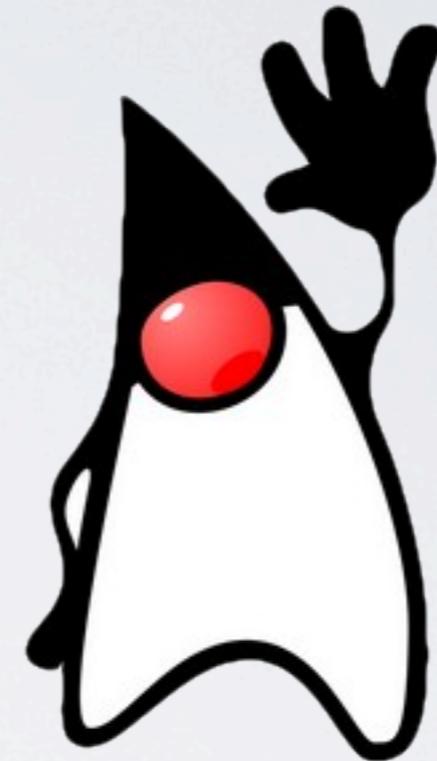


Raspberry Pi

3.14159265358979323846264338327950288419716939937
5105820974944592307816406286 2089986280348253421170679
82148086513282306647093844609550582231725359408128481117
450284102701938521105559644622948954930381964428810975665
93344612847564823378678316527120190914564856692346034861045
432 6648213 3936072
602 4914127 3724587
00 6606315 5881748
8 1520920 9628292
5409171 5364367
8925903 60011330
5305488 20466521
3841469 51941511
6094330 57270365
7595919 53092186
11738193 26117931
05118548 07446237
996274956 735188575 272
4891227938 1830119491 2983
36733624406 56643086021 39494
6395224737190 702179860943 70277
053921717629317 67523846748184676694051
320005681271452 635608277857713427577
89609173637178 7214684409012249534
3014654958537 1050792279689258
923542019 9561121290
21960 864034



Bekk Consulting
Tuesday 2. April

These slides are available at:
http://stuff.haagen.name/raspberrypi_bekk.pdf

3.14159265358979323846264338327950288419716939937
5105820974944592307816406286 2089986280348253421170679
82148086513282306647093844609550582231725359408128481117
450284102701938521105559644622948954930381964428810975665
93344612847564823378678316527120190914564856692346034861045
432 6648213 3936072
602 4914127 3724587
00 6606315 5881748
8 1520920 9628292
5409171 5364367
8925903 60011330
5305488 20466521
6094330 57270365
11738193 26117931
05118548 07446237
996274956 735188575 272
4891227938 1830119491 2983
36733624406 56643086021 39494
6395224737190 702179860943 70277
053921717629317 67523846748184676694051
320005681271452 635608277857713427577
89609173637178 7214684409012249534
3014654958537 1050792279689258
923542019 9561121290
21960 864034

About me

Joachim Haagen Skeie

About me

Joachim Haagen Skeie

Owner @ Haagen Software AS
Independent Consultant and Course Instructor

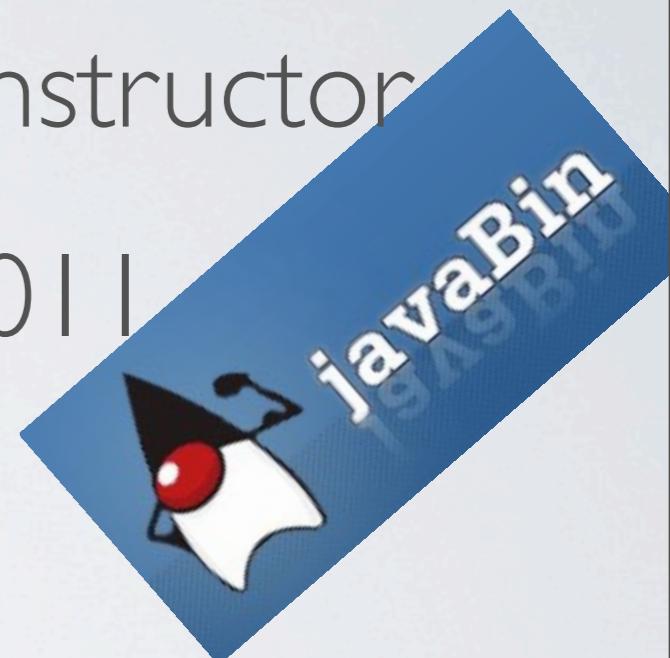
About me

Joachim Haagen Skeie

Owner @ Haagen Software AS

Independent Consultant and Course Instructor

Boardmember @ javaBin since 2011



joachim@haagen-software.no

@joachimhs

hs

haagen
software AS

About me

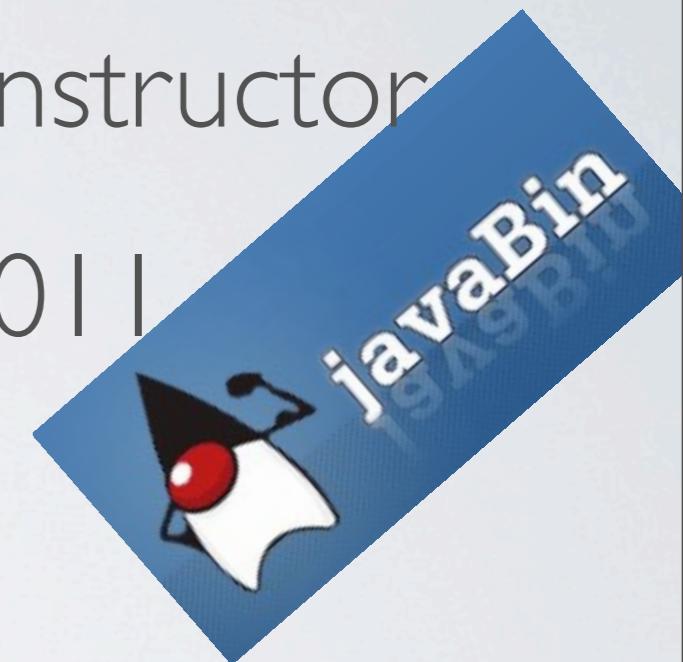
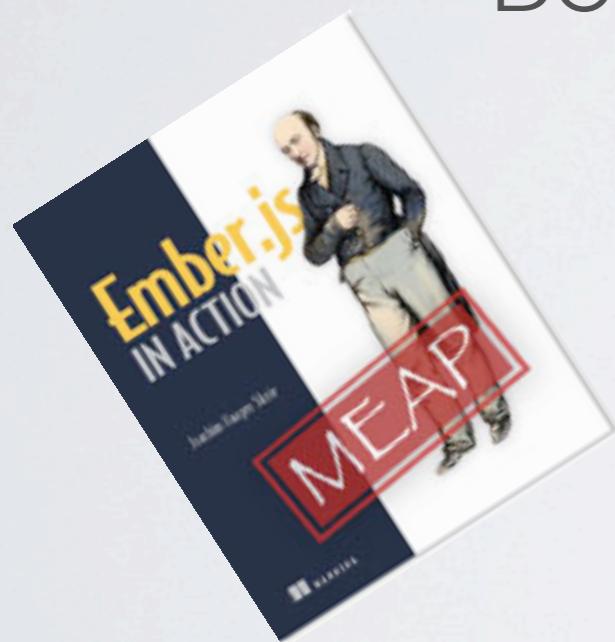
Joachim Haagen Skeie

Owner @ Haagen Software AS

Independent Consultant and Course Instructor

Boardmember @ javaBin since 2011

Author @ Ember.js in Action

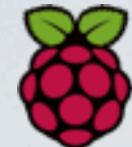


Email: joachim@haagen-software.no

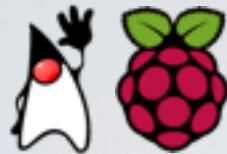
Twitter: @joachimhs

Todays achievements

16:00 - 17:30



Introduction to Raspberry Pi



Installation of Raspbian and OpenJDK



Setup of our workstations and getting started

17:45 - 19:00



Introduction to GPIO and Breadboards



Building our very first electrical circuit

19:30 - 22:00



Building a memory-game using LEDs and GPIO



Using input buttons and measuring temperature

What is a Raspberry Pi ?

A Complete Miniature Computer

700 MHz ARM CPU

512 MB RAM (2012)

Ethernet, USB, HDMI, GPIO

45 grams

U\$D 35



What is a Raspberry Pi ?

Can run multiple OSes

Raspian recommended (Debian)

Alpha/Beta versions of
MozillaOS
Android
WebOS

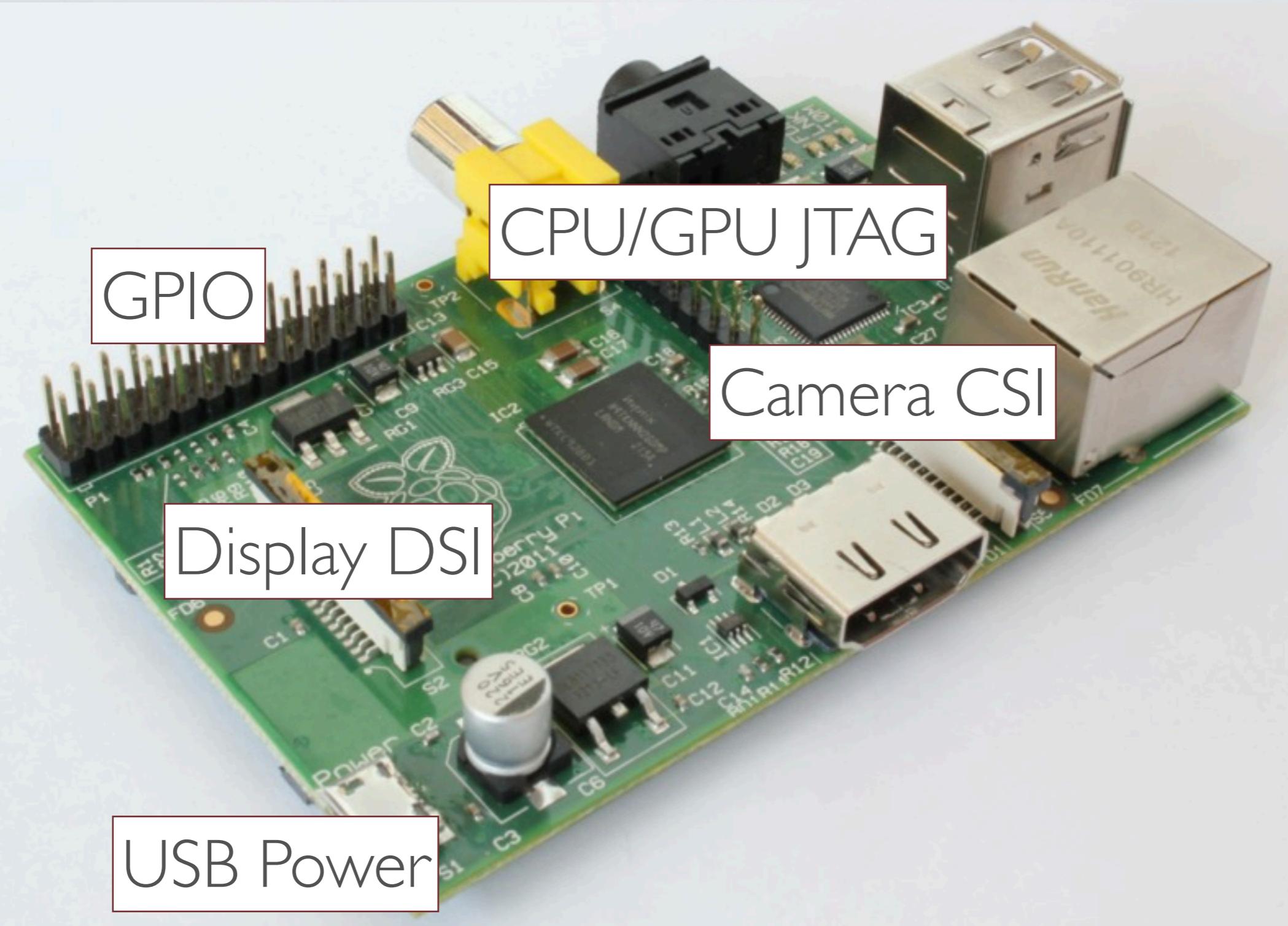
Oracle Embedded Java

OpenJDK

What is a Raspberry Pi ?



What is a Raspberry Pi ?



Downloading Raspian

Choose between Soft-float and Hard-float

OpenJDK supports Hard-float, while Oracles Embedded JRE only support Soft-float

We will be installing the Hard-float version and OpenJDK

<http://www.raspberrypi.org/downloads>

Raspbian "wheezy"

If you're just starting out, this is the image we recommend you use. It's a reference root filesystem from Alex and Dom, based on the [Raspbian](#) optimised version of Debian, and containing LXDE, Midori, development tools and example source code for multimedia functions.

Torrent

[2013-02-09-wheezy-raspbian.zip.torrent](#)

Direct download

[2013-02-09-wheezy-raspbian.zip](#)

SHA-1

b4375dc9d140e6e48e0406f96dead3601fac6c81

Default login

Username: pi Password: raspberry

Installing Raspbian

There is a comprehensive installation guide available for your OS:

http://elinux.org/RPi_Easy_SD_Card_Setup

Mac / Linux

Insert the SD card in your card-reader and find its name

```
joahaa:raspi joahaa$ df -h
Filesystem      Size   Used  Avail Capacity  iused   ifree %iused  Mounted on
/dev/disk0s2    112Gi  105Gi  7.1Gi   94%  27549529 1862839   94%   /
devfs          227Ki  227Ki   0Bi   100%     786      0  100%   /dev
map -hosts      0Bi    0Bi    0Bi   100%       0      0  100%   /net
map auto_home   0Bi    0Bi    0Bi   100%       0      0  100%   /home
/dev/disk1s2    145Mi  145Mi   0Bi   100%    74314      0  100%   /Volumes/Google Chrome
/dev/disk13s1    15Gi   2.1Mi   15Gi    1%       0      0  100%   /Volumes/NO NAME
joahaa:raspi joahaa$ sudo diskutil unmount /dev/disk13s1
Password:
Volume NO NAME on disk13s1 unmounted
```

/dev/disk13s1 => rdisk13

IMPORTANT: Getting this wrong can erase your system HD!!

```
joahaa:raspi joahaa$ sudo dd bs=1m if=2012-08-08-wheezy-armel.img of=/dev/rdisk13
1850+0 records in
1850+0 records out
1939865600 bytes transferred in 109.348608 secs (17740195 bytes/sec)
```

When completed, eject the SD card

Logging in via SSH

You need an SSH client

Windows: Putty for SSH, WinSCP for file transfers

Mac/Linux: Use the terminal/console with ssh and scp

ssh pi@your-pis-IP-address

password: raspberry

Logging in via SSH

You need an SSH client

Windows: Putty for SSH, WinSCP for file transfers

Mac/Linux: Use the terminal/console with ssh and scp

ssh pi@your-pis-IP-address
password: raspberry

```
joahaa:target joahaa$ ssh pi@192.168.2.2
pi@192.168.2.2's password:
Linux raspberrypi 3.1.9+ #272 PREEMPT Tue Aug 7 22:51:44 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

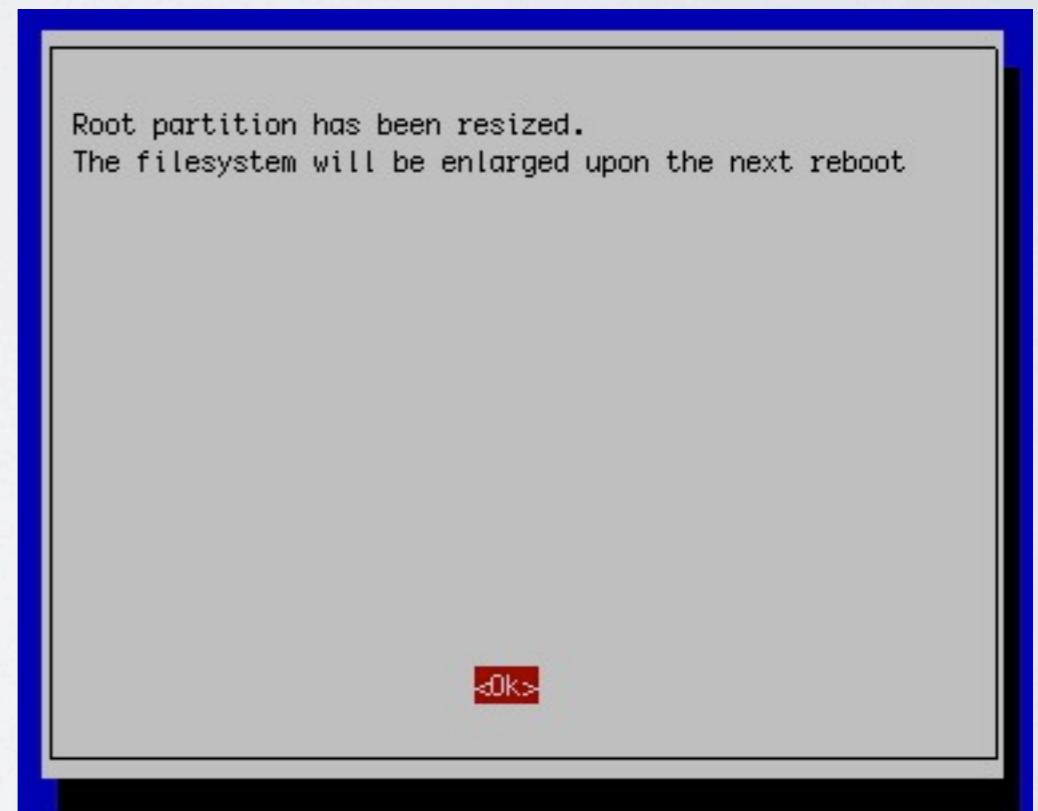
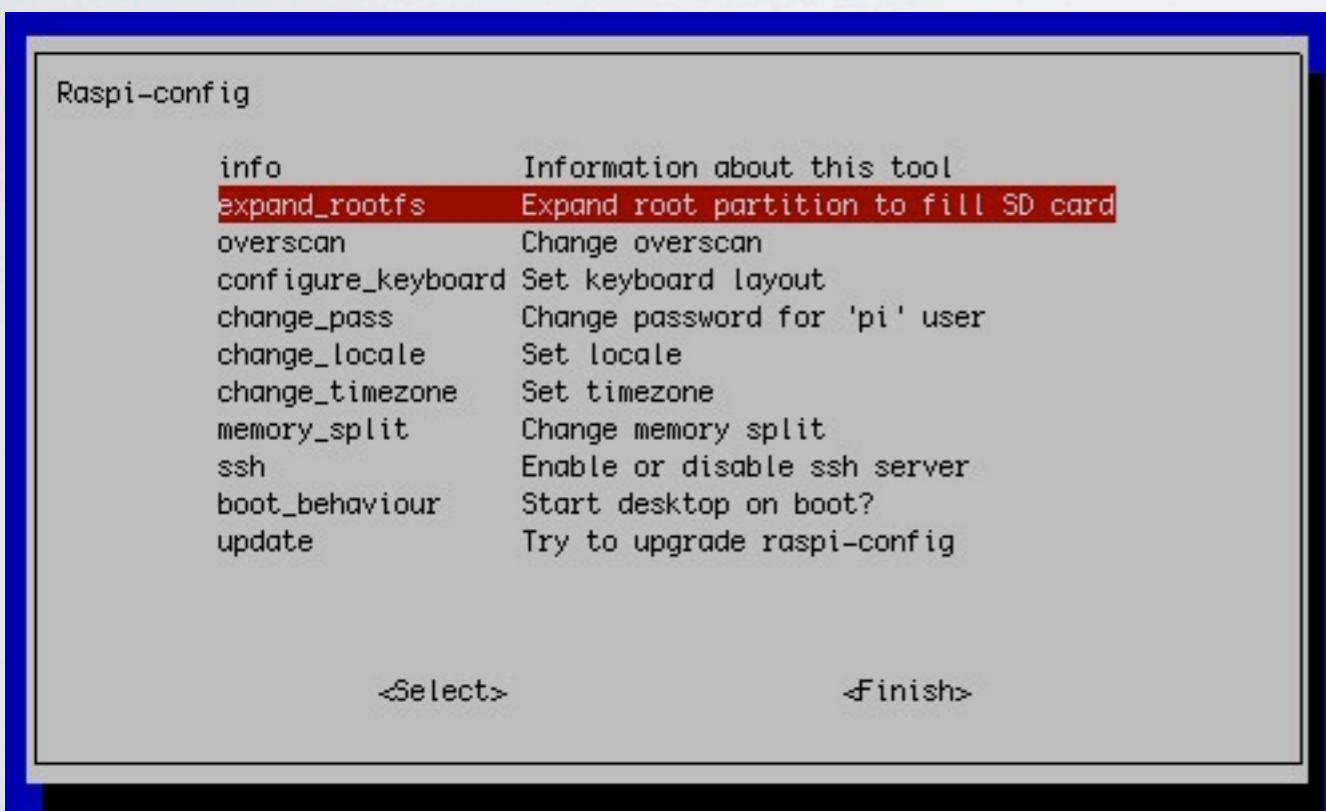
Type 'startx' to launch a graphical session

Last login: Wed Aug 8 23:01:30 2012 from 192.168.2.1
pi@raspberrypi ~ $ |
```

Expanding the OS Partition

```
pi@raspberrypi ~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
rootfs          1.8G  1.4G  309M  82% /
/dev/root        1.8G  1.4G  309M  82% /
tmpfs           19M   200K  19M   2% /run
tmpfs           5.0M    0  5.0M   0% /run/lock
tmpfs           37M    0  37M   0% /tmp
tmpfs           10M    0  10M   0% /dev
tmpfs           37M    0  37M   0% /run/shm
/dev/mmcblk0p1   56M   35M  22M  61% /boot
```

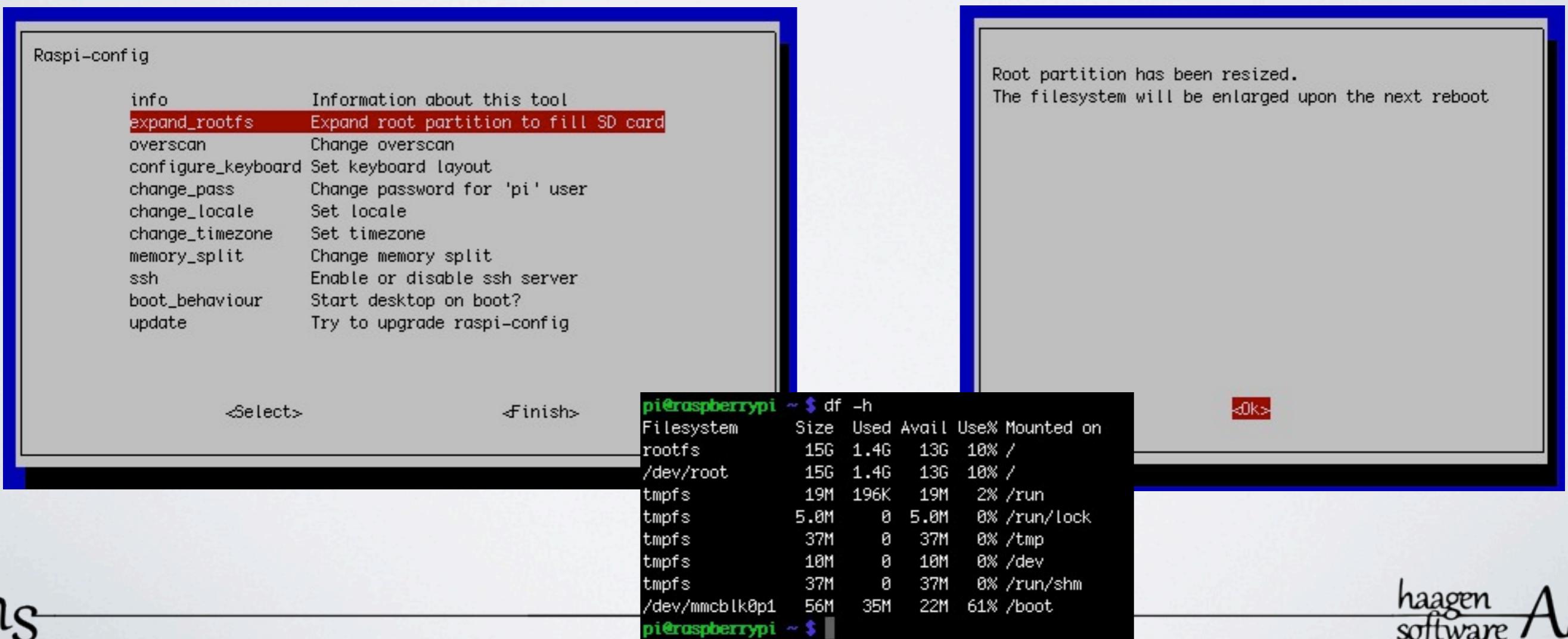
sudo raspi-config



Expanding the OS Partition

```
pi@raspberrypi ~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
rootfs          1.8G  1.4G  309M  82% /
/dev/root        1.8G  1.4G  309M  82% /
tmpfs           19M   200K  19M   2% /run
tmpfs           5.0M    0  5.0M   0% /run/lock
tmpfs           37M    0  37M   0% /tmp
tmpfs           10M    0  10M   0% /dev
tmpfs           37M    0  37M   0% /run/shm
/dev/mmcblk0p1   56M   35M  22M  61% /boot
```

sudo raspi-config



Installing OpenJDK

We will be using apt-get the Debian package manager

Start by updating the apt-get index

```
pi@raspberrypi ~ $ sudo apt-get clean  
pi@raspberrypi ~ $ sudo apt-get update  
0% [Connecting to http.debian.net] [Connecting to archive.raspberrypi.org]
```

Install OpenJDK

```
pi@raspberrypi ~ $ sudo apt-get install openjdk-7-jdk  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following extra packages will be installed:  
  ca-certificates-java gnome-mime-data icedtea-7-jre-cacao  
  libgnome2-common libgnomevfs2-0 libgnomevfs2-common libgnomevfs2-common  
  libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-7-jdk  
  xorg-sgml-doctools xtrans-dev
```

```
pi@raspberrypi ~ $ java -version  
java version "1.7.0_10"  
Java(TM) SE Embedded Runtime Environment (build 1.7.0_10-b18, headless)  
Java HotSpot(TM) Embedded Client VM (build 23.6-b04, mixed mode)
```

Introduction to GPIO

GPIO = General Purpose Input Output

RaspberryPi have 8 true GPIO pins

26 pins in total

One 1.3V, one 1.5V and one Ground

6 pins are Do Not Connect(DNC)

The other pins have a special purpose,
like serial communication

Raspberry Pi P1 Header	
PIN #	NAME
1	3.3 VDC Power
8	SDA0 (I2C)
9	SCL0 (I2C)
7	GPIO 7
6	DNC
0	GPIO 0
2	GPIO2
3	GPIO3
12	MOSI
13	MISO
14	SCLK
52	DNC
2	5.0 VDC Power
4	DNC
6	0V (Ground)
8	TxD
10	RxD
11	GPIO1
12	DNC
14	GPIO4
16	GPIO5
18	GPIO6
20	DNC
22	CE0
24	CE1
26	CE1

<http://www.pi4j.com>

Pi4J (pi4j.com)

Provisioning pins as output

```
GpioController gpioController = GpioFactory.getInstance();
gpioStateList.get(0).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_00, "gpio0", PinState.LOW));
gpioStateList.get(1).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_01, "gpio1", PinState.LOW));
gpioStateList.get(2).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_02, "gpio2", PinState.LOW));
gpioStateList.get(3).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_03, "gpio3", PinState.LOW));
gpioStateList.get(4).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_04, "gpio4", PinState.LOW));
gpioStateList.get(5).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_05, "gpio5", PinState.LOW));
gpioStateList.get(6).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_06, "gpio6", PinState.LOW));
gpioStateList.get(7).setGpioPinDigitalOutput(gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_07, "gpio7", PinState.LOW));
```

```
public void setGpioState(Boolean gpioState) {
    if (gpioPinDigitalOutput != null) {
        int counter = 0;
        if (gpioState) {
            gpioPinDigitalOutput.setState(PinState.HIGH);
            while (!gpioPinDigitalOutput.isHigh()) {
                counter++;
            }
        } else {
            gpioPinDigitalOutput.setState(PinState.LOW);
            while (!gpioPinDigitalOutput.isLow()) {
                counter++;
            }
        }
        pinStateChangeTime = (long)counter;
        logger.info("Pin changed to " + gpioPinDigitalOutput.getState() + " in " + pinStateChangeTime + " clock cycles.");
    }
    this.gpioState = gpioState;
}
```

Pi4J

Provisioning pins as input

```
GpioPinDigitalInput input = gpioController.provisionDigitalInputPin(RaspiPin.GPIO_07, "gpio7", PinPullResistance.PULL_DOWN);
input.addListener(new GpioPinListenerDigital() {
    @Override
    public void handleGpioPinDigitalStateChangeEvent(GpioPinDigitalStateChangeEvent gpioPinDigitalStateChangeEvent) {
        logger.info("PIN Changed to: " + gpioPinDigitalStateChangeEvent.getState());
    }
});
```

Install Pi4J

Get the pi4j.deb Package

```
pi@raspberrypi ~ $ wget http://pi4j.googlecode.com/files/pi4j-0.0.5-SNAPSHOT.deb
```

Install package

```
pi@raspberrypi ~ $ sudo dpkg -i pi4j-0.0.5-SNAPSHOT.deb  
(Reading database ... 62444 files and directories currently installed.)  
Preparing to replace pi4j 0.0.5+SNAPSHOT (using pi4j-0.0.5-SNAPSHOT.deb) ...  
Unpacking replacement pi4j ...  
Setting up pi4j (0.0.5+SNAPSHOT) ...
```

This will install libraries to **/opt/pi4j/lib**

```
pi@raspberrypi ~ $ javac -classpath .:classes:/opt/pi4j/lib/*
```

```
pi@raspberrypi ~ $ sudo java -classpath .:classes:/opt/pi4j/lib/*
```

Dont forget about sudo!

Dont forget about sudo!

```
Exception in thread "main" java.lang.RuntimeException: Unable to open GPIO export interface: Permission denied
```

```
at com.pi4j.wiringpi.GpioUtil.export(Native Method)
at com.pi4j.io.gpio.RaspiGpioProvider.export(RaspiGpioProvider.java:66)
at com.pi4j.io.gpio.impl.GpioPinImpl.export(GpioPinImpl.java:151)
at com.pi4j.io.gpio.impl.GpioControllerImpl.provisionPin(GpioControllerImpl.java:561)
at com.pi4j.io.gpio.impl.GpioControllerImpl.provisionDigitalOutputPin(GpioControllerImpl.java:614)
at com.pi4j.io.gpio.impl.GpioControllerImpl.provisionDigitalOutputPin(GpioControllerImpl.java:627)
at com.pi4j.io.gpio.impl.GpioControllerImpl.provisionDigitalOutputPin(GpioControllerImpl.java:640)
at no.haagensoftware.netty.webserver.pipeline.NettyWebserverPipelineFactory.<init>(NettyWebserverPipelineFactory.java:62)
at no.haagensoftware.netty.server.Main.run(Main.java:86)
at no.haagensoftware.netty.server.Main.main(Main.java:24)
```

Tell Pi4J the board revision!

We need to tell Pi4J that we have a rev 2 board

Edit /boot/config.txt

```
pi@raspberrypi ~ $ sudo nano /boot/config.txt
```

Add program_board_rev=2 to the top of the file

```
program_board_rev=2

# uncomment if you get no picture on HDMI for a default "safe" mode
#hdmi_safe=1
```

Exit nano with CTRL+x

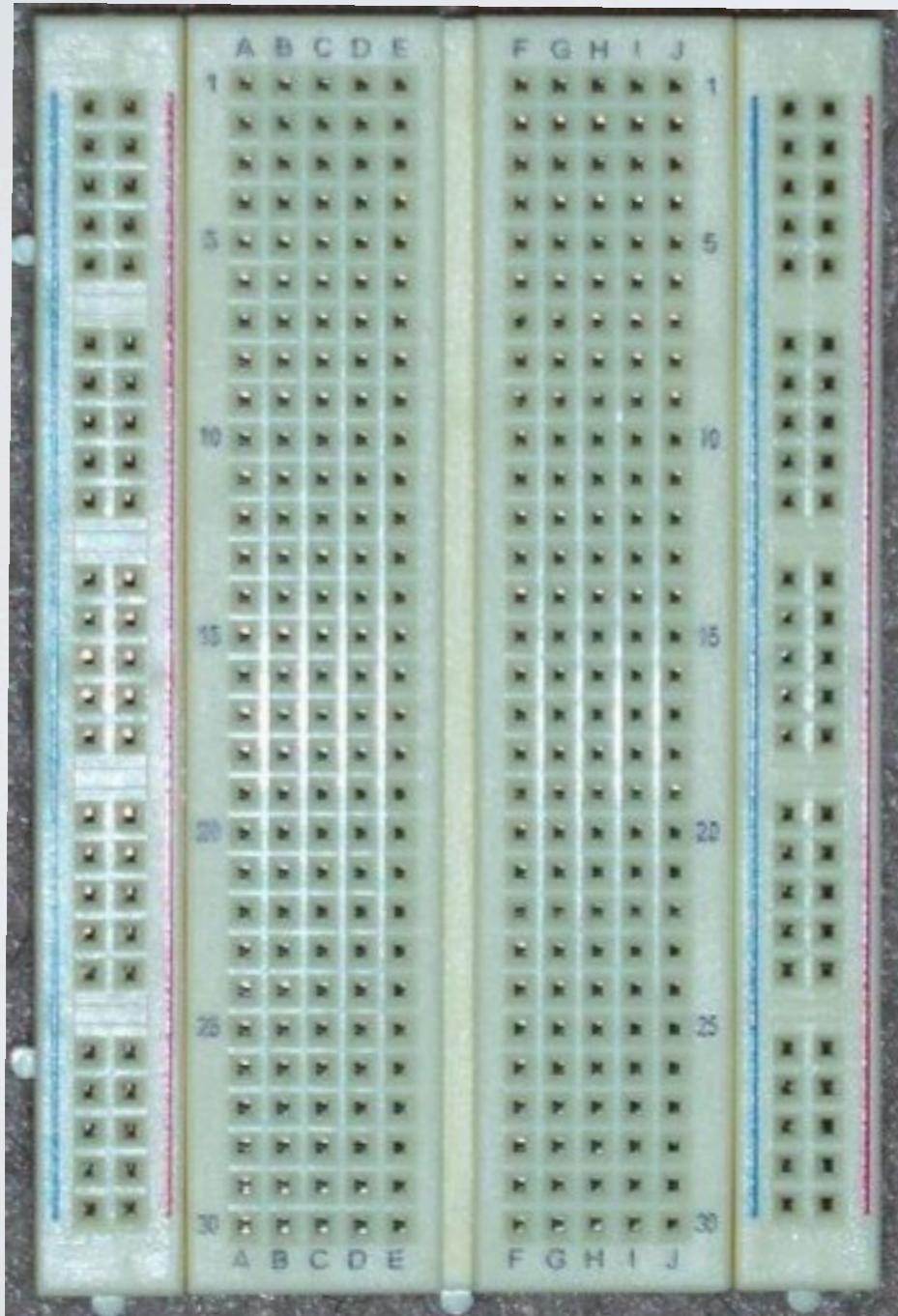
Reboot the Pi with ‘sudo shutdown -r now’

```
pi@raspberrypi ~ $ sudo shutdown -r now

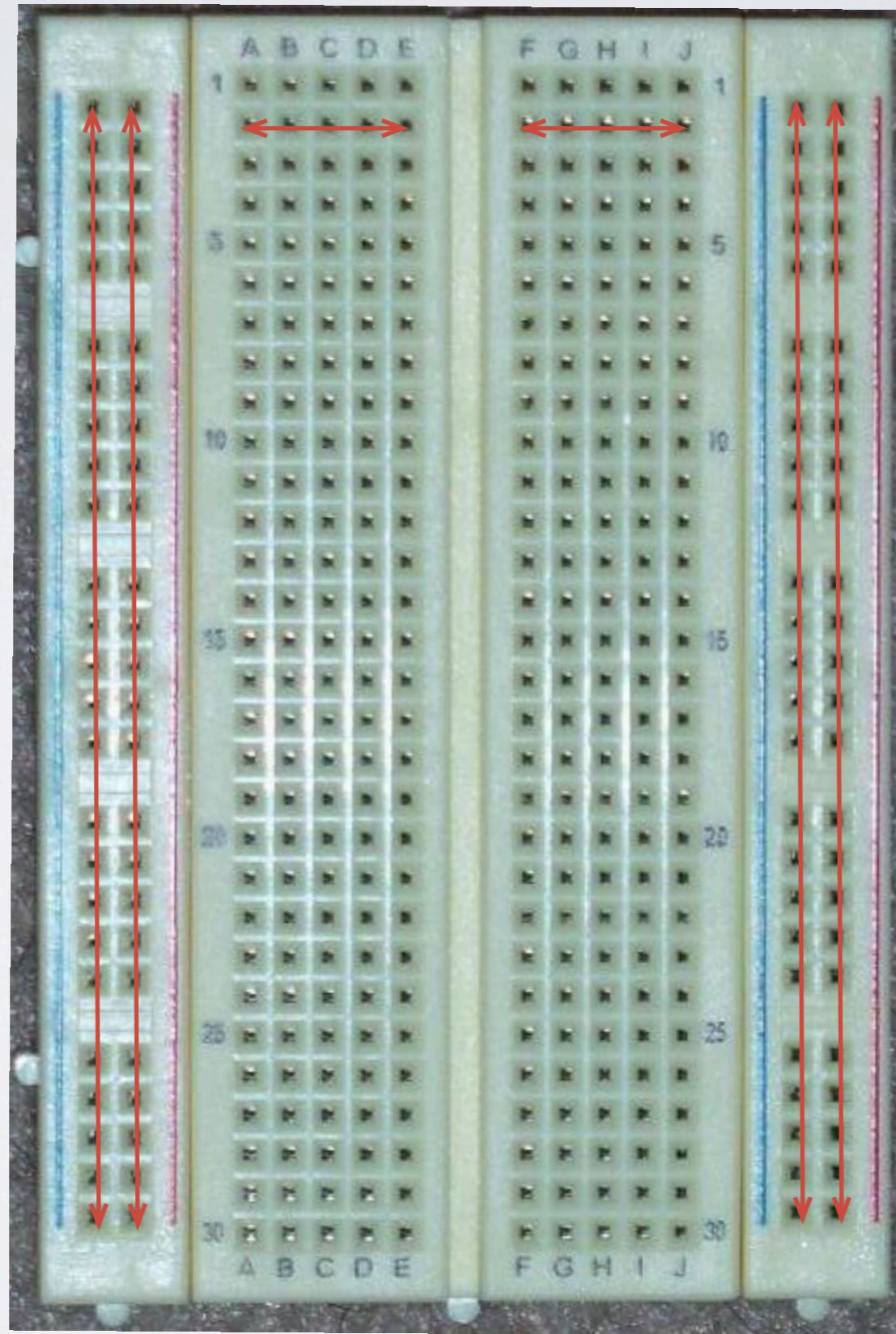
Broadcast message from root@raspberrypi (pts/0) (Thu Mar 14 22:45:26 2013):
The system is going down for reboot NOW!
pi@raspberrypi ~ $ Connection to 192.168.2.5 closed by remote host.
Connection to 192.168.2.5 closed.
```

Really quick intro to the Breadboard

You have/will each received a solder-less breadboard and a bag of tech-goodies :)

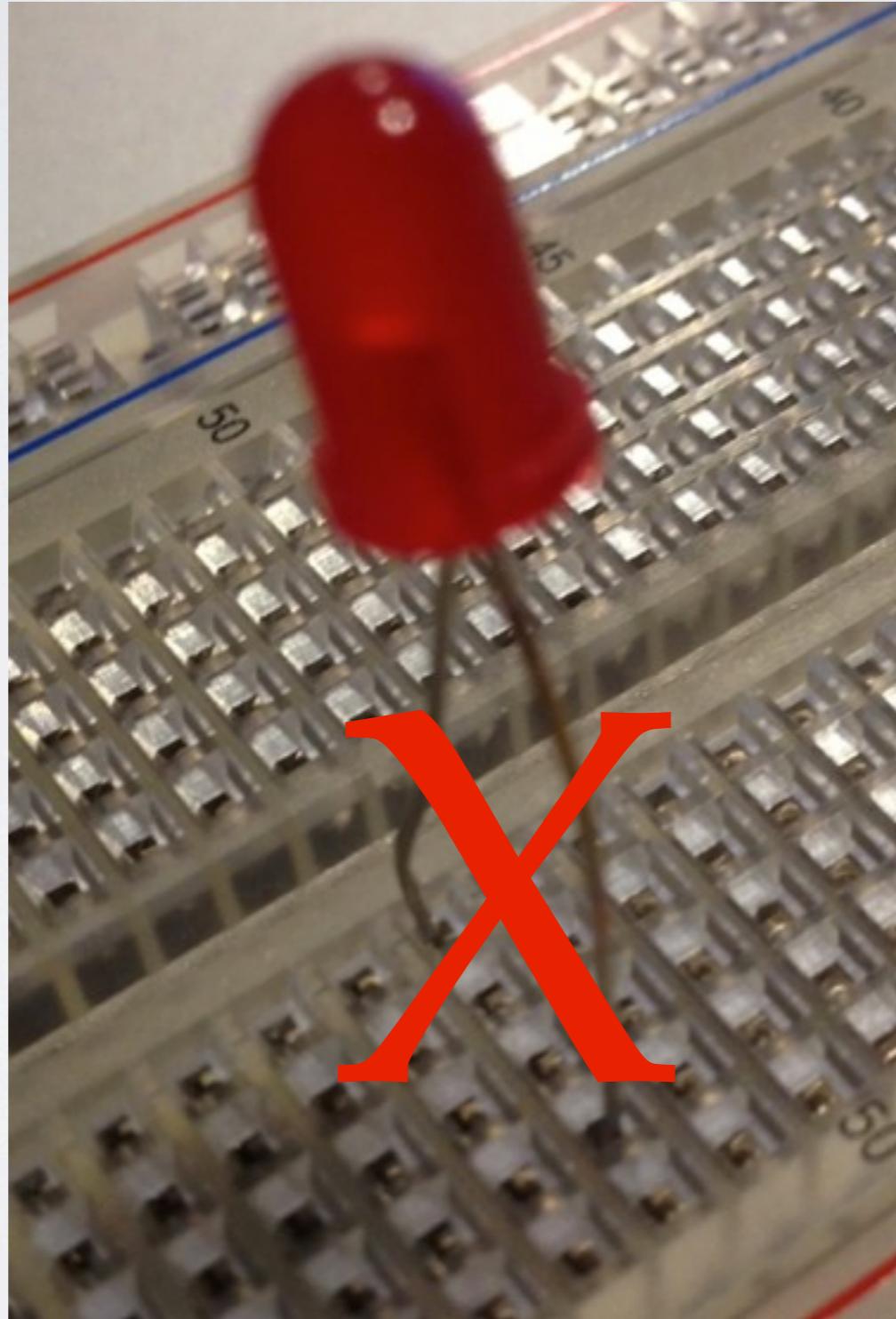


Really quick intro to the Breadboard



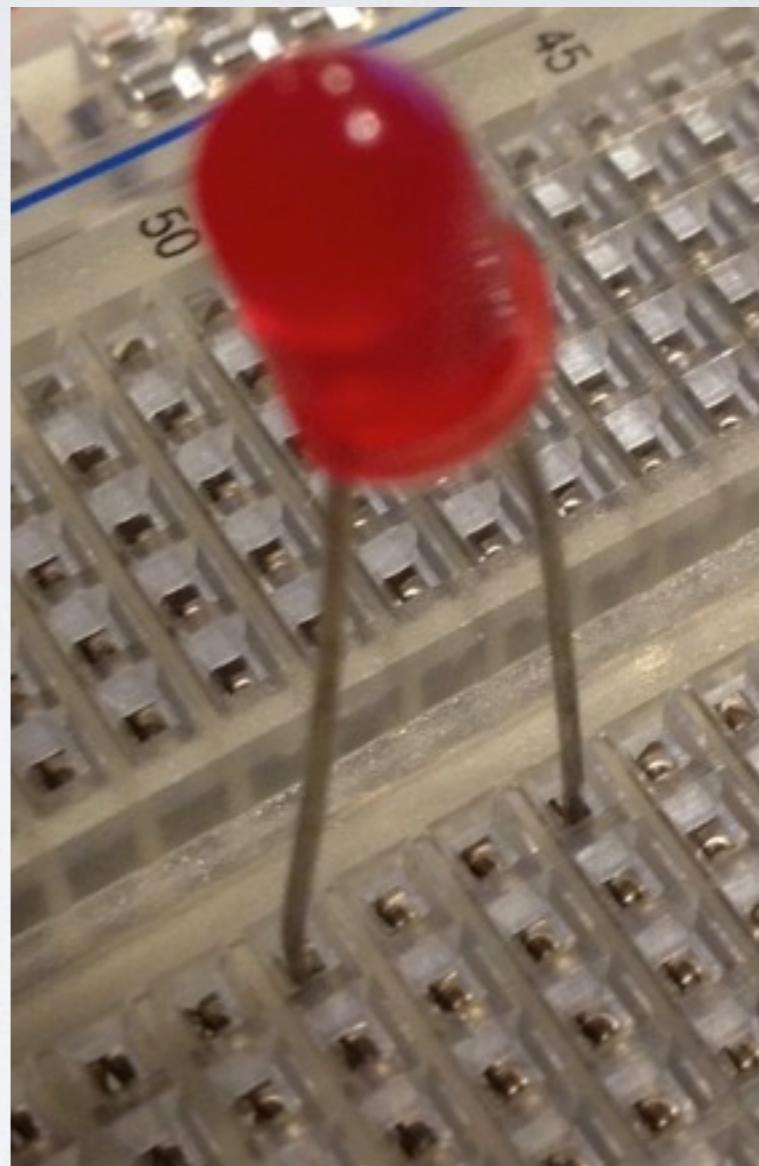
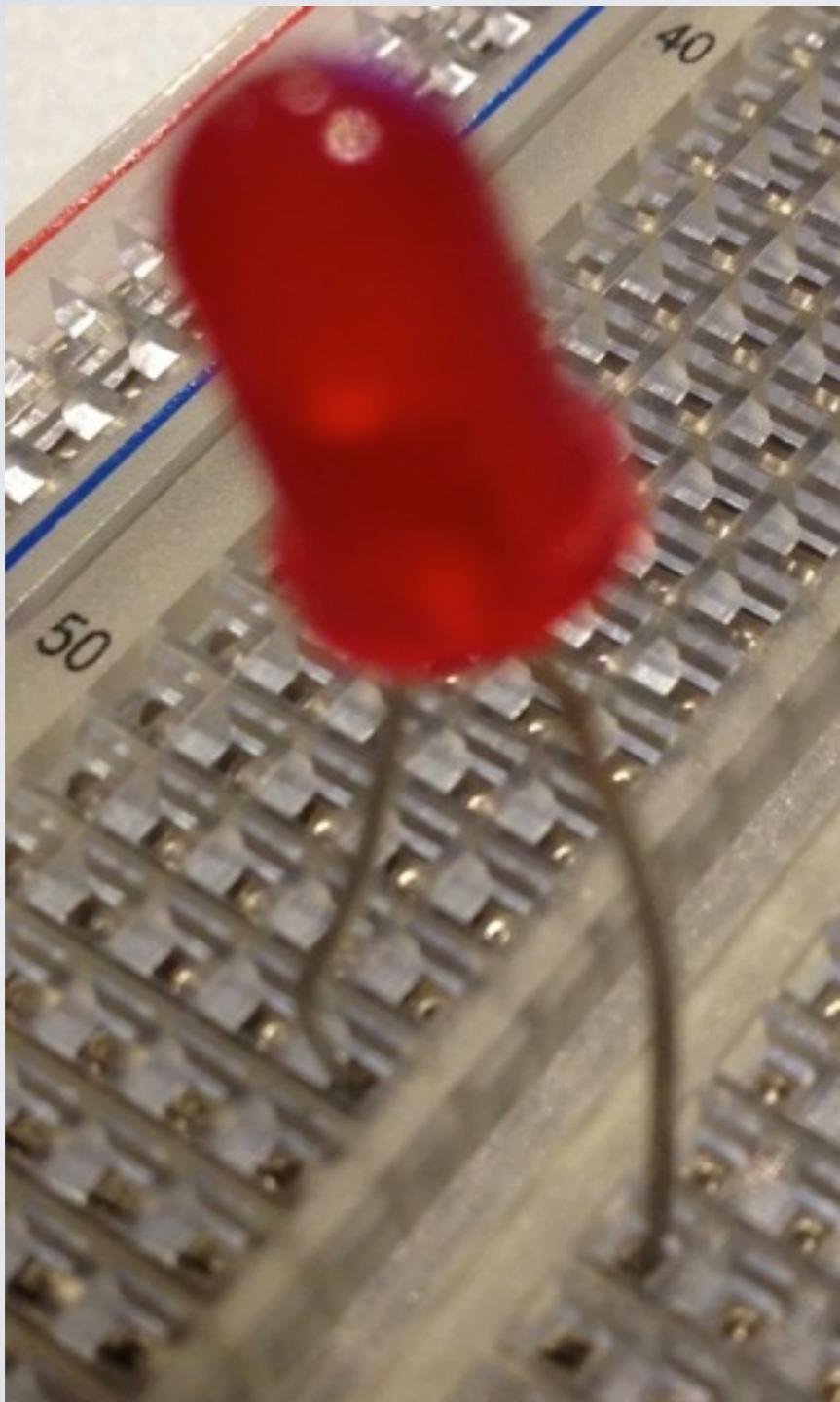
Really quick intro to the Breadboard

Don't short your components!

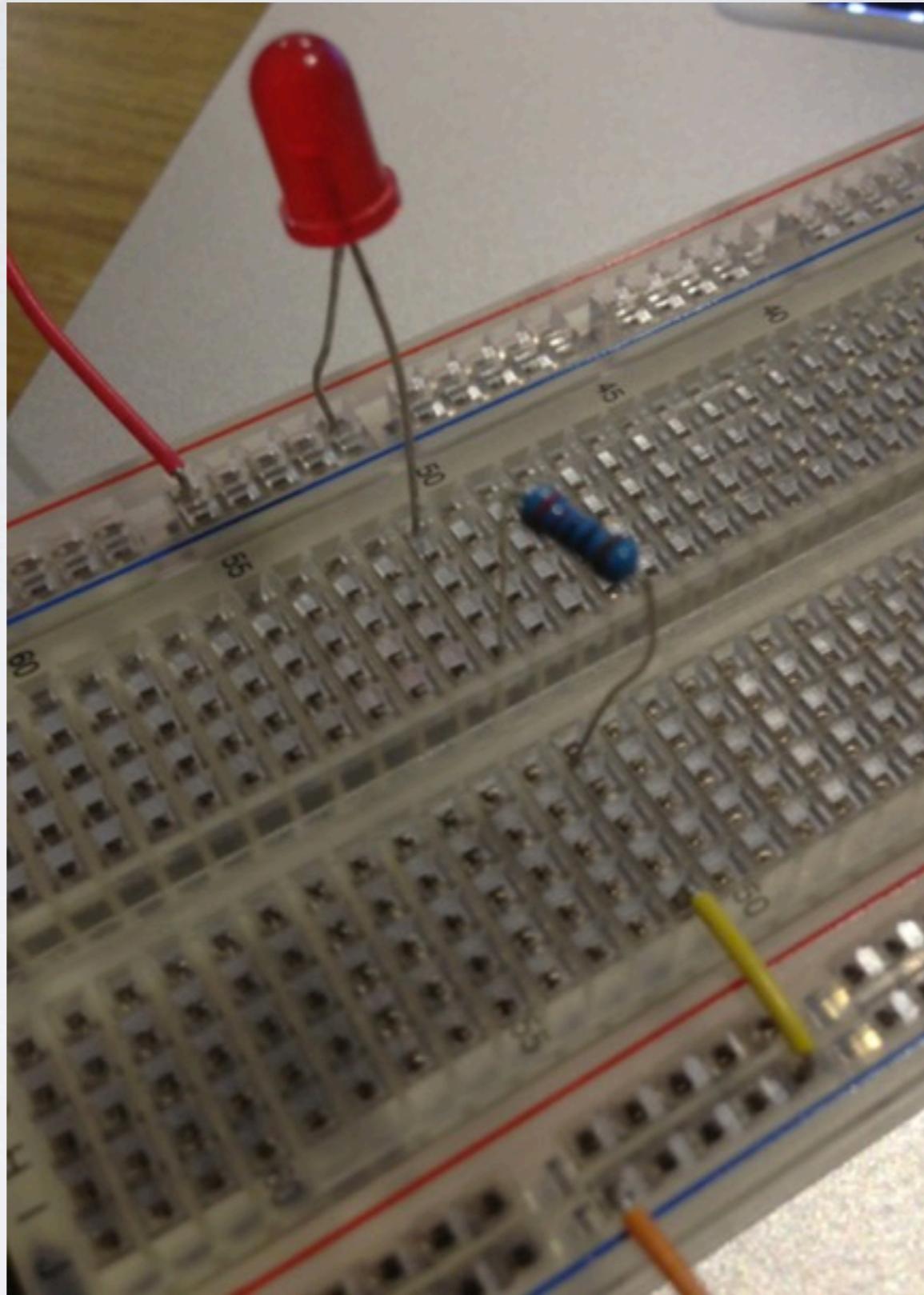


Really quick intro to the Breadboard

Connect across the bridge, or across rows



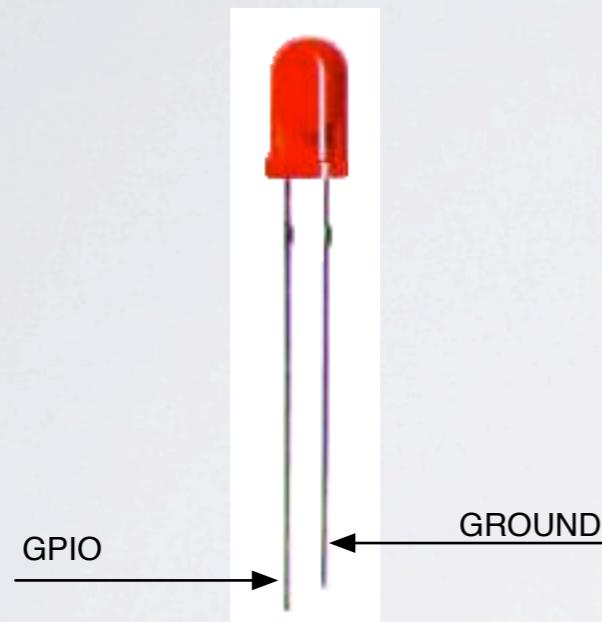
Really quick intro to the Breadboard



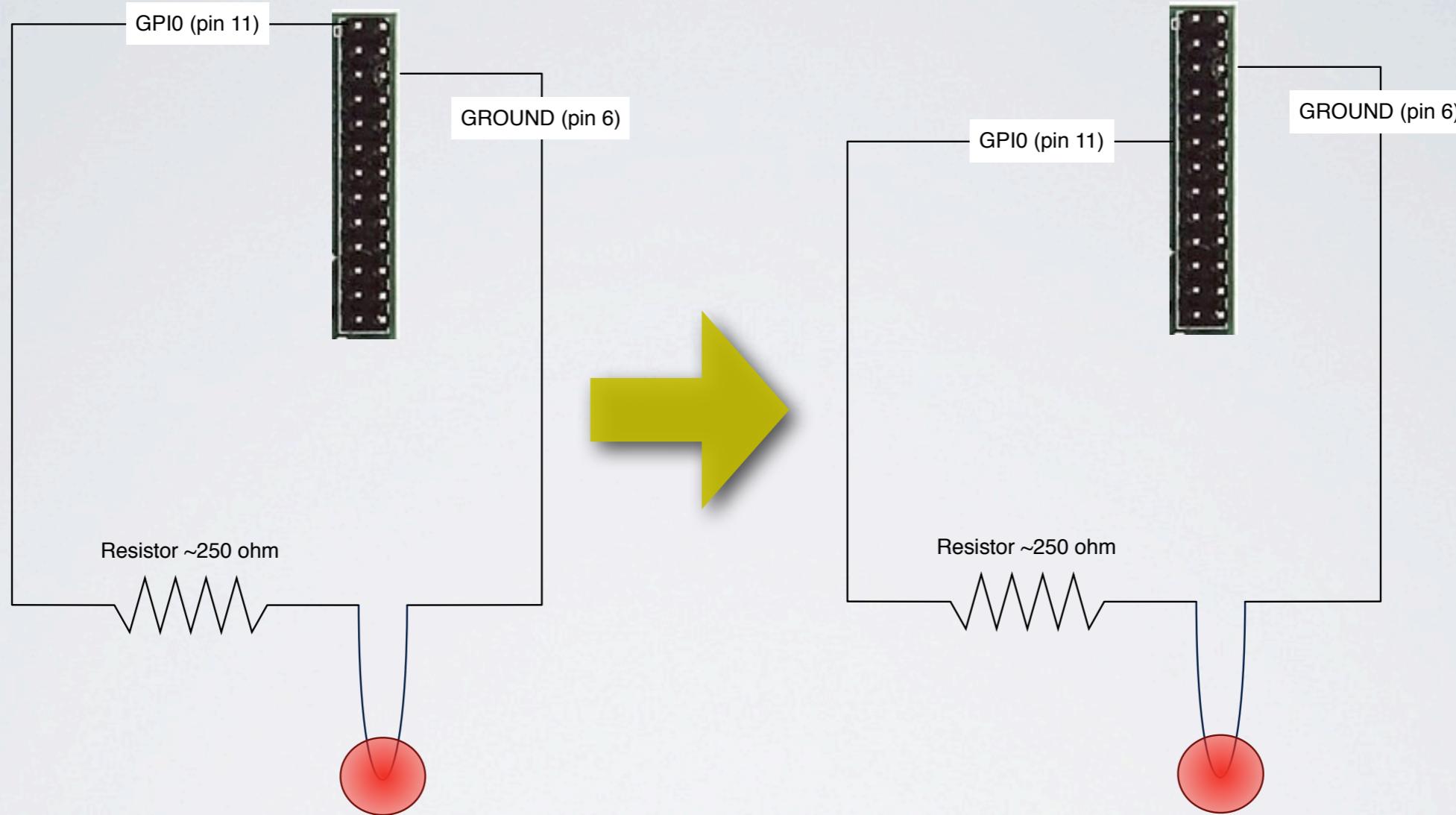
hs

haagen
software AS

Remember, we are in DC land :)



Your first Circuit!



```
GpioController gpioController = GpioFactory.getInstance();
GpioPinDigitalOutput gpio0 = gpioController.provisionDigitalOutputPin(
    RaspiPin.GPIO_00, "gpio0", PinState.LOW);
gpio0.setState(PinState.HIGH);
```

```
GpioController gpioController = GpioFactory.getInstance();
GpioPinDigitalOutput gpio0 = gpioController.provisionDigitalOutputPin(
    RaspiPin.GPIO_00, "gpio0", PinState.LOW);
gpio0.pulse(1000);
```

Jasberry

Jaspberry is a small Java web application that lets you easily switch your GPIO pins on and off

Built with Netty and Ember.js, communication via HTTP and JSON

1: 3.3V	[ON OFF]	2: 5.0V	[ON OFF]
3: SDA0	[ON OFF]	4: DNC	[ON OFF]
5: SCL0	[ON OFF]	6: Ground	[ON OFF]
7: GPIO7	[ON OFF]	8: TxD	[ON OFF]
9: DNC	[ON OFF]	10: RxD	[ON OFF]
11: GPIO0	[ON OFF]	12: GPIO1	[ON OFF]
13: GPIO2	[ON OFF]	14: DNC	[ON OFF]
15: GPIO3	[ON OFF]	16: GPIO4	[ON OFF]
17: DNC	[ON OFF]	18: GPIO5	[ON OFF]
19: MOSI	[ON OFF]	20: DNC	[ON OFF]
21: MISO	[ON OFF]	22: GPIO6	[ON OFF]
23: SCLK	[ON OFF]	24: CE0	[ON OFF]
25: DNC	[ON OFF]	26: CE1	[ON OFF]

Installing Jasberry

cd /home/pi

Install wget: apt-get install wget

Download jasperry.zip and unzip it

```
pi@raspberrypi ~/test $ wget http://stuff.haagen.name/jasperry.zip
```

```
pi@raspberrypi ~ $ unzip jasperry.zip
Archive:  jasperry.zip
  creating: jasperry/
  inflating: jasperry/.DS_Store
  creating: __MACOSX/
  creating: __MACOSX/jasperry/
  inflating: __MACOSX/jasperry/._DS_Store
  inflating: jasperry/config.properties
  inflating: __MACOSX/jasperry/_config.properties
  inflating: jasperry/jasperry-0.0.1-jar-with-dependencies.jar
  creating: jasperry/site/
```

Installing Jasberry

Navigate to the jasperry directory and verify config.properties

```
pi@raspberrypi ~ $ cd jasperry/  
pi@raspberrypi ~/jasperry $ cat config.properties  
jasperry.port=8081  
jasperry.webappDirectory=/home/pi/jasperry/sitepi
```

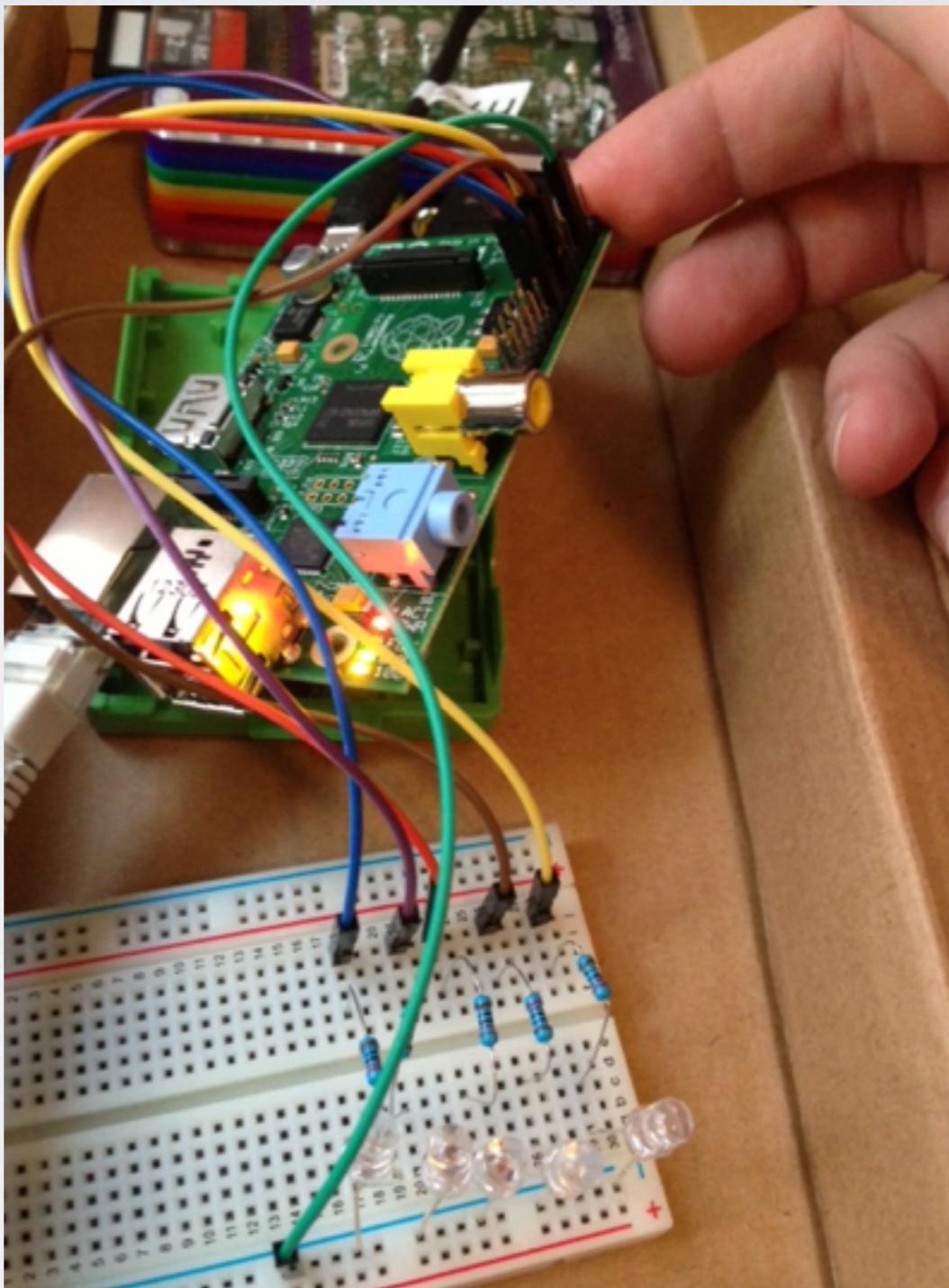
Start jasperry 0.0.1

```
pi@raspberrypi ~/jasperry $ sudo ./run.sh  
log4j:WARN No appenders could be found for logger (no.haagensoftware.netty.server.Main).  
log4j:WARN Please initialize the log4j system properly.  
Aug 08, 2012 11:45:32 PM com.pi4j.util.NativeLibraryLoader load  
INFO: Library [pi4j] loaded successfully using embedded resource file: [jar:file:/home/pi/jas  
-- EXPORTING PIN [0] to mode [output]  
-- EXPORTING PIN [1] to mode [output]  
-- EXPORTING PIN [2] to mode [output]  
-- EXPORTING PIN [3] to mode [output]  
-- EXPORTING PIN [4] to mode [output]  
-- EXPORTING PIN [5] to mode [output]  
-- EXPORTING PIN [6] to mode [output]  
-- EXPORTING PIN [7] to mode [output]
```

Source is on GitHub

<https://github.com/joachimhs/Jaspberry>

Connecting 5 LEDs



Develop a Memory Game: Pinbat

Pinbat is a simple memory-based game.

First similar version developed in the 70s

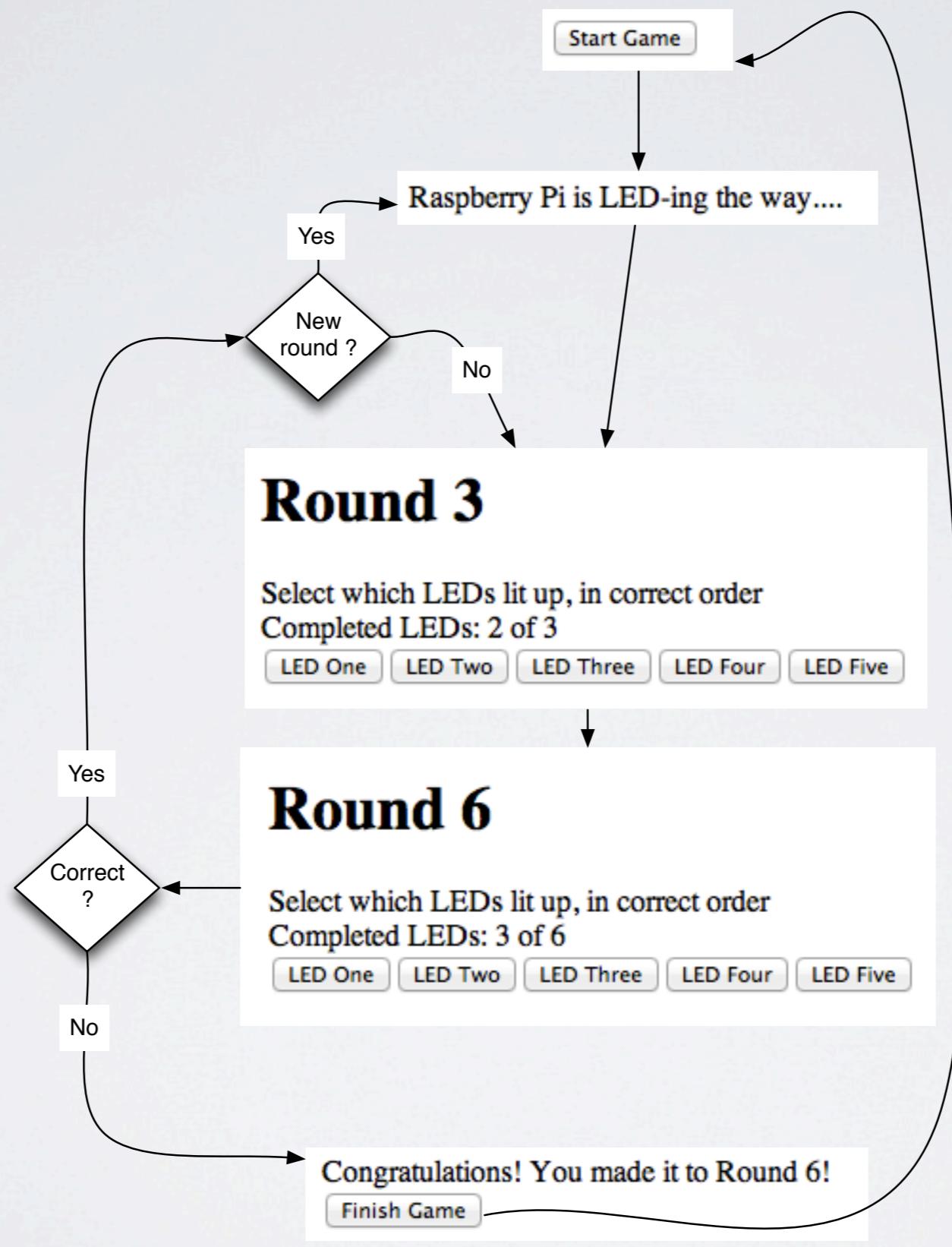
The purpose of the game is to train your memory

The game starts by blinking one of the 5 LEDs

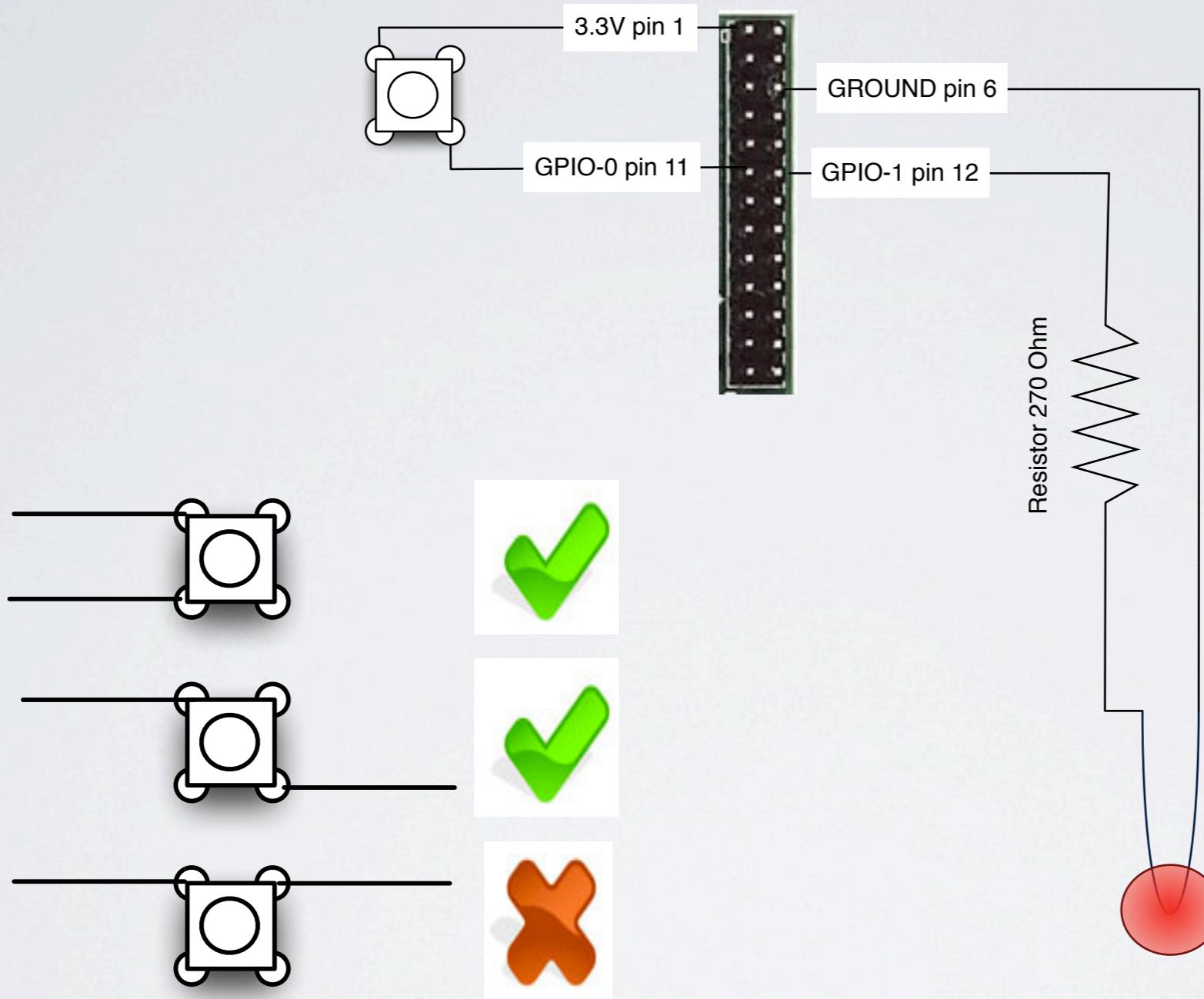
Then, the user will tell the system which LED blinked

If the input is correct, the game moves to the next round where it adds a new LED. The game continues until your memory fails you :)

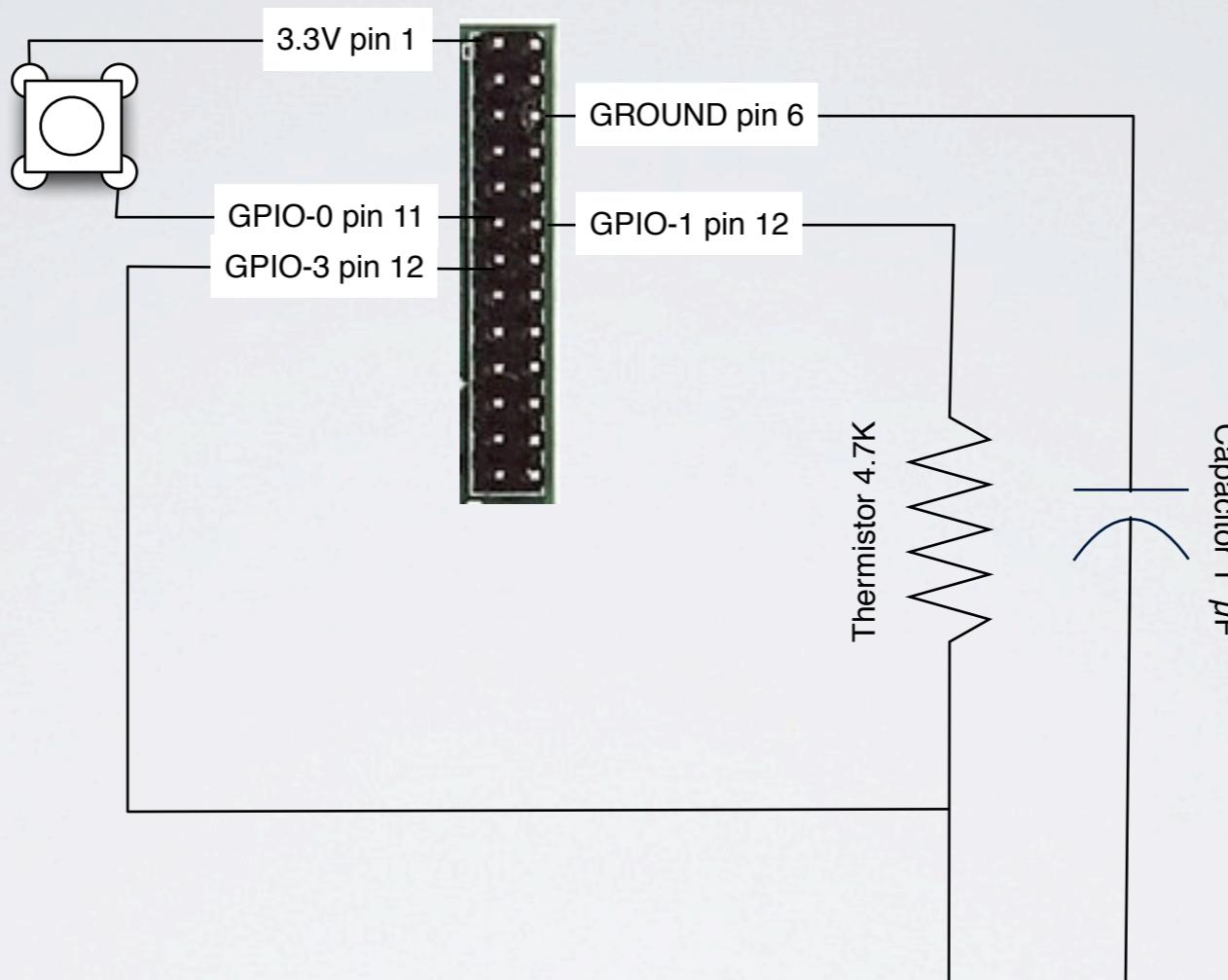
Pinbat



Reading Input



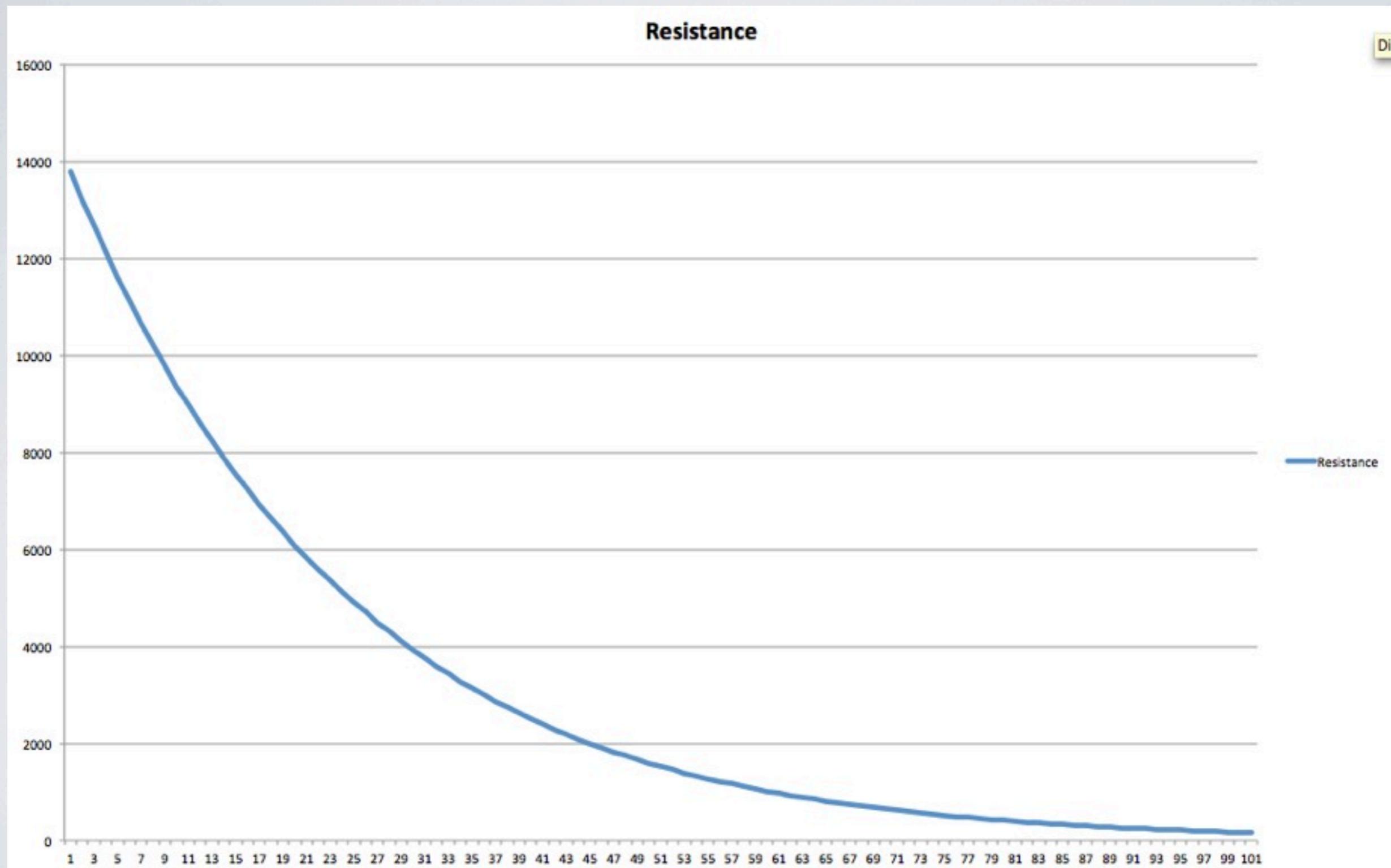
Measure Temperature



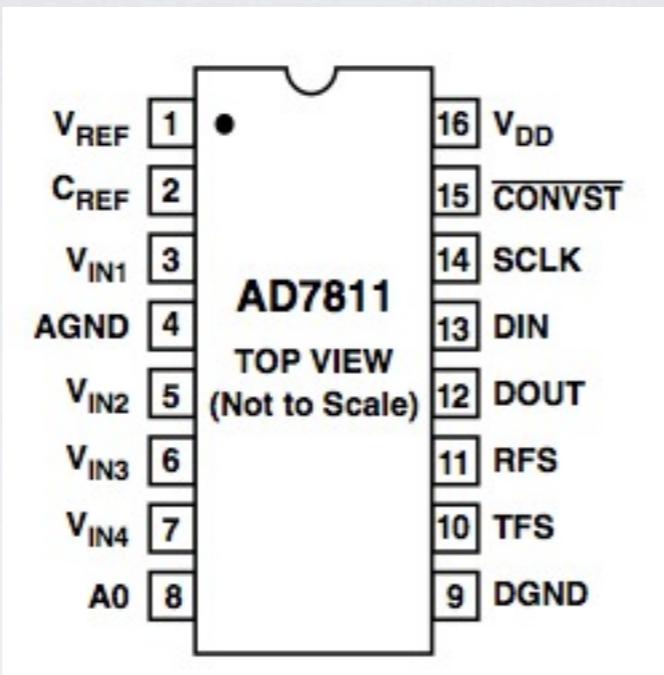
$$t = RC$$

$$(0.0047 \text{ seconds} = 4700\Omega * 0.000001 \text{ F})$$

Measure Temperature



Analog/Digital Converter



<http://scruss.com/blog/2013/02/02/simple-adc-with-the-raspberry-pi/>

Thank you!

Your feedback is greatly appreciated!

Email: joachim@haagen-software.no

Twitter: [@joachimhs](https://twitter.com/joachimhs)

These slides are available at:

http://stuff.haagen.name/raspberrypi_bekk.pdf