RASPBERRY PI BASICS
System on Chip (SoC)

• What is System on Chip?
  – A complex IC that integrates the major functional elements into a single chip or chipset.
    • programmable processor
    • on-chip memory
    • accelerating function hardware (e.g. GPU)
    • both hardware and software
    • analog components

• Benefits of SoC
  – Reduce overall system cost
  – Increase performance
  – Lower power consumption
  – Reduce size
SoC in Raspberry Pi: Broadcom BCM2835 SoC Multimedia processor

- **CPU**
  - ARM 1176JZF-S (armv6k) 700MHz
  - RISC Architecture and low power draw
  - Not compatible with traditional PC software

- **GPU**
  - Broadcom Video IV
  - Specialized graphical instruction sets

- **RAM**
  - 512MB (Model B rev.2)
  - 256 MB (Model A, Model B rev.1)
SoC in Raspberry Pi: Broadcom BCM2835 SoC

BCM2835 SoC (right) and Samsung K4P2G324ED Mobile DRAM (left)
Connecting a Display and Audio

- **HDMI**
  - Digital signal
  - Video and audio signal
  - DVI cannot carry audio signal
  - Up to 1920x1200 resolution

- **Composite RCA**
  - Analog signal
  - 480i, 576i resolution

- **3.5mm jack**
RPi Remote Connections

http://pihw.wordpress.com/guides/direct-network-connection/
Universal Serial Bus

- Two USB 2.0 ports in RPi
- Buy a powered USB hub

Passive models are cheaper and smaller, but lack the ability to run current-hungry devices like CD drives and external hard drives.
Storage: Secure Digital (SD)

• Form factor
  – SD, Mini SD, Micro SD

• Types of Card
  – SDSC (SD): 1MB to 2GB
  – SDHC: 4GB to 32 GB
  – SDXD up to 2TB

The card should be at least 2GB in capacity to store all the required files
Storage: Continue

SD Formatter:

https://www.sdcard.org/downloads/formatter_4/

How to mount USB flash drive from command line:

http://linuxcommando.blogspot.co.uk/2007/12/how-to-mount-usb-flash-drive-from.html
Networking

Ethernet (IEEE 802.3)

USB Ethernet Converter

Wi-Fi Adapter
Networking - wireless

- IEEE 802.11 Wi-Fi
  - Protocols
    - 802.11 b, up to 11Mbps
    - 802.11 g, up to 54Mbps
    - 802.11 n, up to 300Mbps
    - 802.11 ac (draft), up to 1Gbps
  - Frequency band
    - 2.4GHz, 5GHz
Low Speed Peripherals

• General Purpose Input/Output (GPIO)
  – Pins can be configured to be input/output
  – Reading from various environmental sensors
    • Ex: IR, video, temperature, 3-axis orientation, acceleration
  – Writing output to dc motors, LEDs for status.
Power Consumption

• microUSB power connector
  – 2.5W (model A)
  – 3.5W (model B)

• Powered USB hub
  – To provide more power for USB peripherals
Useful links

• Raspberry Pi official website

• Raspberry Pi wiki
  – http://elinux.org/RaspberryPiBoard

• Raspberry Pi verified peripherals
  – http://elinux.org/RPi_VerifiedPeripherals

• The MagPi
  – http://www.themagpi.com

• Raspberry Pi on Adafruit Learning System:
  – http://learn.adafruit.com/category/learn-raspberry-pi
Raspberry Pi Setup

1. Download the Raspberry Pi operating system
   - Linux releases compatible with the Pi:
     - The recommended OS is Raspbian:
       http://downloads.raspberrypi.org/raspbian_latest

2. Unzip the file that you just downloaded
   - Right click on the file and choose “Extract all”.
   - Follow the instructions—you will end up with a file ending in .img
3. Download the Win32DiskImager software
   - a) Download win32diskimager-binary.zip (currently version 0.6) from:
     https://launchpad.net/win32-image-writer/+download
   - b) Unzip it in the same way you did the Raspbian .zip file
   - c) You now have a new folder called win32diskimager-binary
• 4. Writing Raspbian to the SD card
  – a) Plug your SD card into your PC
  – b) In the folder you made in step 3(b), run the file named Win32DiskImager.exe
  – c) If the SD card (Device) you are using isn’t found automatically then click on the drop down box and select it
  – d) In the Image File box, choose the Raspbian .img file that you downloaded
  – e) Click Write
  – f) After a few minutes you will have an SD card that you can use in your Raspberry Pi
5. Booting your Raspberry Pi for the first time
   - On first boot you will come to the Raspi-config window
   - Change settings such as timezone and locale if you want
   - Finally, select the second choice: `expand_rootfs` and say ‘yes’ to a reboot
   - The Raspberry Pi will reboot and you will see raspberrypi login:
     - Username: pi, password: raspberry
   - Start the desktop by typing: `startx`
   - The desktop environment is known as the Lightweight X11 Desktop Environment (LXDE)
<table>
<thead>
<tr>
<th>info</th>
<th>Information about this tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>expand_rootfs</td>
<td>Expand root partition to fill SD card</td>
</tr>
<tr>
<td>overscan</td>
<td>Change overscan</td>
</tr>
<tr>
<td>configure_keyboard</td>
<td>Set keyboard layout</td>
</tr>
<tr>
<td>change_pass</td>
<td>Change password for 'pi' user</td>
</tr>
<tr>
<td>change_locale</td>
<td>Set locale</td>
</tr>
<tr>
<td>change_timezone</td>
<td>Set timezone</td>
</tr>
<tr>
<td>memory_split</td>
<td>Change memory split</td>
</tr>
<tr>
<td>ssh</td>
<td>Enable or disable ssh server</td>
</tr>
<tr>
<td>boot_behaviour</td>
<td>Start desktop on boot?</td>
</tr>
<tr>
<td>update</td>
<td>Try to upgrade raspi-config</td>
</tr>
</tbody>
</table>
Re-mapping Keyboard:

• sudo vi /etc/default/keyboard
  XKBLAYOUT="gb"
  Change “gb” to “us”

• (This assumes you want a us mapping, if not replace the gb with the two letter code for your country)
Install and Start SSH

- Update apt-get package index files:
  - sudo apt-get update
- Install SSH:
  - sudo apt-get install ssh
- Start SSH server:
  - sudo /etc/init.d/ssh start
- To start the SSH server every time the Pi boots up:
  - sudo update-rc.d ssh defaults
• SSH client for Windows:
  – PuTTY

• SSH Secure File Transfer
Install Java

1. JDK 8 (with JavaFX) for ARM Early Access
   http://jdk8.java.net/fxarmpreview/
   - Download from Raspberry pi
   - Download from your own PC and copy it (scp) to Raspberry pi

2. Extract the JDK tar.gz file
   - tar –zxvf fileToExtract.tar.gz
   - You will get a folder “jdk1.8.0”
Set Java PATH

• If you put the folder “jdk1.8.0” in the home directory (i.e. /home/pi), you will see the java executables (e.g. javac, java, appletviewer) in the directory: /home/pi/jdk1.8.0/bin

• open /etc/profile
  add:
    PATH=$PATH:/home/pi/jdk1.8.0/bin
  export PATH

• Reboot:
  sudo reboot
Linux System Administration
Kernel and Distribution
Kernel and Distribution

Although only the kernel itself is rightly called Linux, the term is often used to refer to a collection of different open-source projects from a variety of companies. These collections come together to form different flavors of Linux, known as distributions.
File System Logical Layout

**boot:** This contains Linux kernel and other packages needed to start the Pi

**bin:** OS-related binary files, like those required to run the GUI, are stored here.

**dev:** Virtual directory, which doesn’t actually exist on the SD card. All devices connected to the system can be accessed from here.

**etc:** This stores miscellaneous configuration files, including the list of users and their encrypted passwords

**home:** Each user gets a subdirectory beneath this directory to store all their personal files

**lib:** This is a storage space for libraries, which are shared bits of code required by different applications.

**lost+found:** A special directory where file fragments are stored if the system crashes.

**media:** A special directory for removable storage devices, like USB memory sticks or external CD drives.
File System Logical Layout

**mnt:** This folder is used to manually mount storage devices, such as external hard drives.

**opt:** This stores optional software that is not part of the OS itself. If you install new software to your Pi, it will usually go here.

**proc:** Another virtual directory, containing information about running programs which are known in Linux as processes.

**selinux:** Files related to Security Enhanced Linux, a suite of security utilities originally developed by the US National Security Agency.

**sbin:** Stores special binary files, primarily used by the root account for system maintenance.

**sys:** This directory is where special OS files are stored.

**tmp:** Temporary files are stored here automatically.

**usr:** This directory provides storage for user accessible programs.

**var:** This is virtual directory that programs use to store changing values or variables.
Software

LXTerminal and Root Terminal: use the Linux command line in a window without leaving the GUI.

Midori & NetSurf: Lightweight web browser

IDLE and IDLE 3: IDE for Python 2.7 and 3

Task Manager: Checks the available memory, processor workload, closes crashed or unresponsive programs

Music player at the console: moc

OpenOffice.org: sudo apt-get install openoffice.org

Image Editing: Gimp

LAMP (Linux, Apache, MySQL and PHP) stack
Sudo apt-get install apache2 php5 php5-mysql mysql-server
Installing, Uninstalling and Updating Software

• Package manager in Debian: apt
  • GUI for apt, Synaptic Package Manager doesn’t work well on Pi due to the lack of memory

• Make sure that the apt cache is up to date:
  • apt-get update

• Finding software:
  • apt-cache search emacs

• Installing software and dependencies:
  • sudo apt-get install emacs

• Uninstalling software:
  • sudo apt-get remove emacs
  • sudo apt-get purge emacs (removes everything including configurations)

• Upgrading software:
  • Sudo apt-get upgrade
  • Sudo apt-get install emacs
Troubleshooting

Keyboard and Mouse Diagnostics

Power Diagnostics

Display Diagnostics

Network Diagnostics

Emergency Kernel
Wired Networking Configuration

```bash
sudo nano /etc/network/interfaces

iface eth0 inet static
 [tab] address 192.168.0.10
 [tab] netmask 255.255.255.0
 [tab] gateway 192.168.0.254

sudo /etc/init.d/networking restart

sudo nano /etc/resolv.conf

nameserver 8.8.8.8
nameserver 8.8.4.4

sudo /etc/init.d/networking restart

ping –c 1 www.raspberrypi.org
```
Wireless Networking Configuration

• USB Wi-Fi adapters are very power-hungry. Connect a powered USB hub to the Pi, and then insert the Wi-Fi adapter into that.

• Print out the entire kernel ring buffer and find out the company that makes the actual chip: mesg | grep ^usb

```
Atmel-firmware
Firmware-atheros
Firmware-brcm80211
Firmeware-intelwimax
Firmware-ipw2x00
Firmware-iwlwifi
Firmware-ralink
Firmware-realteck
Zd1211-firmware
```

• Check the current status of the network: iwconfig
Configurating the Raspberry Pi

RPi doesn’t have a BIOS menu. It relies on text files containing configuration strings that are loaded by the chip when powers on.

- Hardware settings: config.txt
- Memory Partitioning: start.elf
- Software Settings: cmdline.txt
References for Python

Beginner’s Guide to Python
http://wiki.python.org/moin/BeginnersGuide

A free, interactive tutorial
http://www.learnpython.org

Learn Python the Hard Way (Shavian Publishing, 2012)

Dive Into Python 3 (APRESS, 2009)