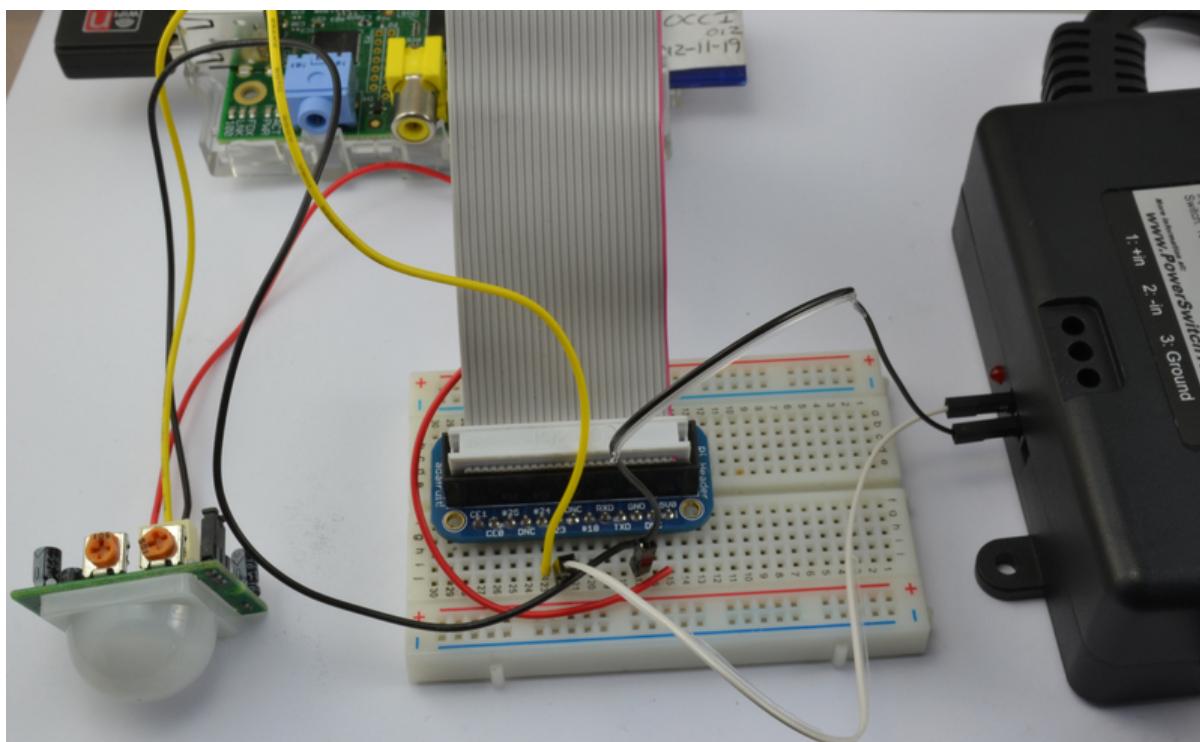




# Adafruit's Raspberry Pi Lesson 13. Power Control

Created by Simon Monk



<https://learn.adafruit.com/adafruits-raspberry-pi-lesson-13-power-control>

Last updated on 2024-06-03 01:15:45 PM EDT

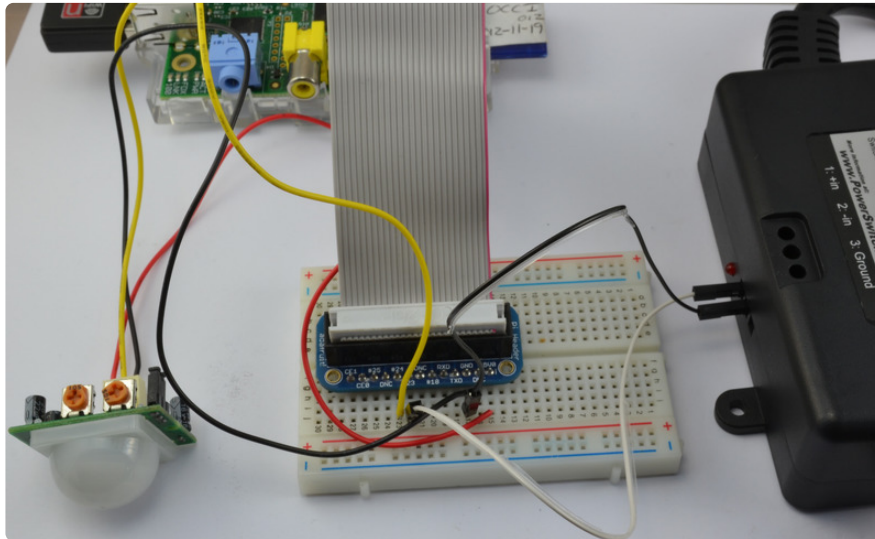
# Table of Contents

<a href="#">Overview</a>	<a href="#">3</a>
<hr/>	
<a href="#">Hardware</a>	<a href="#">3</a>
<hr/>	
<ul style="list-style-type: none"><li>• <a href="#">Power Tail 2</a></li><li>• <a href="#">Controllable Four Outlet Power Relay Module Version 2</a></li><li>• <a href="#">40-Pin (A, B, B+ and Zero) Cobbler Plus Schematic</a></li><li>• <a href="#">20-Pin (Raspberry Pi Rev 1 and Rev 2) Cobbler Schematic</a></li></ul>	
<a href="#">Software</a>	<a href="#">6</a>
<hr/>	
<ul style="list-style-type: none"><li>• <a href="#">The Setup</a></li><li>• <a href="#">The Code</a></li></ul>	
<a href="#">Configure and Test</a>	<a href="#">7</a>
<hr/>	
<ul style="list-style-type: none"><li>• <a href="#">Get the Code</a></li><li>• <a href="#">Running the Code</a></li></ul>	

---

# Overview

The Raspberry Pi can make a great home automation controller. But to be really useful, it needs to be able to turn electrical appliances and lights on and off. That means it needs to be able to control 110V safely.



In this lesson, you will combine the PIR sensor from [Lesson 12 \(https://adafru.it/aZ3\)](https://adafru.it/aZ3) with the Powerswitch Tail 2 module from Adafruit, to automatically switch something on when movement is detected.

---

## Hardware

If you have 220V mains, this tutorial wont work for you - If we knew of an equivalent 220V controller like the PST we would have a link to it here

### Power Tail 2

This tutorial works with all versions of Raspberry Pi (rev 1, 2, A, B, B+ and Zero) except the compute module which has no headers.

The Powerswitch Tail 2 from Adafruit, looks like a 110V electrical lead with a box in the middle.

The box is actually an opto-isolated solid-state relay. The opto-ilsolation bit means that there is no actual electrical connection between the low voltage switching circuit and the 100V mains. This makes it extremely safe and removes any chance of accidentally frying your Pi.

What is more the control input consumes just 3 mA of current at 3.3V, which means we can control it directly from one of the Pi's output pins.

The PIR sensor is connected to a different pin from lesson 12, so be sure to move the yellow lead.

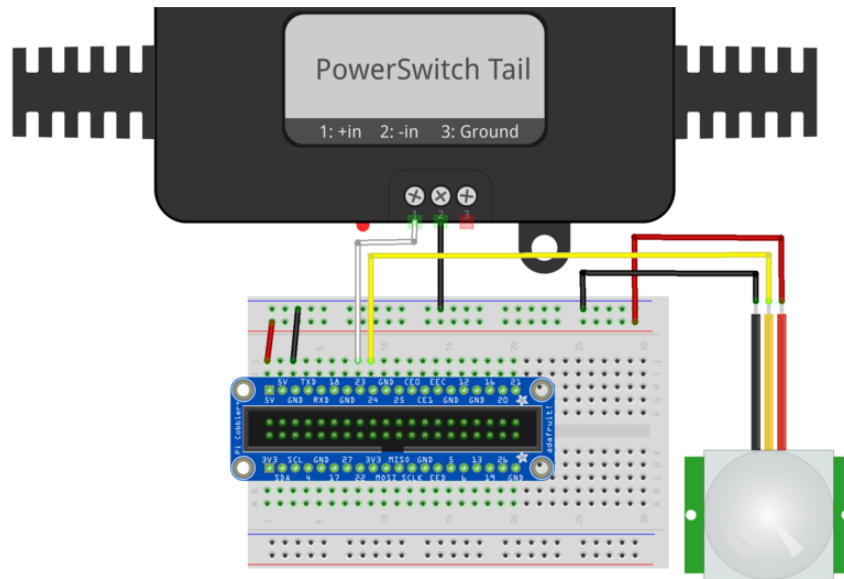


## Controllable Four Outlet Power Relay Module Version 2

The [Controllable Four Outlet Power Relay Module Version 2](http://adafru.it/2935) (<http://adafru.it/2935>) can be used as an alternative for this project.

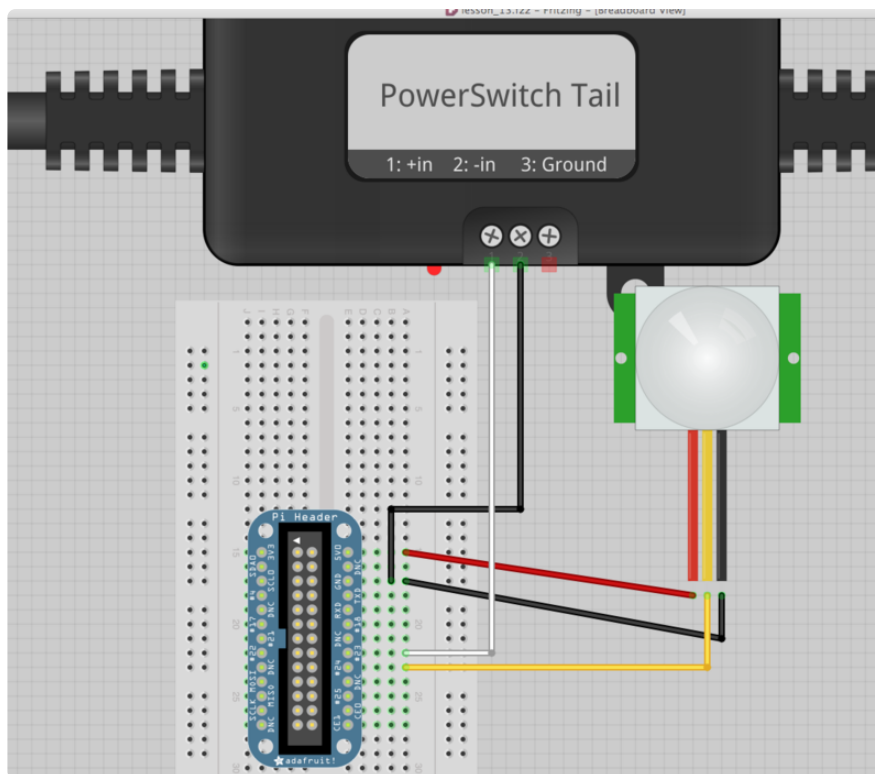


## 40-Pin (A, B, B+ and Zero) Cobbler Plus Schematic



## 20-Pin (Raspberry Pi Rev 1 and Rev 2) Cobbler Schematic

The Powerswitch Tail comes with a little indicator LED that shows when it is on, so you do not need to connect anything high voltage to it yet.



---

# Software

## The Setup

We are using the CircuitPython Libraries that are part of adafruit-blinka. See [CircuitPython Libraries on Raspberry Pi \(https://adafru.it/Deo\)](https://adafru.it/Deo) to get a fresh Raspberry Pi setup.

If you have a running Raspberry Pi with an up to date copy of Raspbian you can simply run the following command to install adafruit-blinka.

The software is exactly the same on all 40-pin and 20-pin Raspberry Pi models.

```
$ sudo pip3 install adafruit-blinka
```

## The Code

This project is possible the most over-engineered automatic light switch ever created. You really do not need a Raspberry Pi to turn on the Powerswitch, but the example can easily be adapted for other purposes. For example, you could use a combination of temperature, humidity, light, and perhaps some internet data on weather forecasts to turn on or off a heater, fan or humidifier.

```
# SPDX-FileCopyrightText: 2019 Mikey Sklar for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import time
import board
from digitalio import DigitalInOut, Direction

pir_pin = board.D24
power_pin = board.D23

pir = DigitalInOut(pir_pin)
pir.direction = Direction.INPUT

power = DigitalInOut(power_pin)
power.direction = Direction.OUTPUT
power.value = False

while True:
    if pir.value:
        print("POWER ON")
        power.value = True
        time.sleep(20)
        print("POWER OFF")
        power.value = False
        time.sleep(5)
    time.sleep(1)
```

The program first sets up the two GPIO pins that are used, one as an input to the PIR sensor and one as an output for the Powerswitch Tail.

The main loop then just waits for the PIR sensor to sense movement and then prints a message and turns on the Powerswitch tail, waits 20 seconds then turns it off again.

It the output was turned on, it waits for 5 seconds to prevent immediate retriggering.

There is also always a 1 seconds delay each time around the loop.

---

## Configure and Test

### Get the Code

Let's put this file right in your home directory for simplicity. The wget command makes things easy as this can be run from an internet connected Raspberry Pi.

```
$ wget https://raw.githubusercontent.com/adafruit/Adafruit_Learning_System_Guides/master/Raspberry_Pi_Power_Control/Raspberry_Pi_Power_Control.py
```

Do not write code that turns the power on and off very rapidly. Many kinds of appliance, including lamps will be damaged by high frequency switching. This type of setup should turn things on for a few seconds minimum.

### Running the Code

Confirm that the Pi connected to the Cobbler (or Cobbler Plus) and run the script.

To start with, cover the PIR sensor with something, so that it is not activated until you are ready.

The following command will start the program:

```
$ sudo python3 ./Raspberry_Pi_Stepper_Motors.py
```

Uncover the PIR sensor and you should see a message saying 'POWER ON' and the little LED on the Powerswitch Tail should light, then turn itself off after 20 seconds.

Now this is all working, you can try attaching a lamp or some electrical appliance to the Powerswitch Tail.

```
pi@pi3b:Raspberry_Pi_Power_Control $ sudo python3 ./Raspberry_Pi_Power_Control.py
POWER ON
POWER OFF
POWER ON
POWER OFF
```