**RasWIK - The Raspberry Pi Wireless Inventors Kit**

Category: Radio projects using XRF, ERF, URF & SRF's
Last Updated on Wednesday, 07 August 2013 14:53


## RasWIK introduction THIS DOCUMENT IS STILL UNDER CONSTRUCTION

>> forward to contents page >> forward to the troubleshooting guide

The Raspberry Pi Wireless Inventors Kit (RasWIK) is an **exciting and affordable** way to build your own wireless devices with the Raspberry Pi. The hardware comes preloaded with software and there are applications to run on your Pi. The RasWIK demonstrates that with our leading edge technology anyone (we do mean anyone) can **build wireless sensors and actuators**. You do not need huge experience, a degree or even any tools. We show you how to connect the devices you build to your Pi and even to "the **Internet of Things**" (IoT) with service providers such as Xively.

You can build wireless devices in just minutes. There is no need to write code. But if you want to, all the code is fully **Open Source**, so you can use, modify and even sell your own hardware based on this very kit. Lots of activities are detailed (over 30) some even with a video "how to". Out of the box we have done everything we can to start you off without you writing a single line of code. The kit comes complete with an SD card image that has starter software and a fully configured Pi operating system already installed. The **Arduino** based development board is preloaded with software. The devices you make are interoperable with our off the shelf LLAP wireless devices. This kit is not a toy, it's very serious hardware, presented, we hope in a fun and easy to get in to way.

The project pack consists of a low power radio transceiver for your Pi and a radio enabled electronics development board (the XinoRF) based on the Arduino UnoR3. The XinoRF was awarded 9/10 by Linux Format **see here**. Unlike any other Arduino and radio device, the Ciseco range of interoperable hardware allow you to prototype, develop and even **produce saleable products**. This kit has been aimed at the novice, however the power within your hands to create wireless devices has for many years been only available to embedded engineers. We, at Ciseco have worked for years to break this and give everyday people the chance to join in this most exciting area of technology.

**Features:**

- The starter examples require no soldering
- They come with plug in wires and a solder-less breadboard to make building easy and fast
- A pre installed and configured Raspberry Pi operating system on a 4Gb SD card image (saves you lots of Pi configuration)
- Examples use LLAP to drive the devices (LLAP devices can be mixed with our out the box wireless devices)
- Optional upgrade called **Further Adventures in Wireless**, it adds more sensors and actuators
- Made in the UK

# Contents of the Wireless Inventors Kit for the Raspberry Pi

>> forward to contents page


1 x Ciseco Slice of Radio (our shop product)
1 x Ciseco XinoRF development board (our shop product)
1 x 4Gb SD card with Pi OS and sample software
1 x USB cable
1 x Small breadboard
5 x Red LED
5 x Yellow LED
5 x Green LED
1 x Blue LED
10 x 10K Resistor
20 x 470R Resistor
1 x Light Dependent Resistor (light sensor)
1 x Thermistor (temperature sensor)
1 x Piezo sounder
3 x Push buttons
25 x Jump wire (assorted colours)
1 x Variable resistor (potentiometer)
1 x Transistor
1 x Diode
Length of solid core hookup wire

## Following these projects without having the full kit

If you want to follow these examples without buying the kit (perhaps you own some of our other radio hardware already) there are details of the steps you will need ( **link** ).

## Table of contents

# Activity 01 - Lets test the hardware by flashing a red light on the XinoRF

[<< back to contents page]

### What will we learn?

How to install the basic hardware, run the supplied software and check there is a wireless connection by turning on and off the on-board LED on the XinoRF (an LED is a type of light)

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

**Let's start!**

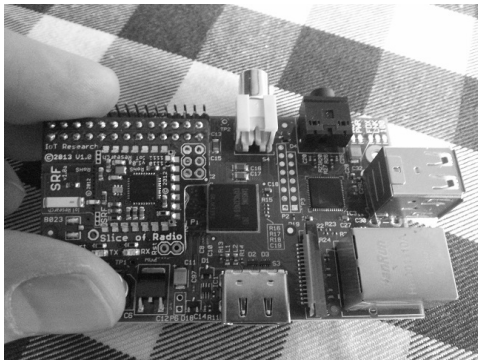
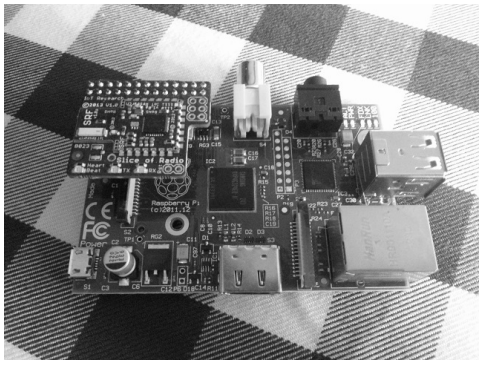
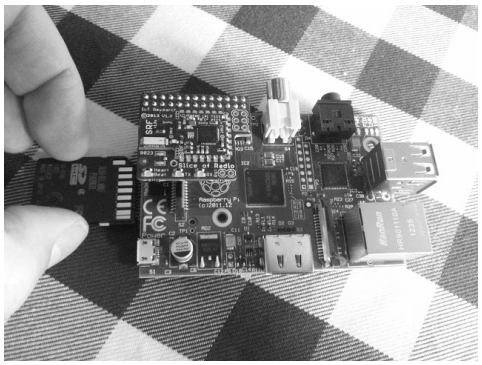
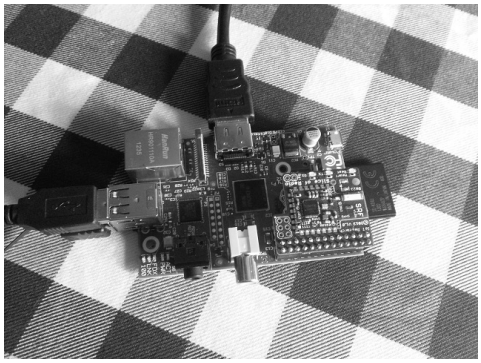
Image 1 


Image 2 

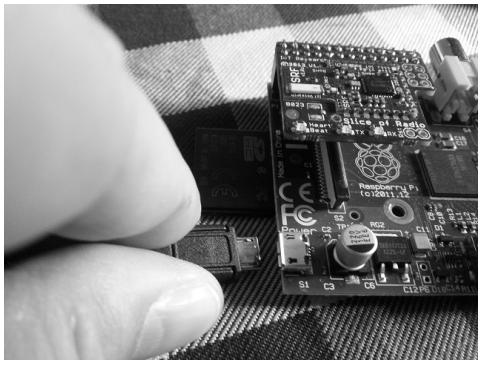

With your Raspberry Pi switched off, pick up the Slice of Radio (image 1) and gentle push it onto the 26 pin connector. Once pressed fully down your Pi should look like (image 2).

Image 3 


Plug the SD card from the kit into your Pi. The Pi's SD card connector can be easily broken so take care and don't force the card in.

Image 4 

Connect the keyboard, mouse and monitor as you would normally to your Pi. If you would like more detailed instructions we suggest the main Pi documentation **http://www.raspberrypi.org/quick-start-guide**

Image 5 

Power up the Pi as you would normally.

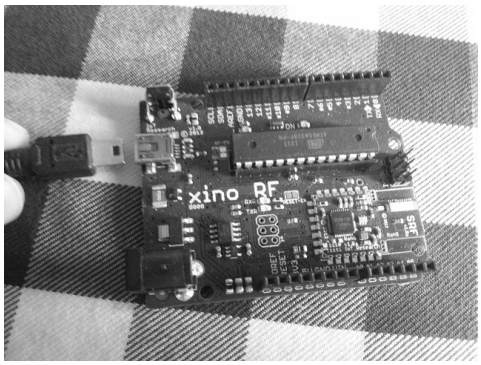
Image 6 

Image 7 

Plug the XinoRF into the USB cable (image 6) and the USB cable into a device to power it (image 7). The device supplying the power might be your Pi, a powered USB hub, a PC or even a USB phone charger. The source you chose will need to be able to supply at least 100 milli amps. A normal USB socket can supply 500 milli amps, the Pi however is far more limited. If your Pi will not boot up, it's often because the USB socket is being asked to supply more than the Pi can supply. For this reason alone, if you can power from an external source it's a good thing to do.

Worth noting is: Don't put the XinoRF too close to the Slice of Radio, we suggest half a meter as a minimum. If they are too close the radio signal is so strong the receiver cannot work out what data is being transmitted.

### That is the hardware all sorted, let's try the application software

You will need to login to your Pi using the standard user name (pi) and password (raspberry). Now type `startx` to launch the graphical interface.

If this is the first time you use the SD card and you use a WiFi dongle as internet connection, now is the right time to ensure the Raspberry Pi is connected to your WiFi network using the WiFi Config app on the desktop. If you use an ethernet cable, your internet connection should just work.

On the desktop you should see a folder called WIK FILES. Open this folder and double click on the file RunMe.py. You may get a warning that says "Do you want to run "RunMe.py", or display its contents?". Select "Execute" and you should see WIK Launcher start. When WIK Launcher starts, it checks to see if it has the latest files and if not it will ask you if you want to download the latest version. It is a good idea to say "yes" when asked.

You should now have WIK Launcher running as shown in the image below.

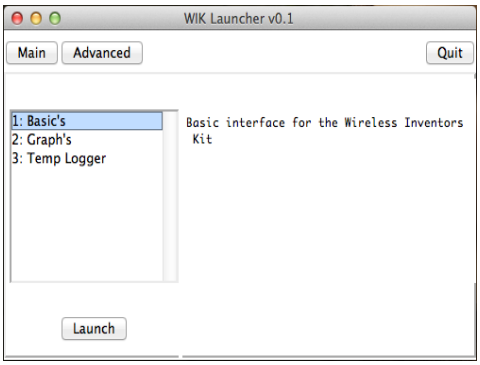


Image 8

Choose "Basic's" from the list and click launch.

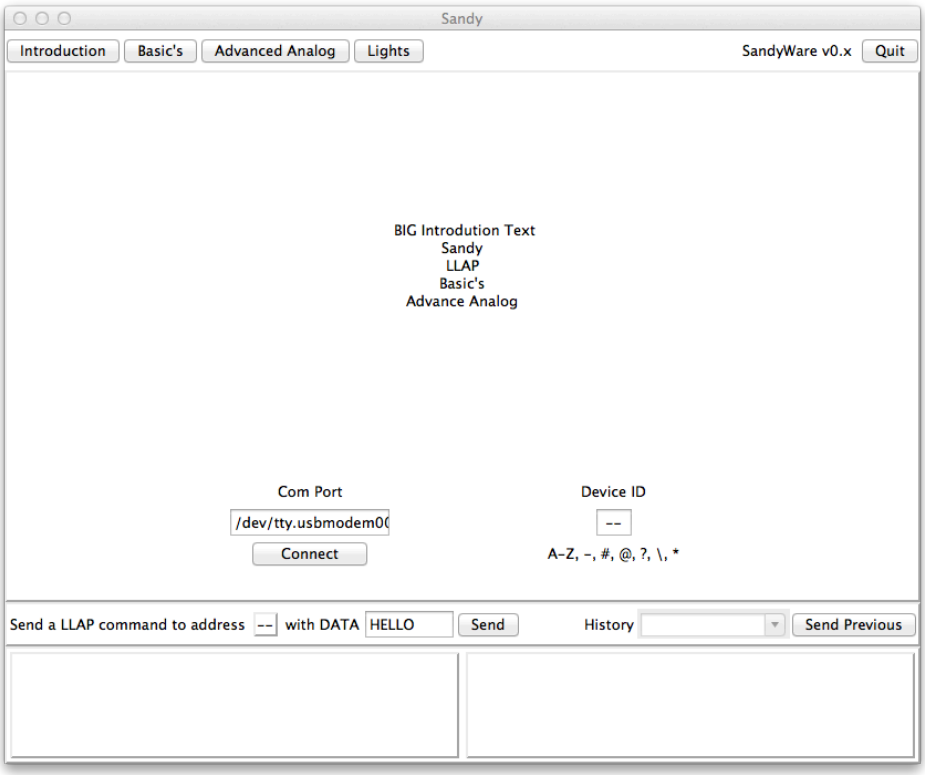


Image 9

Once the WIK Basics Interface has opened you will be presented with a box to enter your Com port. On the Raspberry Pi, the radio is configured on port "ttyAMA0". Make sure this is what is shown in the box and click "Connect".

Next go to the "Basics" tab. You should see a window as in the image below.

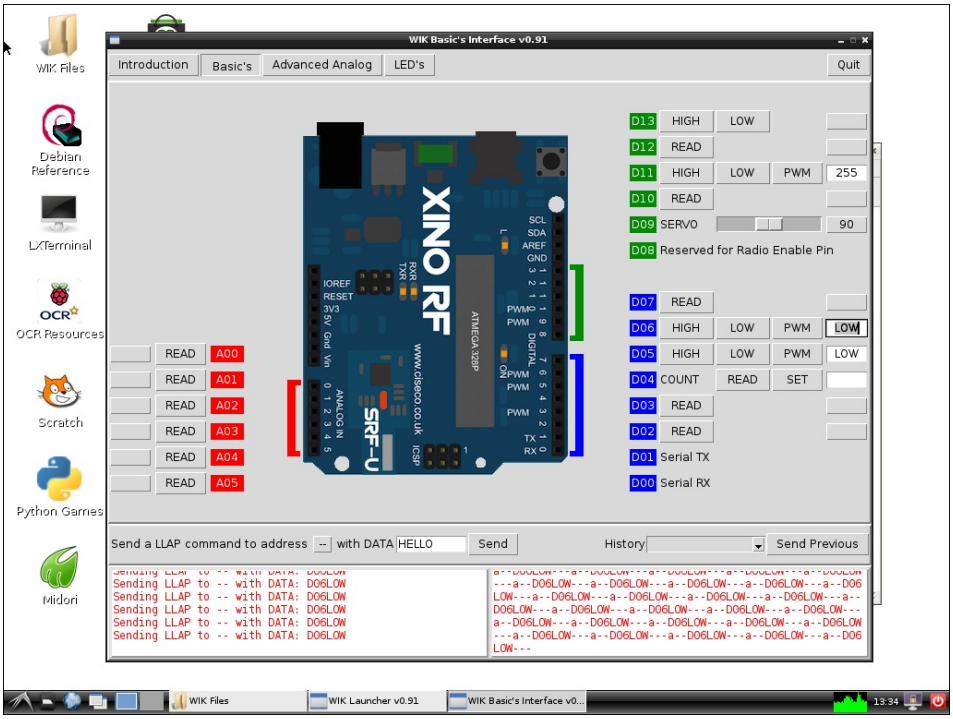


Image 10

The image you see is of the XinoRF that shipped with your kit. The red, blue and green boxes correspond with input and output channels on the XinoRF. Red are analog inputs, green and blue are digital inputs and outputs. We will explain all these in more detail later, so don't worry about this now.

Now let's send something wireless, after all that's what we want to do! If you click the buttons D13 (they are on the top right) "HIGH" or D13 "LOW", the LED on the XinoRF will light up or switch off respectively.

When you press the button a message is sent. You can see the actual data in the two lower windows. Don't worry yet what it all means we'll come to that later. But if you can understand already what is going on from the messages, we have achieved one of our goals: sending messages to devices is as easy as sending a text or a tweet!

If you don't see the LED on the XinoRF change, then press its reset button on the XinoRF and try again.

**That's it, easy wasn't it?**

## Activity 02 - Understanding basic digital inputs on the XinoRF by using a button

[<< back to contents page]

**What will we learn?**

Digital inputs are the pins on the XinoRF which read if a voltage is high (5 volts) or low (0 volts). Many things in life can be represented by just these two states, i.e. is the light on (high) or off (low). With digital inputs we can read many different sensors. We can demonstrate this on or off behaviour very easily using just a single standard PCB button.

**How long will this take?**

No more than 5 minutes

**How difficult is this project?**

Very easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x PCB button

1 x Breadboard

2 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01.

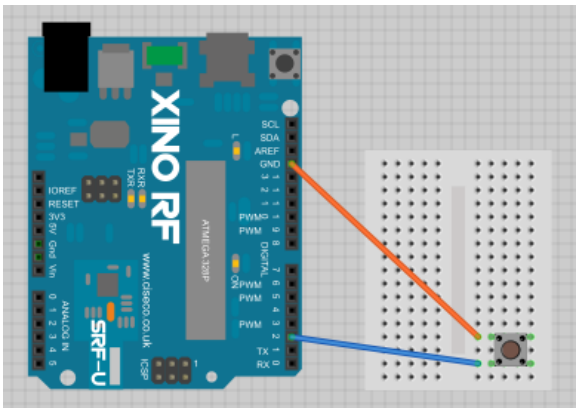



Image 11

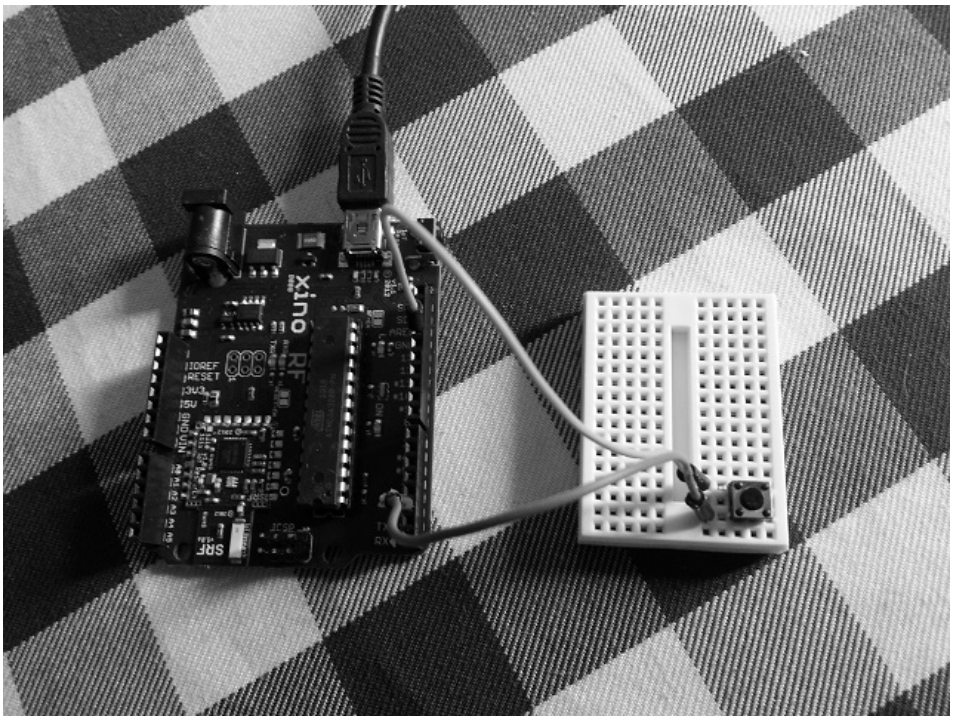


Image 12

Very gently push the switch into the breadboard as shown in the photos. Take care to get the orientation right, the legs should stick out left and right.

From the two pins on the left side of the switch, push in two different coloured jumper wires.

Push the top wire into GND

Push the other into pin D03

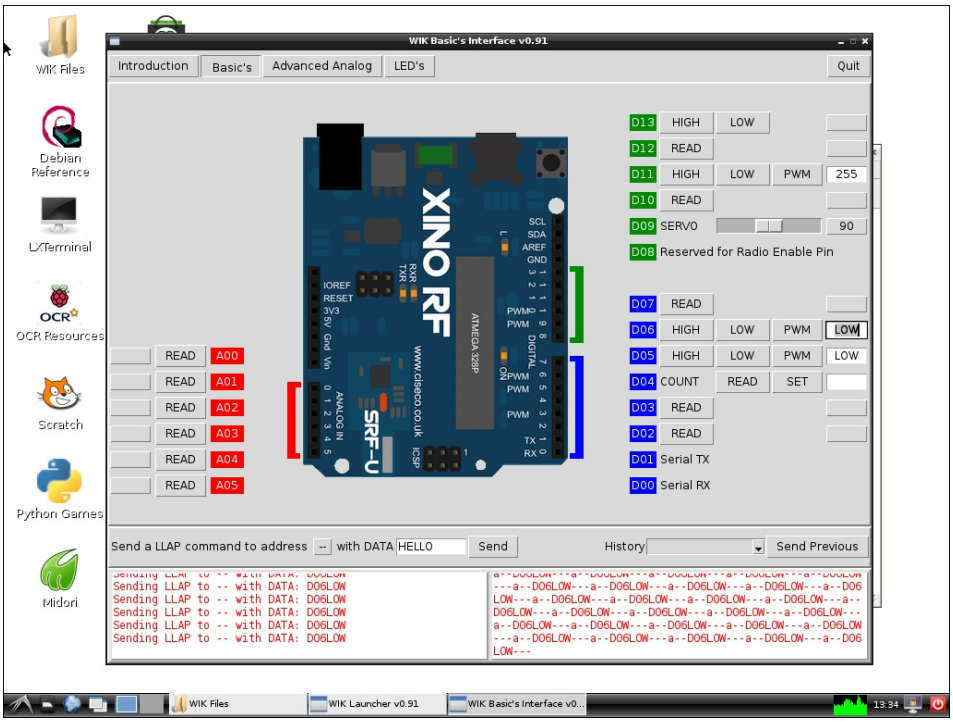



Image 13

With WIK Basics running as pictured above. Press and hold the button on the breadboard. you will see D03 (blue, bottom right) will show LOW in the grey box, if you release the button it will change to HIGH. Try it a few times. The lower two large text boxes show the actual data being sent over the air waves. For now don't worry what it means, just press the button a few times to see the data being received (BLUE).

**That's it....**

## Activity 03 - Understanding basic digital outputs on the XinoRF by using an LED

[<< back to contents page]

### What will we learn?

Digital outputs are pins on the XinoRF you can command to switch between HIGH (5 volts) or LOW (0 volts, also known as ground or GND for short). The on-board ATmega328 chip can power a small amount of current through each pin (40 milli amps maximum). This is good for things like Light Emitting Diodes (LEDs) as we will see. Driving motors, light bulbs and many other things needs additional hardware; we'll come to this later.

### How long will this take?

No more than 5 minutes

**How difficult is this project?**

Very easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x LED (the choice of colour is up to you)

1 x 470 ohm resistor

1 x Breadboard

2 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01. That is, have WIK Basics running so you can interact with the XinoRF.

At the XinoRF end, we need to build a simple circuit, as in the pictures below.

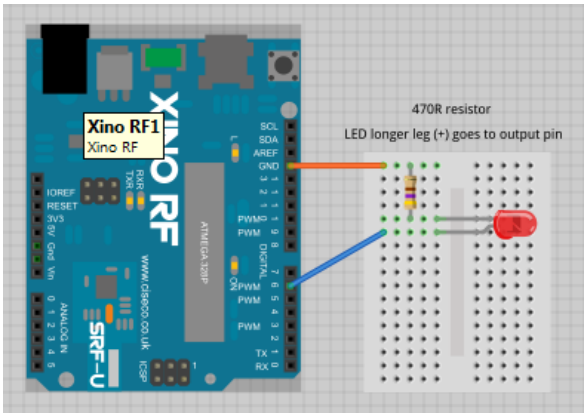



Image 14

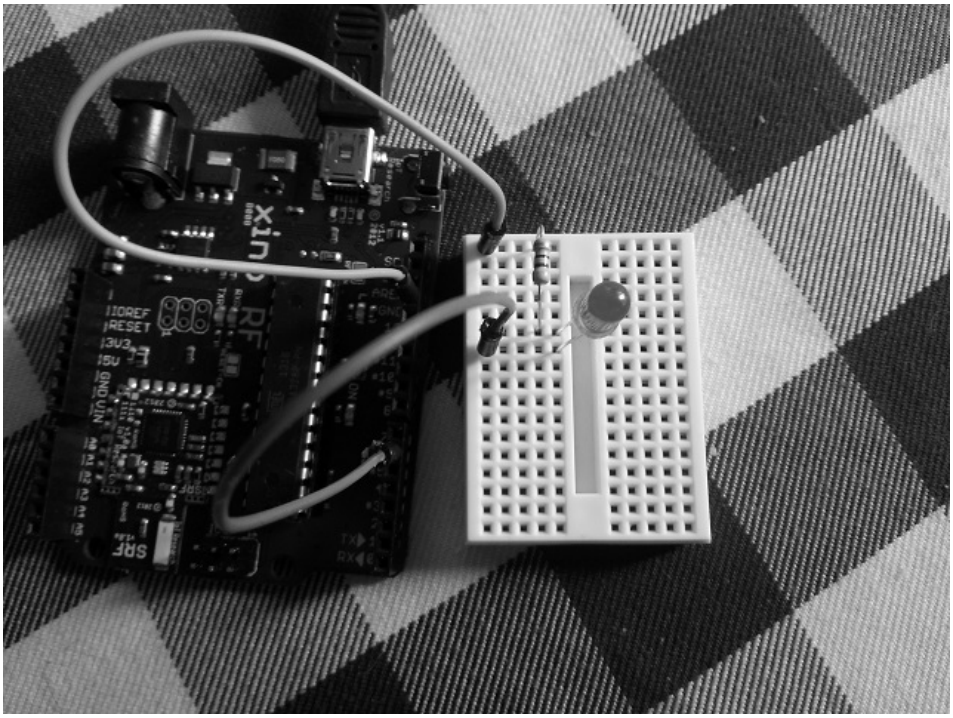


Image 15

Gently push the LED into the breadboard as shown in the photos. Take care to get the orientation right, the shorter leg is minus, the longer one is positive and should be connected to the output pin. Gently bend the 470R resistor legs on and push into the board as shown.

From the longer of the pins of the LED push in a coloured wire, connect this to pin D06. Push a different coloured wire into the board where the resistor in and the into GND on the XinoRF

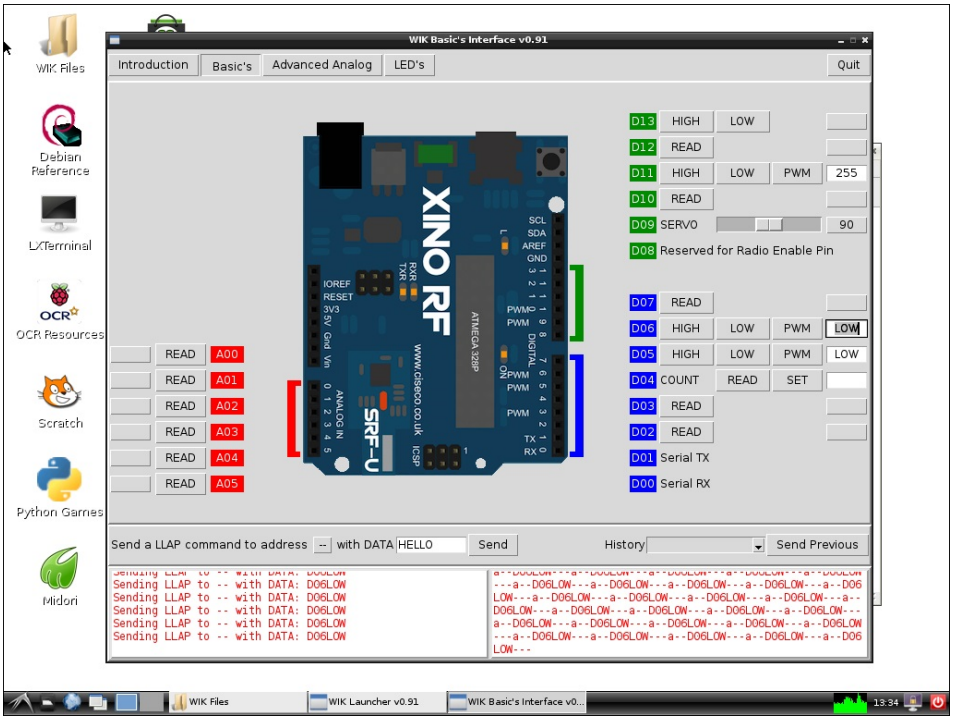



Image 16

On the WIK interface, press the HIGH or LOW button on D6 and observe the LED switching on when you select HIGH, and switching off when you select LOW.

If your project fails to light then double check the LED is the right way round. LED's will only work one way round. If the LED is very dim then perhaps check you used a 470R resistor and not a 10K by accident.

# Activity 04 - Lets build a simple traffic light

<< back to contents page

### What will we learn?

That wireless commands are sent and received faster than you can see and this is much faster than familiar communications means like SMS's, tweets, email or browsing webpages. When you press an on screen button it sends 3 commands, 2 to turn off the other two LED's and one to turn on the one you clicked. Unless you are quite unique (yeah me too) then you won't see this happening.
What does this mean? That the chip on the XinoRF runs code in a linear fashion very fast, this is quite different to a PC which can multitask.

### How long will this take?

No more than 10 minutes

### How difficult is this project?

Easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

3 x LED (one red, one yellow and one green)

3 x 470 ohm resistor

1 x Breadboard

7 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01. That is, have WIK Basics running so you can interact with the XinoRF.

At the XinoRF end, we need to build a simple circuit, as in the pictures below.

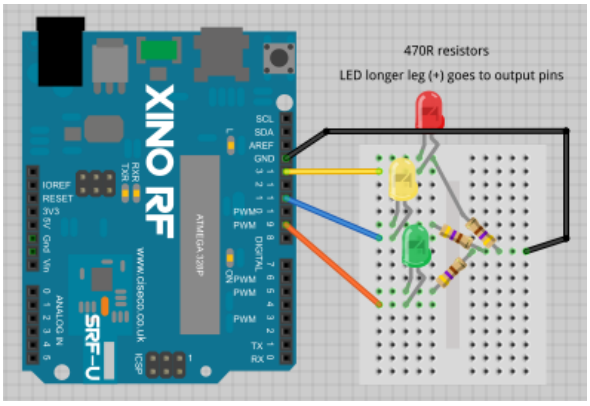



Image 17

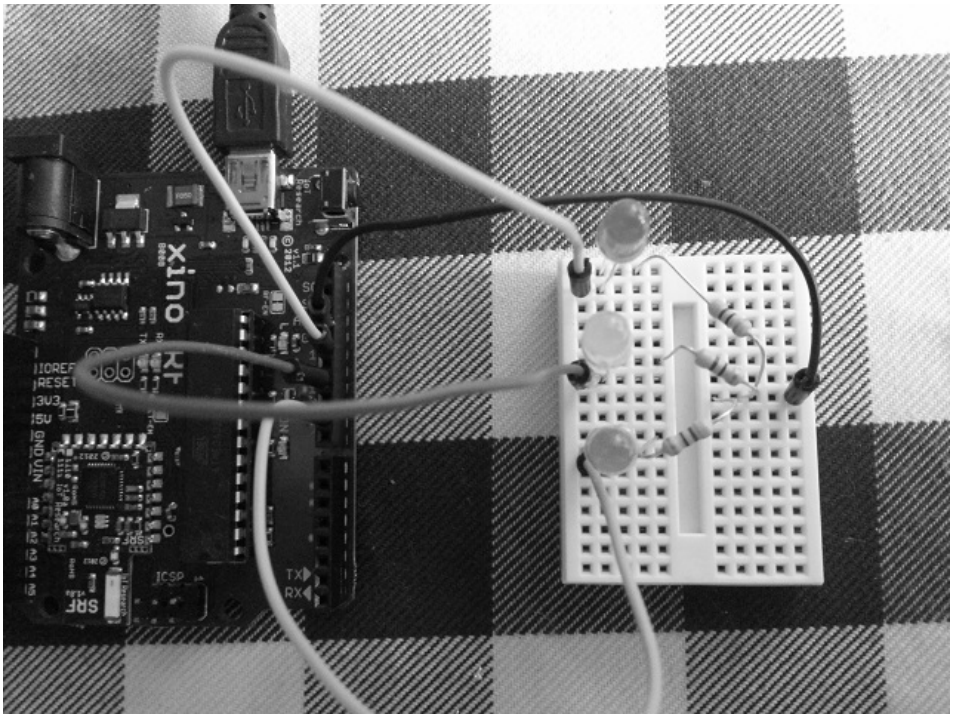


Image 18

Gently push the LED's into the breadboard as shown in the photo. Take care to get the orientation right, the shorter leg is minus, the longer positive. Gently bend the resistor legs and push them into the board as shown.

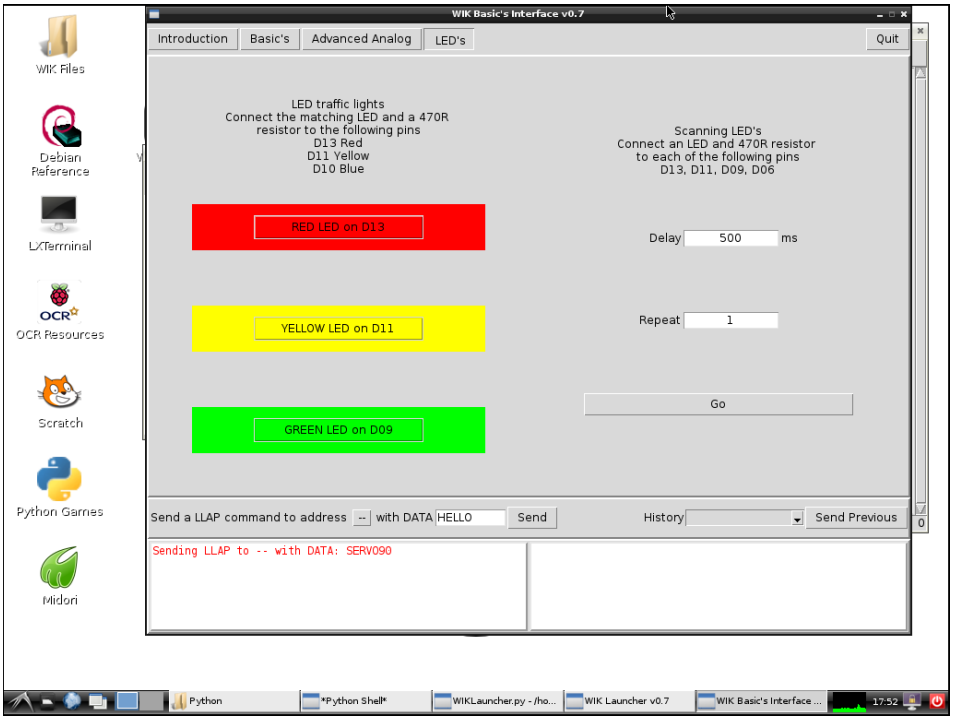



Image 19

Click the LED's tab in WIK basics interface to reveal a page like this. By pressing the buttons for RED, YELLOW and GREEN the application will send 3 commands (you can see them in the bottom text windows). As an example, if you press RED it will send a command to turn off both yellow and green before sending a command to turn the red LED on.

## Activity 05 - A 4 LED sweeping scanner

<< back to contents page

### What will we learn?

That traffic lights aren't the only easy thing to do with LED's. You probably don't remember a TV show called Knight Rider. In it there was a car with light chaser in the bonnet (hood if you are not from the UK) it swept from side to side. This activity recreates a little of those memories for me.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

4 x LED (all red)

4 x 470 ohm resistor

1 x Breadboard

8 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

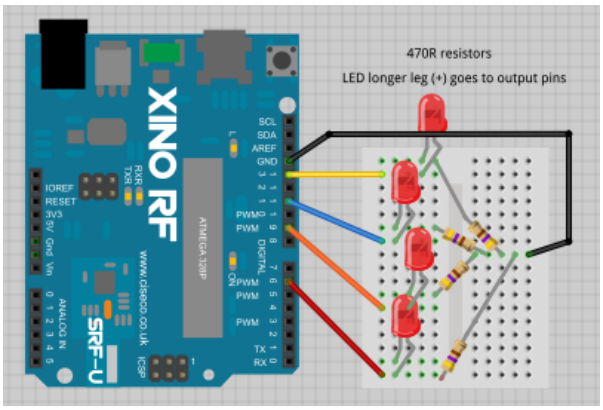
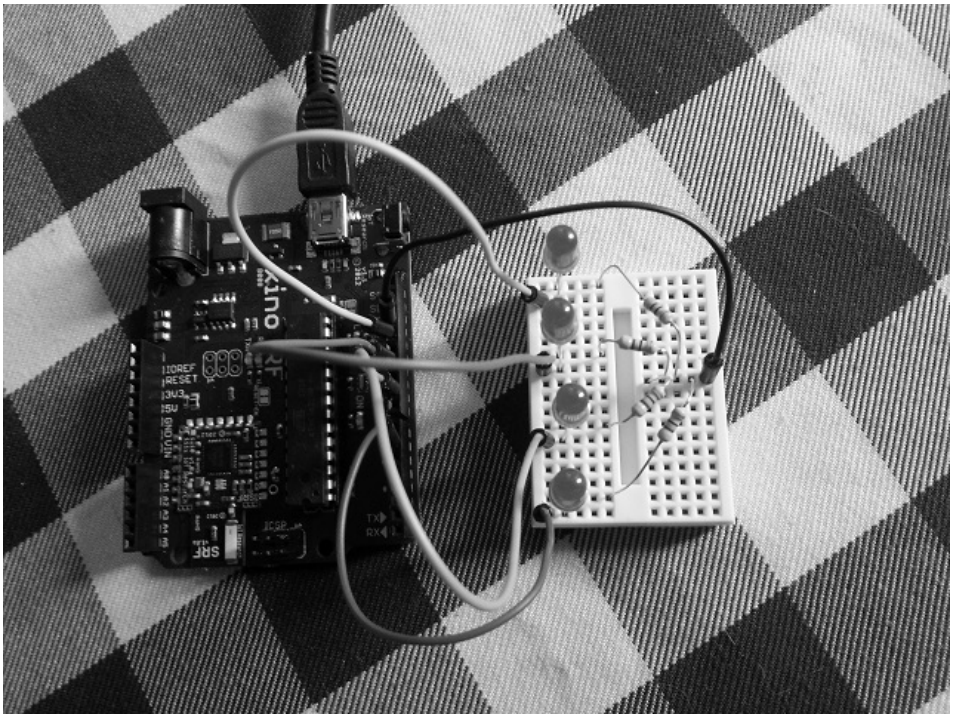
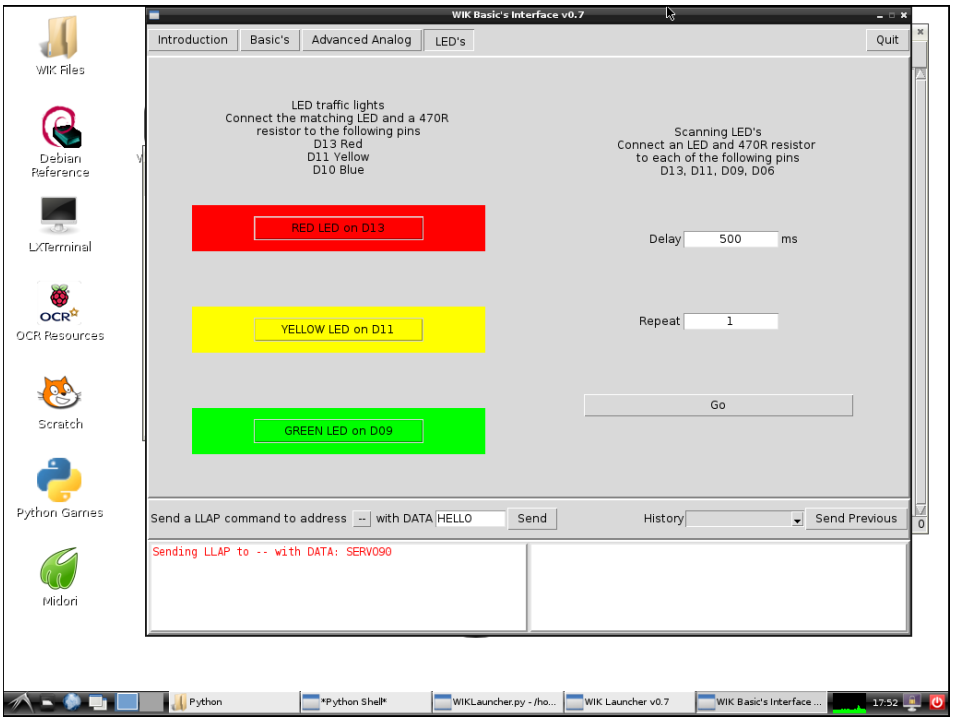



Image 20

Image 21 

Gently push the LED's into the breadboard as shown in the photo. Take care to get the orientation right, the shorter leg is minus, the longer positive. The positive side is connected to the XinoRF output pin. Gently bend the resistor legs and push them into the board as shown also.

Image 22 


In the WIK basics interface, click the LED's tab to show the page shown (image 22). On the right hand side we can set the speed between transmissions and how many sweeps (repeats) we'd like to happen. The Pi is not especially fast so treat the delay time with an element of flexibility.

## Activity 06 - Driving LED's between pins

<< back to contents page

### What will we learn?

An LED is quite different to a bulb, even though both emit light. One of the major things to understand is that an LED will only light when the voltage passes in the right direction. In this activity we will put two LED's back to back. By changing the polarity of the current flowing we can light either LED from just 2 pins on the XinoRF.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

2 x LED (both red)

2 x 470 ohm resistor

1 x Breadboard

2 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

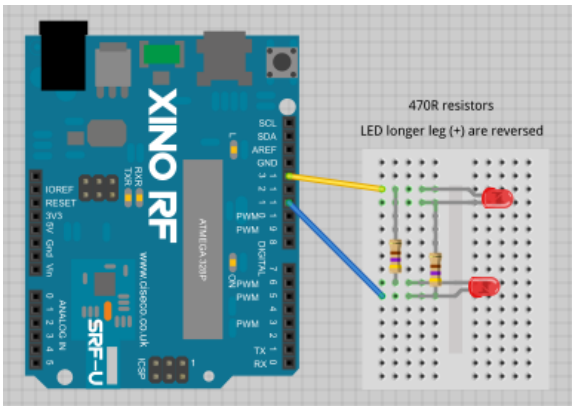



Image 24

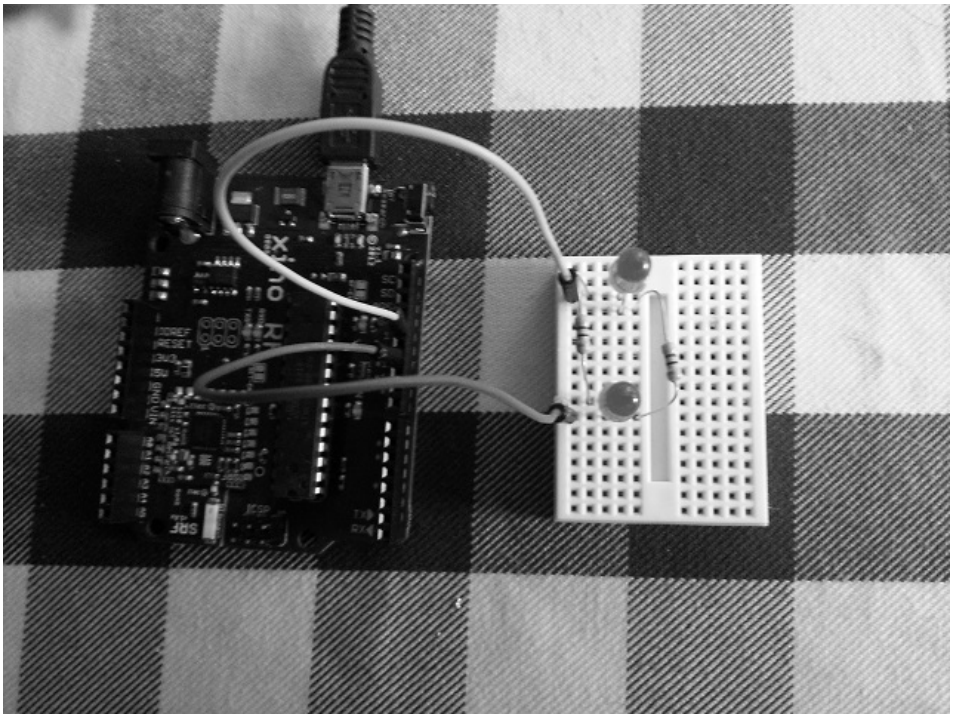


Image 25

Gently push the LED's into the breadboard as shown in the photo. Take care to get the orientation right, the shorter leg is minus, the longer positive. Gently bend the resistor legs and push them into the board as shown also. Remember to have one LED's connections face one way and the other the opposite way.

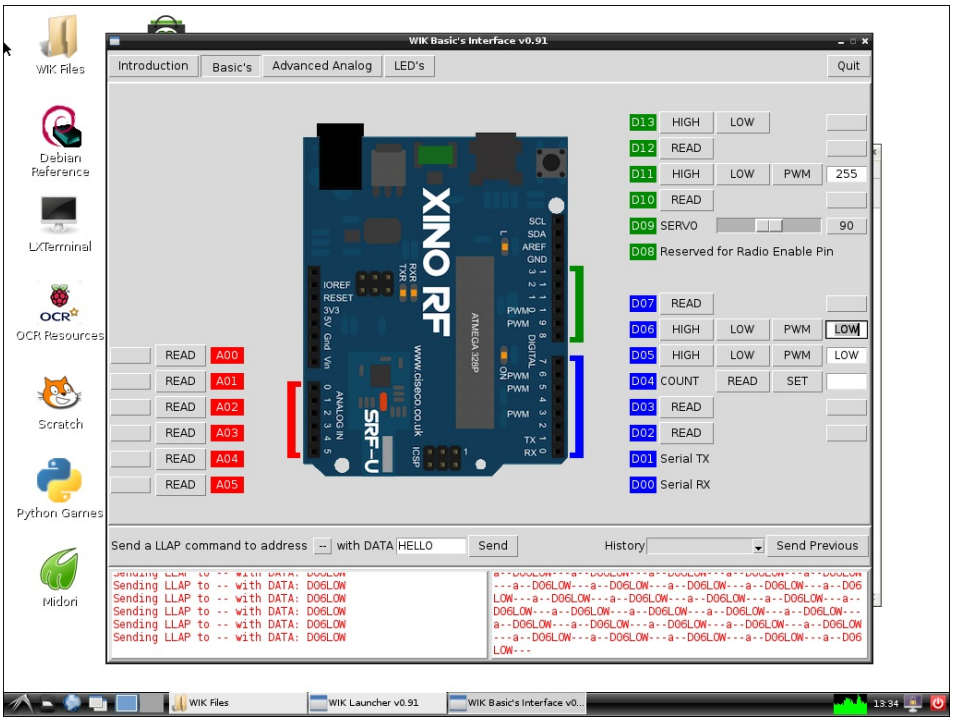



Image 26

By changing the state of D11 and D13 we can make one of the LED's light at a time. What you need to do is set D13 HIGH and D11 LOW and then reverse this so that D13 is LOW and D11 is HIGH.

## Activity 07 - Using two resistors to divide a voltage


<< back to contents page

**What will we learn?**

That we can divide a voltage by using a pair of resistors. How we do this is by connecting two resistors across the voltage we want to measure and then we connect the XinoRF analog input between the middle of the two resistors. The voltage in the middle is proportional to the values of the resistors. In this example we will use two resistors of the same value. This will divide the voltage by half.

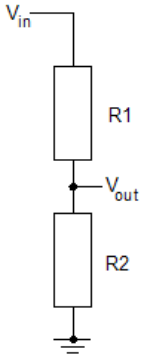



Image 27

Why would we want to do this?

The XinoRF can measure upto 5 volts and no more. Any more would damage the "chip". By using this example we could measure upto 10 volts as we can simply measure the middle voltage and multiple by a factor of two. If you want to calculate different values this is a useful site

**How long will this take?**

No more than 5 minutes

**How difficult is this project?**

Very easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

2 x 10K resistor

1 x Breadboard

2 x Jumper wire

**Lets start!**

Make sure your Pi is installed and running as detailed in activity 01.

Image 28

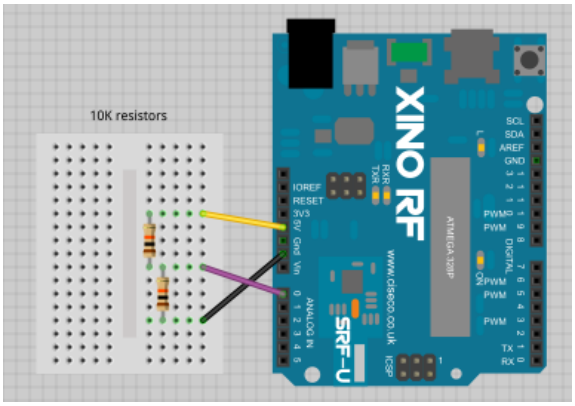
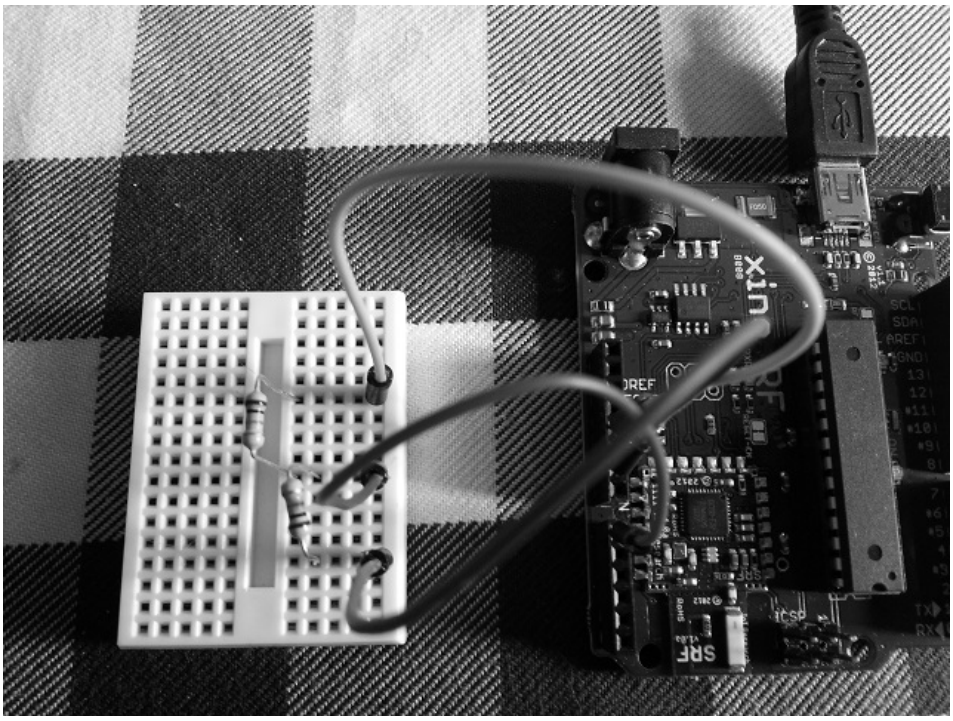
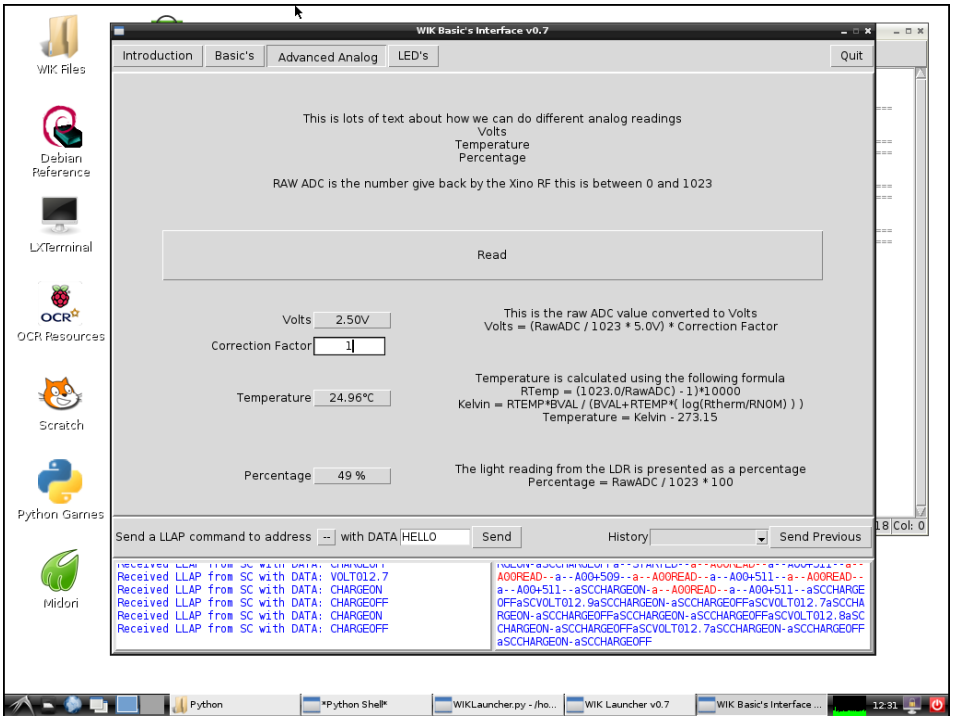



Image 29



Gently bend the resistor legs and push them into the board as shown. Connect a jumper wire from the top resistor leg to 5v, the middle connection to A0 and the bottom resistor leg to GND .

Image 30



In the Advanced analog tabbed page, if you click the "Read" button then we can see the voltage is read at 2.50 volts instead (half the original).

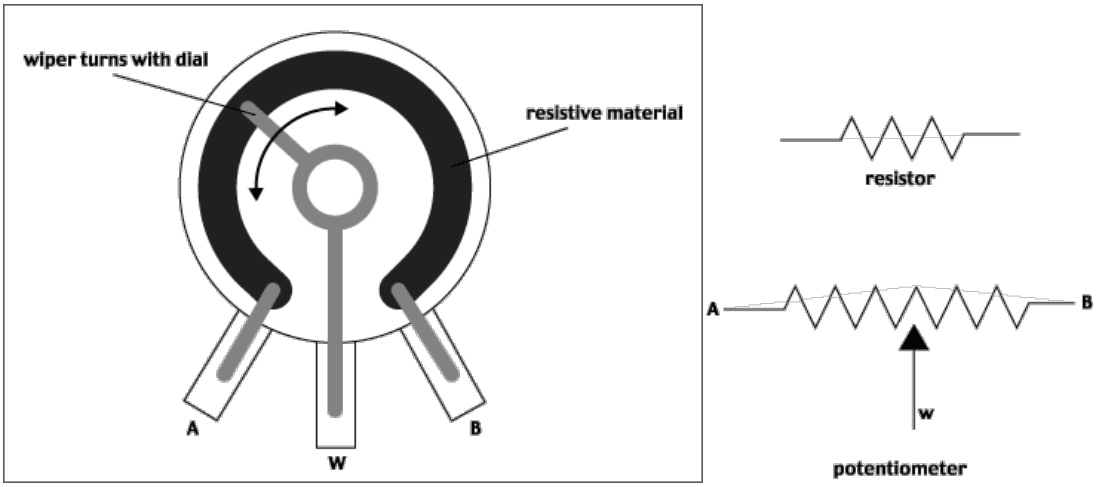
Don't worry if you are reading is a little different, the tolerance of the resistors is 5% so will vary slightly.

## Activity 08 - Using a variable resistor to change voltage

<< back to contents page

### What will we learn?

That a voltage can be divided by using a variable resistor (its proper name is a potentiometer or pot for short). If you look to the last activity we divided a voltage by using two resistors, this is essentially what a pot does but (variably) and why it has 3 legs. The centre pin (called the wiper) sweeps up and down a continuous resistor. By doing this it essentially creates two resistors intersected by the wiper. When you move the wiper the proportion of each side and its resistance changes accordingly.




In your kit there is a 10K pot. If you position the wiper half way, there will be 5K either side so it will divide the voltage in half. If you moved to 1/3 of the way you would get 3.3K and 6.6K. To try this out with your kit, we connect the two outer legs to 5v and GND, the centre pin (wiper) to an input. By turning the pot we vary the voltage and this can be read.

For more details on what a pot is see **wikipediea**.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x 10K potentiometer

1 x Breadboard

3 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01.

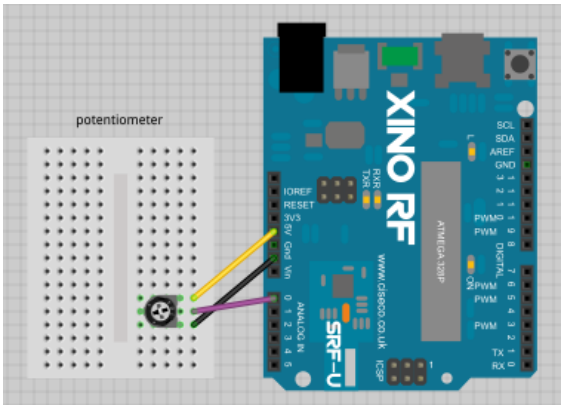



Image 31

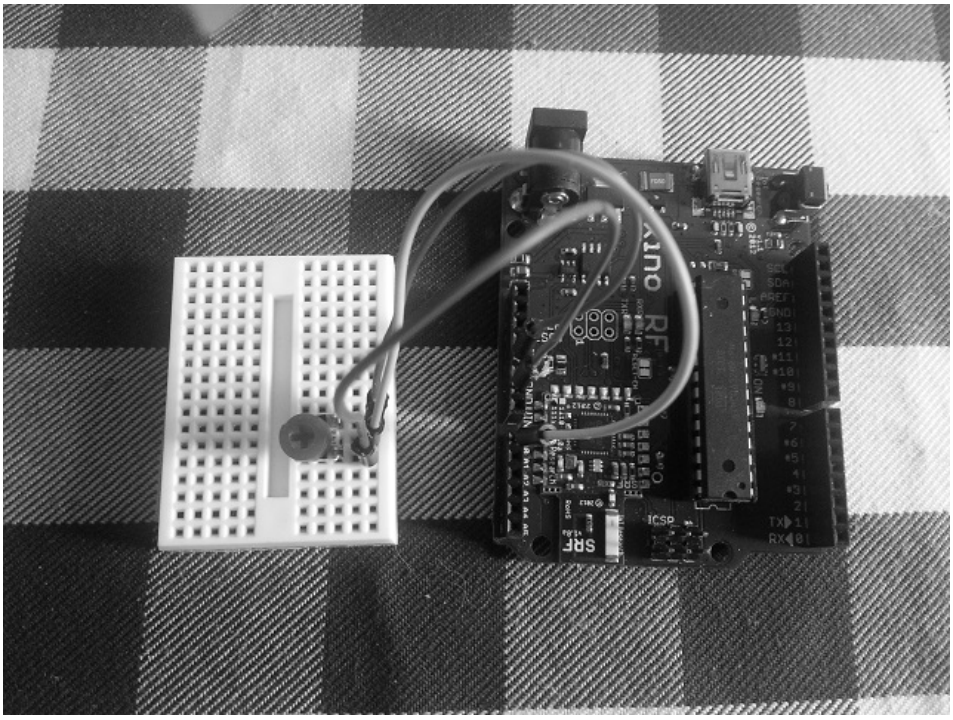


Image 32

Gently push the pot into your breadboard and connect the 3 jumpers as shown. Turn the wiper up and down in various positions, then read the voltage by clicking READ next to A00 in the Basic's tab.

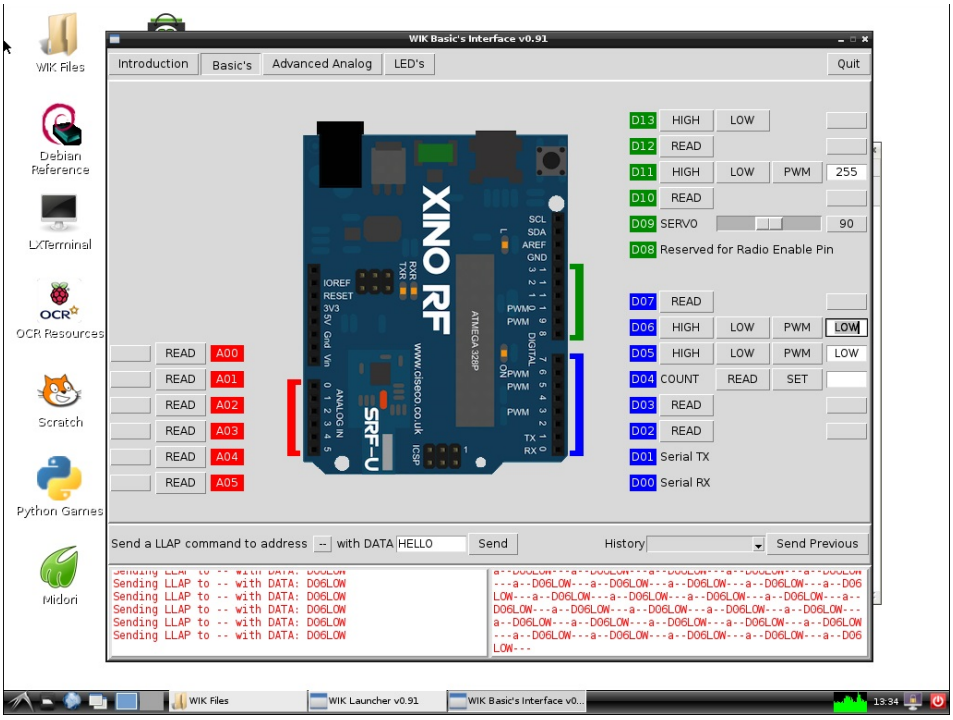



Image 33

As you move the wiper and read the voltage you can see it change.

## Activity 09 - Let's measure the light in the room

[<< back to contents page]

### What will we learn?

That by reading a variable voltage we can measure the light levels in our room. Using the same techniques as in activity 7 we divide a voltage by two resistors. One is a normal resistor, the second a light dependant resistor (or LDR for short). The LDR's resistance varies when light hits it, this causes the voltage to change, it's this voltage we read.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x 10K resistor

1 x LDR (light dependant resistor)

1 x Breadboard

3 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01.

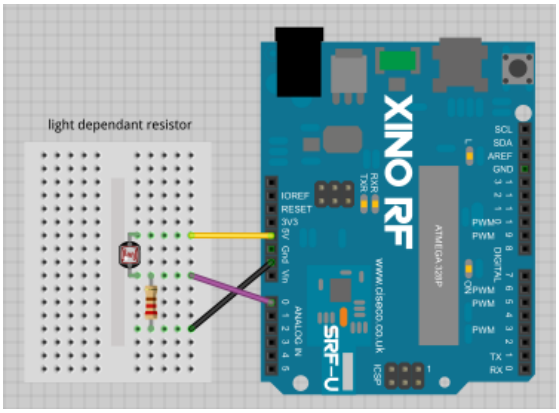


Image 34

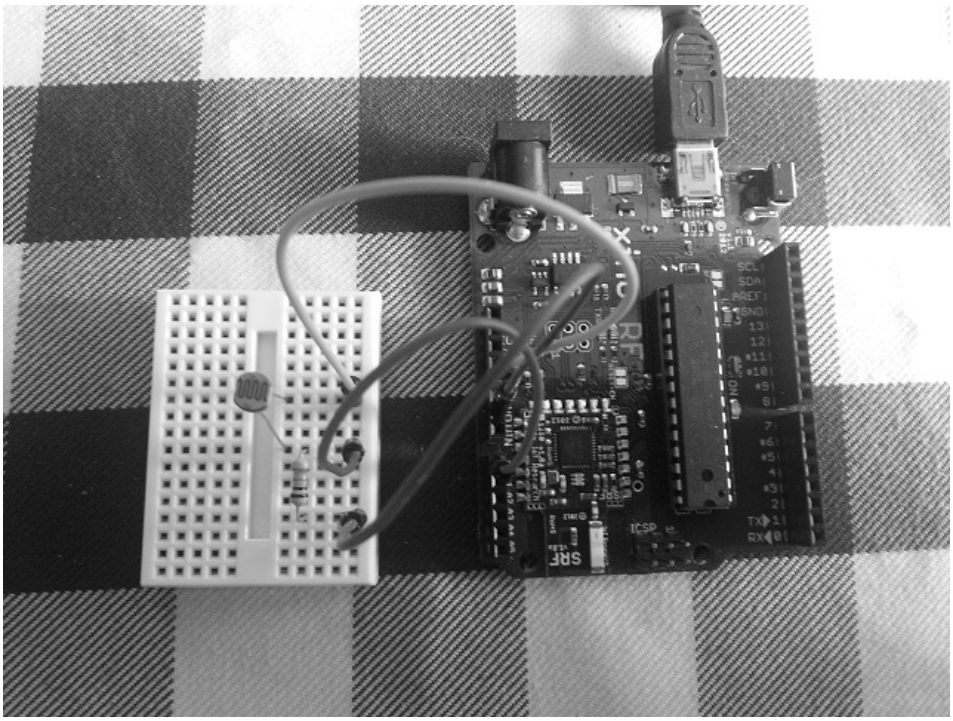


Image 35

Gently push the legs of the LDR and the resistor into the breadboard as shown.

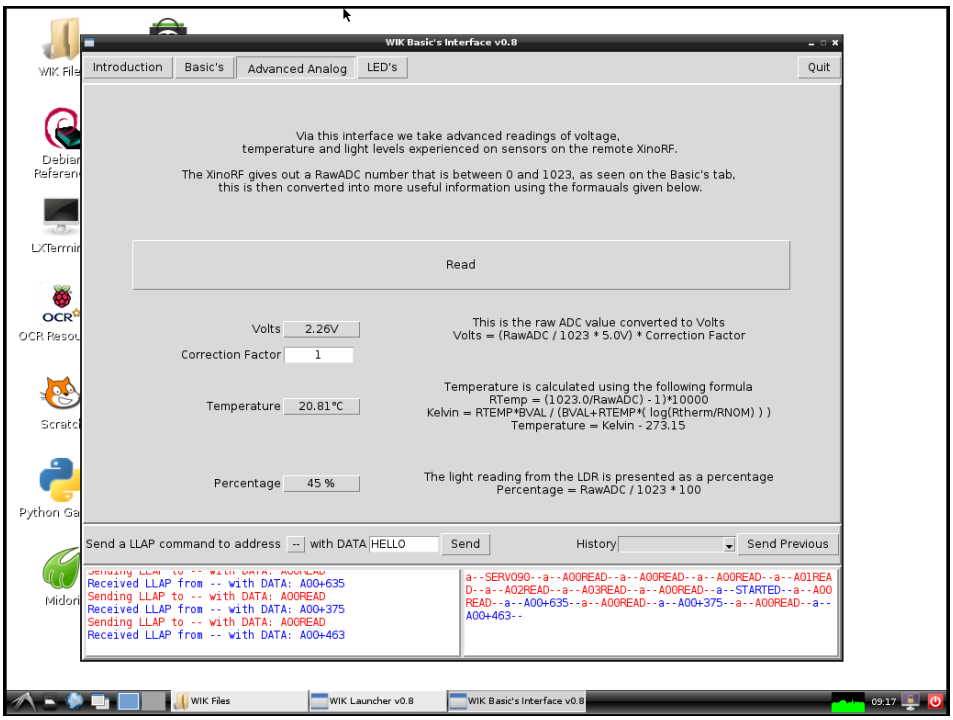


Image 36

Read in the voltage by clicking the button "Read", you will see all the numbers change. The lowest box is the calculated light level as a percentage of the supply voltage in percent. This method is very basic and gives an indication of brighter or dimmer conditions. With such a sensor and code it is not possible to calculate actual Lux or Lumens.

## Activity 10 - Let's measure the temperature in the room

[<< back to contents page]

### What will we learn?

That by reading a variable voltage we can measure the temperature. Using the same techniques as in activity 7 we divide a voltage by two resistors. One is a normal resistor, the second a thermistor. The thermistors resistance varies with temperature, this causes the voltage to change, it's this voltage we read. This change is far from linear so to work out the actual temperature requires some maths. The WIK Basics app will do this for you.

If you would like to know more about how the thermistor works or the different ways there are to calculate the actual temperature take a look **here**

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x 10K resistor

1 x Thermistor

1 x Breadboard

3 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01.

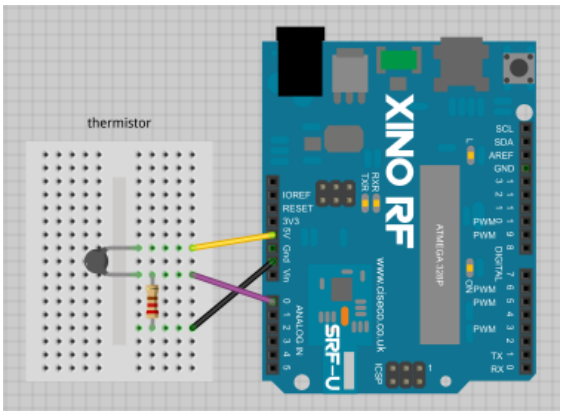


Image 37

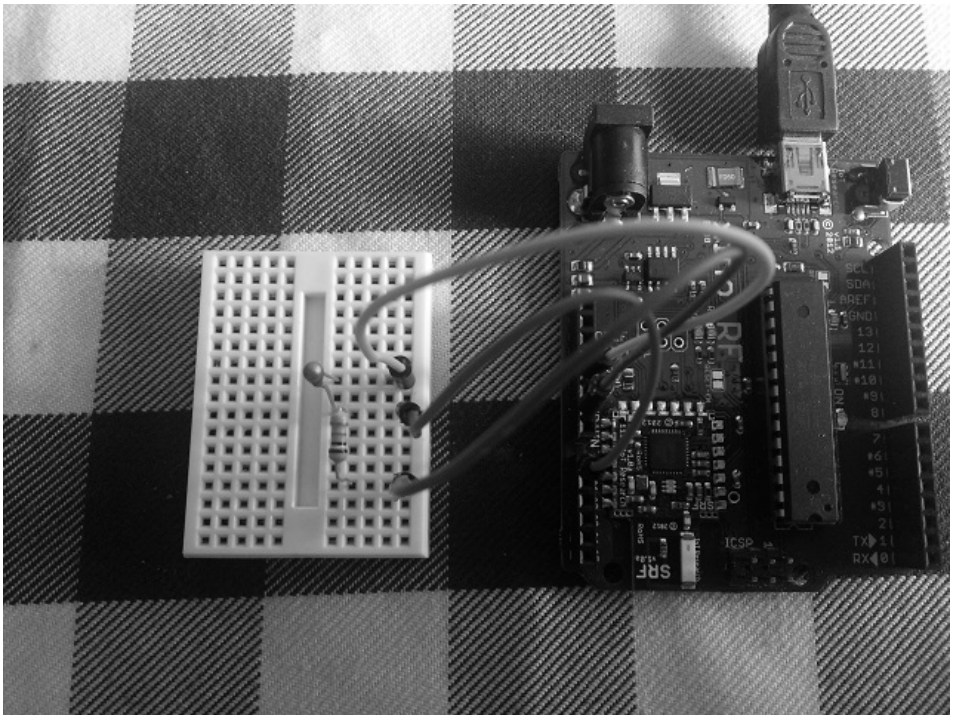


Image 38

Gently push the legs of the thermistor and the resistor into the breadboard as shown.

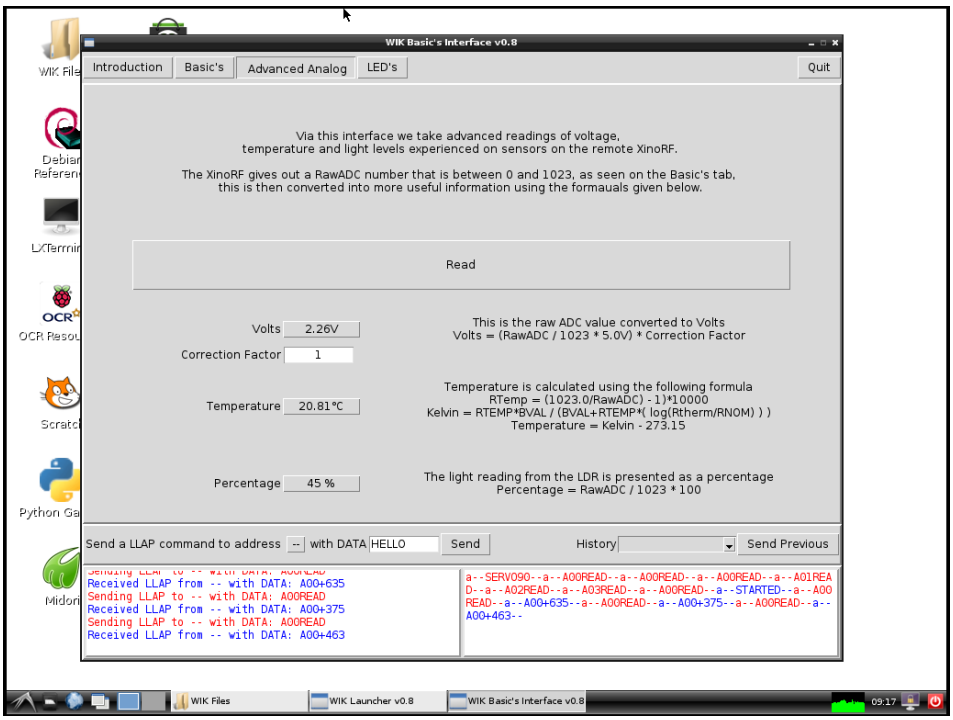



Image 39

Read in the voltage by clicking the button "Read", you will see all the numbers change. The middle box is the calculated temperature. This method is basic and gives an indication within 10% of the current temperature between 0 and 40 degrees. Outside these ranges a more complex calculation and calibration is required. See Steinhart-Hart equation **here**

## Activity 11 - Making a temperature graph

[<< back to contents page]

### What will we learn?

That we can display a series of temperature readings in a live graph.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x 10K resistor

1 x Thermistor

1 x Breadboard

3 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01 and use the exact same circuit as in Activity 10 above.

Quit the Basics Interface, or go to the Introduction tab and click Disconnect to free the Com Port.

From WIK Launcher, launch Graphs. On the Introduction tab, click Connect to connect the Com port for the radio (/dev/ttyAMA0) to the application.

Go to the Temperature tab, where you see a graph area laid out on the right, and a copuple of boxes with variables in on the left.

Delay is the value in milliseconds between readings being taken. It is set to 1000 ms or 1 second to begin with.

Repeat is the number of times a reading is collected. This is set to 20 to start with.

When you click Go, the program will send a request to read A0 every second for 20 seconds and display the temperature in the graph.

Image to be inserted here

Run the program a few times and vary the settings.

When done, disconnect the com port, quit the WIK Graph application and return to the WIK Basics app for the following activities. Don't forget to reconnect the Com port in the Basic Interface.

## Activity 12 - Detecting a finger tap with the piezo sounder

<< back to contents page

**What will we learn?**

That a piezo sounder can be used as a sensor too. When the piezo sounder is tapped by a finger a small voltage is created. We can use this voltage to emulate a button press like in activiy 02. The hardest part of the task is making a connection to the piezo. The piezo has much more flexible wire than the jumper pins and can be very difficult to push into the bread board. What I have found works well is to use the jumper pin to push and hold the sounder wire in to the breadboard. It can take a few attempts. The other option could be to strip a little of the insulation back and twist around the resistor legs. I'd try and avoid this if you can.

**How long will this take?**

No more than 5 minutes

**How difficult is this project?**

Easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x 10K resistor

1 x Piezo Sounder

1 x Breadboard

2 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

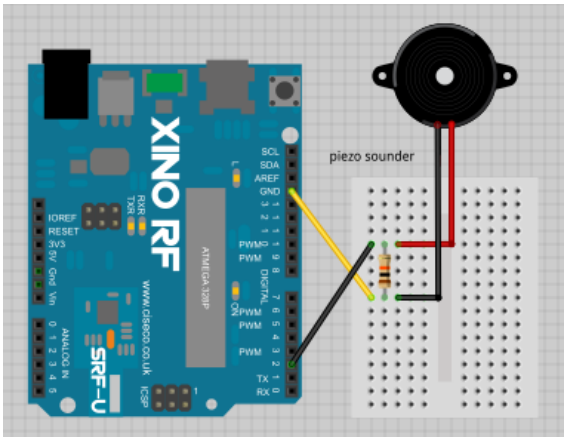


Image 40

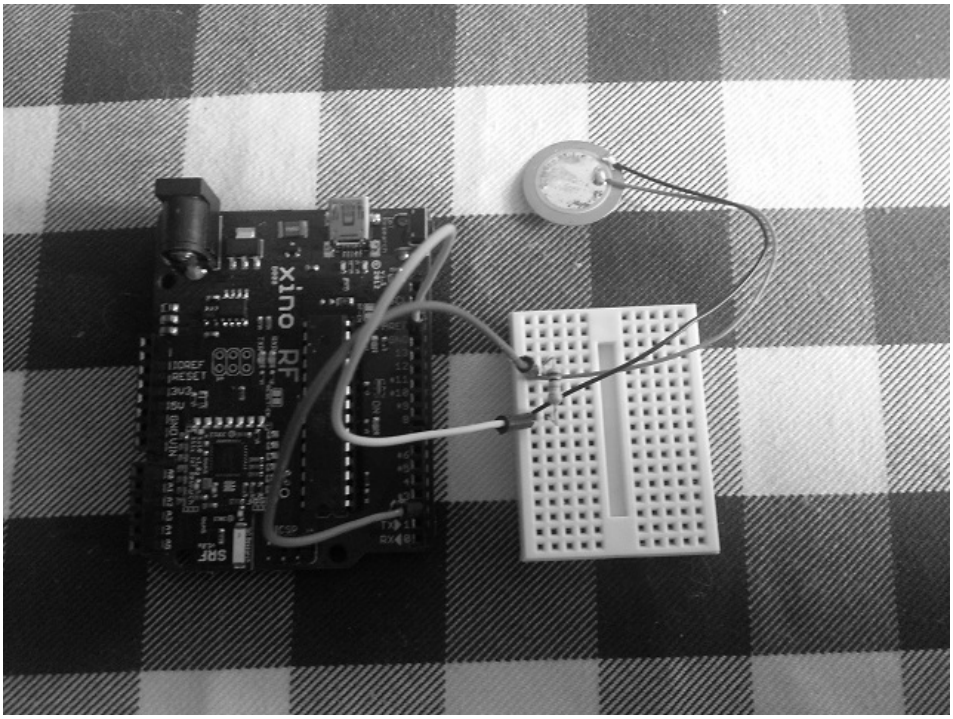


Image 41

Gently insert the piezo wires one by one by using the jumper pin as a way to push both into the breadboard. Add the 10K resistor.

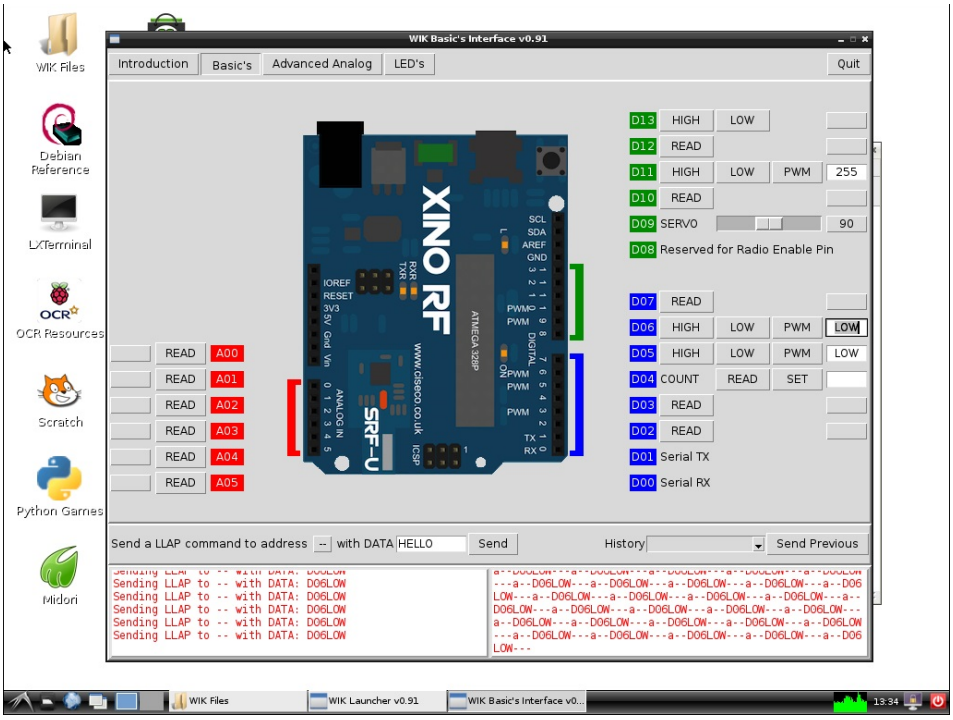



Image 42

On the basics tab, when you tap with your finger or finger nail the sounder, the pin D02 will flick between on and off.

## Activity 13 - Let's first understand what PWM means

[<< back to contents page]

### What will we learn?

This is activity is not a practical one, you only need to read this section. PWM stands for Pulse Width Modulation, I'll explain why it's important later.

If you want a much more detailed idea of PWM see **here**

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

Nothing

### Let's start!

The chip in the centre of the XinoRF is an ATmega328, this inexpensive little computer has no means to generate variable voltages (almost all microcontrollers don't). What we have instead is PWM.

PWM is a way to switch the voltage supply to a pin on and off at varying speeds and for varying durations.

**Why?**

Let's imagine for a moment we switched a motor on and off very fast, off half the time and on half the time, as you can imagine it would go slower. One of the primary uses of PWM, controlling motors. Another common use is lighting. Our eyes are convinced of perfect motion if an image is updated fast enough (about 25 times a second). If you pulse (there's one word of PWM) electricity into a bulb faster than our eyes can detect if you miss a few pulses out, the bulb will appear dimmer. This is how dimmers work in your house.

So we can of create analog style behaviour out of low cost digital hardware, this is why PWM is used so much.

## Activity 14 - Dimming an LED

<< back to contents page

**What will we learn?**

That if we switch an LED on and off fast enough we can dim an LED.

**How long will this take?**

No more than 5 minutes

**How difficult is this project?**

Very easy

**Requirements:**

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x Yellow LED (the choice of colour is up to you but the green is extremely bright)

1 x 470 ohm resistor

1 x Breadboard

2 x Jumper wire

**Let's start!**

Make sure your Pi is installed and running as detailed in activity 01.

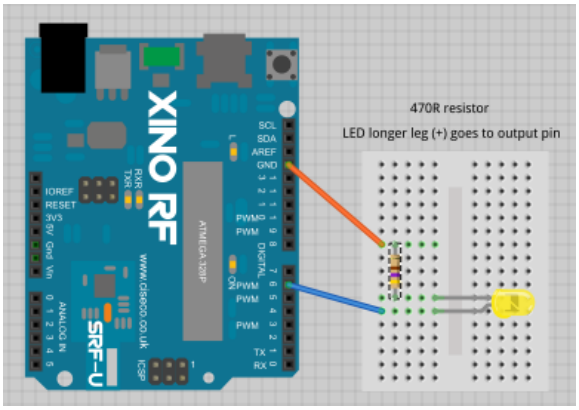


Image 43

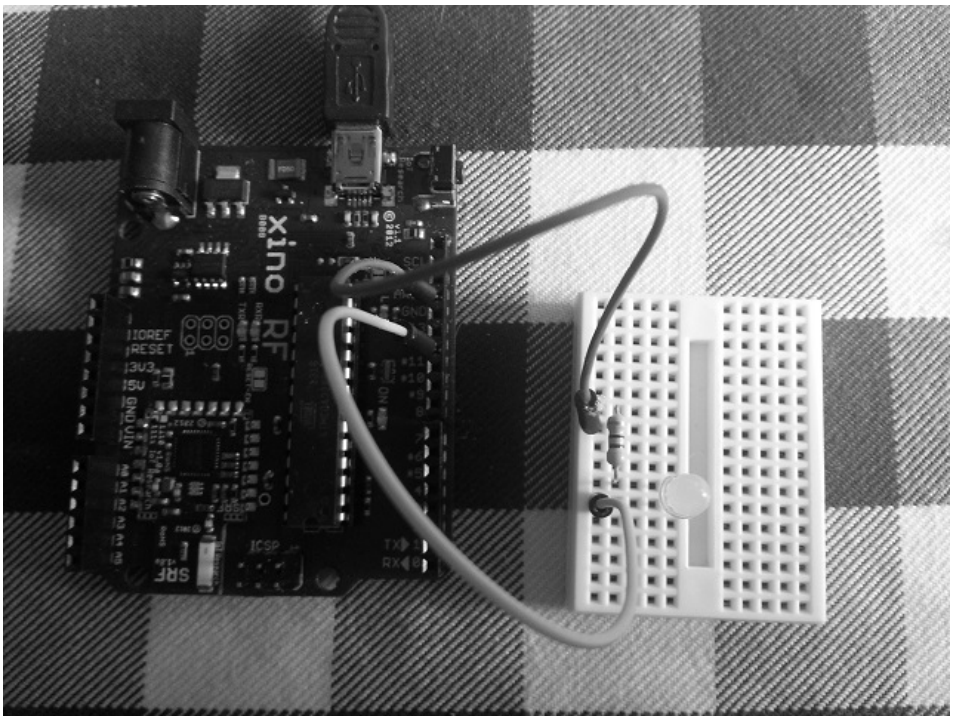


Image 44

Gently push the legs of the LED and the resistor into the breadboard as shown.

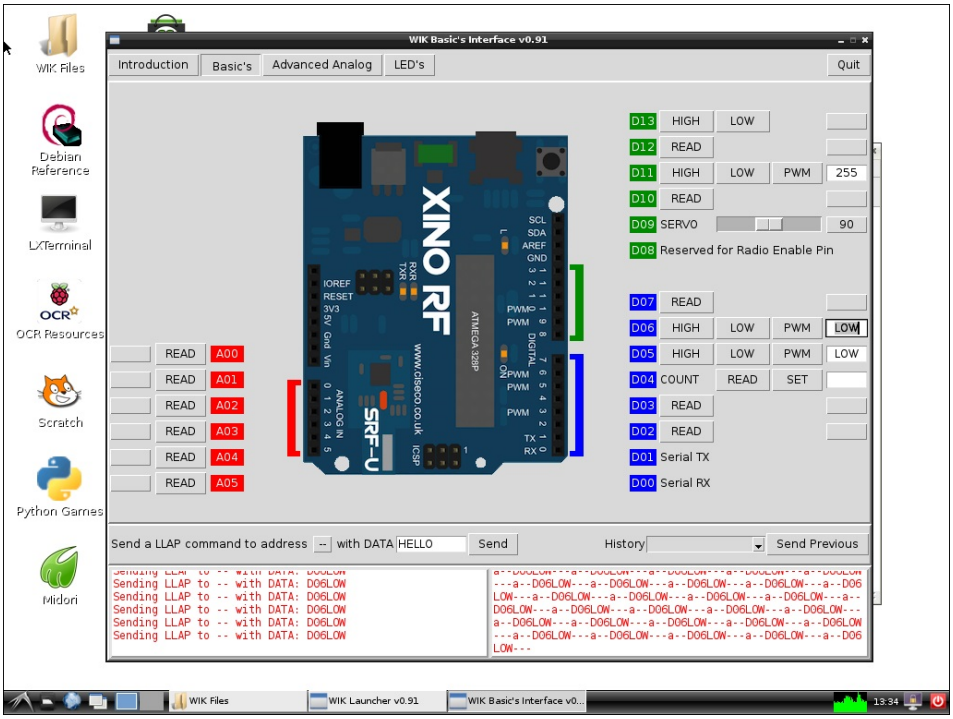


Image 45

Try changing the value of PWM on D11 between 001 (very dim) and 255 (full bright).

## Activity 15 - Dimming 3 LED's to make colours

<< back to contents page

### What will we learn?

That we can mix the light from the LED's to make colours. This is what happens in your TV. The red, green and blue light is mixed to make all the colours. In our kit we have red, green and blue LED's so we can try this out. In your TV all the colours are matched for brightness. In our kit the levels vary so getting true colours will be difficult. It does show

### How long will this take?

No more than 10 minutes

### How difficult is this project?

Easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x Red LED

1 x Green LED

1 x Blue LED

3 x 470 ohm resistor

1 x Breadboard

4 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

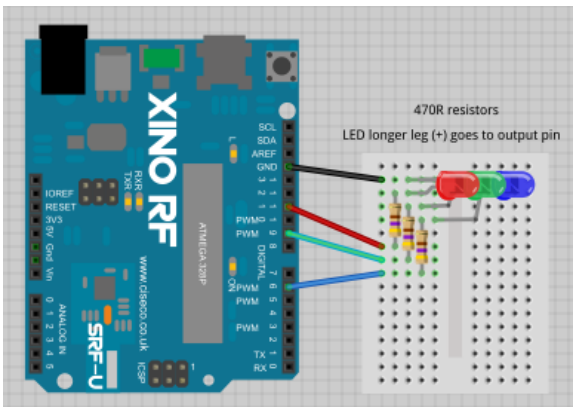



Image 46

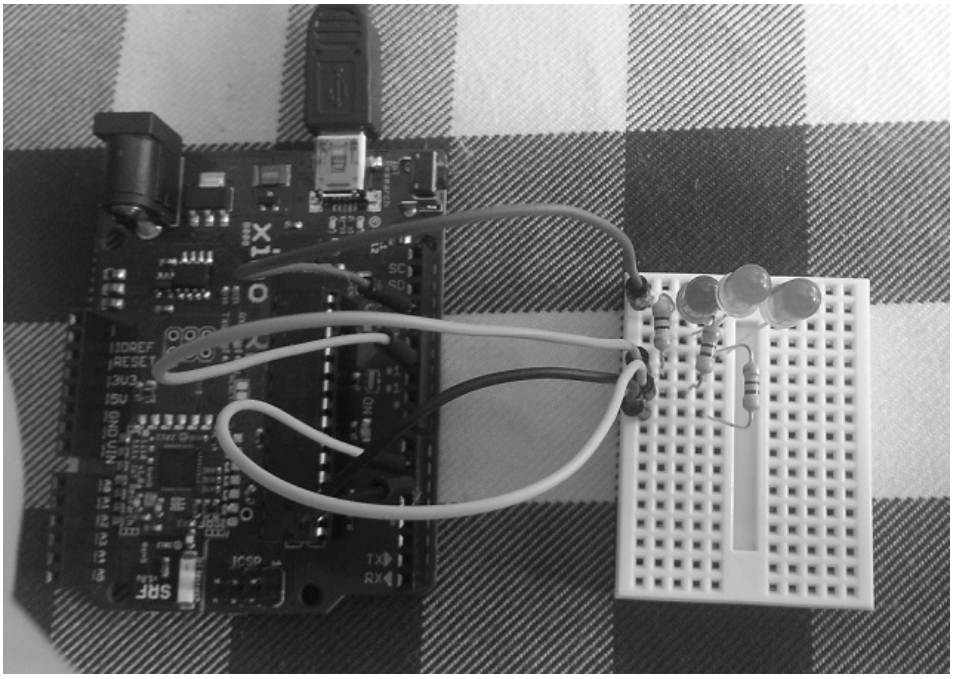


Image 47

Gently push the legs of the LEDs and the resistors into the breadboard as shown.Connect 3 jumper wires to outputs D11 (RED), D06(GREEN) and D05(BLUE)

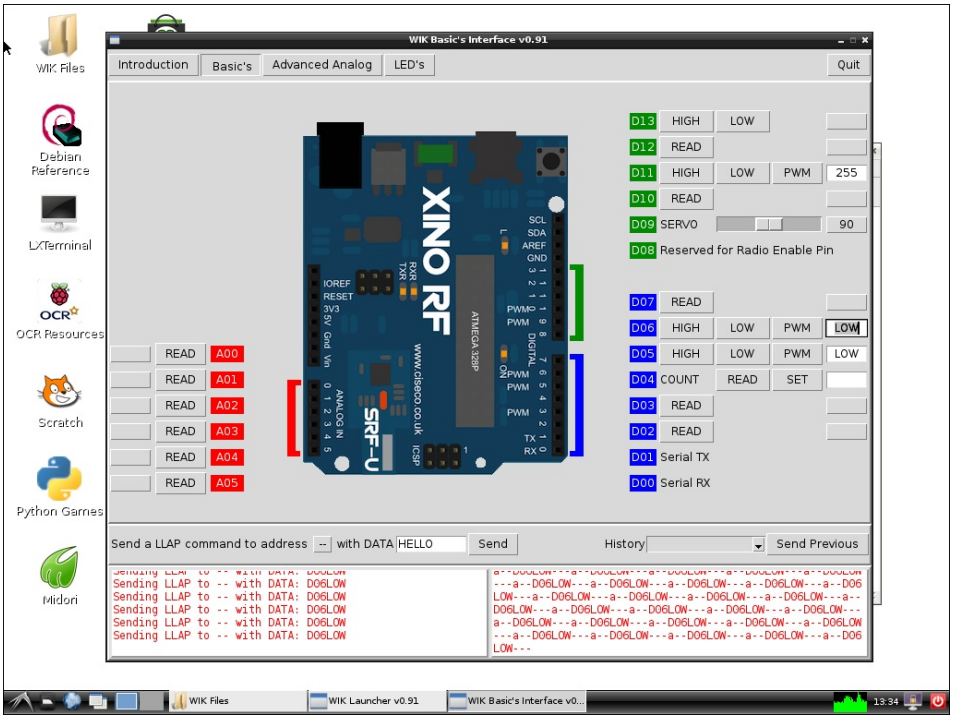



Image 48

In the text box next to D11 enter 018 and click the PWM button next to it. The Red LED should light dimly. In the text box next to D06 enter 250 and click PWM, this will light the green LED. In the text box next to D05 enter 020 and again click the PWM button next to it.

You now have 3 equally lit LED's. You will see the red LED has a much lower number, this is because the red LED is very much brighter than the rest.

Try different combinations against a white piece of paper. You will get some colour mixing, this demonstrates how most colour lighting works. By mixing red, green and blue (RGB) in this way the 255 levels of brightness equates to 16.7 million colour combinations.

# Activity 16 - Generating a noise with the Piezo sounder

<< back to contents page

### What will we learn?

That the piezo sounder can make a sound if fed with a PWM signal. With a single PWM setting we can make single tones, this is what will do in this activity.

With the XinoRF it is possible to play very basic melodies, to do this you will need to get into coding with the Arduino IDE in activity 31. This is the Arduino page that covers melodies **here**

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x Piezo Sounder

1 x Breadboard

2 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

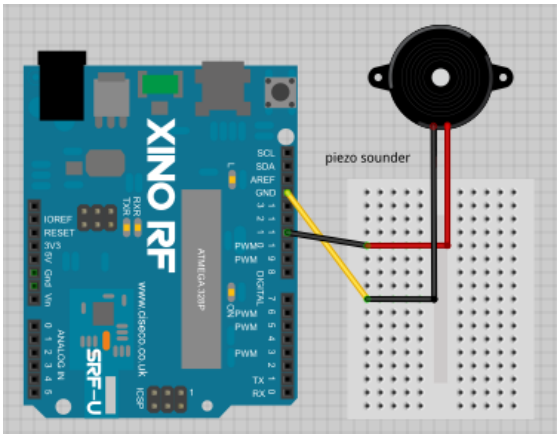
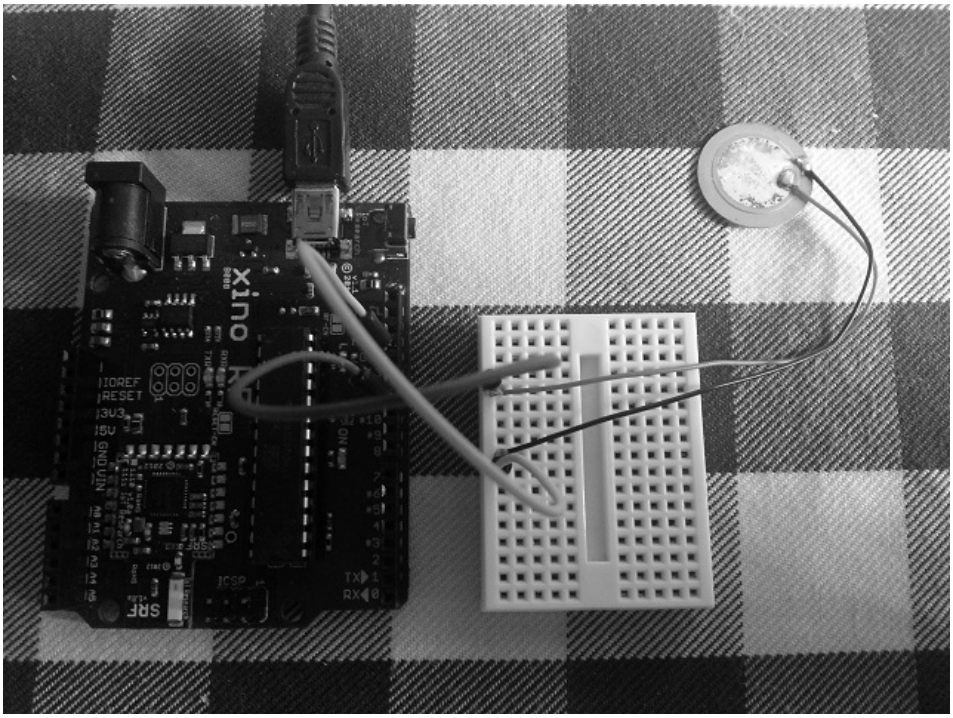
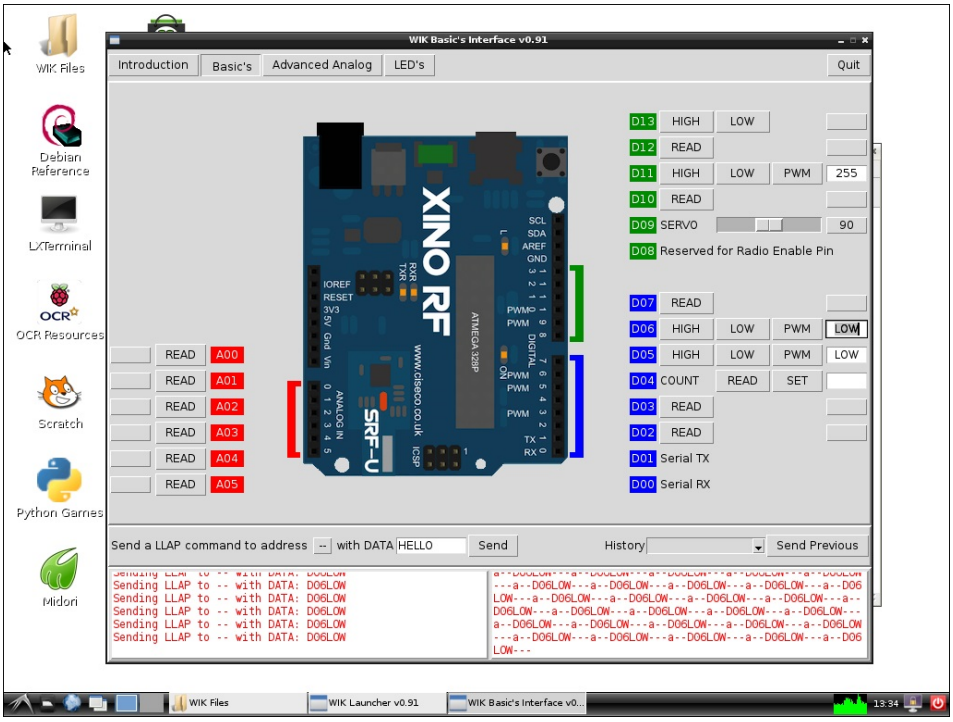



Image 49

Image 50 

Gently push the parts into the breadboard as shown in the photo. The piezo has much more flexible wire than the jumper pins and can be very difficult to push into the bread board. What I have found works well is to use the jumper pin to push and hold in the sounder wire in to the breadboard. It can take a few attempts. The other option could be to strip a little of the insulation back and twist around the resistor legs. I'd try and avoid this if you can.

Image 51 


Enter "060" into the text box on D11 and press PWM, you will hear a tone. Now try "100" or "150" in the txt box and press PWM you'll get a different tone.

The piezo sounder makes a louder tone if pressed onto a hard surface. Just pressing down with you finger will produce a louder sound.

It's not music, but that's not what we were after :)

# Activity 17 - Let's count button presses

<< back to contents page

### What will we learn?

Earlier we saw how to show how button presses could change the input on the XinoRF. We can use a similar circuit and pin D04 to count button presses instead.

### How long will this take?

No more than 5 minutes

### How difficult is this project?

Very easy

### Requirements:

1 x XinoRF

1 x USB cable or DC jack to power the XinoRF (6v to 12v is ideal)

1 x Raspberry Pi (Model A or Model B) + keyboard + mouse + monitor

1 x Slice of Radio

1 x SD card as supplied with your kit

1 x Switch

1 x Breadboard

2 x Jumper wire

### Let's start!

Make sure your Pi is installed and running as detailed in activity 01.

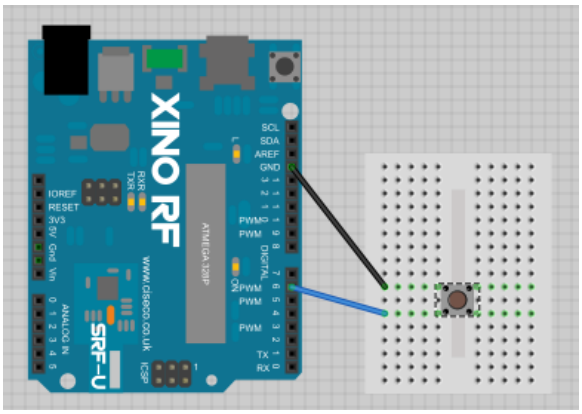



Image 52

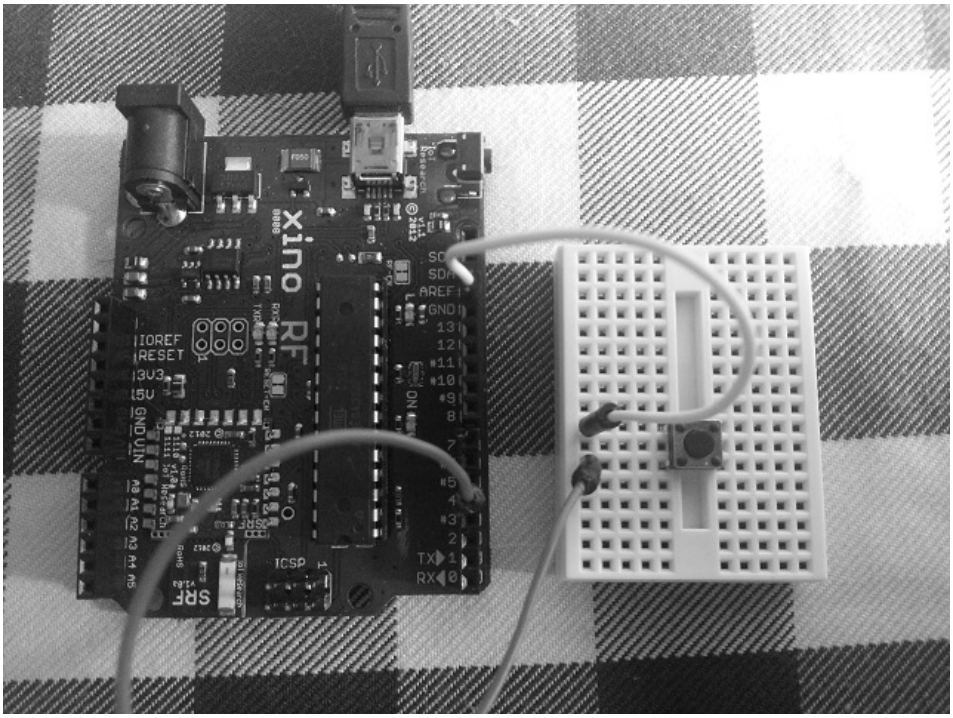



Image 53

Gently push the parts into the breadboard as shown in the photo.

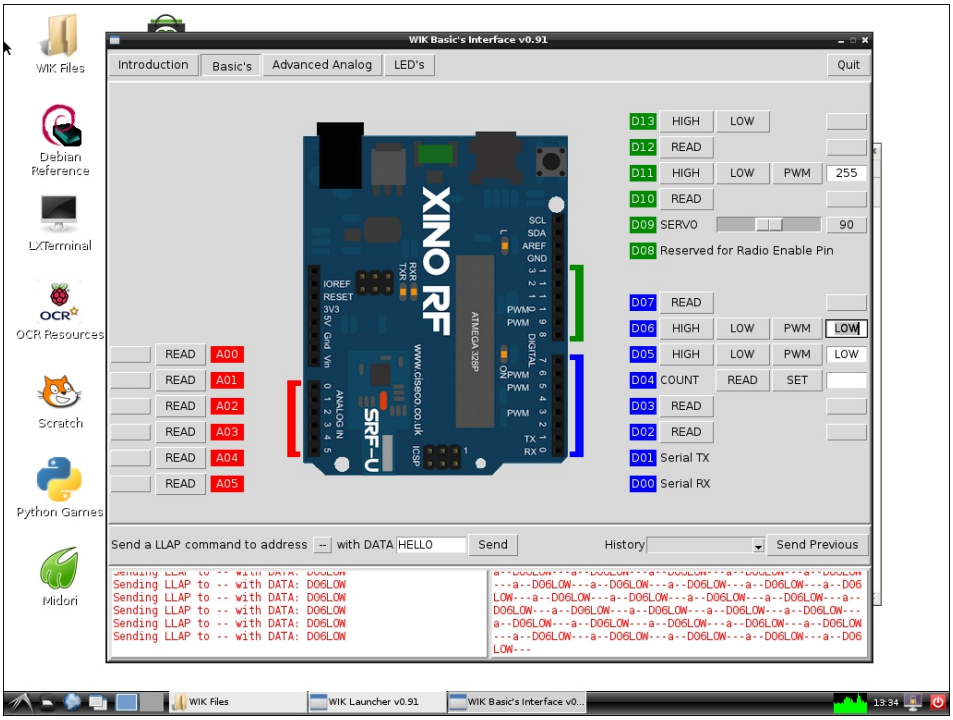



Image 54

Press the button a few times and then click "READ" next to D04 you will see the count has increased by the number of times you pressed the number. This time rather than send immediately the XinoRF is counting in its internal memory. Try this a few more times.

As the number increases you might want to set the counter back to zero, simply type 000 into the text box and click "SET". You will see the next time you click "READ" the counter will be a zero once more.

You can set the counter to any number you wish between 0-9999. Try setting to 9999 and pressing the button a few times, you'll see it reset to zero and began counting again. This is an important thing to observe as most computers when they reach the maximum number circle back to zero again.

## Further activities

This is the end of Part 1 of the RasWIK instructions.
In Part 2 we will start to write our own programs to send and receive LLAP messages and control the XinoRF. We will also log data in a file and with an Internet of Things (IoT) service on the Internet.