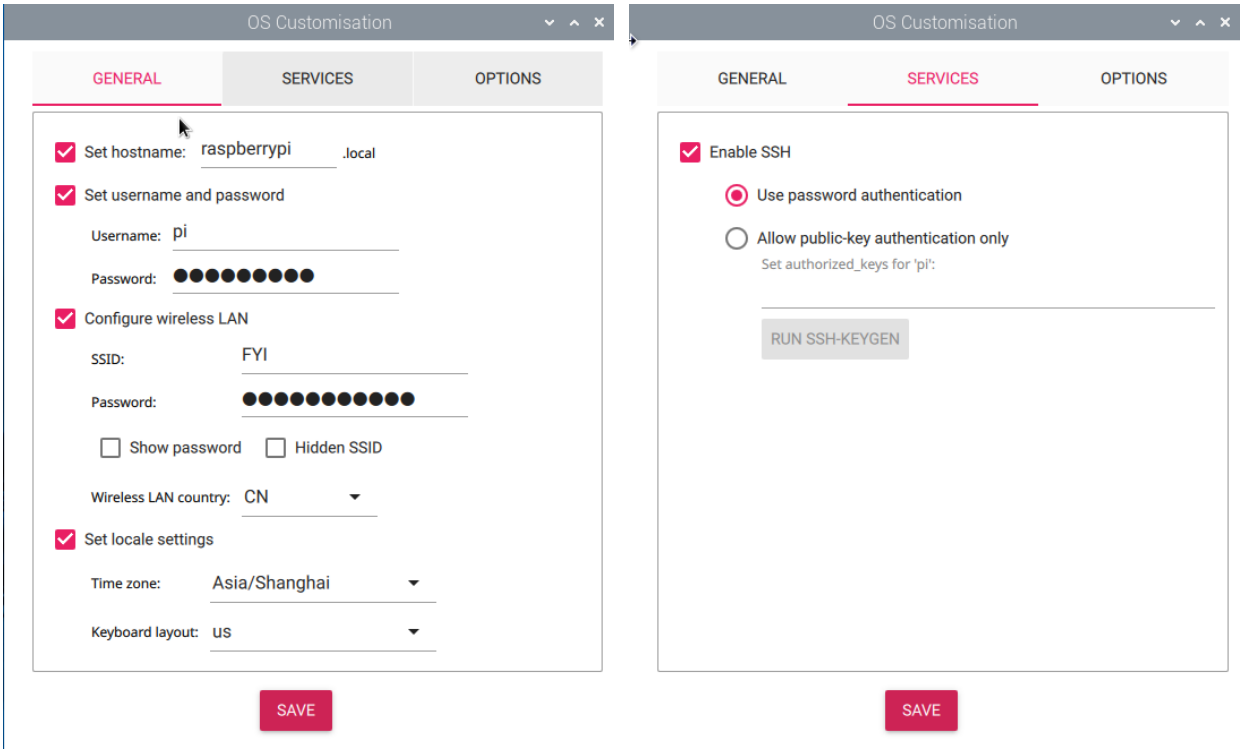
Select the Raspberry Pi 5 as your device and choose either an online download or an offline file for the operating system; in this case, an offline file is selected. (It is recommended to use a 64-bit Raspberry Pi system with recommended software). Choose your NVME SSD as the storage device. Click NEXT.



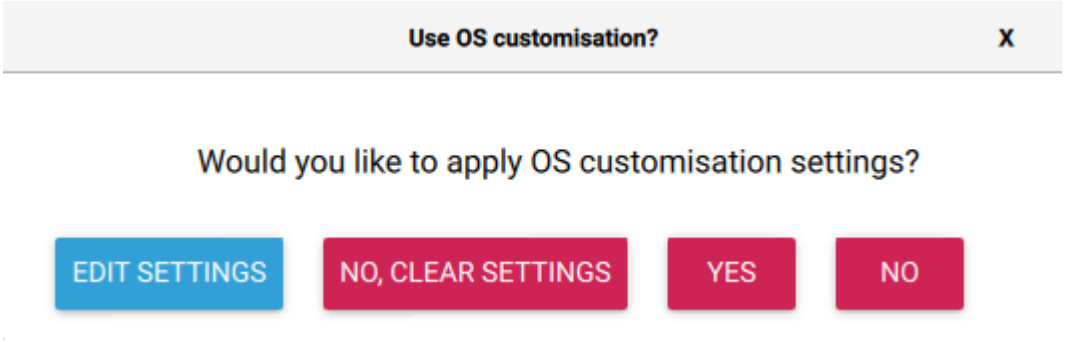
Click on EDIT SETTINGS.



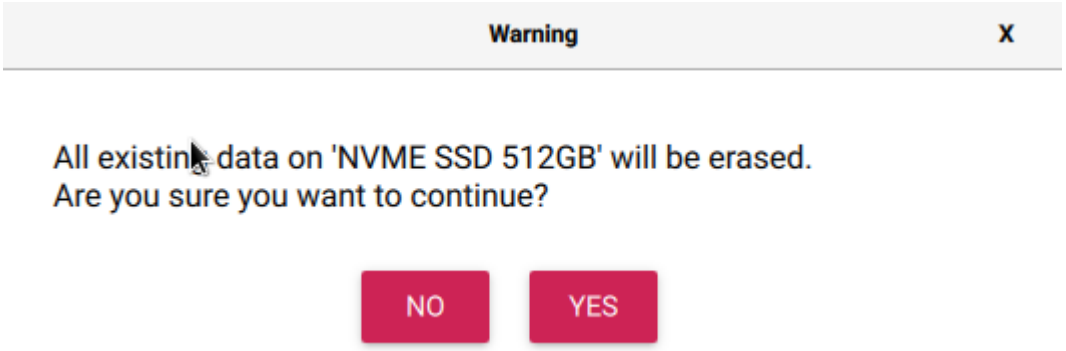
Wireless LAN Country must be correctly set; otherwise, it may fail to search the WiFi.
Enable SSH and click Save.



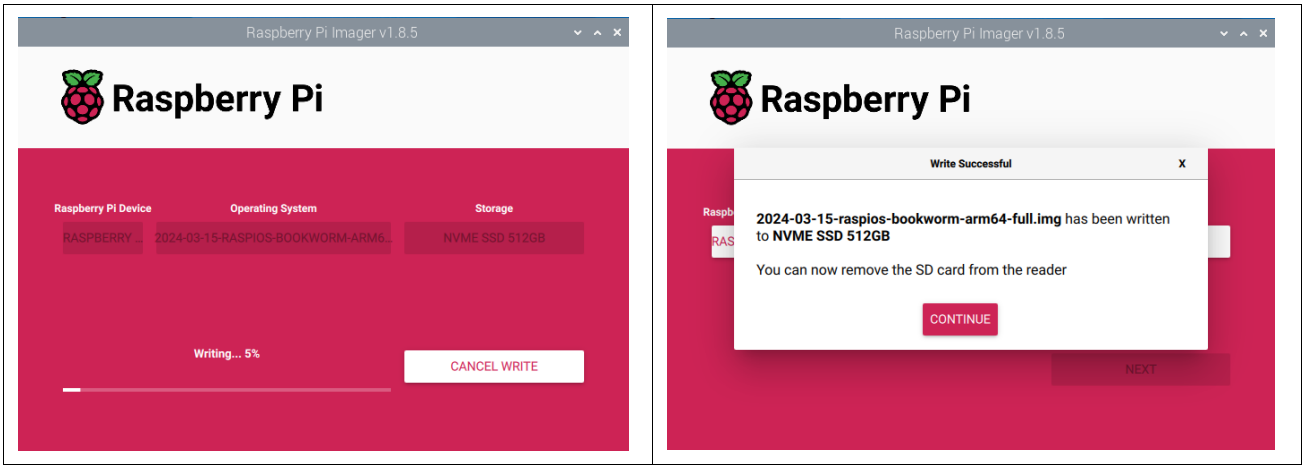
Click on YES.



Click on YES.



Wait for it to finish.



Congratulations! You have done the most tricky and the time-comsuming part. Now that you have successfully installed the operating system onto the NVMe SSD, you are very close to achieving a triumph.

Next, boot into the system from SSD.

2.2.5 Enable PCIE3.0 (on system written into SSD)

If you have confirmed that SSD is with Phison controller in step 2, then you also need to enable PCIE3.0 on the system written into SSD.
If the controller of your SSD is not from Phison, **you can skip this section.**

The operation is as below:

1. Run the command `lsblk` to check the partitions of the SSD with Raspberry Pi OS written, as shown below:

```
pi@raspberrypi:~ $ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
mmcblk0     179:0    0  29.3G  0 disk
├─mmcblk0p1 179:1    0   512M  0 part /boot/firmware
└─mmcblk0p2 179:2    0  28.8G  0 part /
nvme0n1     259:0    0 119.2G  0 disk
├─nvme0n1p1 259:1    0   512M  0 part
└─nvme0n1p2 259:2    0 118.7G  0 part
pi@raspberrypi:~ $
```

(The above screenshot is the result of an 128GB SSD with Phison as main controller.)

Run the following commands one by one to mount partition 1 of the SSD to the directory of /media/pi.

```
sudo mkdir /media/pi
sudo mount /dev/nvme0n1p1 /media/pi
```

```
pi@raspberrypi:~ $ sudo mkdir /media/pi
pi@raspberrypi:~ $ sudo mount /dev/nvme0n1p1 /media/pi
```

If it mounts successfully, you'll see the following disk icon on the desktop.

Open and modify the config.txt file with the following command.

```
sudo nano /media/pi/config.txt
```

```
pi@raspberrypi:~ $ sudo nano /media/pi/config.txt
```

Add the line `dtparam=pciex1_gen=3` to the end of the file, as shown below:

```
# Disable compensation for displays with overscan
disable_overscan=1

# Run as fast as firmware / board allows
arm_boost=1

[cm4]
# Enable host mode on the 2711 built-in XHCI USB controller.
# This line should be removed if the legacy DWC2 controller is required
# (e.g. for USB device mode) or if USB support is not required.
otg_mode=1

[cm5]
dtoverlay=dwc2,dr_mode=host

[all]
dtparam=pciex1_gen=3
```

Press Ctrl-O to save the file, Enter to confirm, and Ctrl-X to exit.

3. Booting from SSD

After finishing flashing the OS to SSD, shutdown Raspberry Pi, remove the power supply, and remove the SD card. Then connect the power, the Raspberry Pi will boot from SSD.

The default boot order of Raspberry Pi is SD card → SSD → USB, Therefore, when the SD card is removed, the Raspberry Pi cannot detect the SD card, it will boot from SSD. By far, the Raspberry Pi can boot successfully from NVME SSD.

```
B1 SD Card Boot  Boot from SD Card before trying NVMe and then USB (RECOMMENDED)
B2 NVMe/USB Boot Boot from NVMe before trying USB and then SD Card
B3 Network Boot  Boot from Network unless override by SD Card
```

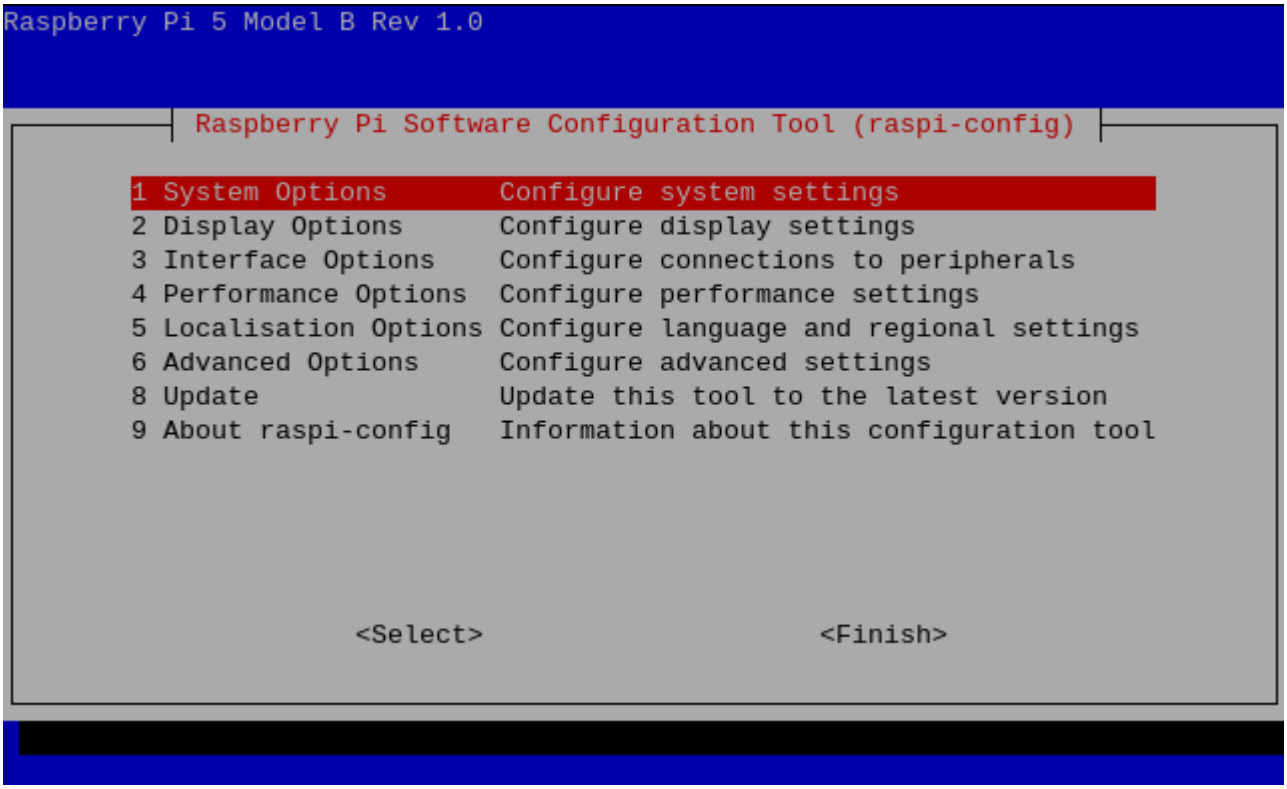
If you want the Raspberry Pi to boot from the SSD first, please continue with the following steps to modify the boot order. The boot order is saved in the Pi's EEPROM, so it does not matter whether you modify the boot order on SD card system or SSD system.

If you do not want to change the boot order, please skip this chapter.

3.1 Configuring the Boot Order

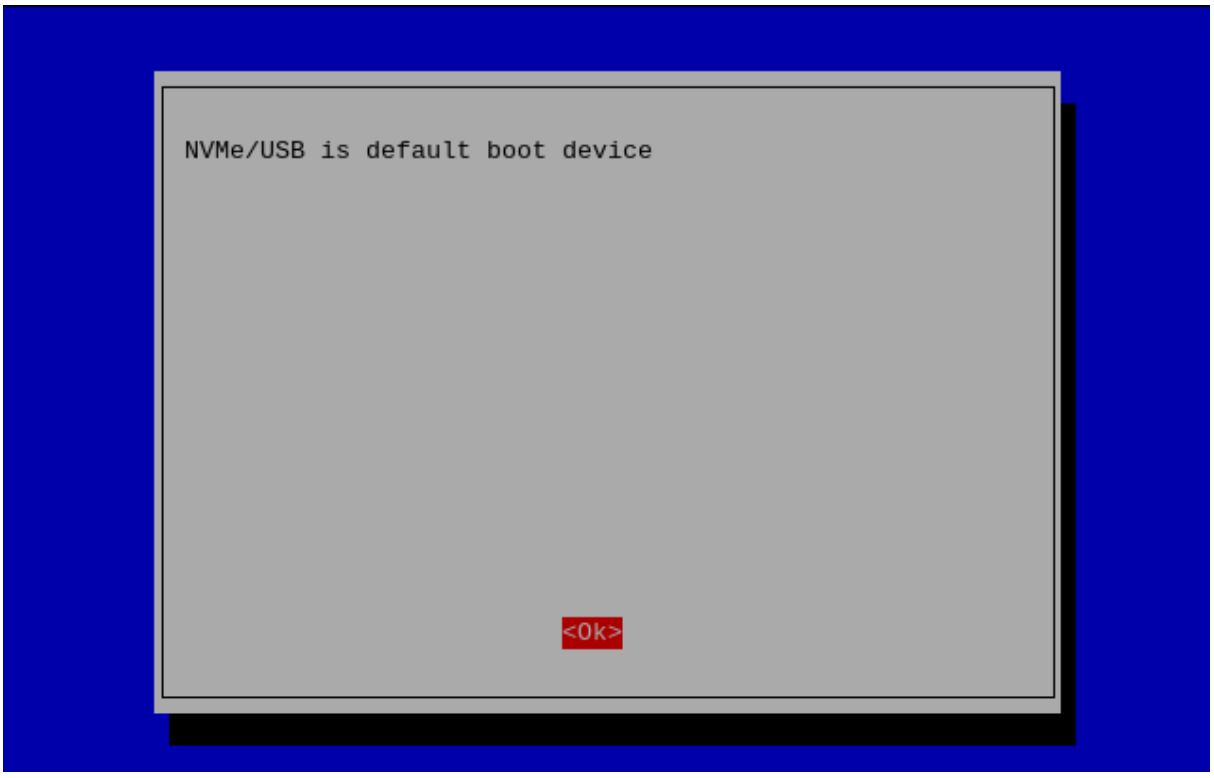
Type the following command in the Terminal.

```
sudo raspi-config
```

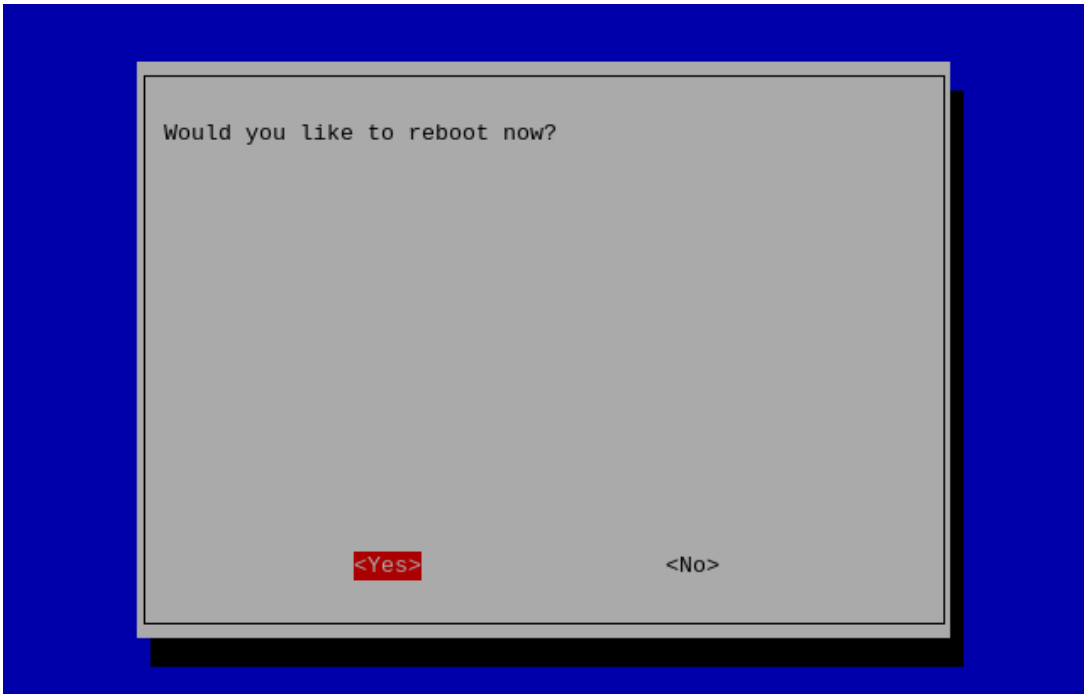


Using the keyboard's arrow keys and the Enter key, select the options in sequence.

"6 Advanced Options" →"A4 Boot Order"→"B2 NVME/USB Boot ···"

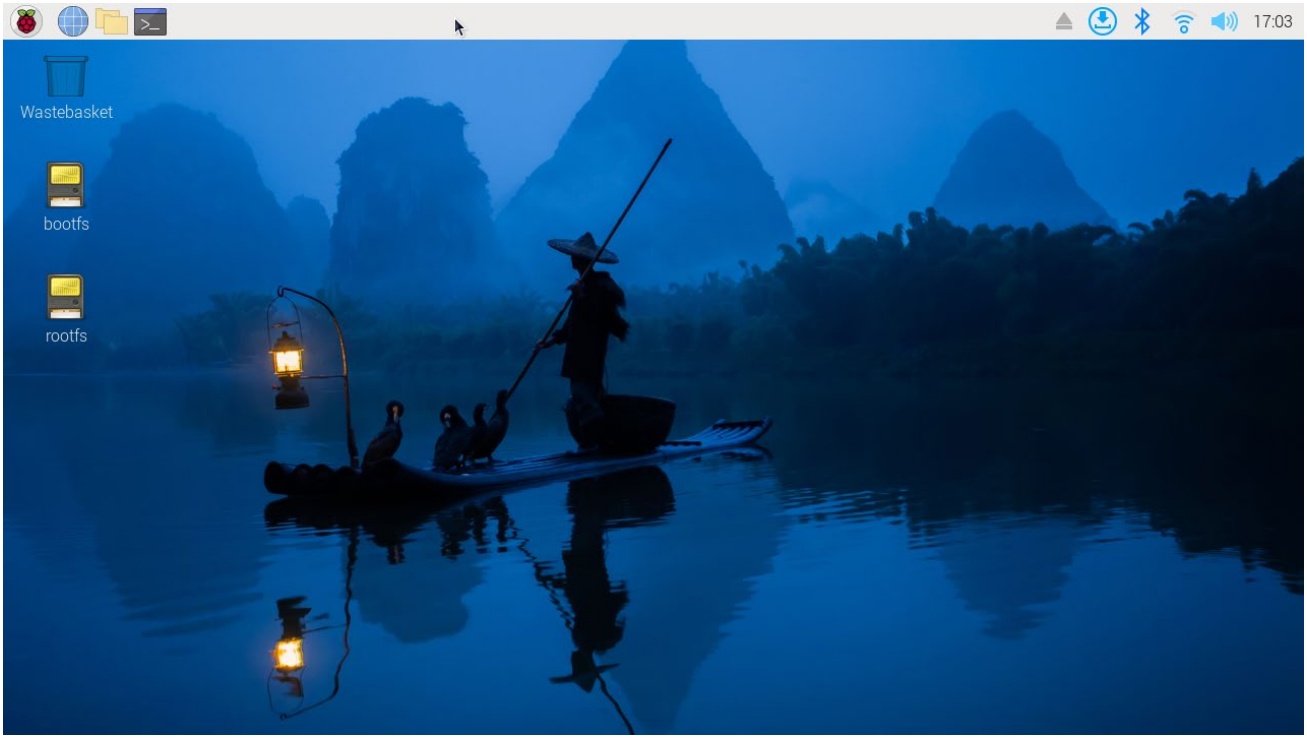


Select "OK"→"Finish"→"Yes", and reboot your Raspberry Pi.



At this point, upon restarting, the Raspberry Pi will boot from the NVME SSD first. If you are using an external monitor, you will see that the Raspberry Pi has booted up correctly. If your SD card is still inserted, you will also see an icon on the desktop as shown below.

With this, the process of booting the Raspberry Pi from the NVME SSD has been fully completed.



If you use VNC viewer, you will need to repeat the previous steps to acitvate the VNC service as it is not yet enabled in the new system on the SSD. Here, we take Windows as an example.

Run the following command:

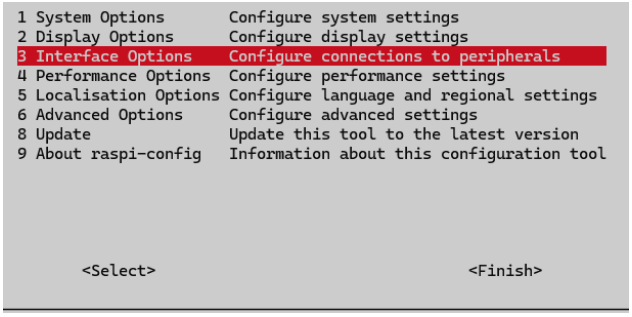
```
ssh pi@raspberrypi.local
```

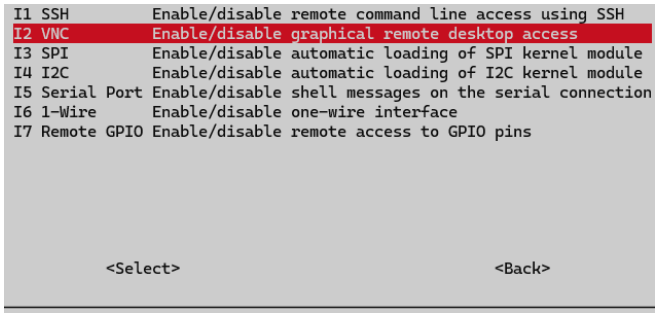
```
C:\Users\Administrator>ssh pi@raspberrypi.local
The authenticity of host 'raspberrypi.local (240e:3b4:3812:1fc0:954e:f55f:a772:fed5)' can't be established.
ECDSA key fingerprint is SHA256:hcx7u6H73nUsIc5WXA3HWa5GPSZEDroiz/mMbQx3ogc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'raspberrypi.local,240e:3b4:3812:1fc0:954e:f55f:a772:fed5' (ECDSA) to the list of known hosts
.
pi@raspberrypi.local's password:
Linux raspberrypi 6.6.20+rpt-rpi-2712 #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: Mon Jun  3 16:50:25 2024

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:~ $
```

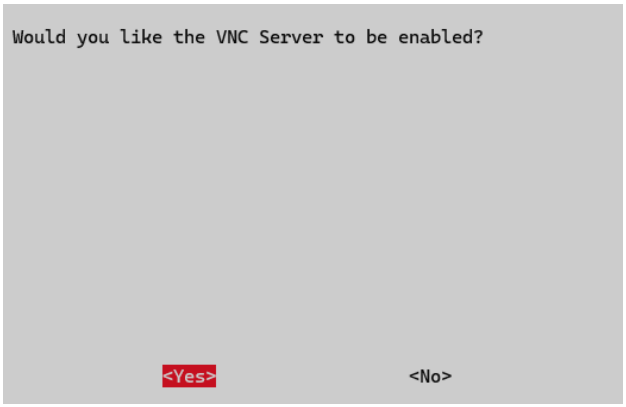
Once successfully ssh into Raspberry Pi, run the following command to open the configuration and enable VNC.

```
sudo raspi-config
```

Select “3 Interface Options”→”I2 VNC”→”Yes”→”Finish”.







Now you should be able to access Raspberrry Pi via VNC.

4.Speed Test & PCIe Gen3.0

The Raspberry Pi 5 includes a PCIe x1 slot that is certified for PCIe Gen 2.0, providng a theoretical maximum throughput of 5GT/sec, whichroughly translates to 500MB/sec for read and write operations. Although this slot is not officially certified for PCIe Gen 3.0, it is possible to force the use of Gen 3.0 for potentially higher speeds.

In actual tests, it is found that most SSDs can work stably at PCIE2.0, but are slightly unstable at PCIE3.0, while others are just the opposite. Therefore, please choose PCIE2.0 or PCIE3.0 according to your actual situation.

The PCIe consortium states that the speed of PCIe Gen 3.0 x1 is up to 8GT/sec, which translates to approximately 985MB/sec; however, Raspberry Pi claims that their implementation can achieve a speed of 10GT/sec, equivalent to around 1231MB/sec.
https://en.wikipedia.org/wiki/PCI_Express#Comparison_table
https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#pcie-gen-3-0

4.1 Disk Speed Test

This is an additional chapter for those who wish to test the read and write speeds of their SSD.

Open the terminal and enter the following command:

```
git clone https://github.com/TheRemote/PiBenchmarks
```

```
pi@raspberrypi:~ $ git clone https://github.com/TheRemote/PiBenchmarks
Cloning into 'PiBenchmarks'...
remote: Enumerating objects: 652, done.
remote: Counting objects: 100% (112/112), done.
remote: Compressing objects: 100% (80/80), done.
remote: Total 652 (delta 69), reused 65 (delta 31), pack-reused 540
Receiving objects: 100% (652/652), 119.41 KiB | 417.00 KiB/s, done.
Resolving deltas: 100% (386/386), done.
pi@raspberrypi:~ $
```

Enter the directory:

```
cd PiBenchmarks/
```

Grant executable permissions to the script:

```
chmod +x Storage.sh
```

```
pi@raspberrypi:~ $ cd PiBenchmarks/
pi@raspberrypi:~/PiBenchmarks $ ls
CODE_OF_CONDUCT.md  CONTRIBUTING.md  LICENSE  README.md  Storage.sh
pi@raspberrypi:~/PiBenchmarks $ chmod +x Storage.sh
pi@raspberrypi:~/PiBenchmarks $
```

Start the speed test. Please be aware that the first execution will involve downloading the required dependencies, so the process could take a relatively long time.

```
sudo ./Storage.sh ~/
```

```
pi@raspberrypi:~/PiBenchmarks $ sudo ./Storage.sh ~/
Trimming and syncing drives ...
/media/pi/bootfs: 433.1 MiB (454152192 bytes) trimmed on /dev/mmcblk0p1
/media/pi/rootfs: 2.5 GiB (2681069568 bytes) trimmed on /dev/mmcblk0p2
/boot/firmware: 436.4 MiB (457588736 bytes) trimmed on /dev/nvme0n1p1
/: 457.4 GiB (491171225600 bytes) trimmed on /dev/nvme0n1p2
Board information: Manufacturer: Raspberry Pi Foundation - Model: Raspberry Pi 5
 Model B Rev 1.0 - Architecture: aarch64 - OS: Debian GNU/Linux 12 (bookworm)
Fetching required components ...
Install lshw
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://deb.debian.org/debian-security bookworm-security InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Hit:4 http://archive.raspberrypi.com/debian bookworm InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
pciutils is already the newest version (1:3.9.0-4).
usbutils is already the newest version (1:014-1+deb12u1).
```

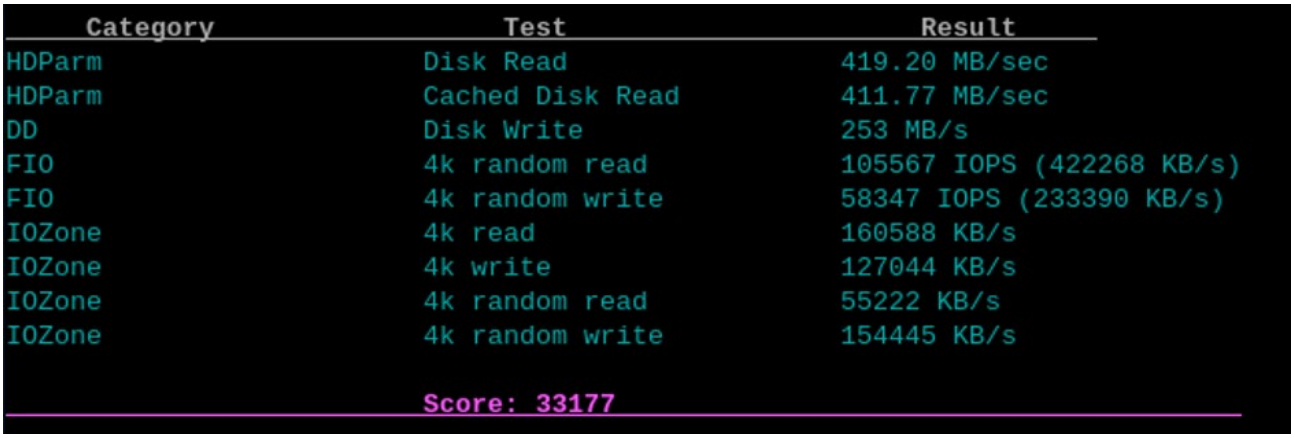
After the speed test is completed, follow the prompts to enter a description and a name for your SSD (you can use any arbitrary characters).

```
iozone test complete.
RandRead: 55044 - RandWrite: 169735 - Read: 194074 - Write: 159843
Enter a description of your storage and setup (Example: Kingston A400 SSD on Pi
4 using StarTech SATA to USB adapter)
Description: ssd1
(Optional) Enter alias to use on benchmark results.  Leave blank for completely
anonymous.
Alias (leave blank for Anonymous): ssd1
Result submitted successfully and will appear live on https://pibenchmarks.com w
ithin a couple of minutes.
```

Test result:

```
    Category                    Test                    Result
HDParm                      Disk Read               381.84 MB/sec
HDParm                      Cached Disk Read        376.91 MB/sec
DD                          Disk Write              312 MB/s
FIO                         4k random read          93090 IOPS (372363 KB/s)
FIO                         4k random write         70378 IOPS (281512 KB/s)
IOZone                      4k read                 194074 KB/s
IOZone                      4k write                159843 KB/s
IOZone                      4k random read          55044 KB/s
IOZone                      4k random write         169735 KB/s

                            Score: 36670
```

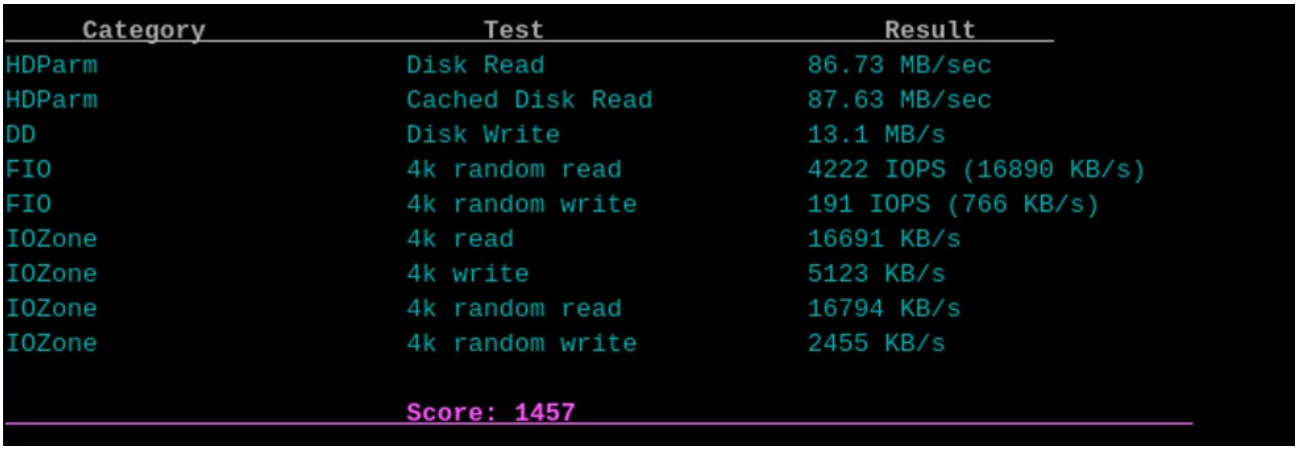
The performance varies among different SSDs, and each test may also have cetain error, which is normal. The following image shows the speed test results for another SSD:



Category	Test	Result
HDParm	Disk Read	419.20 MB/sec
HDParm	Cached Disk Read	411.77 MB/sec
DD	Disk Write	253 MB/s
FIO	4k random read	105567 IOPS (422268 KB/s)
FIO	4k random write	58347 IOPS (233390 KB/s)
IOZone	4k read	160588 KB/s
IOZone	4k write	127044 KB/s
IOZone	4k random read	55222 KB/s
IOZone	4k random write	154445 KB/s

Score: 33177

This is a speed test result for a TF (microSD) card, and it shows a significant difference in speed compared to an SSD.



Category	Test	Result
HDParm	Disk Read	86.73 MB/sec
HDParm	Cached Disk Read	87.63 MB/sec
DD	Disk Write	13.1 MB/s
FIO	4k random read	4222 IOPS (16890 KB/s)
FIO	4k random write	191 IOPS (766 KB/s)
IOZone	4k read	16691 KB/s
IOZone	4k write	5123 KB/s
IOZone	4k random read	16794 KB/s
IOZone	4k random write	2455 KB/s

Score: 1457

4.2 PCIe Gen 3.0

In the Preface, it is mentioned that the Raspberry Pi's PCIe Gen 3.0 has not been officially certified. While it is functional, its performance is not as reliable as desired. This chapter is presented as an exploratory section for assessing the speed capabilities of SSDs when used with PCIe Gen 3.0. **For practical applications, it is advised to opt for PCIe Gen 2.0 to ensure greater stability and dependability.**

EnablePCIe Gen3.0

Add the line `dtparam=pciex1_gen=3` to /boot/firmware/config.txt to enable PCIe Gen3.0.
As shown below, enter the command to open the file.

```
sudo nano /boot/firmware/config.txt
```

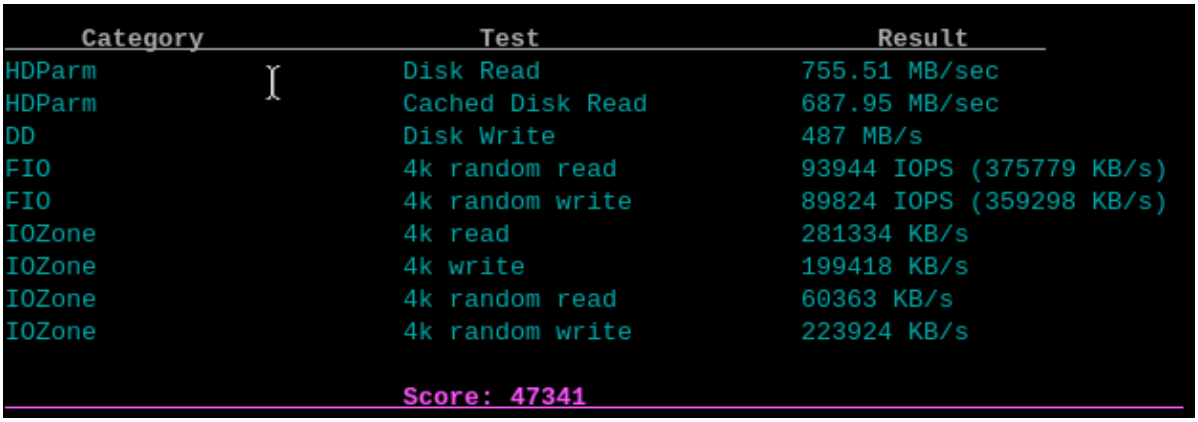
Add the line `dtparam=pciex1_gen=3` to the end of the file, as shown below:

Press Ctro-O to save the file, Enter to confirm, and Ctrl-X to exit.

Reboot your Raspberry Pi.

```
sudo reboot
```

After rebooting, test the speed again.



The speed of another SSD.

Category	Test	Result
HDParm	Disk Read	804.45 MB/sec
HDParm	Cached Disk Read	670.21 MB/sec
DD	Disk Write	404 MB/s
FIO	4k random read	206868 IOPS (827474 KB/s)
FIO	4k random write	92252 IOPS (369009 KB/s)
IOZone	4k read	224411 KB/s
IOZone	4k write	167052 KB/s
IOZone	4k random read	61980 KB/s
IOZone	4k random write	190269 KB/s
	Score: 47057	

Disable PCIe Gen3.0

Delete the line added with the previous step to disable PCIe Gen3.0.

Delete the line `dtparam=pciex1_gen=3` in the boot/firmware/config.txt file.

After the line is removed, it will change to PCIe Gen2.0.

What's Next?

THANK YOU for participating in this learning experience! If you have completed all of the projects successfully you can consider yourself a Raspberry Pi Master.

We have reached the end of this Tutorial. If you find errors, omissions or you have suggestions and/or questions about the Tutorial or component contents of this Kit, please feel free to contact us: **support@freenove.com**
We will make every effort to make changes and correct errors as soon as feasibly possible and publish a revised version.

If you are interesting in processing, you can study the Processing.pdf in the unzipped folder.

If you want to learn more about Arduino, Raspberry Pi, Smart Cars, Robotics and other interesting products in science and technology, please continue to visit our website. We will continue to launch fun, cost-effective, innovative and exciting products.
http://www.freenove.com/

Thank you again for choosing Freenove products.