Week 6 Studio 2 Appendix – Getting Started with your Pi

Core Objectives:

- **C1. Writing the Raspberry Pi OS to the SD Card**
- **C2. Booting up our RPi**
- **C3. Setup Wifi for PI**

Optional Objectives (if time permits):

- **C4. Exploring PI**
- **C5. Setup Remote Desktop for PI**

Preparation (Before the studio):

- Download the latest 32bit "Raspberry Pi OS with desktop and recommended software" image from:
  https://www.raspberrypi.com/software/operating-systems/
  As of writing this version can be found it here:
  https://downloads.raspberrypi.org/raspios_full_armhf/images/raspios_full_armhf-2022-01-28/2022-01-28-raspios-bullseye-armhf-full.zip
  This is a big file (approx. 3.2 GB) so please download it beforehand!
- Download and install the **Raspberry Pi Imager** from:
  https://www.raspberrypi.com/software/
- **Bring your laptop to the studio.**

Studio Setup:

- Each sub group is given one set of components:
  - Raspberry Pi 3 or 3B
  - 16Gb microSD card with SD Card reader
  - Power Bank and Power cable
  - HDMI to HDMI cable or a HDMI to VGA cable
  - Ethernet Cable
  - Monitor (From the workbench)
  - Keyboard (From the workbench)
  - Mouse (From the workbench)

Notes:

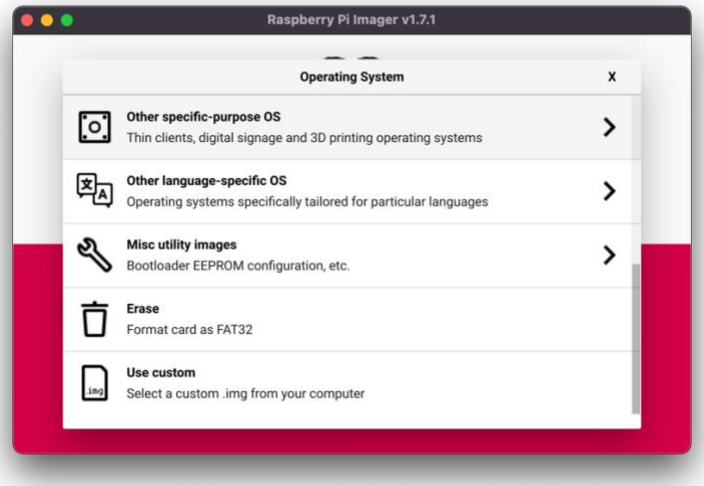
Optional activities are entirely optional ☺, give them a try only when time permits.

C1. Writing the Raspberry Pi OS to the SD Card

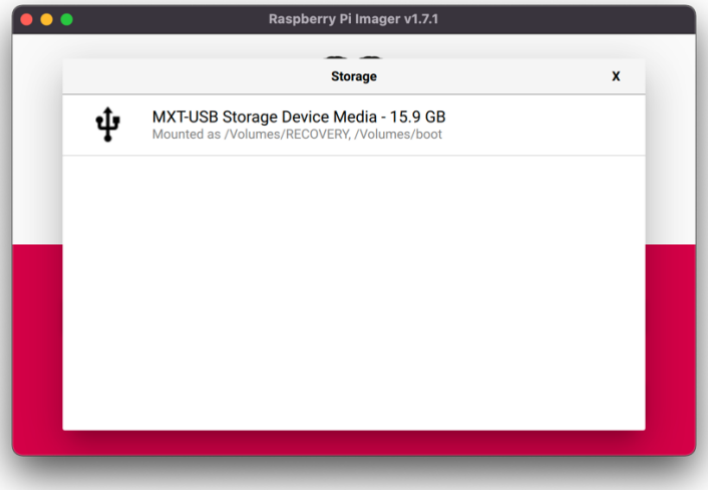
What are we doing?
Raspberry Pi, despite its cute size, is similar to all other PCs in that it requires an operating system (OS). Here in this section we will write the Raspberry Pi OS to the SD Card to prepare the RPi for booting.

1. Slot the SD card into the SD Card reader, then plug the reader into your laptop.
2. Run the "Raspberry Pi Imager" software:

a. Click **Choose OS** and select **Use Custom**. Choose the Raspberry Pi OS image you downloaded before the lab (you definitely did that right?).



b. Click **Choose Storage** and select the SD card (not it may look different on yor machine).



c. Now click **Write** (note that this operation can take 20-30 minutes)

3. Drink coffee / tea while waiting (coffee / tea not provided) ☺

4. Once it has finished writing, you may eject the SD card and close the Raspberry Pi Imager software.

C2. Booting up our RPi

What are we doing?
The SD Card functions as the "hard disk" for the PI. With the OS installed on it, we can now boot our RPi. There is also some configuration to be done on first boot.

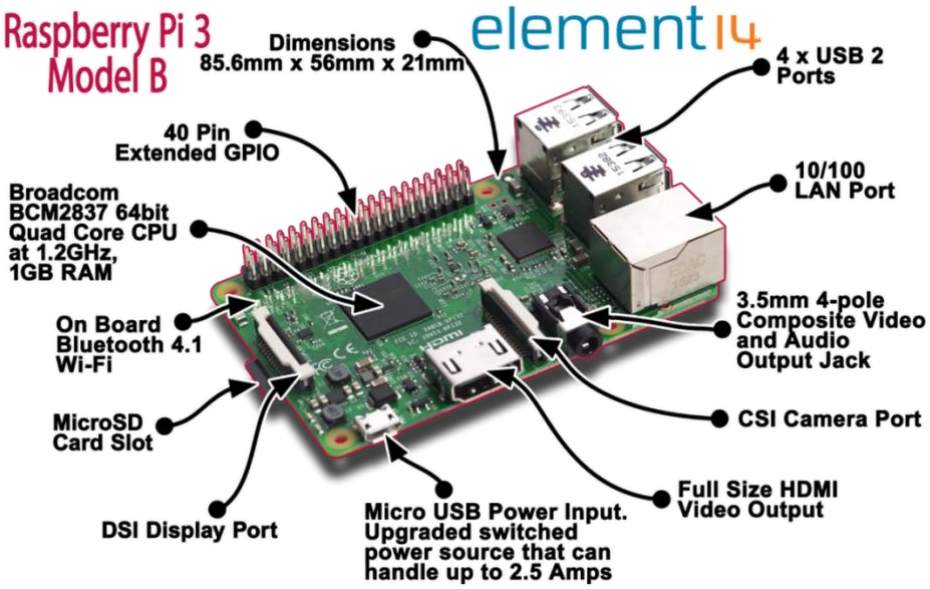


Figure 1. Raspberry PI 3 - Model B (image taken from element14)

Steps:

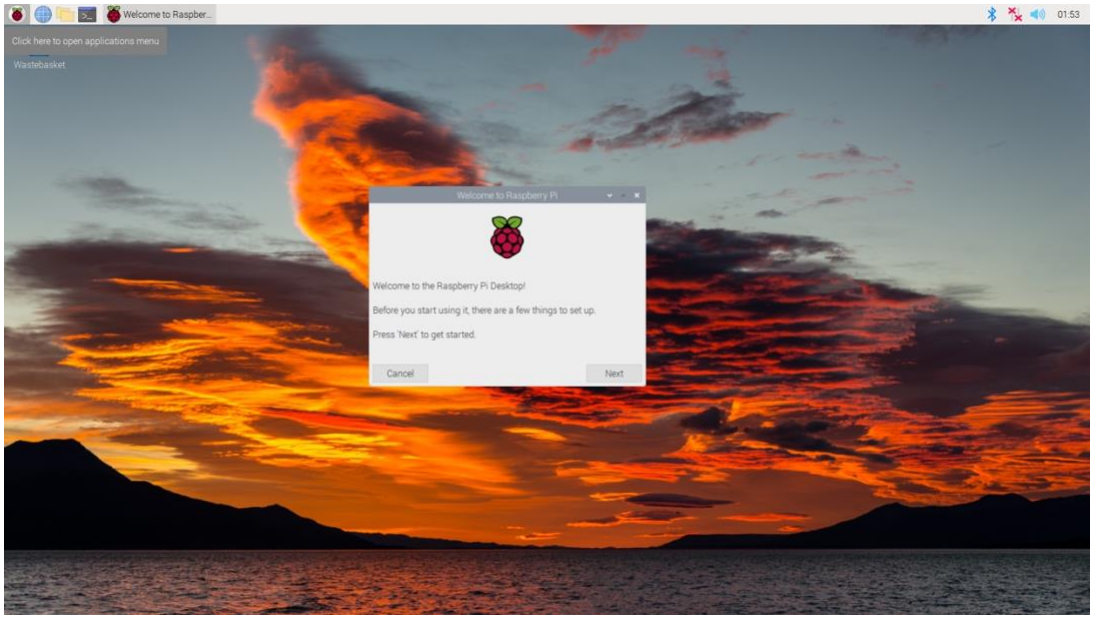
1. Insert microSD card into RPi's microSD Card Slot (see above image for the slot).

2. Connect a HDMI connector to the HDMI output on then connect to the monitor.

3. Connect the USB Keyboard + Mouse.

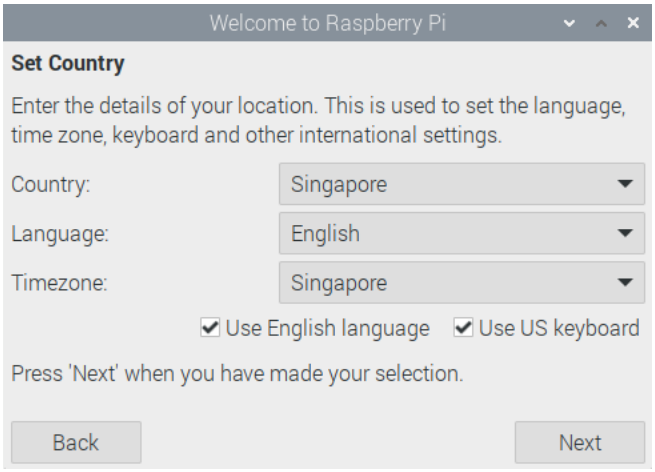
4. Use Micro USB power cable to connect the Pi to a power source, e.g. Power Bank. Note that connecting to laptop / desktop USB port may not work well as Pi 3 requires optimally **5 Volt at 2.5 Ampere**.

5. As soon as power is supplied, you should see LED blinking near the Micro USB power input. The Pi may take about a minute to boot to the Raspberry Pi OS.



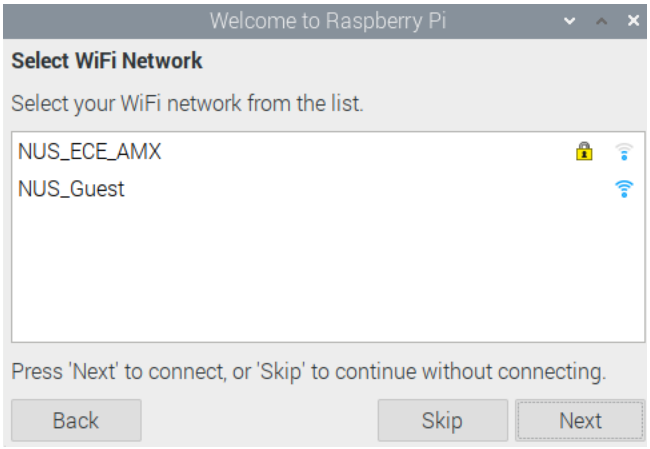
6. You will be greeted with the first time boot configuration screen

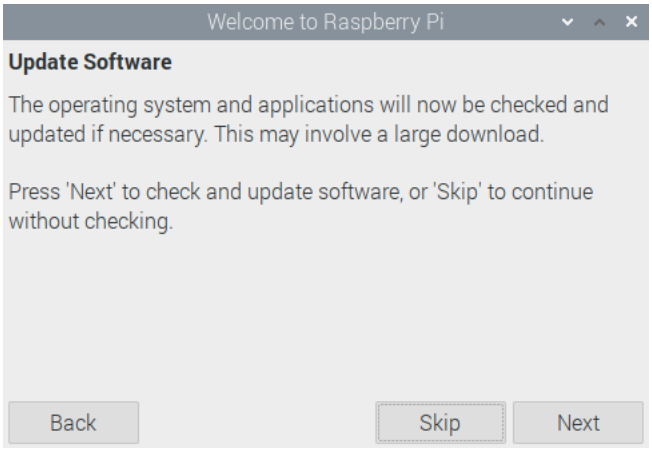
7. Set your **Country** and **Timezone** to **Singapore**, and your **Language** to **English**. Be sure to select **Use US Keyboard**.



8. Change password to something your whole subgroup agrees on (and **can remember**).

9. You can skip the wifi setup and software update.





10. For those using the VGA cable, you may need to change your resolution.
    Open the Applications Menu by clicking the Raspberry Pi icon on the top left of the Task Bar.

    Then select `Preferences > Screen Configuration > Configure`

    A suitable resolution is 1920x1080

C3. Setup Wifi

What are we doing?
Like all computing devices nowadays, Pi works much better with an internet connection. Wifi is the easiest / most common choice to setup.

Wifi networking on Pi can be quite simple for most wireless network setup. From the GUI desktop, clicking on the wifi icon can help you to log onto ***most*** wifi networks effortlessly.

Unfortunately, NUS wifi network uses very tight authentication method and requires additional setup. In the most recent **Raspberry Pi OS** update (September 2022), NUS Wifi ceases to work with Raspberry Pi due to software library configuration bugs. We include the complete steps as an appendix at the end of this document (in case the bugs are fixed in subsequent updates).

For this studio, we will rely on your (or your group mate's) **mobile hotspot** for network connectivity.

1. Ask one of your group mates to setup mobile hotspot.
2. Use the Wifi Icon (top right corner) to locate and connect to the hotspot.
3. With the wifi connection, you can now surf the web / perform software installation / update. **For example**, get/upgrade the ***vim*** software packages for Raspberry Pi OS:
    a. **`sudo apt-get install vim`** (if you are familiar with vim editor)
    b. Download a simple **`.vimrc`** configuration file (e.g. google "`minimalist vimrc c c++`")

Optional Tasks (Attempt Week 6 Studio 2 first)

C4. Exploring PI

Key Idea:
Raspberry Pi is simply a very tiny computer. The major obstacle for most users is that OS used on Pi seems unfamiliar. Raspberry Pi OS, the OS that we have just installed, is a **Debian**-based Linux variant. So, if you are familiar with other Linux/Unix based OSes, you should be quite at home!

Raspberry Pi OS comes with a standard GUI desktop. Feel free to browse around in the menu options. Below are several simple tasks to familiarize you with the environment.

1. Create a **Project** folder on the desktop.
2. Create a simple `Readme.txt` file under the Project directory.
3. Start a terminal and find out whether gcc is installed.
4. Write a "hello world" C program and execute it.

Although it is very convenient to use a GUI environment, you are likely to use only command line input (CLI) interface for your project. Not only CLI is much lighter on the processing power / memory, it is actually equally powerful as the GUI. The only drawback is that you need to know / memorize some of useful commands.

A small set of useful commands are summarized below:

Linux Command	Functionality
man ***XXXX***	Get the manual (help page) on ***XXXX*** command if available. Useful way to learn more about a command.
ls	List the content of the current directory
cd ***YYYY***	Change directory to ***YYYY***
cd	Change directory to home directory
rm ***ZZZZ***	Delete file ***ZZZZ***
sudo ***AAAA***	Execute the command ***AAAA*** as the superuser. Needed for restricted command that make changes to the system. Use with care.
apt-get	Command to install additional packages for Raspberry Pi OS. Need to use together with "sudo", e.g. "sudo apt-get install vim" which install the vim editor on Pi.
sudo reboot now	Reboot Pi immediately.
sudo shutdown now	Shutdown Pi immediately.

There are a couple of text editors installed in Raspberry Pi OS by default:

1. **nano** (simple, command list on the bottom (use ctrl-<key> to activate).
2. **vi/vim** (powerful, hard to learn ☺)

C5. Setup SSH for "headless" PI

Why?
Our Raspberry PI setup is a little cumbersome with all the peripherals attached (monitor, keyboard, mouse). We are going to setup **remote connection** capability so that you can talk to the Pi through network!

Key Idea
As long as you know the IP address of the Pi, you can easily SSH into it from your laptop. The tricky part is that to know the IP address you need to run some commands on the Pi first, i.e. a Chicken-and-Egg Problem.

Caveat
Due to many possible configurations of your laptop / desktop, there is no single set of universal steps. We suggest two most common setups for you to try out. Please don't be overly stressed if they don't work out for you. Your sub group can skip this Core Objectives if all options fail :-)

Option 1. Direct Ethernet Cable with No Additional Configuration

1. Connect the Ethernet cable from your Pi to your laptop **then** **Reboot Pi.**
2. Open a terminal window and key the command "**`ifconfig`**", check the interface "**`eth0`**" (Ethernet adapter 0) for IP address.
3. If you see a valid IP address, something like "**`inet XXX.XXX.XXX.XXX`**", note down the IP address then proceed to section "**SSH from your laptop**".
4. [Exploration] It is worthwhile to setup this option for your own project later as it requires only an Ethernet cable between your laptop and the Pi. One common option is to setup a "Static IP Address" (i.e. fixed IP address for eth0, so that you can always connect).

Option 2. Via Wireless Network (Ethernet Cable not needed)

1. Connect your **Pi and laptop** to either NUS wireless network or mobile hotspot that you control. Just make sure **both Pi and your laptop** are connected to the same network.
2. Check the wireless IP address (mouse over the wifi icon on the top menu bar). Note down the IP address, proceed to section "**SSH from your laptop**".
3. [Exploration] Note that this option works even better if you can discover the IP address of Pi from your mobile device. There are "IP scanner" or similar Apps that can help with this.

<u>SSH from your laptop</u> (If you succeeded using one of the previous methods)

On your **laptop:**

1. Install a SSH Client, e.g. the ***bitvise SSH Client*** (in Luminus files) if needed.
2. Connect to the Pi using the following information:

host: **(the ip address you get for the Raspberry Pi)**
user id: **pi**
password: **<password you chose during Pi Setup>**

Once connected, you can use both a terminal to enter command and the file transfer window (SFTP) to send/receive files to/from the Pi!

You now have the ability to interact with Pi via SSH on a laptop! In future, you can remove the monitor, keyboard and mouse from the Pi and use the laptop as the main way to interact.

References / Resources:

1. Raspberry Pi official website (https://www.raspberrypi.org)

Appendix (Procedure for Connecting to NUS Wifi)

1. Enter "**`sudo vi /etc/wpa_supplicant/wpa_supplicant.conf`**" into terminal. If you are not familiar with the **`vi`** editor, you can use the simpler editor "**nano**".
2. Append the following configuration to the file, pay attention to the identity and password portion. Please use your NUSNET id and password (you have to ask your group mates to politely look away while you type). Also, the parser is quite unforgiving, so please follow **<u>exactly</u>** the following:

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=SG
network={
    ssid="NUS_STU_2-4GHz"
    key_mgmt=WPA-EAP
    eap=PEAP
    identity="nusstu\a0123456"
    password="mypassword"
    phase2="auth=MSCHAPV2"
}
```

Use "Tab" to indent the lines in this block.

3. Once you have finished editing and saving the `.conf` file. **Reboot your Pi.**