

Getting Started with the Raspberry Pi

Let's install our first program!

pi@raspberrypi:~ \$ sudo apt install speedtest-cli

pi@raspberrypi:~ \$ speedtest-cli





Throw in the flag!

-a, --all do not ignore entries starting with .

-I use a long listing format

-h, --human-readable with -l and -s, print sizes like 1K 234M 2G etc.

pi@raspberrypi:~ \$ Is --help

pi@raspberrypi:~ \$ man ls



Updating WiFi



The configuration file for WiFi APs is titled: wpa_supplicant.conf

The config file is in a folder titled: wpa_supplicant

That folder is located in the directory: **etc**

pi@raspberrypi:~ \$ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf



Shutdown / Reboot

pi@raspberrypi:~ \$ sudo reboot

pi@raspberrypi:~ \$ sudo shutdown -h now



General Purpose Input Output

40 Pin 'Header'

Power / Ground / Specialty Pins

Physical vs BCM Numbering

Digital vs Analog

ON/OFF...1/0...True/False

Logic Level Voltage

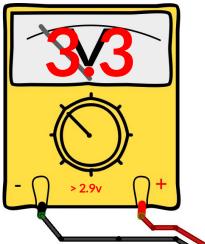
- 3.3v vs 5v
- ON/TRUE=3.3v OFF/FALSE=0v



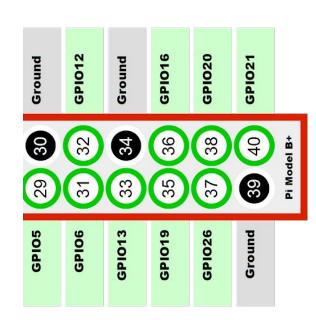




GPIO as **OUTPUT**



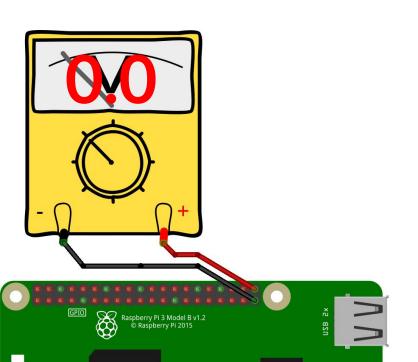
Programmatically we assign GPIO pin 21 (40) as an OUTPUT

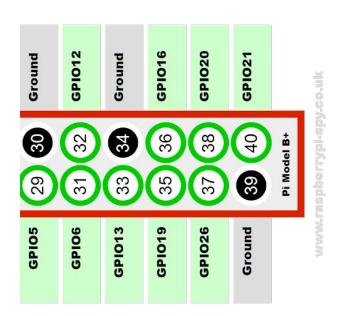


And can then set it to "ON"



GPIO as **OUTPUT**



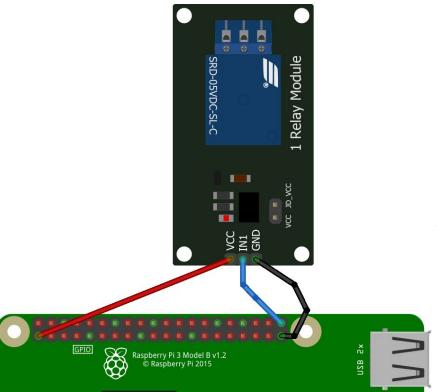


...and then back to "OFF"



NZB 5×

What can we do with 3.3 volts?



And <50mA!



GPIO as **INPUT**



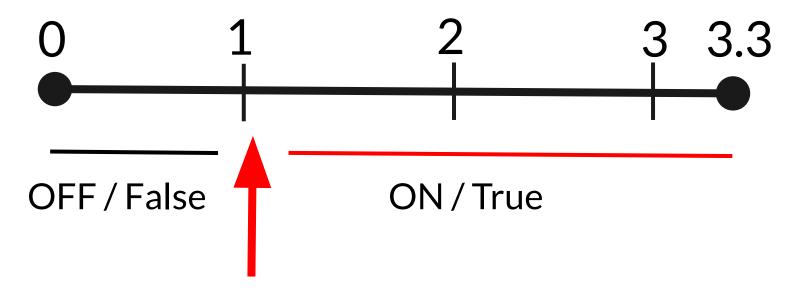
With a button connected across the 3.3v power and GPIO 21 (40):

- Pressing the button, GPIO 21 reads 3.3v or ON / TRUE
- Releasing the button, GPIO 21 reads 0v or OFF / FALSE



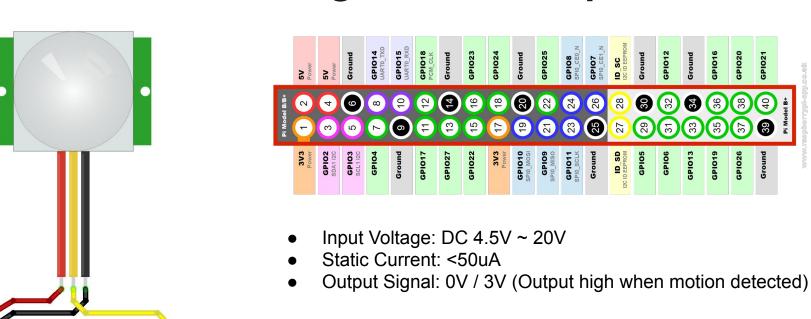


VOLTS





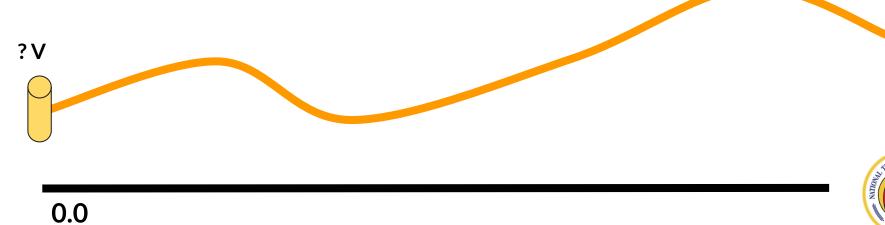
How else can we get 3.3v for input?





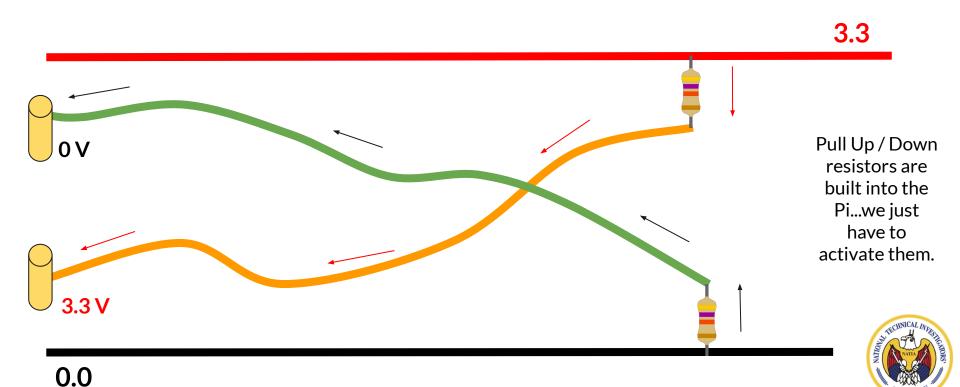
Floating Voltage

3.3





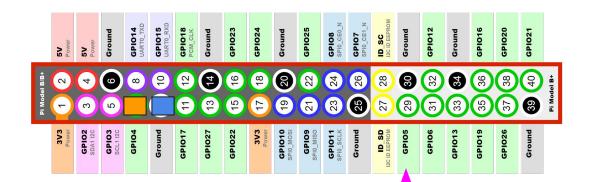
Floating Voltage - Pull Up / Down Resistor



Sensor with High Trigger

Output: Digital pulse **high (3V)** when triggered (motion detected) digital low when idle (no motion detected)



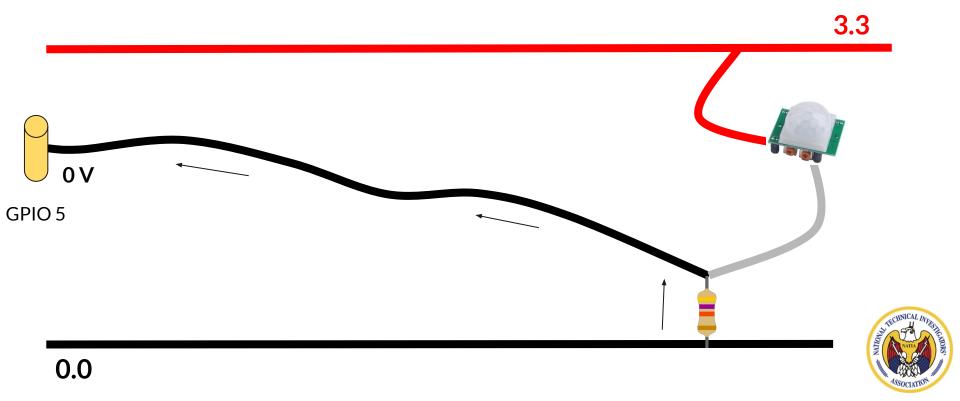


When we have a sensor that goes HIGH (or UP) when triggered connected to a GPIO Pin, we want to pull that GPIO Pin DOWN (or low) so that we can detect when the sensor goes HIGH.

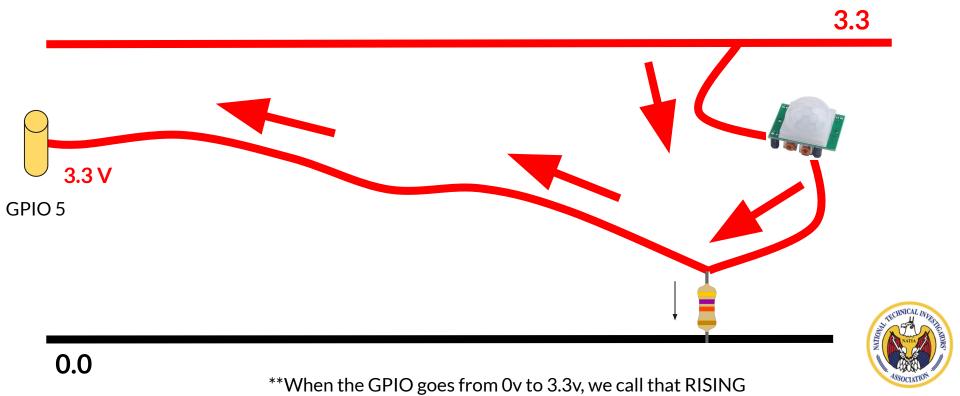
If we pulled it UP to 3.3v, we would not be able to detect when the sensor sent 3.3v because we're already reading 3.3v. We won't see the change.



Floating Voltage - Pull Down Resistor



Floating Voltage - Pull Down Resistor



Switch Between Ground and Pin

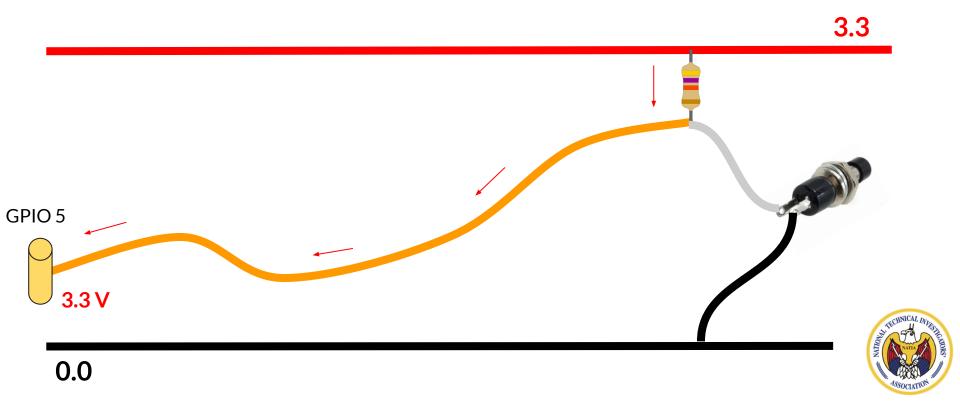
 If we pulled it DOWN to 0v, we would not be able to detect when the button was pressed because we're already reading 0v. We won't see the change.



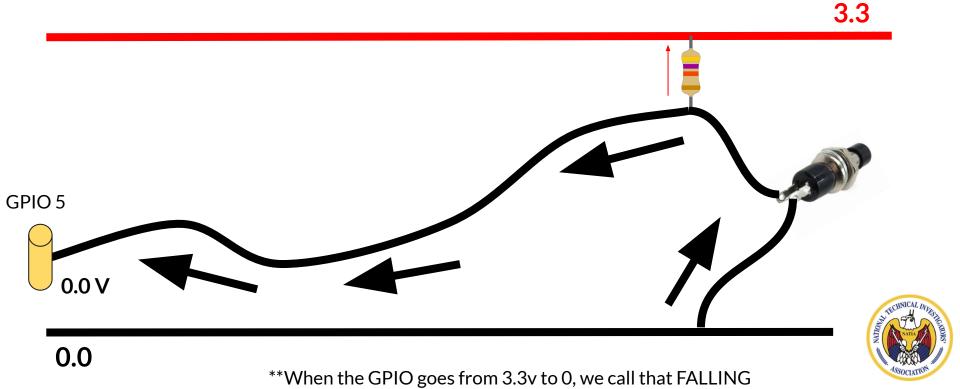
When we have a switch connecting a GPIO Pin and a Ground Pin, we want to pull that GPIO Pin UP (or HIGH) so that we can detect when the switch closes and that pin becomes connected to Ground



Floating Voltage - Pull Up Resistor



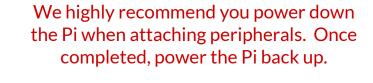
Floating Voltage - Pull Up Resistor

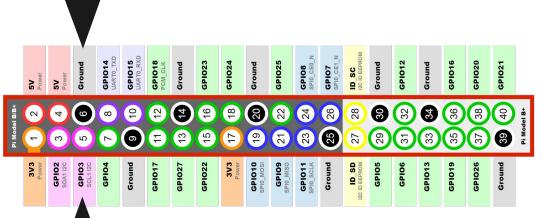


Power Button



Using one of the pre wired momentary push button switches, attach one lead to pin 5 (GPIO3) and one lead to pin 6







Github

pi@mypi:~ \$ sudo apt install RPi.GPIO

pi@mypi:~ \$ cd /usr/local/bin

pi@mypi:/usr/local/bin \$ sudo wget

https://github.com/corbitt799/natia/archive/master.zip

pi@mypi:/usr/local/bin \$ sudo unzip master.zip

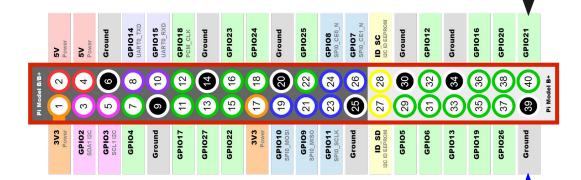
pi@mypi:/usr/local/bin \$ sudo mv natia-master natia

pi@mypi:/usr/local/bin \$./natia/power_button/install

pi@mypi:/usr/local/bin \$ sudo reboot



LED Blink



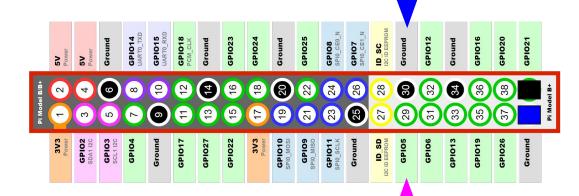
Using one of the pre wired LEDs, attach the BLACK lead to pin 40 (GPIO 21) and the BLUE or YELLOW lead to pin 39



Does polarity matter?



Doorbell (1)



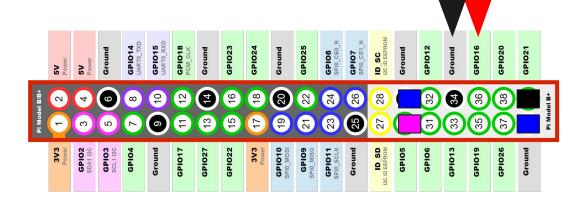
Does polarity matter?

Leave LED attached

Using one of the pre wired momentary push button switches, attach one lead to pin 29 (GPIO 5) and one lead to pin 30



Doorbell (2)





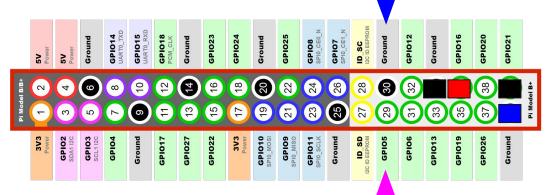
Using the buzzer, attach the RED lead to pin 36 (GPIO 16) and the BLACK lead to pin 34

Using one of the pre wired momentary push button switches, attach one lead to pin 29 (GPIO 5) and one lead to pin 30

Does polarity matter?



Door Open / Relay (1)



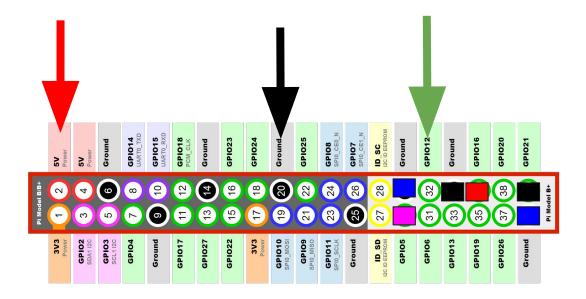


Using the pre wired door contacts, attach one lead to pin 29 (GPIO 5) and one lead to pin 30 (Replacing push button)





Door Open / Relay (2)





INPUT CONNECTION:

DC +: Positive power supply (VCC)

DC-: Connect power negative (GND)

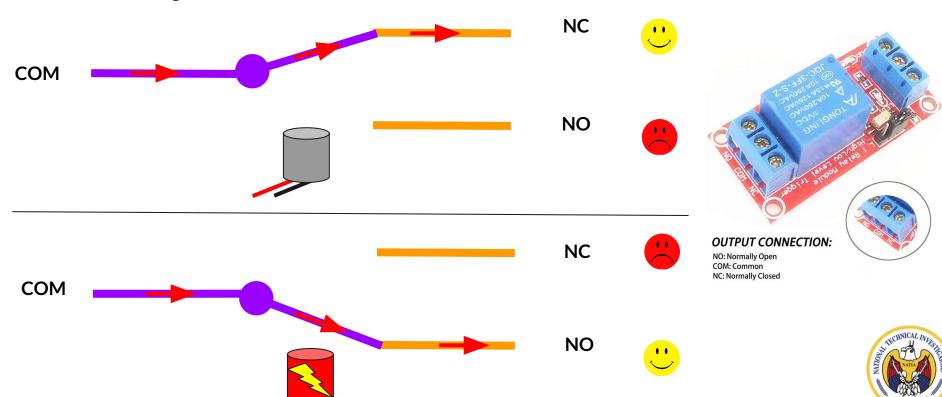
IN: Control the pick up of replay by low level or high level

Using the relay:

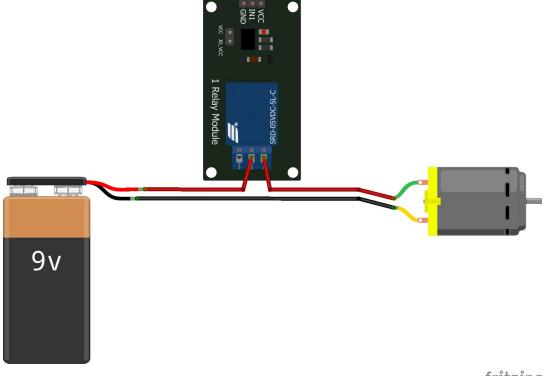
- GRD to pin 20
- High/Low to pin 32 (GPIO 12)
- +Power to pin 2



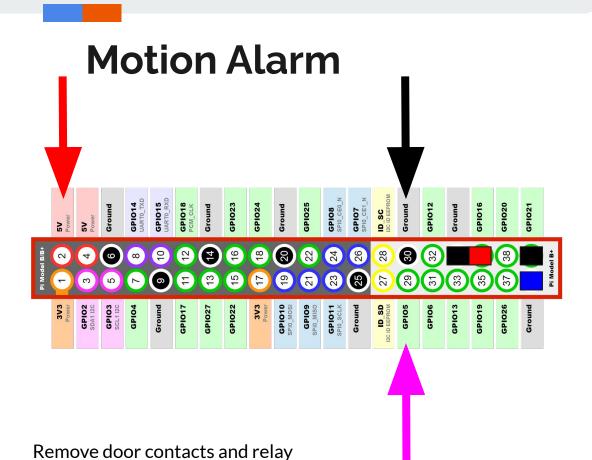
Relays

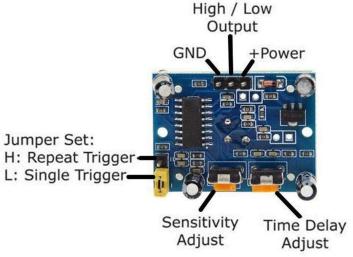


Relays 2









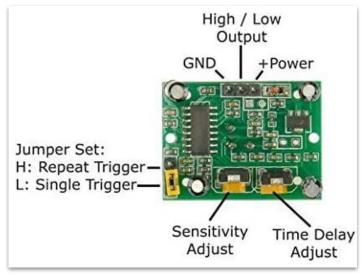
Using the motion sensor:

- GRD to pin 30
- High/Low to pin 29 (GPIO 5)
- +Power to pin 2



Motion Sensor Tuning



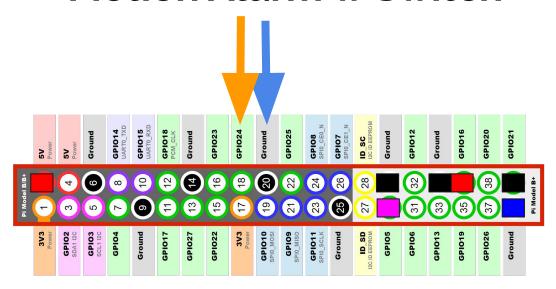


How long the output trigger remains HIGH

Single: Timer begins with initial motion Repeat: Output remains HIGH as long as there's motion



Motion Alarm w Switch



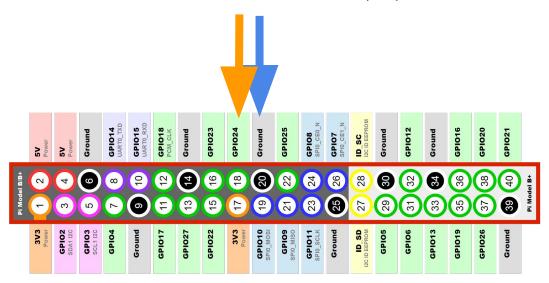


Using the pre wired toggle switch, attach one lead to pin 18 (GPIO 24) and one lead to pin 20

Does polarity matter?



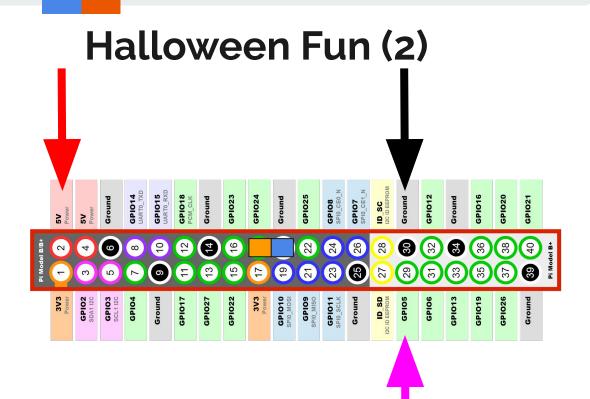
Halloween Fun (1)

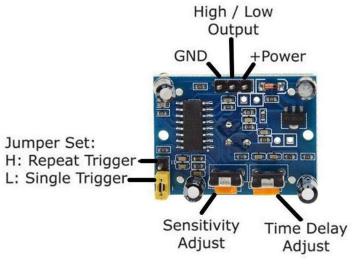




Using the pre wired toggle switch, attach one lead to pin 18 (GPIO 24) and one lead to pin 20





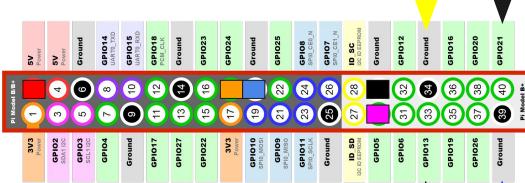


Using the motion sensor:

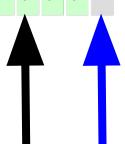
- GRD to pin 30
- High/Low to pin 29 (GPIO 5)
- +Power to pin 2



Halloween Fun (3)



Attach one LED's BLACK lead to pin 40 (GPIO 21) and BLUE/YELLOW lead to pin 39 and the other LED's BLACK lead to ping 33 (GPIO 13) and the BLUE/YELLOW lead to pins 34







Encrypt WPA Passphrase

wpa_passphrase YOUR_SSID YOUR_PASSWORD

```
Sample output:
```

```
network={
    ssid="YOUR_SSID"

    #psk="YOUR_PASSWORD"

    psk=6a24edf1592aec4465271b7dcd204601b6e78df3186ce1a62a31f40ae9630702
}
```

