Raspberry Pi Workshop

Rev 0

Peter Rogers (peter.rogers@gmail.com)

About this workshop

- This workshop will introduce you to the Raspberry Pi, an inexpensive single-board computer designed for educational and hobbyist use.
- The workshop is aimed at hobbyists, artists and makers who wish to incorporate the Pi into their projects.

What comes in the kit

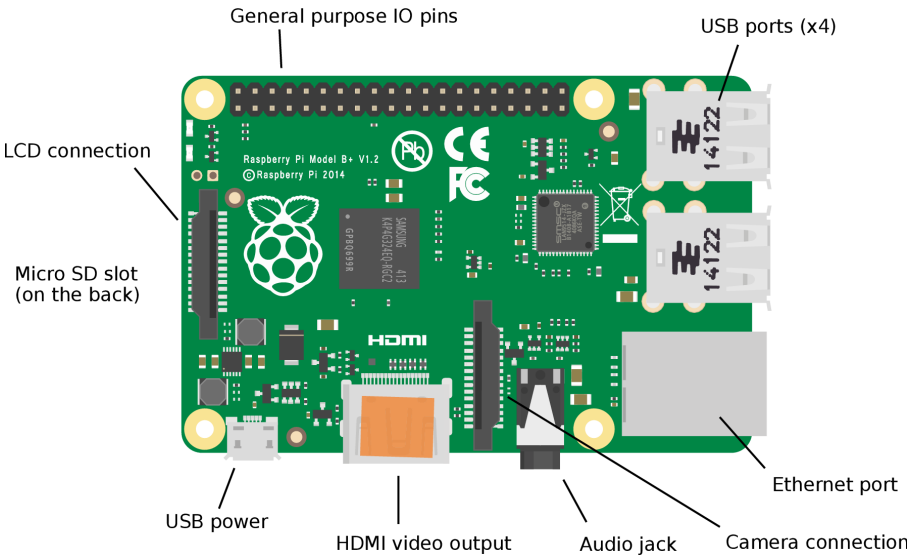
- Raspberry Pi Model B+
- Micro USB cable
- Micro SD card preloaded with Linux
- Various resistors
- Small servo (SG-90)
- Ultra-sonic range finder
- Pi breakout / protection board
- Hall-effect transistor
- MOSFETs
- Zener diodes
- Various LEDs
- Thermistor
- Photoresistor

Overview of the Raspberry Pi

- Created by the Raspberry PI Foundation, a registered charity in the UK.
- Designed as an educational tool for students, hackers, makers
- Similar to the Arduino in some respects, also much more capable (and lacking in other respects)
- Four versions to date (A, A+, B, B+) - all very similar

Rasbperry Pi Specs

- 512 MB RAM
- Broadcom system on a chip
- ARM based processor (most desktops are Intel based)
- Broadcom system on a chip (SoC)
- Boots OS from SD card (hard drive)
- 4x USB ports
- On-board Ethernet
- On-board audio jack (or audio is delivered via HDMI cable)
- USB connector for power (not data)
- Power light shows when power is connected
- General Purpose IO header (40 digital input+output pins)
- No analog input pins, only digital
- Linux installation – Debian (distribution) Wheezy (version)



Warnings – Care and Feeding of your Pi

- The Pi can supply two voltages – 3.3v and 5v
- Don't feed 5v into the board, or bad things will happen
- Be careful of stray wires (disconnect from the PI first, then from your breadboard)
- Power down your PI properly before disconnecting the supply
- Be careful not to eject the SD when handling your Pi
- Avoid touching the Pi while it is powered
- Be careful plugging / unplugging USB devices when powered up. It's physically awkward when plugging them in. (lots of friction and slipping)

A note about USB wall adapters

- The Pi is powered through the USB (input) port
- The model B+ will draw at most 2A so you should be using a PSU that will deliver that much (generally)
- With few number of peripherals you can get away with 1A
- Most laptops and desktops won't be able to supply enough
- Cheap USB wall adapters can be dangerous so be cautious
- Read for more information: http://www.righto.com/2012/03/inside-cheap-phone-charger-and-why-you.html

Installation

- Your kit comes with Linux pre-installed, updated and useful packages installed. (batteries mostly included)
- Install images are available online, and they must be copied onto an SD card using a separate computer
- Installation is not covered here
- See http://www.raspberrypi.org/downloads/ for disk images and instructions

A brief tour of the Pi

- Booting
- Setup instructions (raspi-config utility)
- Starting X
- Some useful built-in features
- Playing video, music, web browser
- Intro to the command-line
- Running programs as root
- Python programming language
- The raspberry pi is a full computer, just smaller
- Connect to monitor / TV with HDMI
- Using an adapter connect with VGA or DVI (will work with most computer displays)
- What if you don't have a spare monitor?
- Or a display with HDMI?

Configuration utility

- raspi-config (starts after a fresh install, or from command-line)
- Allows you to change some basic options
- Change the host name
- Enable/disable onboard peripherals
- Change audio output (through HDMI or the jack)

Fixing the keyboard layout

- You may need to change your default keyboard layout
- Run the *raspi-config* utility
- Choose internationalizing options
- Change keyboard layout
- Choose *Generic 101 key*
- Reboot your Pi

About the workshop files

- There are examples and notes under the *piworkshop* folder
- These are available for download from: https://github.com/parogers/piworkshop
- Update your local copy by running *git pull* in the folder

Connecting to the Pi remotely

Advantages

- You don't need a spare monitor
- Very portable – can develop on the go with a Pi + laptop
- Connect to Pi wirelessly (useful if the Pi isn't accessible)
- Run GUI applications remotely using X

Disadvantages

- No fancy graphics
- Somewhat slower than working locally
- May experience lag spikes if you are connected through wireless

Connect to your Pi using ssh

- Connect on Linux with: *ssh pi@ip_address*
- Or you can use the script *sshpi.sh* which will connect via ssh, not cache the password, and enable X-forwarding
- On Windows you can connect with putty
- On MacOS you can connect with ssh as well

Shutting down the Pi remotely

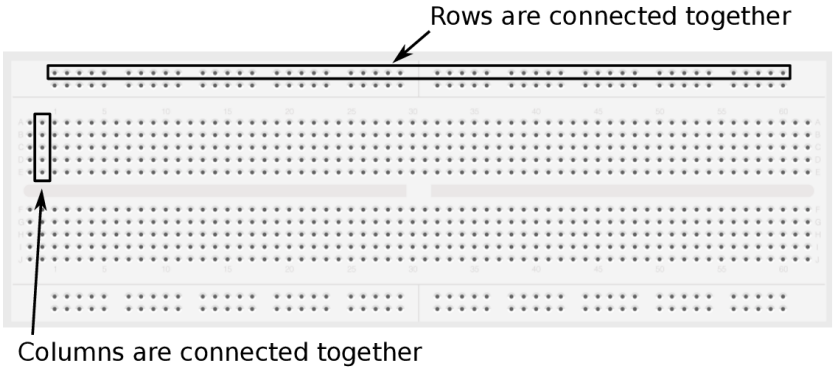
- Run the command: *sudo shutdown -h now*
- The command *sudo* runs as root (super-user)
- Shutdown is the utility for shutting down or rebooting, '-h' requests that the machine halts, and the time is 'now'
- Or reboot with: *sudo reboot*

Finding your Pi on the network

- mDNS protocol allways for DNS resolution without a DNS server (distributed DNS)
- Run a discovery protocol on your Pi (uses avahi package)
- Connect to your Pi with the host name, followed by *.local*
- Example: *ping rasppi.local*
- Linux command to display hosts: *avahi-browse --all*

Breadboards

- Fast prototyping – wire circuits and quickly test things without soldering
- Central columns are common, as are top and bottom rows
- Pre-made jumper wire makes wiring test circuits very easy
- You can mishandle the breadboard generally without problems, but you can't mishandle the Pi



GPIO Header

BCM	Name	GPIO Pinout		Name	BCM
-	3.3v	1	2	5v	-
2	SDA1	3	4	5v	-
3	SCL1	5	6	GND	-
4	GPIO.7	7	8	TX	14
-	GND	9	10	RX	15
17	GPIO.0	11	12	GPIO.1	18
27	GPIO.2	13	14	GND	-
22	GPIO.3	15	16	GPIO.4	23
-	3.3v	17	18	GPIO.5	24
10	MOSI	19	20	GND	-
9	MISO	21	22	GPIO.6	25
11	SCLK	23	24	CE0	8
-	GND	25	26	CE1	7
0	SDA0	27	28	SCL.0	1
5	GPIO.21	29	30	GND	-
6	GPIO.22	31	32	GPIO.26	12
13	GPIO.23	33	34	GND	-
19	GPIO.24	35	36	GPIO.27	16
26	GPIO.25	37	38	GPIO.28	20
-	GND	39	40	GPIO.29	21

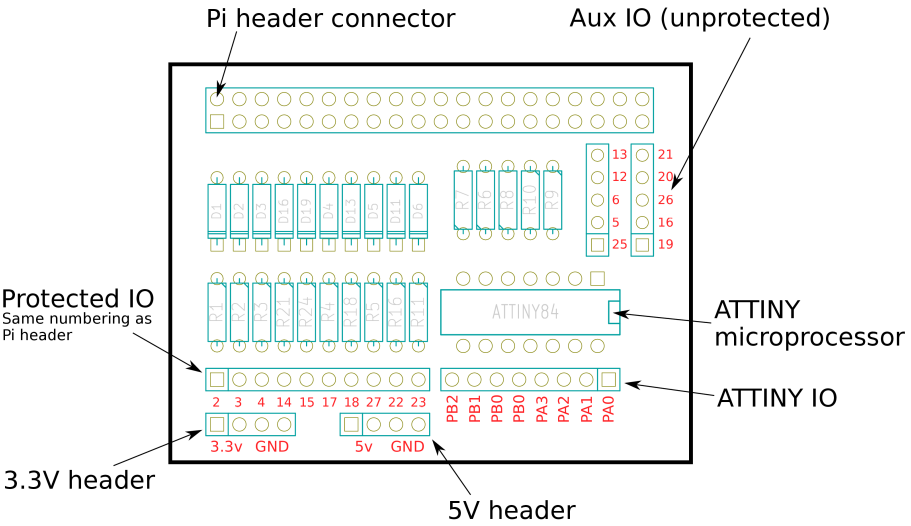
The central two columns (#1-40) refer to the physical GPIO header on the Raspberry Pi. The BCM column refers to the pin numbering on the Broadcom chip. (the CPU)

The pin names are somewhat descriptive and indicate the pins function. (though all pins are GPIO, some can be made special purpose)

- I2C pins are SDA and SCL (there are two I2C buses)
- SPI pins are MISO, MOSI, SCLK and CE0, CE1
- UART pins are TX and RX

PI safety shield

- Protected IO pins prevent you feeding in too much voltage (somewhat)
- Also prevents you from drawing too much current from pins
- Current limiting resistor prevents too much current draw
- How it works – a Zener diode clamps input voltage to 3.3V
- The diode is placed backwards and begins conducting when the *Zener Voltage* is exceeded. (3.3v in this case)

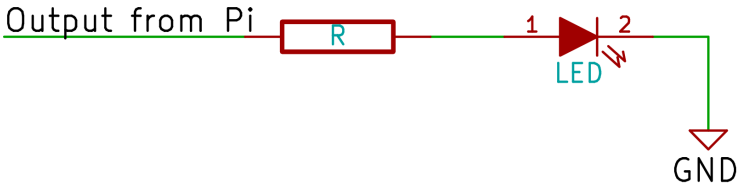


Note: Pins are numbered according to BCM scheme

LED example

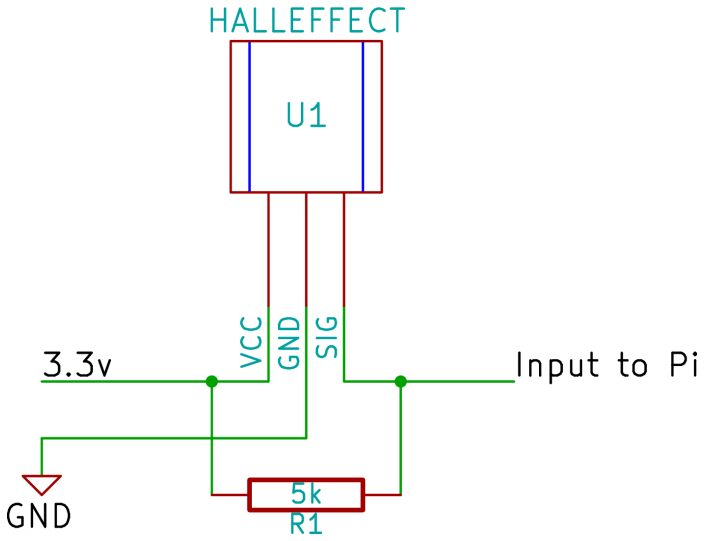
- Connect LED to PI output pin
- Be careful to pick the right resistor, or you will draw too much current and blow the LED / blow the PI
- Choose a high value resistor such as 500 ohm to be safe
- This limits the amount of current that can be drawn from IO
- The maximum current draw from the 3.3v pins is 50mA (that's only a handful of LEDs at proper brightness!)
- The maximum current draw from any IO pin is 16mA (barely a single LED)
- The maximum current draw from all IO pins is 50mA (combined)
- Use an online resistor/LED calculator to determine the proper value of the resistor to use

NOTE: The Pi is not designed to drive heavy loads like LEDs, motors, etc. Always be very careful before plugging into the IO pins. The protection board will provide some buffering and prevent some bad things from happening but it's not a guarantee.



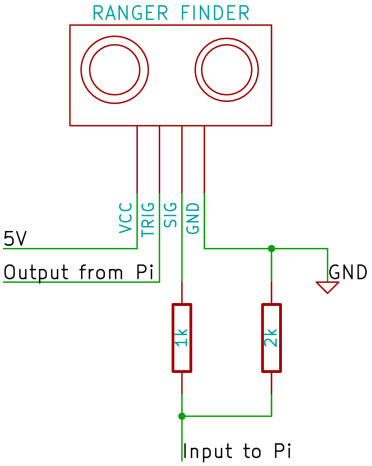
Hall-effect transistor

- Switches in the presence of a magnetic field
- Useful for creating a switch that doesn't need physical contact
- Very close proximity sensor



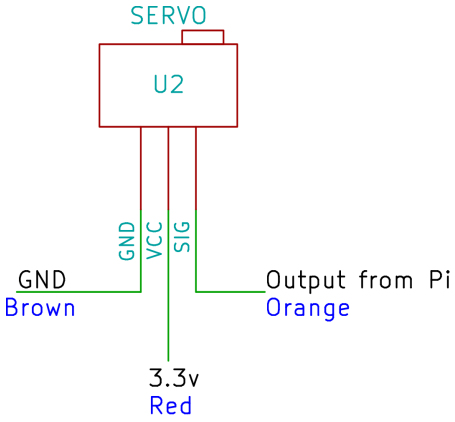
Range-finder example

- Uses ultra-sonic pulses
- Power with 5v (be careful!), returns a 5v signal
- Send a pulse, time how long it takes to come back
- Estimate distance based on the speed of sound
- Not very accurate, has problems with some materials not reflecting sound (possibly multiple reflections too)



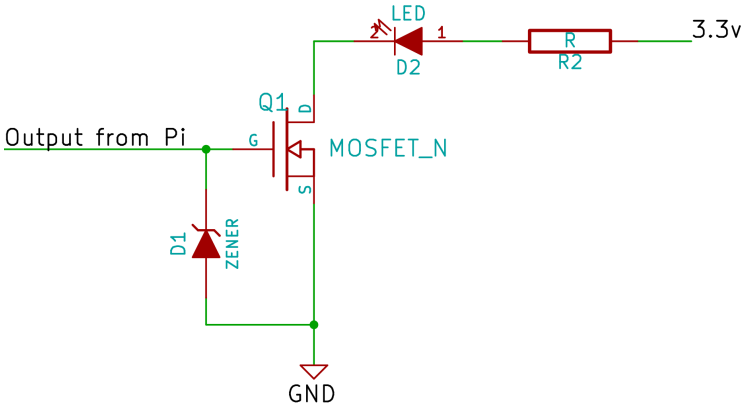
Servo example

- PWM signal determines angle of rotation
- Motor has feedback so it “knows” where it is in the rotation
- Use servos to create robots, interactive pieces
- Don't connect too many of these to the Pi directly without a better power source (or too large of a single servo)



MOSFET example

- Driving an LED, multiple LEDs
- You need to be very careful when handling MOSFETs, they are very sensitive to ESD (electro-static discharge)
- Mishandling them can damage or destroy them very quickly
- We use a Zener diode to protect them from accidental ESD (same component on the protection board)

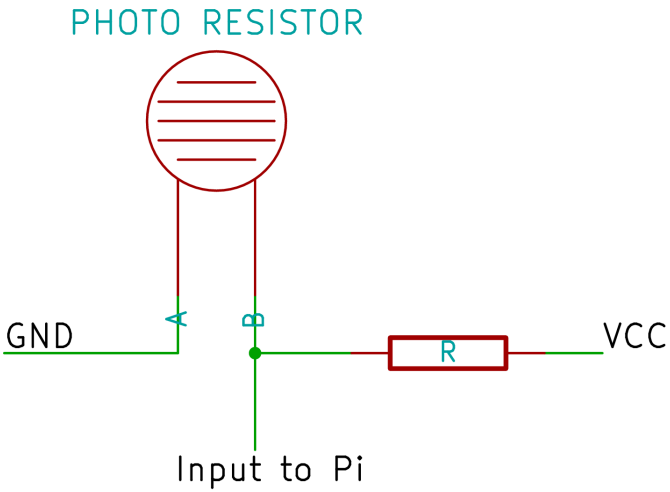


Using the on-board ATTINY for extra functionality

- The protection board has a built-in ATTINY microprocessor
- Can be programmed with the Arduino IDE via SPI bus
- Uses a modified version of avrdude and Arduino ATTINY extension
- Can utilize the attiny's built-in analog to digital converter

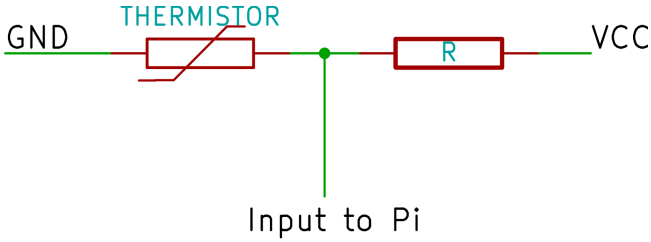
Photo resistor example

- Measure light levels
- The photo cell changes resistance based on received light
- Basic voltage divider circuit



Thermistor example

- Resistor that changes resistance based on temperature (similar to a photo resistor)



Notes