Button Input

Project

Interaction between a button and an LED.

Description

When the user presses the button an LED will light up, when they stop pressing the button the LED will go back off.

Equipment You Will Need

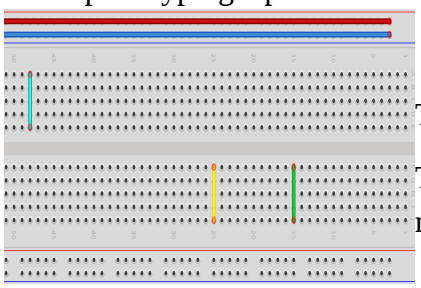
- Raspberry pi and an SD card
- Mouse and Keyboard
- Monitor and HDMI cable
- 1 x Red LED
- 4 x male to female jumper wires
- 1 x male to male jumper wire
- 1 x 400 point breadboard
- Power supply
- 1 x 330Ω resistor
- 1 x 4.7 k Ω resistor

The Parts You Will Need

Within this section I will go through how the parts that we are going to use work and how they should be connected.

The Breadboard

The breadboard is used to connect electronics together instead of soldering. The breadboard is often used in prototyping a product.



The red line connects all the dots within this row.

The blue line connects all the dots within this row.

The turquoise line connects all the dots within this column, but is not connected to either the yellow or green lines

The yellow line connects all the dots within this column, but is not connected to either the green or the turquoise lines.

The green Line connects all the dots within this column, but is not connected to either the yellow or turquoise lines.

The LED's

LED stands for Light Emitting Diode. They glow when electricity is passed through them. When looking at LED's you will notice that there is one leg longer than the other. The longer leg known as the anode is connected to the positive supply of the circuit. The shorter leg known as the cathode is connected the the negative side of the circuit also known as ground.

If you connect the LED's the wrong way round this is not the end of the world because you will not damage them, they just will not light up.

The Resistors

Resistors limit the amount of current that can flow through an electrical circuit. Resistors are measured in ohms Ω, The larger the resistance the more the current is limited.

Resistors can be identified by the coloured bands along the body. For example the resistor that we are using is 330 Ω, this is identified by orange, orange, brown then gold.

We have to use resistors to connect the LED's to the raspberry pi, or the LED's will pull too much current from the raspberry pi and this will cause the raspberry pi to stop working.

The resistors can be connected any way round because the current of the electricity will flow both ways.

The Jumper Wires

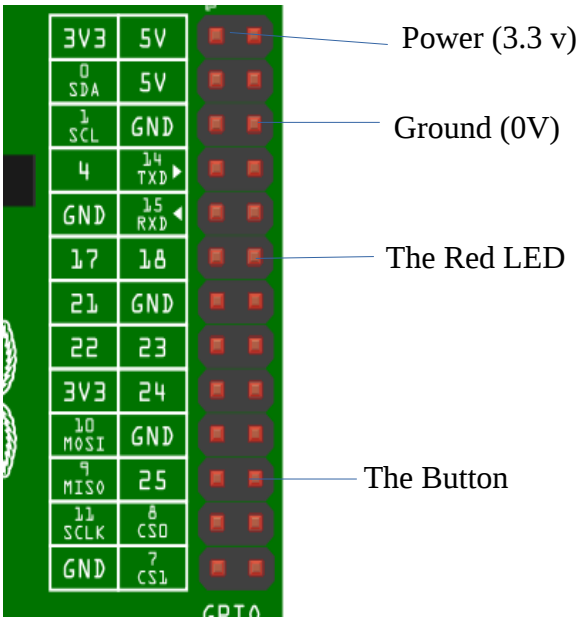
Jumper wires are used on breadboards to jump from one connection to another. The jumper wires we are using are called male to female jumper wires. The end with the pin on it will go into a hole on the breadboard, and the end with the hole will connect to a GPIO pin on our raspberry pi.

Making The Circuit

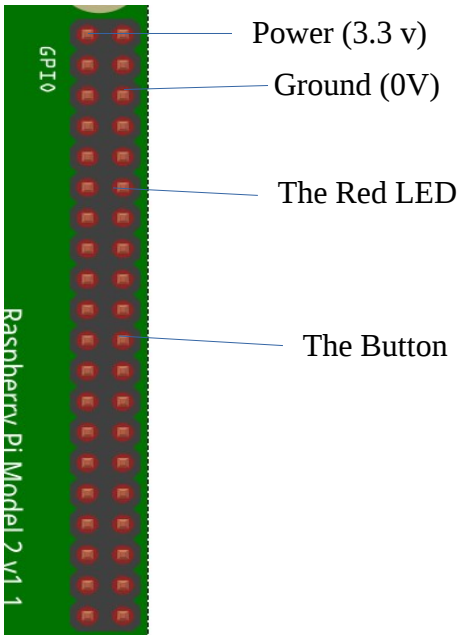
To make the circuit it is always best to have the raspberry pi turned off at this stage.

The first thing we are going to do is take a look at the GPIO (General Purpose Input Output) pins.

Models A & B



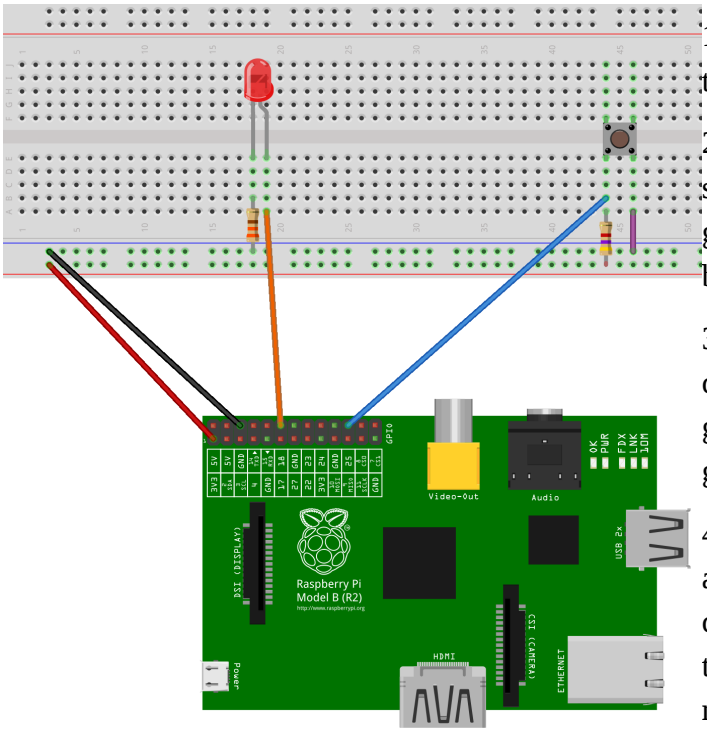
Model A+, B+ & Raspberry Pi 2



It is through the GPIO pins that the raspberry pi can control and even monitor the outside world by being connected into an electronic circuit. The raspberry pi can control LED's Motors, button presses and even buzzers.

The diagrams above show the pin layouts of the model A and B on the left and Models A+, B+ and the raspberry pi 2 on the right. I have marked out the pins we will be using and what will be connected to them.

We are now ready to set up our circuit.



1. Put the Red LED into the breadboard with the longer leg on the right hand side.

2. Connect the 330 Ω resistor in between the short leg of the Red LED and the common ground rail, which is the second row from the bottom.

3. Take a male to female jumper wire and connect the end with the pin into the common ground rail and the side with the hole over the ground pin on the raspberry pi.

4. Take another male to female jumper wire and connect the end with the pin into the same column as the longer leg of the Red LED and the side with the hole onto pin 18 on the raspberry pi.

5. take another male to female jumper wire and connect the pin end into the power rail which is the very bottom row on the breadboard and end with the hole over the 3.3 v pin on the raspberry pi.

6. Now insert the button on to the breadboard, with one set of legs on either side of the gap in the middle.

7. Takes the 4.7 k Ω resistor and connect it between the left hand set of legs and the power rail on the breadboard.

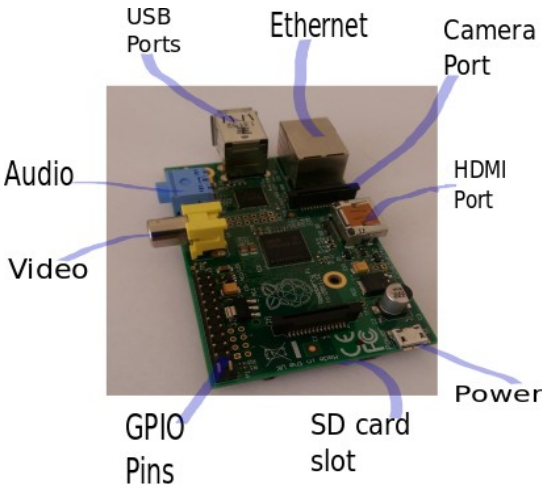
8. Take the last male to female jumper wire and connect the end with the pin in between the legs of the button and the resistor and the end with the hole to pin 25 on the raspberry pi.

9. Take the male to male jumper wire and connect it between the legs on the righthand side of the button and the other end into the ground rail on the breadboard, which is the second row from the bottom.

Now we have our circuit ready we can power up the pi and do some coding.

Setting Up The Raspberry Pi

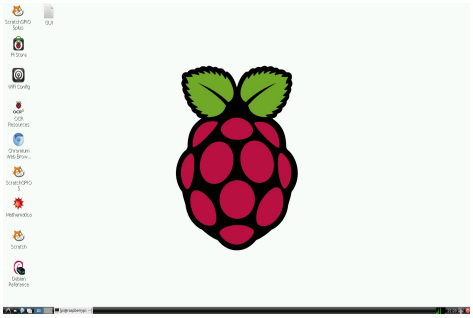
- Plug in your SD card or Micro SD card depending on the raspberry pi you have
- Plug one end of the HDMI cable into the raspberry pi and the other end into the monitor
- Plug in the mouse to one of the USB ports
- Plug the keyboard into another USB port
- plug in the power supply into the micro USB port.

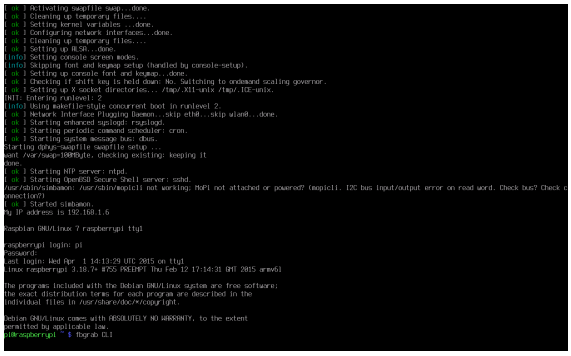


After you have connected all the wires and inserted your SD card, your pi should look a little like this.

After the raspberry pi has finished booting, you will either end up in the command line interface (CLI) as in the picture below on the right or in the graphical user interface (GUI) as the picture below on the left. If you see the CLI then you will need to:

- Type the user name “pi”
- Type the password. The default password is “raspberry”





Code

You are now going to create your first python program that will take input from you and display it on the screen.

If you are not in the CLI then you will want to open the LxTerminal application. 

The first thing we are going to do is create a new directory called Python. This is where all our python code is stored. Type the following commands which are in blue pressing the “enter” key after every line:

cd ~	Changes to the home directory
mkdir Python	Creates a directory (folder) called Python
cd Python	Changes to the python directory
nano ButtonInput.py	Opens the “nano” text editor creating the file called “ButtonInput.py”

You will now be in the “nano” text editor. This is where we will type our code. Type in the code exactly as seen below, which is in red.

#! /usr/bin python	Telling the pi to use the python interpreter
import os	This is for operating system functionality
import time	This is for time related commands
import Rpi.GPIO as GPIO	Importing the GPIO library as GPIO
GPIO.setmode (GPIO.BCM)	Set the GPIO pins naming mode
GPIO.setwarnings (False)	Telling python not to display GPIO warnings to screen
Led = 18	Setting pin 18 as a variable for the Red LED

Button = 25	Setting pin 25 as a variable for the button
Print “Button + GPIO”	Printing to screen
GPIO.output (Led, True)	Turning the LED off
while 1:	
if GPIO.input (Button):	Checking to see if the button has been pressed or not
GPIO.output (Led, False)	Turning the LED on if the button has been pressed
else:	Button has not been pressed
GPIO.output (Led, True)	LED off
GPIO.cleanup ()	Setting the GPIO pins back to default

Running The Code

Sudo python ButtonInput.py

Exiting The Program

To exit the program press ctrl + z.

That is all for this tutorial please join me again for the next one

Happy coding from the RaspiKidd

You can also find all my tutorials on my website: www.raspikidd.me.uk, where you can find my links to Facebook and twitter.

If you have any questions please leave a comment tweet post on Facebook or just email me. The email address is at the bottom of the page.