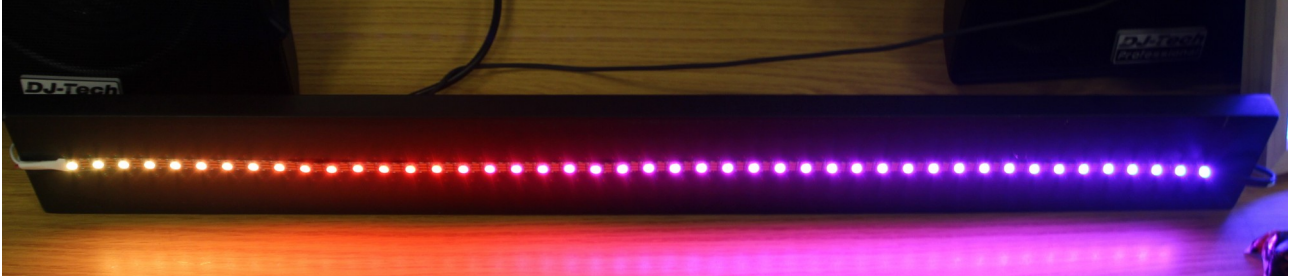


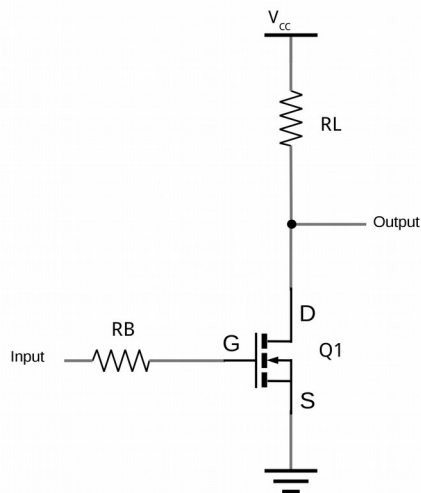
Getting Started with NeoPixels

NeoPixels are individually addressable multicoloured LEDs. In this exercise we will light up two breadboard NeoPixels. These are also available as rings, strings and as a matrix display.



To drive NeoPixels from a Raspberry Pi then we need to increase the output voltage from the GPIO port from 3.3V to 5V. This is achieved using a MOSFET as a voltage level shifter.

The circuit diagram is shown below:

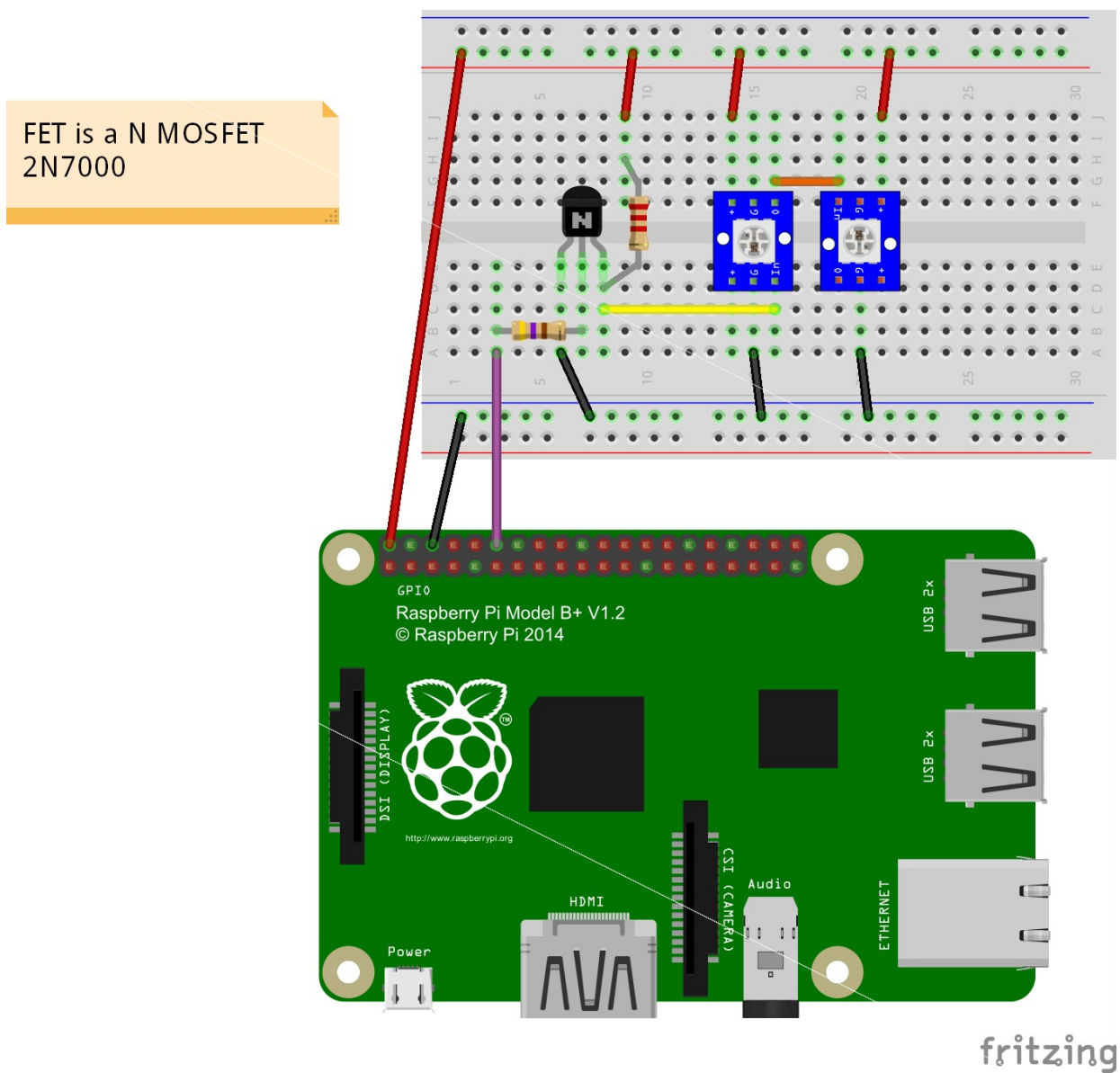


Here the MOSFET acts as a switch, which is turned on when the input is high and off when the input is low. The output is taken from above the MOSFET so when the MOSFET is switched on then the output goes low and when the MOSFET is switched off then the output is pulled high through resistor RL. This means that the voltage shifter is inverting (you get the opposite to the output compared with the input). This is something we can handle in the software.

The components used are as follows:

- RL – 2.2k Ω (Red, Red, Red, Gold)
- RB – 470 Ω (Yellow, Violet, Brown, Gold)
- Q1 - 2N70000

Using breadboard NeoPixels then the following circuit can be made. Note that this uses the power from the Raspberry Pi, which will work when using only one or two Neopixels, but an external supply is required if more NeoPixels are used.



Note that the two NeoPixels are mounted the opposite way around from each other. This is so that the output from the first is close to the input for the second NeoPixel.

We will be using a version of the Adafruit NeoPixel library that has been modified so that it works on all versions of the Raspberry Pi (the original AdaFruit version only works with the Raspberry Pi 1). The Python module is not included in the Raspberry Pi repositories and needs to be compiled from source code.

First we will need some developer libraries which will allow us to compile the software. This is installed using the normal Raspberry Pi installer as below:

```
..... sudo apt-get install build-essential python3-dev git scons swig
```

Next download the neopixel code from github using the clone command, which copies all the source code to your local computer.

```
..... git clone https://github.com/jgarff/rpi_ws281x.git
```

Change to that directory and run scons to compile the software.

```
..... cd rpi_ws281x
```

```
..... scons
```

We then need to change to the python directory and install the Python module from there:

```
..... cd python
```

Next install the Python 3 library file using

```
..... sudo python3 setup.py install
```

There are some test programs in that directory which you can try, but I have also written the following simple test script using Python 3.

The test script is provided below:

```
#!/usr/bin/python3
from neopixel import *
import time

LEDCOUNT = 2          # Number of LEDs
GPIOPIN = 18
FREQ = 800000
DMA = 5
INVERT = True        # Invert required when using inverting buffer
BRIGHTNESS = 255

strip = Adafruit_NeoPixel(LEDCOUNT, GPIOPIN, FREQ, DMA, INVERT, BRIGHTNESS)
# Intialize the library (must be called once before other functions).
strip.begin()

while True:
    # First LED white
    strip.setPixelColor(0, Color(255,255,255))
    strip.setPixelColor(1, Color(0,0,0))
    strip.show()
    time.sleep(0.5)
    # Second LED white
    strip.setPixelColor(0, Color(0,0,0))
    strip.setPixelColor(1, Color(255,255,255))
    strip.show()
    time.sleep(1)
    # LEDs Red
    strip.setPixelColor(0, Color(255,0,0))
    strip.setPixelColor(1, Color(255,0,0))
    strip.show()
    time.sleep(0.5)
    # LEDs Green
    strip.setPixelColor(0, Color(0,255,0))
    strip.setPixelColor(1, Color(0,255,0))
    strip.show()
```

```
time.sleep(0.5)
# LEDs Blue
strip.setPixelColor(0, Color(0,0,255))
strip.setPixelColor(1, Color(0,0,255))
strip.show()
time.sleep(1)
```

Save this as file neopixel.py
Set executable permission using:

```
..... chmod +x neopixel.py
```

It needs to be run with root permission using sudo

```
..... sudo ./neopixel.py
```

In the code the first part of the script sets up the appropriate values for the Raspberry Pi and the NeoPixels, this may need to be changed particularly if there are more NeoPixels than the two we have specified. Note that the INVERT option is set to True which is required due to the inverting nature of the circuit we have created.

A graphical user interface is available from:
<https://github.com/penguintutor/neopixel-gui>

For more details see:

<http://www.penguintutor.com/news/raspberrypi/discolights>

Follow: @stewartwatkiss or @penguintutor