



Introduction to the Raspberry Pi

Jon Macey

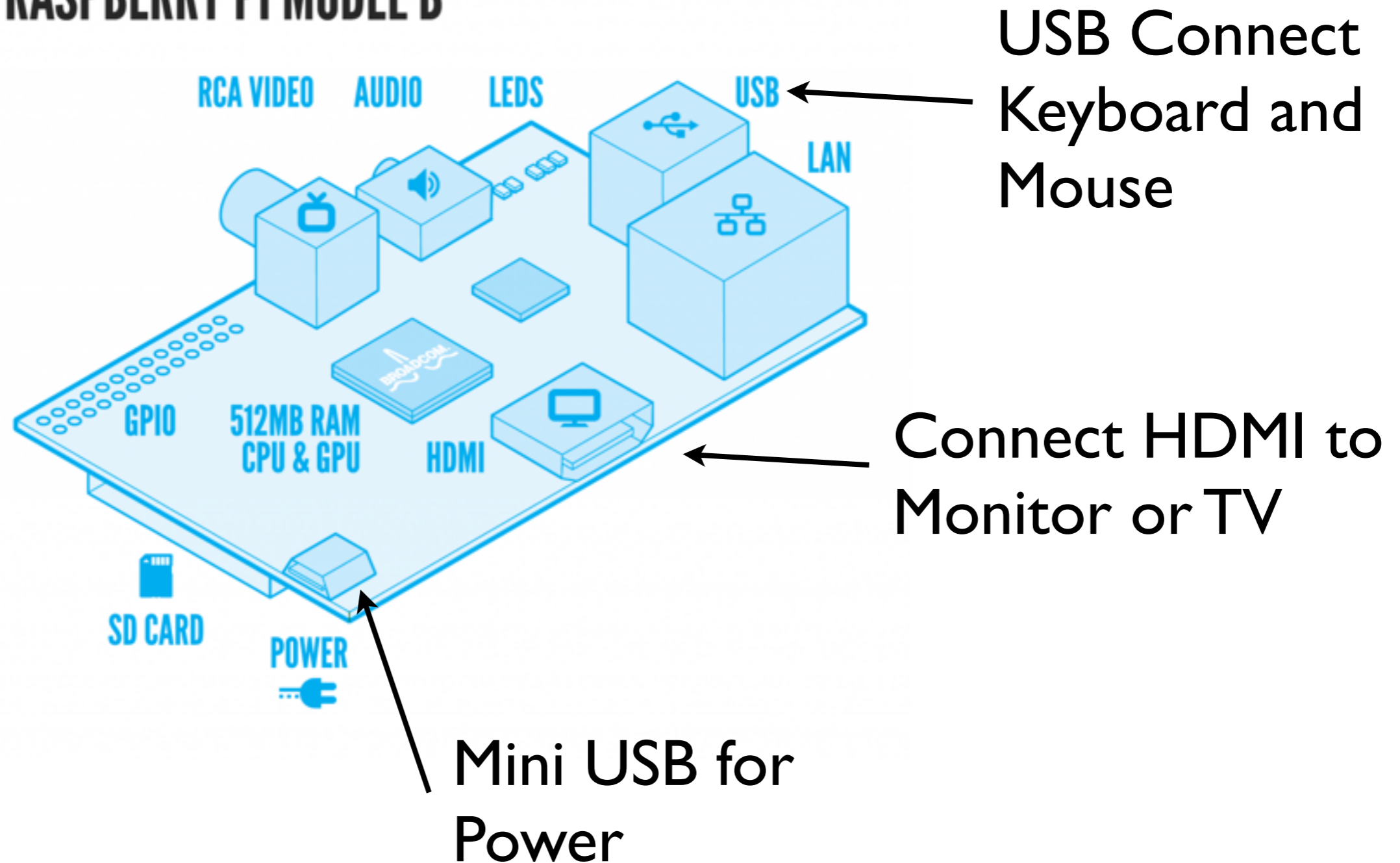
jmacey@bournemouth.ac.uk

Course Outline

- Connecting the pi
- Introduction to Linux
- Updating an upgrading
- Different Operating systems
- Introduction to programming
- Project ideas

Connecting things

RASPBERRY PI MODEL B



Connection Order

- Ensure that all things are connected before powering up
- Sometimes if HDMI or Keyboard and mouse inserted afterwards they don't get recognized
- If in doubt pull the power and re-start, you are unlikely to break anything!

Raspbian “wheezy”

- This is the default operating system installed on the supplied SD cards
- It is perhaps the easiest and most well supported
- There are other different operating systems each have their own advantages
- There are also dedicated images for things like media players (raspbmc)

raspi-config

- A program that allows us to “tune” the pi
- can set memory split / overclock
- enable / disable features
- update / upgrade the OS
- must be run as the “root” user as modifies the operating system

Introduction to Linux

- Linux is an operating system (like windows or Mac OSX) and based on Unix
- Whilst you may not think you have used it, you will be surprised where it turns up
 - Most web servers run unix (approx 60%)
 - Many embedded devices (DVR, Entertainment systems, mobile devices)
 - Mac OSX is a type of Unix (called BSD)
- And of course the raspberry pi

- Written in C - the system is portable, modifiable and relatively easy to understand.
- Systems shell - provides a simple and effective user interface.
- Hierarchical file structure - consistent, efficient and easily maintained.
- Multi-user, multi-tasking - many users and many processes per user and per machine.
- Machine architecture hidden - allowing for portability of programs between

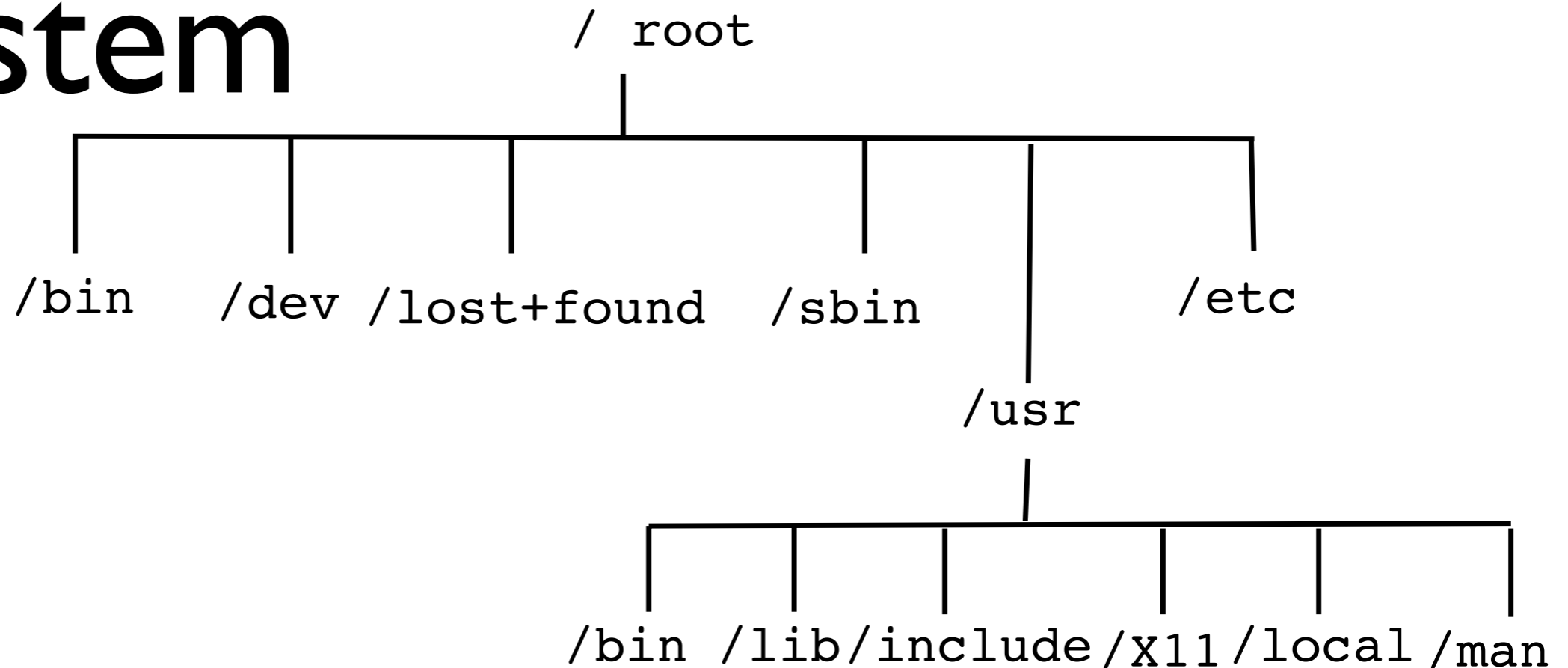
Users

- Linux systems have many different users
- Each user has a home directory
- There are many special users pre-configured on the pi
 - user pi (password raspberry) is the default user on the wheezy installation
 - root the administrator / super user account
- We can add as many users as we like

root

- The root user is the owner of the system, root can
 - create new users
 - install / remove software
 - access all file systems
 - destroy the system very easily!!
- root is accessed via the pi user using the command sudo (super user do)

Filesystem



- Everything is a file, no drive letters (C:\) just mount points
- Notice the / not \ like in windows.

Home Directories

- Every user has a home directory
- On most systems (including the pi) it is
 - /home/[User Name]
- So for the “pi” user it will be
 - /home/pi
- We can change access to this directory so only the current user can see it (more on this later)



A trip back in time




- Early electronic computing (pre 80's) didn't have the GUIs we have today.
- This meant that all interactions with the computer were done with typing into a terminal.
- Most modern operating systems still have the option to do this
- And in some cases this method is quicker than using the GUI (but does require some additional knowledge)



The Shell

- By default the raspberry pi has 6 virtual terminals created at boot
- These are known as `tty1` - `tty6`
- We can access each of these using `alt + F1` - `F6`
- Each one is a new login and when we login we get the users default shell
- The default shell on the pi is the Bourne Again Shell (or `bash`)

Bash

- The bash shell is where we type unix commands
- It is also a fully programmable scripting environment.
- It features auto-completion for file names / paths
- Once learnt is a very powerful interface to the computer

- **Tab**  : **Autocompletes** from the cursor position.
- **Ctrl** + **a** : Moves the cursor to the line start (equivalent to the key **Home** ).
- **Ctrl** + **b** : Moves the cursor back one character (equivalent to the key ).
- **Ctrl** + **c** : Sends the signal **SIGINT** to the current task, which aborts and closes it.

- **Ctrl** + **e** : **(end)** moves the cursor to the line end (equivalent to the key **End** ).
- **Ctrl** + **f** : Moves the cursor forward one character (equivalent to the key ).
- **Ctrl** + **g** : Abort the research and restore the original line.
- **Ctrl** + **h** : Deletes the previous character (same as backspace).

Some Unix Commands

- ls (lists directory contents)
- pwd (print working directory)
- mkdir (make directory)
- cd (change directory)
- rmdir (remove directory)
- rm (remove)
- touch (create empty file or re-set file flag)
- chmod (change mode)
- man (manual page)

Adding a user

- To add a new user we need to be root,
- To become the root user we can use the following command
 - `sudo su`
- `su` means substitute user and by default it will attempt to become the root user
- You will notice the prompt now says `root@raspberrypi`

Adding a user

sudo su

useradd -m -d /home/jmacey

passwd jmacey

- `useradd` adds a user (must be root)
- `-d` indicates we wish to create a home directory (in this case `/home/jmacey`) `-m` creates it
- `jmacey` is the name of the user to create
- `passwd` is the command to change a password (or set one)
- any user can change their own password, root can change all passwords (re-set)

removing a user

- To remove a user we use the userdel command
- It is followed by the username so
 - userdel jmacey
- Will remove the previous user however it will not delete the home directory
- To do this use the -r flag

Using Packages

- Most debian based linux distributions use the apt package manager (Advanced Packaging Tool)
- In the shell we can use the following commands (as root)
 - apt-cache search (package)
 - apt-get install (package)
- To update / upgrade the os use
 - apt-get update
 - apt-get upgrade

gui based packages

- The synaptic package manager can make things easier
- We can install this using apt-
 - `sudo apt-get install synaptic`
- We need to run as root once installed
 - `sudo synaptic`

Different Raspi OS

- As SD Cards are cheap we can have many different OS's and change when we need to
- Just need to image each card
 - Pidora (a version of Fedora Linux)
 - RISC OS
 - RaspBMC (a full media player)

Programming

- The default raspbian image contains the following programming languages
 - C / C++ (demo)
 - Python (python 2.x and python 3) as well as pygame (demo)
 - scratch (demo)

Extension Boards

- There are many extension boards / accessories for the pi
 - Camera Module (demo)
 - piface (demo)
 - gertboard (demo)

References

- <http://www.raspberrypi.org/quick-start-guide>
- [https://en.wikipedia.org/wiki/Bash_\(Unix_shell\)](https://en.wikipedia.org/wiki/Bash_(Unix_shell))
- <http://trevorappleton.blogspot.co.uk/2013/03/keeping-your-raspberry-pi-updated-with.html>
- <http://www.raspbmc.com/>