

# Rather Useful Seminars

## A Slice of Raspberry Pi

## Roadmap

- Introduction to the Raspberry Pi device
- What can you use a Raspberry Pi for?
- Talking to the Hardware
- A Raspberry Pi Arcade table
- Q & A

# Raspberry Pi Introduction

# What is the Raspberry Pi?

- The Raspberry Pi is a small computer system built onto a tiny circuit board
- It has around the same computing power as a modern Smartphone
- It runs a version of the Linux operating system (Debian)

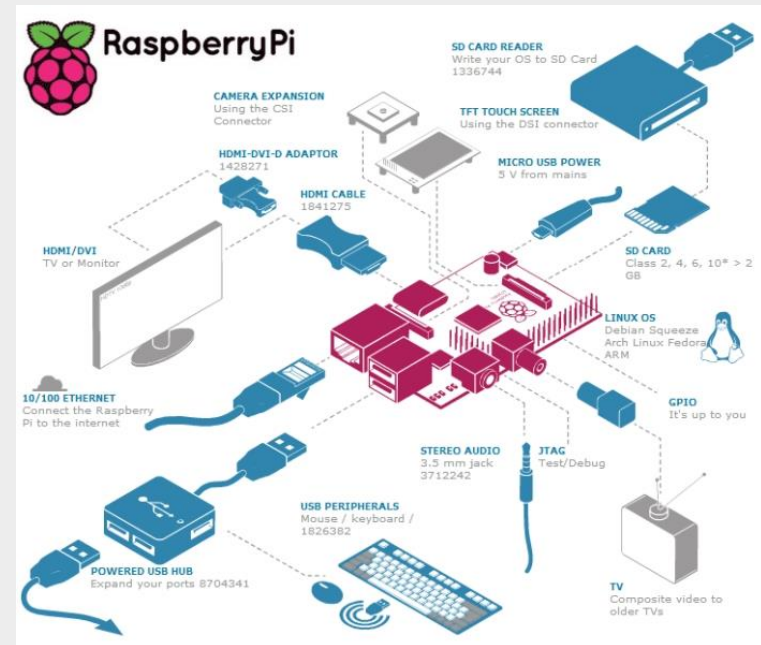


## What do I need to get started?

- You can buy a Pi for a very low price, but remember that you will need at least a case, keyboard, mouse, screen and power supply as well
- You can use the ones from your PC if you like
  - The keyboard and the mouse must be USB
  - The monitor must be HDMI (you can use composite but the quality is not very good)
  - You will need a USB power supply and a micro-usb cable
- The Raspberry Pi can be a bit picky about some devices
  - Best to seek out ones sold to work with the Pi

## What can I connect a Raspberry Pi to?

- There are lots of connection options:
  - USB peripherals
  - HDMI monitor
  - Wired network
  - Stereo Audio
- You can even use input/output pins to link to electrical components



## Data Storage on Raspberry Pi

- The Raspberry Pi has 256 or 512 Mbytes of RAM which is used for programs, data and graphics as it runs
- Instead of a hard disk it uses an SD card
- The SD card holds the operating system and data files used by programs
- You can buy an SD card pre-loaded with the operating system or you can use a PC to load an operating system onto an empty one



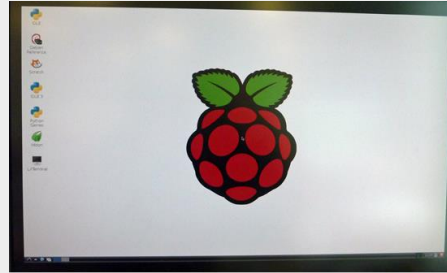
## Running the Raspberry Pi

- When you turn the Raspberry Pi on it will go through a boot sequence
- This is just the same as any modern computer
- Once the system has booted you will log on to it
- Then you can start the X-  
Windows desktop for a windowed interface

A screenshot of a terminal window showing the boot process of a Raspberry Pi. The terminal text includes: "INIT: version 2.88 booting", "info] Using makefile-style concurrent boot in runlevel S.", "...] Starting the hotplug events dispatcher: udevd[114]: starting versio", "175", "ok", "...] Synthesizing the initial hotplug events...udev: version magic '3.1.9+", "preempt mod\_unload modversions ARMv6 ' should be '3.1.9+ mod\_unload ARMv6 '", "udev: version magic '3.1.9+ preempt mod\_unload modversions ARMv6 ' should be", "3.1.9+ mod\_unload ARMv6 '", "ata\_id[213]: HDIO\_GET\_IDENTITY failed for '/dev/sr0': Invalid argument", "done.", "ok | Waiting for /dev to be fully populated...done.", "Starting fake huclock: loading system time.", "Fri Nov 23 13:17:01 UTC 2012", "ok | Setting preliminary keymap...done.", "ok | Setting parameters of disc: (none).", "ok | Activating swap...done.", "EXT4-fs (sda2): re-mounted. Opts: (null)", "...] Loading kernel modules...snd\_page\_alloc: version magic '3.1.9+ preempt", "mod\_unload modversions ARMv6 ' should be '3.1.9+ mod\_unload ARMv6 '", "done.", "ok | Activating lvm and md swap...done.", "...] Checking file systems...fsck from util-linux 2.20.1".



## Windows on Raspberry Pi



- The windowed interface on Raspberry Pi is quite familiar
- You can surf the web and there are also programs that are a bit like Microsoft Office that you can use
- But you must remember that you are using the same kind of processing power as you find in a Smartphone

## Things worth knowing..

- You can swap the SD card from one Raspberry Pi to another with no problems
- If you completely break the operating system on an SD card it is easy to copy a new version onto it
- The Raspberry Pi can update itself (and load lots of useful extra programs) via its network connection
- You can buy a WiFi adapter for a Raspberry Pi for around ten pounds
- You can emulate a Pi on a PC using the qemu emulator

What is a Raspberry Pi good for?

## The Raspberry Pi as a Computer

- The Raspberry Pi makes a fine computer
- It is not the fastest or the most powerful machine but it is perfectly capable of doing “proper” computing
- You can learn to program the device, use it to surf the internet, host web pages and generally use it anywhere you would use a Linux computer, because that is what it is
- You can have full administrative rights on the machine and if you make a mistake you only have to copy the software back onto your SD card

## The Raspberry Pi as a Media Centre

- You can obtain versions of the operating system on SD card that make a Raspberry Pi into a dedicated media centre
- This turns the Pi into a means of accessing and viewing music and video stored on networked hard drives
- The low price of the Pi means that you can afford to dedicate one to this task

## Embedded Device

- The Raspberry Pi can serve as an embedded device that is directly connected to external hardware
- It can then be used to remotely monitor or control the device via its network connection
- The low price of the device itself makes it very attractive in this role
- You can even use it to make your own “arcade cabinet” for playing retro arcade games

## Programming on the Pi

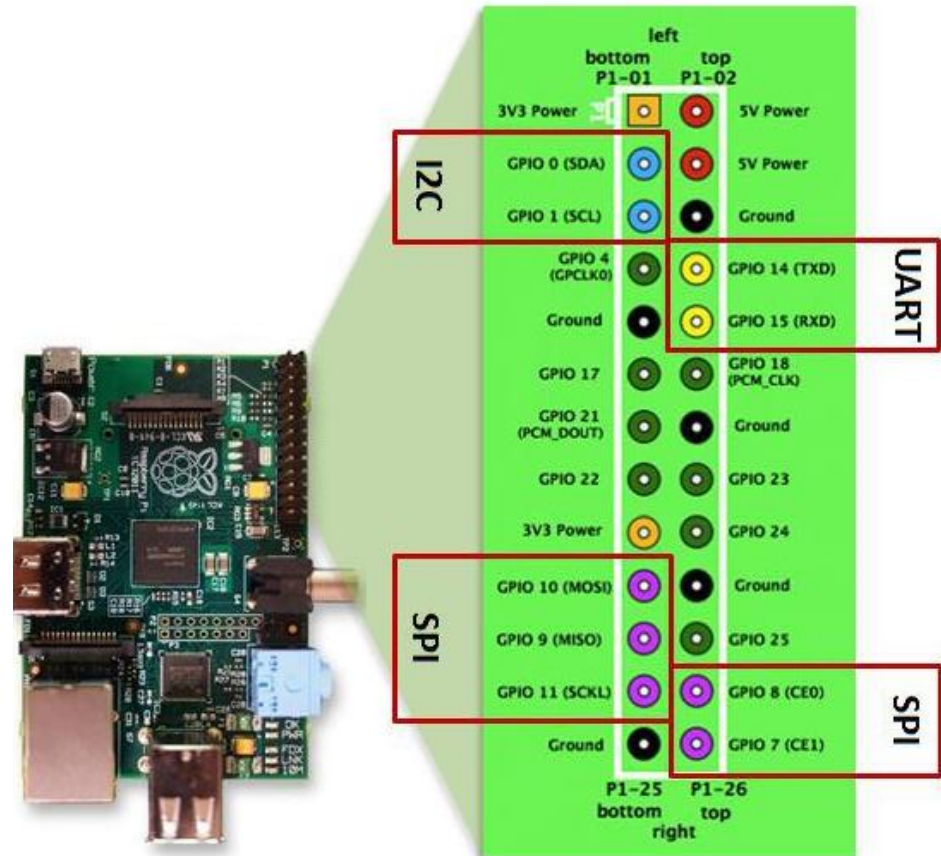
- Scratch
  - Teaches programming concepts in a non-threatening environment
- Python is a good first language
  - High level
  - Emphasis on readability
  - Interactive
- PyGame
  - A set of tools to help make games
- Minecraft on the Raspberry Pi
  - But with a Python API
- General Purpose Input Output
  - Breaking out of the Raspberry Pi

# Talking to the Hardware



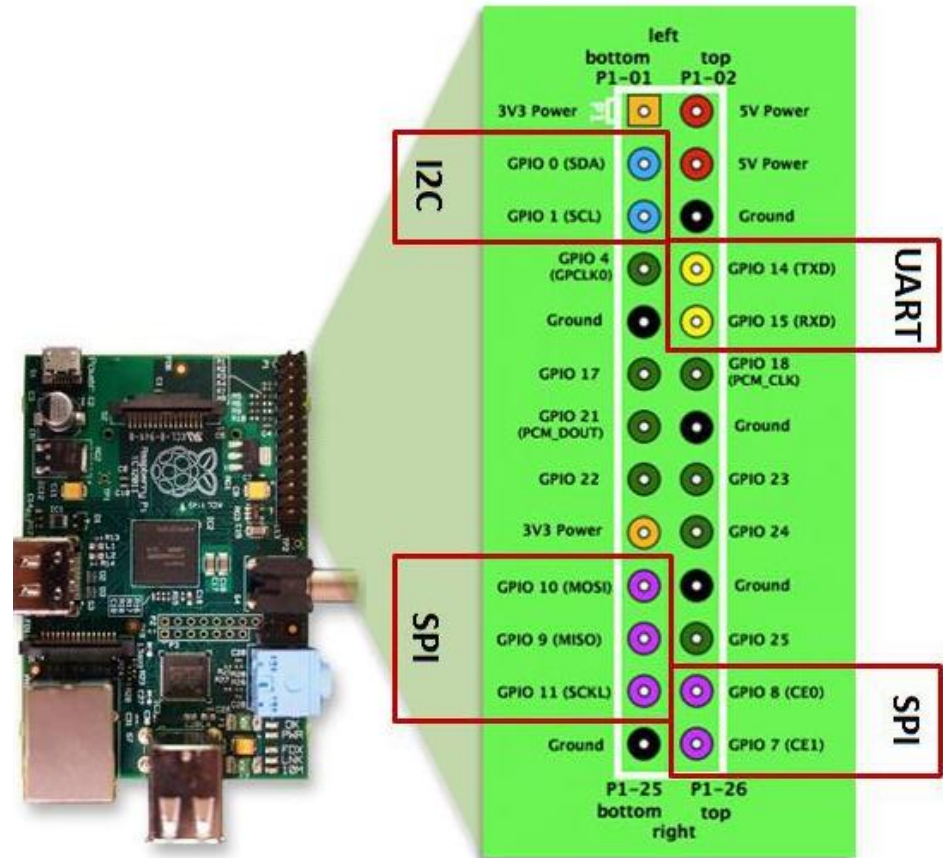
# Raspberry Pi Hardware Interfacing

- The Raspberry Pi provides a set of pins that can be used for direct hardware interfacing
- You can connect these to lights and switches, or to more complex digital devices



# Raspberry Pi Hardware Interfacing

- I2C – a serial connection
- SPI – a different kind of serial connection
- UART – the original serial connection
- In addition there are clock and Pulse Width Modulation outputs



# Mild Health Warning

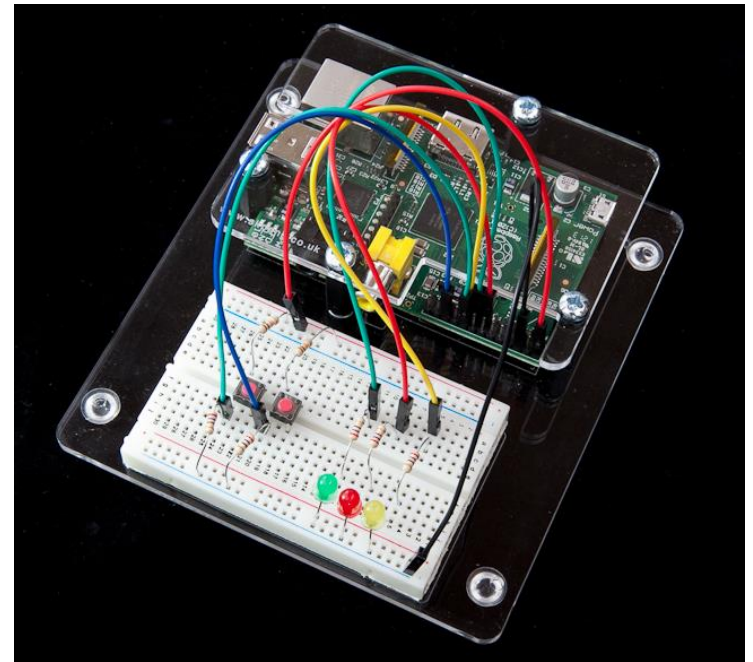
- Bad connections can cause damage to the device
- This is particularly true if you send 5Volt signals into input ports
  - The Raspberry Pi runs on 3.3 volt levels and the higher voltage signals will destroy the input/output connections and possibly the entire device.. which cannot be repaired
- The good news is that the voltages cannot actually hurt you

# Inputs and Outputs

- There is enough power in the output from the Pi to drive an LED
- If you want to switch large amounts of power you will have to add an amplifier of some kind
- The Pi can also sense digital inputs
- If you want to read from analogue devices you will need to add extra hardware

# Interfacing Kit

- You can get an interfacing kit that is provided with some jumper cables, a breadboard and some switches and LEDs
- This can be the basis of some interesting experiments



<http://www.skpang.co.uk/catalog/starter-kita-for-raspberry-pi-pi-not-include-p-1070.html>

# Hardware Interfacing with Python

```
sudo Python
```

- A normal user is not allowed to interface directly with the hardware
- However the Supervisor user does have permission to do this
- This means that you must start the Python run time environment as a super user
- The command `sudo` is used for this

# importing the libraries

```
import RPi.GPIO as GPIO
import time
```

- There are some libraries that make it very easy to connect to hardware from a Python program
  - The program must import these
- We can also import the time library, which we can use to control our flashing light

# Setting the pin numbering scheme

```
GPIO.setmode(GPIO.BOARD)
```

- There are two ways that a program can refer to pins on the Raspberry Pi device
  - Numbering based on the numbers assigned by the Broadcom chipset
  - Numbering based on the numbers on the pins on the board
- We have to tell the GPIO library which scheme we are using
  - The board numbering is much easier to use



# Creating an output pin

```
GPIO.setup(11,GPIO.OUT)
```

- This sets up pin 11 on the connector as an output pin
- Once we have done this we can set the output to be high or low in the software
- We can also set pins up as inputs

# Writing to the output

```
GPIO.output(11,True)
```

- This enables the output from the pin
- It will cause 3.3 volts to appear on the output connection
- We can also write False to turn the output signal off

# A complete program

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11,GPIO.OUT)
while True:
    GPIO.output(11,False)
    time.sleep(1)
    GPIO.output(11,True)
    time.sleep(1)
```

- This sets up the output and then turns it on and off once per second

# Creating an Input

```
GPIO.setup(12, GPIO.IN)
if GPIO.input(12):
    print("input high")
```

- Input pins are created in exactly the same way
- A program can read them to determine the state of the input signal
- Note that with this software we can only poll inputs, we do not have any support for interrupt driven inputs

# A Raspberry Pi Arcade Table

# An Arcade Table of My Own



## An arcade table

- I've always fancied owning my own arcade coffee table
- And the Raspberry Pi make this affordable and possible
  - Cheap
  - Small
  - HDMI output
  - Runs MAME

## MAME

- Multiple Arcade Machine Emulator
- Emulates the hardware of an arcade machine
- Provides an environment where the original arcade ROMS can run
- Available for lots of machines
  - <http://pimame.org/>



## Hardware

- Table from Argos
- Joystick and adapter from ebay
- Flat screen monitor and speakers from Amazon
- Wireless keyboard from Amazon
- Raspberry Pi and USB hub from CPC  
<http://www.robmiles.com/journal/category/pitabletop-arcade>

# Questions

# Links

- [www.raspberrypi.org](http://www.raspberrypi.org)
- [www.modmypi.com](http://www.modmypi.com)
- [www.themagpi.com](http://www.themagpi.com)
- [learn.adafruit.com/category/raspberry-pi](http://learn.adafruit.com/category/raspberry-pi)
- [scratch.mit.edu](http://scratch.mit.edu)
- [www.python.org](http://www.python.org)
- [www.pygame.org](http://www.pygame.org)
- <http://pimame.org/>