Introduction to Raspberry Pi 3 Model B

Updated: 9/18/17

A. Objectives

1. Learn about basics of Pi 3 embedded system
2. Learn how to operate your Pi 3 using different interfaces
3. Learn how to run basic Linux commands on Pi 3
4. Learn how to install different packages on Pi.

B. Time of Completion

This laboratory activity is designed for students with very little knowledge of Raspberry Pi and it is estimated to take about 2.0 hours to complete. Answer all the pre-lab questions and the questions at the end of this lab.

C. Requirements

1. A Raspberry Pi 3 Model 3
2. 32 GByte MicroSD card → Give your MicroSD card to the **lab instructor** for a copy of latest Ubuntu Mate.
3. USB adaptor to power up the Pi

D. Pre-Lab

Pay attention to different components on the board. Refer to the reference section and the figure below and answer the following questions:

1- What type of processor does RPi 3 have?
2- How much RAM does the Pi have?
3- Does RPi have a WiFi interface? What about Bluetooth?
4- Where do you connect the power to?
5- Does the RPi 3 have a MicroSD card?

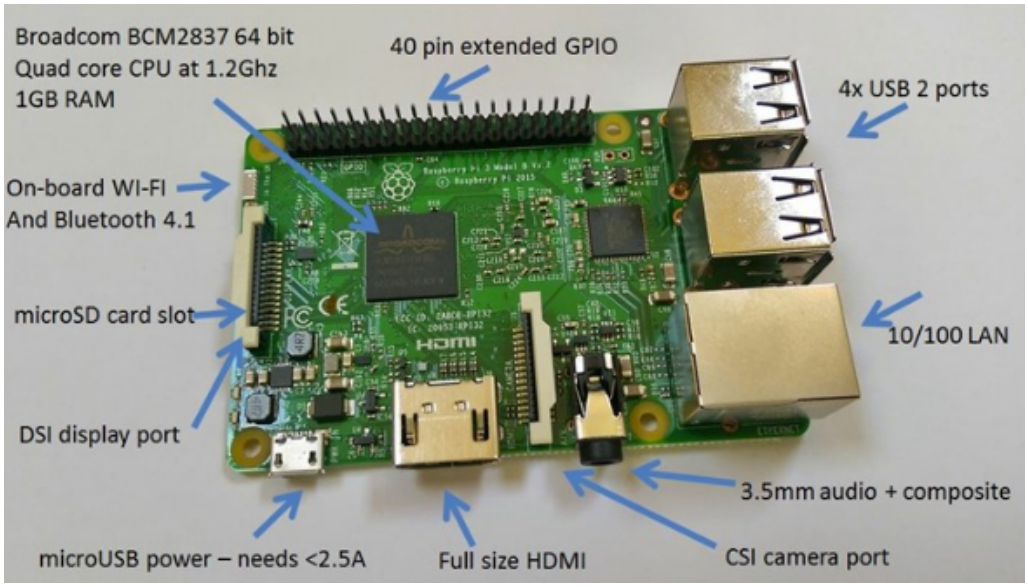


Figure 1 – Diagram of Pi 3 Model B

The table below compares various features of different Pis [1]. What is the key difference between Pi 2 and Pi 3, if any?

As shown in the table below, built specifically for the new Pi 3, the Broadcom BCM2837 system-on-chip (SoC) includes four high-performance ARM Cortex-A53 processing cores running at 1.2GHz with 32kB Level 1 and 512kB Level 2 cache memory, a VideoCore IV graphics processor, and is linked to a 1GB LPDDR2 memory module on the rear of the board. To lean more about ARM Cortex-A53 please see http://www.arm.com/products/processors/cortex-a/cortex-a53-processor.php.

It must be noted that there's no need to connect an external antenna to the Raspberry Pi 3. In order to keep the size of the device to a minimum, the radios are connected to a chip antenna (as opposed to PCB antenna).

The operating system (OS) recommended by Raspberry Pi on the first two generations is their own Linux type, called **Raspbian OS**. Raspberry Pi 3 brings new capabilities. Note that Pi 3 can also run Windows 10 IoT core and Ubuntu. In our application we like to run Linux **Ubuntu Mate**.

	Raspberry Pi	Raspberry Pi 2	Raspberry Pi 3
Released	February 2012	February 2015	February 2016
CPU	ARM1176JZF-S	ARM Cortex-A7	ARM Cortex-A53
CPU speed	700MHz single core	900MHz quad core	1,200MHz quad core
RAM	512MB 256MB Rev 1	1GB	1GB
GPU	Broadcom Videocore IV	Broadcom Videocore IV	Broadcom Videocore IV
Storage	SDHC slot MicroSDHC Model A+ and B+	MicroSDHC slot	MicroSDHC slot
USB Ports	2 on Model B	4	4
WiFi	No built-in wifi	No built-in wifi	802.11n and Bluetooth 4.1

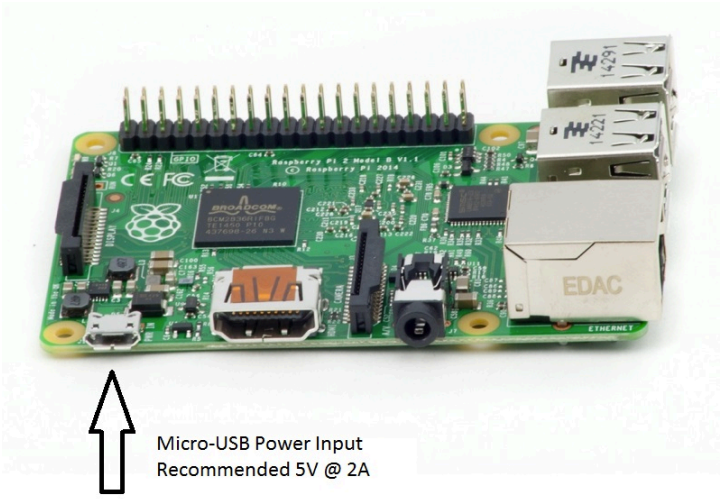
[1] See http://socialcompare.com/en/comparison/raspberrypi-models-comparison for more information.

Ubuntu MATE is a free and open-source Linux distribution and an official derivative of Ubuntu. Its main differentiation from Ubuntu is that it uses the MATE desktop environment as its default user interface, based on GNOME 2, which was used for Ubuntu versions prior to 11.04, instead of the Unity graphical shell that is the default user interface for the Ubuntu desktop. Ubuntu MATE supports PowerPC and ARMv7 & A53 (on the Raspberry Pi 2 and 3, respectively).[2]

E. Access to you Raspberry PI

Follow the steps below:

Place your 32 Gbyte MicroSD card that you received from your **instructor** in the appropriate slot on the Pi. Power up the Pi using the Micro-USB adaptor. At this point you should see the RED LED on. Connect the Pi to an active LAN line, using the 10/100 Ethernet LAN connector. Note that at this point the GREEN LED on the LAN connector should be on.



We now look at different methods we can communicate with the Pi.

Method 1) It is always possible to access your Pi using a keyboard and a monitor. This can be done by connecting your monitor into the HDMI port and using the USB port for your keyboard (See Figure 1). This approach is very easy and convenient. However, it may be expensive and not so portable!

Method 2) Another approach to access your Pi using **SSH remote login**. Find out the IP address of your computer (use `ifconfig`). Let's assume it is `192.168.1.73`. Assuming your Pi is connected to the same LAN, the IP address of the Pi is going to be within the same subnet (in this case 192.168.1.xx). Run the following commands:

```
$> ping 192.168.1.73
$> arp -a
```

Alternatively you can try

```
$> nmap -sP 192.168.1.1/24
```

At this point you should see something like the following:

```
$> ssuee-desktop (192.168.1.75) at b8:27:eb:1d:14:d1 on en1 ifscope
[Ethernet]
```

Note that *ssuee* is the username for your Pi. In this case the IP address of the Pi happened to be `192.168.1.75`. From a terminal `ssh` into your Pi: `ssh ssuee@192.168.1.75`. Type *Yes*, and use *student* for password. NOTE: If you don't have `ssh` capability, download PuTTY. Once you are in your Pi, type `ls`. Run a few other Linux commands to ensure everything is ok. Read about SSH protocol here: https://en.wikipedia.org/wiki/Secure_Shell .

[2] For more information see: https://en.wikipedia.org/wiki/Derivative_work

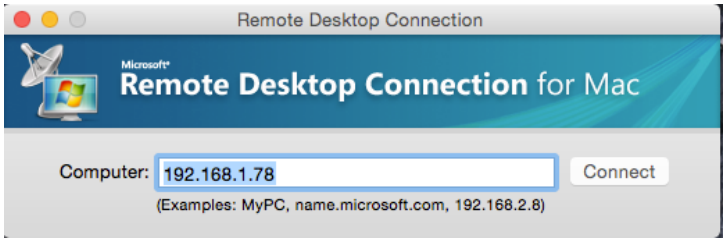
Follow the steps below (note that some of the packages may have already been installed):

- Update your Pi's OS: `sudo apt-get update`
- Install `xrdp` package: `sudo apt-get install xrdp`
- Install tree utility: `sudo apt-get install tree`
- Install play utility: `sudo apt-get install sox`

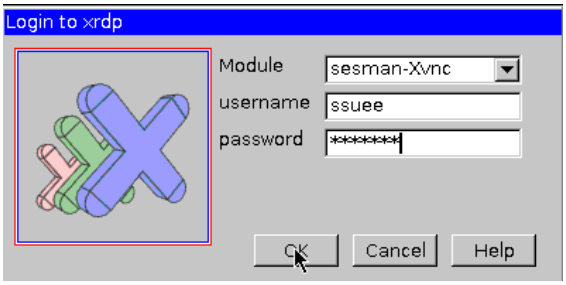
APT is a package manager that is used by Ubuntu. It allows us to install, delete, or update a software package. There are many other package managers. For example, to download packages for Python we usually use `pip`. Other Linux distros may use different package managers, such as `yum` in Fedora. Most OSX systems use `brew` to install software packages.

Method 3) The third approach to access your Pi is using **remote desktop**.

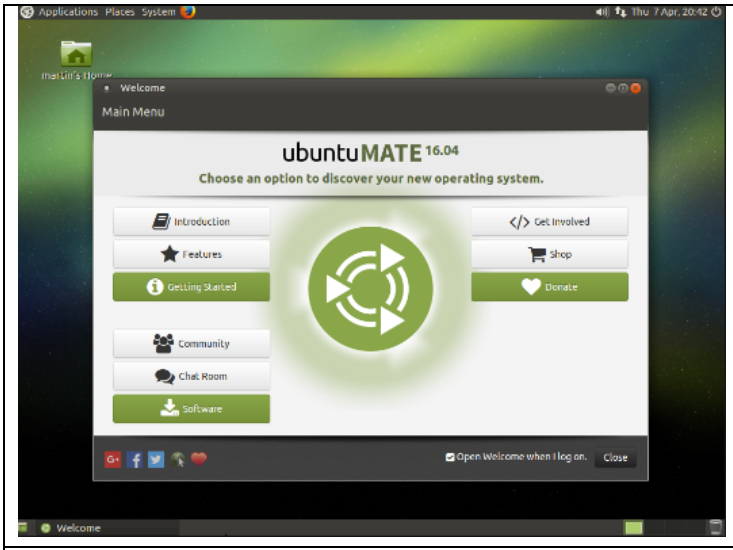
Download Remote Desktop Connection on your computer. You may already have the program if you are using Windows or MAC.



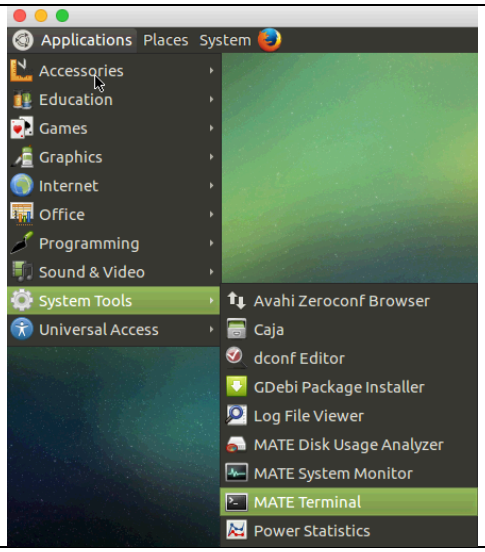
Enter your username and password as shown here (username= `ssuee` and password= `student`). Click "Ok" and you will be logged into the Ubuntu GUI.



You will get the following screen. Note that you can open a terminal from the desktop, as shown below.

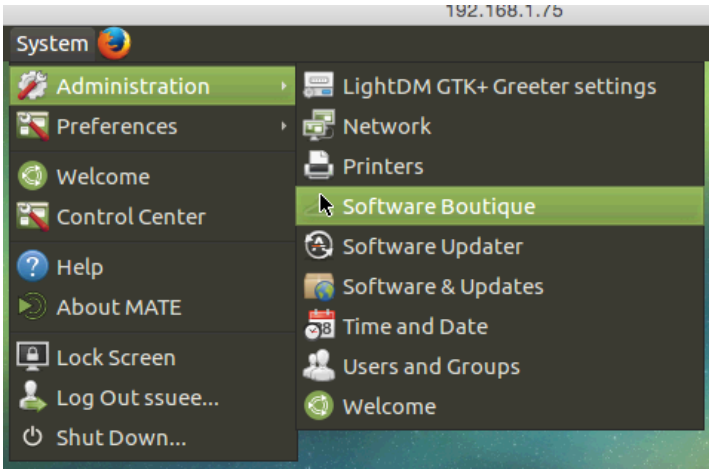


GUI desktop



Opening a terminal

Download Wireshark in your Pi by clicking on *Software Boutique*. Download Wireshark, as shown below:

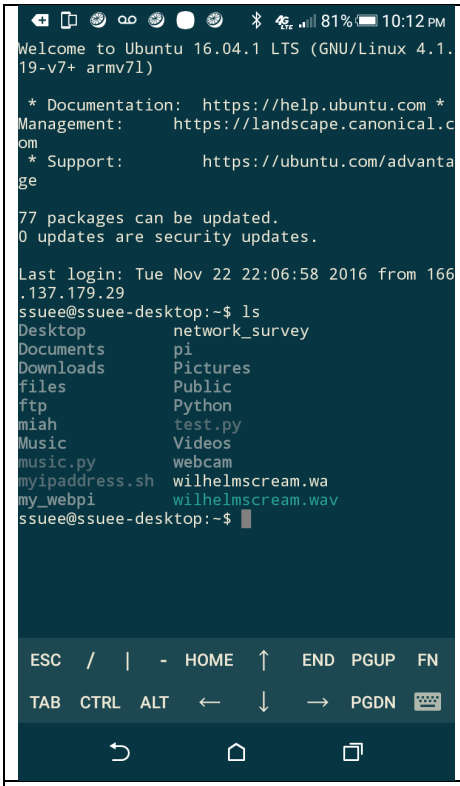


Advanced Activity: Using Weaved you can access your Pi from outside your LAN: https://www.weaved.com/raspberry-pi-remote-connection/

You can try Weaved on your own!

Method 4) Accessing RPi From **Smart Phone:**

Using various apps for Smart Phone, it is possible to control the RPi. Below, we show a snapshot of an App called *JuiceSSH*[3] that can be configured to access the RPi via SSH. Note that we can run NANO editor on the application. Obviously, if *port forwarding* is enabled, the RPi can be accessed from anywhere.



JuiceSSH

Using nano editor in JuiceSSH.

[3] For more information see here: https://play.google.com/store/apps/details?id=com.sonelli.juicessh&hl=en

Answer the following questions:

1- Run the following command: `whoami`. What is the response?
2- How many NIC connections the Pi has?
3- Record how much space is available on your MicroSD card.
4- What is the name of the active NIC card?
5- What does the following command do: `nmap -sP 192.168.1.1/24`
6- What would the following commands do: `uname -a` and `uname -mrs`?
7- Try `tree -d`. What does it do?
8- Does do you create a tree for `/etc` directory content?
9- Refer to Appendix A and check which directories are available. What is the purpose of `/etc` directory?
10- What happens if you type `ls -la | less`?
11- Does the Pi support `vi`? Try doing `vi test.txt`. What happens?
12- Use command `date` and made sure the time and dates are correct. If not use `-- set` option to correct them.
13- Looking at Appendix B, what is the advantage of Kali Linux? You may want to do some searching!
14- What does apt in `apt-get` stand for?
15- What are the key differences between Pi2 and Pi3, if any?
16- What does `arp -a` accomplish?
17- If you are using SSH, how can you install a package such as *wireshark*?
18- What is the difference between `pip` and `brew` package managers?
19- Other than Ubuntu MATE, name FIVE Linux distros that can be used on RPi.
20- What will be the advantage of enabling *port forwarding* on a RPi?
21- Try `nano test.txt` – what do you see? What is the difference between `nano` and `vi`?
22- Measure the power consumption of a Pi in the following conditions (HINT: you need to use a digital multi meter, remember P=VI):
    a. When there is no WiFi connection.
    b. When there is WiFi connection.

H. References

[1] Very good historical introduction to computing systems and Raspberry Pi: http://www.slideshare.net/nipunmaster/a-seminar-report-on-raspberry-pi

[2] Information about hardware architecture of the Pi: https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2836/README.md

[3] Good comparison between Pi 3 and other versions: https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/

I. Credits

Special thanks to many online resources and all SSU students who assisted putting together this lab.

A ppendix A

Some of the most important directories in the Raspbian filesystem. You Ubuntu Mate may have many of these directories. Try which ones are available. Read the description of each one.

Directory	Description
/	
/bin	Programs and commands that all users can run
/boot	All the files needed at boot time
/dev	Special files that represent the devices on your system
/etc	Configuration files
/etc/init.d	Scripts to start up services
/etc/X11	X11 configuration files
/home	User home directories
/home/pi	Home directory for pi user
/lib	Kernel modules/drivers
/media	Mount points for removable media
/proc	A virtual directory with information about running processes and the OS
/sbin	Programs for system maintenance
/sys	A special directory on the Raspberry Pi that represents the hardware devices
/tmp	Space for programs to create temporary files
/usr	Programs and data usable by all users
/usr/bin	Most of the programs in the operating system reside here
/usr/games	Yes, games
/usr/lib	Libraries to support common programs
/usr/local	Software that may be specific to this machine goes here
/usr/sbin	More system administration programs
/usr/share	Things that are shared between applications like icons or fonts
/usr/src	Linux is open source; here's the source!
/var	System logs and spool files
/var/backups	Backup copies of all the most vital system files
/var/cache	Any program that caches data (like apt-get or a web browser) stores it here.
/var/log	All of the system logs and individual service logs
/var/mail	All user email is stored here, if you're set up to handle email
/var/spool	Data waiting to be processed (e.g. incoming email, print jobs)

Appendix B

Available OS systems for Pi:

- Angstrom Linux
- Arch Linux ARM Images are no longer maintained for Raspberry.
- Chromium OS
- Debian ARM
- Fedora ARM Pi 2/3 only.
- Fedberry , a Fedora Remix for Pi 2 and 3
- FreeBSD
- Gentoo
- IPFire
- Kali Linux
- LibreELEC
- Meego MER + XBMC
- Nard SDK (Embedded systems)
- NetBSD
- OpenELEC + XBMC
- OpenWrt
- OSMC , Open source media center
- PiBang
- Pidora , a Fedora Remix for Pi 1 [No longer maintained]
- PwnPi , a Raspbian clone for penetration testing.
- QtonPi
- Plan 9
- Raspbian , a Debian derivative
- Raspbmc is now OSMC
- RaspBSD
- RetroPie
- Risc OS
- Slackware ARM
- SliTaz
- Ubuntu Mate
- Void Linux
- Windows 10 IoT Core