Embedded Linux Systems Programming: Introduction

Norman McEntire

Outline

- Welcome!
- Concepts
- Case Studies

References

- https://en.wikipedia.org/wiki/Embedded_system
- https://en.wikipedia.org/wiki/Linux_on_embedded_systems
- https://www.raspberrypi.com/

What is an Embedded Linux System?

What is an embedded system?

- An **Embedded Linux System** is a **computer system** build upon the Linux operating system that has a **dedicated function** within a larger system
- Everyday Examples
  - Setup Boxes, Smart TVs, Personal Video Recorders, In-Vehicle Info Systems, Network Equipment (routers, etc), industrial equipment, medical equipment, etc. etc. etc
  - Embedded Linux Systems are everywhere!

Example Embedded Linux System

Itron Electric Meter

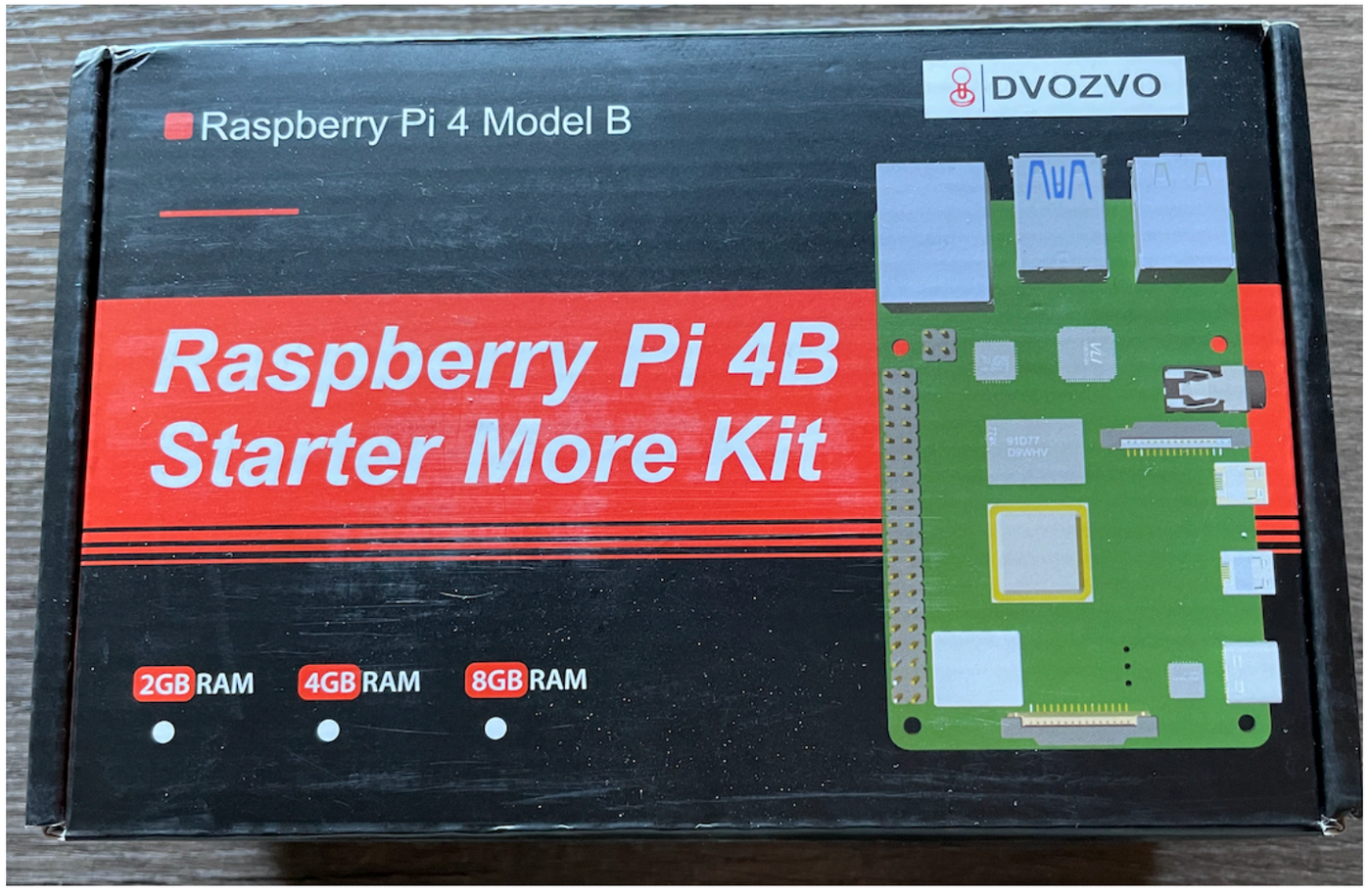
Example Embedded Linux System

Key Parts of All Embedded Linux Systems

- Boot Loader
- Linux Kernel / Linux Kernel Modules
- Root Filesystem
- Busybox (Command-Line Interface)
- Systems Programs / Daemons

Using Raspberry Pi To Explore an Embedded Linux System

Raspberry Pi 4B Starter More Kit

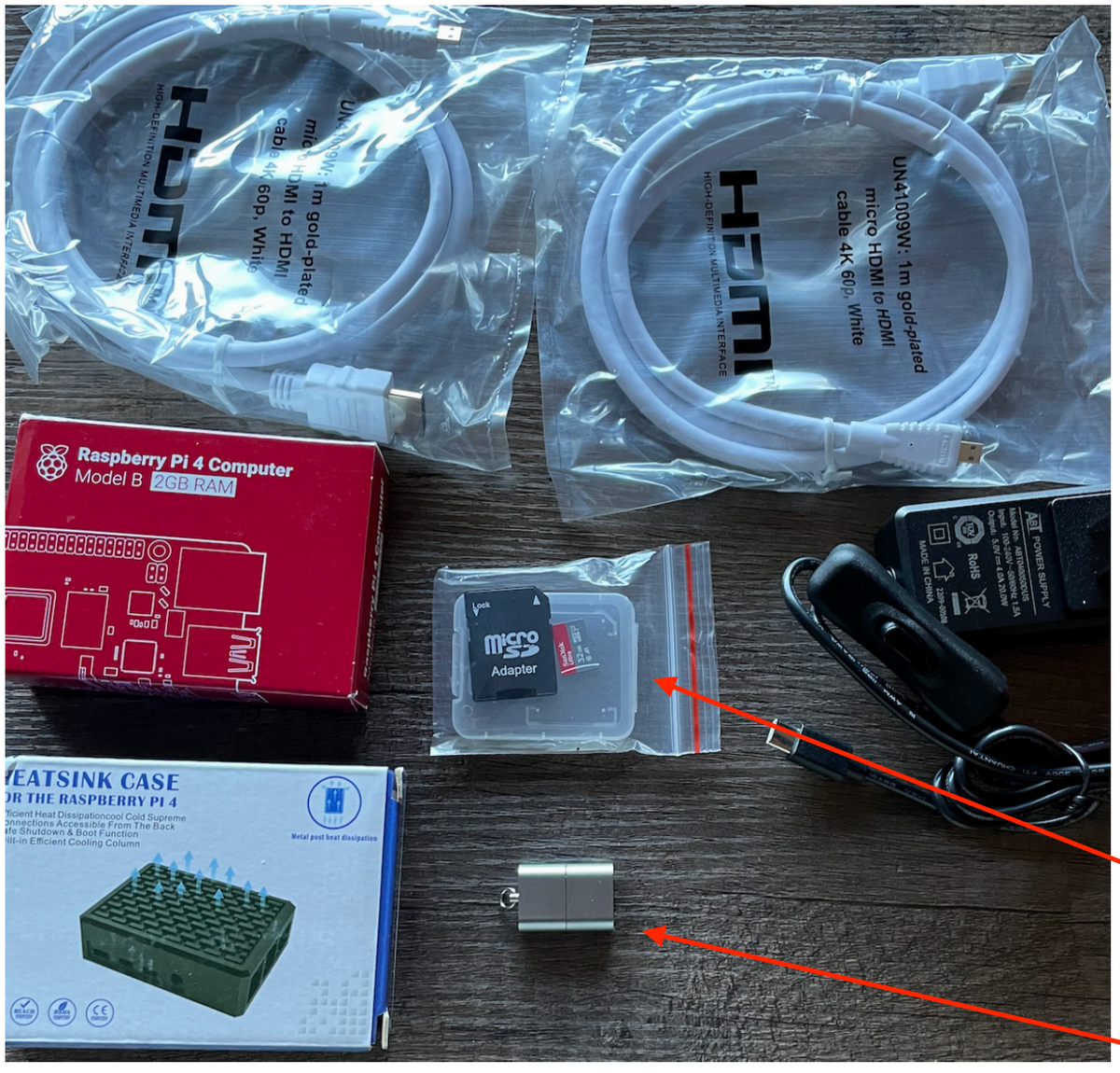


Raspberry Pi 4B Starter More Kit - Items Inside Box

2X HDMI Cables →

Raspberry Pi 4B W/ 2GB RAM →

Heatsink →

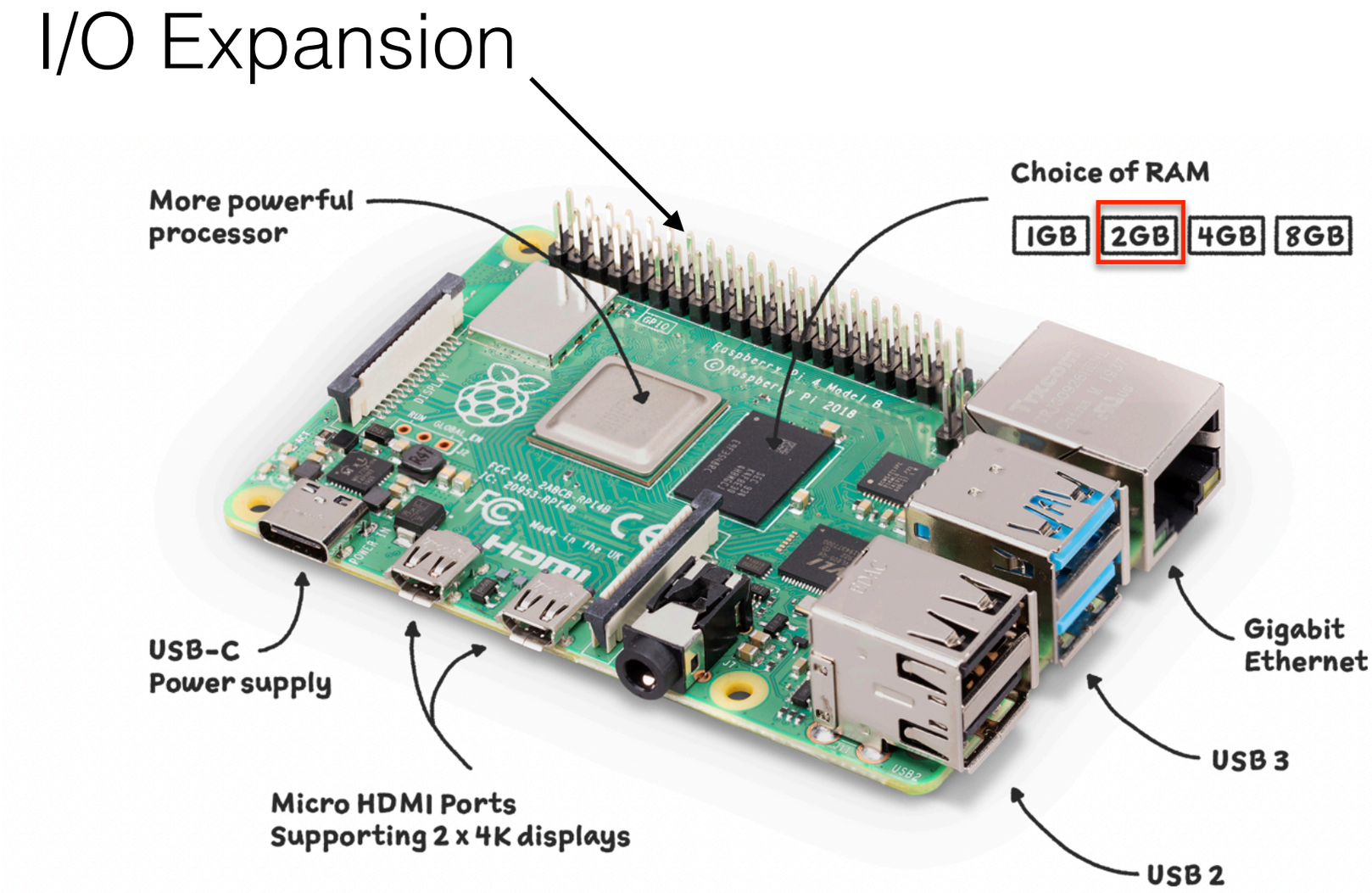


← Power Supply W/On/Off Switch

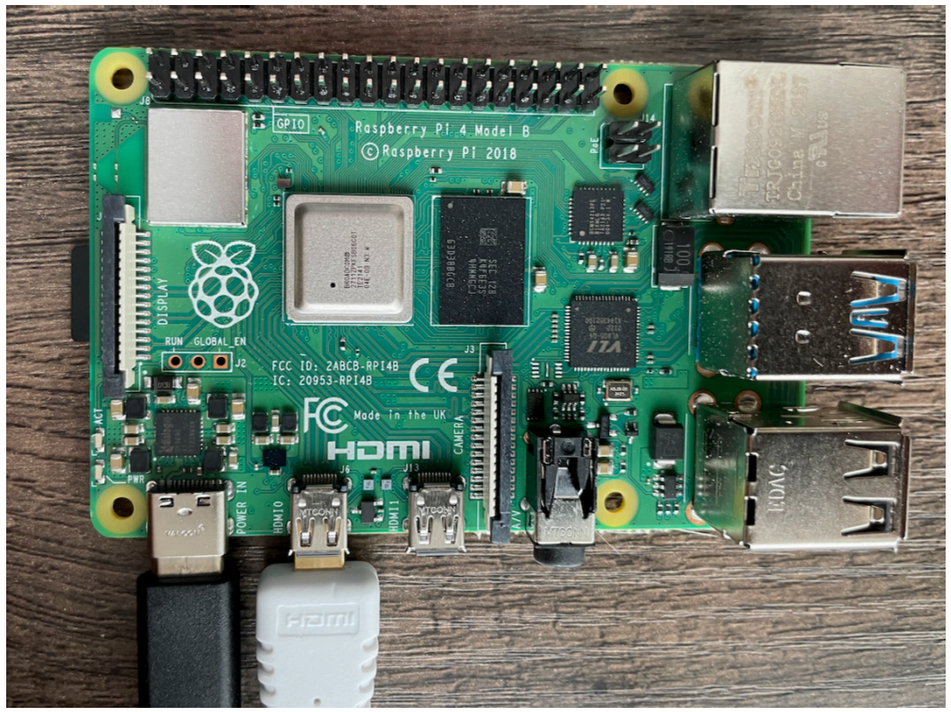
Micro SD Car For Booting

USB Storage

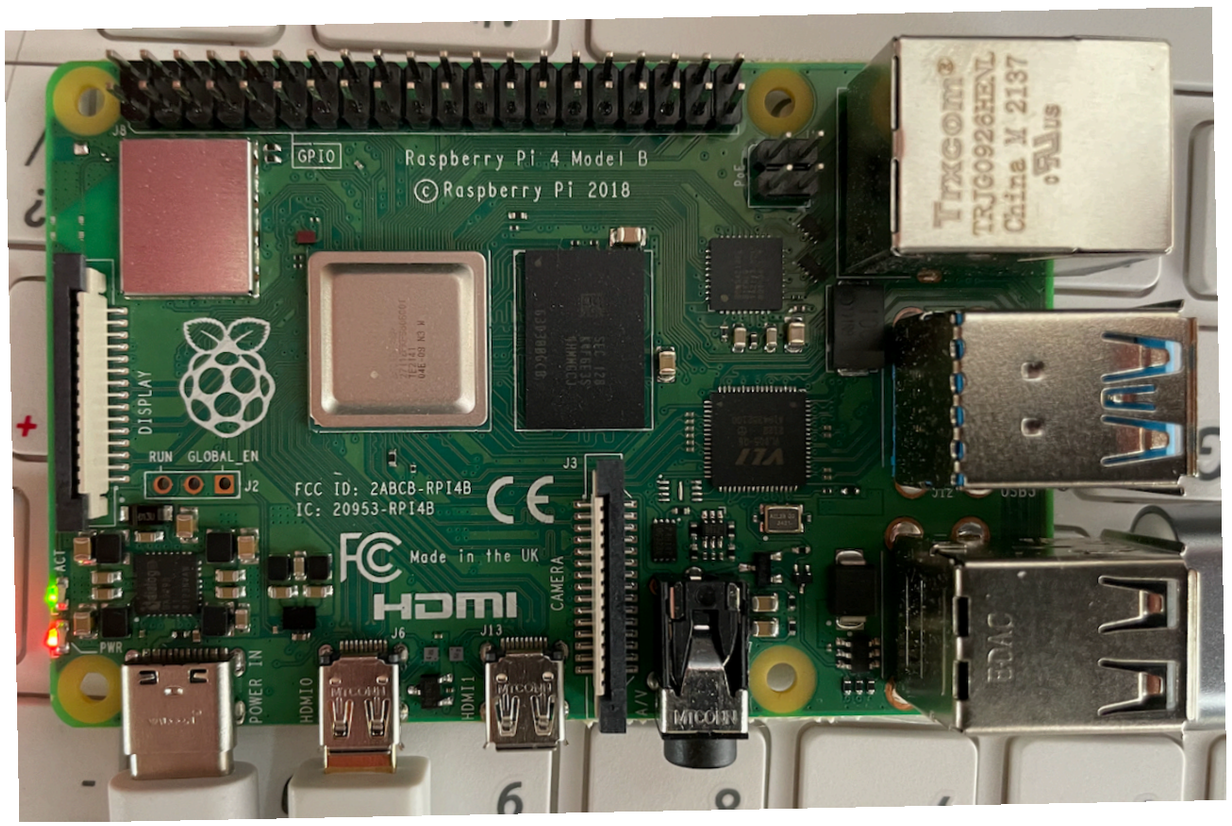
Raspberry Pi 4B



Connect Video and Power Cables

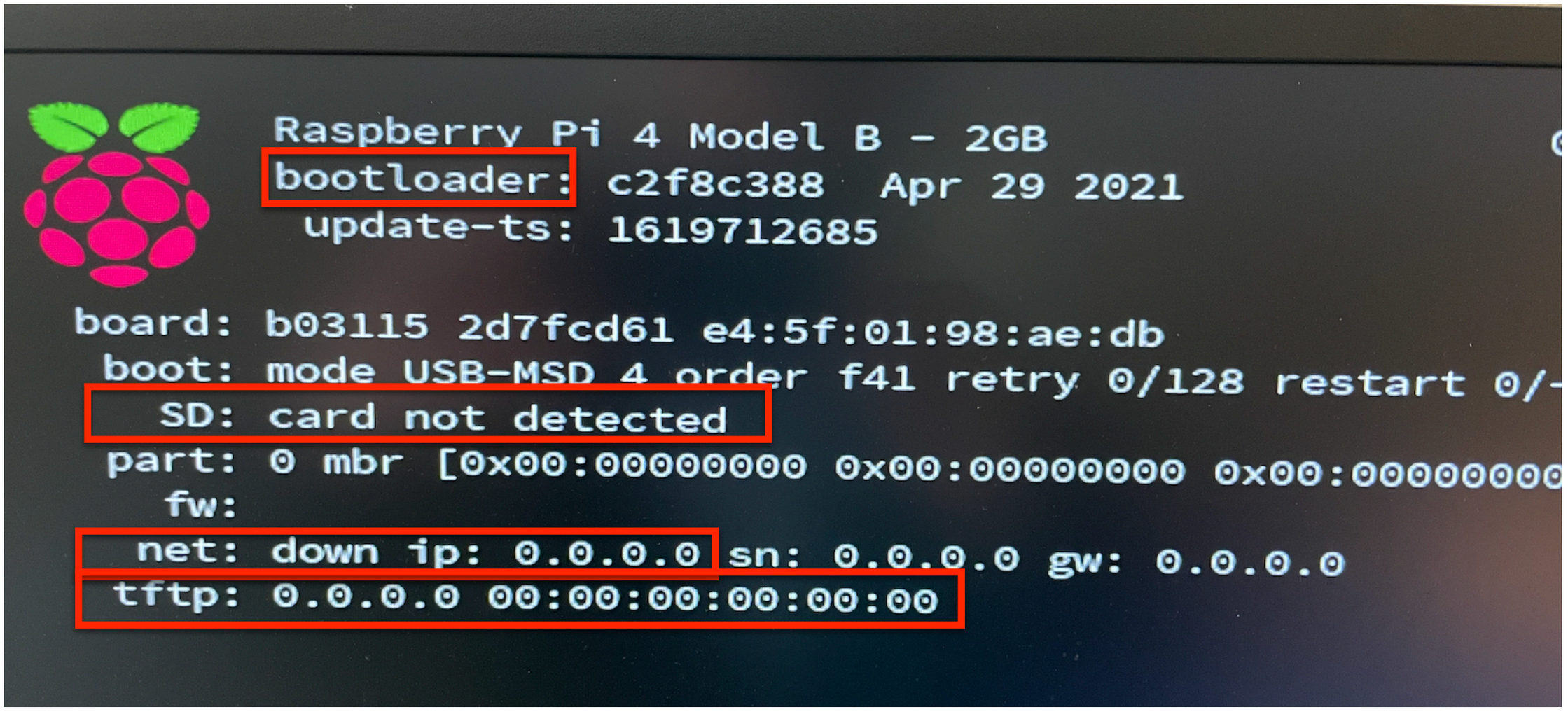


Power Video



LEDs

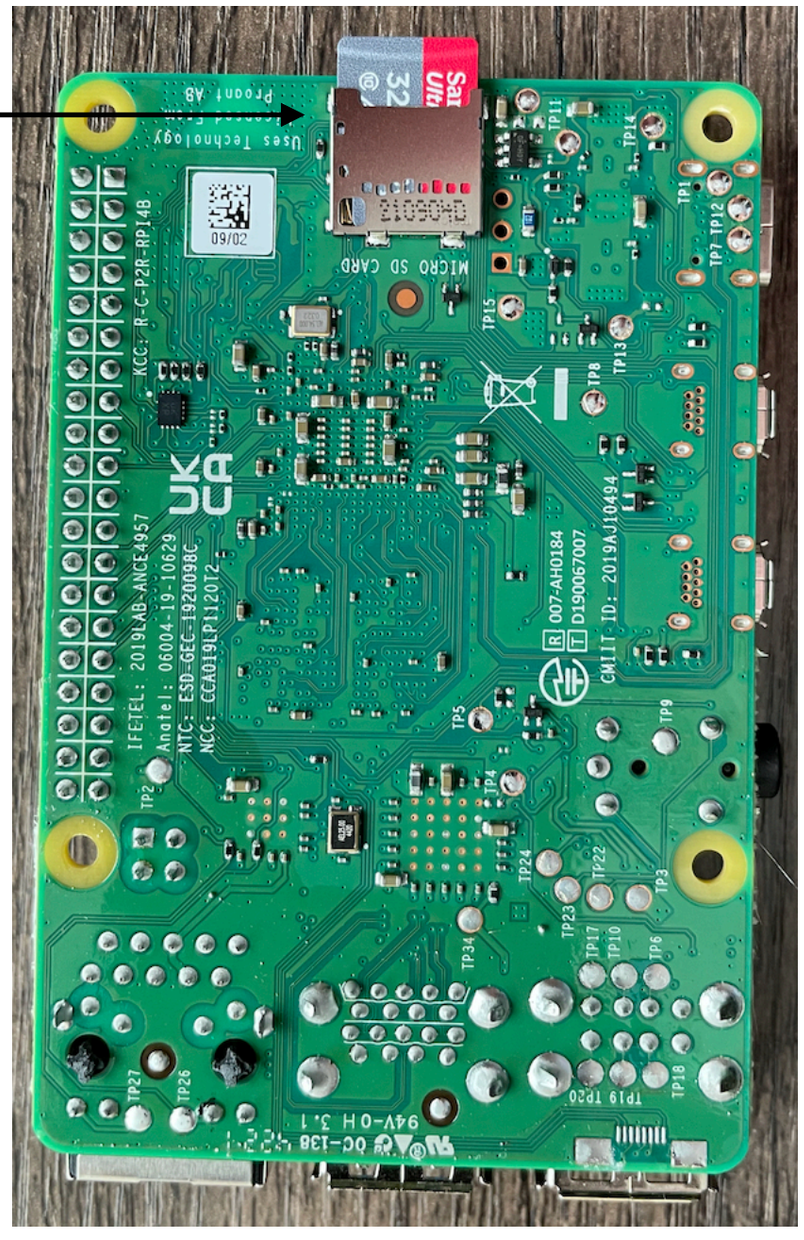
Initial Message After Power (No SD Card Inserted)



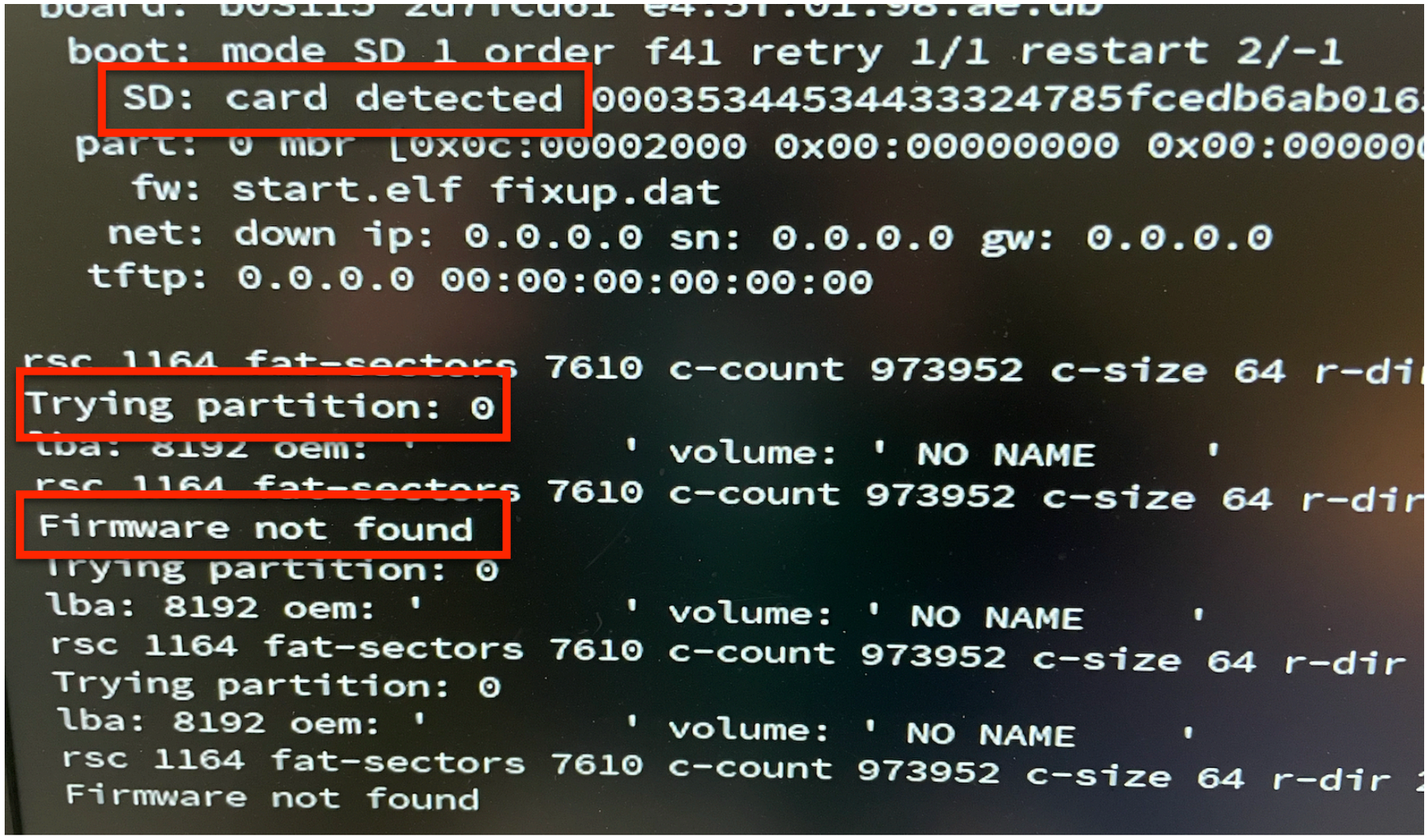
tftp : Trivial File Transfer Protocol

Insert Micro SD Card (Although not yet formatted)

Micro SD Card

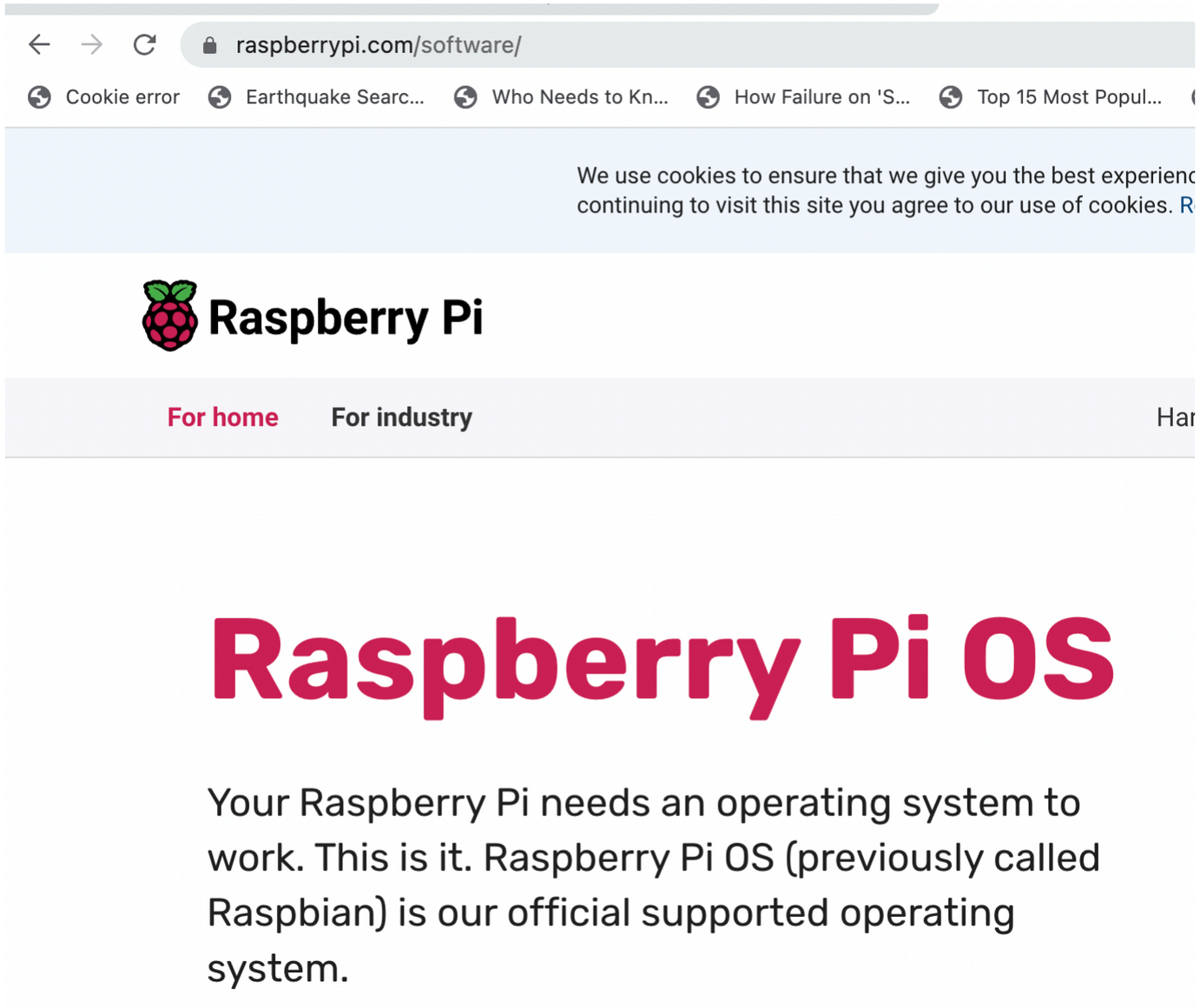


Power
(But SDCard Not Formatted)



To Download Raspberry Pi Firmware

https://www.raspberrypi.com/software/



Raspberry Pi Imager

Install Raspberry Pi OS using Raspberry Pi Imager

Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi. Watch our 45-second video to learn how to install an operating system using Raspberry Pi Imager.

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

Raspberry Pi Imager

Step 1

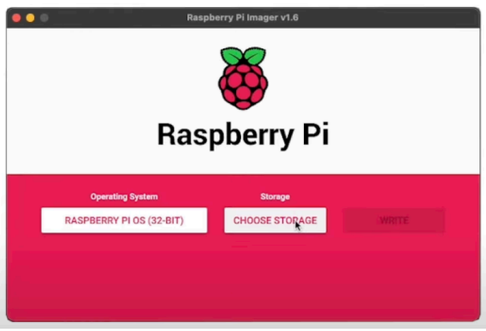
Insert SD card

Step 2

Choose Operating System

Step 3

Choose SD card

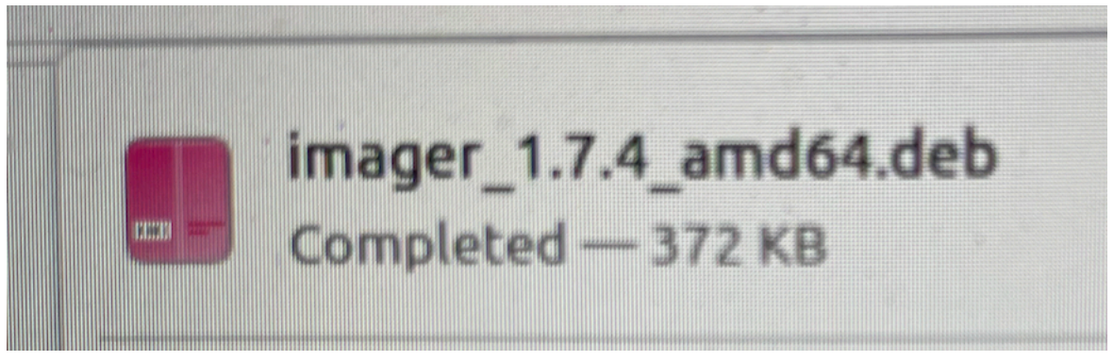


Step 4

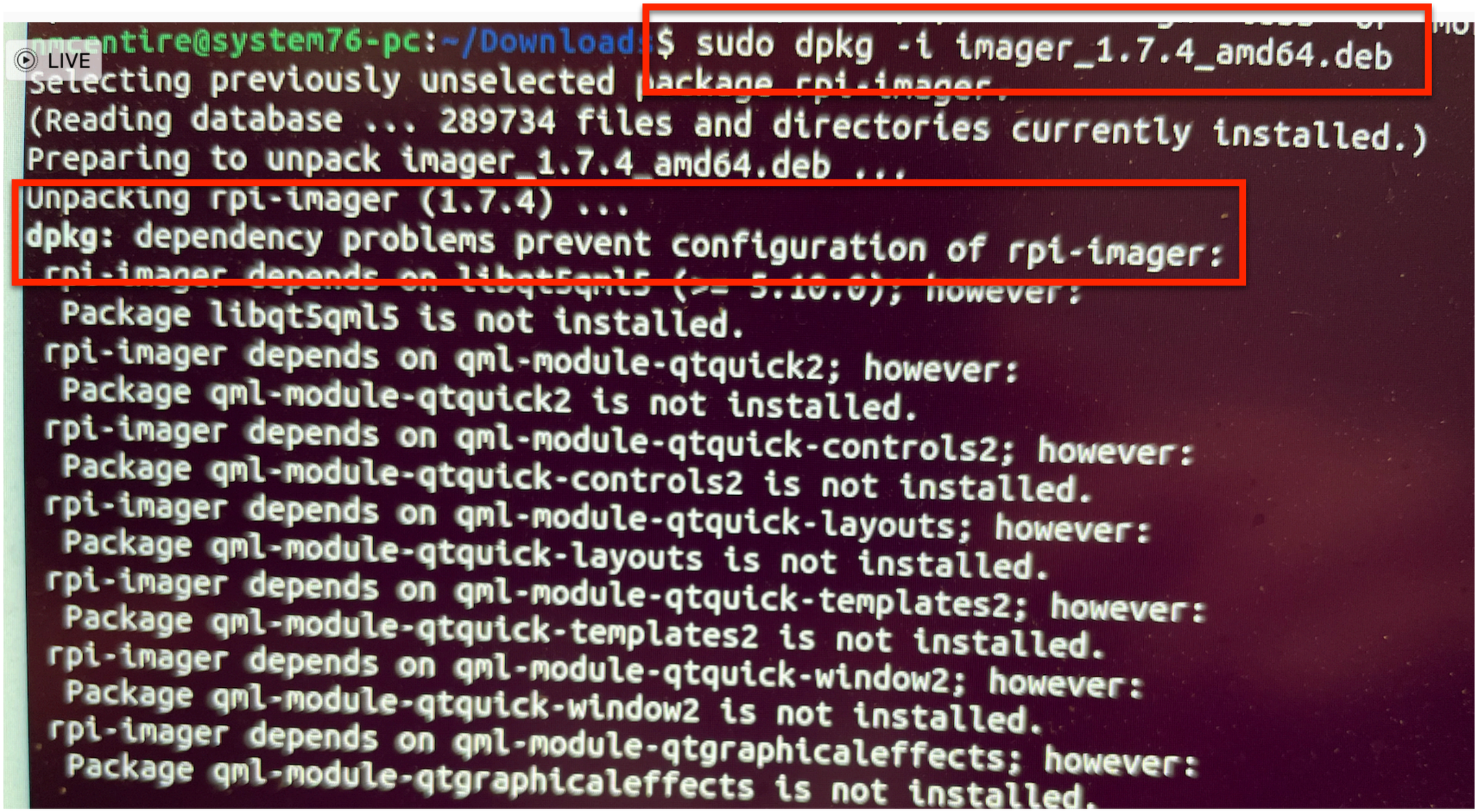
Write and verify SD card

Preparing Micro SD Card

Download Imager (On Ubuntu 22.04)



First Install Attempt (Using dpkg)



Transition

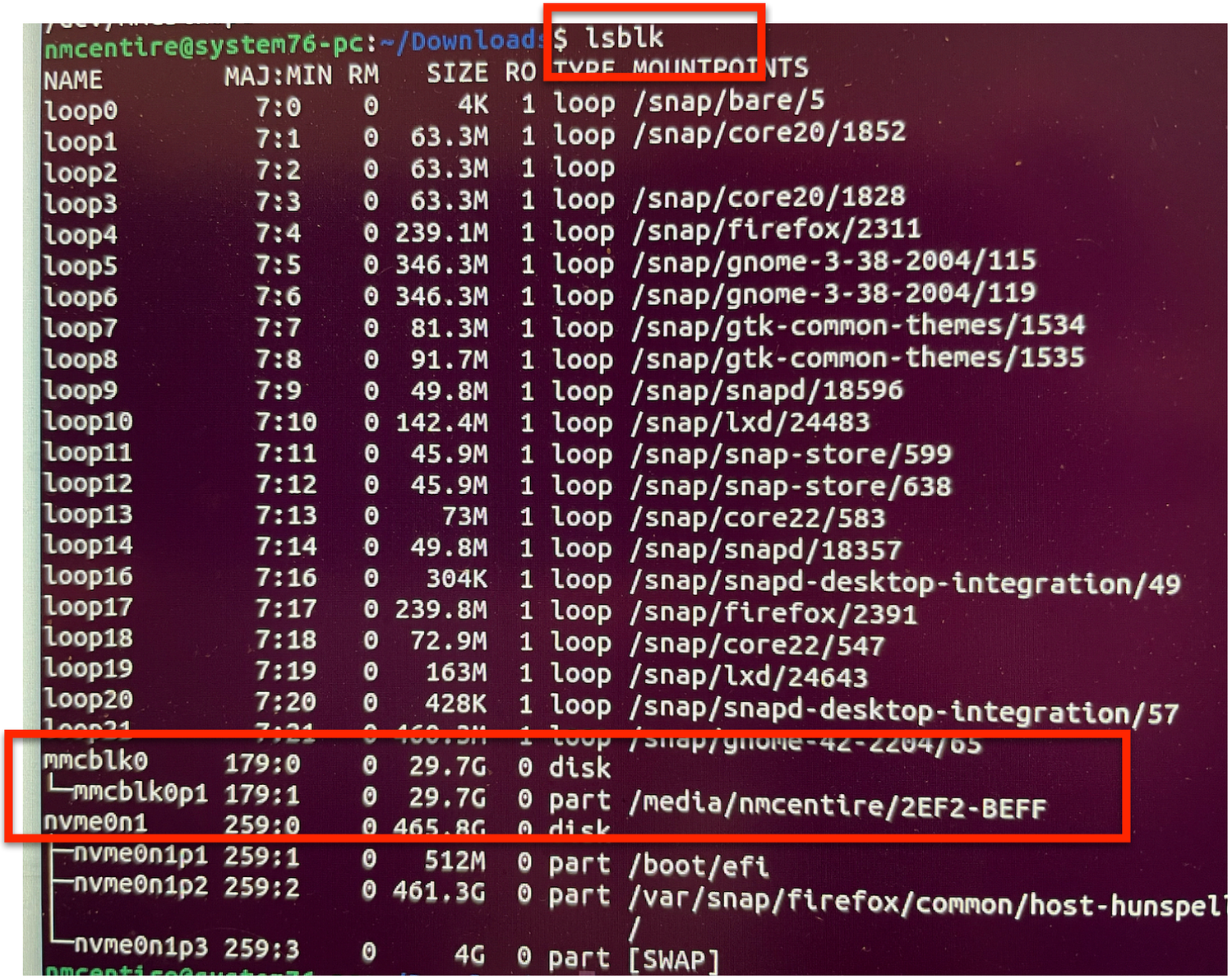
- Embedded Linux often uses the dpkg (Debian Package) command to install .deb packages
- However, the dpkg command does NOT resolve dependencies
- To resolve dependences, use the apt command
  - However the .deb must be installed into a local repo (repository) to do that
    - Instead of doing that here, I decide to use the dd (disk direct) copy command

First, notice how Micro SD Card appears on Linux

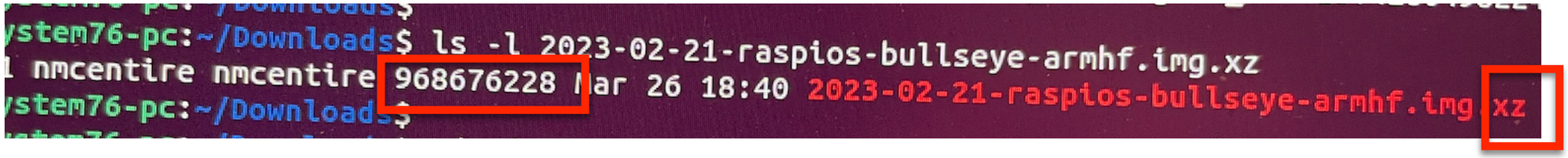
```
nmcentire@system76-pc:~/Downloads$ df /media/nmcentire/2EF2-BEFF/
Filesystem     1K-blocks  Used Available Use% Mounted on
/dev/mmcblk0p1  31166464    32  31166432   1% /media/nmcentire/2EF2-BEFF
nmcentire@system76-pc:~/Downloads$
```

df = Disk Filesystem Space

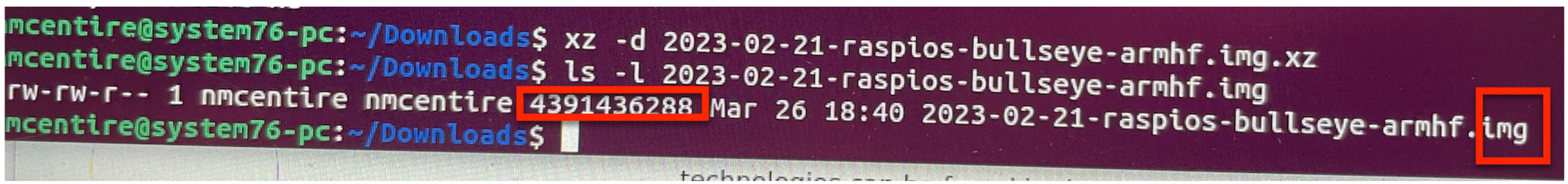
Use lsblk to see your block devices



Size of Image - Before and After Decompression



About 1GB



About 4GB

Using dd to send image to Micro SD Card

```
s$ sudo dd if=2023-02-21-raspios-bullseye-armhf.img of=/dev/mmcblk0 bs=4M conv=fsync status=progress
```

dd = disk direct

if = input file

of = output file

bs = block size

conv=convert (fsync says do not end command until all are copied)

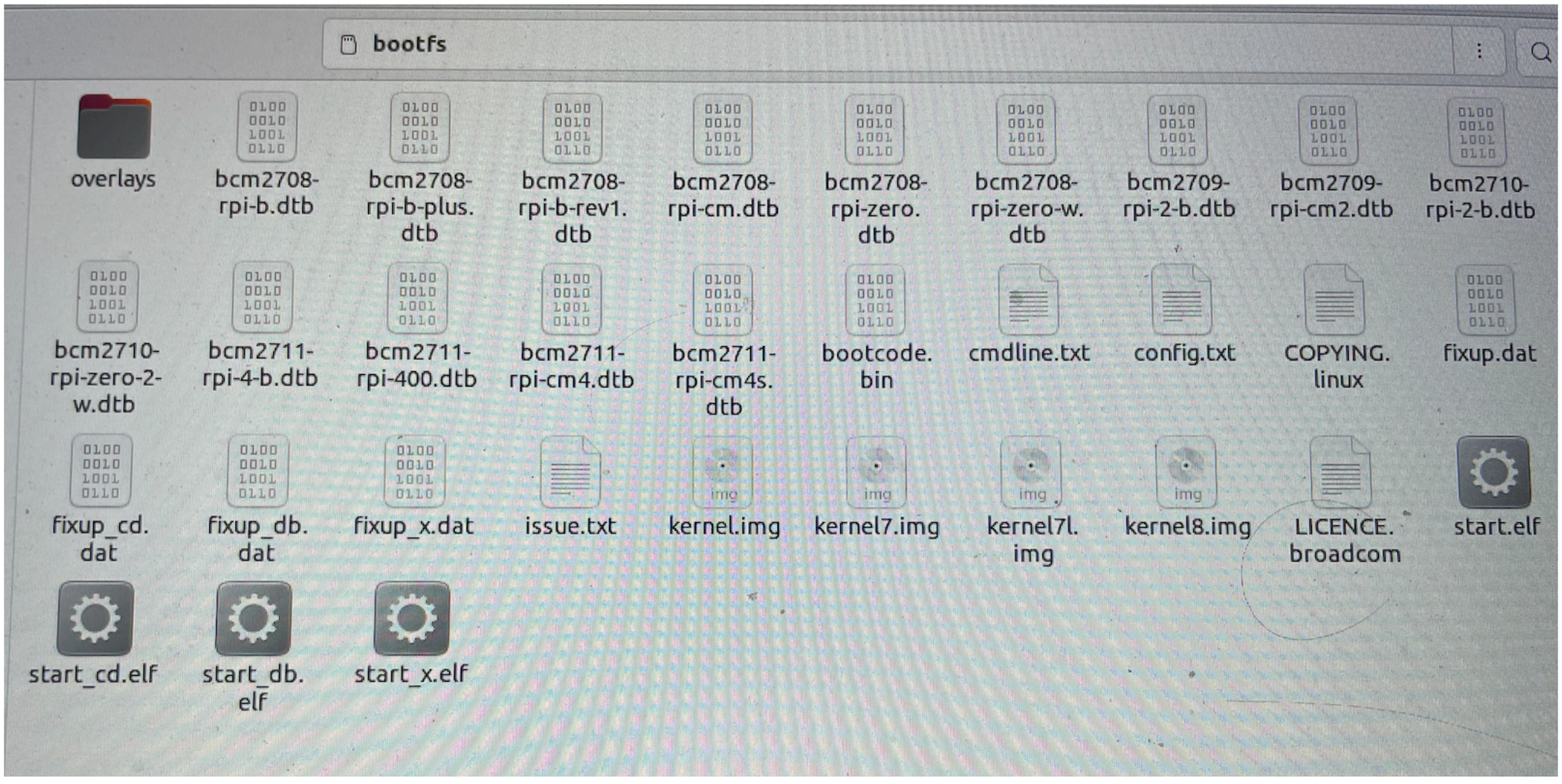
status=progress (show progress)

Showing Progress

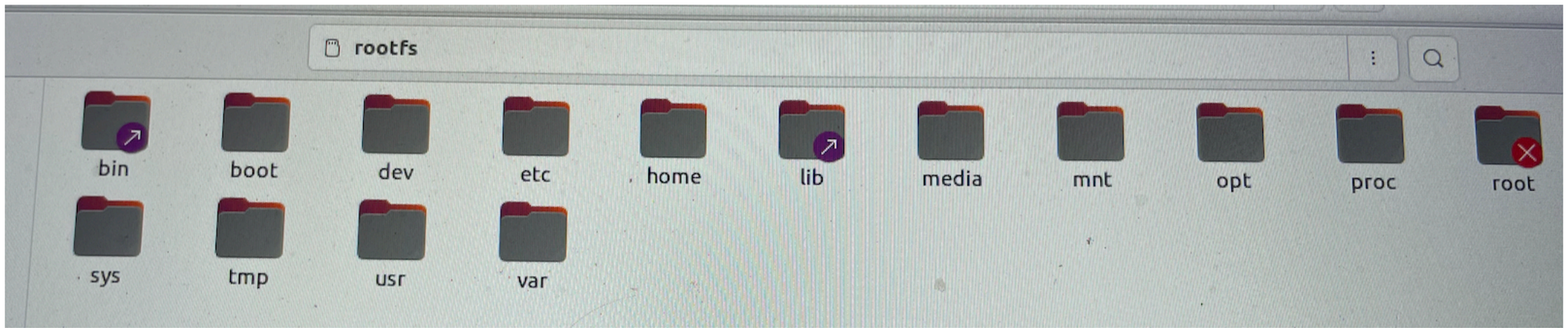
```
4378853376 bytes (4.4 GB, 4.1 GiB) copied, 95 s, 46.0 MB/s

1047+0 records in
1047+0 records out
4391436288 bytes (4.4 GB, 4.1 GiB) copied, 216.705 s, 20.3 MB/s
```

Results - Partition 1 bootfs (Boot Filesystem)

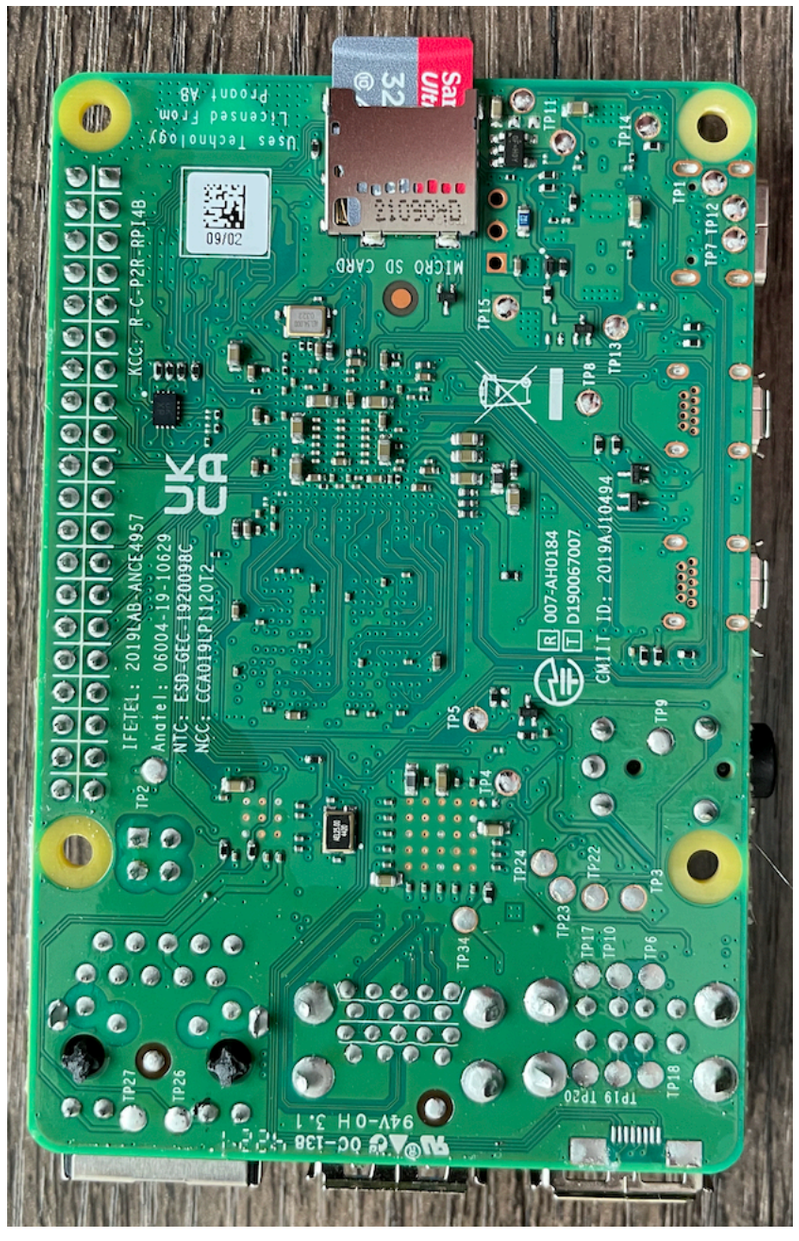


Results - Partition 2 rootfs (Root Filesystem)

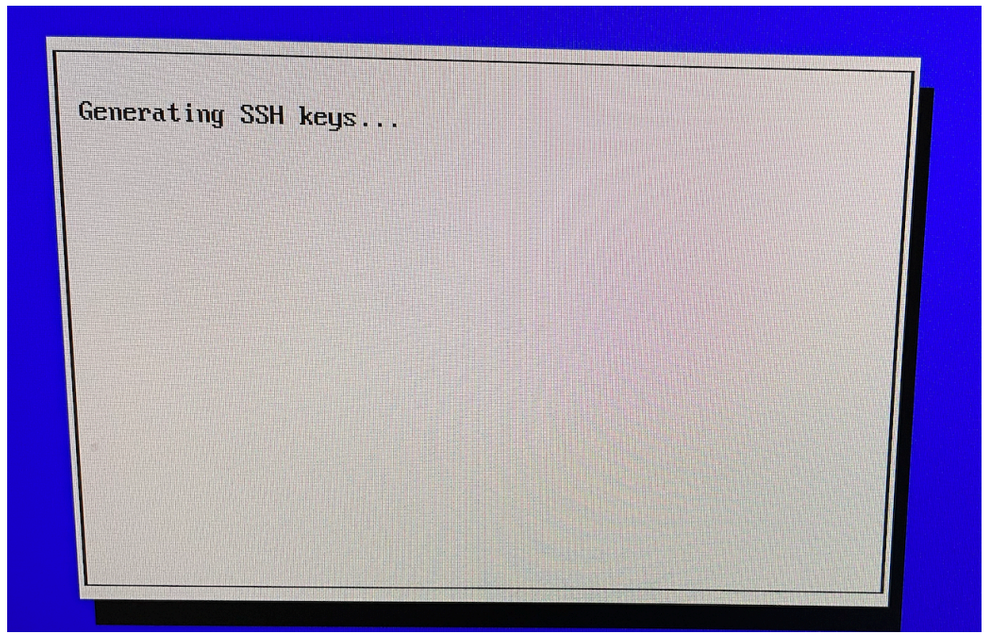


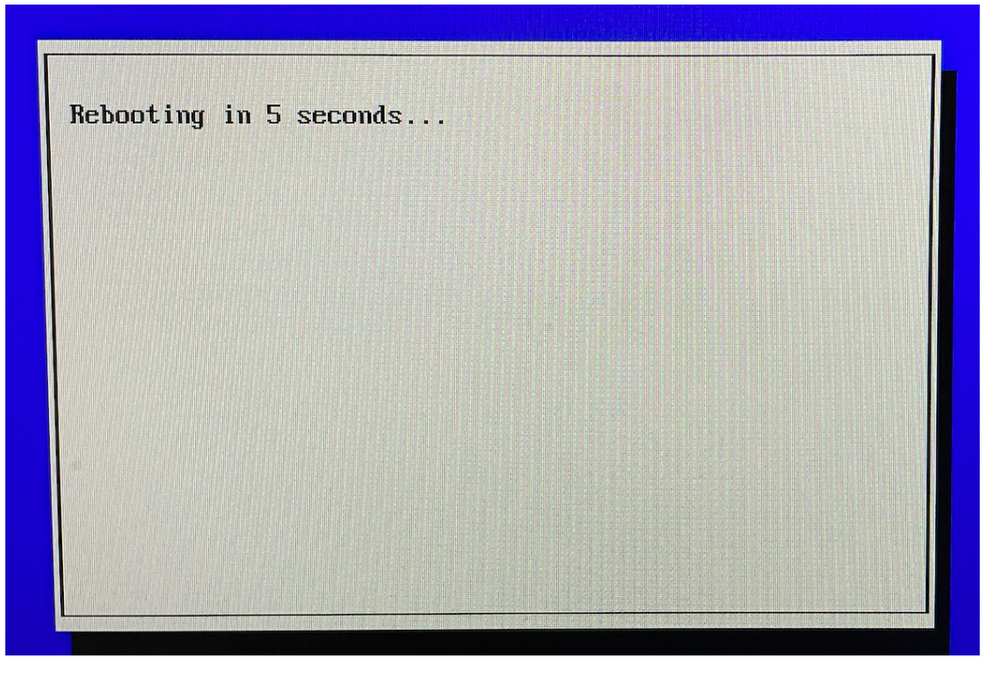
You can now remove the Micro SD Card from Ubuntu and Insert into the Raspberry Pi

Micro SD Card Inserted into Raspberry Pi

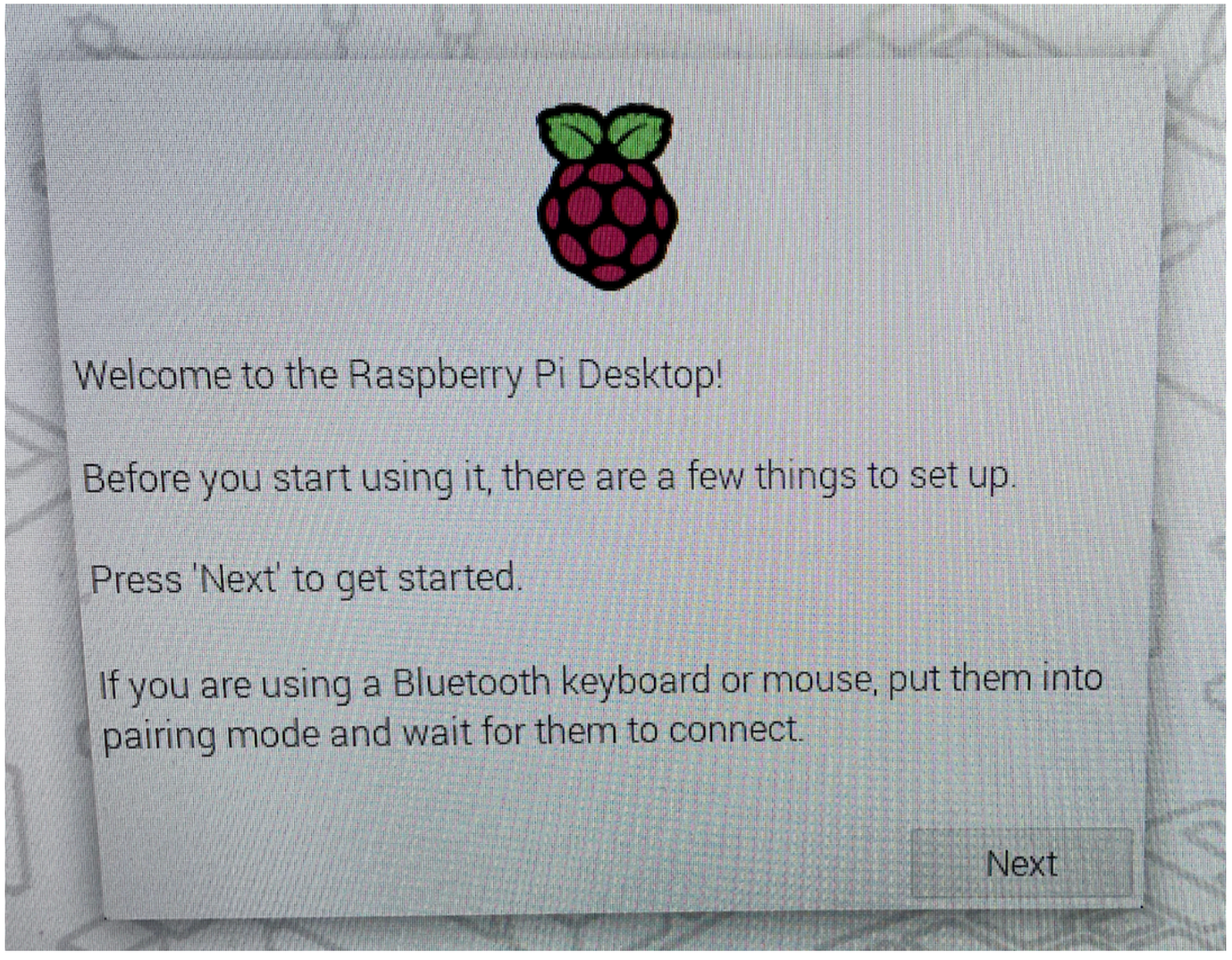


Booting Up First Time

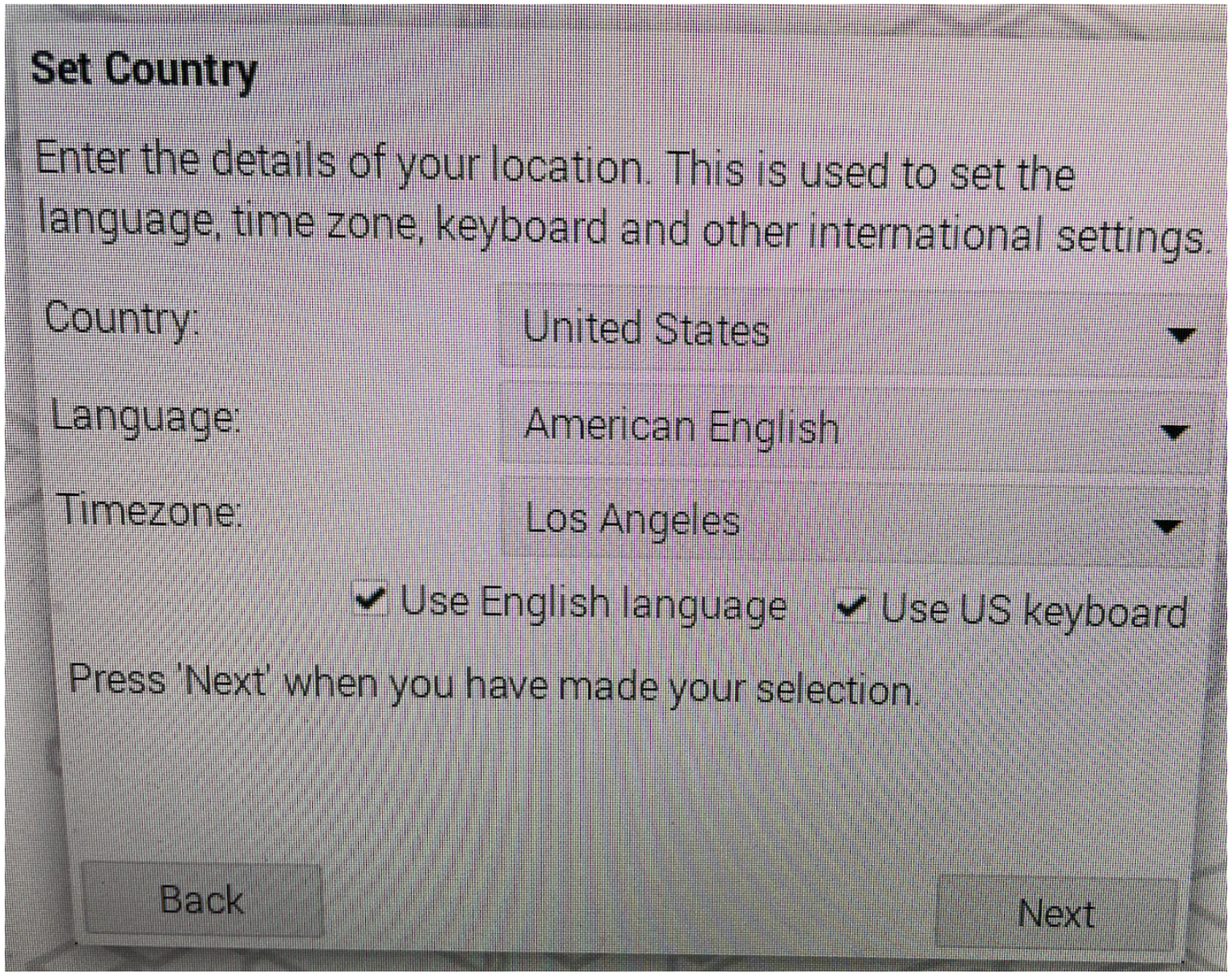




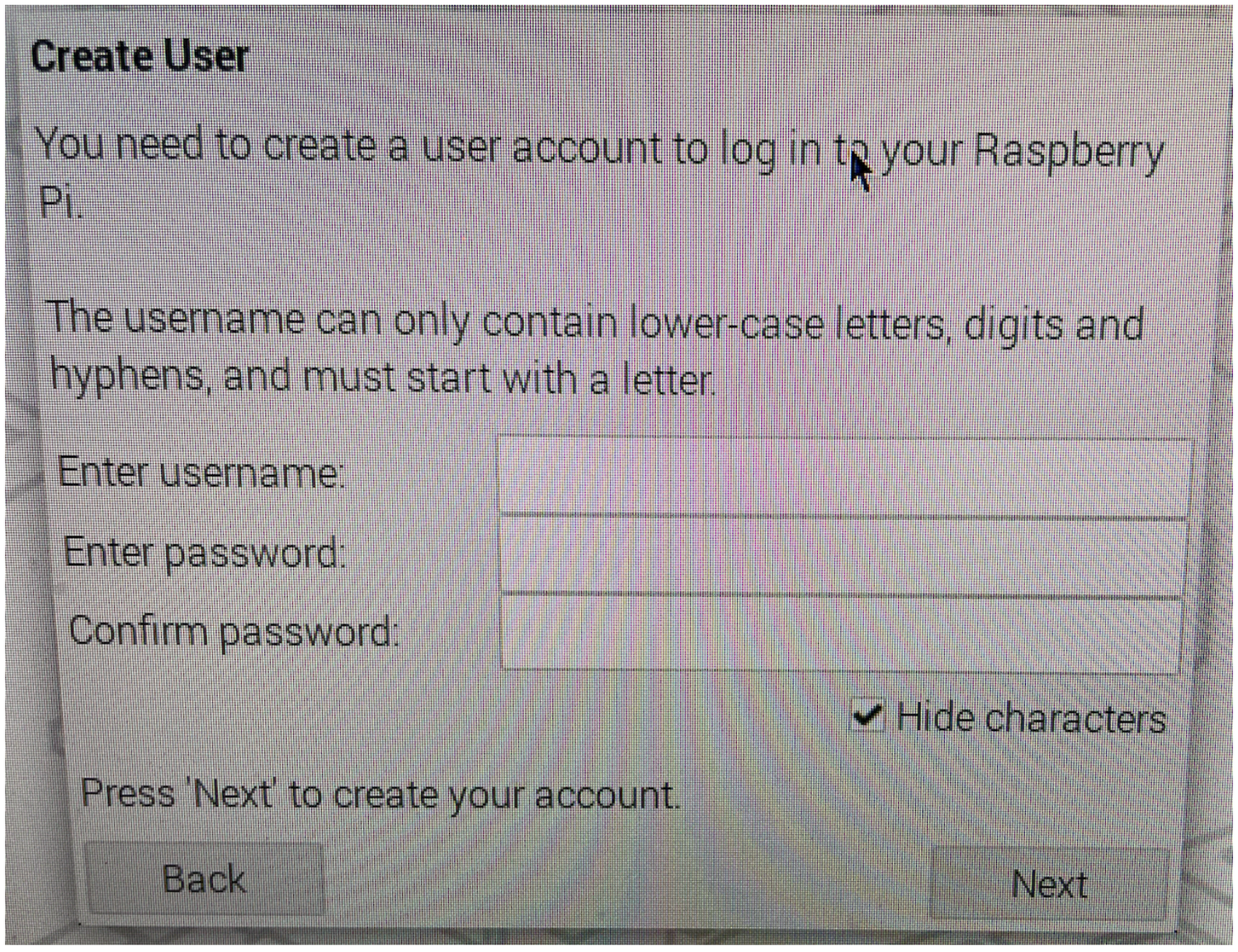
Setup Screen - Intro



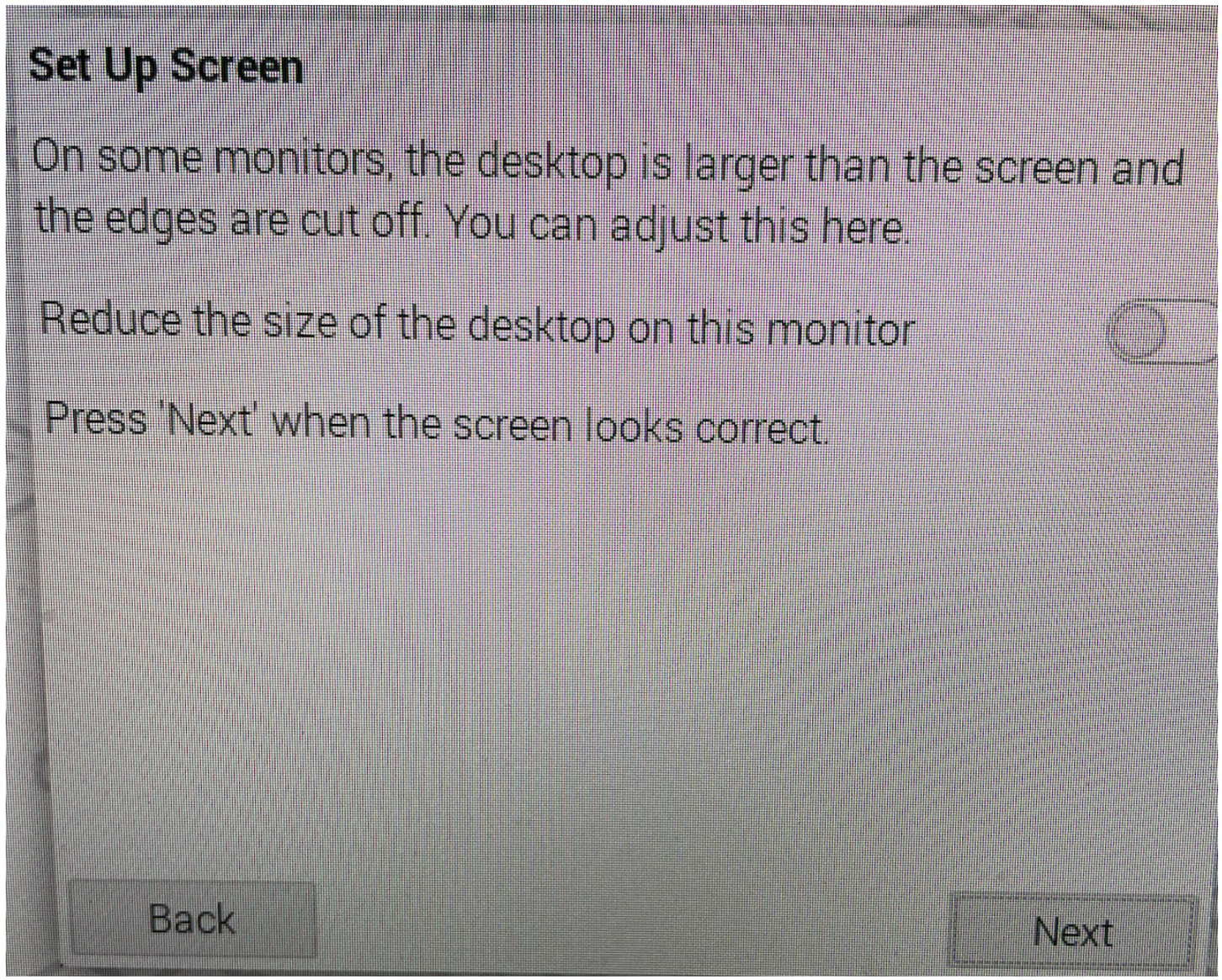
Setup Screen - Country



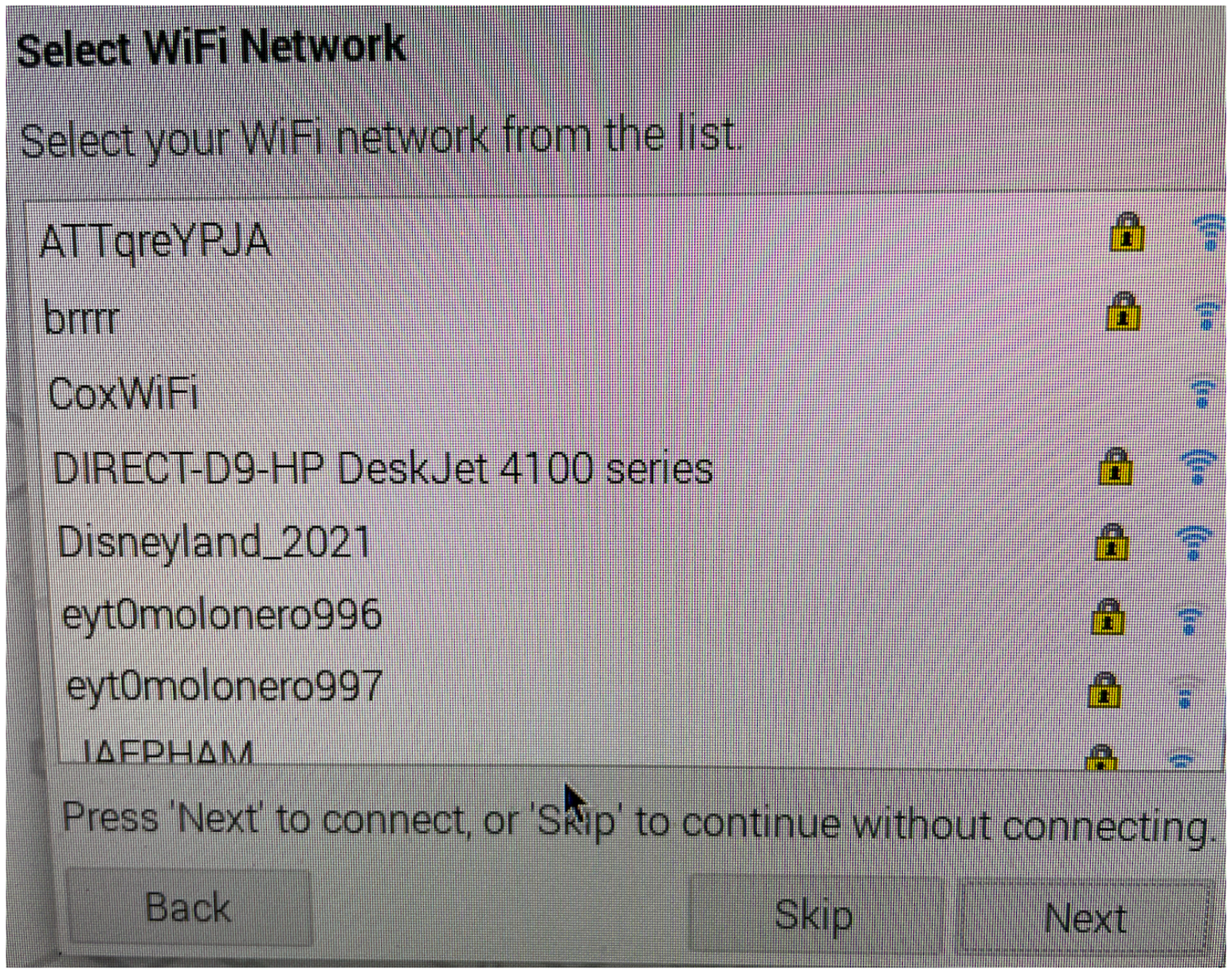
Setup Screen - Create User



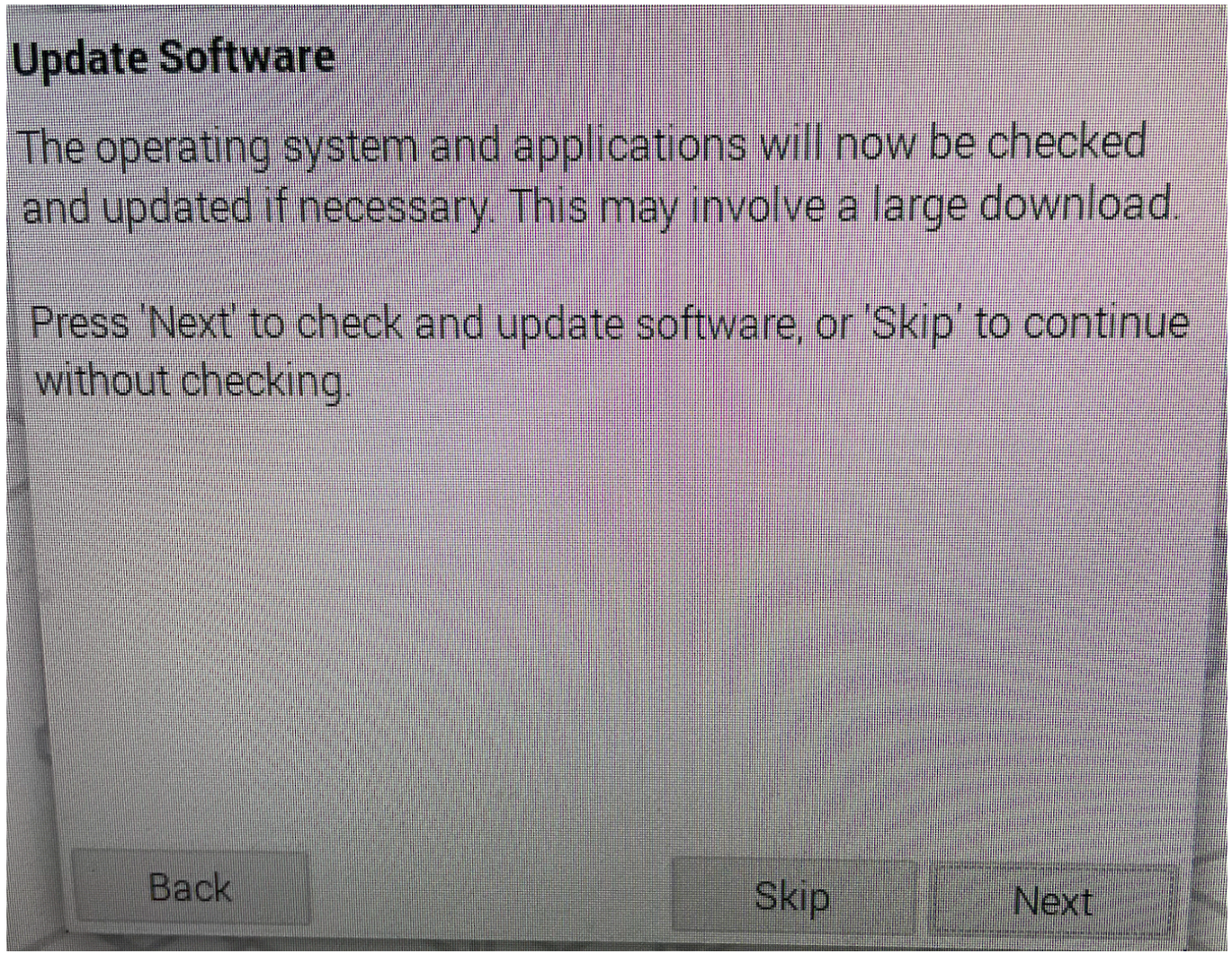
Setup Screen - Setup Screen



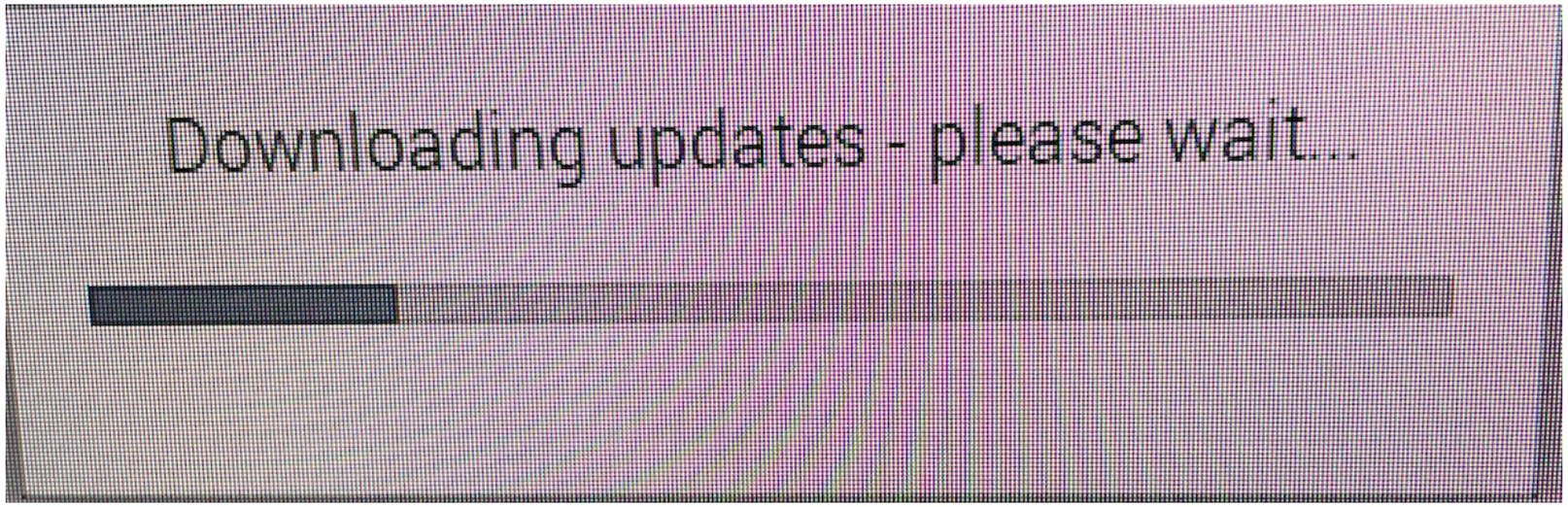
Setup Screen - Select Wifi Network



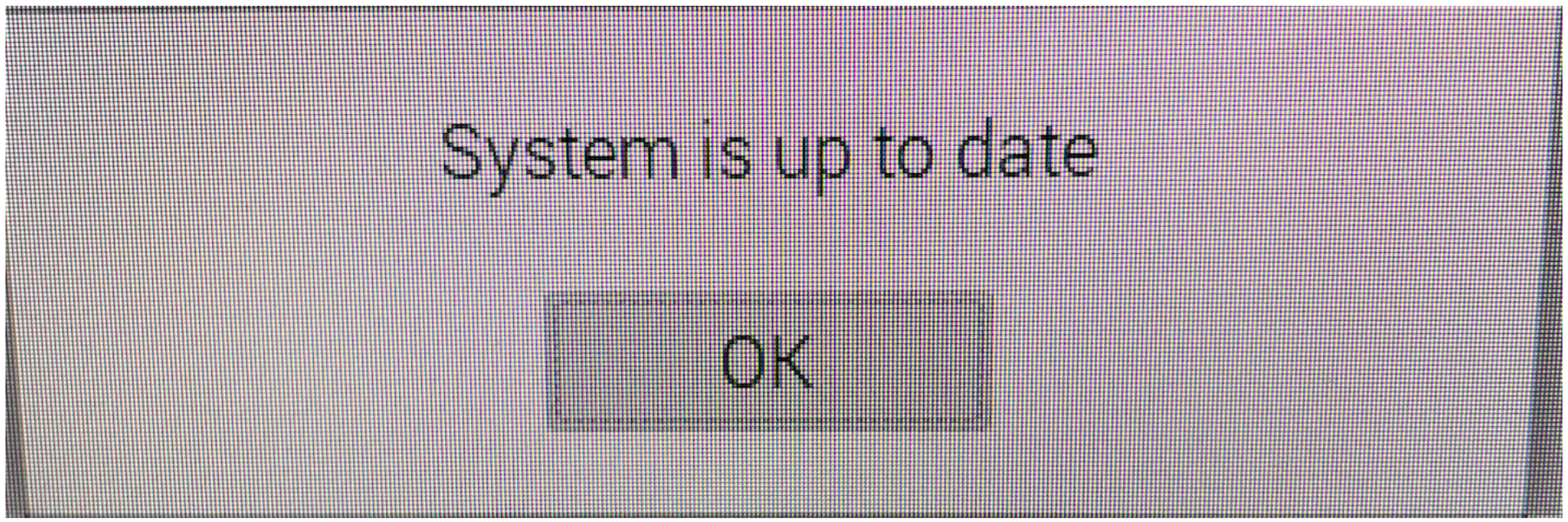
Setup Screen - Update Software



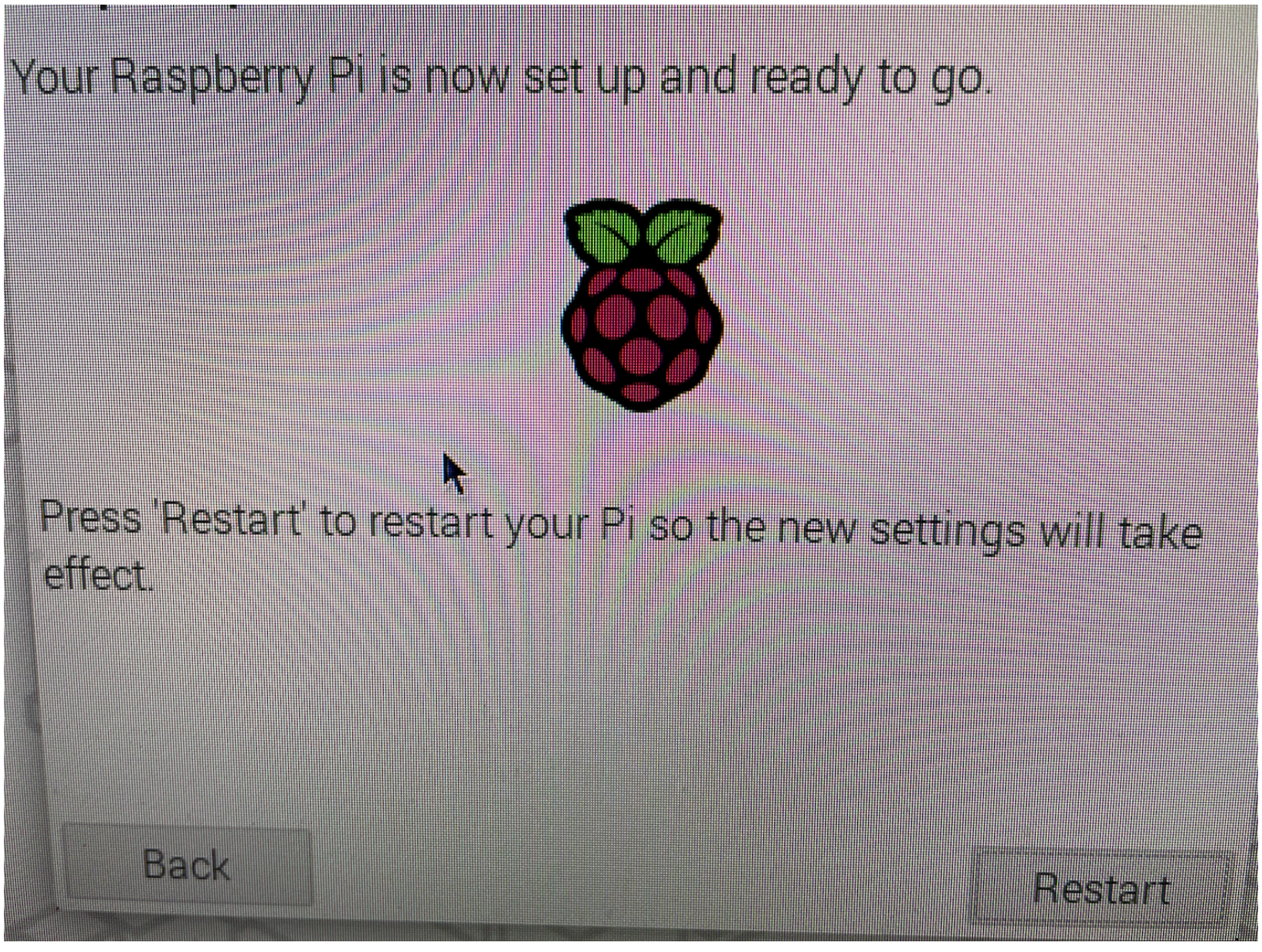
Setup Screen - Downloading Updates



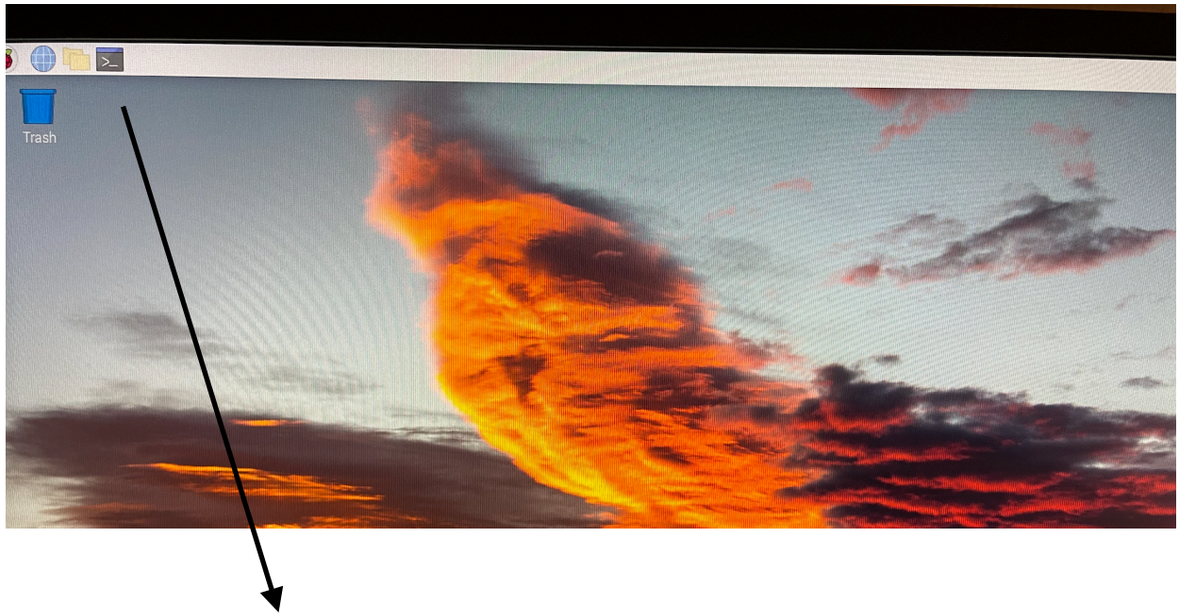
Setup Screen - System Is Up To Date

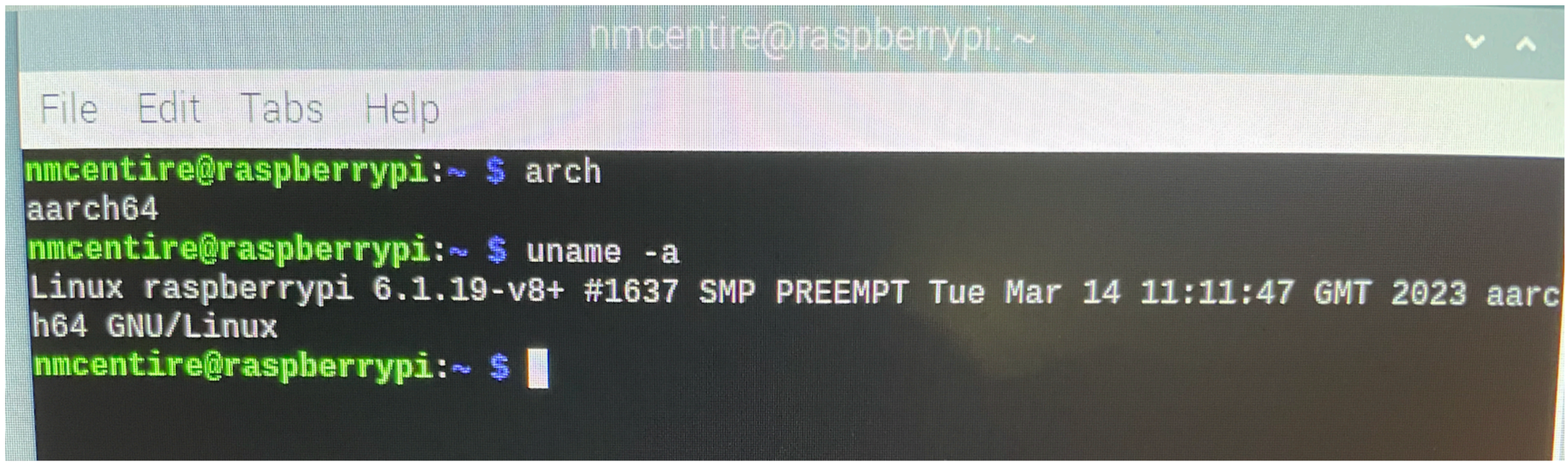


Setup Screen - Ready For Restart

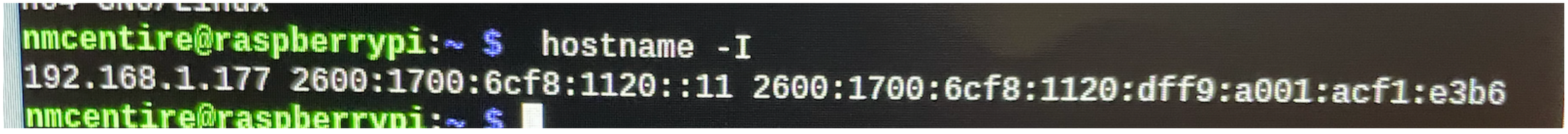


Desktop





Finding your RPI IP address (for remote ssh)



Ping your remote RPi

```
nmcentire@nmcentire-Galago-Pro:~$ ping -c3 192.168.1.177
PING 192.168.1.177 (192.168.1.177) 56(84) bytes of data.
64 bytes from 192.168.1.177: icmp_seq=1 ttl=64 time=100 ms
64 bytes from 192.168.1.177: icmp_seq=2 ttl=64 time=3.02 ms
64 bytes from 192.168.1.177: icmp_seq=3 ttl=64 time=142 ms
```

Try to ssh into RPI

```
nmcentire@nmcentire-Galago-Pro:~$ ssh nmcentire@192.168.1.177
ssh: connect to host 192.168.1.177 port 22: Connection refused
```

Default has sshd (Secure Shell Daemon) not installed

To install sshd on RPI

sudo apt install openssh-server

sudo systemctl enable ssh

sudo systemctl start ssh

Logging onto RPi remotely via ssh

```
nmcentire@nmcentire-Galago-Pro:~$ ssh nmcentire@192.168.1.177
The authenticity of host '192.168.1.177 (192.168.1.177)' can't be established.
ECDSA key fingerprint is SHA256:LkPSMjbzuiR5WImRorOjt6W7N5nFploB+lTOTyIpZ98.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.177' (ECDSA) to the list of known hosts.
nmcentire@192.168.1.177's password:
Linux raspberrypi 6.1.19-v8+ #1637 SMP PREEMPT Tue Mar 14 11:11:47 GMT 2023 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: Sun Mar 26 19:30:39 2023
nmcentire@raspberrypi:~ $
```

Summary

- At this point your RPi is ready to explore how it can be used for Embedded Linux Development

- As a development system

- As a deployment system

Questions?