Traffic Lights

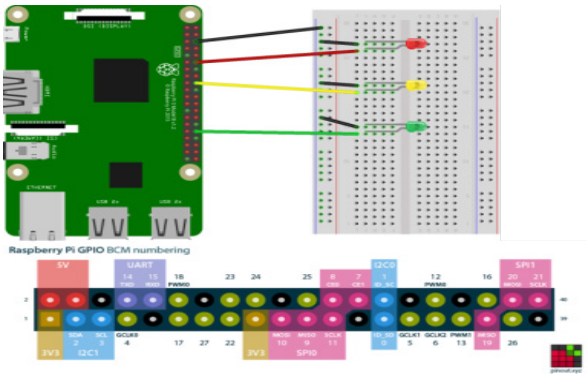
Within this tutorial we are going to create a simple circuit using 3 LEDs 1 x red, 1 x yellow and 1 x green. We are going to use EduBlocks and a Raspberry Pi to code and control the LEDs to simulate traffic lights.

You Will Need

- 1 x breadboard
- 1 x Red LED
- 1 x Yellow LED
- 1 x Green LED
- 3 x male to male jumper wires
- 4 x male to female jumper wires
- 1 x Raspberry Pi setup

Circuit

Lets build the circuit. Make sure you have your LED the right way round. The positive leg is the longer leg on the LED, this is represented by the bent leg in the diagram below:



Starting EduBlocks

There are two things we need to do to start EduBlocks

1. Go to menu -> programming and click on EduBlocks Connect. (if you have EduBlocks set to start on Startup go to step 2)
2. 2open a web browser and type app.edublocks.org and click on Raspberry Pi.

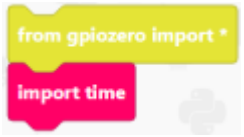
NOTE: If you have a raspberry pi set up on your home network with EduBlocks installed and running you can access it from another computer using it's IP address.

Code

Setting up libraries

1. Click on **gpiozero**
2. Click on **General**
3. Click and drag **from gpiozero import *** to the coding are and drop it there.
4. Click on **Basic**
5. Click and drag an **import time** block and attach it under **from gpiozero import ***

Your code should look like this:



This imports Python Libraries. gpiozero helps us communicate with the GPIO pins on the Raspberry Pi and the time library lets us control pauses within the code.

Setting up pin numbers

1. Click on **gpiozero** menu click on **Outputs**
2. Click on **LED**, click and drag an **led=LED()** to the code area and attach it under **from gpiozero import ***
3. Click on the arrow next to **led** and click on **Reanme variable** type **red** into the text box and click on OK.
4. Within the blank space between the **()** type **18** (this refers to the pin number that the red LED is connected to on the Raspberry Pi)
5. Click on **LED,** click and drag an **led=LED()** block to the coding area and attach it under the **red=LED(18)** block.
6. Click on the small arrow next to **led** and click on **Rename variable** in the text box type **yellow** and click OK.
7. Within the blank space between **()** type **24**

(this corresponds to the pin on the Raspberry Pi that the yellow LED connects to)

8. Click on **LED**, click and drag an **led=LED()** block to the code area and attach it under the **yellow=LED(24)**
9. Click on the small arrow next to **led** and click on **Rename variable** and type **green** into the text box and click OK.
10. Click within the blank space between **()** and type **12** (this refers to the pin on the Raspberry Pi where the green LED connects to)

Your code should now look like this:

Setting up a loop

Click on **Basic**, click and drag a **while True:** block to the code area and attach it under the **green=LED(12)** block.

Your code should now look like this:

All code from now on will go inside the **while True:** This will keep our code running forever simulating traffic lights.

Turning the red LED on

1. Click on **LED**, click on an **led.on()** block and drag it to the coding area and attach it within the **while True:** block
2. Click on the small arrow next to **led** and click on **red**
3. Click on **Basic**, click and drag a **time.sleep(1)** block to the code area and attach it under the **red.on()** block. Change the **1** to a **3**.

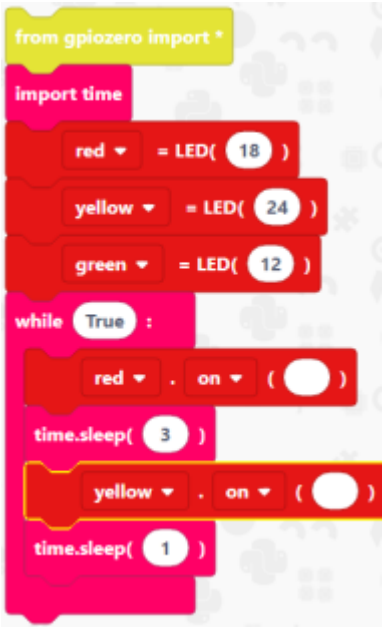
Your code should look like this:

This will turn the red LED on for 3 seconds.

Turning the yellow LED on

1. Click on **LED**, click and drag an **led.on()** block to the code area and attach it under the **time.sleep(3)**
2. Click on the small arrow next to **led** and click on **yellow**
3. Click on **Basic,** click and drag a **time.sleep(1)** block to the coding area and attach it under the **yellow.on()** block.

Your code should now look like this:

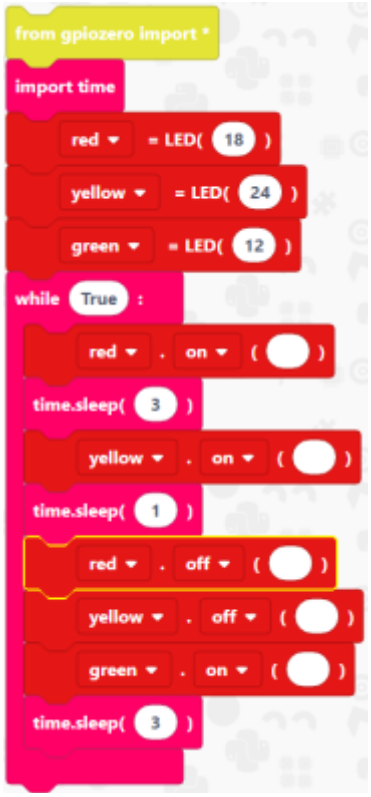


The red LED is still on and the Yellow LED turns on for 1 second.

Turning the red and yellow LEDs off and the green LED on

1. Click on **LED**, click and drag an **led.on()** block to the code area and attach it under the **time.sleep(1)** block
2. Click on the small arrow next to **led** and click on **red**
3. Click on the small arrow next to **on** and click on **off**
4. Click on **LED**, click and drag an **led.on()** block to the code area and attach it under the **red.off()** block
5. Click on the small arrow next to **led** and click on **yellow**
6. Click on the small arrow next to **on** and click **off**
7. Click on **LED**, click and drag an **led.on()** block to the code area and attach it under the **yellow.off()** block
8. Click on the small arrow next to **led** and click on **green**
9. Click on **Basic**, click and drag a **time.sleep(1)** block to the code area and attach it under the **green.on()** block.
10. Change the **1** to a **3**.

Your code should now look like this:



This turns the Red and Yellow LEDs off and turns the Green LED on for 3 seconds.

Turning the yellow LED back on

Click on **LED**, click and drag an **led.on()** block to the code area and attach it under **time.sleep(3)** block
Click on the small arrow next to **led** and click on **green**
Click on the small arrow next to **on** and click on **off**
Click on **LED**, click and drag an **led.on()** block to the code area and attach it under **green.off()**
Click on the small arrow next to **led** and click on **yellow**
Click on **Basic**, click and drag a **time.sleep(1)** block to the code area and attach it under **yellow.on()**

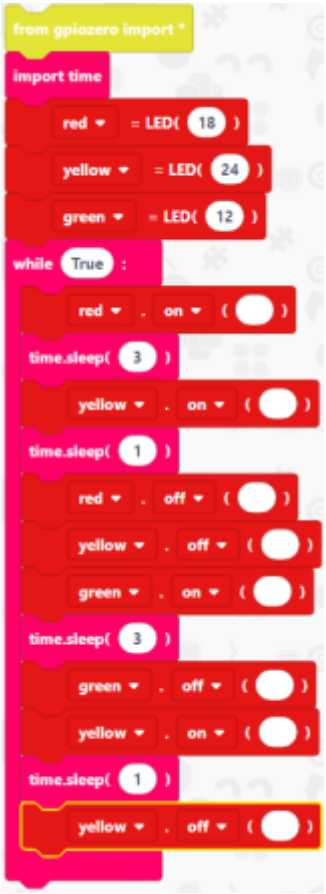
Your code should now look like this:

This turns the green LED off and the yellow LED back on for 1 second.

Turning the yellow LED off

Click on **LED,** click and drag an **led.on()** block to the code area and attach it under the **time.sleep(1)** block
Click on the small arrow next to **led** and click on **yellow**
Click on the small arrow next to **on** and click on **off**

Completed Code



Running the Code

To run your code click on **Run** on the toolbar above the coding area.

You should now see the LED's attached to your Raspberry Pi turn on and off in the sequence of taffic lights.

Well done you have made a set of Traffic lights by using your Raspberry Pi and code!