Workshop 1, Exercise 4

Gemma with one switch and a NeoPixel

Note. Gemma, NeoPixels, Adafruit are trademarks of Adafruit Industries. All rights acknowledged.

In this exercise you control a single NeoPixel from Gemma using an on/off switch to control light details such as 'flash red' or 'flash blue'. You learn how Gemma can read input as well as send output signals.

Activity 1. Upload the start code 'NeoSwitch_stage1'. Before doing that, make sure you have:

- a Gemma, a NeoPixel, a switch
- five croc clip leads (ideally 1 red, 1 black, 1 green, 2 yellow)

Here is the code to upload to Gemma. It should be on your workshop PC, and is also on our website.

```
#include <Adafruit_NeoPixel.h>
// 20 Mar 2014 DMc

// set variables with colours to use
uint32_t color1  = 0x00ff00, // green
         color2  = 0x0000ff;   // blue
// attach one leg of the switch to pin (D1), other to (3Vo)
int switchpin = 1;
// attach the NeoPixel control lead to pin (D0)
int neopin = 0;
// there is one NeoPixel, and it is controlled through D0 (i.e. neopin)
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(1, neopin);

void setup() {
  // tell Gemma to listen to input on D1
  pinMode (switchpin, INPUT);
  // following sets D1 to low - so Gemma will try to keep it LOW
  // the internal red LED is lit when this pin is HIGH
  // (if high it goes to earth through LED and resistor)
  // if switch is closed, it will power D1, and light the red LED
  digitalWrite(switchpin, LOW);
  // prepare NeoPixel(s)
  pixels.begin();
  pixels.setBrightness(25);
}

void loop() {
  // see if switch is closed or open then set NeoPixel colour
  if(digitalRead(switchpin)== HIGH){
    neoPattern(color1);}
  else {neoPattern(color2);}
  delay(100);
}

// flashes NeoPixel with set colour
void neoPattern(uint32_t acol) {
    pixels.setPixelColor(0, acol);
    pixels.show();
    delay(200);
    pixels.setPixelColor(0, 0);
    pixels.show();
    delay(200);
}
```
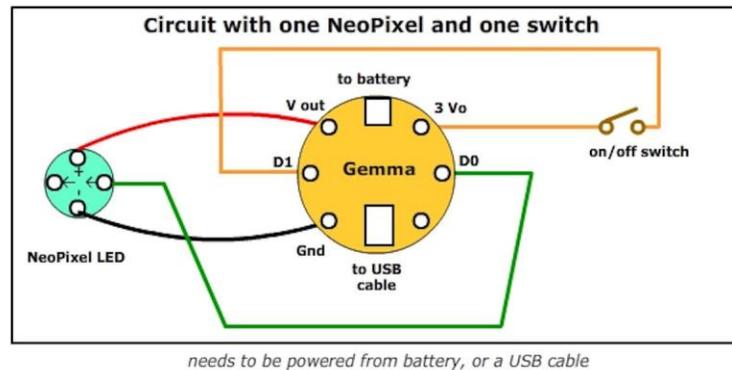
Activity 2. Construct and test your NeoSwitch circuit. Look at the diagram below, then use your crocodile clips to carefully make the circuit. If you have been successful, the NeoPixel should flash one colour when the switch is open, and a different one when it is closed.

Circuit with one NeoPixel and one switch

needs to be powered from battery, or a USB cable

When you have a working circuit, show the person who is running your workshop. They will ask you which parts of the code make the circuit work, so be sure to look back at it and work that out.

In particular, do you understand why the red LED is lit when it is, and why it goes off? How does that relate to the pins the switch is connected to?

Activity 3. Extend and improve the code. Save a copy of the code using your name, then see if you can achieve the following.

Idea 1. Easy. Change the NeoPixel colours, delays, brightness.

Idea 2. Easy. Add another NeoPixel or two, using another colour or two.

Idea 3. Harder. Have the switch make the first NeoPixel cycle through three colours.

Idea 4. Harder. Have the switch make all your NeoPixels go through many colour patterns.

Well done. You now know how you can write code, upload it, respond to input and control output. All you need now is a good project to get working on.