

# Introduction to IoT Programming using the Raspberry Pi

How to stream sensor data to a streaming API for real  
time visualization using the Raspberry Pi computer

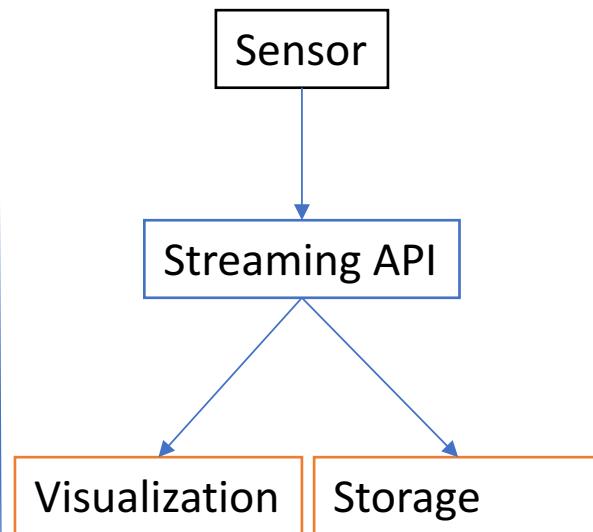
# Contents

- IoT Concept
- Introduction to the Raspberry Pi
- Setting up a Raspberry Pi
- Connecting the Temperature Sensor
- Setting up a Streaming API
- Streaming Data

# IoT Concept

- Basic idea: a network of physical devices, embedded with sensors and connectivity to the internet.
- Important features
  - Low cost of devices to enable scaling
  - Integration of devices/sensors into processes without disrupting or altering the process. (i.e size, interaction)
  - No human to computer interaction after setup
  - Real time
  - Unique identification of devices
- A network of things connected to the internet

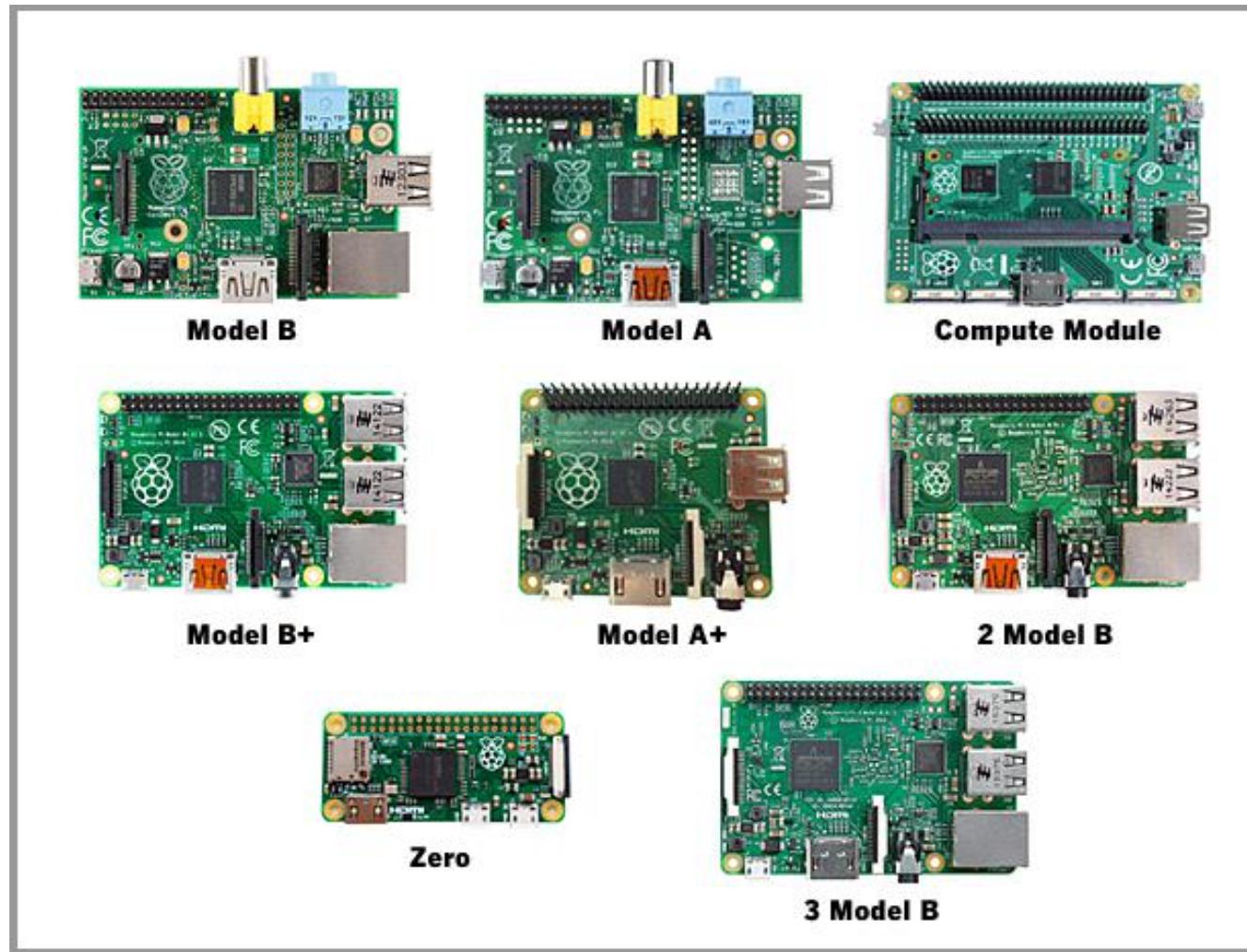
For Today:



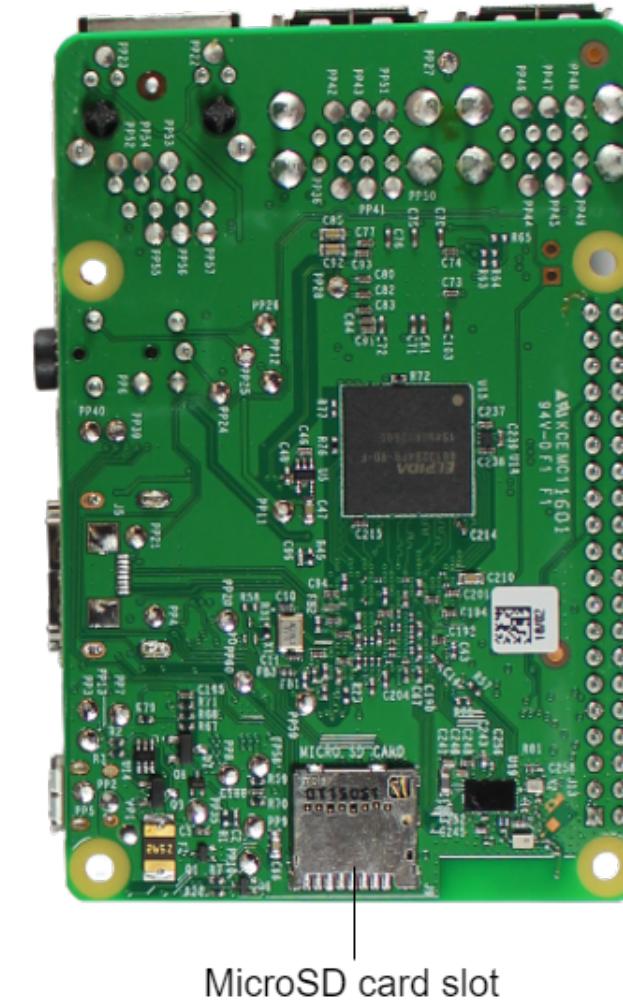
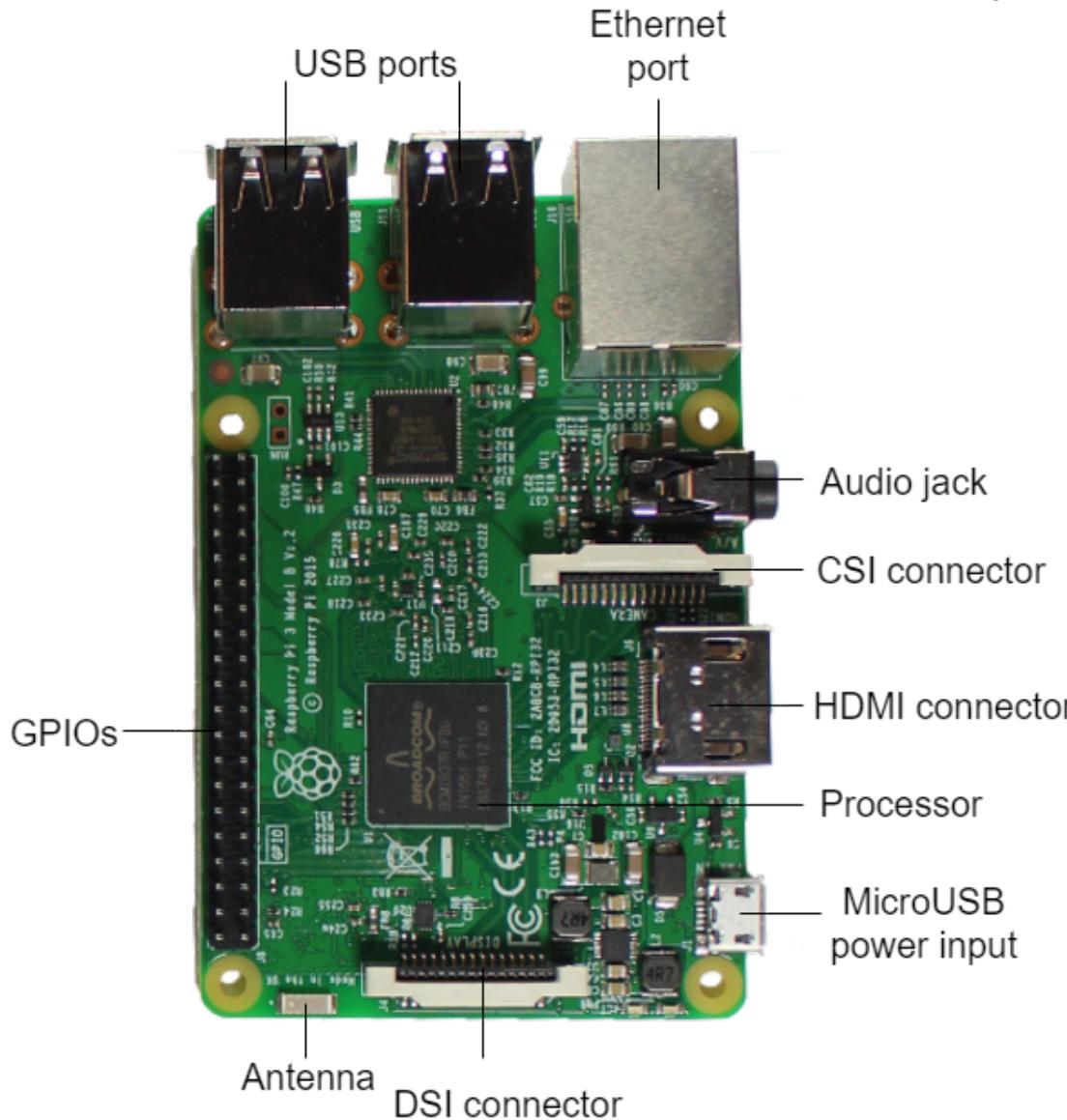
# Raspberry Pi Foundation

- A charity founded in 2009 to promote the study of computer science in schools in the UK.
- Supported by the University of Cambridge.
- Founded by a group of Professors, CEOs and engineers, who wanted to improve the quantity and quality of students applying for Computer Science degrees.
- First Pi computer was developed in 2011 and retailed for US\$35.
- In 2015, the Raspberry Pi Zero was released and retailed for **US\$5**.

# Introduction to the Raspberry Pi: Models

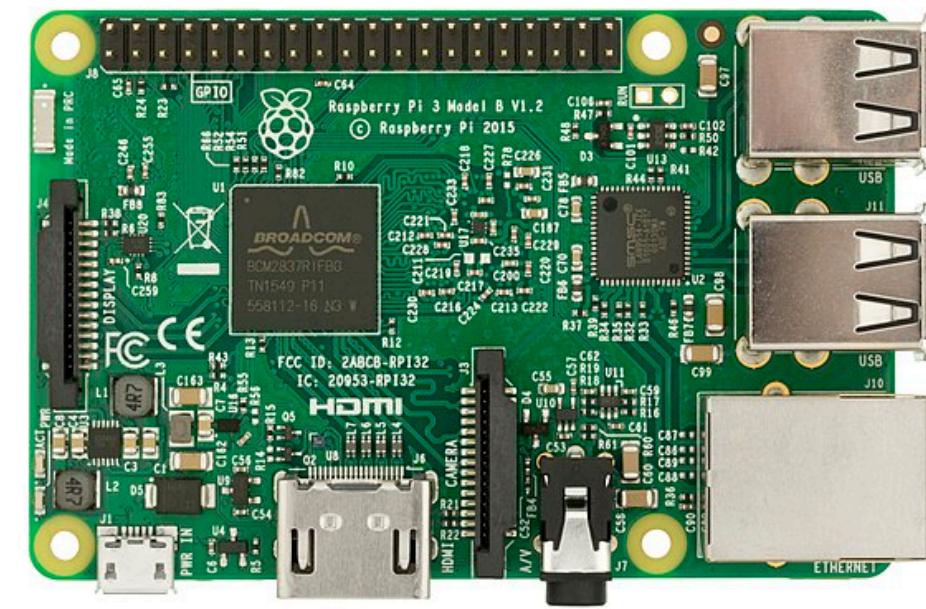


# Introduction to the Raspberry Pi: 3 Model B



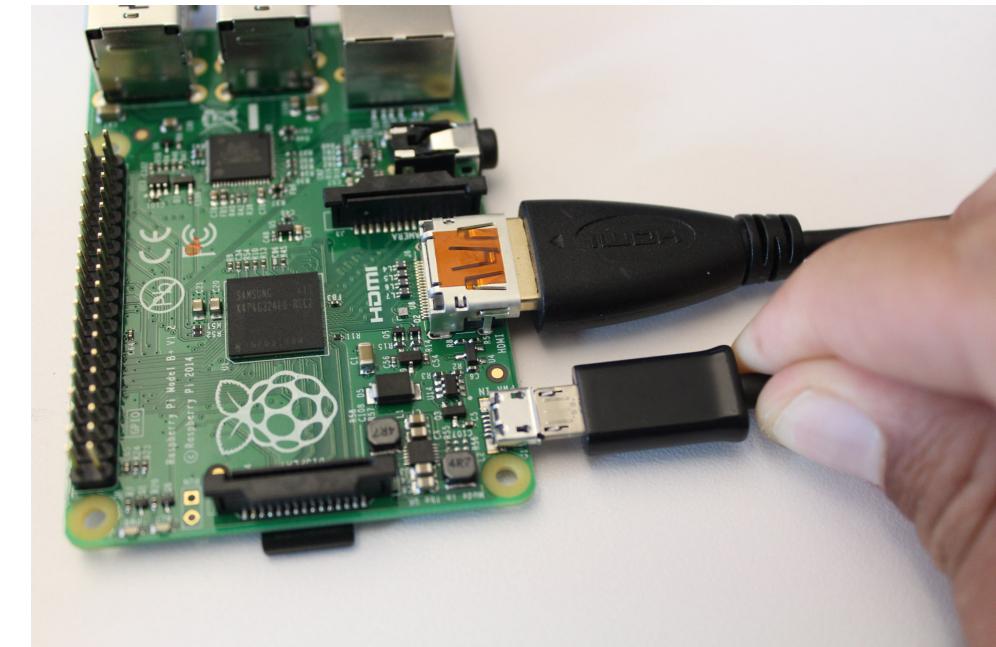
# Introduction to the Raspberry Pi

- Small, single-board computers from the United Kingdom, designed for teaching students basic computer science skills. Linux operating system.
  - Original models became far more popular than anticipated and the Pi has become the best-selling British computer with more than 15 million units sold (July 2017).
- Raspberry Pi 3 Model B (\$35)
  - 1.2 GHz 64-bit quad-core ARM Cortex-A53
  - 1 GB RAM (shared with GPU)
  - Broadcom VideoCore IV @ 250 MHz GPU
- Raspberry Pi Zero W (\$10)
  - 1 GHz single-core ARM1176JZF-S
  - 512 MB RAM (shared with GPU)
  - Broadcom VideoCore IV @ 250 MHz GPU



# Setting up a Raspberry Pi: What you will need

- A Raspberry Pi
- Micro USB cable
- HDMI Cable
- Keyboard and Mouse
- Display
- Mini HDMI connector (Zero model only)
- SD Card (8GB+)
- Default password: raspberry



# Setting up a Raspberry Pi: Operating System

A guide to setting up your Raspberry Pi

## What you will need

### ESSENTIAL (FOR GENERAL USE)

#### [SD Card](#)

- We recommend an 8GB class 4 SD card, ideally preinstalled with [NOOBS](#).

#### [Display and connectivity cable](#)

- Any HDMI/DVI monitor and any TV should work as a display for the Pi. For best results, use one with HDMI input, but other connections are available for older devices.

#### [Keyboard and mouse](#)

- Any standard USB keyboard and mouse will work with your Raspberry Pi.
- Wireless keyboards and mice will work if already paired.
- For keyboard layout configuration options see [raspi-config](#).

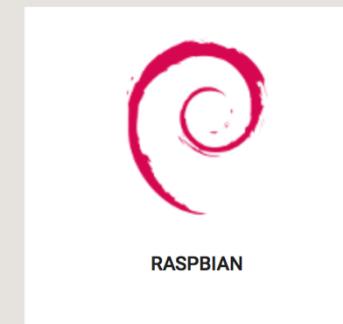
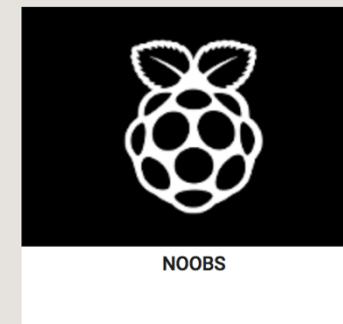
#### [Power supply](#)

- The Pi is powered by a USB Micro power supply (like most standard mobile phone chargers).
- You'll need a good-quality power supply that can supply at least 2A at 5V for the Model 3B, or 700mA at 5V for the earlier, lower powered models.
- Low current (~700mA) power supplies will work for basic usage, but are likely to cause the Pi to reboot if it draws too much power.

## DOWNLOADS

**Raspbian** is our official operating system for **all** models of the Raspberry Pi.

Download it here, or use **NOOBS**, our easy installer for Raspbian and more.



# Setting up a Raspberry Pi: Writing and Image

## Writing an image to the SD card

You will need to use an image writing tool to install the image you have downloaded on your SD card.

**Etcher** is a graphical SD card writing tool that works on Mac OS, Linux and Windows, and is the easiest option for most users. Etcher also supports writing images directly from the zip file, without any unzipping required. To write your image with Etcher:

- Download [Etcher](#) and install it.
- Connect an SD card reader with the SD card inside.
- Open Etcher and select from your hard drive the Raspberry Pi `.img` or `.zip` file you wish to write to the SD card.
- Select the SD card you wish to write your image to.
- Review your selections and click 'Flash!' to begin writing data to the SD card.

# Setting up a Raspberry Pi: Setting up SSH

As of the November 2016 release, Raspbian has the SSH server disabled by default. It can be enabled manually from the desktop:

1. Launch `Raspberry Pi Configuration` from the `Preferences` menu
2. Navigate to the `Interfaces` tab
3. Select `Enabled` next to `SSH`
4. Click `OK`

Alternatively, `raspi-config` can be used in the terminal:

1. Enter `sudo raspi-config` in a terminal window
2. Select `Interfacing Options`
3. Navigate to and select `SSH`
4. Choose `Yes`
5. Select `Ok`
6. Choose `Finish`

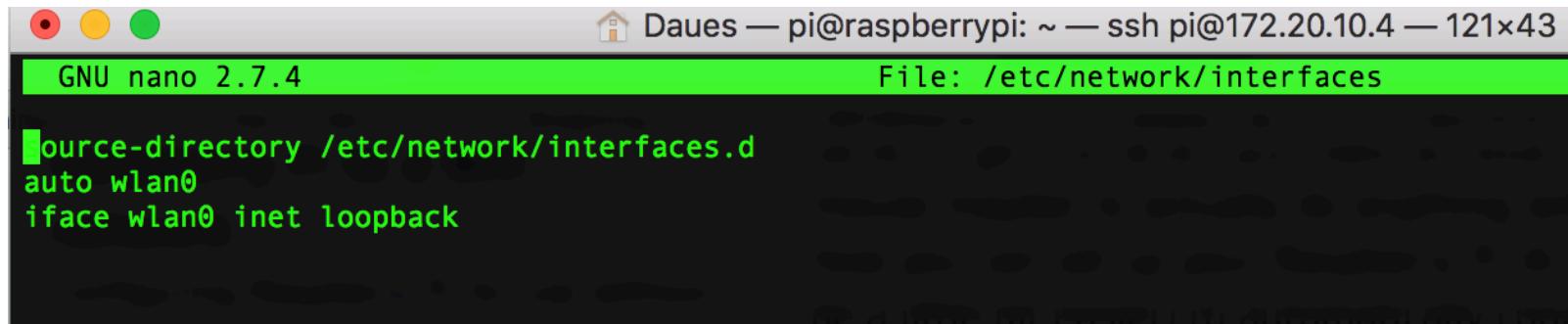
Alternatively, use `systemctl` to start the service

```
sudo systemctl enable ssh
sudo systemctl start ssh
```

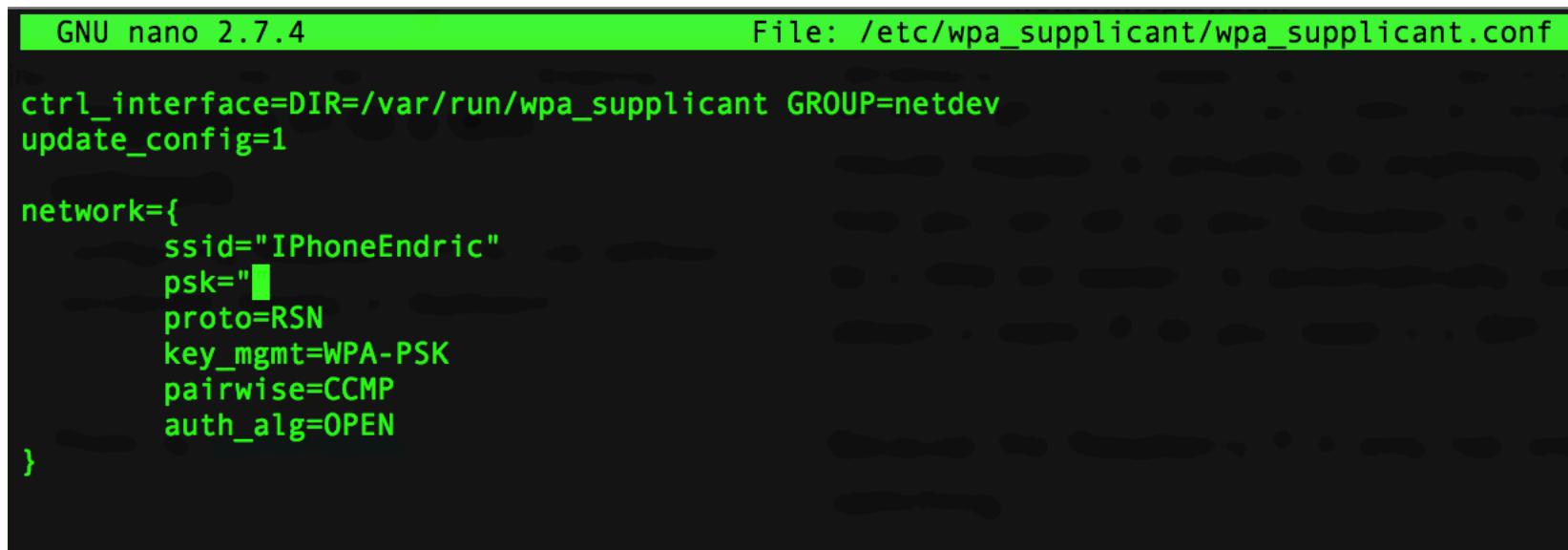
# Setting up a Raspberry Pi

- I recommend the following setup:
  - Download Raspbian and boot the Pi in the GUI using the HDMI port
  - Connect to your smartphone hotspot and check the Pi's IP address (command line: `hostname -I`)
  - Set a password (command line: `passwd`) **default is “raspberry”**
  - Connect your laptop to the hotspot and access the Pi's command line using SSH: `ssh pi@ipaddress` (MAC) or Putty for Windows
  - Every time you boot the Pi, it should automatically connect to your hotspot if you implement the following setup (next slide)

# Setting up a Raspberry Pi: Wifi



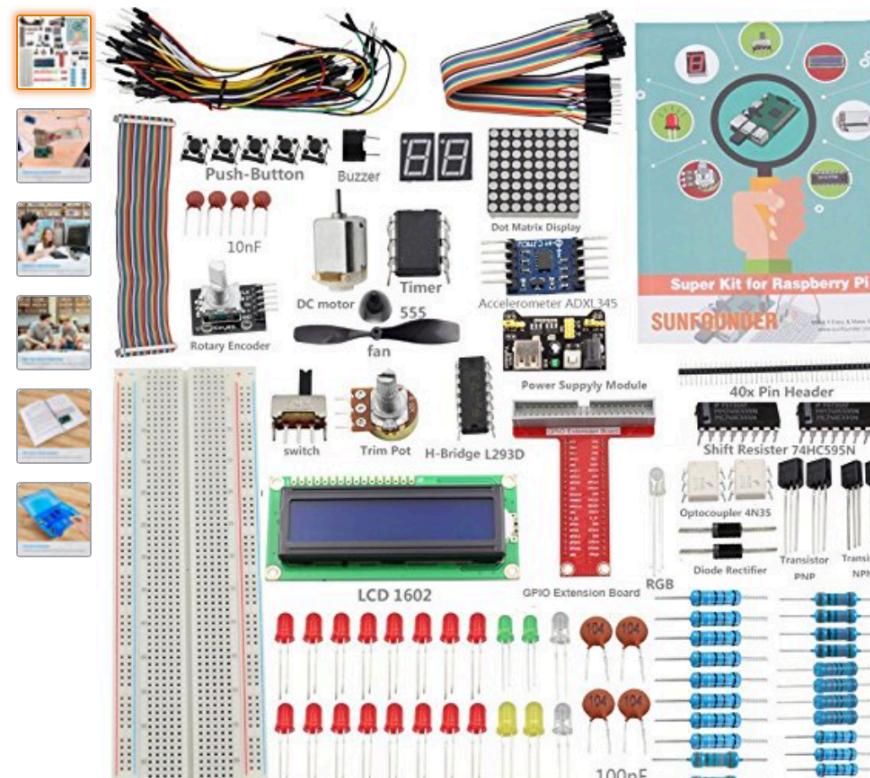
```
GNU nano 2.7.4 File: /etc/network/interfaces
source-directory /etc/network/interfaces.d
auto wlan0
iface wlan0 inet loopback
```



```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="iPhoneEndric"
    psk="■"
    proto=RSN
    key_mgmt=WPA-PSK
    pairwise=CCMP
    auth_alg=OPEN
}
```

# Starting with IoT Projects



Roll over image to zoom in

SunFounder

Raspberry Pi 3 Zero Starter Kit - SunFounder Project Super Kit for Raspberry Pi 3 2 Zero Model B+ A+ Including GPIO Breakout Board Breadboard LCD DC Motor LED RGB Dot Matrix 73 Page Manual User Guide

 113 customer reviews | 29 answered questions

Amazon's Choice for "raspberry pi accessories"

Price: **\$35.99** + \$16.24 Shipping & Import Fees Deposit to Germany [Details](#)

Get \$40 off instantly: Pay \$0.00 upon approval for the Amazon.com Store Card.

**Buy 1, get a discount on selected products** 3 Applicable Promotion(s) ▾

In Stock.

Color: V2.0

 \$35.99  
✓prime

 \$45.99  
✓prime

- STEM EDUCATION KIT - Perfect choice for beginners to learn Raspberry Pi, electronics and program.
- LEARN TO PROGRAMMING - Python code and C code are provided for hardware to control

Share [Email](#) [Facebook](#) [Twitter](#) [Pinterest](#)

Qty: 1 ▾

Add a Protection Plan:

- 4-Year Protection for **\$4.01**  
 3-Year Protection for **\$1.04**



Add to Cart



Buy Now

Turn on 1-Click ordering for this browser

Ship to:

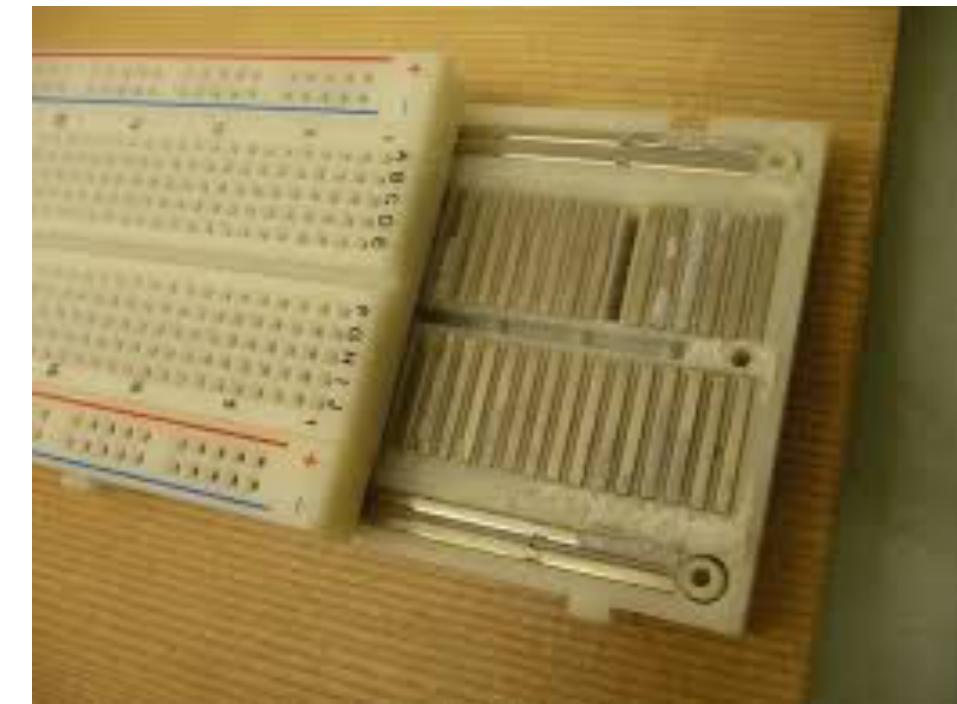
Endric Daues- Sehnde - 31319 ▾

Add to List

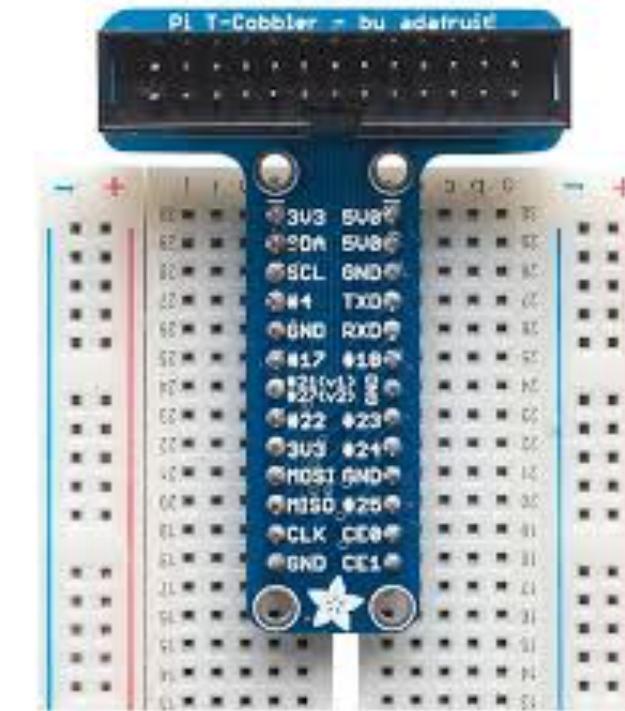
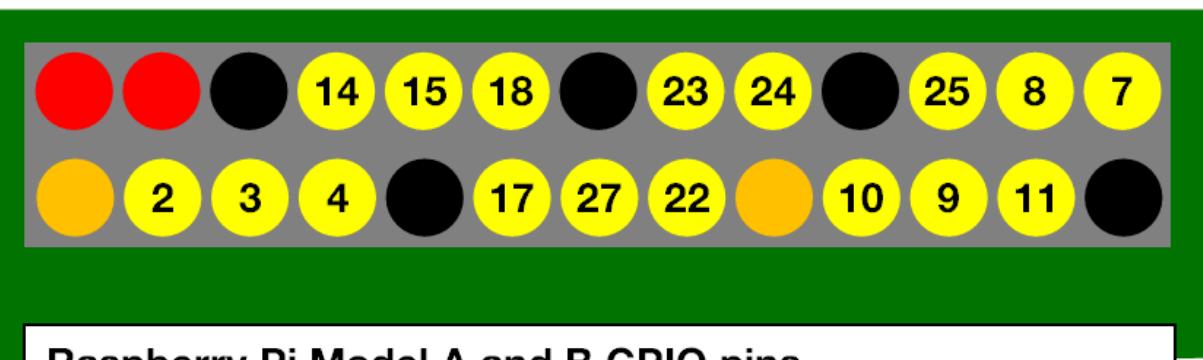
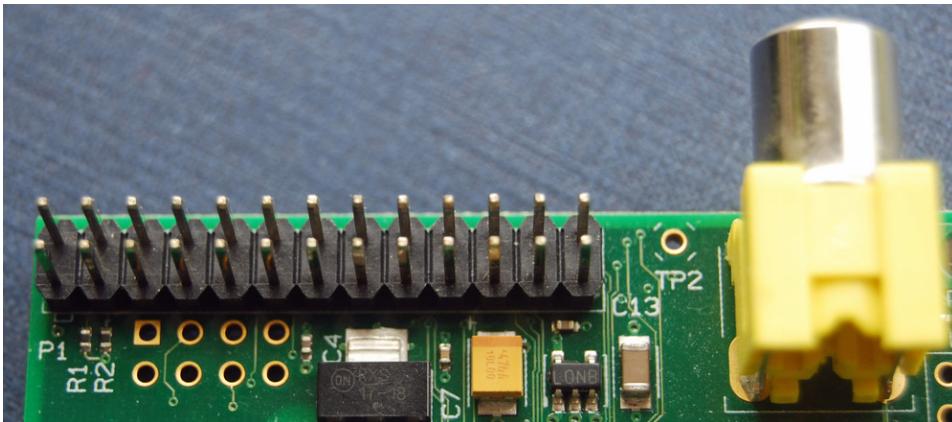
Add to your Dash Buttons

# Connecting the Temperature Sensor

- You will need the following:
  - Solderless breadboard (\$5)
  - Breadboard jumper wires
  - DS18B20 temperature sensor (\$1)
  - 4.7K resistor (\$1)

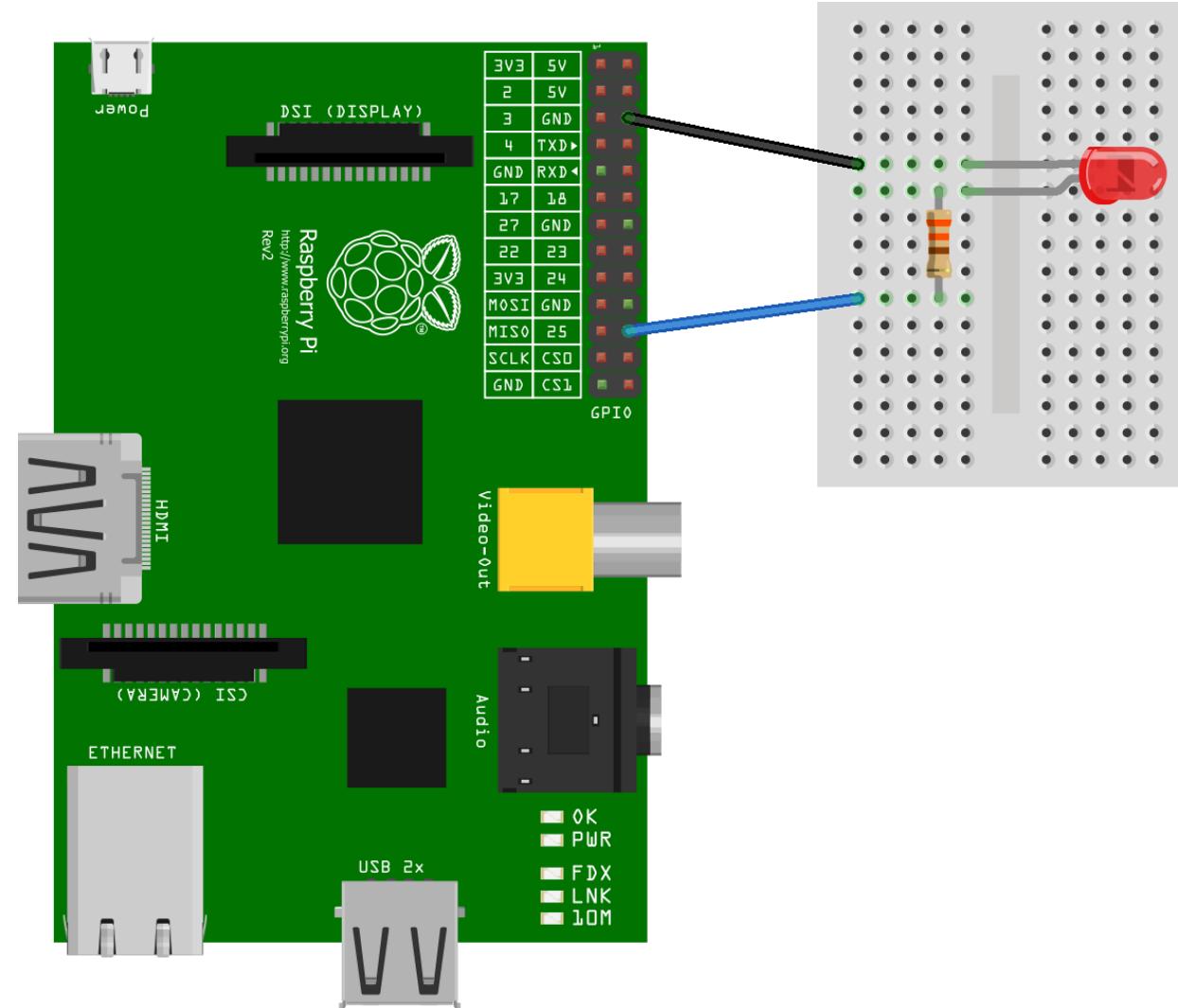


# GPIO Pins

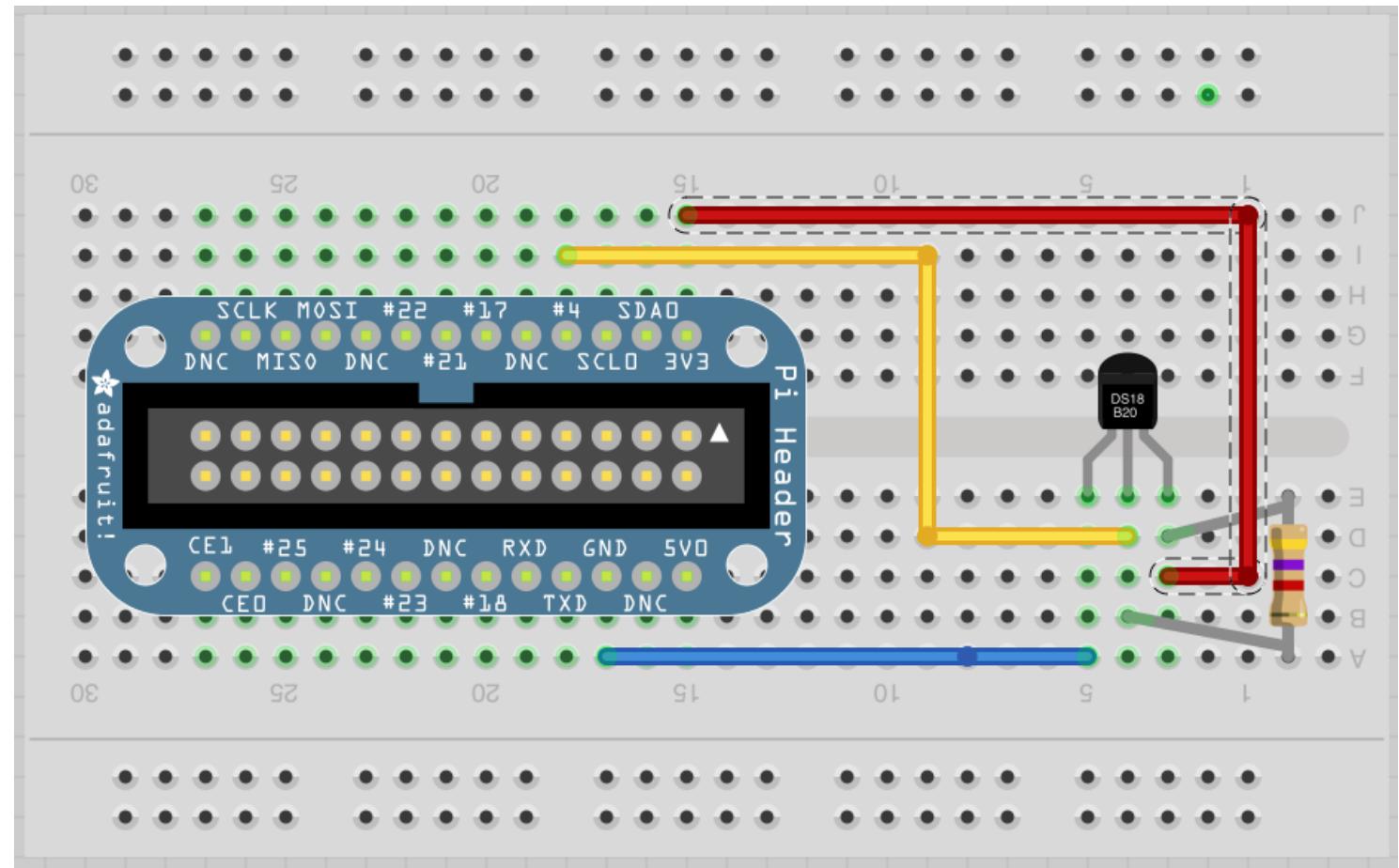
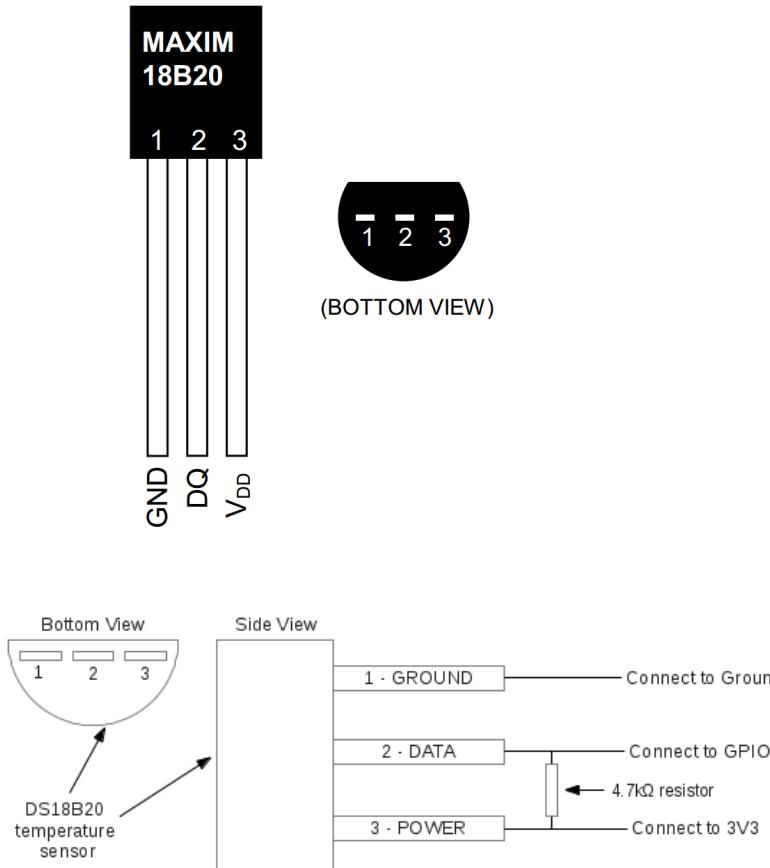


T-Cobbler: Great for more complex wiring projects that require heavy use of the breadboard. The T-Cobbler connects to the Pi's GPIO pins.

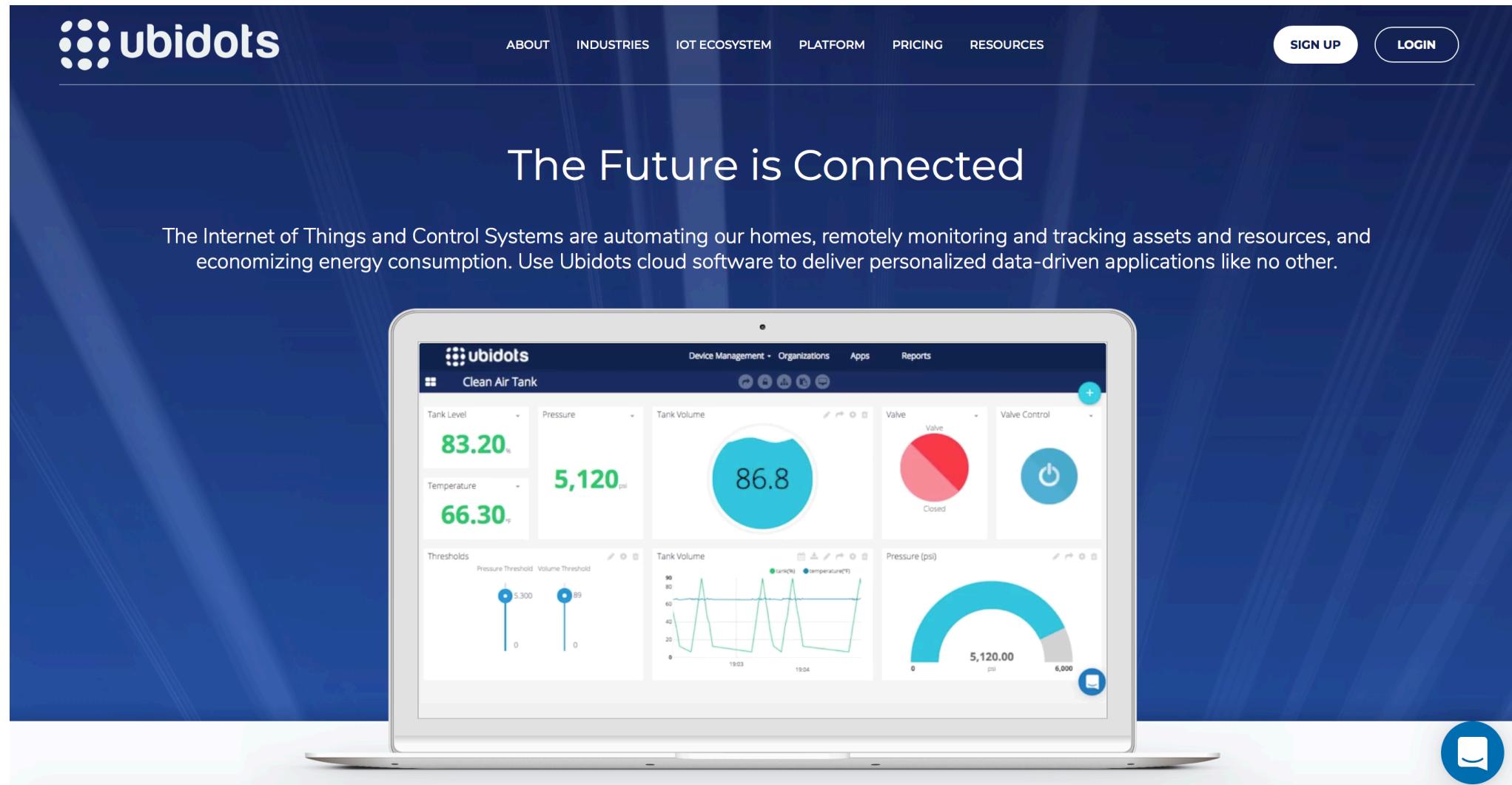
# GPIO Pins



# Connecting the Temperature Sensor



# Setting up a Streaming API



The Ubidots website features a dark blue header with the logo on the left, a navigation bar with links for ABOUT, INDUSTRIES, IOT ECOSYSTEM, PLATFORM, PRICING, and RESOURCES, and two buttons for SIGN UP and LOGIN on the right.

## The Future is Connected

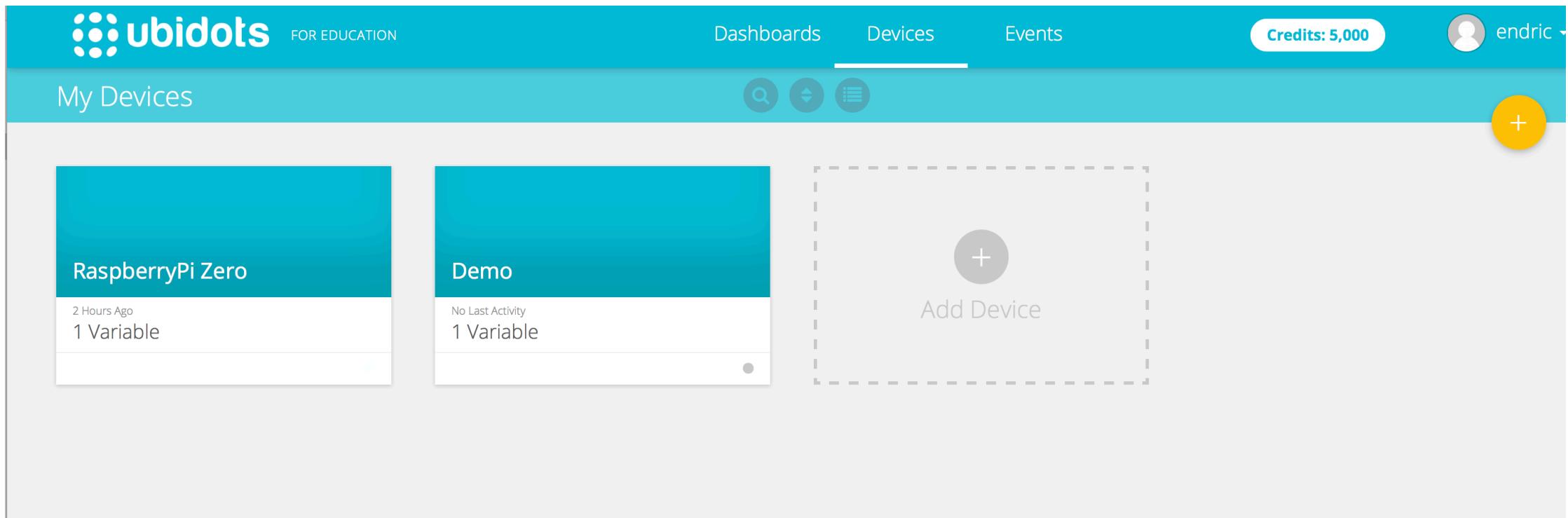
The Internet of Things and Control Systems are automating our homes, remotely monitoring and tracking assets and resources, and economizing energy consumption. Use Ubidots cloud software to deliver personalized data-driven applications like no other.

A central feature is a tablet displaying the Ubidots platform interface. The interface shows real-time data for a "Clean Air Tank". Key metrics displayed include:

- Tank Level: 83.20%
- Pressure: 5,120 psi
- Temperature: 66.30 °F
- Tank Volume: 86.8
- Valve Status: Closed
- Pressure (psi) Gauge: 5,120.00 psi (out of 6,000)

The dashboard also includes a line graph showing tank level and temperature over time, and a section for setting thresholds.

# Setting up a Streaming API



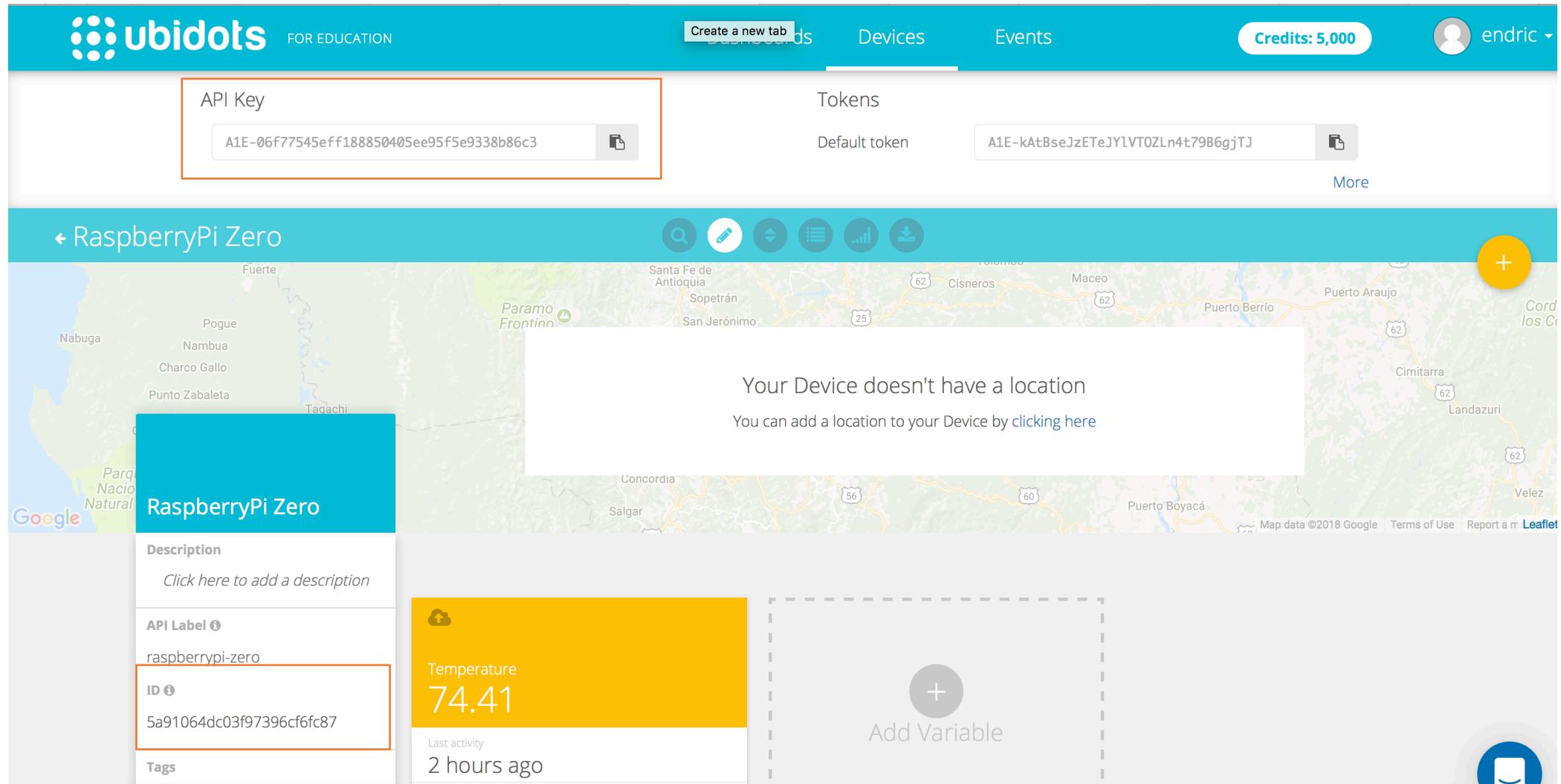
The screenshot shows the Ubidots platform interface for managing devices. The top navigation bar includes the Ubidots logo (with 'FOR EDUCATION' text), 'Dashboards', 'Devices' (which is the active tab), 'Events', 'Credits: 5,000', and a user profile for 'endric'. Below the navigation is a search bar and a 'My Devices' section.

**My Devices**

- RaspberryPi Zero**  
2 Hours Ago  
1 Variable
- Demo**  
No Last Activity  
1 Variable

A large yellow circular button with a '+' sign is located in the bottom right corner of the device list area. A dashed rectangular box surrounds this button and contains the text 'Add Device' with a central '+' icon.

# Setting up a Streaming API



The screenshot shows the Ubidots platform interface for setting up a streaming API. The top navigation bar includes 'ubidots FOR EDUCATION', 'Create a new tab', 'Dashboards', 'Devices' (selected), 'Events', 'Credits: 5,000', and a user profile for 'endric'. The main area displays a map of a region in Colombia with a device card for 'RaspberryPi Zero' overlaid. The device card contains the following information:

- Description:** Click here to add a description
- API Label:** raspberrypi-zero
- ID:** 5a91064dc03f97396cf6fc87
- Tags:** (empty)

A yellow summary box shows the current temperature reading of 74.41, with a note that the last activity was 2 hours ago. To the right, there's a dashed box labeled 'Add Variable' with a plus sign icon. A message on the map states: "Your Device doesn't have a location. You can add a location to your Device by [clicking here](#)".

# Python Code

## 🔗 Add OneWire support

---

Start by adding the following line to **/boot/config.txt**

You can edit that file with nano by running **sudo nano /boot/config.txt** and then scrolling to the bottom and typing it there

[!\[\]\(ef62519991500c3a77af2e8766280b93\_img.jpg\) Download file](#)[Copy Code](#)

```
1. dtoverlay=w1-gpio
```

# Python Code

```
GNU nano 2.7.4                                         File: DS18B20.py

import os
import glob
import time
from ubidots import ApiClient

api = ApiClient("A1E-06f77545eff188850405ee95f5e9338b86c3")
Temperature=api.get_variable("5a91065ac03f973989a988b3")

os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

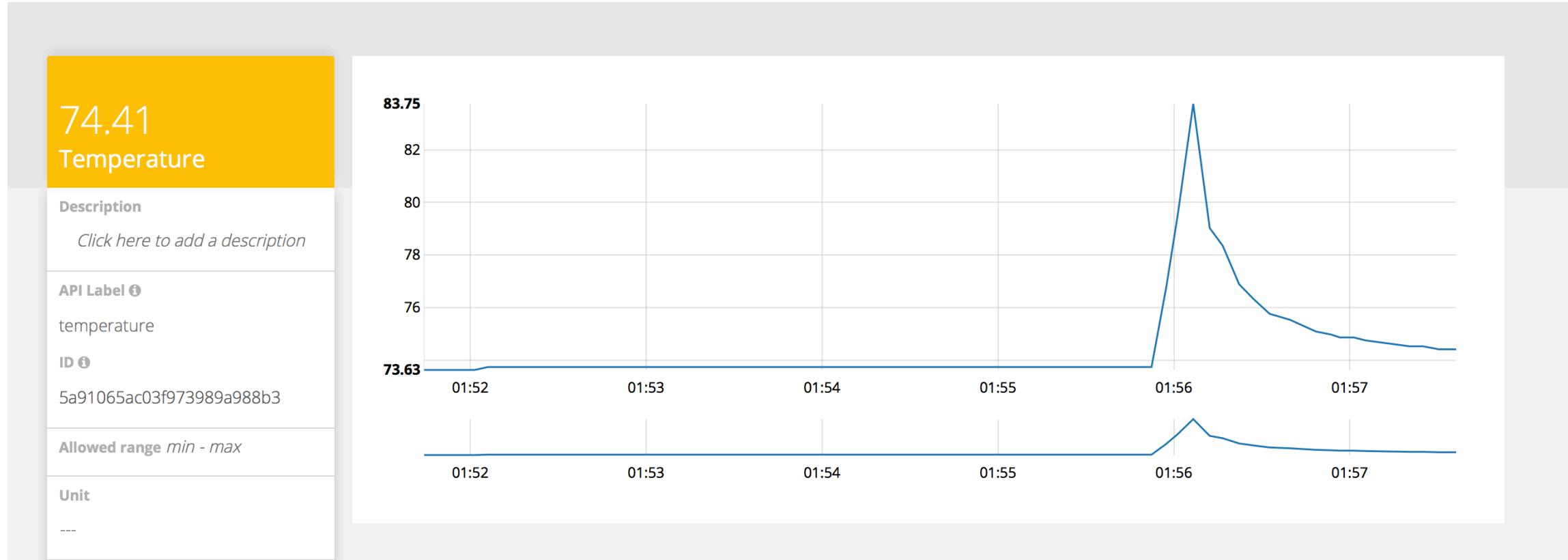
base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'

def read_temp_raw():
    f = open(device_file, 'r')
    lines = f.readlines()
    f.close()
    return lines

def read_temp():
    lines = read_temp_raw()
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines = read_temp_raw()
    equals_pos = lines[1].find('t=')
    if equals_pos != -1:
        temp_string = lines[1][equals_pos+2:]
        temp_c = float(temp_string) / 1000.0
        temp_f = temp_c * 9.0 / 5.0 + 32.0
    return temp_f

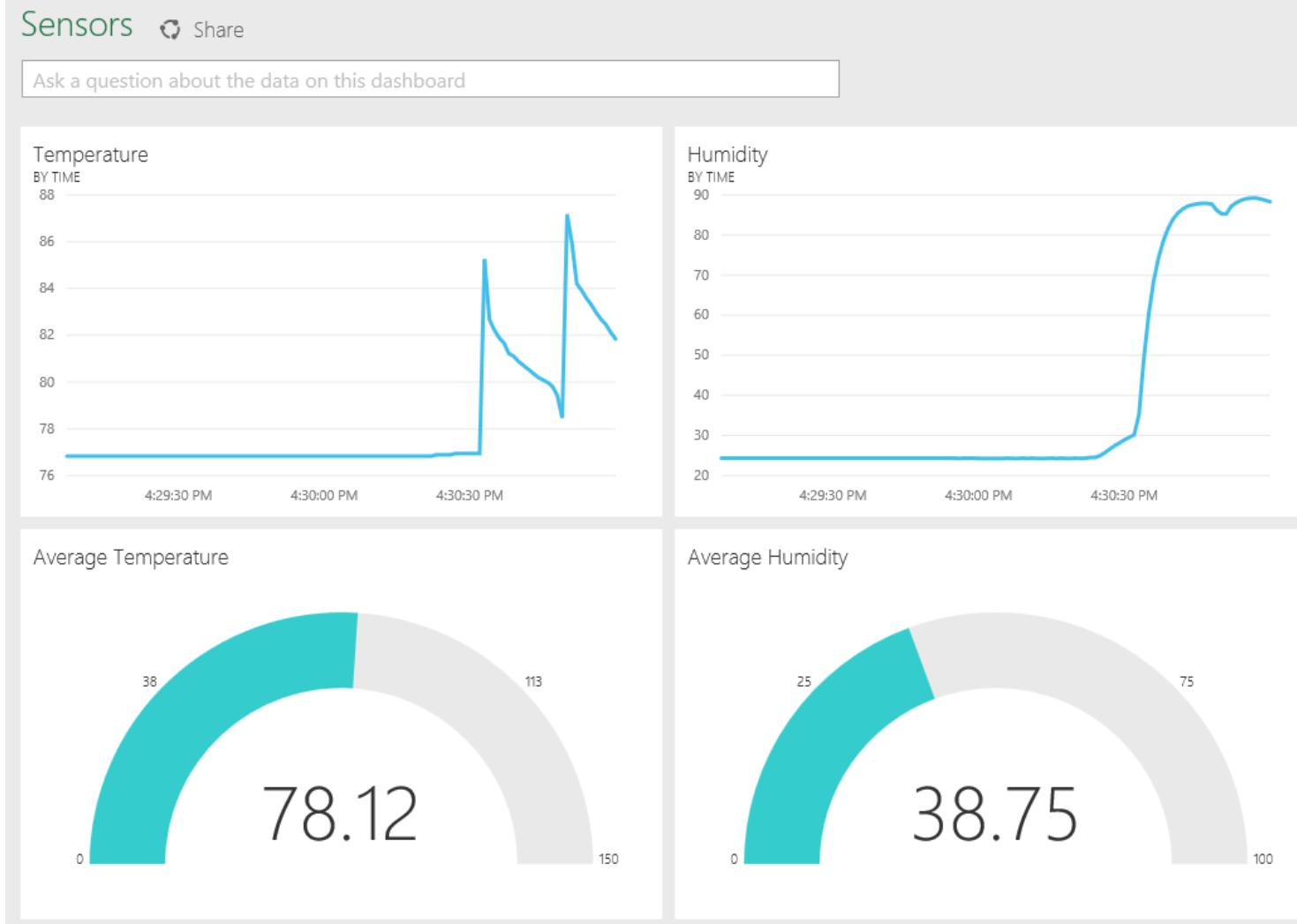
while True:
    #print(read_temp())
    Temperature.save_value({'value':read_temp()})
    time.sleep(1)
```

# Streaming Data



Next step: generate a timestamp in the Python program!

# Streaming Data using Microsoft PowerBI



 Streaming dataset created

The schema for Weather station is created.

Push URL

`https://api.powerbi.com/beta/e86e67d0-2b0b-41da-b28f-9ebb9c289183/dat`

Raw

cURL

PowerShell

```
[  
  {  
    "timestamp" : "2016-09-10T01:31:11.439Z",  
    "temperature" : 98.6,  
    "humidity" : 98.6  
  }  
]
```

Always try to generate the timestamp with the sensor data, as done here!