



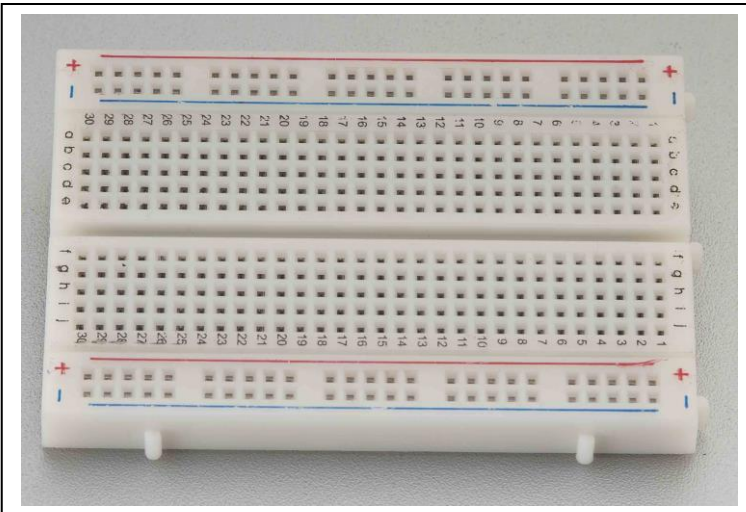
SoSLUG - BIKE Basic Interfacing Kit - Electronics GPIO Starter Kit Information

Southend-on-Sea Linux Users Group (SoSLUG)
<http://www.soslug.org/>

Kit contents

- 1 x 400 point breadboard
- 10 x Female – Male jumper wires, multi-coloured
- 5 x Male – male jumper wires, multi-coloured
- 3 x Red 5mm LED
- 2 x Yellow 5mm LED
- 2 x Green 5mm LED
- 1 x Buzzer
- 2 x 12mm square push buttons
- 1 x Red button cap
- 1 x Green button cap
- 1 x NPN transistor (BC348B)
- 7 x 220R resistors (coloured bands red, red, brown)
- 1 x 4K7 resistor (yellow, purple, red)
- 2 x 10K resistors (brown, black, orange)
- 1 x storage container

Breadboard

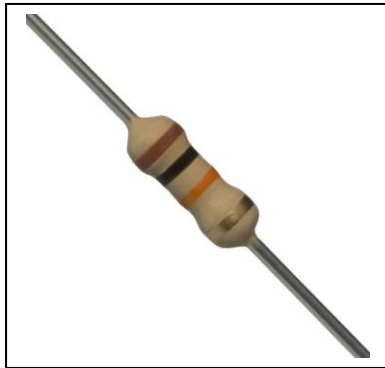


The breadboard has a number of rows that are linked together, these can be used to interconnect the components. There are also at each long side lines of holes that are connected together, these are intended to connect power to. By convention the red line is for a positive connection and the blue for the ground.

Push buttons and caps

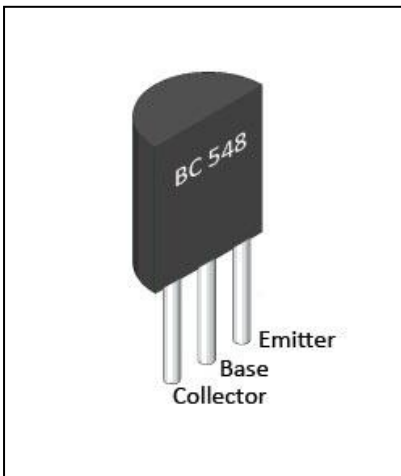


Press the red button cap carefully on to the shaft of one of the buttons. Ensure that the pins are not damaged and be careful how you hold the button whilst you do this, the pins are sharp and could pierce your skin. Do the same with the green cap on the other button. There are four pins on each button, ensure when you place them on the breadboard that the switch contact pairs are not bridged. The switch contacts are between the narrow pins rather than the longer ones, they are the same top and bottom.



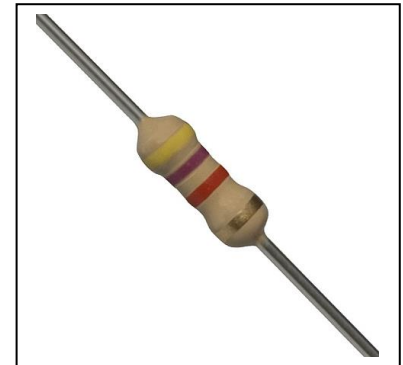
The 10K resistor can be used with a push button to form a pull-up or pull-down arrangement. This is not strictly necessary with the Raspberry Pi as software configuration can be used to turn internal pulling on and off and to set the direction.

NPN Transistor



When using the NPN transistor it is important to connect it the right way round otherwise your circuit might not work and erroneous connections could cause damage.

Always ensure the 4K7 resistor is connected in series with the transistor base connection, otherwise damage could occur.

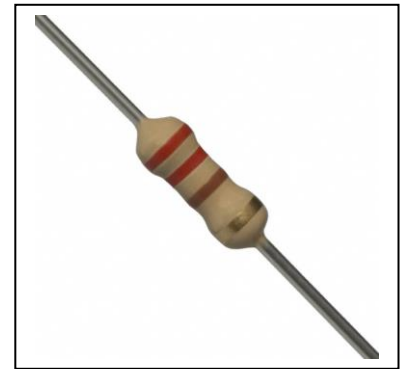
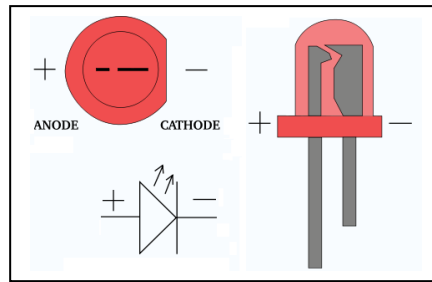


Buzzer



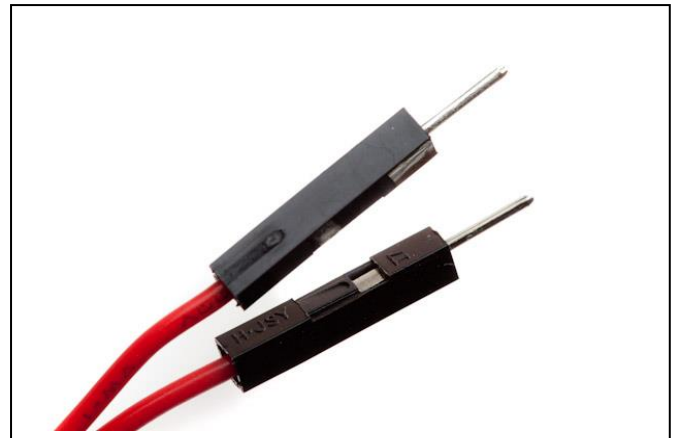
The buzzer is intended to work on 5 Volts (pin 2 on the Pi), do not connect directly to a GPIO pin, it is not likely to work very well and might damage your Pi. Always use the transistor circuit to drive it. The buzzer is a polarised device make sure that positive (+) pin is connected to the 5 volt supply . Remove the protective sticker before use.

LED's



LED's are polarised devices and must be connected with the correct orientation for them to work. They must always be used with one of the 220R resistors otherwise damage could result to the LED and the Raspberry Pi.

Jumper wires



The kit contains 10 different coloured female to male jumper wires, these are used to connect the pins of the General Purpose Input Output (GPIO) ports to the breadboard. The kit also contains 5 different coloured male to male jumper wires, these are used to make connections on the breadboard.

The colours of the jumpers shown on the layout diagrams may not match the colours in the kit, having the correct colour is not important, on the drawings the use of colour coding for the jumpers is for reasons of clarity.

GPIO Use

This kit is designed to be used with the Raspberry Pi, although any compatible microprocessor or micro-controller could be used.

There are various revisions of the Raspberry Pi and there are differences in the pin-outs of the GPIO header. The table below shows the correct pin out for the various models.

GPIO Numbers

There are two GPIO number schemes that are in existence for the Raspberry Pi and various programming languages. There is the 'chip' or 'Broadcom' version, these are the numbers shown like GPIO24. Then there is the second version, known as the 'board' or 'pin' scheme, this refers to the actual pin number of the header. ScratchGPIO can only use this second 'board' identification scheme.

To make the wiring of the components and connections easier, we have adopted throughout the second scheme – pin or board numbering system. As far as possible again for ease we have tried to adopt a consistent allocation for each of the components, this is shown in the table below.

Item	Pin Number
Buzzer	21*
Push button 1	8
Push button 2	10
LED Red 1	12
LED Yellow 1	16
LED Green 1	18
LED Red 2	11
LED Yellow 2	13
LED Green 2	15
LED Red 3	21*
3V3	1
Ground	6

* Buzzer and LED Red 3 use the same pin (but not at the same time)

Using Python for accessing the GPIO's on the Raspberry Pi

All the example programs have been created and tested in Python version 3.2. If you get errors about the GPIO check that you are using version 3.2 To use GPIO pins with Python you need to run it as a Super User (sudo = super user do) or root. You can alternately start the editor as the root user – open a terminal window and enter `sudo idle3` and then you can run the programs from the editor

Using Scratch for accessing the GPIO's on the Raspberry Pi

Scratch is a popular means programming for young novices. There is a version that runs on the Raspberry Pi. It can not be used to access the GPIO's, to do this a specially adapted version is required, it is called ScratchGPIO. The current release is version 5 and needs to be installed.

When ScratchGPIO5 is installed it puts two icons on the desktop – ScratchGPIO5 and ScratchGPIO 5Plus. ScratchGPIO5 is intended for beginners using simple circuits whilst ScratchGPIO 5Plus is for use with specific add-on boards. At the elementary level both work almost the same. All the example programs have been created and tested in ScratchGPIO5. You can run scratch without having to be the 'root' user.

The 'Plus' option has nothing to do with the Raspberry Pi B+ model and although both ScratchGPIO5 and ScratchGPIO 5Plus will work on the Raspberry Pi B+, the extra GPIO pins of the B+ model can not be accessed. ScratchGPIO will not work on other computers as there are no GPIO pins, these only exist on the Raspberry Pi.

To get you up and running with the minimum of fuss ScratchGPIO comes by default with six pins set for outputs (Pins 11,12,13,15,16 and 18) and all the rest (22,7,3,5,24,26,19,21,23,8 and 10) are configured as simple inputs.

The configuration can be changed by way of 'broadcast'. If a GPIO currently configured as an input is broadcast to it will automatically be changed to output mode.

There is a slight difference with the inputs of version 5 and 5Plus. By default ScratchGPIO5 configures all inputs with an internal pull-up resistor enabled. ScratchGPIO 5Plus default inputs are configured without any pulling resistor enabled. The pull-up/pull-down status can be altered using the keywords SetPinsLow, SetPinsHigh and SetPinsNone. Note there appears to be no means of altering pull-up/down on individual pins, all are affected.

The full range of simple commands that can be used is available at <http://simplesi.net/scratchgpio/visual-command-guide/> Links from this page (menu bar top left) describe some of the more sophisticated commands, for example speed control of a motor.

Further information

A range of project sheets along with sample code is available on github see <https://github.com/soslug/bike.git>

These can be downloaded and viewed on the Raspberry Pi, along with the examples. To do this you will need to be connected to the Internet either wirelessly or by cable (Ethernet). You will also need to have git installed on your Raspberry Pi and it is always a good idea when installing something new to perform an update. Type in the following commands in a terminal window:

```
sudo apt-get update
```

```
sudo apt-get install git
```

Then you can access the files by typing

```
git clone https://github.com/soslug/bike.git
```

This will save the files to a folder in your directory home called 'bike'. You can use the command:

```
cd bike
```

To navigate to that directory once installed.

If you want to get the latest worksheets and examples and have used git clone already you need to navigate to the 'bike' folder as shown above and type

```
git pull
```

The notes and examples can be grabbed from github as a zip file and opened with any archive manager from <https://github.com/soslug/bike/archive/master.zip>

There is a wealth of information on the Raspberry Pi and the various components as well as theoretical knowledge available on the Internet. It is too vast to provide links

to suitable resources, but in general the Google search engine and Wikipedia are usually good starting places for information.

We are also in the process of creating a forum for discussion of matters related to the kit, at the time of writing details are not finalised, be sure to check periodically on the Soslug website for details of the forum.

Until the forum is up and running we have set up a subscribed group email for help with the kits. To be able to use this, you first need to subscribe by sending a blank email to bike-subscribe@soslug.org Once you have received an automatically generated welcoming response you will be able to post any question or respond to any group email on bike@soslug.org This email group is moderated, so your email will wait for the one of the administrators to approve it before being circulated to the group. You can un-subscribe from this group by sending a blank email to bike-unsubscribe@soslug.org

Soslug holds meetings on Mondays starting at 7pm where more information and help is available – see www.soslug.org

Caution

Electronic devices such as the Raspberry Pi and components included can be irreparably damaged as a result of mistakes and accidents. Care has been made in ensuring that with proper use and connection the components will not damage the raspberry Pi nor will the Raspberry Pi damage the components. However it must be realised that during experimentation there is a high risk that mistakes and accidents may occur and that either the Raspberry Pi and/or attached components may not tolerate wrong connections and fatal damage might be caused.

ALWAYS switch off the Raspberry Pi before changing leads and components.

SoSlug and the producers of this kit can not accept responsibility for any damage to the Raspberry Pi or components or anything attached to parts used in this kit.