Lab - 08
Leaving the GPIO Pins in a Safe State

Problem

You want all the GPIO pins to be set to inputs whenever your program exits so that there is less of a chance of an accidental short on the GPIO header, which could damage your Raspberry Pi.

Solution

Use a try: finally: construction and the GPIO.cleanup method.

The blink example can be rewritten to exit safely as shown below. The file for the code is called *led_blink_safe.py*.

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(18, GPIO.OUT)

try:
    while (True):
        GPIO.output(18, True)
        time.sleep(0.5)
        GPIO.output(18, False)
        time.sleep(0.5)
finally:
    print("Cleaning Up!")
    GPIO.cleanup()
```

Now, when you Ctrl-C the program to close it, GPIO.cleanup will be called before the program exits.

Discussion

If cleanup is not called or the Pi is not rebooted, then pins set to be outputs will remain as outputs after the program has finished. If you were to start wiring up a new project, unaware of this problem, your new circuit might accidentally short a GPIO output to one of the supply rails or another GPIO pin in the opposite state.

A typical scenario where this might happen would be if you were to connect a push switch, connecting a GPIO pin that you had configured as an output and HIGH to GND.

In summary, either be careful when swapping hardware or use GPIO.cleanup as shown earlier, or reboot your Pi. In any case, its a good idea to power down your Pi while you are connecting new hardware to it.