[Tiny] Home IOT

Part 2 – Security & Air Quality

Part 1 – Lights & Ambient Conditions

Table of Contents

Introduction

Today we will be building a smart home in miniature. We will be installing all the real-world components onto tiny homes. When you get home today you can scale up instantly to your real-world home.

Components

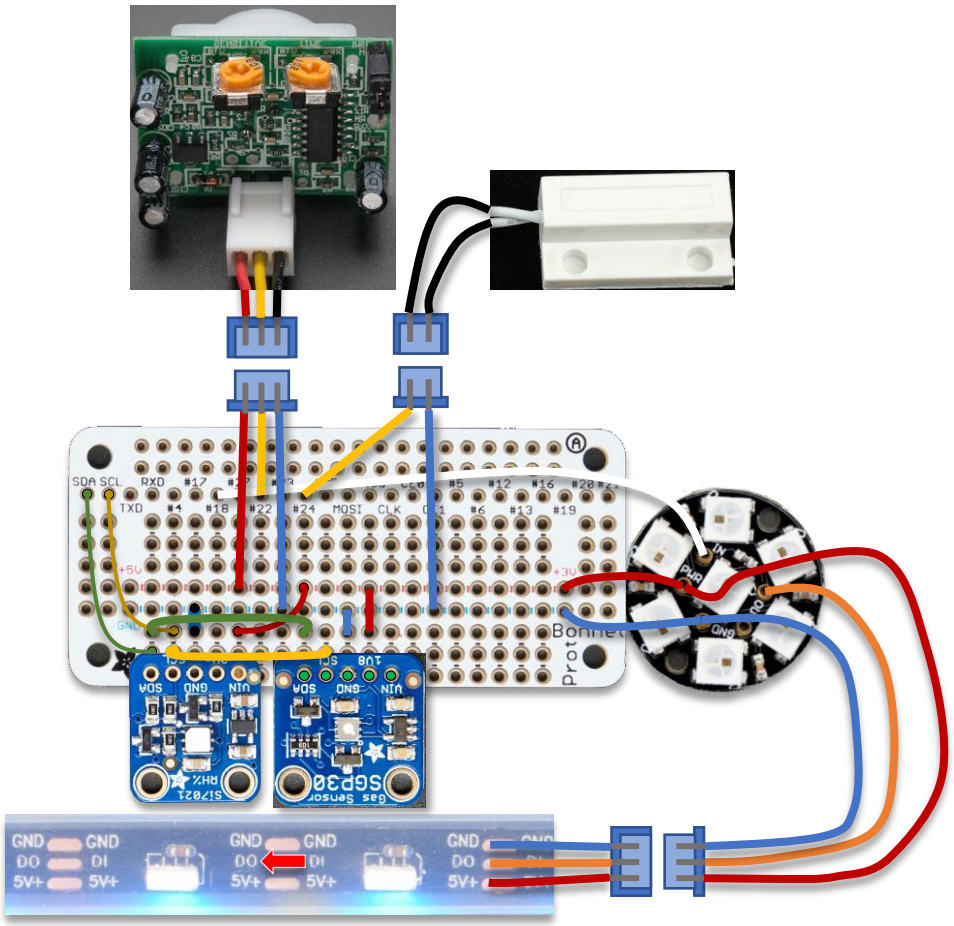
We will be working with the following components:

1. Things to monitor / measure / control
    a. Door Sensor - Magnetic Contact Switch
    b. Motion – PIR (Motion) Sensor
    c. Carbon Dioxide – SGP30 Air Quality Sensor
    d. Volatile Organic Compounds - SGP30 Air Quality Sensor
    e. Alarm System LED light device [indoor] – NeoPixel Jewel with 7 addressable RGB LEDs
2. Controllers & Hardware
    a. Raspberry Pi Zero W – WiFi controller for everything
    b. Raspberry Pi Zero Prototype “Bonnnet” – Prototype board to organize connections
    c. JST connectors – allows you to quickly connect or disconnect components
3. Other
    a. Wire & soldering wire
    b. Peripherals for your Pi – mouse, keyboard, monitor, 5V 2A power supply
    c. A tiny house

Assembly Instructions

The instructions here assume you’ve already completed Part 1 of the [Tiny] IOT Home series.

Wiring Diagram of the Finished Product – With Part 1

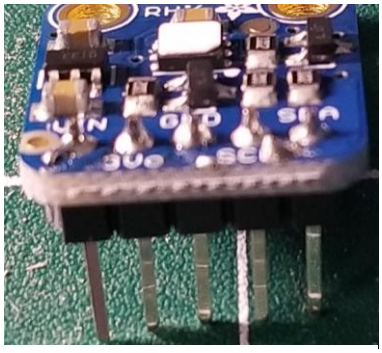


Step by Step Instructions

1. Solder the header onto your SGP30 Air Quality sensor
    a. The top of the board says SGP30 and has the components mounted to it
    b. The bottom of the board is relatively flat and has the Adafruit Logo screen printed on it.
    c. Break the header pins using needle nose pliers, so you have 5 pins in the group.
    d. Insert the short pins from the bottom of the board (logo) up and solder from the top.

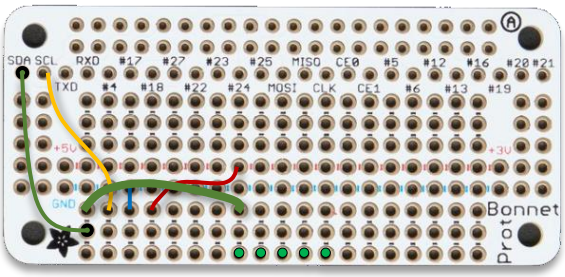
Top of SGP30

Bottom of SGP30

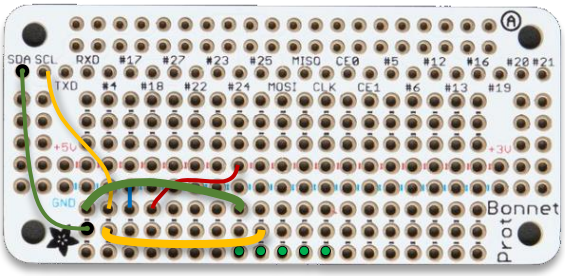


SGP30 w/Header

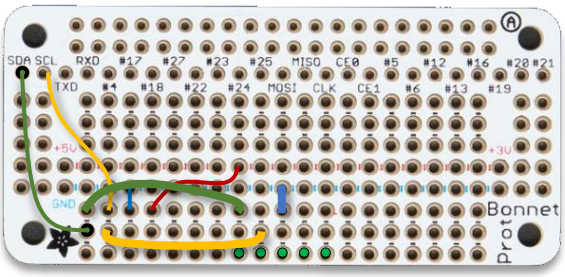
2. Set up for the second sensor
    a. Solder a short green wire from the **SDA** column (the top hole on the left most column; with a green wire already attached) on the bonnet to the top hole of the 8th from the left column of holes (3 holes vertically to a column) at the bottom of the board.



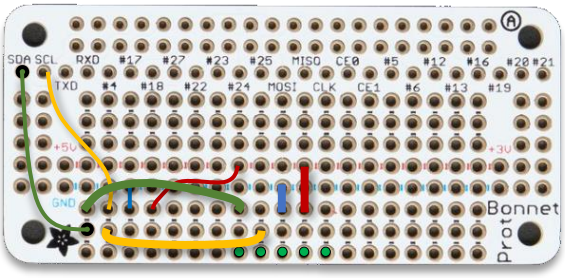
    b. Solder a short yellow wire from the **SCL** middle hole on the second column from the left of the bonnet to the middle hole of the 9th (from left) column of holes (3 to a column) at the bottom of the board.



c. Solder a short blue wire from the ground holes (strip of holes across the middle of the board with blue paint connecting them) to the top hole of the 10th (from left) column of holes (3 to a column) at the bottom of the board.



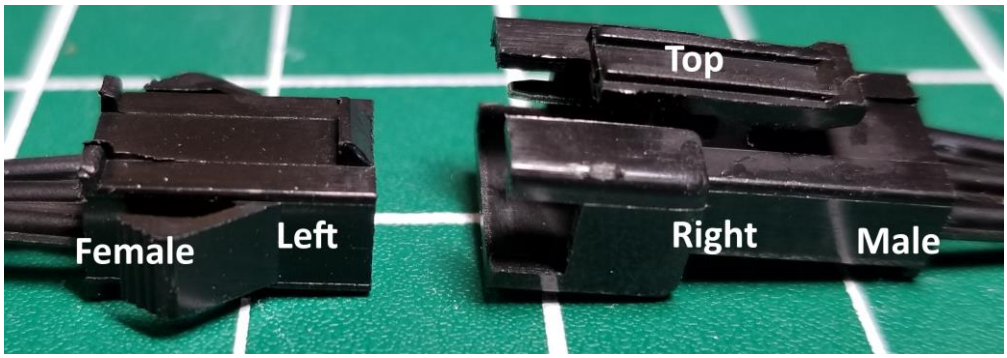
d. Solder a red wire from the left most **3V power** holes (strip of holes across the right side of the middle of the board with red paint connecting them) to the top hole of the 11th (from left) column of holes (3 to a column) at the bottom of the board.



3. Attach the Air Quality sensor
   a. Insert the pins on your sensor into the bottom holes on your bonnet starting at the 8th hole (from the left).
      i. **SDA** pin is on the left and matches the green wire,
      ii. Ground (**GND**) pin in the middle matches the blue wire,
      iii. And the **1V8** pin aligns with the red wire.

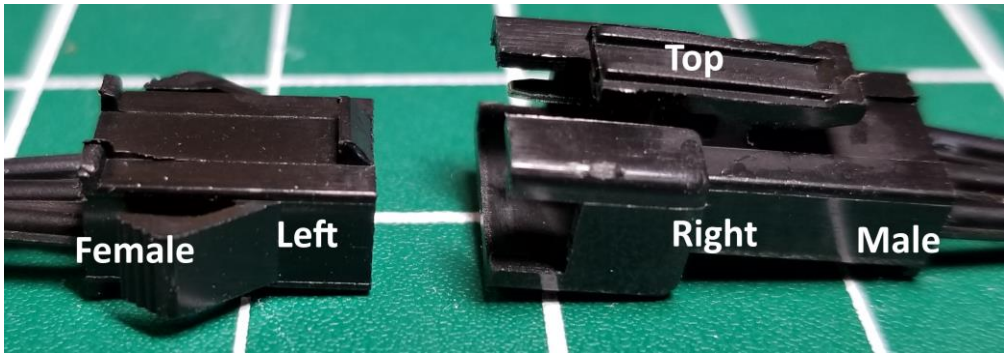
   b. Solder the pins onto your bonnet from the **bottom** of the bonnet.
4. Place the **JST 3 pin connector** on the table and identify both sides of the connector
   a. The ***'right'*** side has the clip to hold the two pieces together.

b. The ***'left'*** side doesn't have a clip but does have a small bump to capture the clip.
c. The ***'top'*** is the part with the clip or the bump for the clip to clamp to on the other side of the connector.



d. Place the connector on the table with the clip up and on the right side.
    i. The left side will go to your PIR motion sensor.
    ii. The right side will go to your Raspberry Pi.
e. From the right (male) connector, solder each of the following connections
    i. The red wire will be power:
        1. Solder to a **5V power** hole on the red strip of holes below hole # 22 on your bonnet.
        2. The 5V power holes are on the left of the bonnet.
        3. Insert from the top (with the writing) and solder from bottom.
    ii. The middle green wire will be signal/data:
        1. Solder to hole # 22 on your bonnet.
        2. Insert from the top (with the writing) and solder from bottom.
    iii. The black wire will be ground:
        1. Solder to a **GRND** hole in the blue row below # 22 on your bonnet.
        2. Insert from the top (with the writing) and solder from bottom.

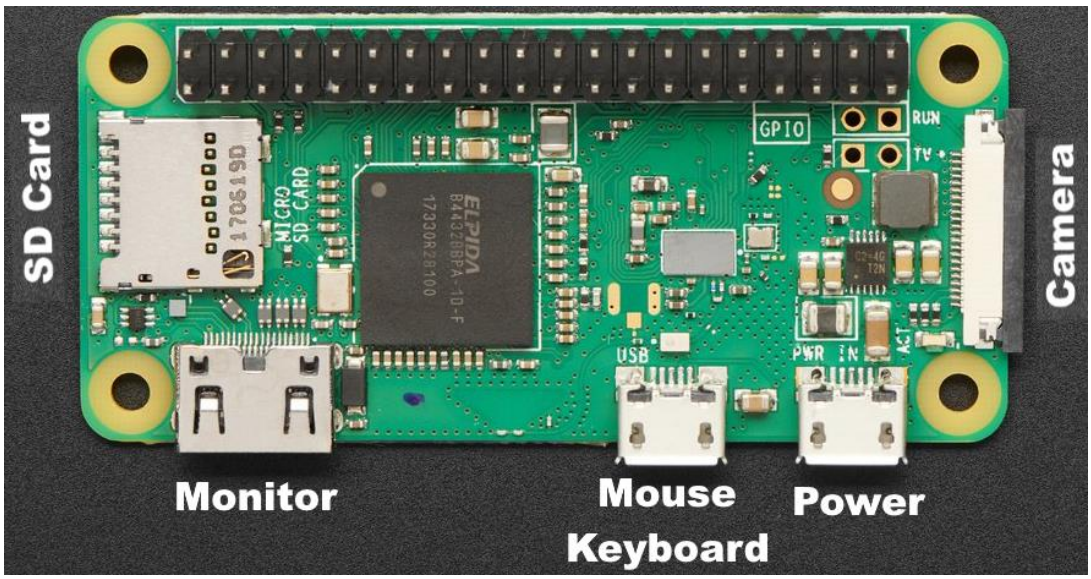
5. Connect the PIR Motion Sensor
    a. On the LEFT (female) JST connector wires. Solder each of the following wired connections.
        iv. The red wire will be connected to the PIR red wire.
        v. The middle wire will connect to the middle (yellow data) wire from the PIR.
        vi. and the black wire will be connected to the PIR black ground wire.
    f. Add a piece of electrical tape and a piece of heat shrink over the soldered ends to insulate them.
6. Place the **JST 2 pin connector** on the table and identify both sides of the connector
    g. The ***'right'*** side has the clip to hold the two pieces together.
    h. The ***'left'*** side doesn't have a clip but does have a small bump to capture the clip.
    i. The ***'top'*** is the part with the clip or the bump for the clip to clamp to on the other side of the connector.



j. Place the connector on the table with the clip up and on the right side.
    i. The right side will go to your Raspberry Pi.
    ii. The left side will go to your Door Switch sensor
k. From the right (male) connector, solder each of the following connections
    i. The bottom wire (closest to you) will be the signal in:
        1. Solder to hole # 24 on your bonnet.
        2. Insert from the top (with the writing) and solder from bottom.
    ii. The top wire (farthest from you) will be ground:
        1. Solder to a **GRND** hole in the blue row below # 24 on your bonnet.
        2. Insert from the top (with the writing) and solder from bottom.

7. Connect the Door Switch Sensor
    a. On the LEFT (female) JST connector wires. Solder each of the wires to each one of the white wire connections on the switch. It doesn't matter which wire is attached to which.
    b. Add a piece of electrical tape and a piece of heat shrink over the soldered ends to insulate them.

8. Finish assembly
    a. Carefully screw the door sensor onto the outside of the house.
        i. Put the half with the wires on the door frame (your right) and thread the wires through the slot above the screw pilot holes.
        ii. Put the un-wired half on the door using the screw pilot holes.
    b. Insert the PIR Motion Detector into one of the round holes at the top of the house.
        i. Insert the PIR motion wire onto the PIR board, making sure you line up the red wire with the **5V** pin!
    c. Carefully insert the prototype bonnet into the female header on the Pi Zero W.
    d. Connect the JST connectors (2 pin and 3 pin) halves together.
    e. Place the entire Pi assembly into your tiny home.
    f. Hook up the wires for your PI's HDMI, power, and mouse/keyboard via the rectangular hole in the bottom-right (when viewed from the front) of the back wall.
        i. See Connecting the PI below for information on how to connect your Pi.
        ii. Be sure you have a pre-configured SD card inserted into your PI as well. https://www.raspberrypi.org/documentation/installation/installing-images/



g. [Optional] Connect the 3 pin JST connector for your strip lights
h. Place the roof on the house.

Congratulations! You have built your Pi Zero W home controller and tiny home.

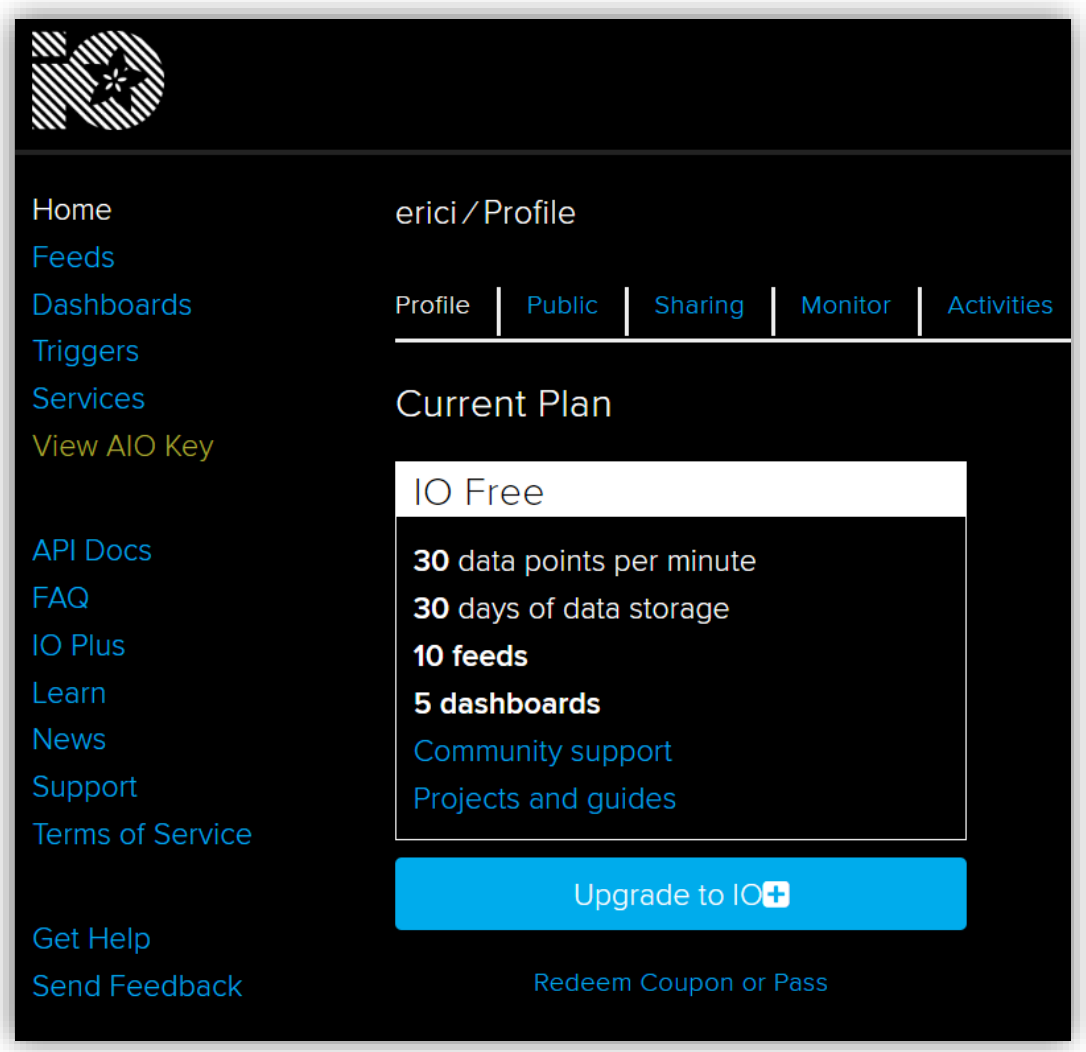
Setup Adafruit IO

For more information on Adafruit IO see the introduction:

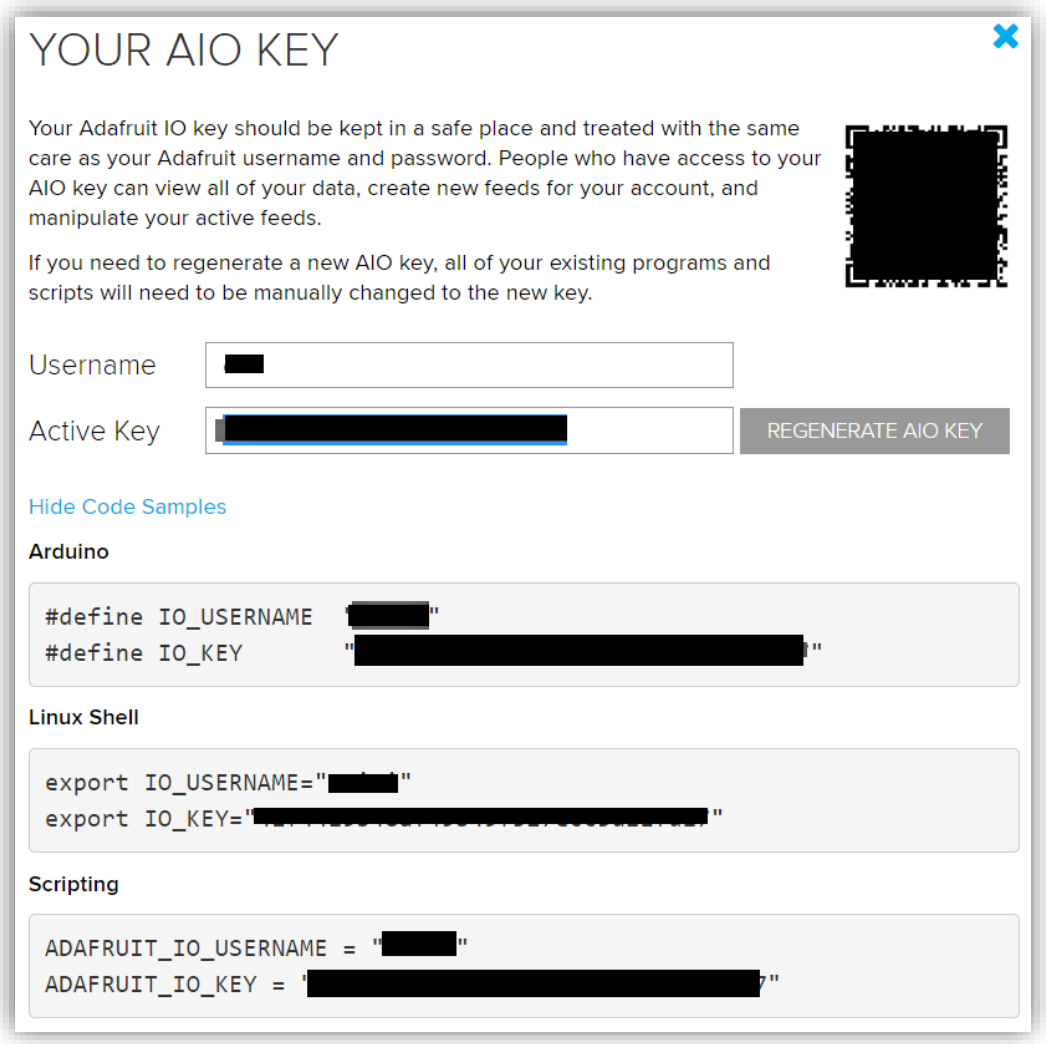
https://learn.adafruit.com/welcome-to-adafruit-io?view=all

[Optional] Create Your Account

1. Open a browser tab and go to https://io.adafruit.com/
2. If you have an account, go ahead and sign in.
    a. If you don't have an account, click the Get Started for Free and create an account.
3. Open your profile page at https://io.adafruit.com/erici/profile



4. Press the **View AIO Key** menu item on the left.



5. Notice the following (you will be using the 'Scripting' option)
    a. ADAFRUIT_IO_KEY
    b. ADAFRUIT_IO_USERNAME

Set Up Feeds

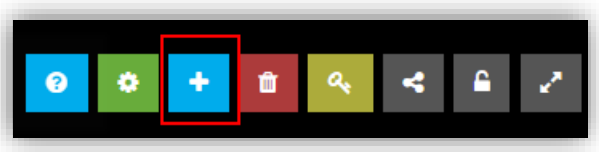
Feeds store the data online for you. Each feed represents a specific variable:
https://learn.adafruit.com/adafruit-io-basics-feeds

1. Navigate to the **IO Feeds** page.
2. Click **Actions -> Create a New Feed**
    a. Name the new feed **tvoc**
    b. Click the **Create** button
3. Repeat for the following feeds
    a. eco2
    b. front-door
    c. motion-detector

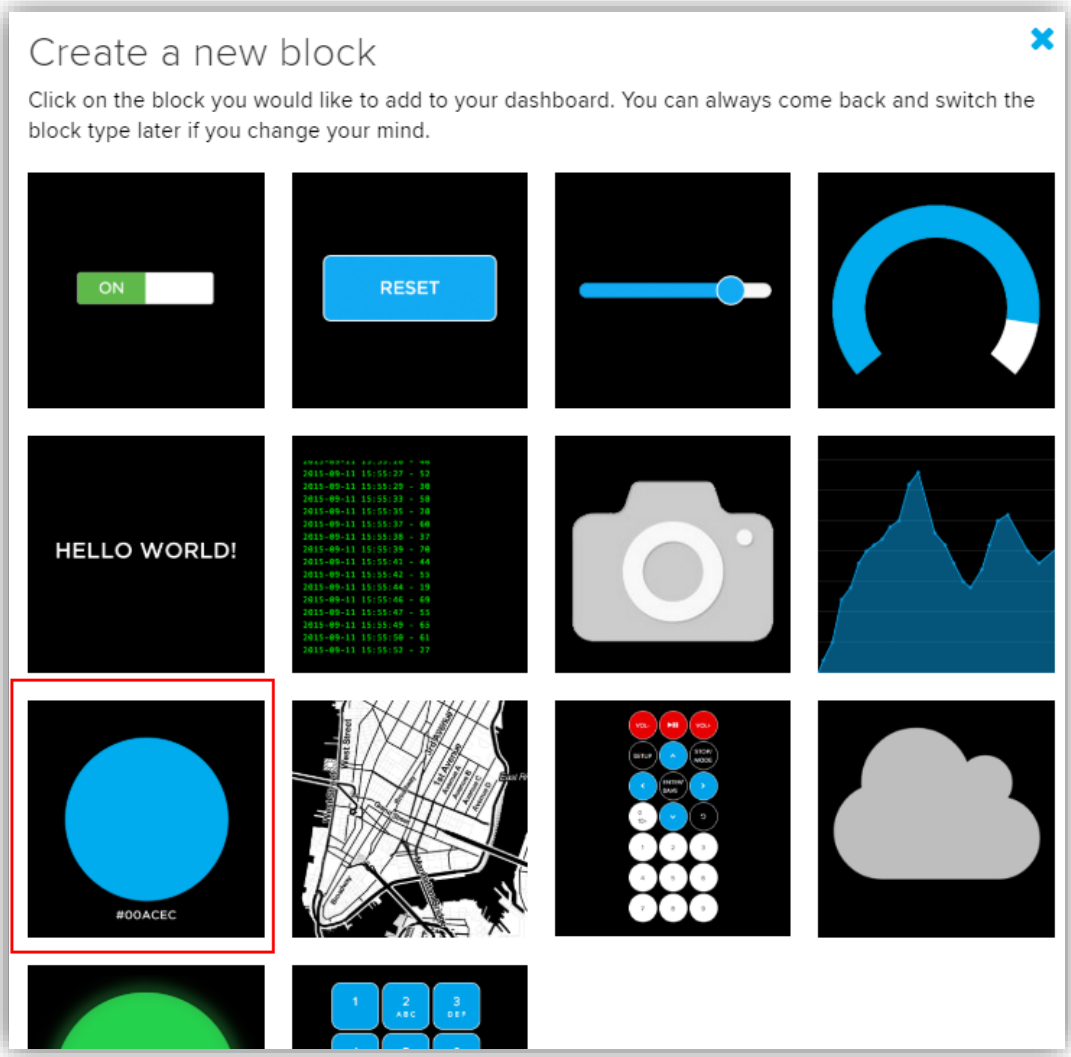
d. **home-alarm**
e. **indoor-lights**
f. **outdoor-lights**
g. **picam**

Create a Dashboard

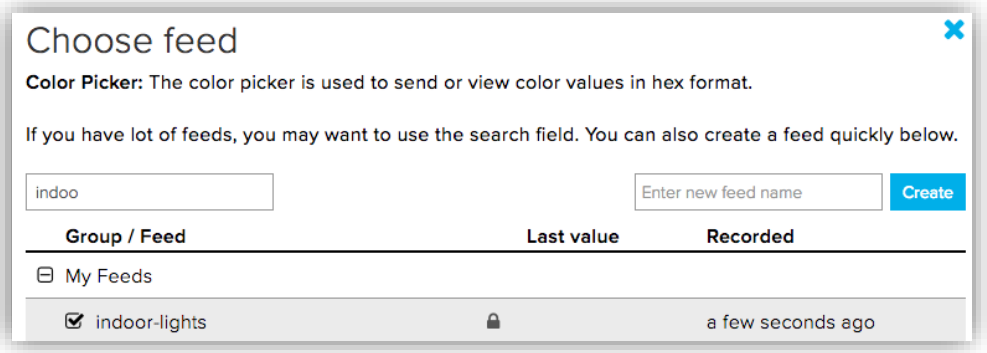
1. Navigate to the Adafruit IO Dashboard page: https://io.adafruit.com/<username>/dashboards
2. Click **Actions > Create a New Dashboard** (or use the same one you started with from part 1)
    a. Name the dashboard **Tiny Home IO 2**
    b. Click the **Create** button
3. Create a **Color Picker** block
    a. Click the blue plus icon



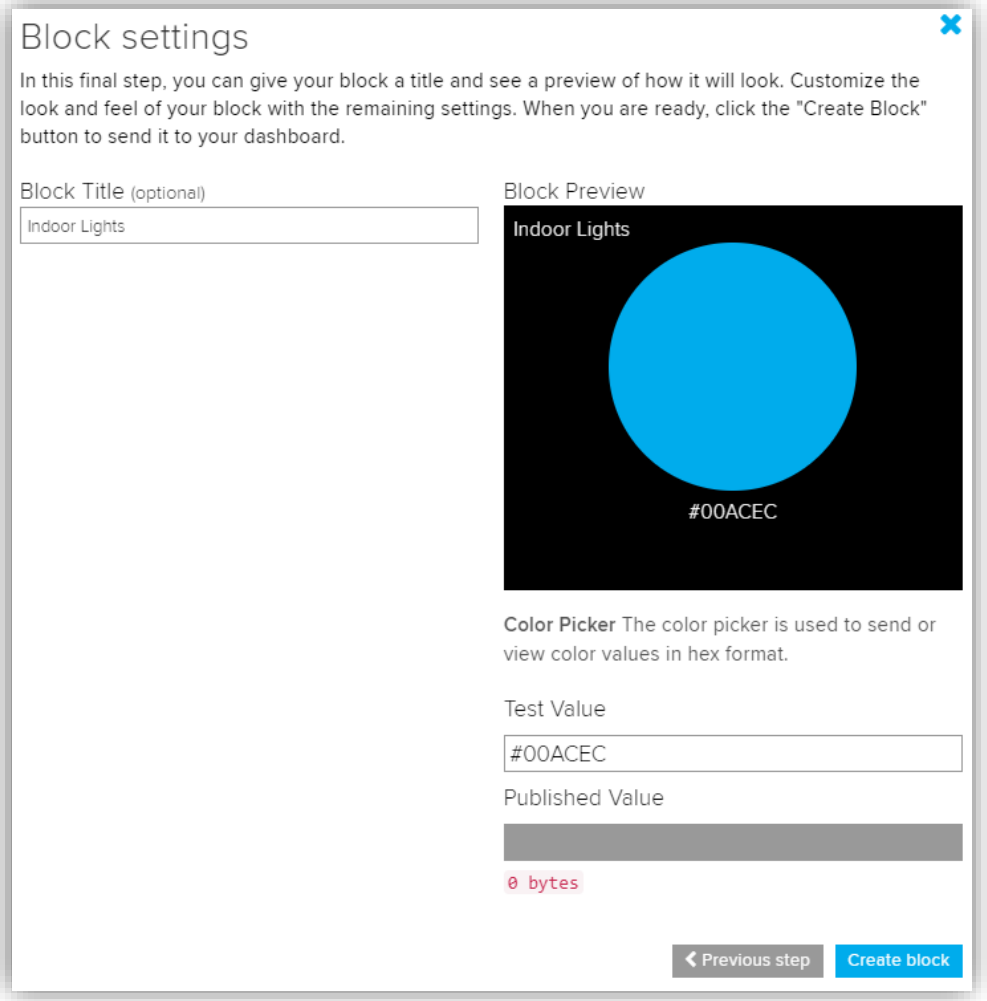
    b. Select the **Color Picker**


Create a new block
Click on the block you would like to add to your dashboard. You can always come back and switch the block type later if you change your mind.
ON
RESET
HELLO WORLD!
#00ACEC

c. Select the **Indoor-Lights** feed

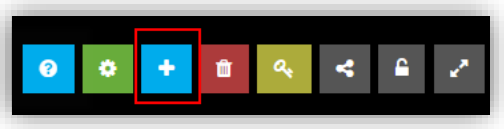


d. Press ***Next step***
e. Enter **Indoor Lights** for the ***Block Title***
f. Press ***Create block***



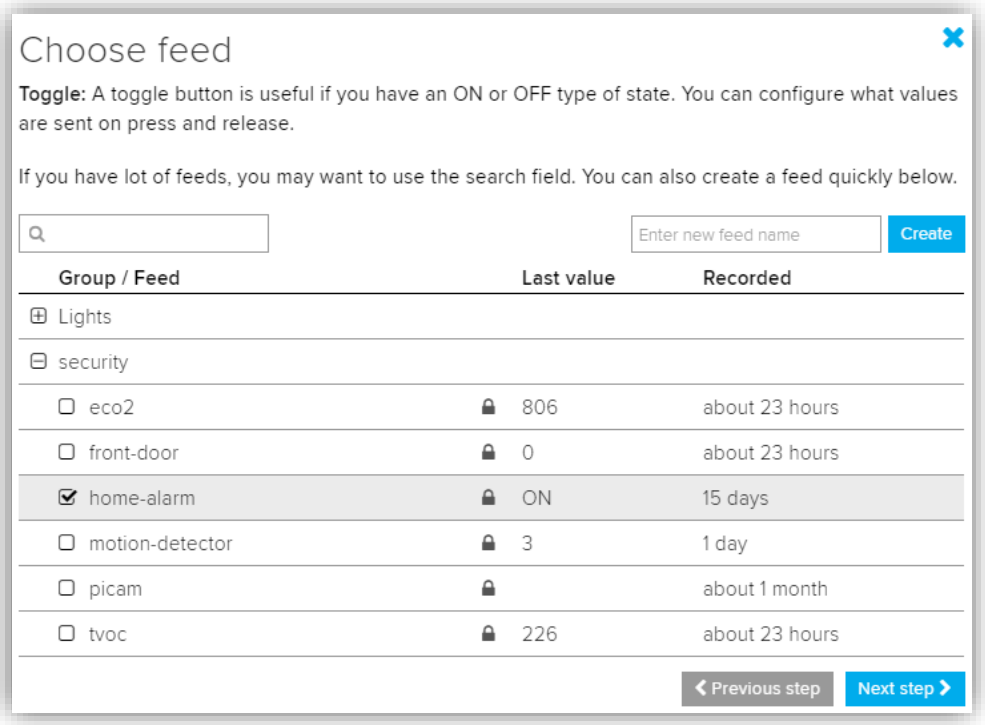
g.

4. Next we'll add the Home Alarm toggle switch (to turn the alarm system on/off).
   a. Press the blue plus icon to add another block

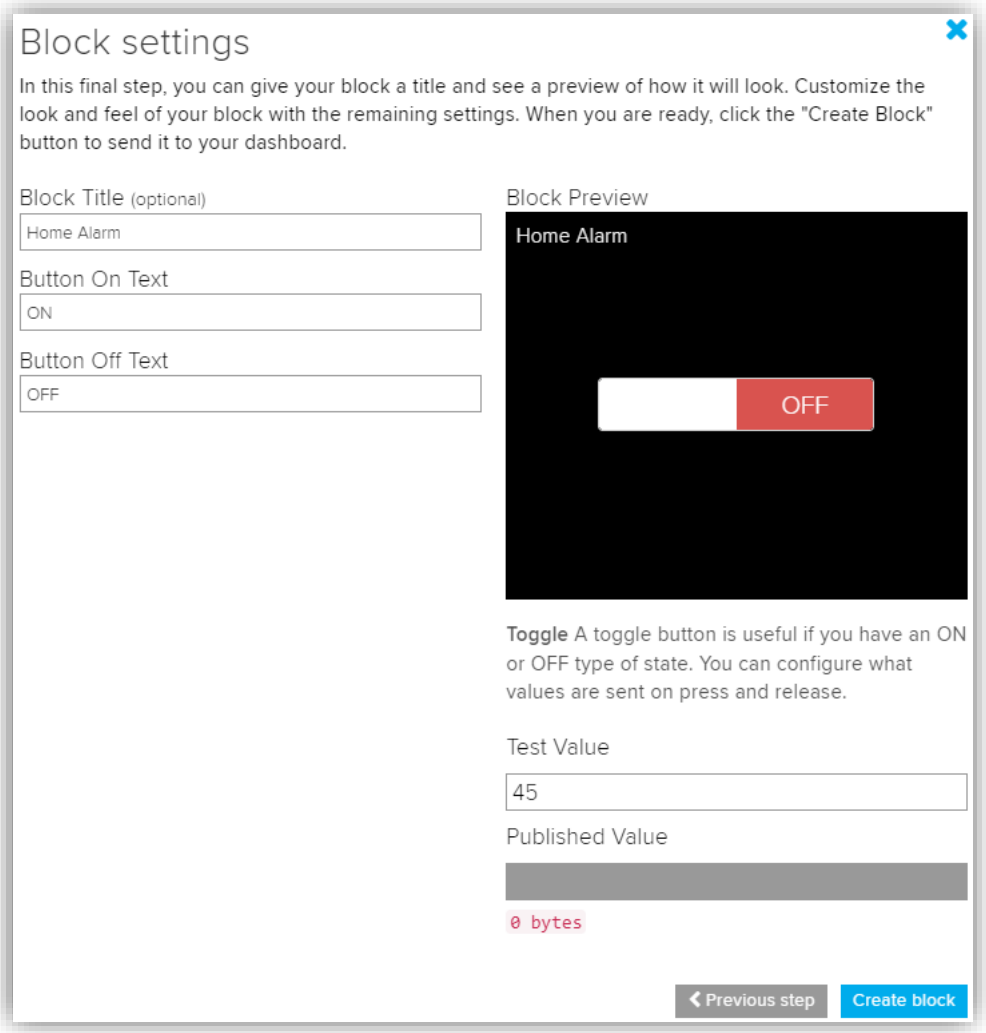


b. Choose the Toggle block

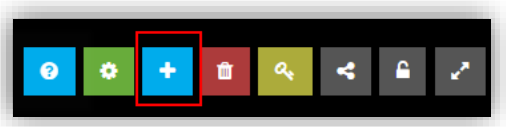
c. Select the **home-alarm** feed and select **Next step >**.



d. Set the **Block Title** to **Home Alarm** and press **Create Block.**

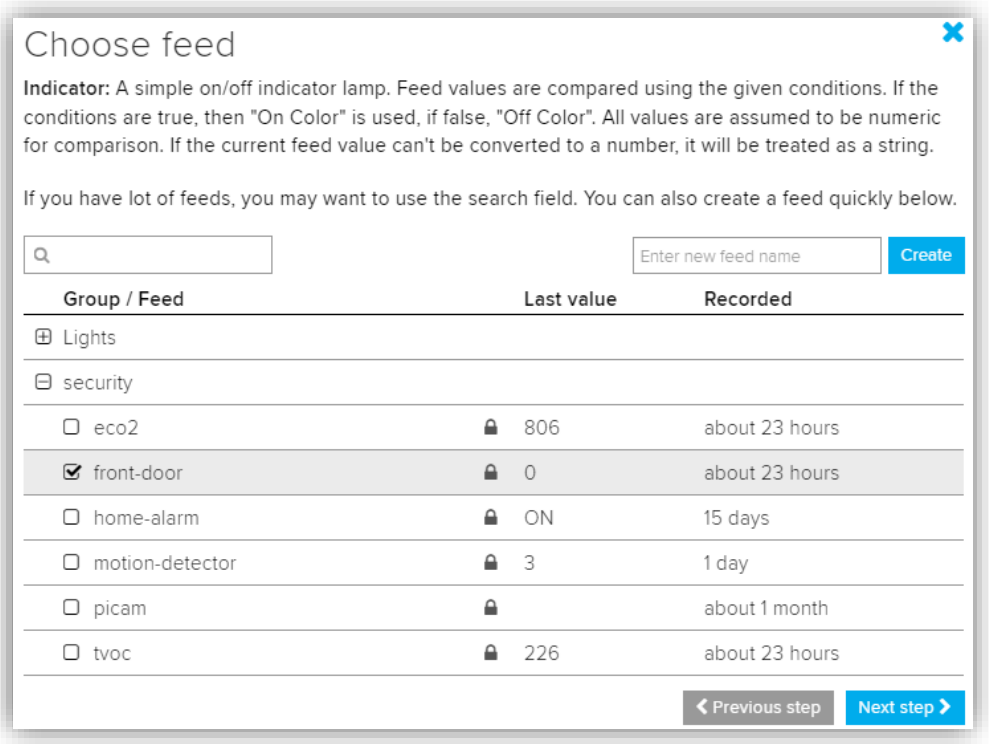


5. Next we'll add the Door Position Indicator (to show if the door is open or closed).
    a. Press the blue plus icon to add another block

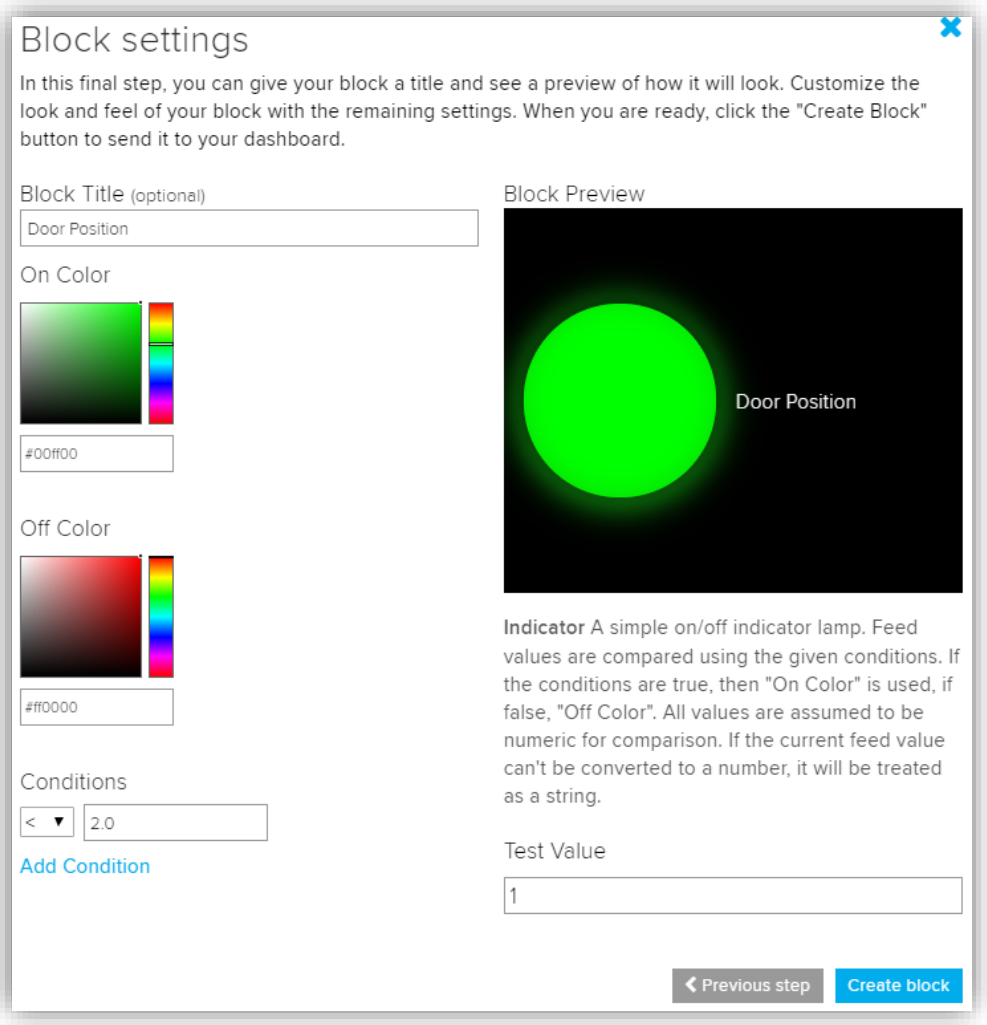


    b. Choose the **Indicator** block

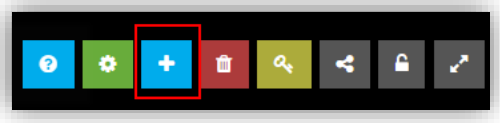
    c. Select the **front-door** feed and select **Next step >**.



d. Set the following and press **Create Block.**
   i. **Block Title** to **Door Position**
   ii. **On Color** to **#00FF00** (green, closed)
   iii. **Off Color** to **#FF0000** (red, open)

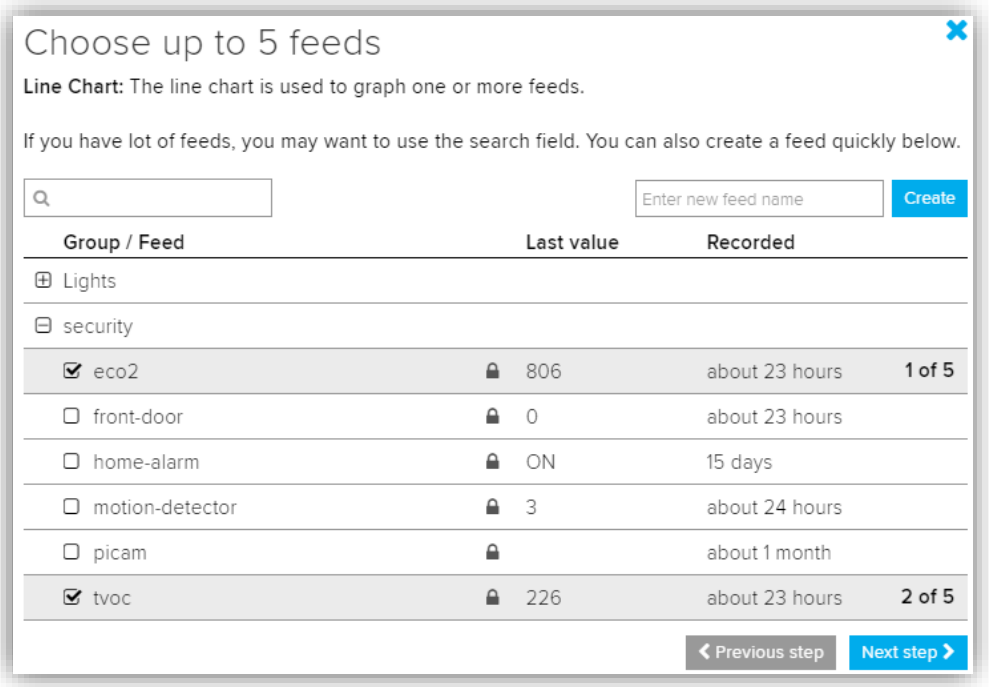


6. Next we'll add the CO2 and TVOC.
    a. Press the blue plus icon to add another block

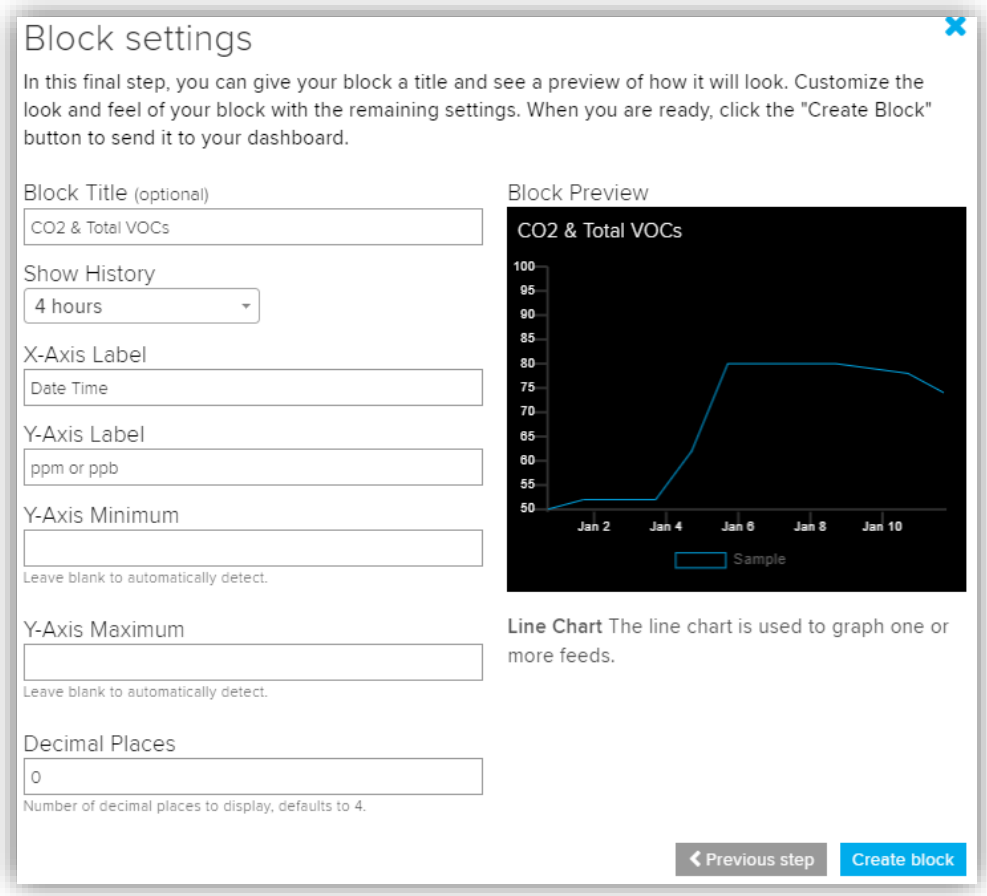


    b. Choose the Line Chart block

    c. Select **tvoc** and **eco2** and press ***Next step***



d. For the Block Title enter **CO2 & Total VOCs**
e. Select a period to show history for **(4 hours)**
f. X-Axis Label is **Date Time**
g. Y-Axis Label **ppm or ppb**
h. Decimal Places is **0**

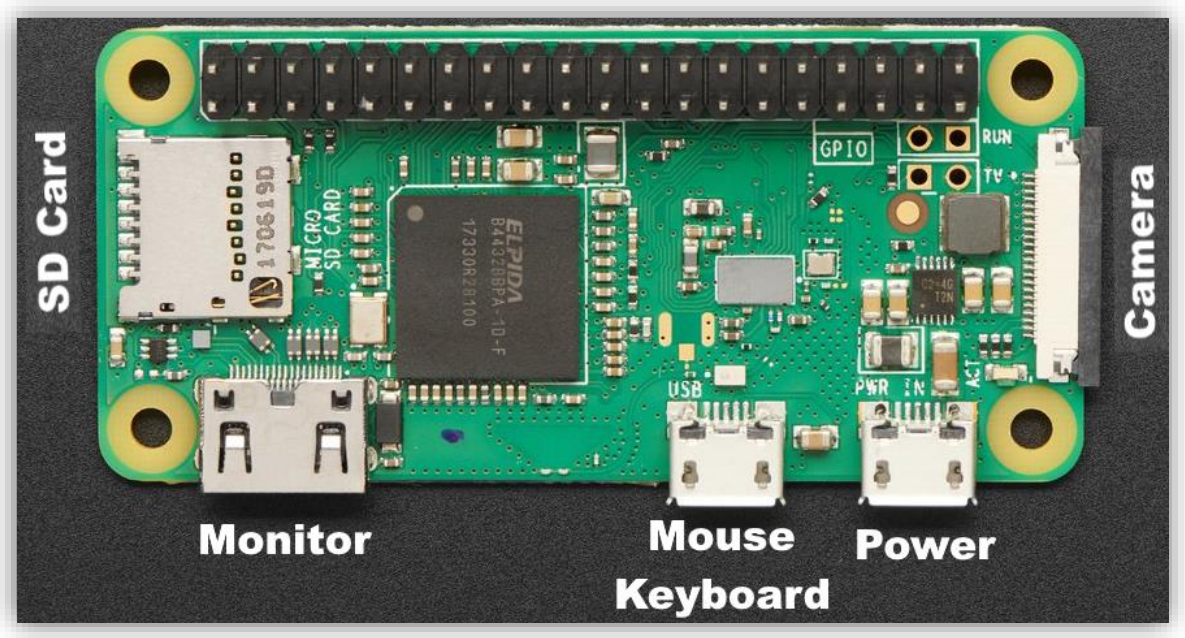


i. Press ***Create block***

At this point, your dashboard is ready.

PI Software Configuration

Your PI Zero W



Connecting the PI

1. The SD card contains your PI operating system. Insert the SD Card into your PI SD Card slot.
    a. To start you must have your SD Card loaded with an operating system.
    b. If you are not starting with a pre-loaded card see the instructions on the following web site to create one yourself.
    https://www.raspberrypi.org/documentation/installation/installing-images/
2. Connect the monitor
    a. Insert the mini-HDMI connector to the Monitor plug on your PI.
    b. Insert the other end into your Monitor.
3. Connect the keyboard and mouse
    a. Connect the USB hub to the left micro-USB plug.
    b. Insert your mouse and keyboard to the USB hub.
4. Add Power
    a. Connect the micro-USB plug from the power plug to the right micro-USB plug.
    b. Plug the other end into the wall.
5. At this point your PI should boot up
    a. You will see a raspberry on your screen, some colors, then the boot command console.
    b. A desktop GUI will display indicating your PI is booted and ready to go.

6. Desktop GUI
    a. Web Browser is the blue globe.
    b. Command Prompt is the black square with **>_** in it.
    c. The raspberry icon contains command actions (like restart, configuration, etc).

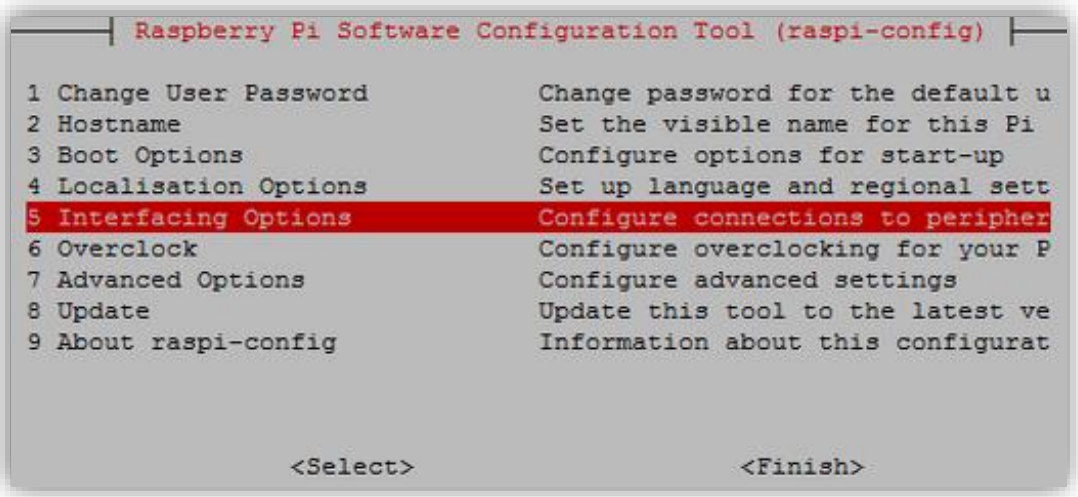
Useful Linux command line commands

1. '**pwd**' – Print Working Directory
2. '**ls**'- List current directory (kind of like 'dir' in windows)
3. '**cd**' – change directory
    a. '**cd ..**' – goes back up one directory
4. '**man**' – give you help.
    a. Example '**man passwd**' will give help on the password command.
5. '**nano**' – edits a specific file.
    a. Press '**CTRL-x**' to exit the program, then press '**y**' to save the file, then press '**Enter**' to save to the specified file (overwrite).
    b. You will see this used later.

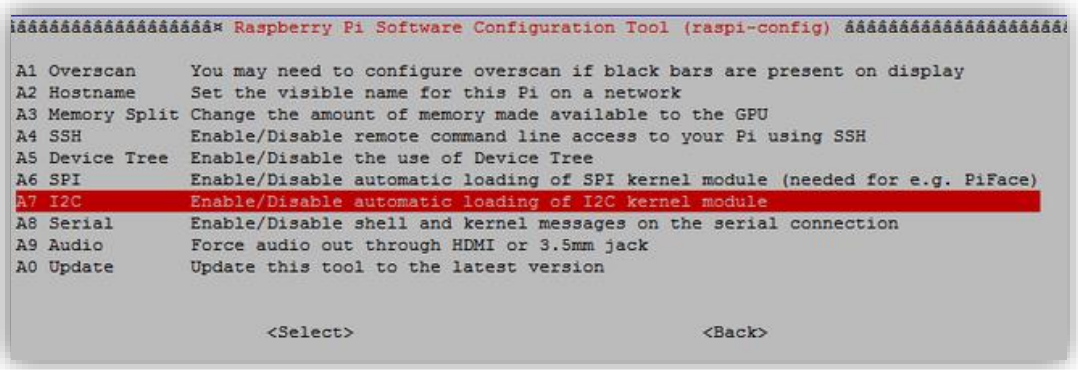
Software Installation

1. [Optional] Log In
    a. Default username and password are "**pi**" and "**raspberry**"
    b. Type in "**sudo passwd**" to change your password.
    c. Write down your new password here: _____
2. Connect to WiFi
    a. Right click the wifi icon on the top right to open the wifi dialog.
    b. Choose the wlan0 adapter and select your network's SSID (example: InventHQ)
    c. Enter the network password.
    d. You should be connected.

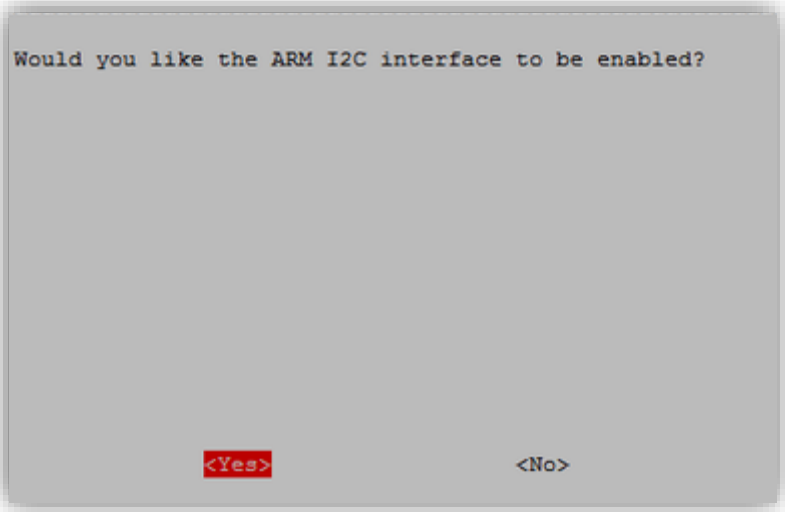
3. Enable I2C Interface
    a. From the Command Line type "**sudo raspi-config**" to open the Configuration GUI.
    b. Go to Interfacing Options

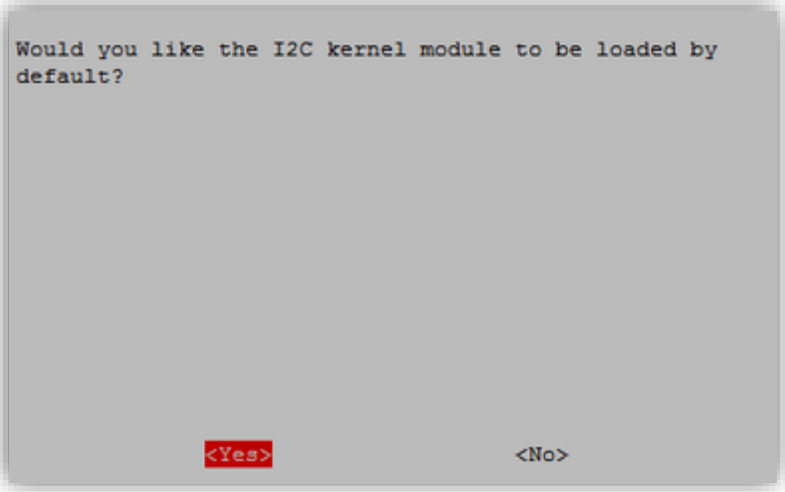


    c. Go to I2C



    d. Enable the I2C





e. Close the Raspi-Config application.

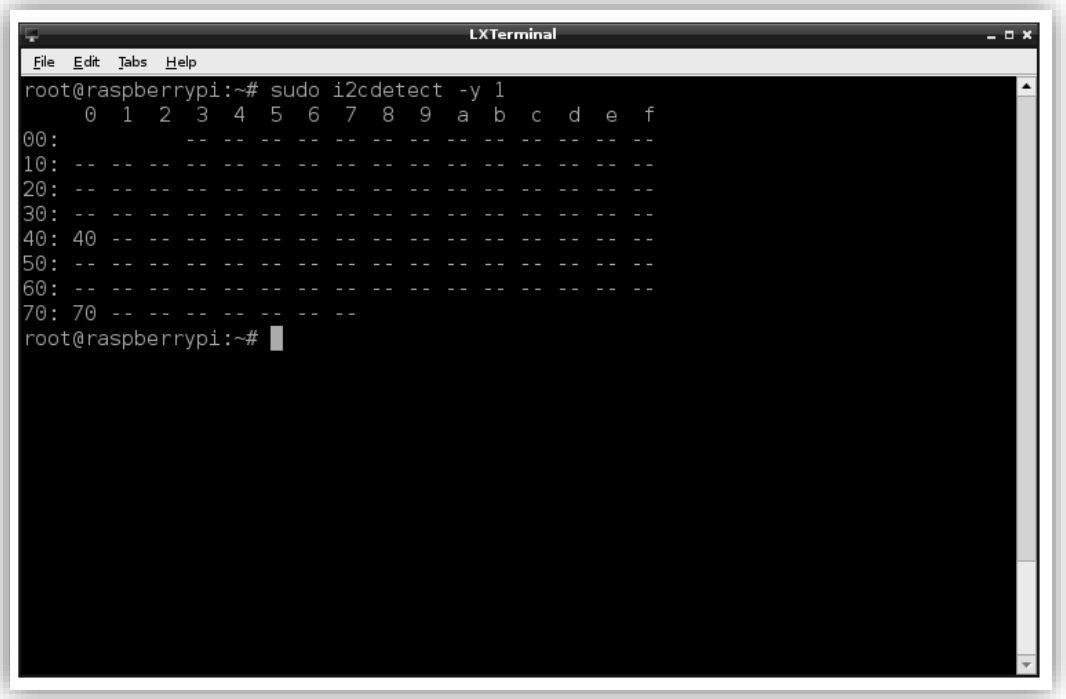
4. System Update (note: you must use the super-user '**sudo**' command when shown)
   a. Open a command line window
   b. Type "**sudo apt-get update**" to update the operating system to the latest libraries.
   c. Type "**sudo apt-get upgrade**" to upgrade the operating system packages.
   d. Type "**sudo shutdown -r now**" to restart the machine and allow the upgrade to complete.
   e. Once your machine has rebooted, re-open the command line and type "**sudo pip3 install --upgrade setuptools**"

Install Software Libraries

Note that we are using '**pip3**' and '**python3**' commands at all times below.

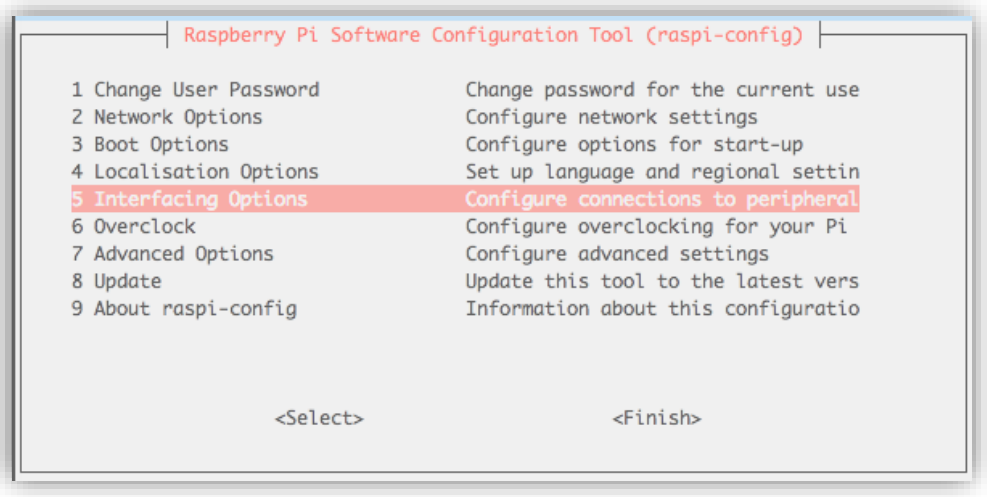
1. Install **Python Blinka** Libraries: https://learn.adafruit.com/adafruit-io-basics-digital-input/python-setup
    a. Type "**sudo pip3 install RPI.GPIO**" to install Raspberry PI GPIO library.
    b. Type "**sudo pip3 install adafruit-blinka**" to install Blinka.
    c. Type "**sudo pip3 install adafruit-io**" to install the Adafruit IO libraries.

2. Install I2C Libraries: https://learn.adafruit.com/adafruits-raspberry-pi-lesson-4-gpio-setup/configuring-i2c
    a. Type "**sudo apt-get install -y python-smbus**" to install Python Bus.
    b. Type "**sudo apt-get install -y i2c-tools**" to install the I2C Tools.
    c. Type "**sudo i2cdetect -y 1**" to verify I2C is enabled

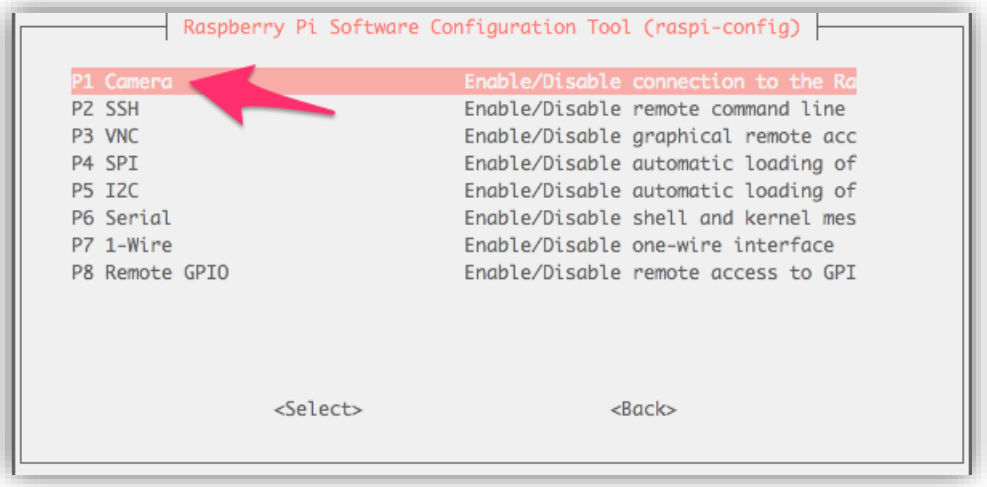


3. Install the Libraries for the Air Quality Board & NeoPixels
    a. Type "**sudo pip3 install adafruit-circuitpython-sgp30**" to install the libraries for the SGP30 board.
    b. Type "**sudo pip3 install Adafruit_CircuitPython_NeoPixel**" to install the libraries for the NeoPixels (the Jewel and the Strip Lights).
4. [Optional] Install the camera libraries
    a. This doesn't hurt to run even if you don't have a camera.
    b. Type "**python3 -c "import picamera"**"
    c. IF you get an error:
        i. Type "**sudo apt-get update**"
        ii. Type "**sudo apt-get install python-picamera python3-picamera**"

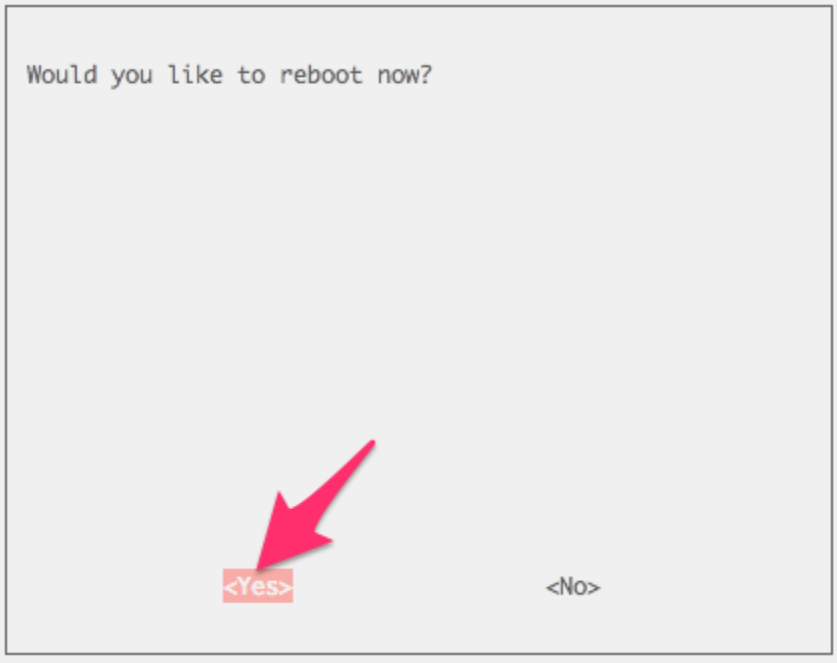
d. Type "**raspi-config**" and press **Enter**
e. Using your arrow keys, navigate to **Interfacing Options** and press the **Enter** key.



f. The first option is to enable the camera, use your arrow keys to navigate to this option and press **Enter**.



g. You'll be prompted to reboot the Pi. Use the arrow keys to select the **Yes** option and press **Enter**.



h. After the Pi reboots, test that the camera works by using the raspistill command "**raspistill -o cam_test.jpg**"
i. Typing "ls" into the terminal should return the cam_test.jpg file. You can copy this file off of the Pi and view it on your computer, too!

Update the Code on the PI

1. Download Example Code
    a. Change to the root directory by typing "**cd ~**"
    b. Type "**git clone** https://github.com/eironside/IO-House-Series" to download the code for this class.
    c. [Optional] Do the following after class, this takes a long time!!

   a. Type "**git clone https://github.com/adafruit/Adafruit_Learning_System_Guides.git**" to download the Adafruit example code.This is a large library with lots to explore

2. Edit the example python file
   a. Type "**cd IO-House-Series**"
   b. Type "**cd Lights_and_Temp**"
   c. Open a web browser on your Pi and go to https://io.adafruit.com/<username>/profile
      i. Press the **View AIO Key** menu item on the left to view your key.
   d. To edit the code, type "**nano io_house_light_temp.py**"
   e. Scroll down to the **Adafruit_IO_KEY** variable, copy the following from the IO web page.
      i. Set your **Adafruit IO Key** from above.
      ii. Set the **Adafruit_IO_USERNAME** to your Adafruit IO Username.
   f. When you're done editing the values, save the file by pressing "**CTRL + x**".
      i. When prompted to save the modified buffer, type **Y** and press **Enter**.
      ii. At the File Name to Write prompt, press **Enter** and you should be directed back to the terminal.
3. Run the Code
   a. Back in the command prompt, type "**sudo python3 io_house_light_temp.py**"
   b. It should run without an error and print out the various values
      i. It may ask you to set a color on the dashboard
4. Check your dashboard
   a. The temperature and humidity should start to show on the graph.
   b. Set a color for the two lights and notice that they change on the house.
5. Check your code and make any changes you want to make

[Optional] Set up for Auto-Start

Sometimes it is nice to have your device boot up and start doing something. This will show you how to make your PI run your python script on startup. There are numerous ways to accomplish this, I'm showing you the **systemd** option.

1. Create a "Unit File":
    a. Type in the command prompt "**sudo nano /lib/systemd/system/homeio_lights.service**"
    b. In the editor make sure your file looks like the following:

```
[Unit]
 Description=My Home IO Lights Service
 After=multi-user.target

 [Service]
 Type=idle
 ExecStart=/usr/bin/python3 /IO-House-Series/Lights_and_Temp/io_house_light_temp.py

 [Install]
 WantedBy=multi-user.target
```

    c. Exit and save the file
        i. **CTRL - x**, **Y** to save the changes, and press **Enter** to write to the unit file.
2. Configure ***systemd***
    a. Enable the unit file via system control
        i. Type "**sudo systemctl daemon-reload**"
        ii. Then type "**sudo systemctl enable homeio_lights.service**"
    b. Reboot the machine by typing "**sudo reboot**"
    c. Upon reboot, your house should light up and be running your code