Home » Raspberry Pi » Raspberry Pi PCIe To M.2 Adapter With PoE Function Instructions

Contents [hide]

Raspberry

Raspberry Pi PCIe To M.2 Adapter With PoE Function

Overview

Introduction

The POE M.2 HAT+(B) is a combination of Power over Ethernet (PoE) and PCIE to M.2 for the Raspberry Pi 5 that supports the IEEE 802.3af/at networking standard. It supports M.2 NVMe hard drives in 2230, 2242, 2260, and 2280 sizes, and supports SSD boot for the Raspberry Pi

Features

- Standard Raspberry Pi 40pin GPIO extension header, supports Raspberry Pi 5 (not

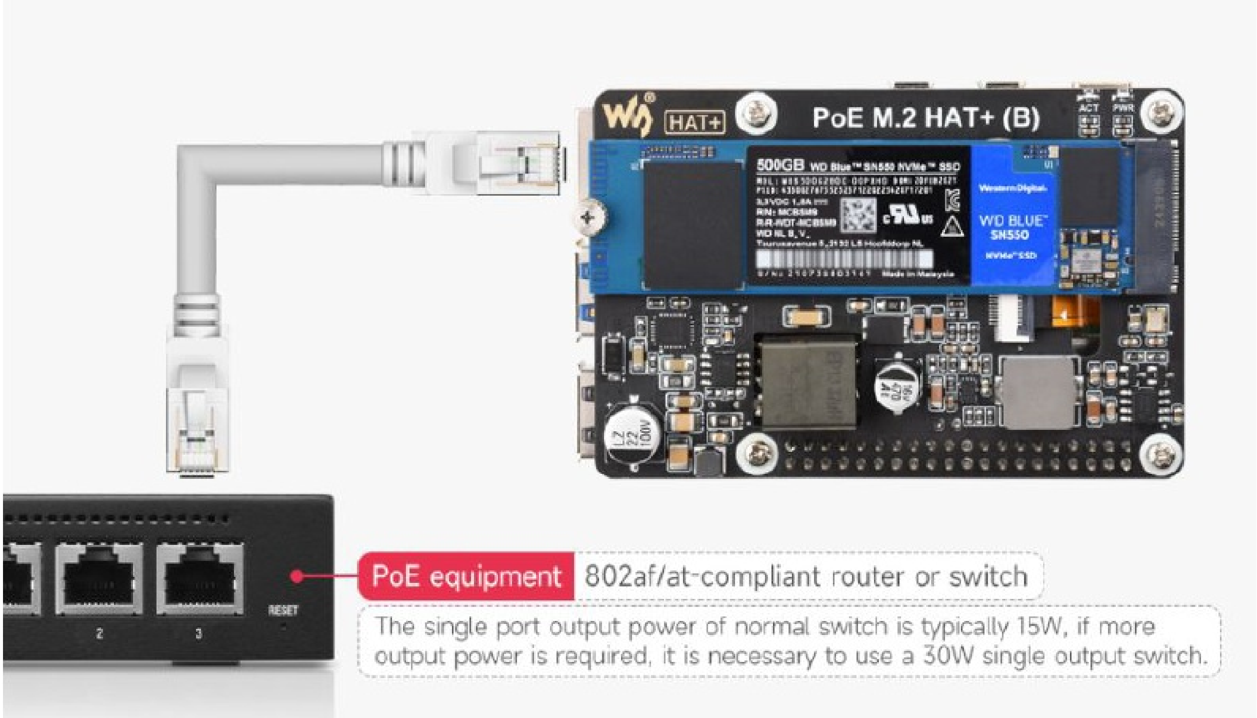
applicable to PI4B and PI3B)
    - Supports Power over Ethernet (PoE), supports IEEE 802.3af/at PoE network standard
- Adopts fully isolated Switch Mode Power Supply (SMPS) mode power
- Supports NVMe protocol M.2 interface hard drives, featuring high-speed read and write, and high work efficiency
- PCI-E×1 Gen2 or Gen3 mode
- Only supports PI5B
- Compatible with M.2 hard drives of 2230 / 2242 / 2260 / 2280 sizes
- Onboard operational indicator lights, the PWR is continuously lit when powered on, and the ACT blinks during read and write operations, making the operational status easily visible

Specifications

- PoE input voltage: 37V ~ 57V DC input
- PoE-GPIO pin header: 5V 4.5A (MAX)
- PoE-2P pin header: 12V 2A (MAX)
- Network standards: Supports IEEE 802.3af/at PoE
- Product size: 56.5mm × 70.0mm

Working with Raspberry Pi

Plug the POE M.2 HAT+ into the Raspberry Pi 5 as shown in the figure below



802af/at-compliant router or switch

The single port output power of normal switch is typically 15W, if more output power is required, it is necessary to use a 30W single output switch.

Notes

If a power supply limitation prompt appears

Please don’t strongly press the transformer on the board to avoid damaging the product when assembling the Raspberry Pi

Important Notes

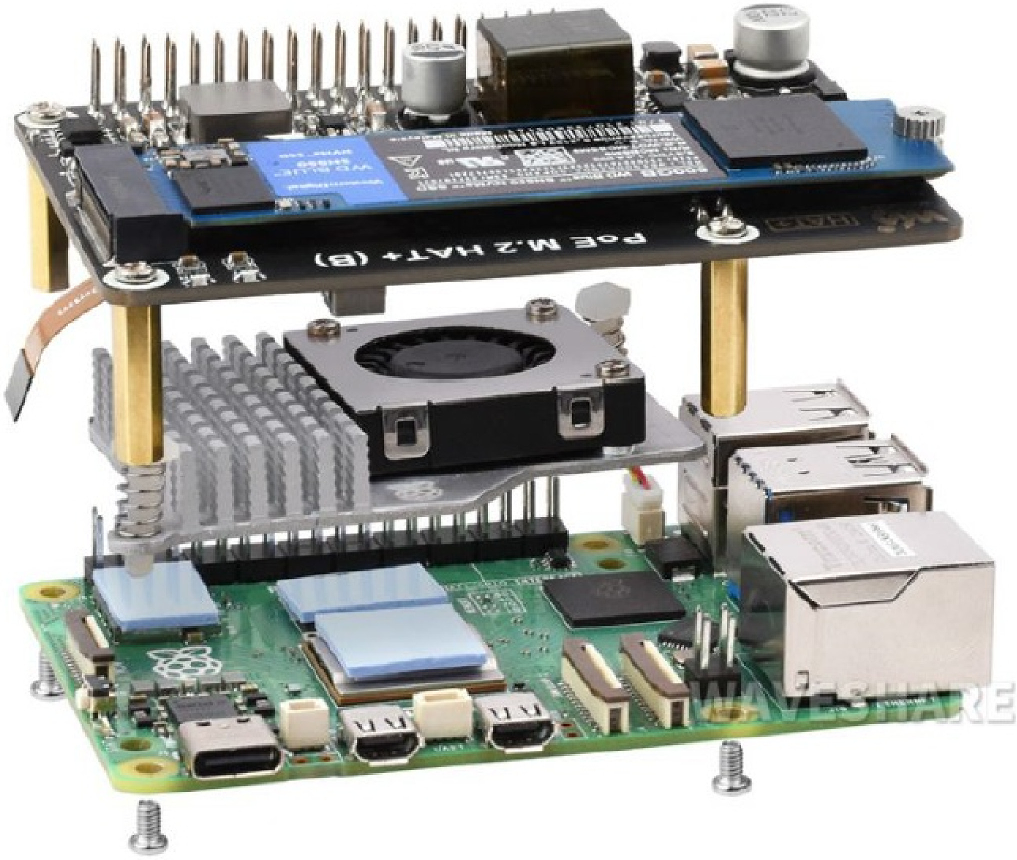
Please Dont Strongly Press The Transformer On The Board To Avoid Damaging The Product

When Assembling The Raspberry Pi

Raspberr-Pi-PCIe-To-M-2-Adapter-With-PoE-Function (3)

* the Raspberry Pi 5 is not included, please refer to the Package Content for detailed part list

Installation

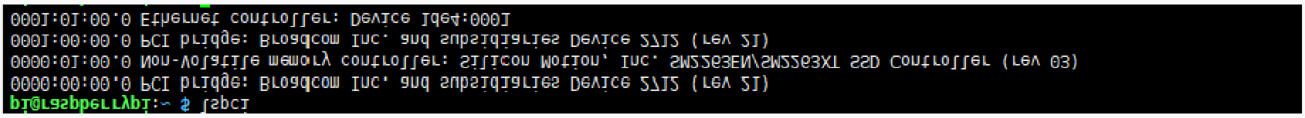


Hard disk mounting

1. Enable PCIE interface

   PI5B defaults to not having the PCIE interface enabled. Add to /boot/firmware/config.txt: dtparam=pciex1
2. PCIE is gen2 by default, if you need to enable PCIE gen3, then add following to /boot/firmware/config.txt: dtparam=pciex1_gen=3
3. After the modification, restart the PI5, and the device can be recognized.

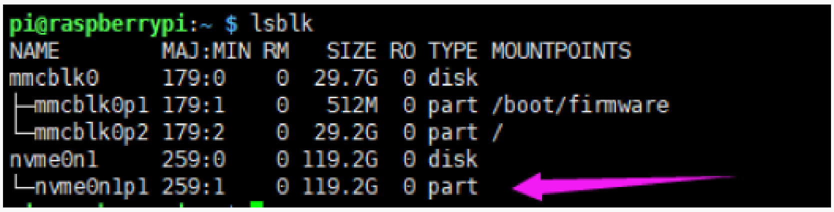
   As shown in the figure below, SM2263 is identified as my SSD solid state drives, and the other PI5 is the RPI chip

```
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
```

4. Partition, skip this step if you have partitioned and formatted on other platforms (Note: partitioning will delete all data on the SSD, proceed with caution)

Lsblk This command is executed to view the disk (if you want to see the details, run the sudo fdisk -l command)



```
pi@raspberrypi:~ $ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
mmcblk0     179:0    0  29.7G  0 disk
├─mmcblk0p1 179:1    0   512M  0 part /boot/firmware
└─mmcblk0p2 179:2    0  29.2G  0 part /
nvme0n1     259:0    0 119.2G  0 disk
└─nvme0n1p1 259:1    0 119.2G  0 part
```

Partition:

sudo fdisk /dev/nvme0n1 The device number is the total device number, do not add p1, that is just a partition

How to use the partitioning tool fdisk:

N New partition

q Quit without saving

p Print the partition table

m Print the selection menu

D Delete the partition

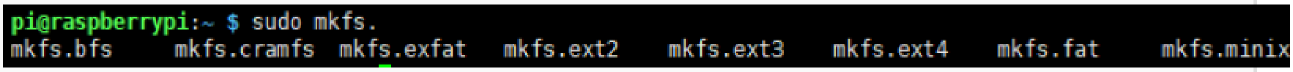
w Save and exit

t Modify the ID number

Add the partition and execute n, then save and exit with w

5. Format

sudo mkfs. Execute the command and press Tab key, you will see a lot of different suffixes, and the different suffixes are the formats you need to format

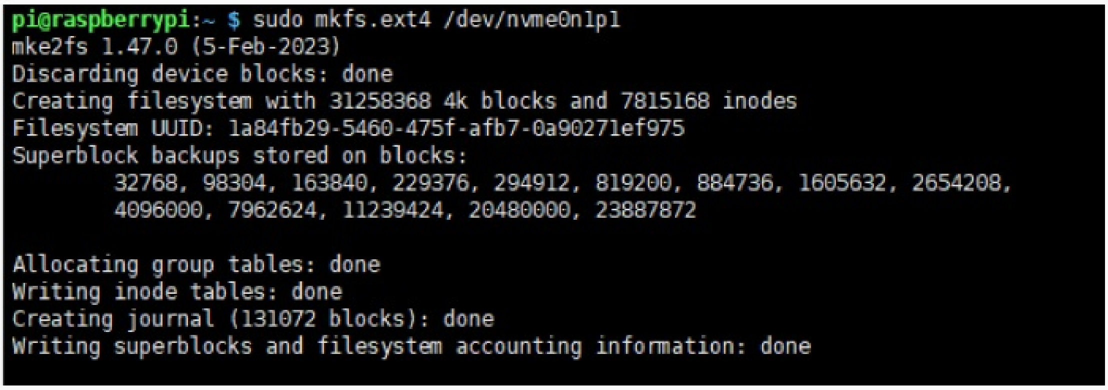


```
pi@raspberrypi:~ $ sudo mkfs.
mkfs.bfs     mkfs.cramfs  mkfs.exfat   mkfs.ext2    mkfs.ext3    mkfs.ext4    mkfs.fat     mkfs.minix
```

If I want to format it in ext4 file format, then execute the command:

sudo mkfs.ext4 /dev/nvme0n1p1

Wait a moment, once all "done" appear as below, it indicates that the formatting is completed



```
pi@raspberrypi:~ $ sudo mkfs.ext4 /dev/nvme0n1p1
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 31258368 4k blocks and 7815168 inodes
Filesystem UUID: 1a84fb29-5460-475f-afb7-0a90271ef975
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872

Allocating group tables: done
Writing inode tables: done
Creating journal (131072 blocks): done
Writing superblocks and filesystem accounting information: done
```

6. Mount

```
Create a mount directory

sudo mkdir toshiba

Mount the device

sudo mount /dev/nvme0n1p1 ./toshiba

Check the disk status

df -h
```

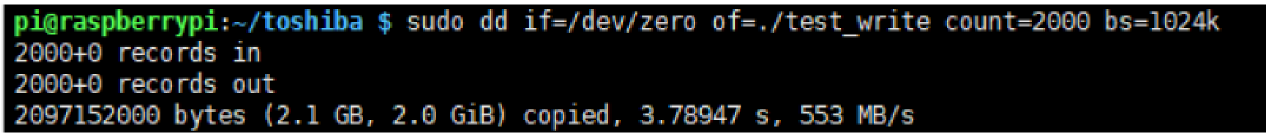
Read/Write test

Enter the directory where the disk is mounted cd toshiba

Free up the memory

sudo sh -c “sync && echo 3 > /proc/sys/vm/drop_caches”

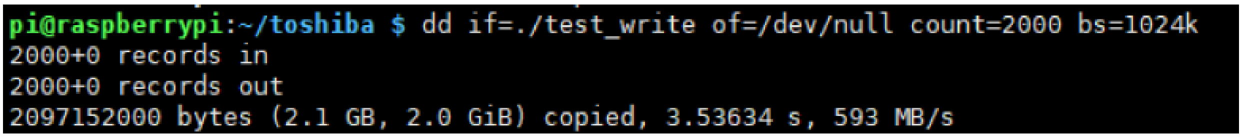
Copy Raspberry Pi memory content to the hard disk (write) sudo dd if=/dev/zero of=./test_write count=2000 bs=1024k



```
pi@raspberrypi:~/toshiba $ sudo dd if=/dev/zero of=./test_write count=2000 bs=1024k
2000+0 records in
2000+0 records out
2097152000 bytes (2.1 GB, 2.0 GiB) copied, 3.78947 s, 553 MB/s
```

Copy the hard drive content to the Raspberry Pi memory (/etc/fstab read)

sudo dd if=./test_write of=/dev/null count=2000 bs=1024k



```
pi@raspberrypi:~/toshiba $ dd if=./test_write of=/dev/null count=2000 bs=1024k
2000+0 records in
2000+0 records out
2097152000 bytes (2.1 GB, 2.0 GiB) copied, 3.53634 s, 593 MB/s
```

Note: The test results vary for different cards and environments. The Raspberry Pi is significantly affected. If you want to test accurate performance, use a PC for the test

Auto mount

Test shows there’s no issue. If it’s not required to be used as a system disk, but only for expanding the disk, set it to auto-mount

```
sudo nano /etc/fstab


#Add at the end

/dev/nvme0n1p1  /home/pi/toshiba  ext4  defaults  0  0

#/dev/nvme0n1p1 is the device name, /home/pi/toshiba refers to mounting to a directory, ext4 is the file system type, defaults uses the default mount option

#Make the changes take effect (reboot only after testing, otherwise it will fail to mount and boot)

sudo mount -a


#Then reboot

Check the device with lsblk
```

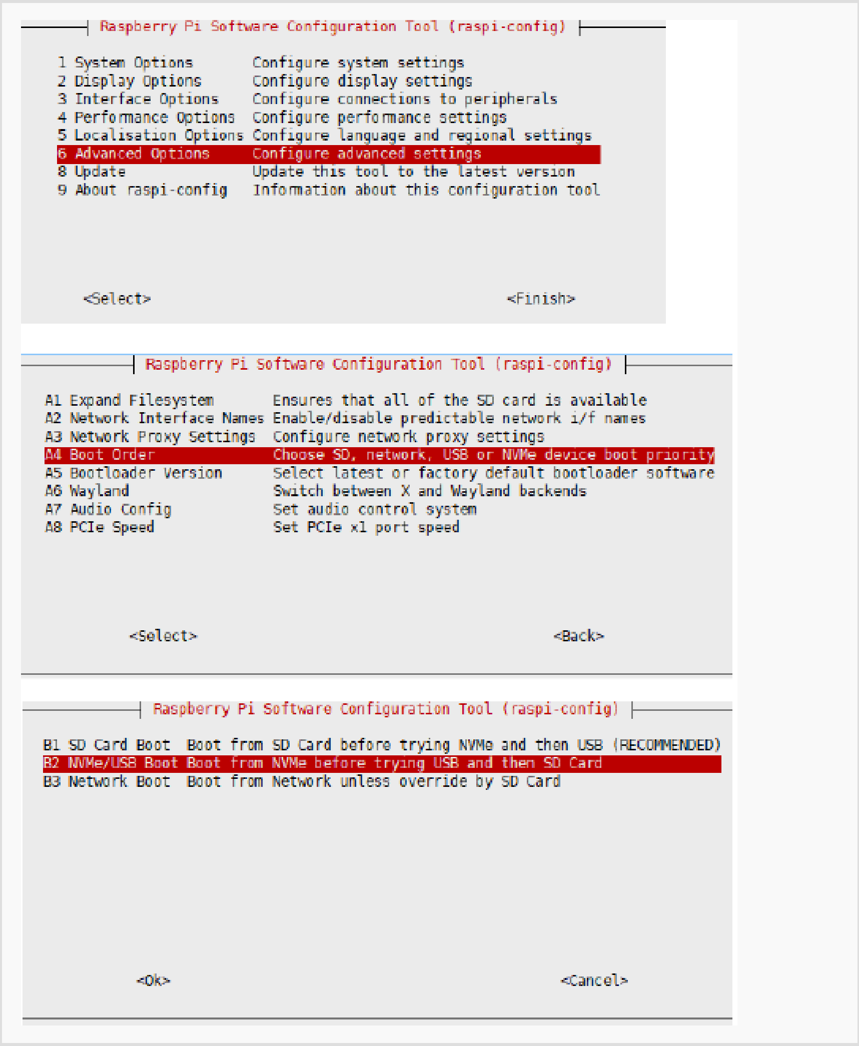
NVMe SSD boot

Boot the Raspberry Pi with a TF card first, mount and test it, and make sure the hardware can work properly

Choose one of the following methods

1. Method 1
   1. Run the following command:
   2.

      ```
      sudo raspi-config
      ```



```
┤ 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>
```

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

 A1 Expand Filesystem       Ensures that all of the SD card is available
 A2 Network Interface Names Enable/disable predictable network i/f names
 A3 Network Proxy Settings  Configure network proxy settings
 A4 Boot Order              Choose SD, network, USB or NVMe device boot priority
 A5 Bootloader Version      Select latest or factory default bootloader software
 A6 Wayland                 Switch between X and Wayland backends
 A7 Audio Config            Set audio control system
 A8 PCIe Speed              Set PCIe x1 port speed

         <Select>                                  <Back>
```

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

 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

          <Ok>                                     <Cancel>
```

Reboot Raspberry Pi

If you find you can't modify it multiple times, please reconnect to the network and then try to modify it (wait for the network to self-calibrate), or modify the file after setting the correct time

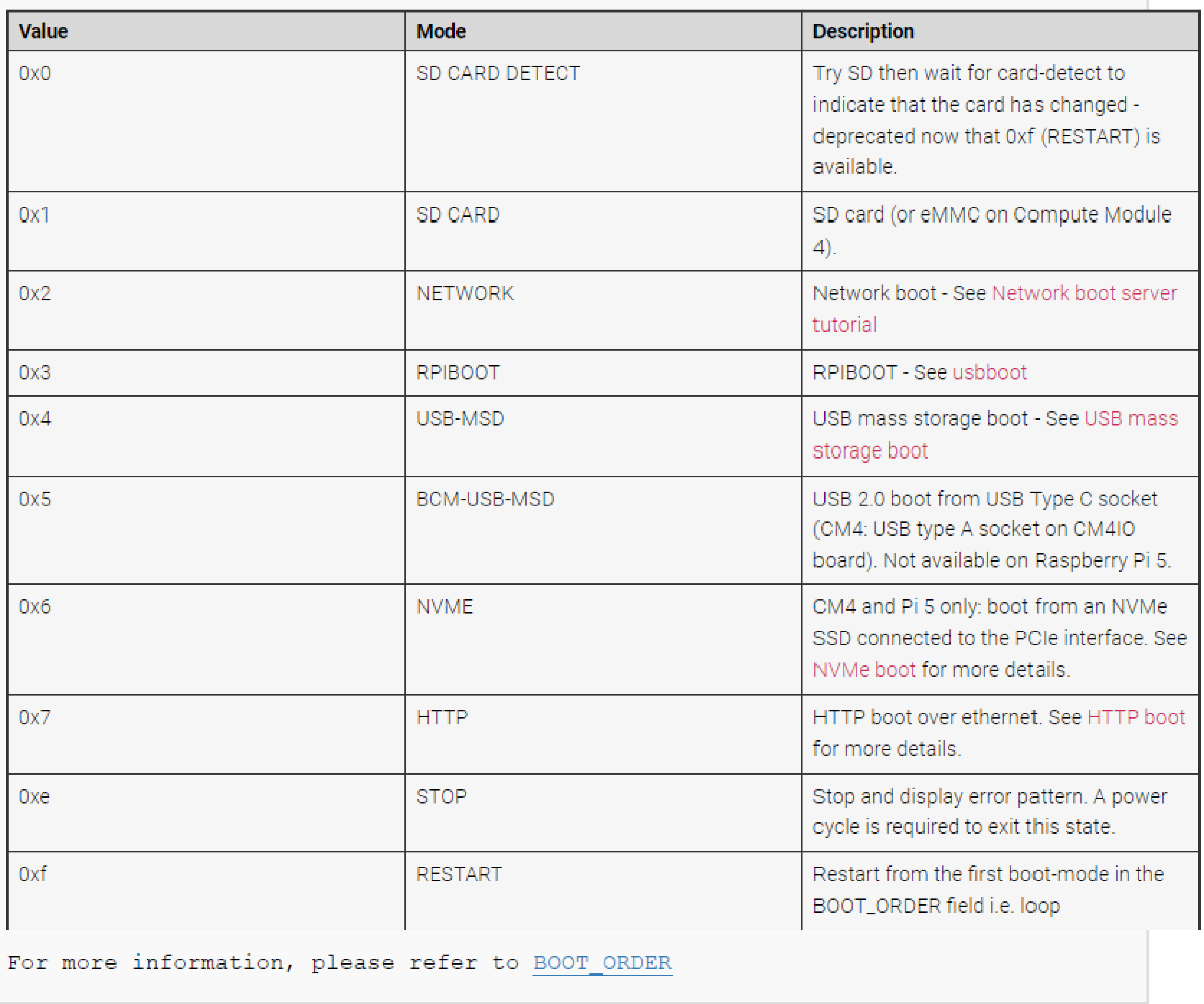
3. Flash the system to NVME, then connect the NVME to the expansion board, remove the TF card and power it on again

2. Method 2

Modify the BOOT_ORDER in the Raspberry Pi boot loader configuration

sudo rpi-eeprom-config –edit

1. Modify BOOT_ORDER=0xf41 to BOOT_ORDER=0xf416



Value	Mode	Description
0x0	SD CARD DETECT	Try SD then wait for card-detect to indicate that the card has changed - deprecated now that 0xf (RESTART) is available.
0x1	SD CARD	SD card (or eMMC on Compute Module 4).
0x2	NETWORK	Network boot - See Network boot server tutorial
0x3	RPIBOOT	RPIBOOT - See usbboot
0x4	USB-MSD	USB mass storage boot - See USB mass storage boot
0x5	BCM-USB-MSD	USB 2.0 boot from USB Type C socket (CM4: USB type A socket on CM4IO board). Not available on Raspberry Pi 5.
0x6	NVME	CM4 and Pi 5 only: boot from an NVMe SSD connected to the PCIe interface. See NVMe boot for more details.
0x7	HTTP	HTTP boot over ethernet. See HTTP boot for more details.
0xe	STOP	Stop and display error pattern. A power cycle is required to exit this state.
0xf	RESTART	Restart from the first boot-mode in the BOOT_ORDER field i.e. loop

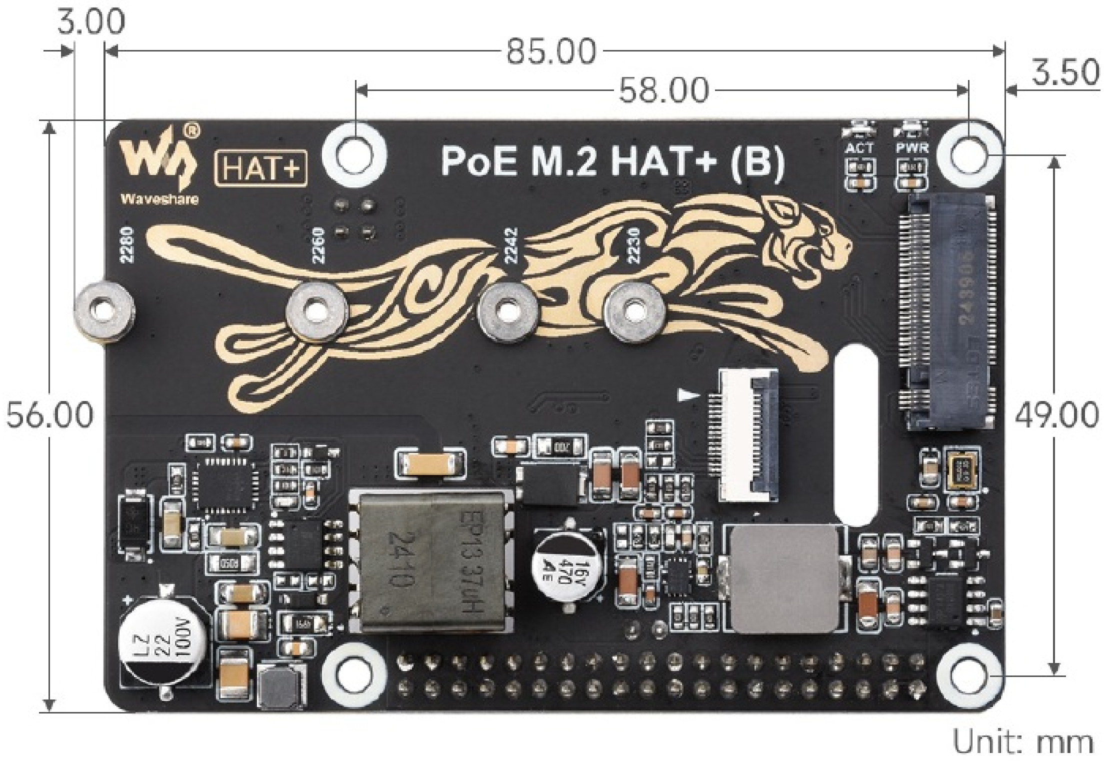
```
For more information, please refer to BOOT_ORDER
```

2. Reboot Raspberry Pi

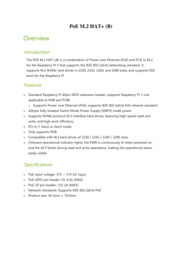
   If you find you can't modify it multiple times, please reconnect to the network and then try to modify it (wait for the network to self-calibrate), or modify the file after setting the correct time

3. Flash the system to NVME, then connect the NVME to the expansion board, remove the TF card and power it on again

Dimensions



Documents / Resources

 PoE M.2 HAT+ (B) Overview Introduction The POE M.2 HAT+(B) is a combination of Power over Ethernet (PoE) and PCIE to M.2 for the Raspberry Pi 5 that supports the IEEE 802.3af/at networking standard. It supports M.2 NVMe hard drives in 2230, 2242, 2260, and 2280 sizes, and supports SSD boot for the Raspberry Pi Features • Standard Raspberry Pi 40pin GPIO extension header, supports Raspberry Pi 5 (not applicable to PI4B and PI3B) ◦ Supports Power over Ethernet (PoE), supports IEEE 802.3af/at PoE network standard • Adopts fully isolated Switch Mode Power Supply (SMPS) mode power • Supports NVMe protocol M.2 interface hard drives, featuring high-speed read and write, and high work efficiency • PCI-E×1 Gen2 or Gen3 mode • Only supports PI5B • Compatible with M.2 hard drives of 2230 / 2242 / 2260 / 2280 sizes • Onboard operational indicator lights, the PWR is continuously lit when powered on, and the ACT blinks during read and write operations, making the operational status easily visible Specifications • PoE input voltage: 37V ~ 57V DC input • PoE-GPIO pin header: 5V 4.5A (MAX) • PoE-2P pin header: 12V 2A (MAX) • Network standards: Supports IEEE 802.3af/at PoE • Product size: 56.5mm × 70.0mm	Raspberry Pi PCIe To M.2 Adapter With PoE Function [pdf] Instructions PCIe To M.2 Adapter With PoE Function, M.2 Adapter With PoE Function, Adapter With PoE Function, PoE Function, Function

References

- User Manual

Raspberry Pi

Adapter With PoE Function, function, M.2 Adapter With PoE Function, PCIe To M.2 Adapter With PoE Function, PoE Function, Raspberry Pi

Leave a comment

Your email address will not be published. Required fields are marked *

Comment *

Name

Email

Website

□ Save my name, email, and website in this browser for the next time I comment.

Post Comment

Search:

e.g. whirlpool wrf535swhz

Search