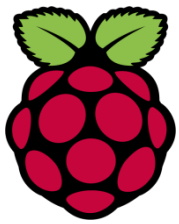
Week 1 Studio 1 – Know your Pi!

Core Objectives:

- **C1. Preparing NOOBS**
- **C2. Setup Raspberry PI – Install Raspbian**
- **C3. Exploring PI**
- **C4. Setup Wifi for PI**
- **C5. Understand the C coding environment on PI**
- **C6. It speaks! Have fun with Text-to-Speech Engine**
- **C7. Setup SSH for "headless" PI**



Optional Objectives (if time permits):

- **O1. Setup Remote Desktop for PI**
- **O2. Setup CVS**

Preparation (Before the studio):

- Download the files in the Luminus Files→Studio Materials→Week 01 - Studio 1.
- Download NOOBS from https://downloads.raspberrypi.org/NOOBS_latest
- Install the SD Formatter Software.
- **Unzip the archive NOOBS_v3_2_1.zip** to a convenient location (e.g. Desktop).
- **Bring your laptop to the studio.**

Studio Setup:

- Each section of the studio group is split into 4 sub-groups (about 2-3 students) for this studio.
- Each sub group is given one set of components:
  - Raspberry Pi 3
  - 16Gb microSD card with SD Card reader
  - Power Bank and Power cable
  - HDMI to DVI cable
  - Ethernet Cable
  - Monitor (From the workbench)
  - Keyboard (From the workbench)
  - Mouse (From the workbench)

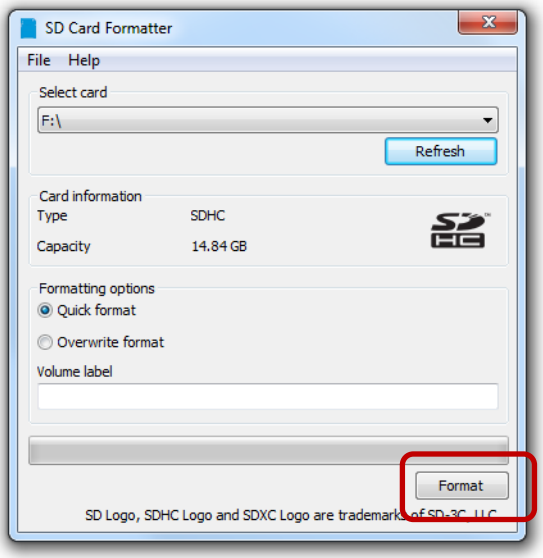
Notes:

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

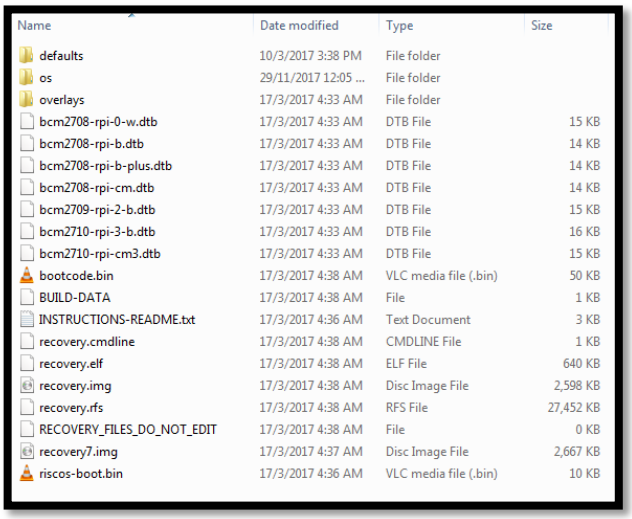
C1. Preparing NOOBS

What are we doing?
Raspberry Pi, despite its cute size, is similar to all other PCs in that it requires an operating system (OS). We will prepare a simple OS installer in this section.

1. Slot the SD card into the SD Card reader, then plug the reader into your laptop.
2. Run the "SD Formatter" software:
    a. Browse to the drive representing the SD Card Reader.
    b. Perform a quick format.



3. Copy the **files and folders** from the unzipped NOOBS archive over to the formatted SD Card. The card content should looks like this afterwards:



Name	Date modified	Type	Size
defaults	10/3/2017 3:38 PM	File folder	
os	29/11/2017 12:05 ...	File folder	
overlays	17/3/2017 4:33 AM	File folder	
bcm2708-rpi-0-w.dtb	17/3/2017 4:33 AM	DTB File	15 KB
bcm2708-rpi-b.dtb	17/3/2017 4:33 AM	DTB File	14 KB
bcm2708-rpi-b-plus.dtb	17/3/2017 4:33 AM	DTB File	14 KB
bcm2708-rpi-cm.dtb	17/3/2017 4:33 AM	DTB File	14 KB
bcm2709-rpi-2-b.dtb	17/3/2017 4:33 AM	DTB File	15 KB
bcm2710-rpi-3-b.dtb	17/3/2017 4:33 AM	DTB File	16 KB
bcm2710-rpi-cm3.dtb	17/3/2017 4:33 AM	DTB File	15 KB
bootcode.bin	17/3/2017 4:38 AM	VLC media file (.bin)	50 KB
BUILD-DATA	17/3/2017 4:38 AM	File	1 KB
INSTRUCTIONS-README.txt	17/3/2017 4:36 AM	Text Document	3 KB
recovery.cmdline	17/3/2017 4:38 AM	CMDLINE File	1 KB
recovery.elf	17/3/2017 4:38 AM	ELF File	640 KB
recovery.img	17/3/2017 4:38 AM	Disc Image File	2,598 KB
recovery.rfs	17/3/2017 4:38 AM	RFS File	27,452 KB
RECOVERY_FILES_DO_NOT_EDIT	17/3/2017 4:38 AM	File	0 KB
recovery7.img	17/3/2017 4:37 AM	Disc Image File	2,667 KB
riscos-boot.bin	17/3/2017 4:36 AM	VLC media file (.bin)	10 KB

More about NOOBS:
The appropriately named NOOBS (New Out Of Box Software) is essentially a OS installer for ~~noob~~… ***first time user*** like you ☺. It provides an easy to use interface and requires only a SD Card with NOOBS files on it. NOOBS also help to recover / reinstall your main OS should something goes drastically wrong (and it probably will…).

C2. Setup Raspberry PI – Install Raspbian

What are we doing?
The SD Card functions as the "hard disk" for the PI. With the NOOBS file on it, we can now install an operating system Raspbian on it.

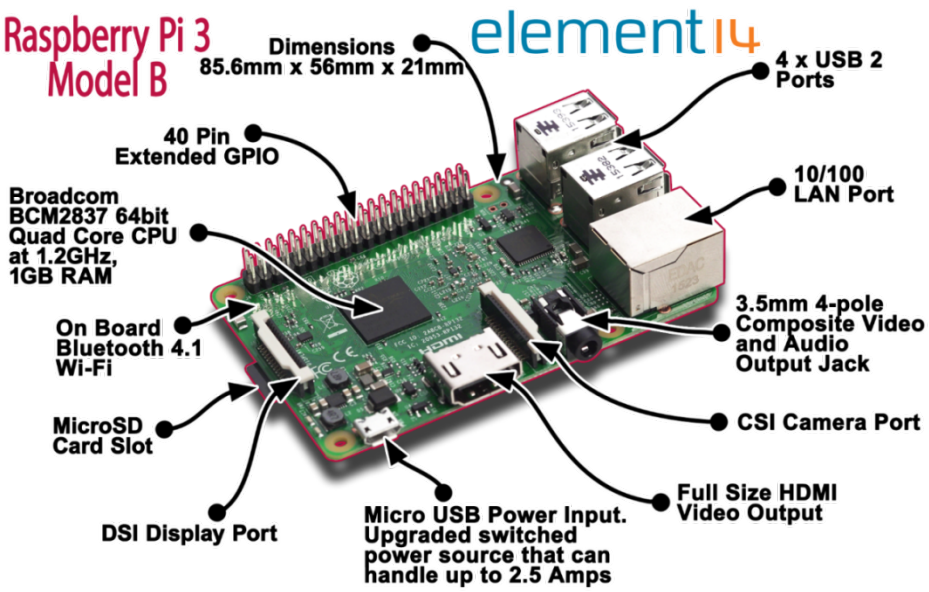


Figure 1. Raspberry PI 3 - Model B (image taken from element14)

Steps:

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

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

3. Connect 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. Pi should boot to the NOOBS installer as shown on next page.

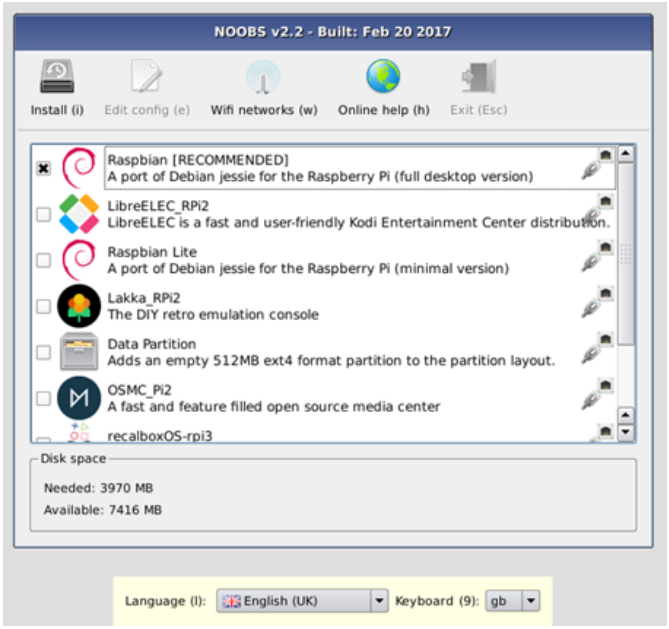


Figure 2. NOOBS (note: image is from an older version)

6. **Select Language and Keyboard layout as US (at the bottom of the screen).**

7. Select "Raspbian Full" (check box) and click "Install (I)" from the menu at the top. Confirm the operation.

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

9. Once the Pi rebooted, you will see the default GUI desktop. You are prompted by a welcome "dialog box" with a few simple configuration:
    a. Country → "Singapore"
    b. Language → "English"
    c. Time Zone → "Singapore"
    d. Ensure the tickbox "Use US keyboard" is ticked.

10. Change password to something your whole subgroup agrees on (and **can remember**).
    a. **SKIP** the wifi setup and software update.

11. Restart Pi if prompted.

12. On the menu bar (top of screen), click on "Menu→Applications →Preferences→Raspberry Pi Configuration"
    a. Go to "Interfaces" Tab, **enable SSH** and **VNC**. Take note of other options, you may need to turn them on in future studios.
    b. Reboot Pi if you are asked to.

13. Show your instructor / TA a working Pi running Rasbpian.

C3. 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. Raspbian, the OS that we have just installed, is a **Debian**-based Linux variant. So, if you are familiar with other Linux/Unix based OSes (say Solaris used on the SoC's Sunfire server), you should be quite at home!

Raspbian come 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. **Show them to your instructor / TA once you got them to work.**

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.
5. [Hidden Quest] Play **minecraft**, craft a stone axe**.**

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 Raspbian. 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 Raspbian by default:

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

C4. 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 network effortlessly.

Unfortunately, NUS wifi network uses very tight authentication method and requires additional setup. In the most recent **Raspbian** update (September 2019), 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 Raspbian:
    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++`")
4. **Show Instructor /TA a youtube video on Pi as an easy proof of connectivity.**

C5. Understand the C coding environment on PI

The terminal should remind you of the gruelling times of using similar environment in CS1010. Let us relive those fun times by writing a simple C code, eh?

Given a user input integer N (N > 0), write a **C program** to display (print on screen) a pyramid of height N (i.e. N lines) with the following layout:

```
Enter N = 4 (sample user input)
   *
  *#*                    //note the pyramid has different "inner layers"
 *###*
*******
```

Optional (Surprise Us!):
Using the same idea (User input only a positive integer), think of the craziest shape you that can generate using a program. Attempt this part only when you have ahem… too much time on your hands.

Show your instructor / TA once you are satisfied with the program.

C6. It speaks! Have fun with Text-to-Speech Engine

Steps:

1. We need to first enable the voice output from the audio jack:
   - Right click on "Volume control" (top left of your screen) and select "Analog"

2. Plug in the headphone (or the portable speaker) into the 3.5mm audio output jack.

3. Enter "**`aplay /usr/share/sounds/alsa/*`**" in a terminal.
   - You should hear a series of test phrases played.

4. Install a simple Text-To-Speech (TTS) engine **eSpeak** by:

   ```
   sudo apt-get install espeak
   ```

5. You can check the installation by entering the following directly in a terminal window:

   ```
   espeak "Hello"
   ```

   If that doesn't work you can try:

   ```
   i.   espeak "Hello" –w hello.wav
   ii.  aplay hello.wav
   ```

[In case you are wondering, espeak normally works on its own. Due to a recent system update, an interface broke and the solution is too painful to duplicate here as this exercise is just for fun ☺. The workaround we use here generate an intermediate wave file for "aplay" to playback.]

6. **Use `espeak` to say something (nice) to the instructor / TA.**
   [**Bonus Challenge**: See whether you can use a female voice instead of the default male voice].

C7. 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.

SSH from your laptop (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.

Show the instructor / TA how you can transfer a file from your connected laptop to the Pi.

Optional Tasks

O1. Setup Remote Desktop

If you are interested to use the GUI desktop while remotely connected to Pi, you can explore:

https://www.raspberrypi.org/documentation/remote-access/vnc/

O2. Setup CVS

For the project, your project group members will likely need to code in parallel. We will learn CVS (Code Versioning System) in a subsequent studio. In the meantime, you can install one of the most popular CVS tool **git** using the following simple steps:

```
sudo apt-get install git
```

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 **exactly** 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.**