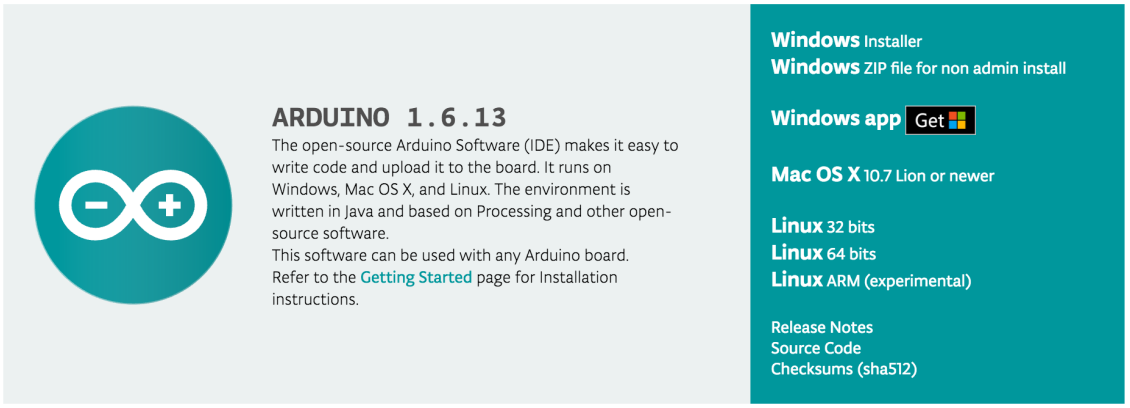
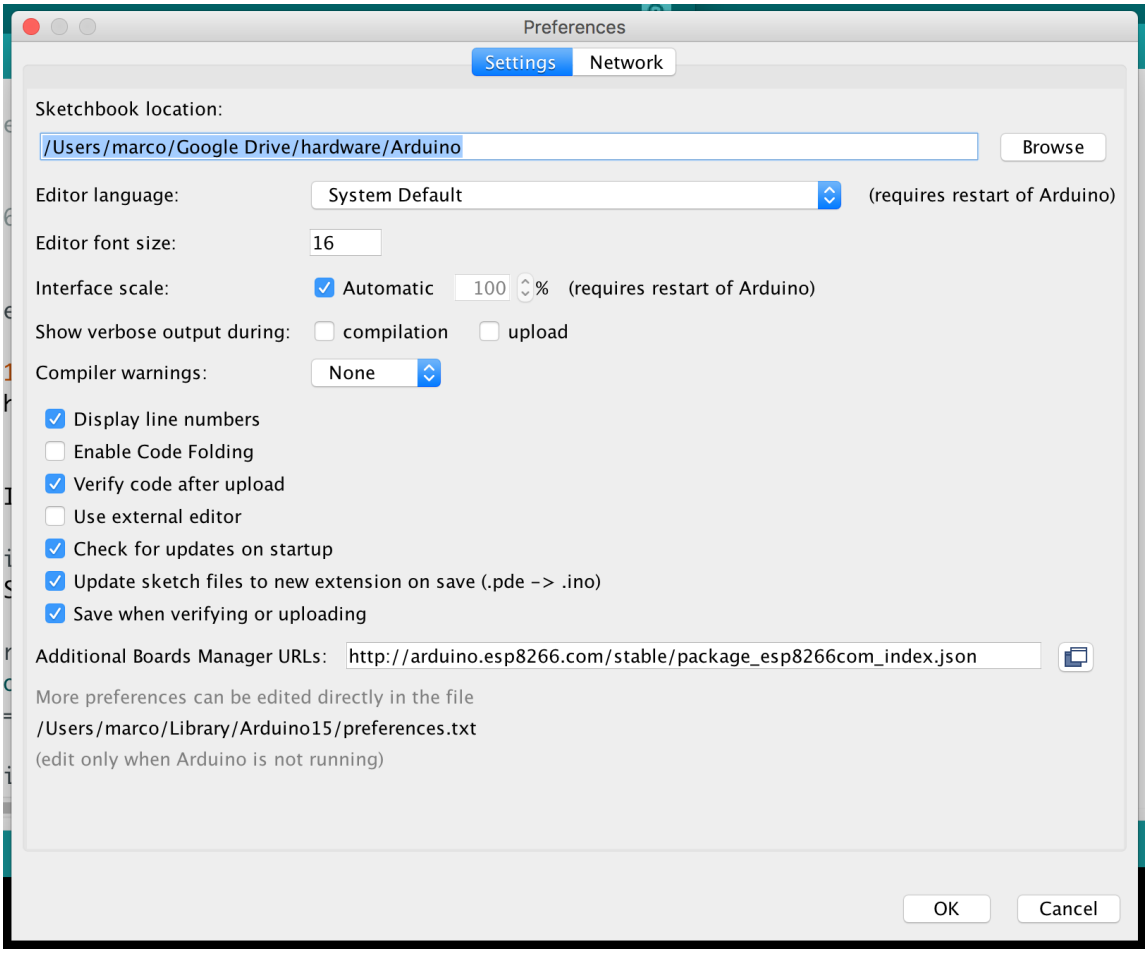
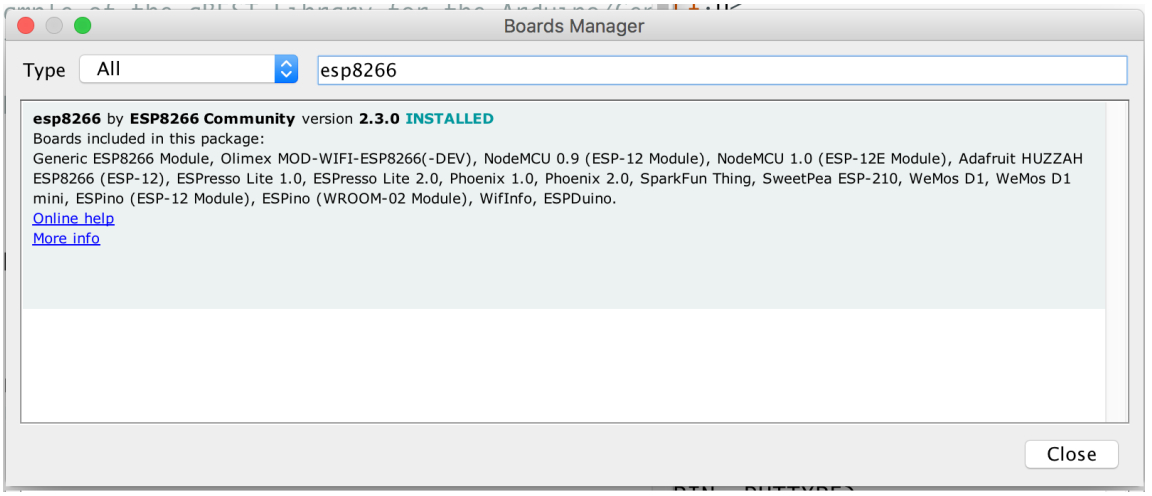
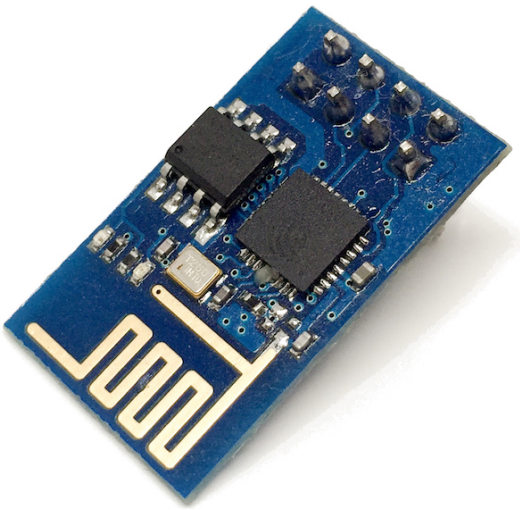
Chapter 1: Configuring the ESP8266

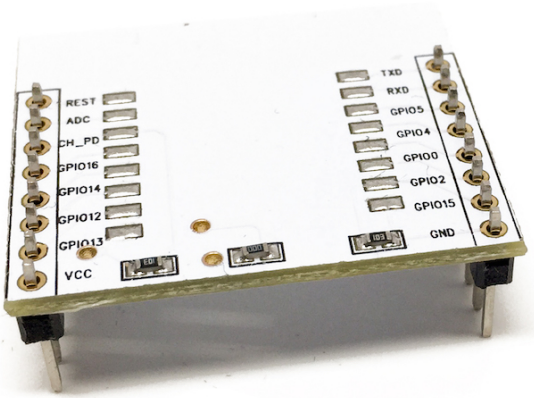
Download the Arduino Software

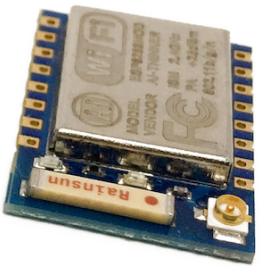


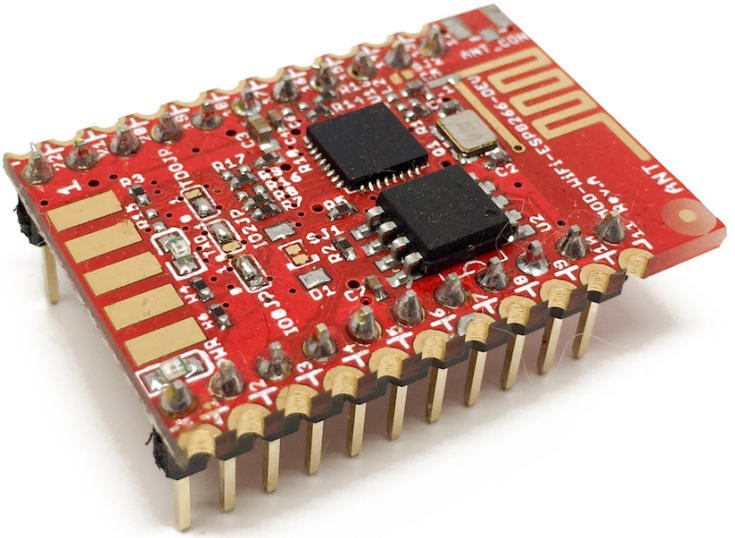

Preferences
Settings
Network
Sketchbook location:
/Users/marco/Google Drive/hardware/Arduino
Browse
Editor language:
System Default
(requires restart of Arduino)
Editor font size:
16
Interface scale:
Automatic
100
%
(requires restart of Arduino)
Show verbose output during:
compilation
upload
Compiler warnings:
None
Display line numbers
Enable Code Folding
Verify code after upload
Use external editor
Check for updates on startup
Update sketch files to new extension on save (.pde -> .ino)
Save when verifying or uploading
Additional Boards Manager URLs:
http://arduino.esp8266.com/stable/package_esp8266com_index.json
More preferences can be edited directly in the file
/Users/marco/Library/Arduino15/preferences.txt
(edit only when Arduino is not running)
OK
Cancel

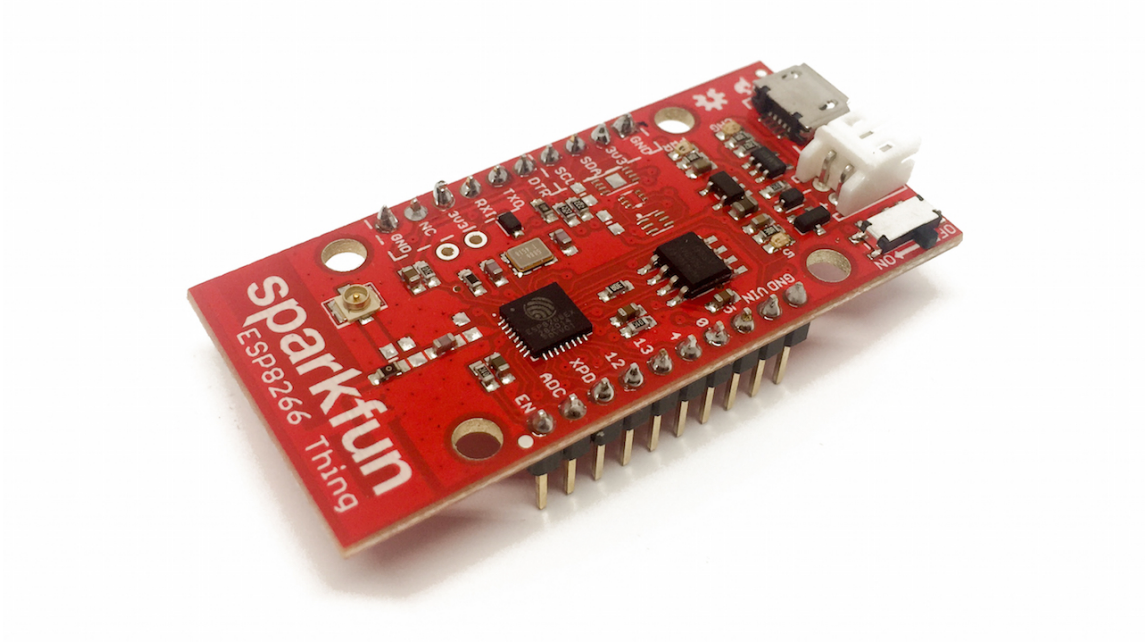

Boards Manager
Type
All
esp8266
esp8266 by ESP8266 Community version 2.3.0 INSTALLED
Boards included in this package:
Generic ESP8266 Module, Olimex MOD-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Adafruit HUZZAH ESP8266 (ESP-12), ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, SparkFun Thing, SweetPea ESP-210, WeMos D1, WeMos D1 mini, ESPino (ESP-12 Module), ESPino (WROOM-02 Module), WifInfo, ESPDuino.
Online help
More info
Close

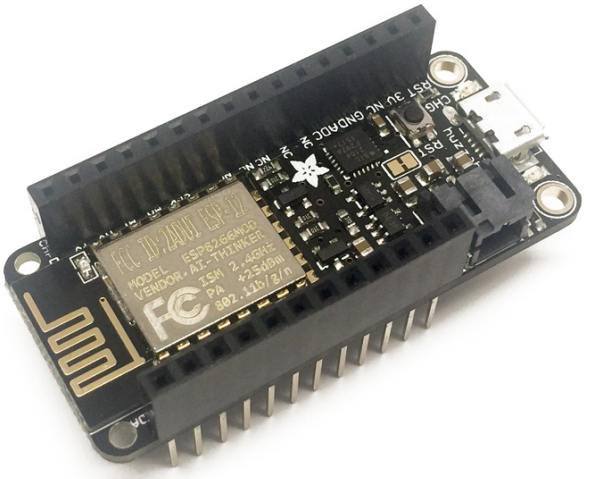


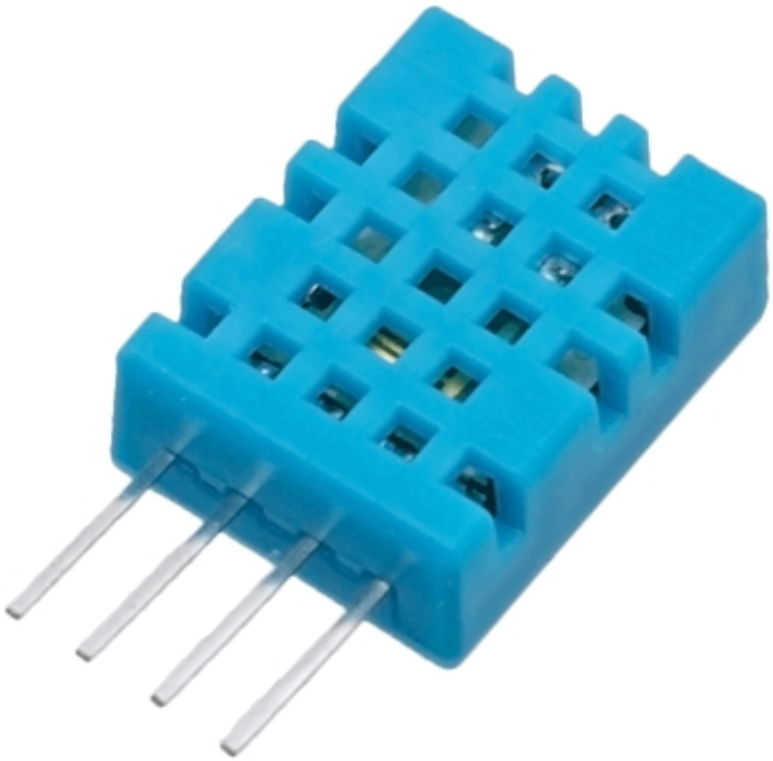

REST
ADC
CH_PD
GPIO16
GPIO14
GPIO12
GPIO13
VCC
TXD
RXD
GPIO5
GPIO4
GPIO0
GPIO2
GPIO15
GND


Ra1nsun


ANT


sparkfun
ESP8266 Thing

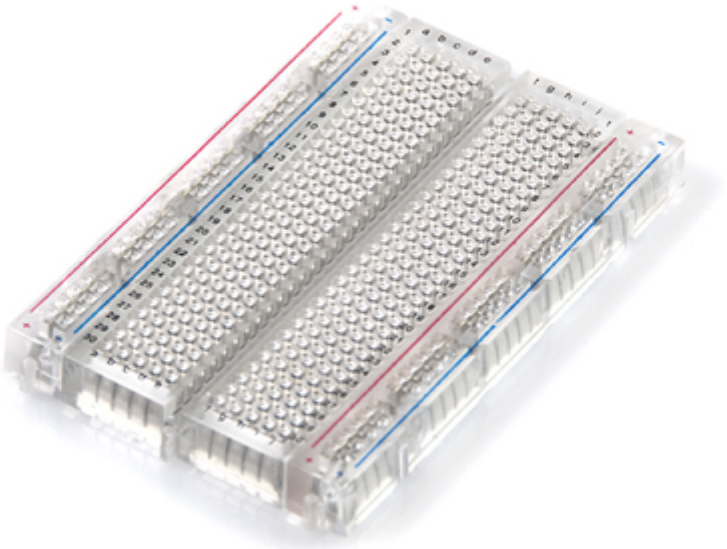


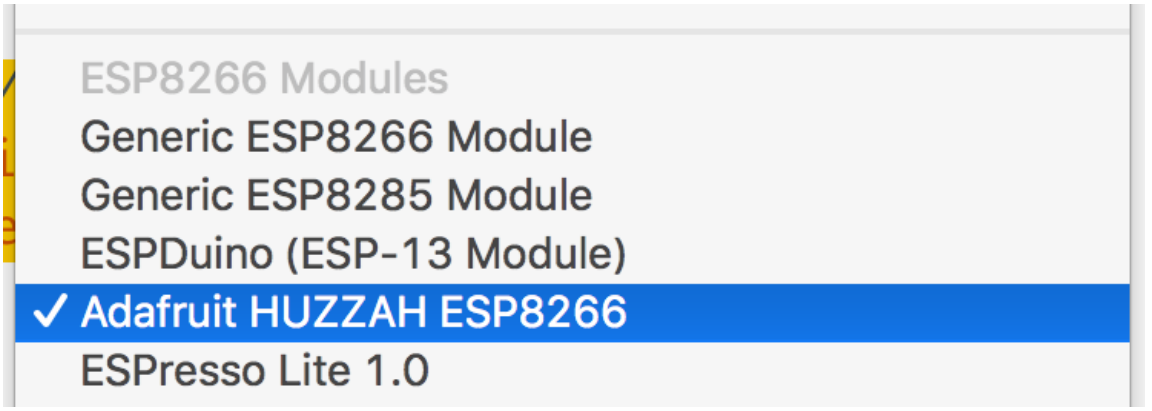


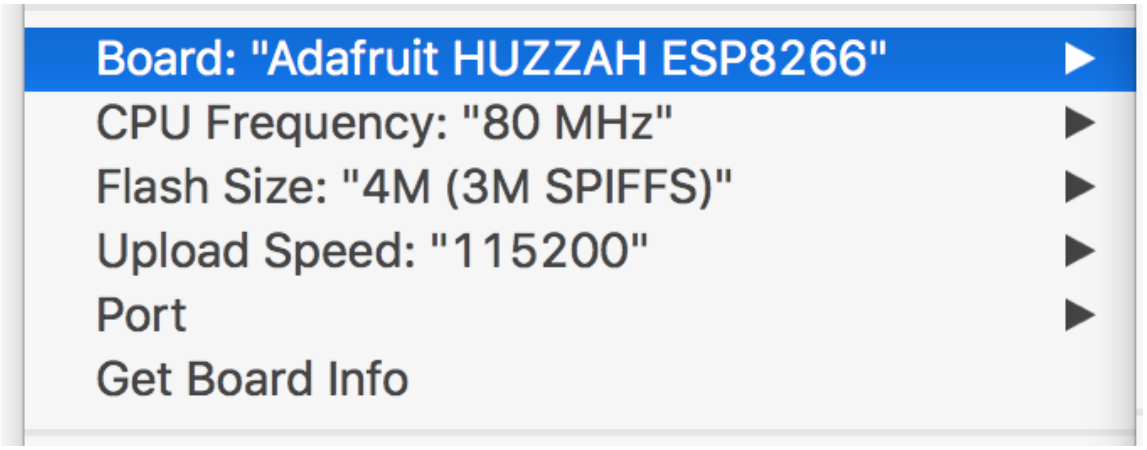

VCC GND SIG
Soil Moisture Sensor

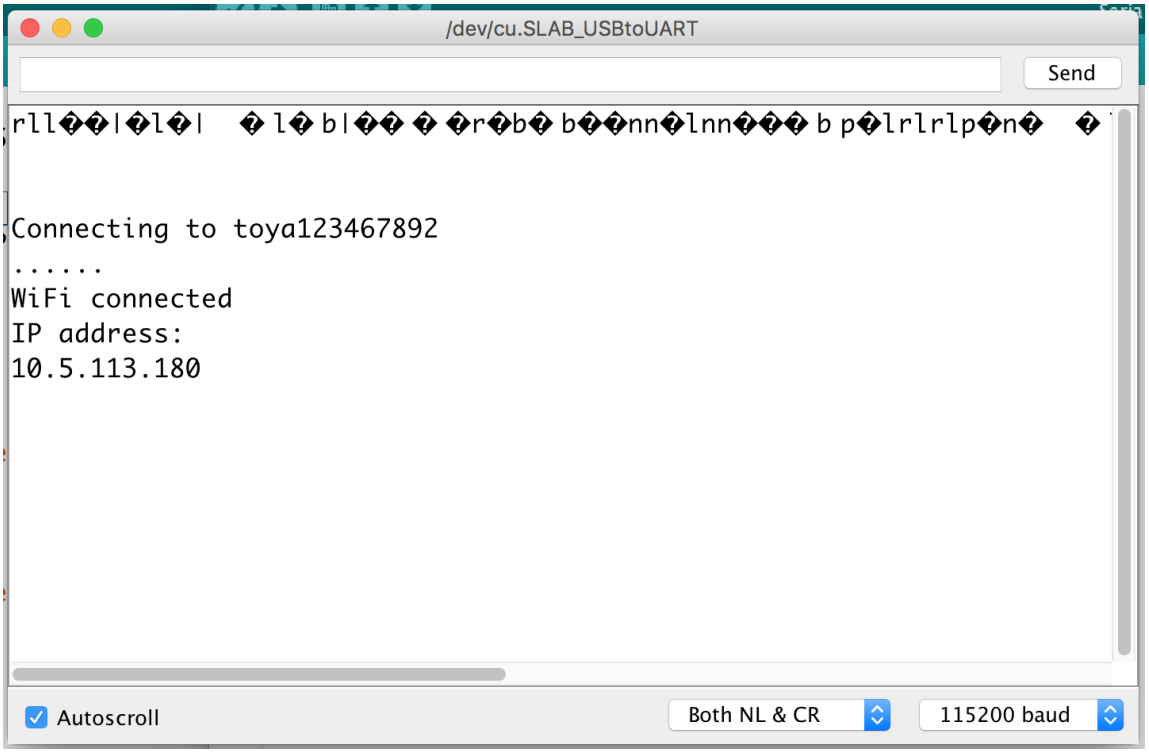

JZC-11F
005-1Z
5VDC
T-85℃ 220/250VAC5A

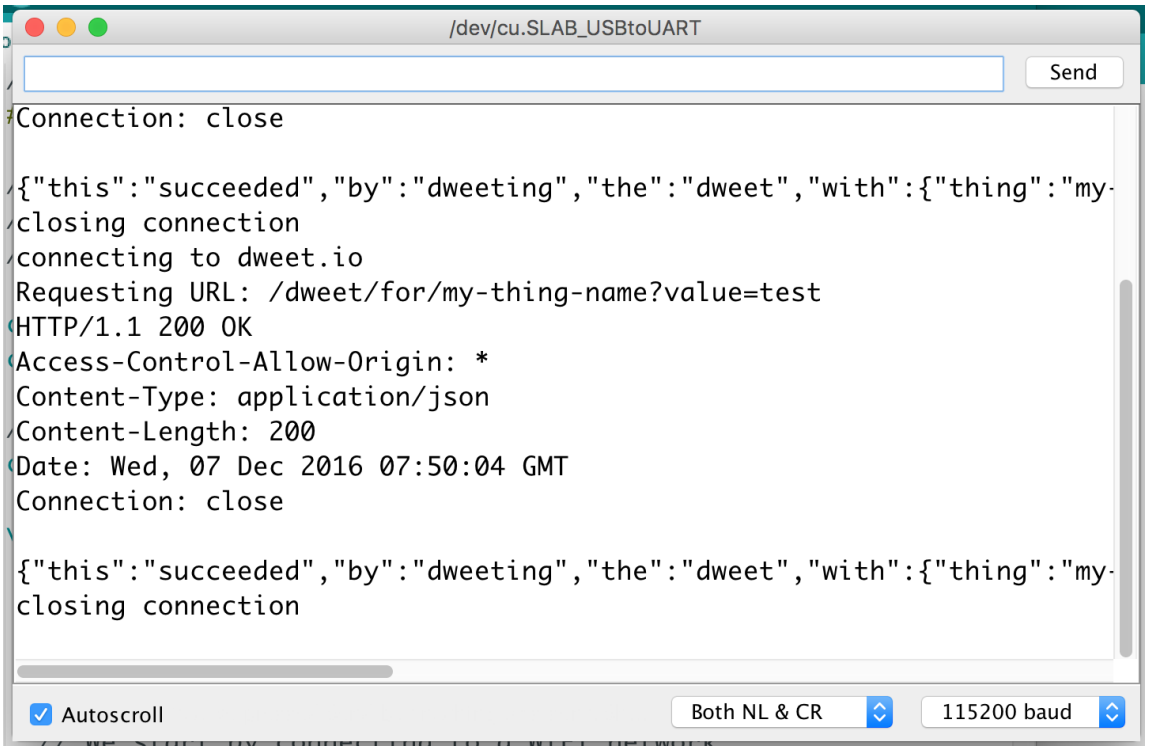

PowerSwitch Tail II
Isolated DC Activated AC Power Switch
www.PowerSwitchTail.com
1: In
2: In
3: Ground



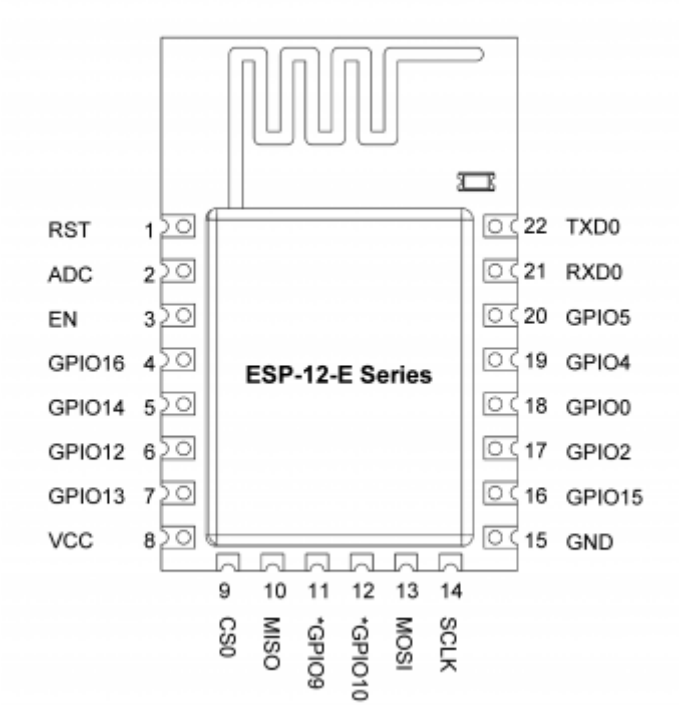

ESP8266 Modules
Generic ESP8266 Module
Generic ESP8285 Module
ESPDuino (ESP-13 Module)
✓ Adafruit HUZZAH ESP8266
ESPresso Lite 1.0

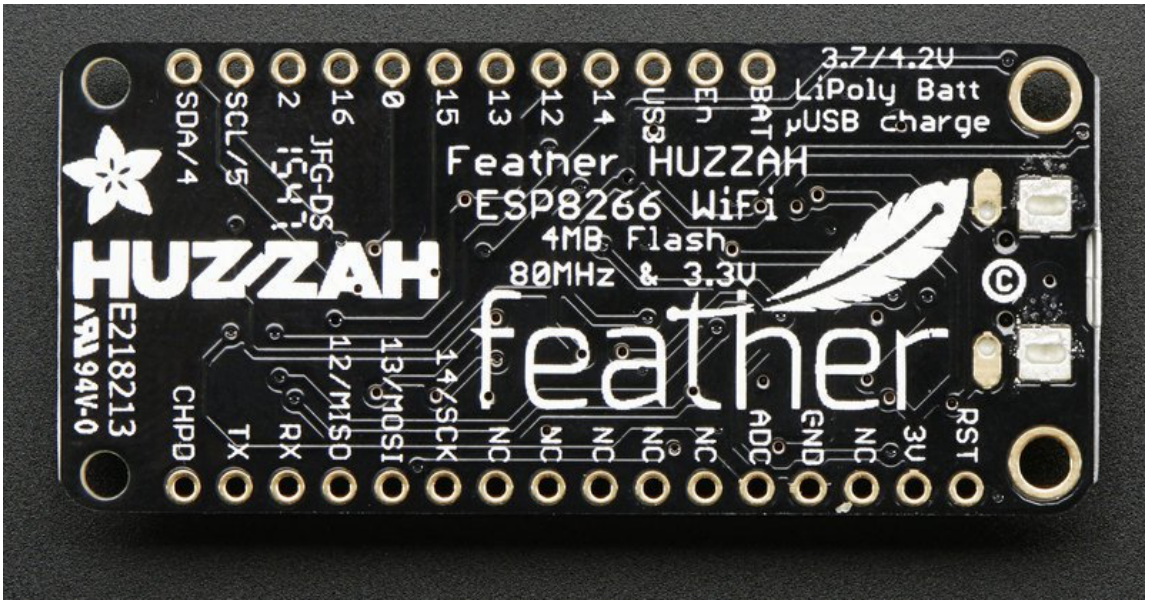

Board: "Adafruit HUZZAH ESP8266"
CPU Frequency: "80 MHz"
Flash Size: "4M (3M SPIFFS)"
Upload Speed: "115200"
Port
Get Board Info

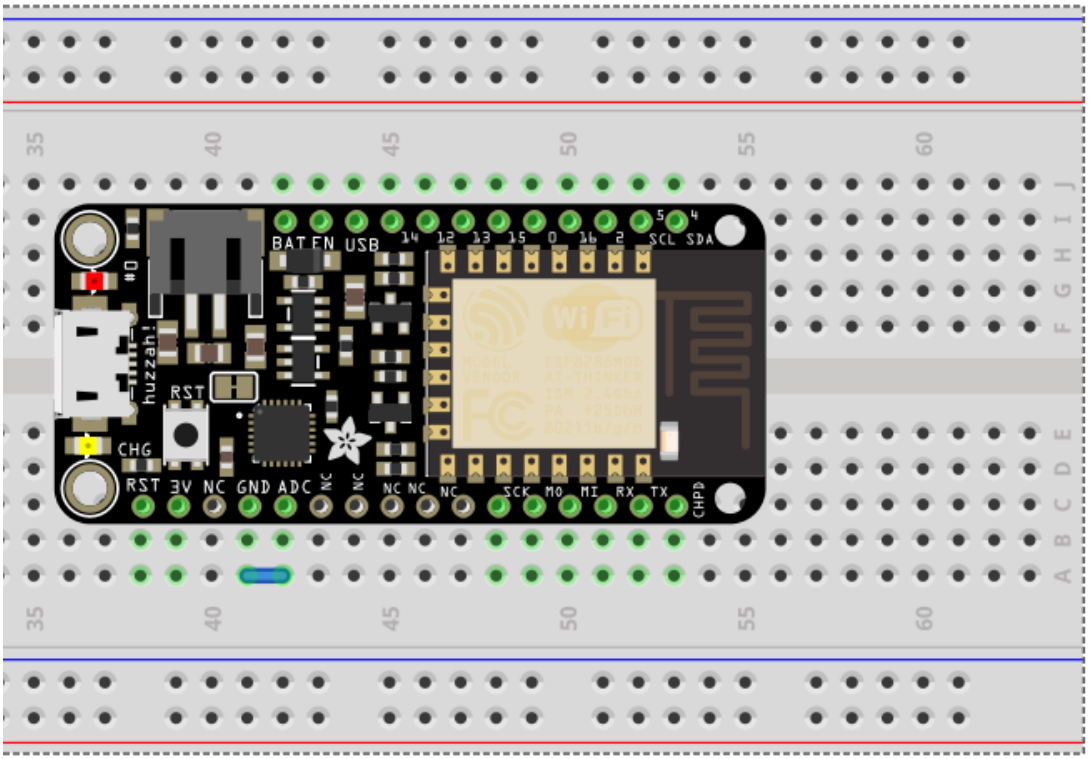

/dev/cu.SLAB_USBtoUART
Send
rll��|�l�| l b| r b b nn lnn b p lrlrlp n
Connecting to toya123467892
.....
WiFi connected
IP address:
10.5.113.180
Autoscroll
Both NL & CR
115200 baud

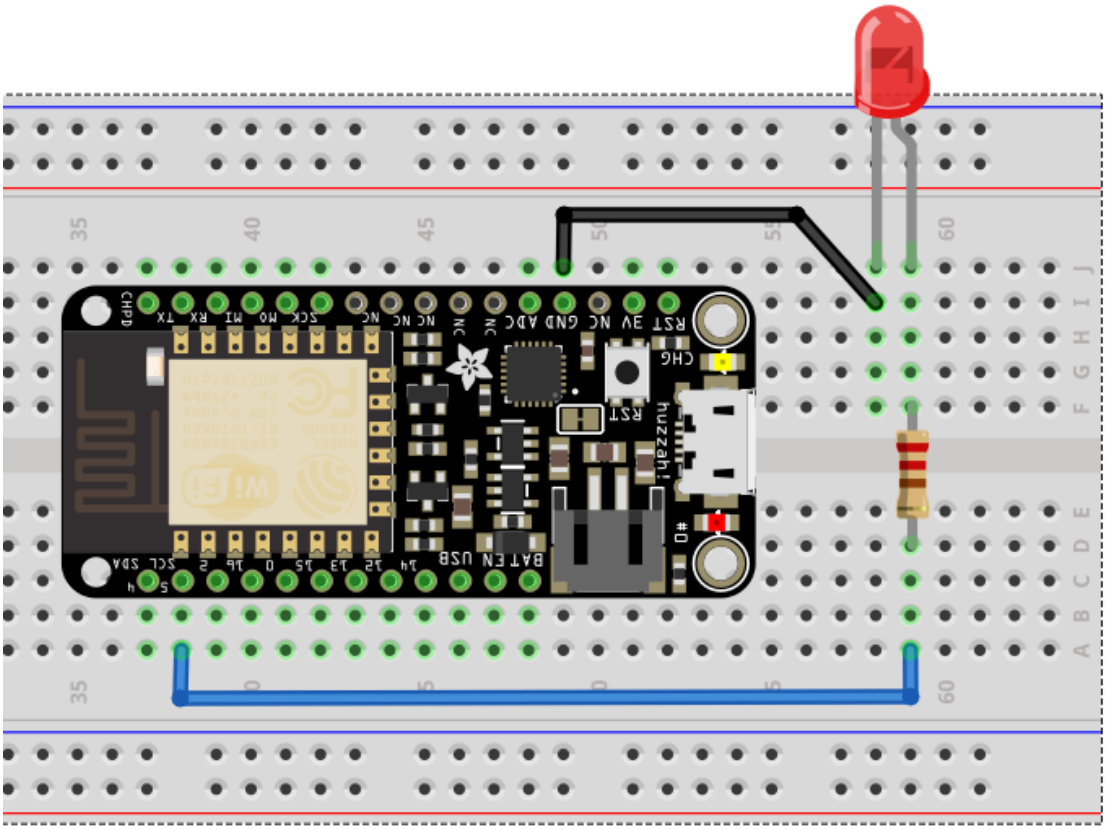

/dev/cu.SLAB_USBtoUART
Send
Connection: close
{"this":"succeeded","by":"dweeting","the":"dweet","with":{"thing":"my-
closing connection
connecting to dweet.io
Requesting URL: /dweet/for/my-thing-name?value=test
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Content-Type: application/json
Content-Length: 200
Date: Wed, 07 Dec 2016 07:50:04 GMT
Connection: close
{"this":"succeeded","by":"dweeting","the":"dweet","with":{"thing":"my-
closing connection
Autoscroll
Both NL & CR
115200 baud

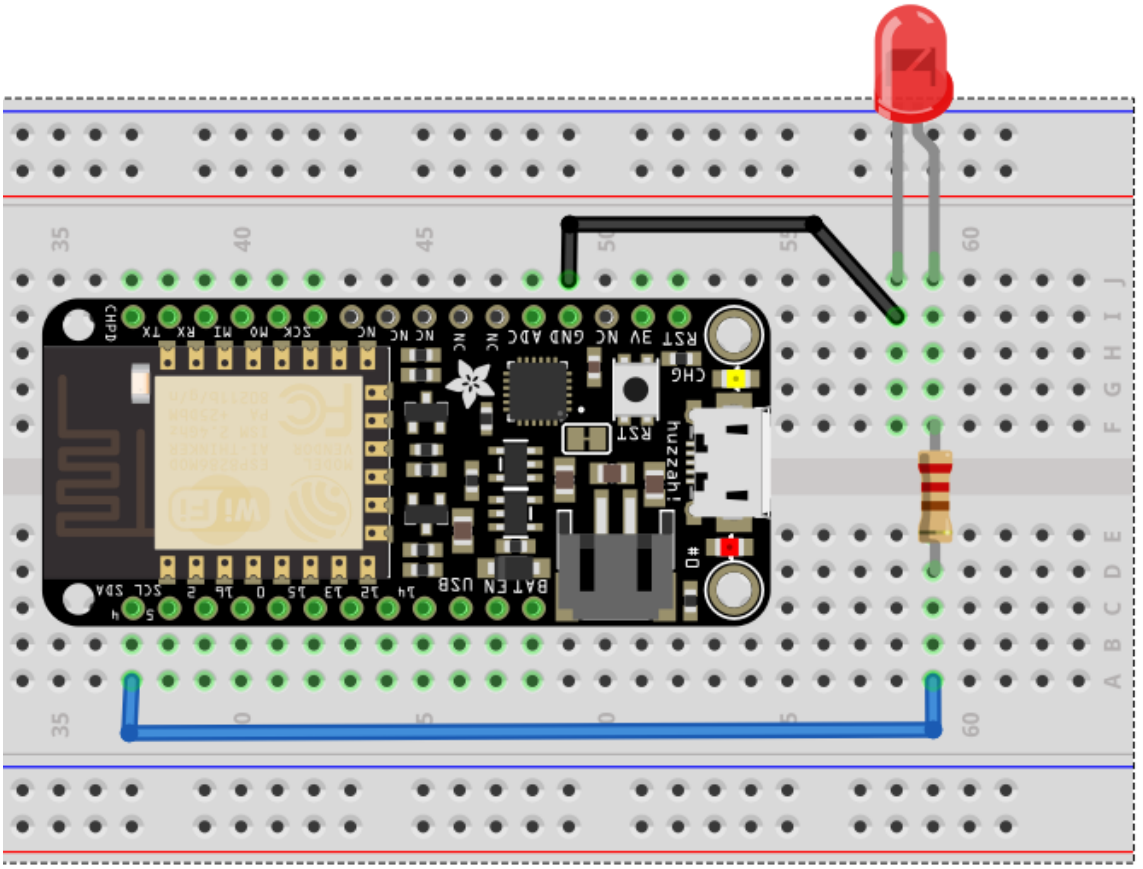
Chapter 2: Your First ESP8266 Projects

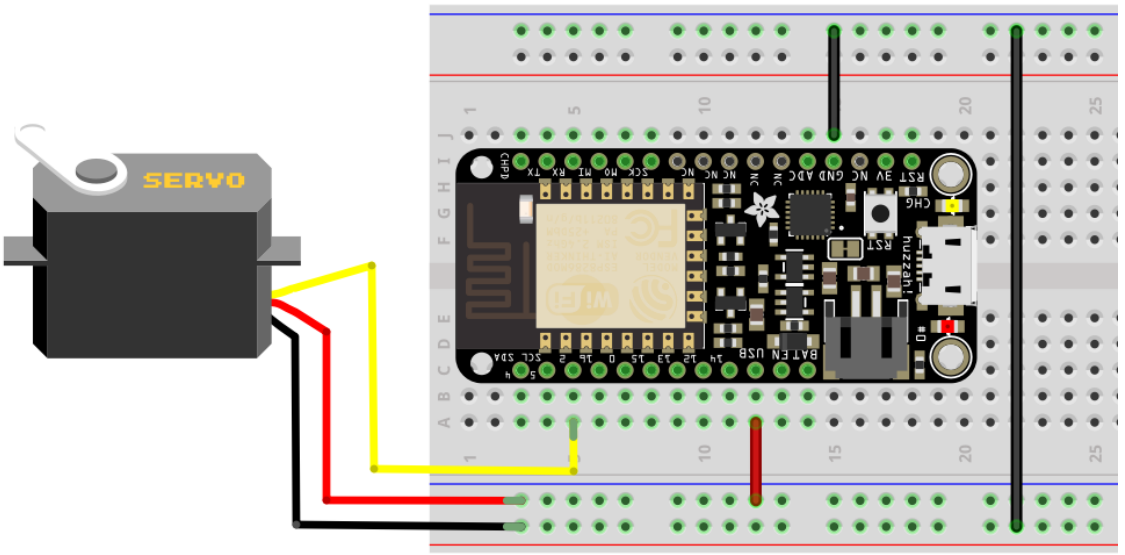


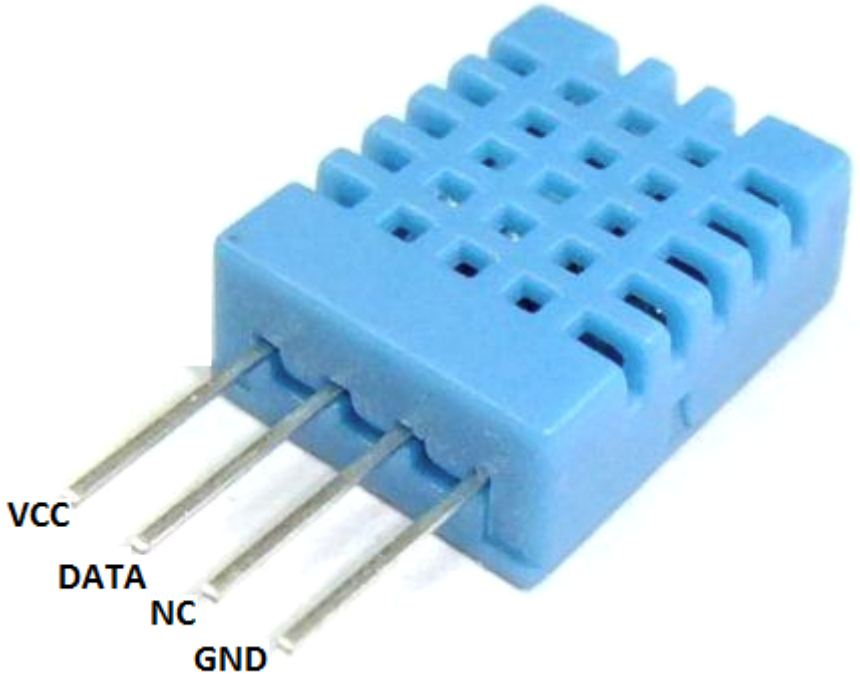

3.7/4.2V
LiPoly Batt
µUSB charge
Feather HUZZAH
ESP8266 WiFi
4MB Flash
80MHz & 3.3V
HUZZAH
feather
SDA/4
SCL/5
JFG-DS
15
13
12
14
USB
EN
BAT
16
0
2
CHPD
TX
RX
12/MISO
13/MOSI
14/SCK
NC
NC
NC
NC
NC
ADC
GND
NC
3V
RST
E218213
94V-0

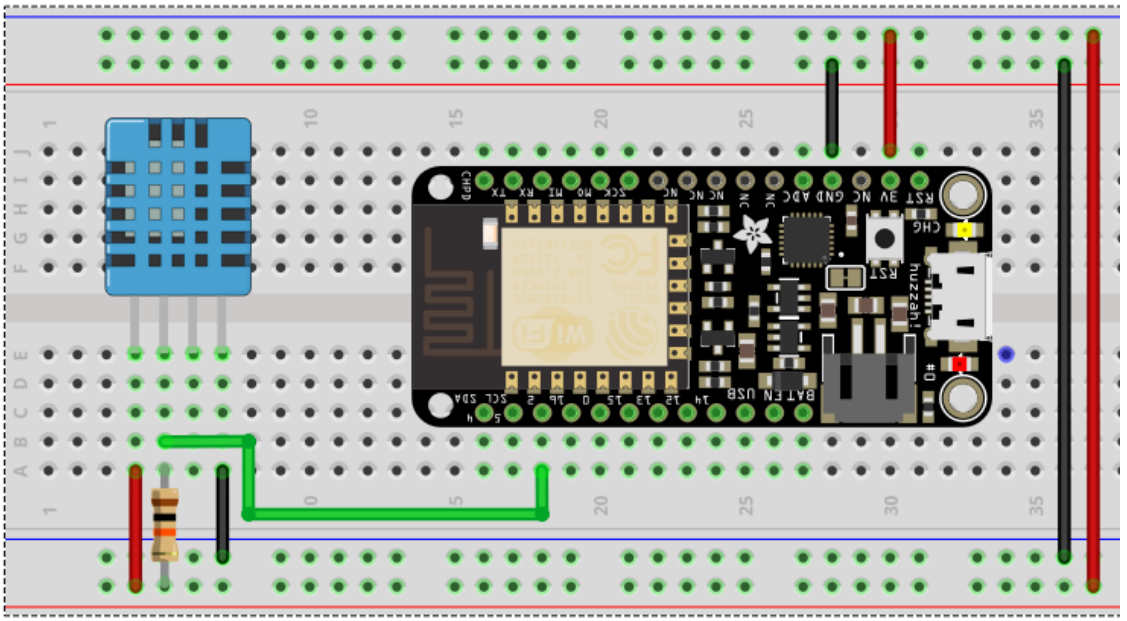

BAT EN USB
14 12 13 15 0 16 2 5 4
SCL SDA
ON
huzzah
RST
CHG
RST 3V NC GND ADC NC NC NC NC NC
SCK MO MI RX TX
CHPD
35 40 45 50 55 60
A B C D E F G H I J


CHPD TX RX MI MO SCK NC NC NC NC NC ADC GND 3V NC RST
CHG
RST
huzzah
BAT EN USB
SCL SDA

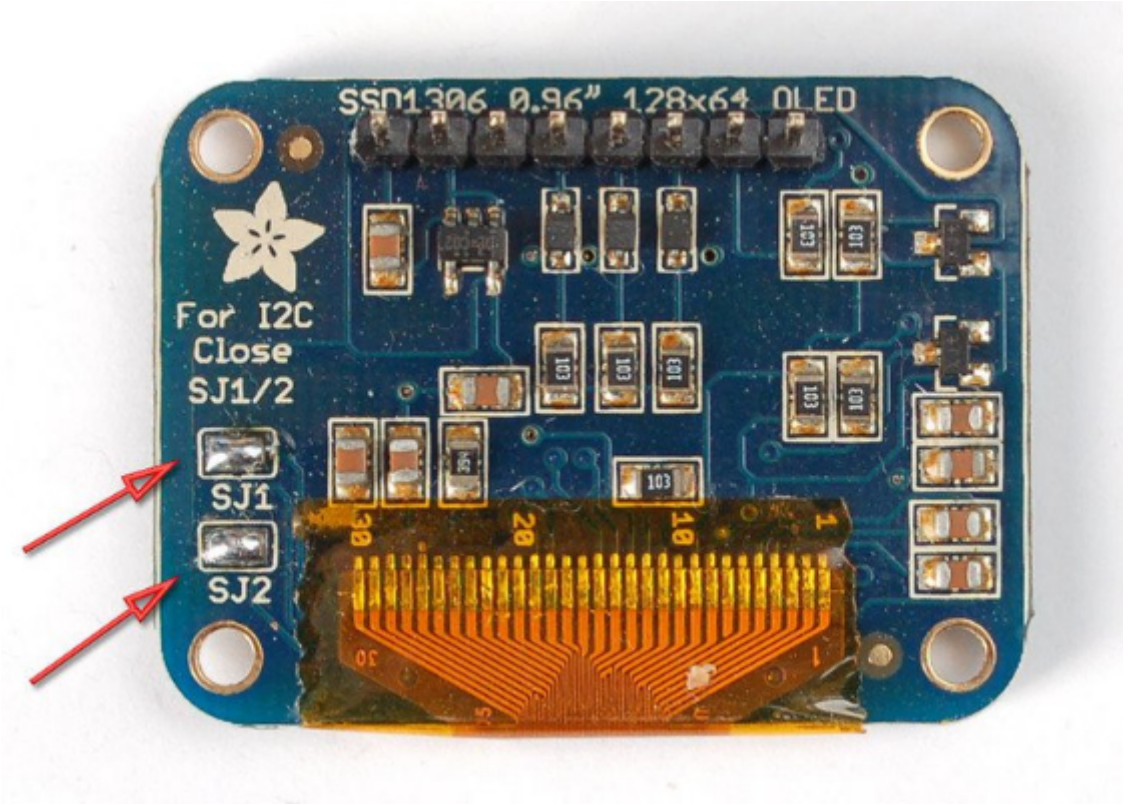

CHPD TX RX MI MO SCK NC NC NC NC NC ADC GND NC 3V RST
RST
CHG
huzzah
#D
BAT EN USB 14 12 13 15 0 16 2 SCL SDA
35 40 45 50 55 60
A B C D E F G H I J

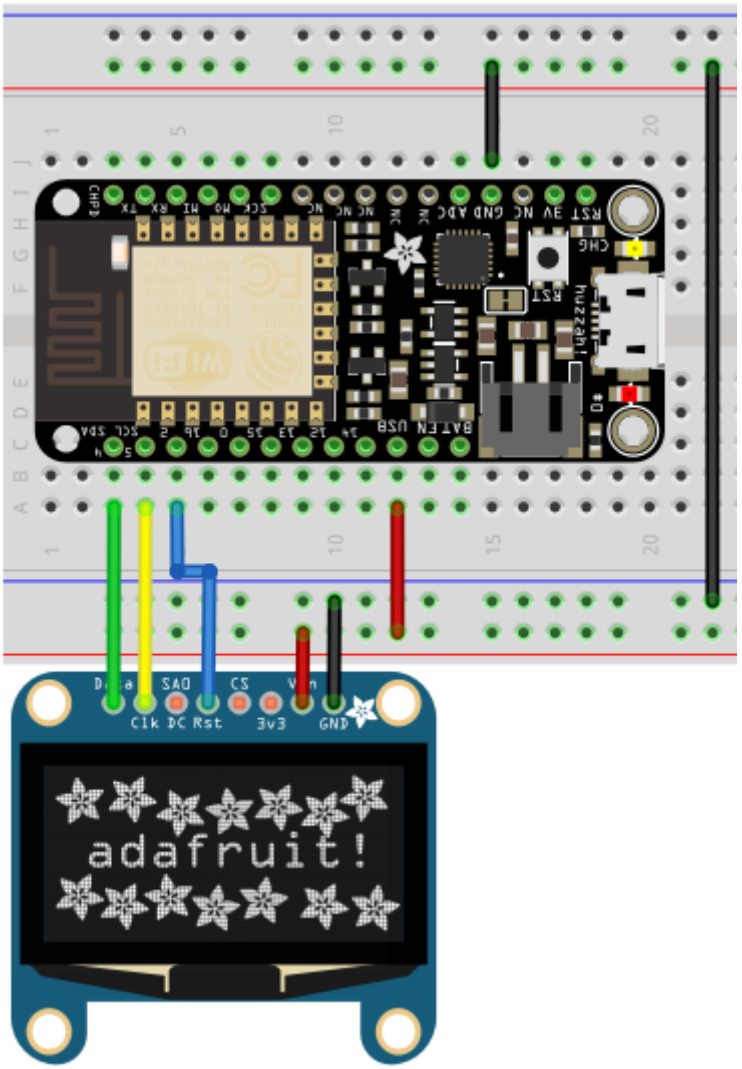

SERVO

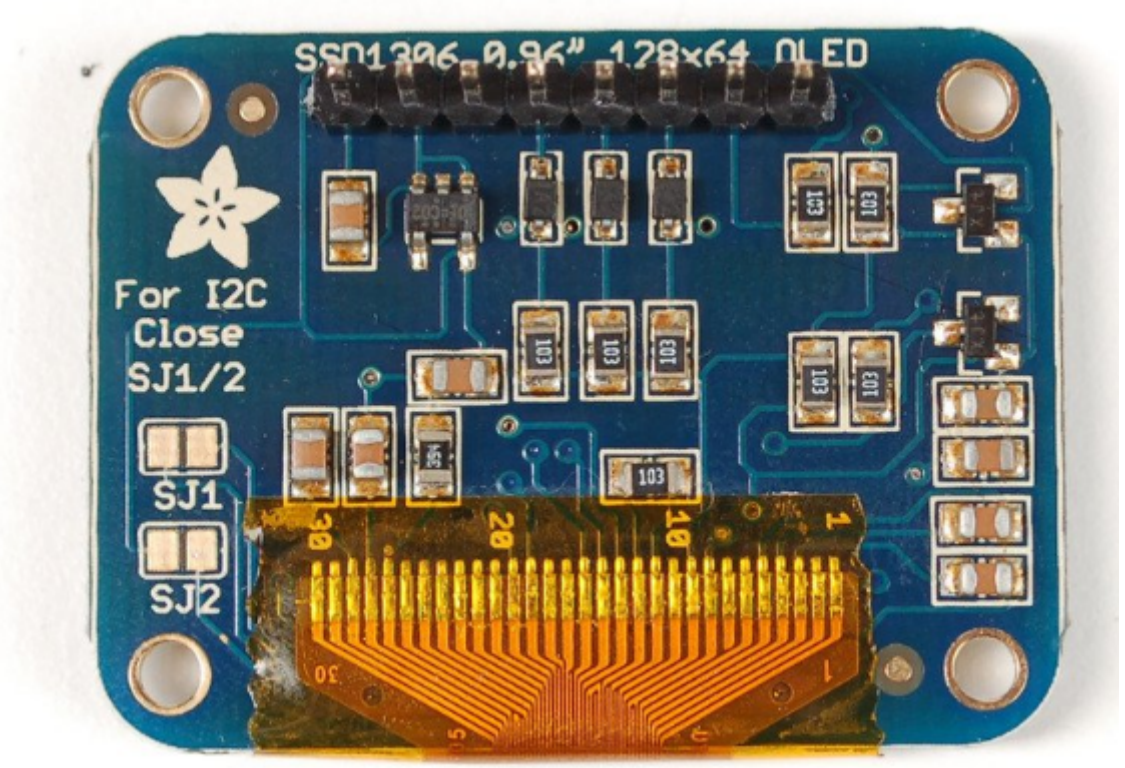

VCC
DATA
NC
GND

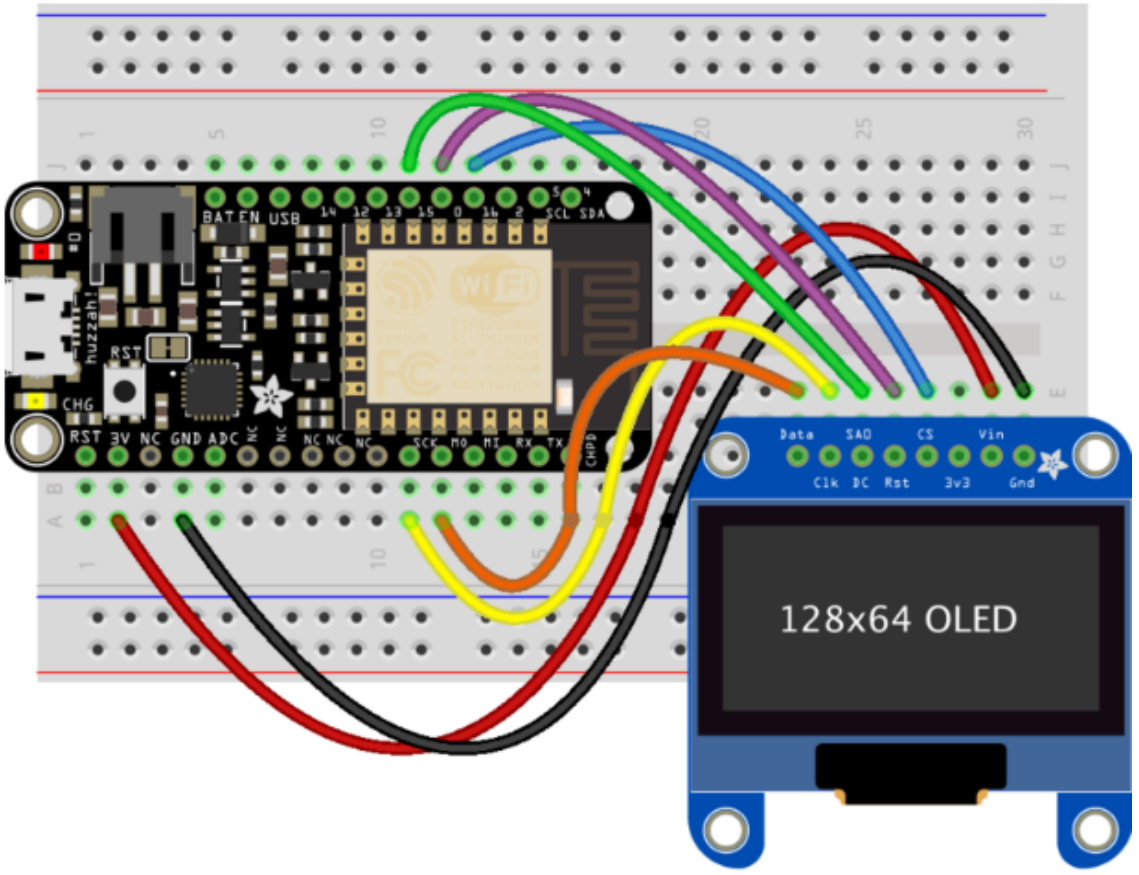



Data
SA0
CS
Vin
Clk
DC
Rst
3v3
Gnd

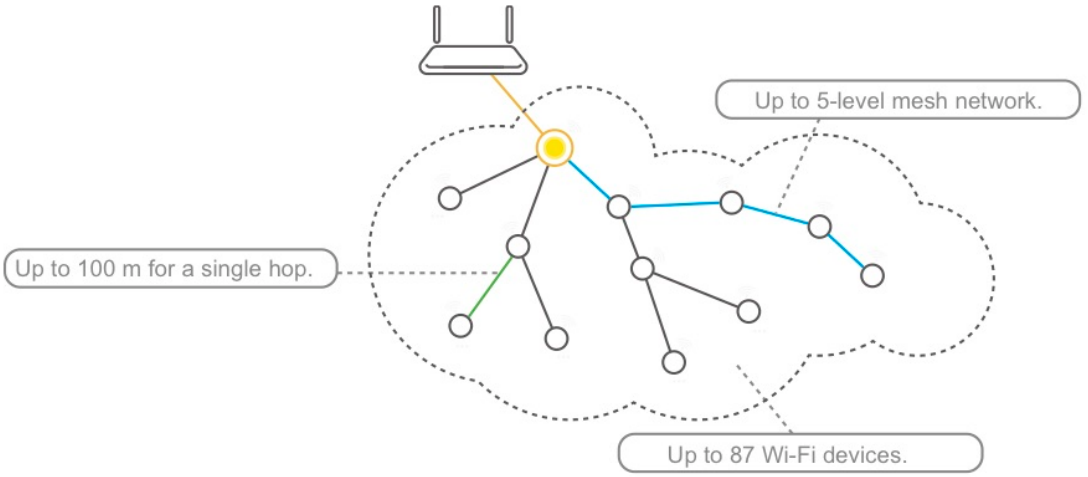

SSD1306 0.96" 128x64 OLED
For I2C
Close
SJ1/2
SJ1
SJ2

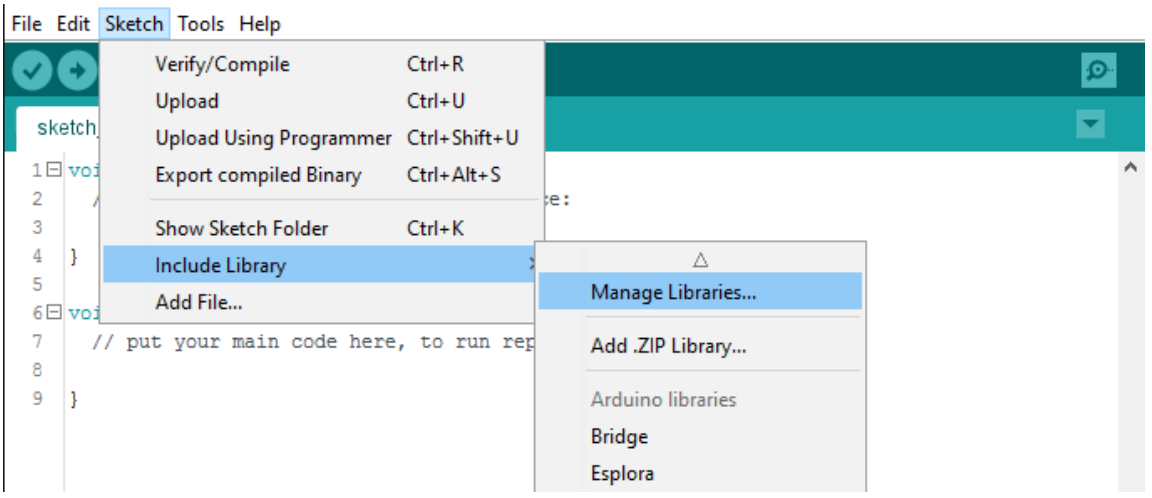

Clk DC Rst
3v3
GND
adafruit!

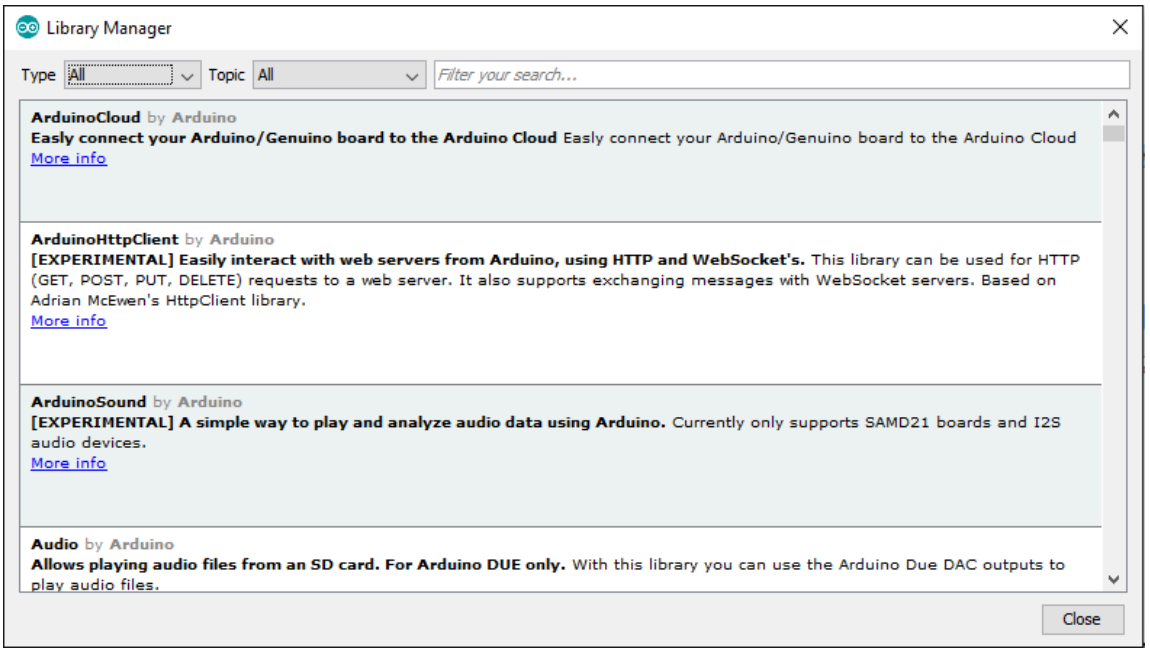

SSD1306 0.96" 128x64 OLED
For I2C
Close
SJ1/2
SJ1
SJ2
103
30
20
10
1

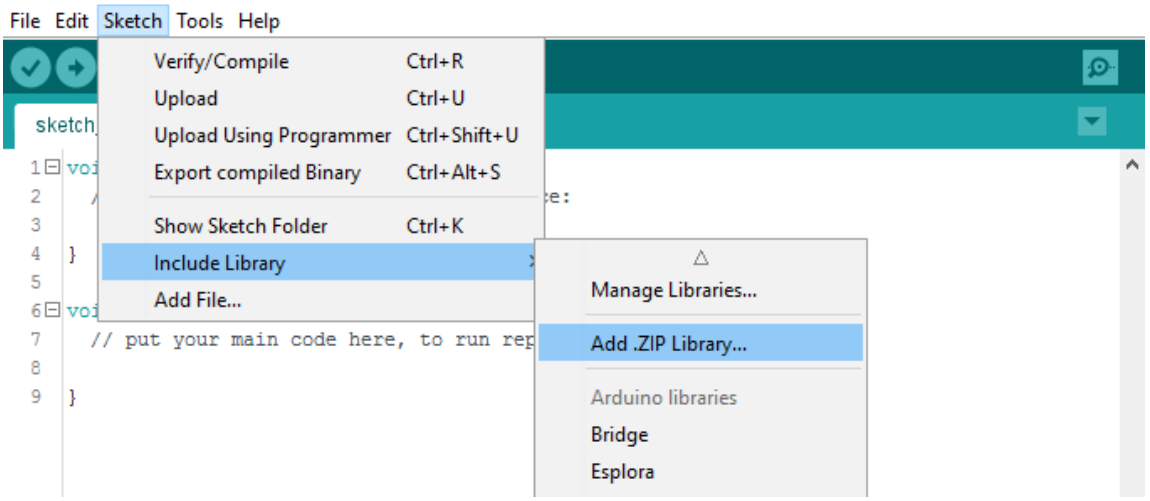

BAT EN USB
SCL SDA
RST
CHG
RST 3V NC GND ADC
SCK MO MI RX TX
Data SAO CS Vin
Clk DC Rst 3v3 Gnd
128x64 OLED

Chapter 3: More ESP8266 Functions

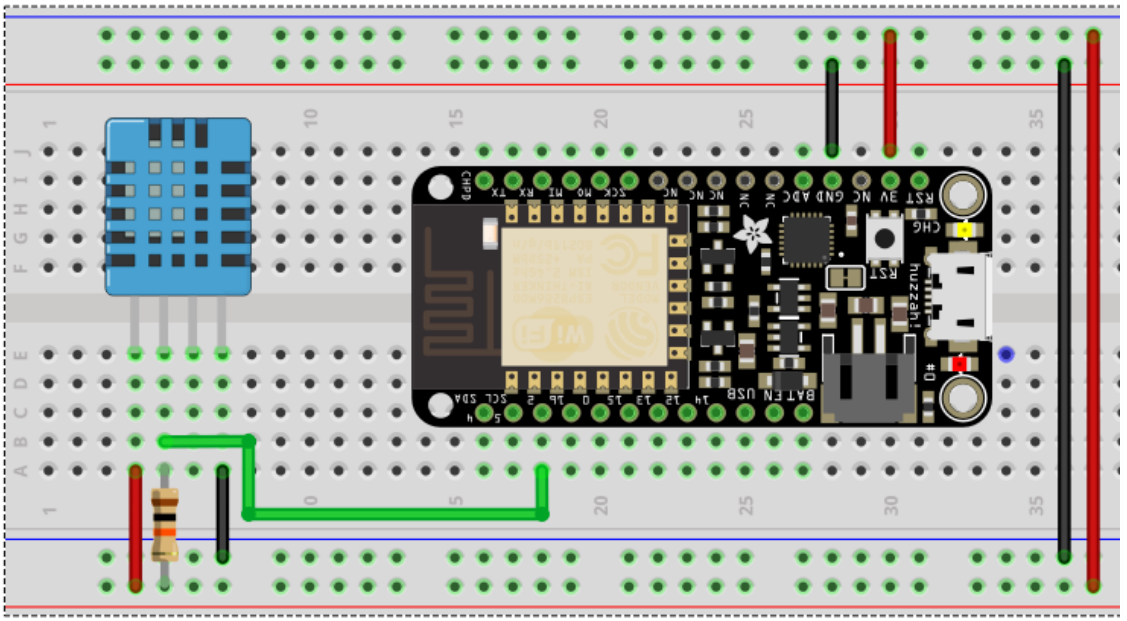


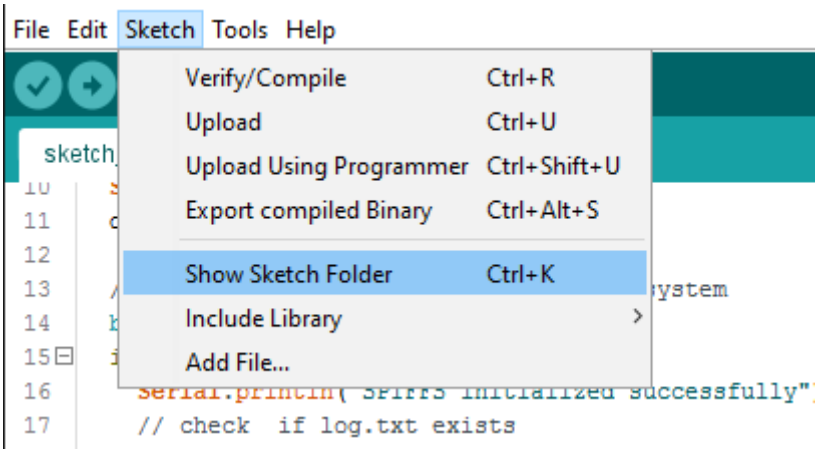


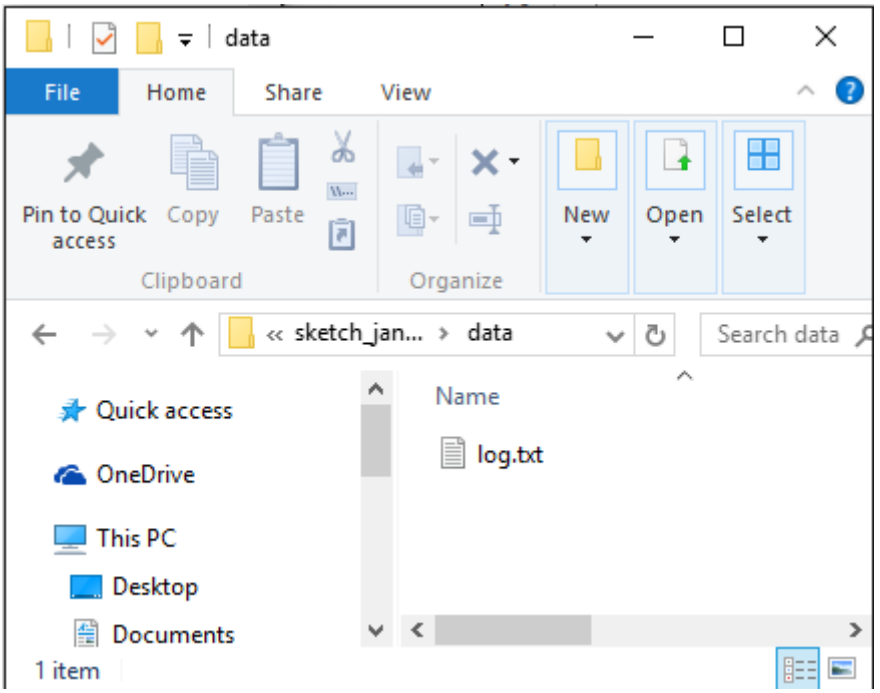


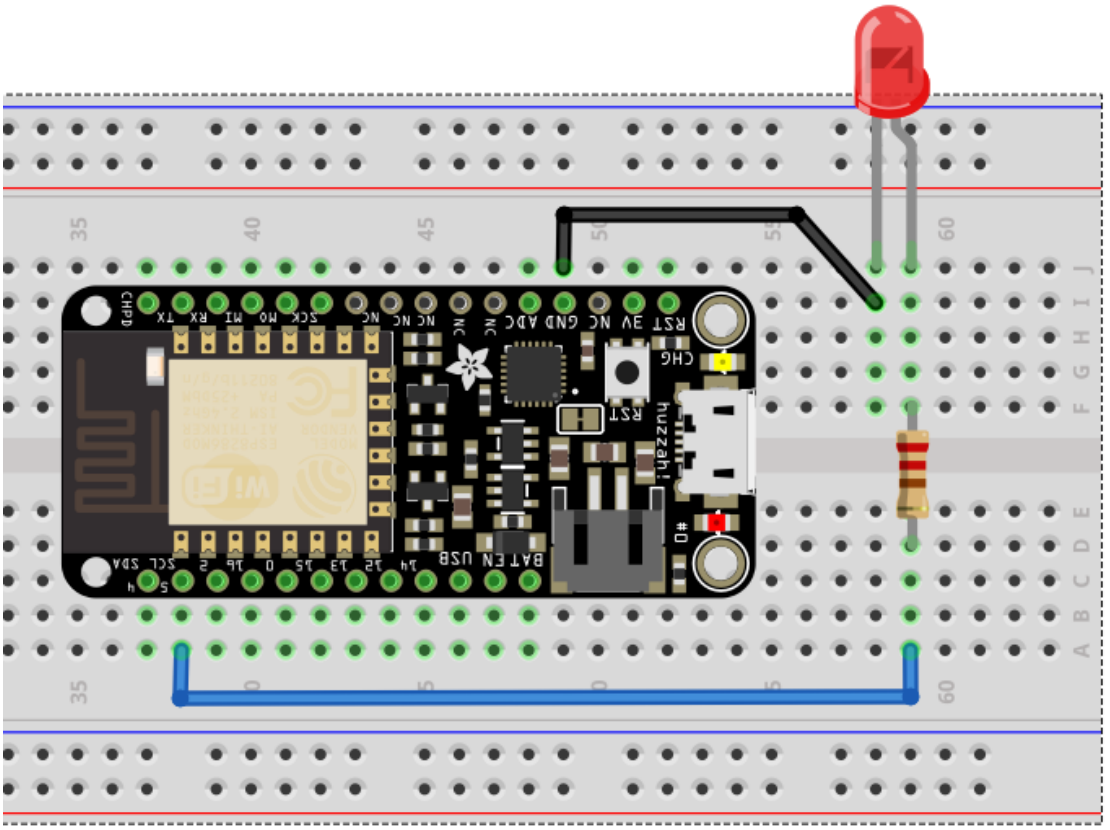



```
|--------------|-------|---------------|--|--|--|--|--|
^              ^       ^               ^     ^
Sketch    OTA update   File system   EEPROM  WiFi config (SDK)
```

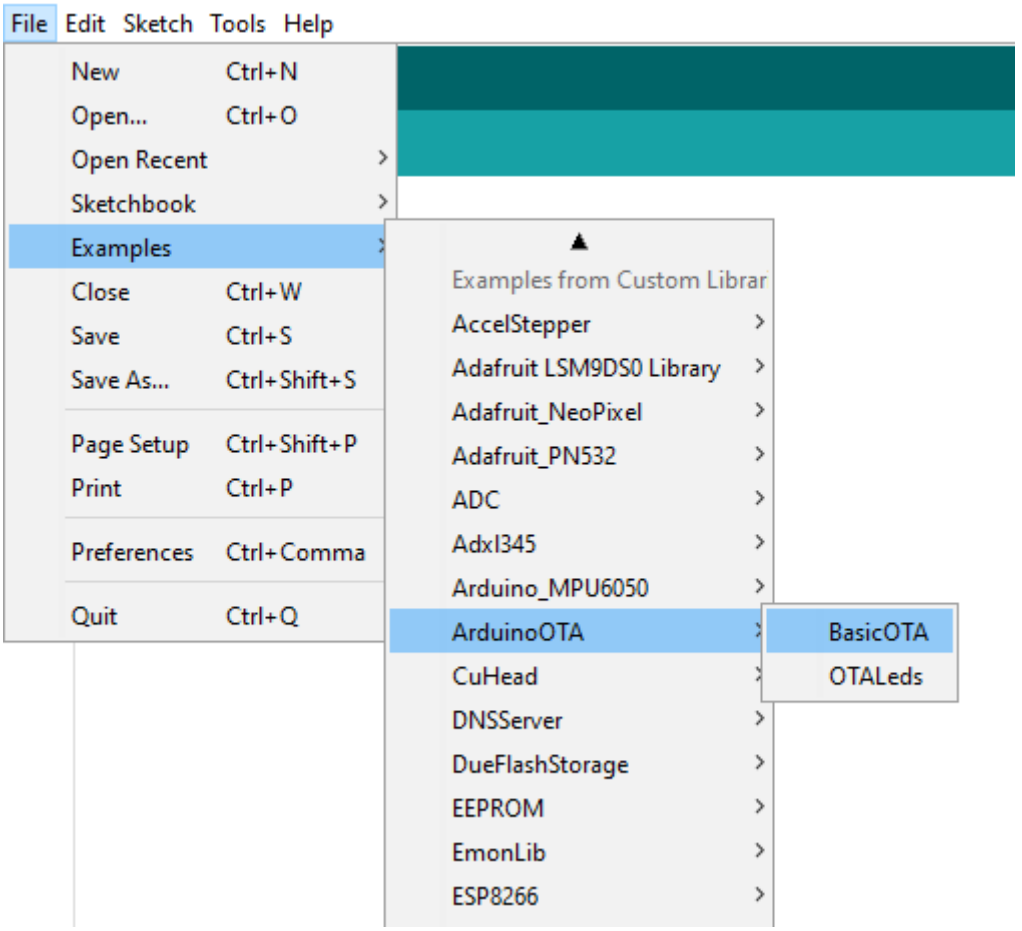


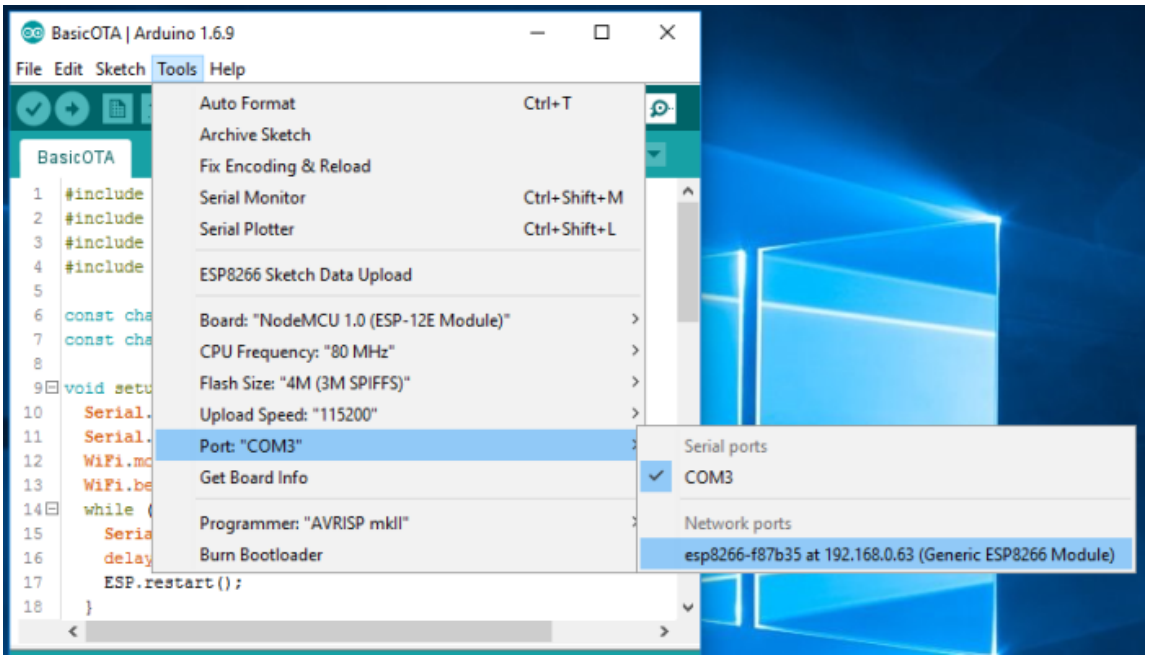

File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
sketch
// check if log.txt exists

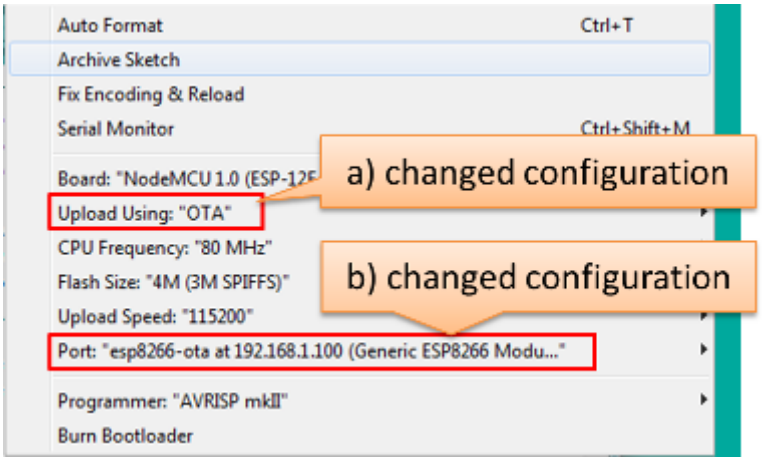

data
File
Home
Share
View
Pin to Quick access
Copy
Paste
Clipboard
Organize
New
Open
Select
« sketch_jan... › data
Search data
Quick access
OneDrive
This PC
Desktop
Documents
Name
log.txt
1 item


CHPD TX RX MI MO SCK NC NC NC NC NC ADC GND 3V NC RST
CHG
RST
huzzah
BAT EN USB
SCL SDA

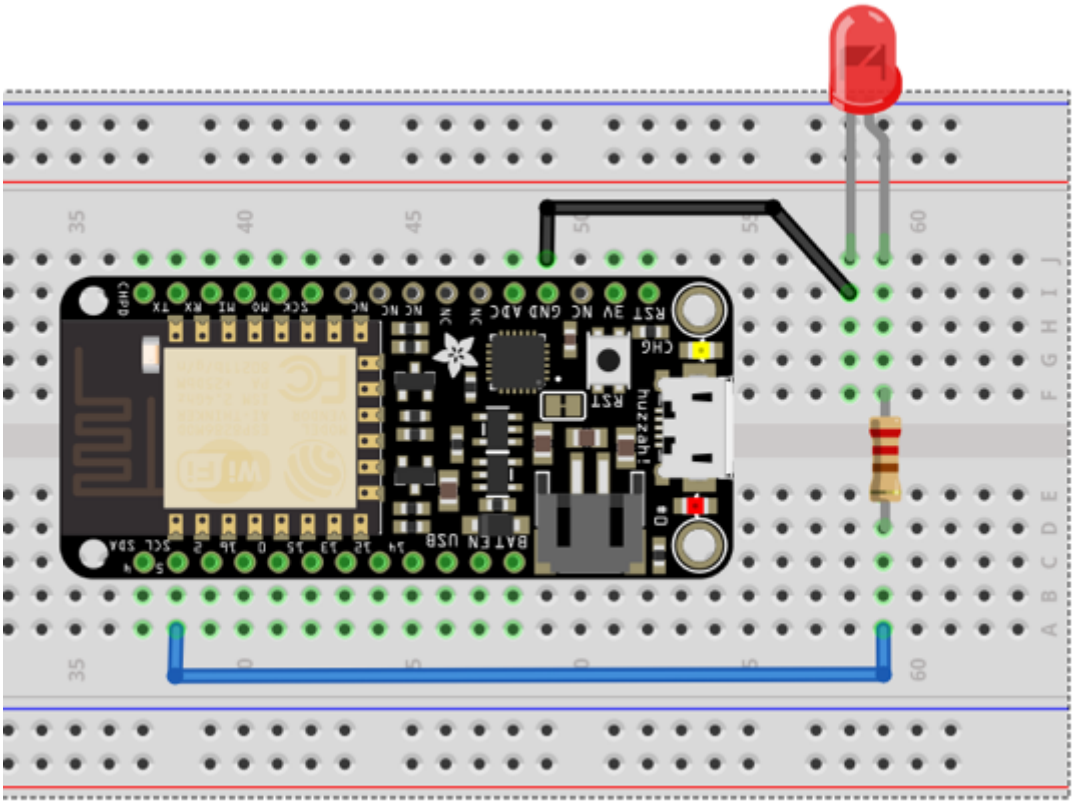

Python 2.7.10 Setup
Customize Python 2.7.10
Select the way you want features to be installed. Click on the icons in the tree below to change the way features will be installed.
Register Extensions
Tcl/Tk
Documentation
Utility Scripts
pip
Test suite
Add python.exe to Path
select this option
Prepend C:\Python27\ to the system Path variable. This allows you to type 'python' into a command prompt without needing the full path.
This feature requires 0KB on your hard drive.
python for windows
Disk Usage
Advanced
< Back
Next >
Cancel

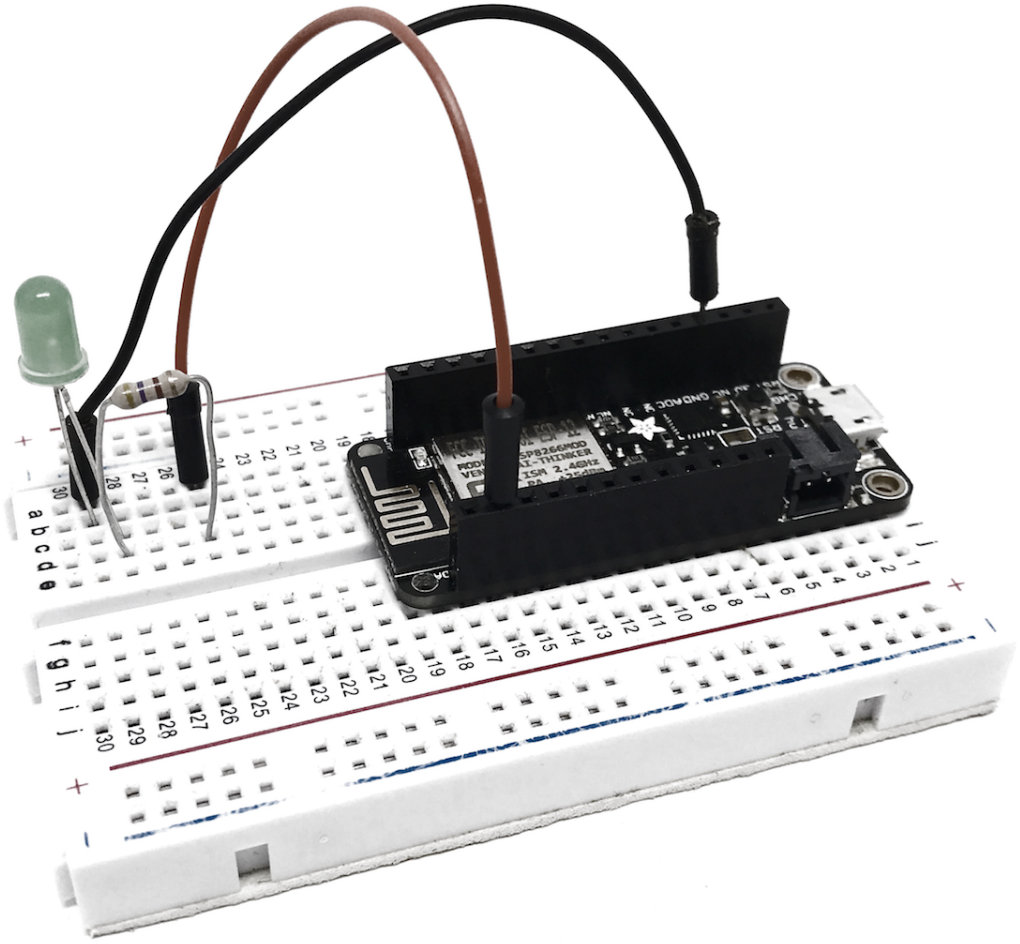

File
Edit
Sketch
Tools
Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q
Examples from Custom Librar
AccelStepper
Adafruit LSM9DS0 Library
Adafruit_NeoPixel
Adafruit_PN532
ADC
Adxl345
Arduino_MPU6050
ArduinoOTA
CuHead
DNSServer
DueFlashStorage
EEPROM
EmonLib
ESP8266
BasicOTA
OTALeds

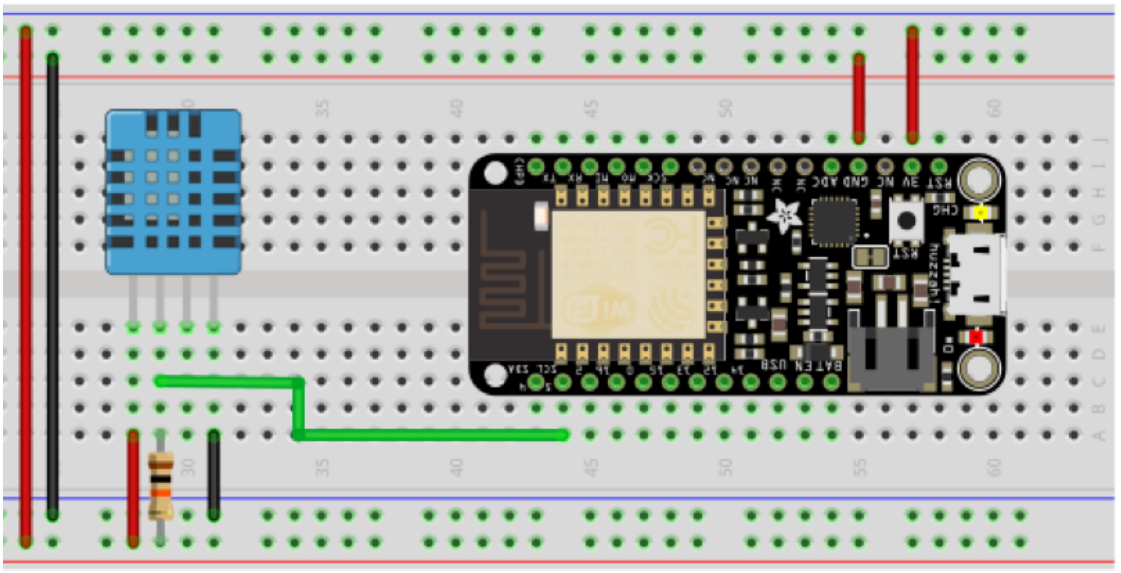

BasicOTA | Arduino 1.6.9
File Edit Sketch Tools Help
BasicOTA
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
ESP8266 Sketch Data Upload
Board: "NodeMCU 1.0 (ESP-12E Module)"
CPU Frequency: "80 MHz"
Flash Size: "4M (3M SPIFFS)"
Upload Speed: "115200"
Port: "COM3"
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Serial ports
COM3
Network ports
esp8266-f87b35 at 192.168.0.63 (Generic ESP8266 Module)


Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Upload Using: "OTA"
CPU Frequency: "80 MHz"
Flash Size: "4M (3M SPIFFS)"
Upload Speed: "115200"
Port: "esp8266-ota at 192.168.1.100 (Generic ESP8266 Modu..."
Programmer: "AVRISP mkII"
Burn Bootloader
a) changed configuration
b) changed configuration

Chapter 4: Using MicroPython on the ESP8266







```
>>> for i in range(3):
...     _
```

```
>>> for i in range(30):
...     if i > 3:
...         _
```

```
>>> for i in range(30):
...     if i > 3:
...         break
...     _
```

```
>>> for i in range(30):
...     if i > 3:
...         break
...     print(i)
...
0
1
2
3
>>>
```

```
>>> machine.
__name__        info            unique_id       reset
bootloader      freq            rng             idle
sleep           deepsleep       disable_irq     enable_irq
Pin
```

```
>>> machine.Pin.AF3_TIM
AF3_TIM10       AF3_TIM11       AF3_TIM8        AF3_TIM9
>>> machine.Pin.AF3_TIM
```

```
>>> for i in range(1000000):
...     print(i)
...
0
1
2
3
...
6466
6467
6468
Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
KeyboardInterrupt:
>>>
```

```
    def foo():
        print('This is a test to show paste mode')
        print('Here is a second line')
    foo()
```

```
>>> def foo():
...         print('This is a test to show paste mode')
...             print('Here is a second line')
...             foo()
...
  File "<stdin>", line 3
IndentationError: unexpected indent
```

```
>>>
paste mode; Ctrl-C to cancel, Ctrl-D to finish
=== def foo():
===     print('This is a test to show paste mode')
===     print('Here is a second line')
=== foo()
===
This is a test to show paste mode
Here is a second line
>>>
```

```
>>> dir()
['__name__', 'pyb']
```

```
>>> i = 1
>>> j = 23
>>> x = 'abc'
>>> dir()
['j', 'x', '__name__', 'pyb', 'i']
>>>
```

```
PYB: sync filesystems
PYB: soft reboot
MicroPython v1.5-51-g6f70283-dirty on 2015-10-30; PYBv1.0 with STM32F405RG
Type "help()" for more information.
>>> dir()
['__name__', 'pyb']
>>>
```

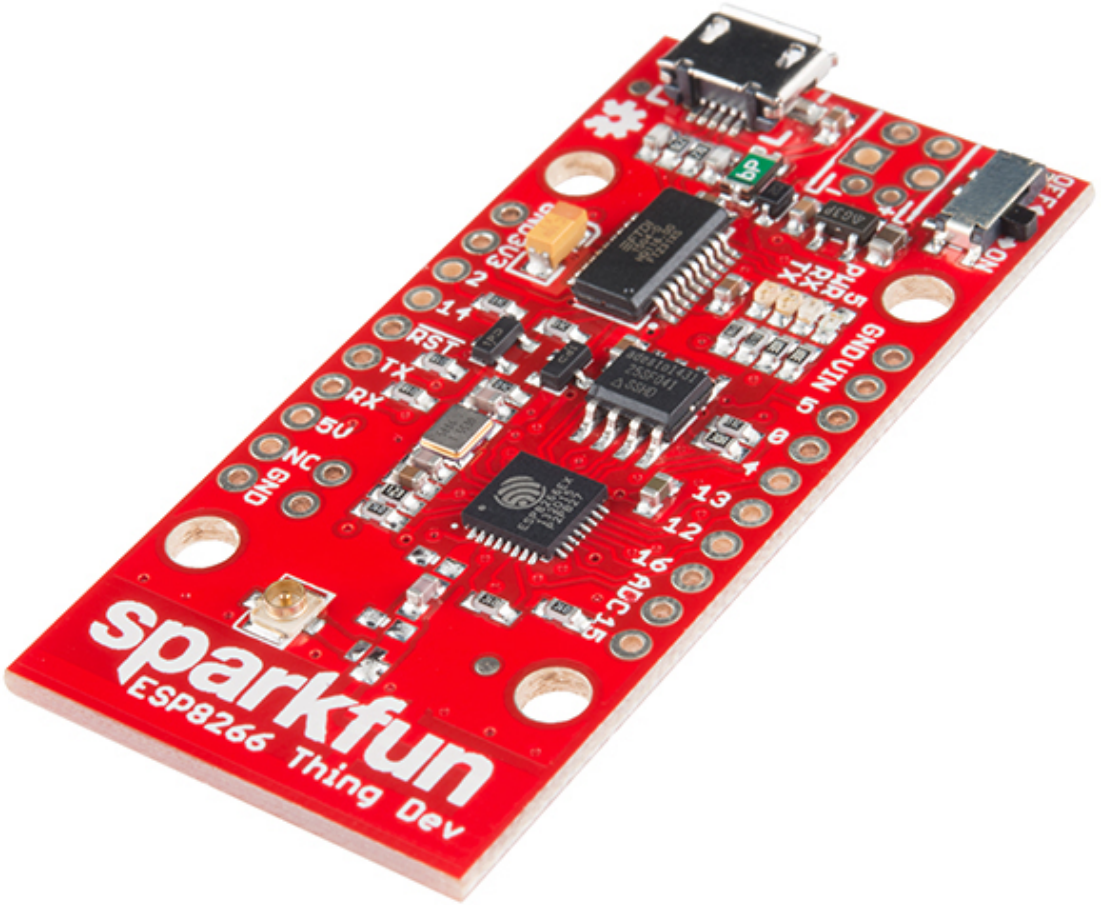
```
>>> 1 + 2 + 3 + 4 + 5
15
>>> x = _
>>> x
15
>>>
```

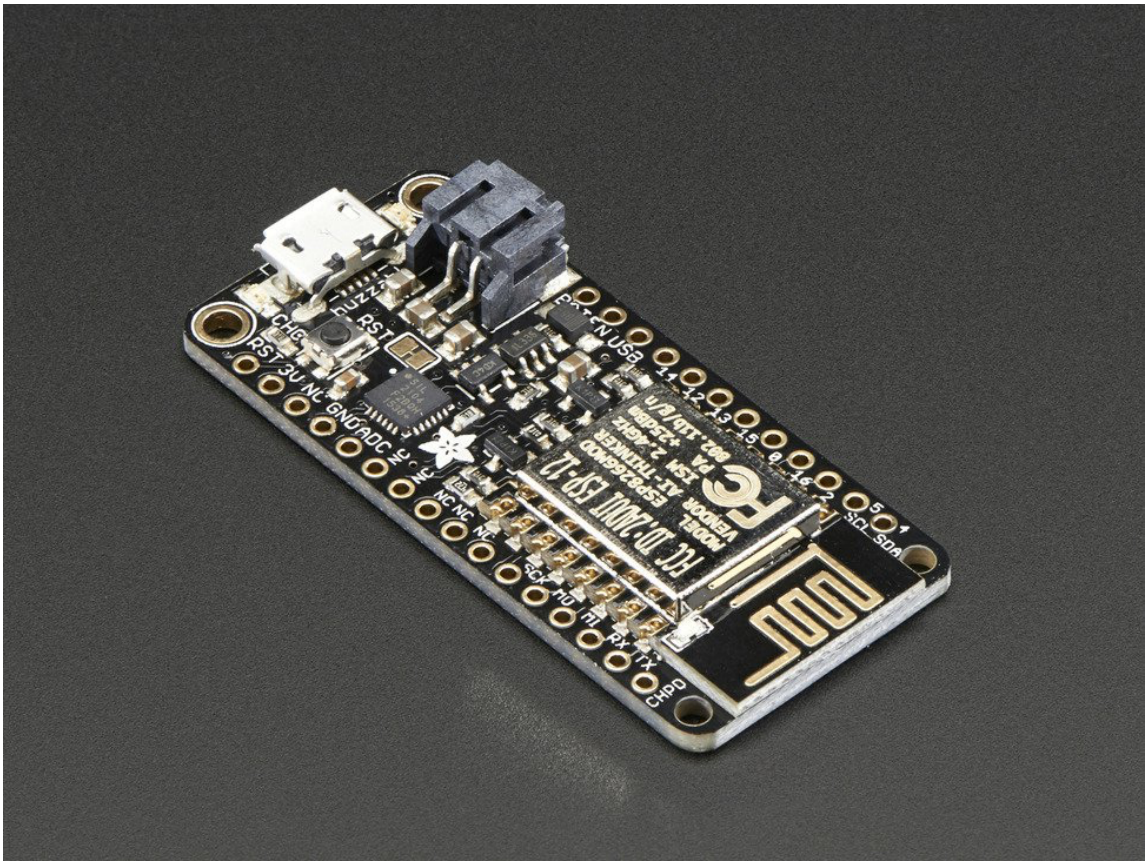
```
picocom /dev/ttyUSB0
```

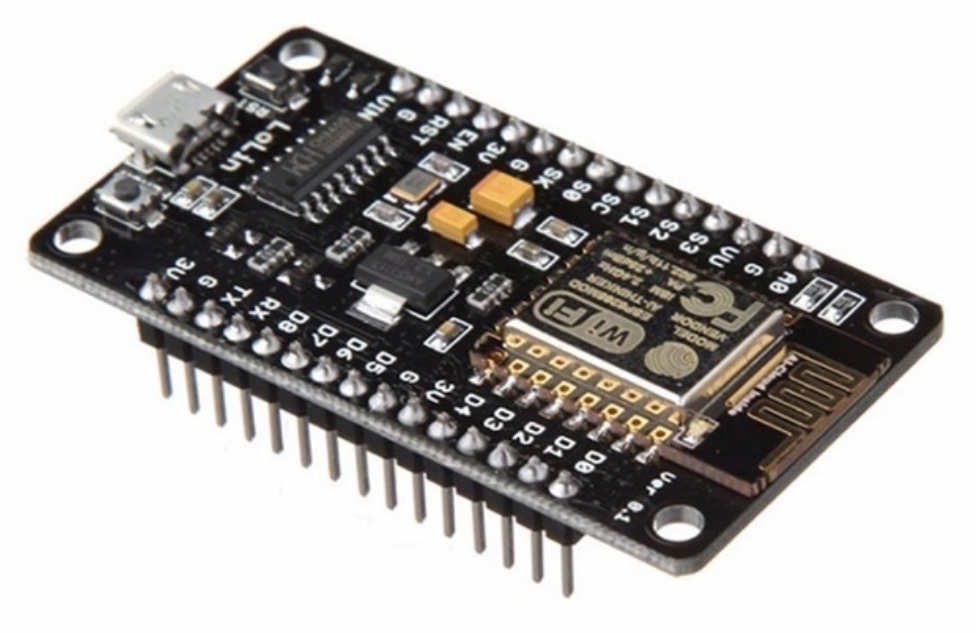
```
>>> pin = machine.Pin(0, machine.Pin.IN, machine.Pin.PULL_UP)
```

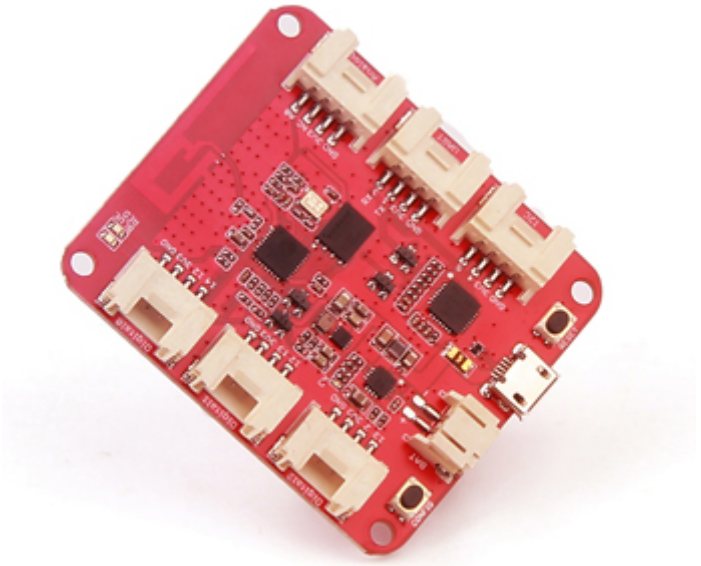
```
raise SystemExit
```

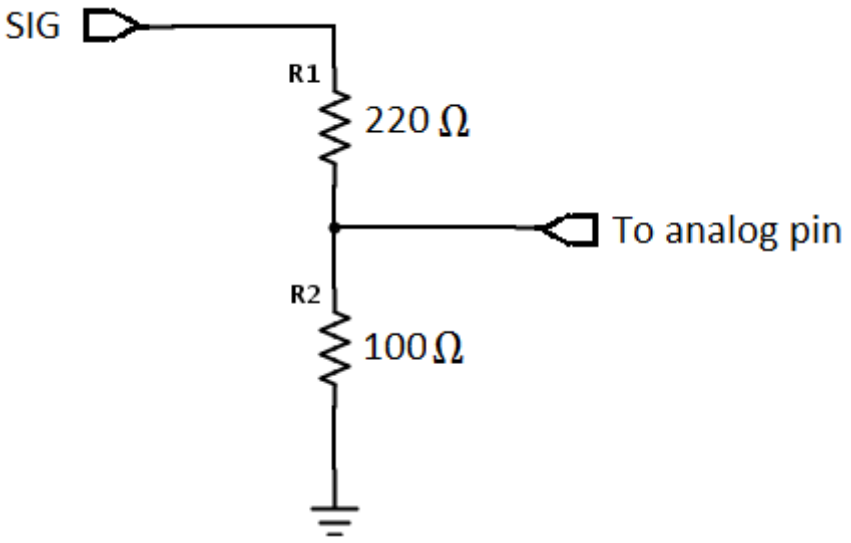
Chapter 5: Cloud Data Monitoring

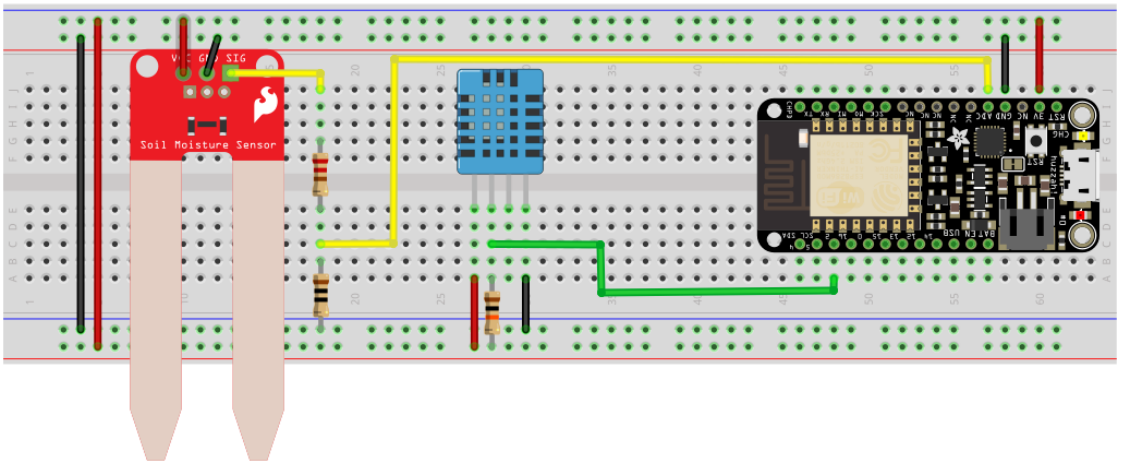



BUZZER
RST
CHG
RST
3V
NC
GND
ADC
NC
NC
NC NC
NC
SCK
MO
MI
RX
TX
CHPD
USB
5
4
SCL
SDA


RST
EN
3V
SK
SO
SC
S1
S2
S3
VV
5
A0
RX
TX
D8
D7
D6
D5
D4
D3
D2
D1
D0
GND
3V
5
Wi Fi




SIG
R1
220 Ω
To analog pin
R2
100 Ω



Use Your Locks

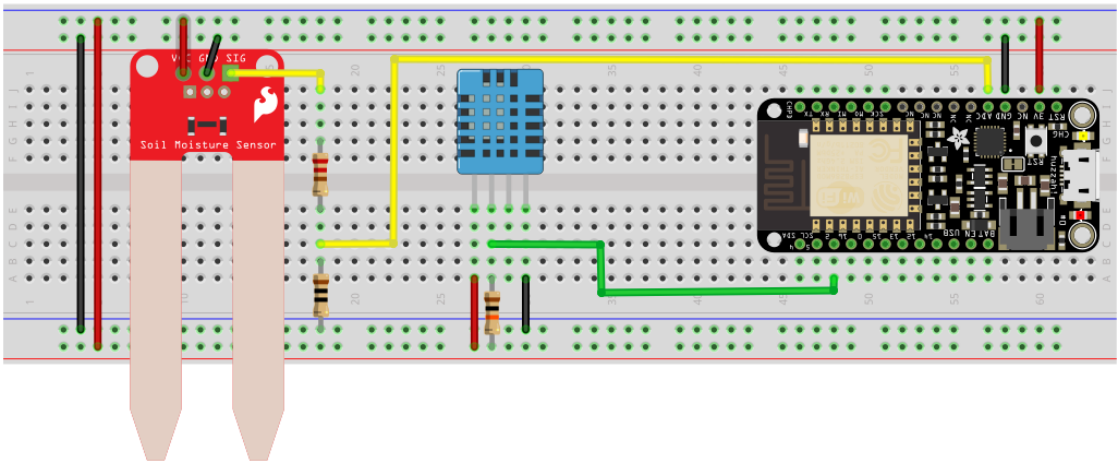
A thing

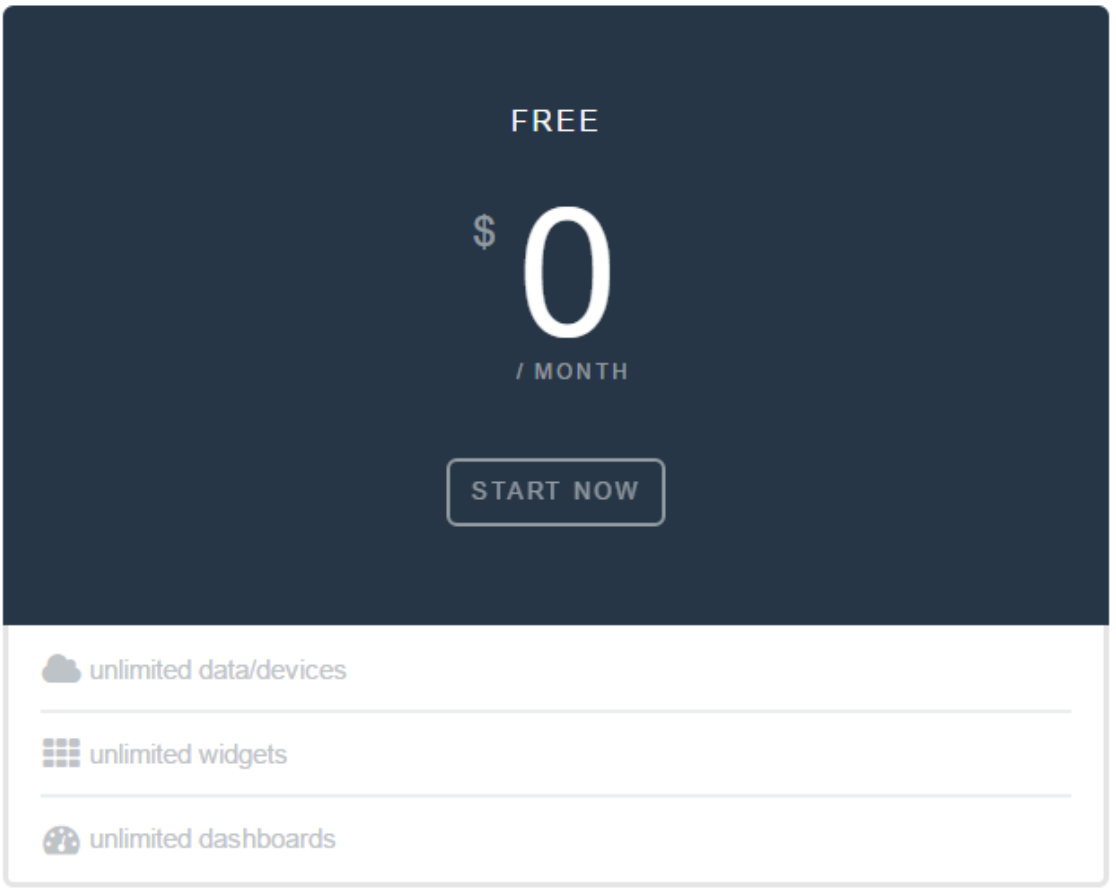
A lock

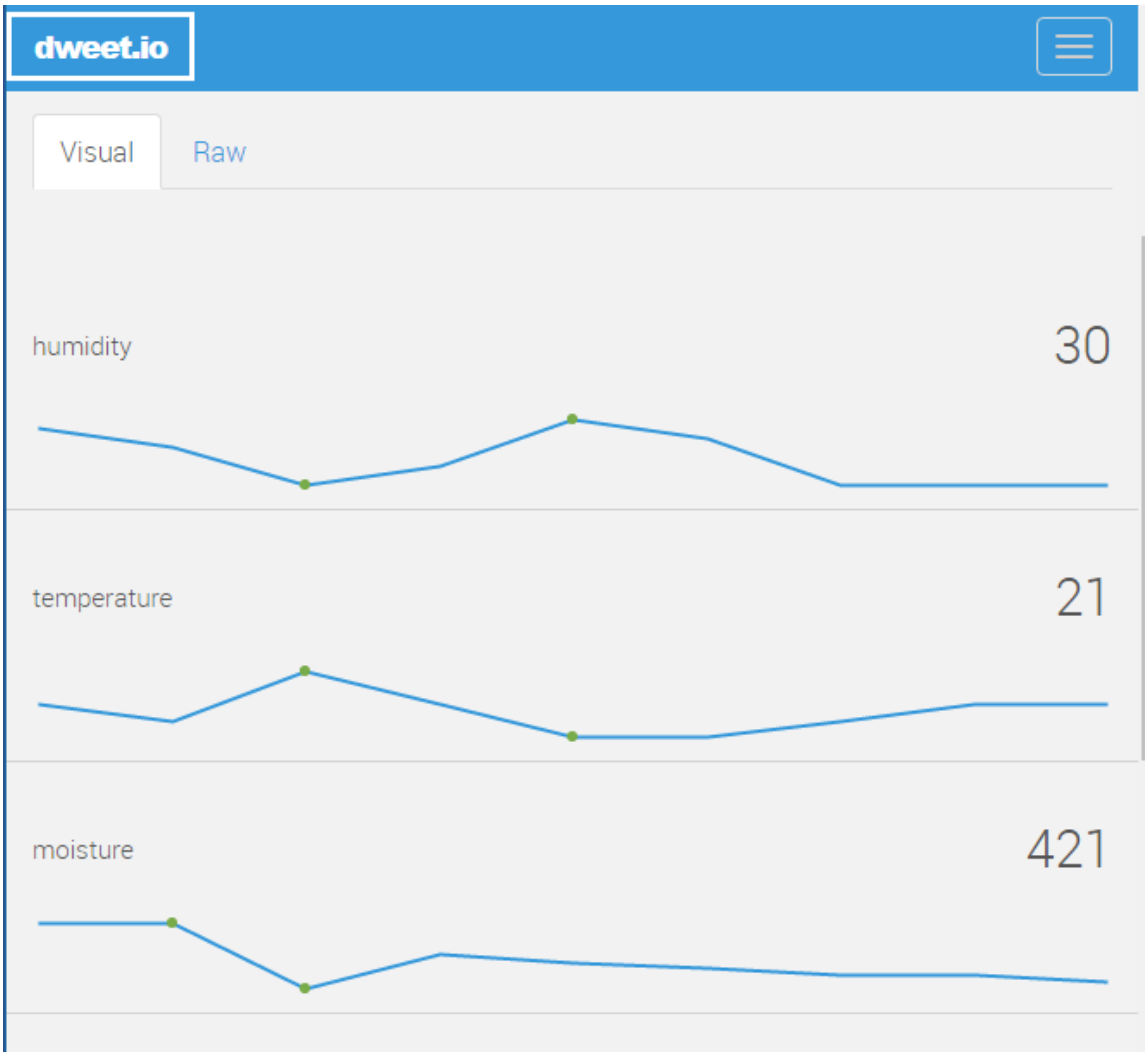
A key

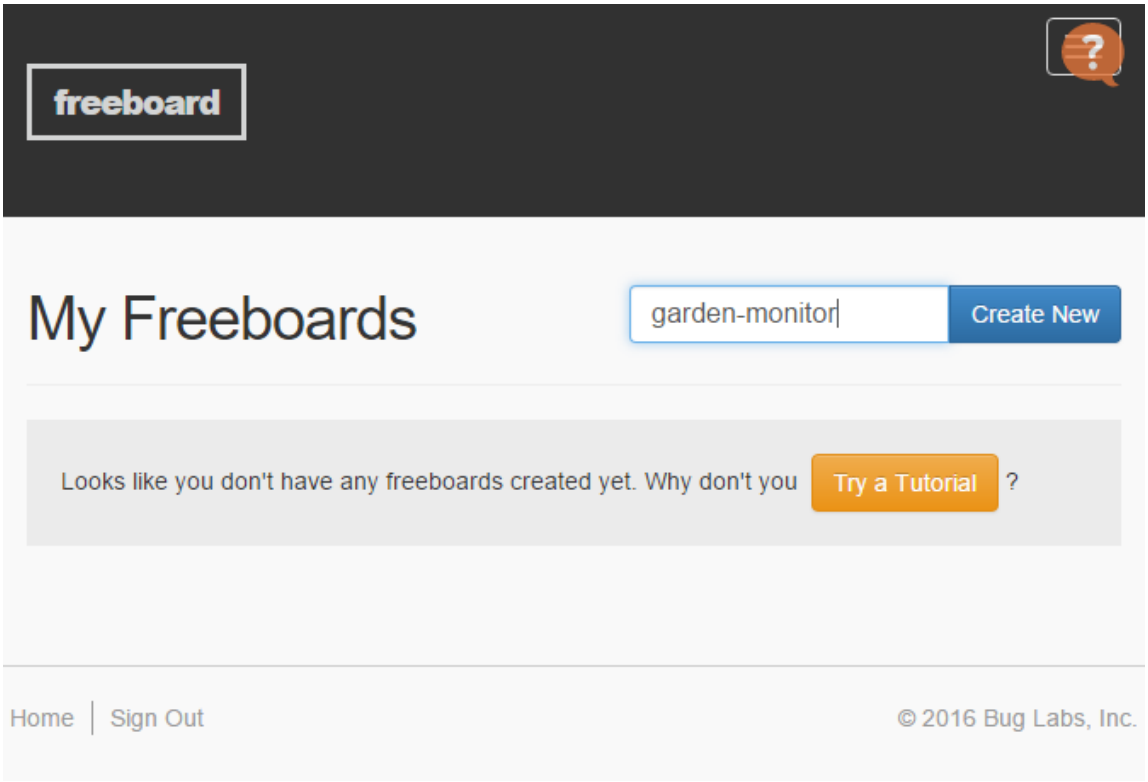
Lock Thing

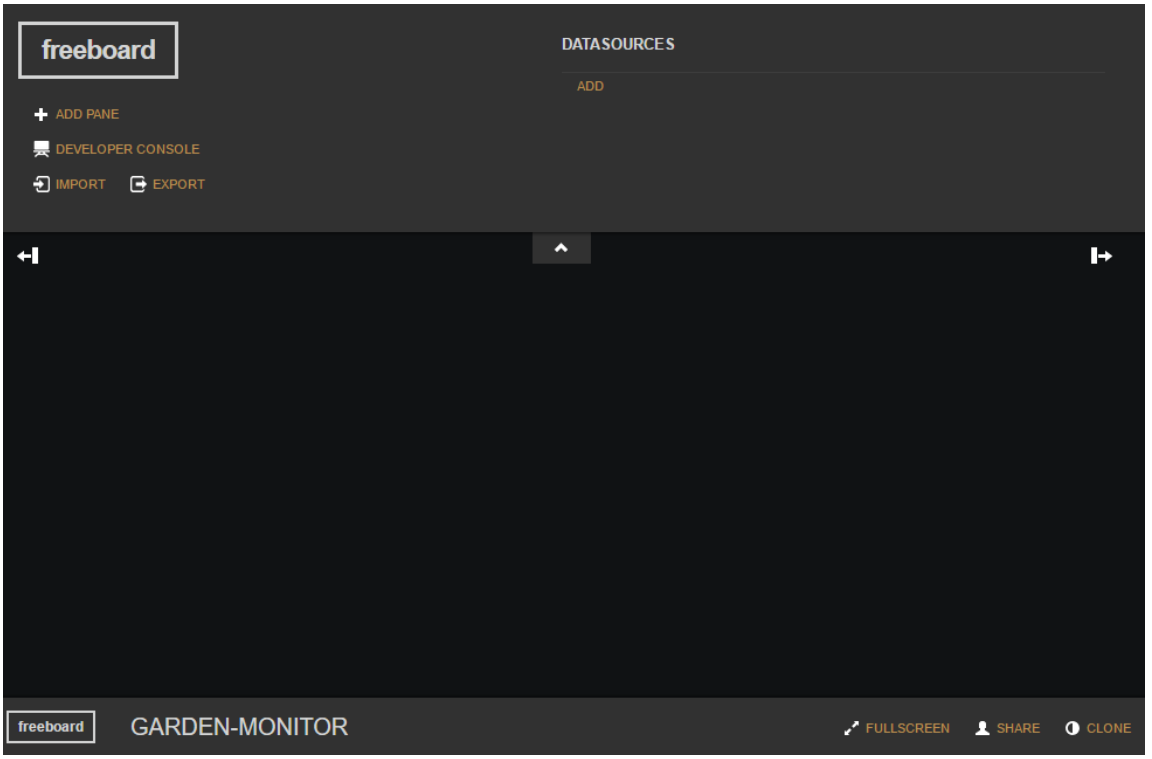
Unlock Thing

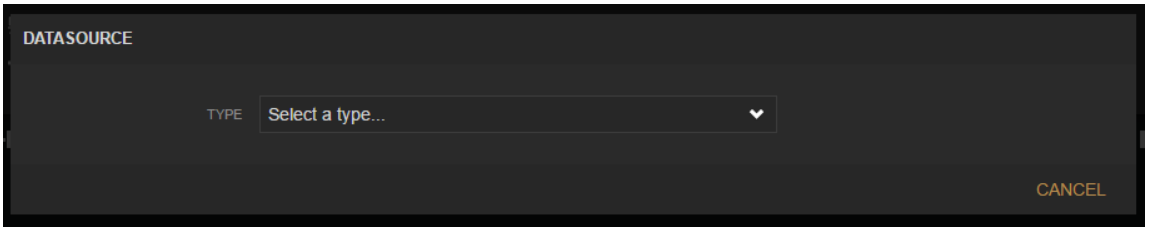

VCC GND SIG
Soil Moisture Sensor

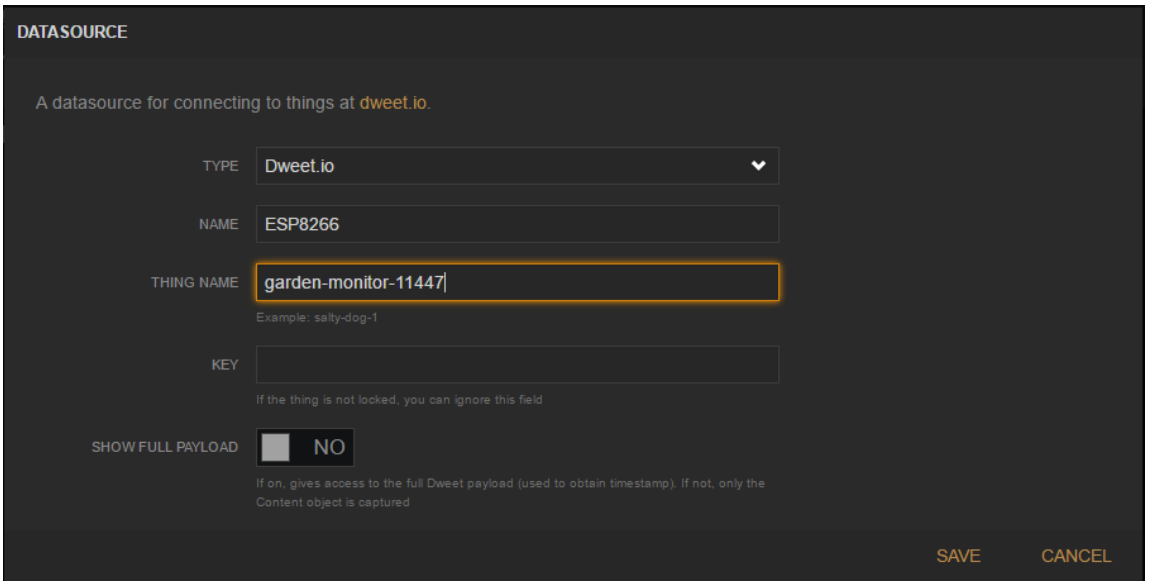

FREE
$ 0
/ MONTH
START NOW
unlimited data/devices
unlimited widgets
unlimited dashboards

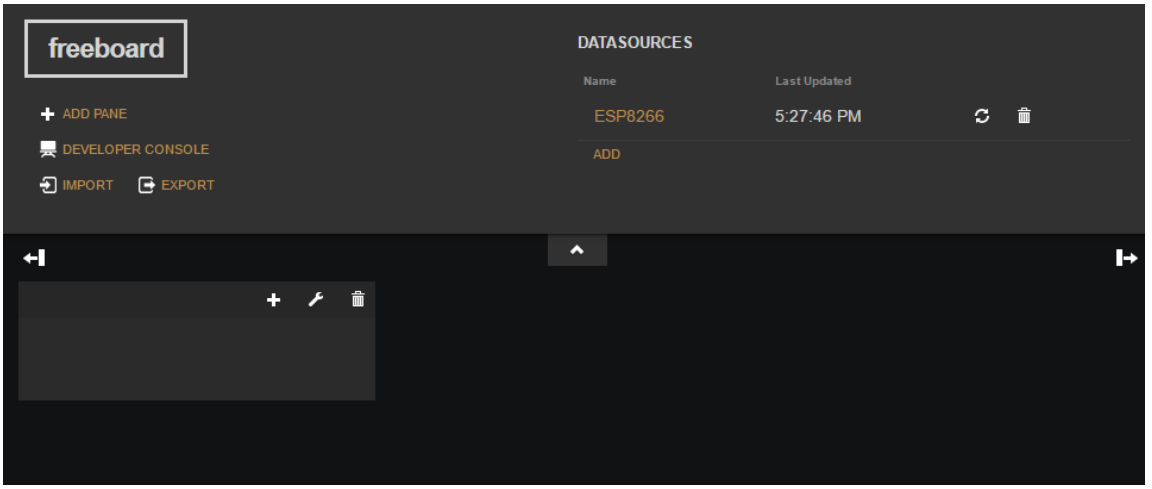

dweet.io
Visual
Raw
humidity
30
temperature
21
moisture
421

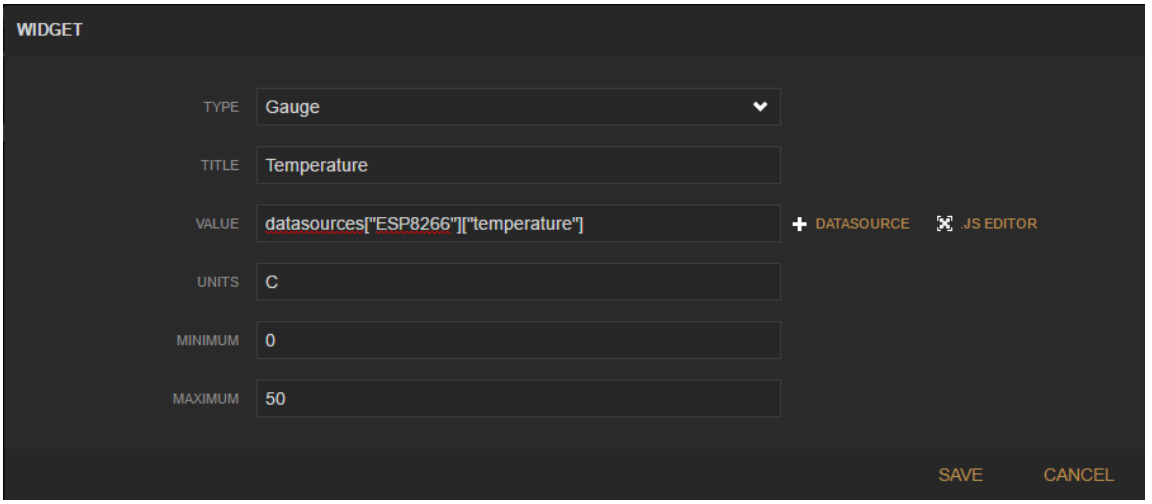

freeboard
My Freeboards
garden-monitor
Create New
Looks like you don't have any freeboards created yet. Why don't you
Try a Tutorial
?
Home
Sign Out
© 2016 Bug Labs, Inc.

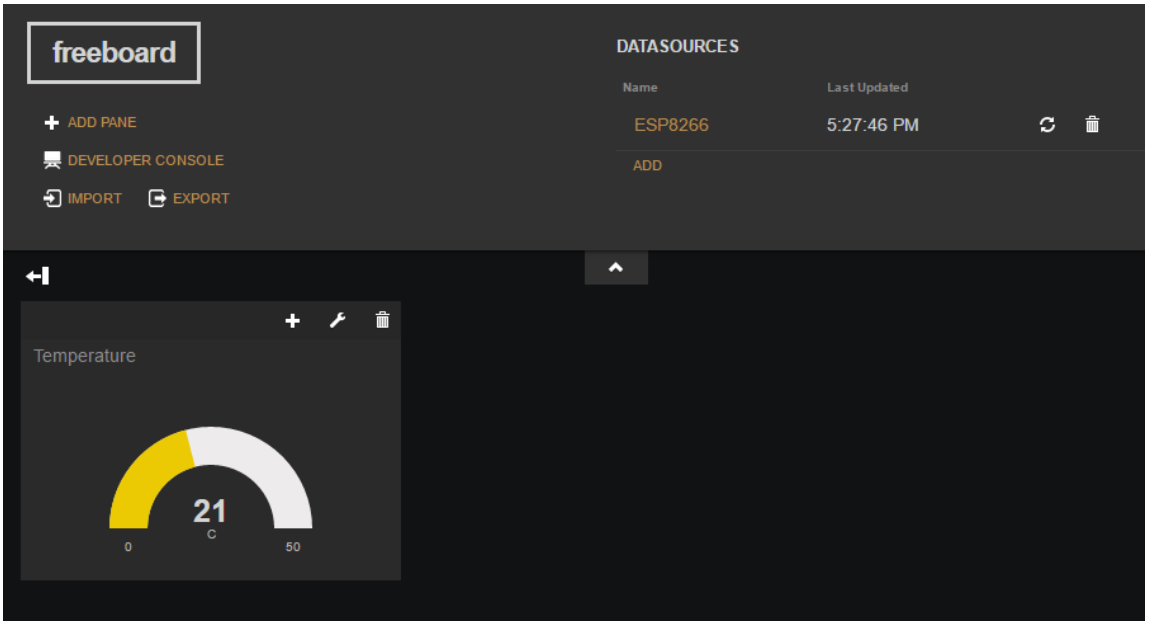

freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
ADD
freeboard
GARDEN-MONITOR
FULLSCREEN
SHARE
CLONE

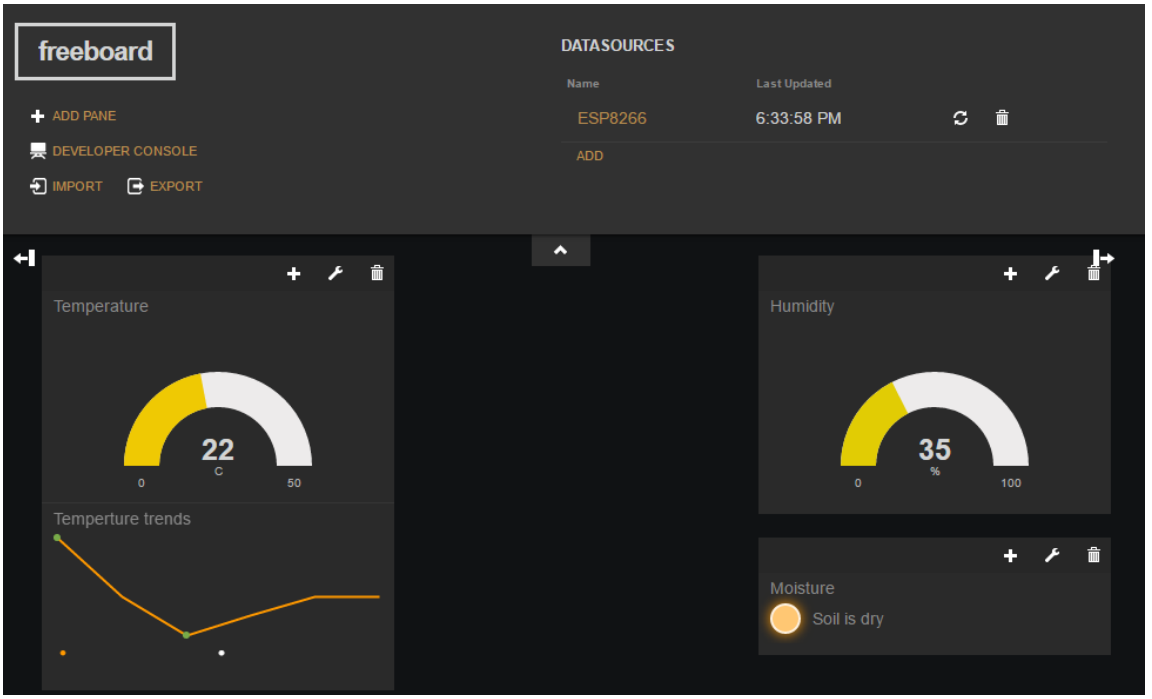

DATASOURCE
TYPE
Select a type...
CANCEL

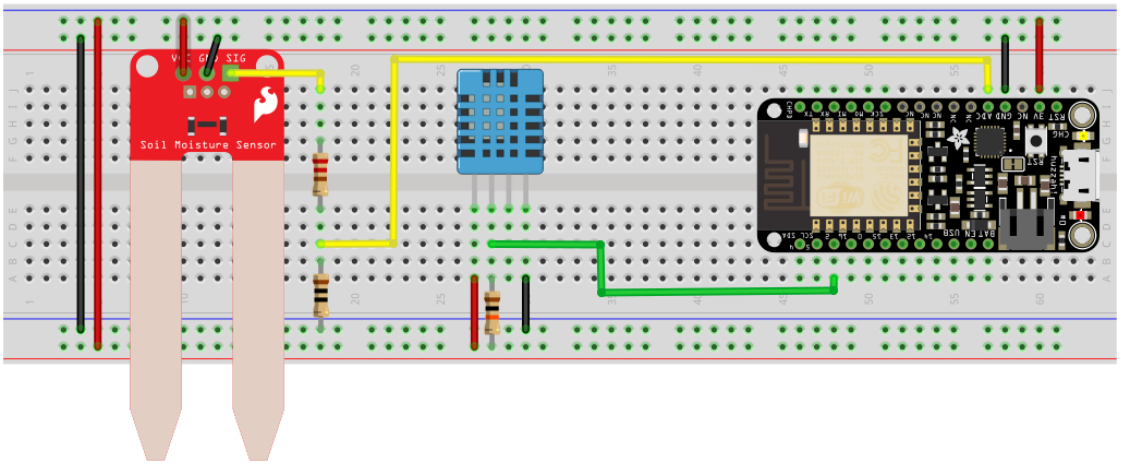

DATASOURCE
A datasource for connecting to things at dweet.io.
TYPE
Dweet.io
NAME
ESP8266
THING NAME
garden-monitor-11447
Example: salty-dog-1
KEY
If the thing is not locked, you can ignore this field
SHOW FULL PAYLOAD
NO
If on, gives access to the full Dweet payload (used to obtain timestamp). If not, only the Content object is captured
SAVE
CANCEL


freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
Name
Last Updated
ESP8266
5:27:46 PM
ADD


WIDGET
TYPE
Gauge
TITLE
Temperature
VALUE
datasources["ESP8266"]["temperature"]
DATASOURCE
JS EDITOR
UNITS
C
MINIMUM
0
MAXIMUM
50
SAVE
CANCEL


freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
Name
Last Updated
ESP8266
5:27:46 PM
ADD
Temperature
21
C
0
50

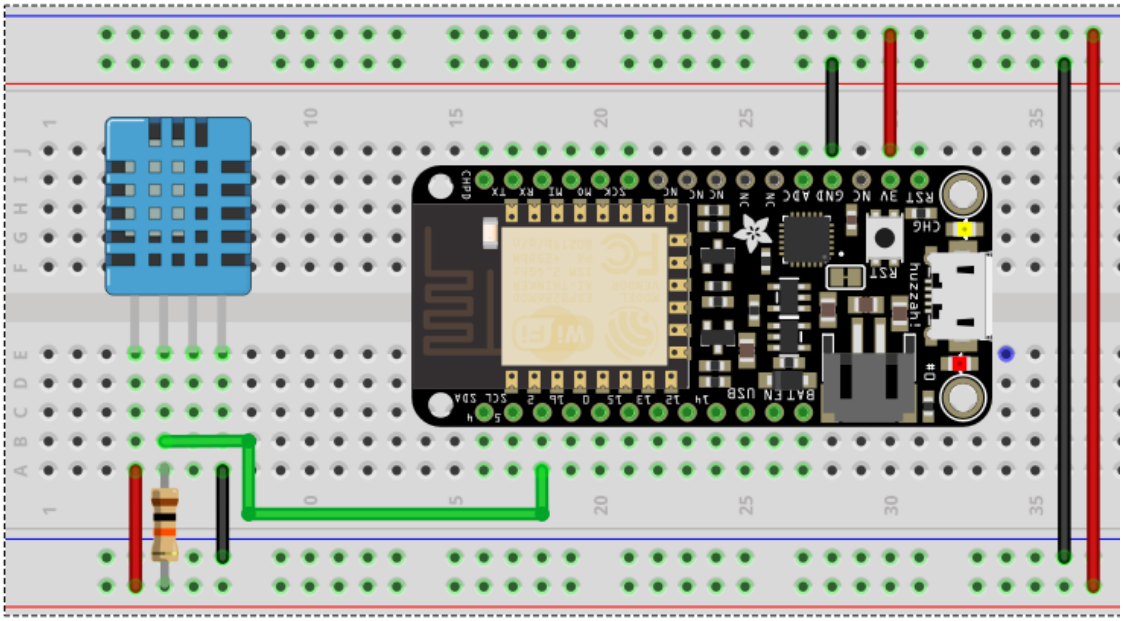

freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
Name
Last Updated
ESP8266
6:33:58 PM
ADD
Temperature
22
C
0
50
Temperture trends
Humidity
35
%
0
100
Moisture
Soil is dry

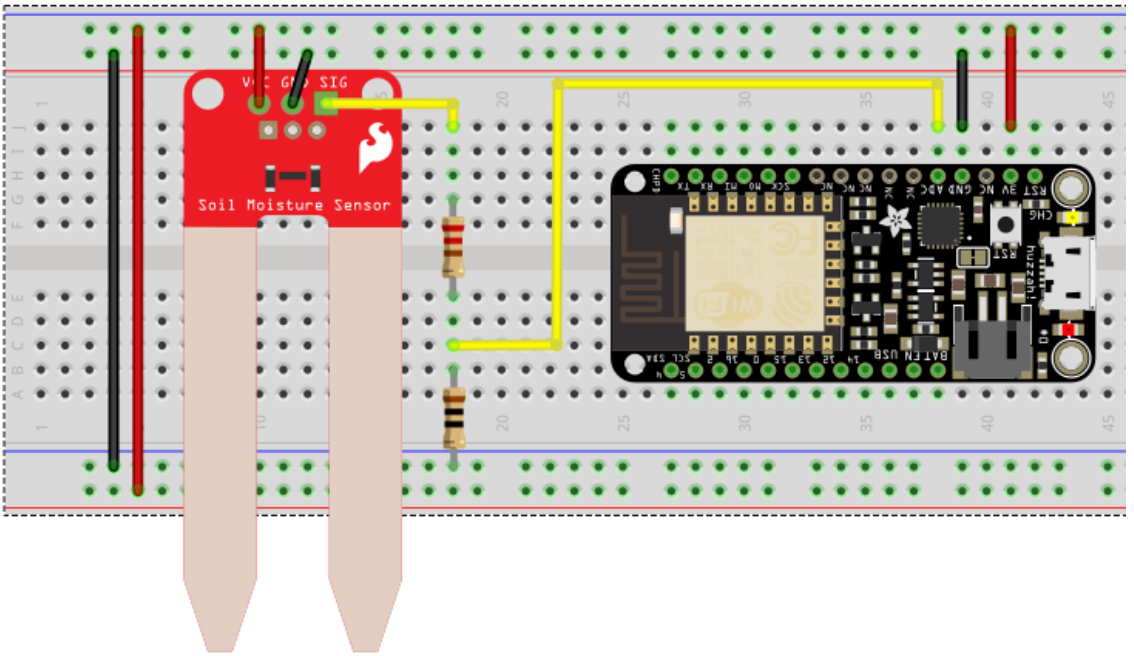

Soil Moisture Sensor

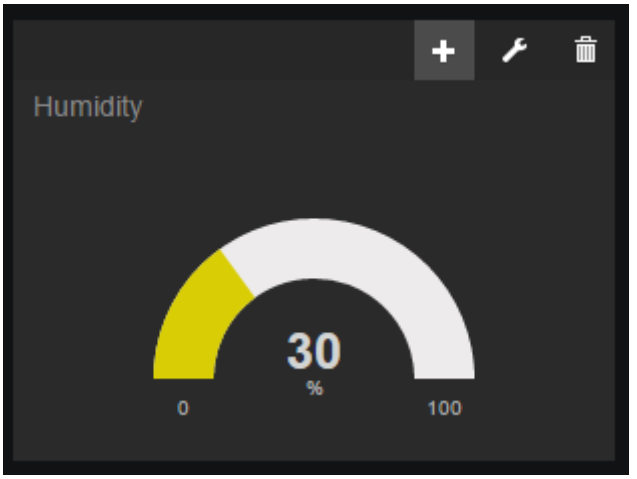
dweet.io

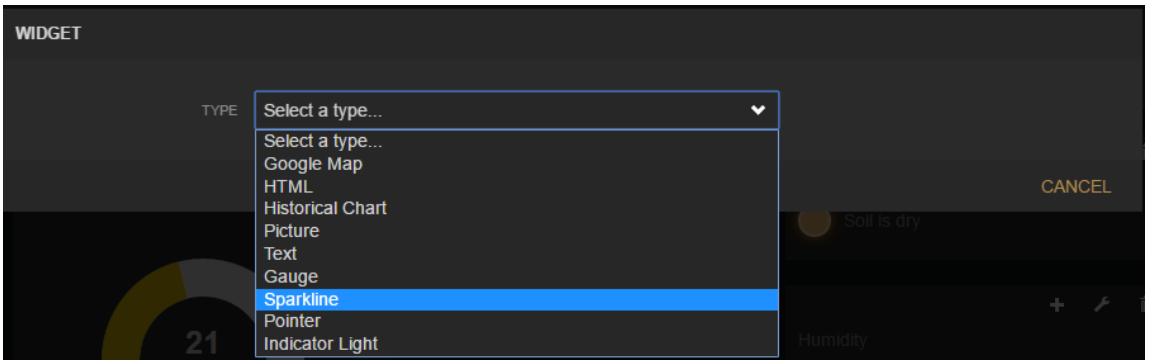
An alert has been OPENED for the thing 'garden-monitor-11447'.

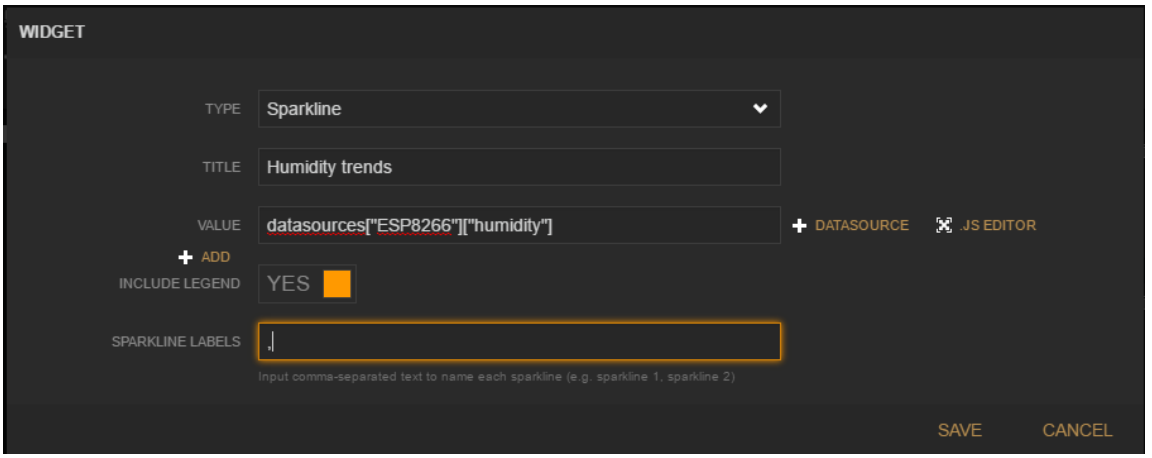
It said: Too hot

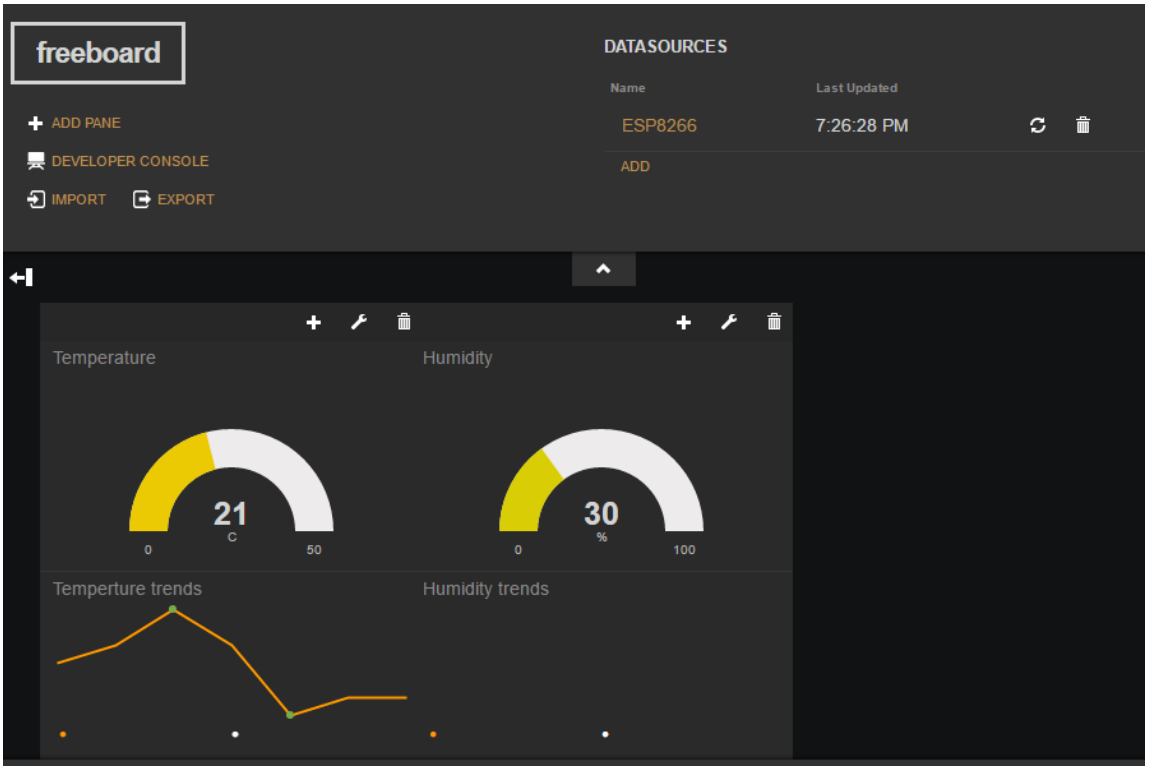


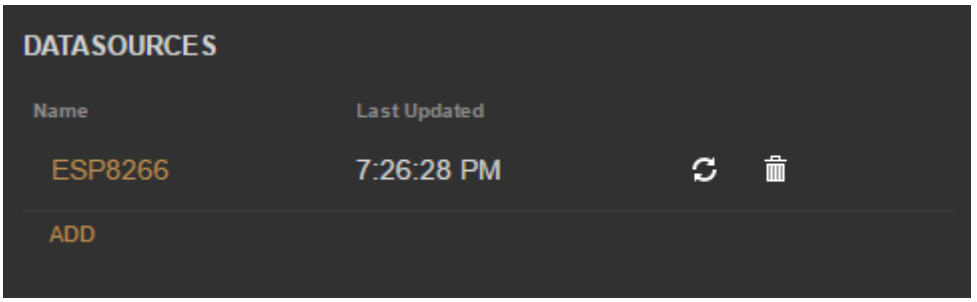

SIG
Soil Moisture Sensor

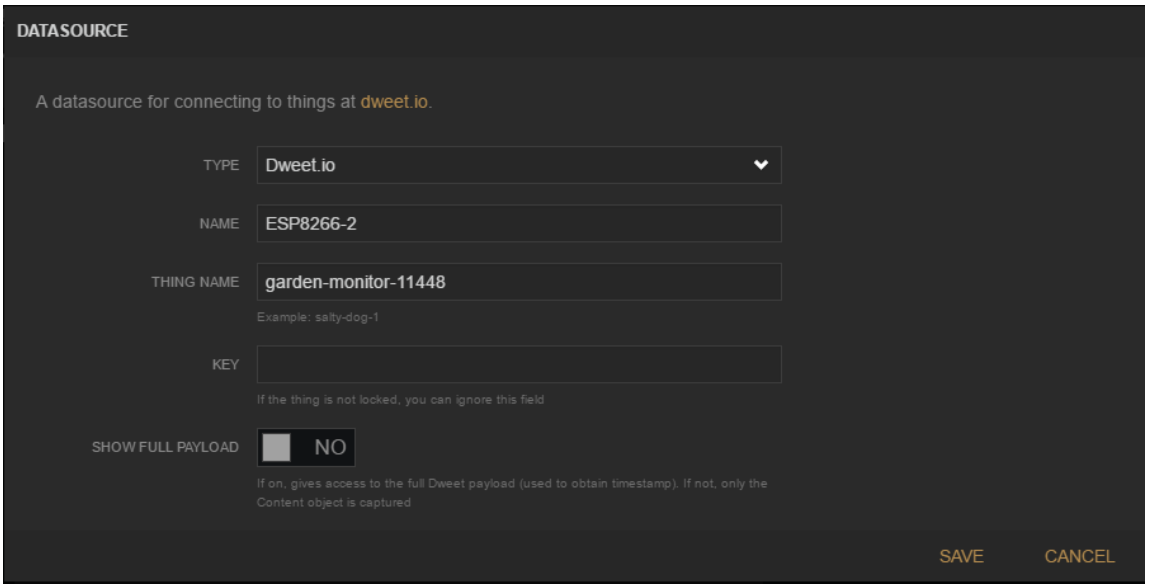

Humidity
30
%
0
100

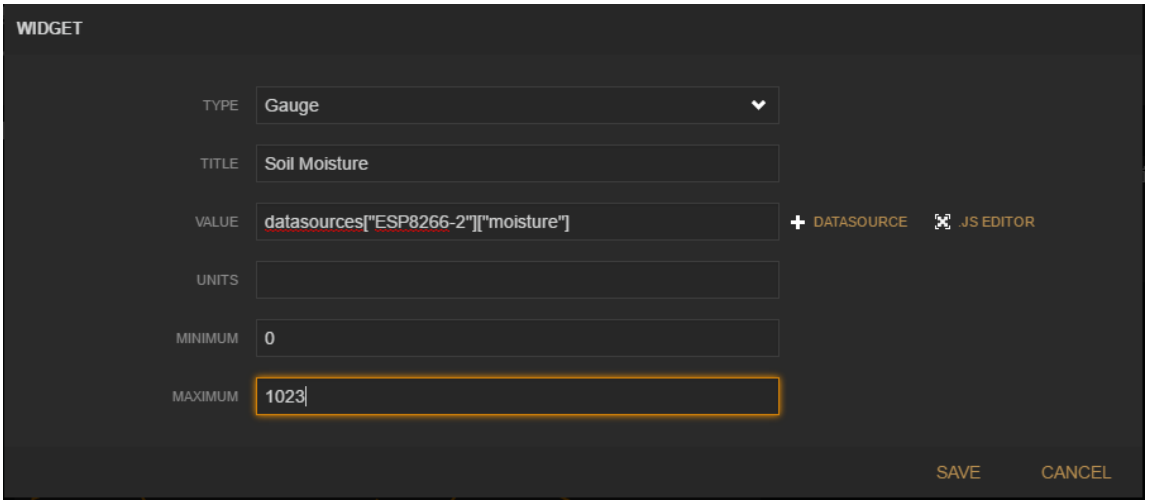

WIDGET
TYPE
Select a type...
Select a type...
Google Map
HTML
Historical Chart
Picture
Text
Gauge
Sparkline
Pointer
Indicator Light
CANCEL
Soil is dry
21
Humidity

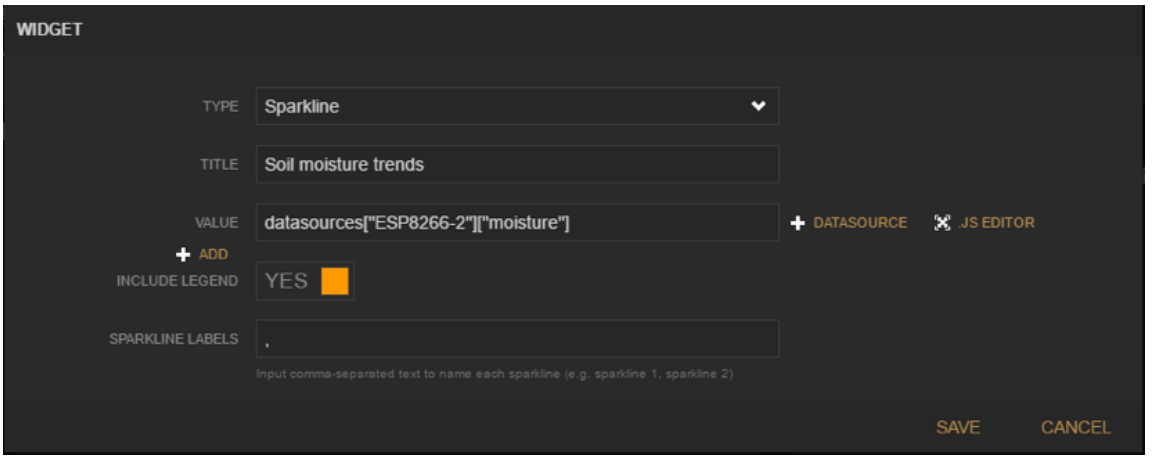

WIDGET
TYPE
Sparkline
TITLE
Humidity trends
VALUE
datasources["ESP8266"]["humidity"]
DATASOURCE
JS EDITOR
ADD
INCLUDE LEGEND
YES
SPARKLINE LABELS
,
Input comma-separated text to name each sparkline (e.g. sparkline 1, sparkline 2)
SAVE
CANCEL

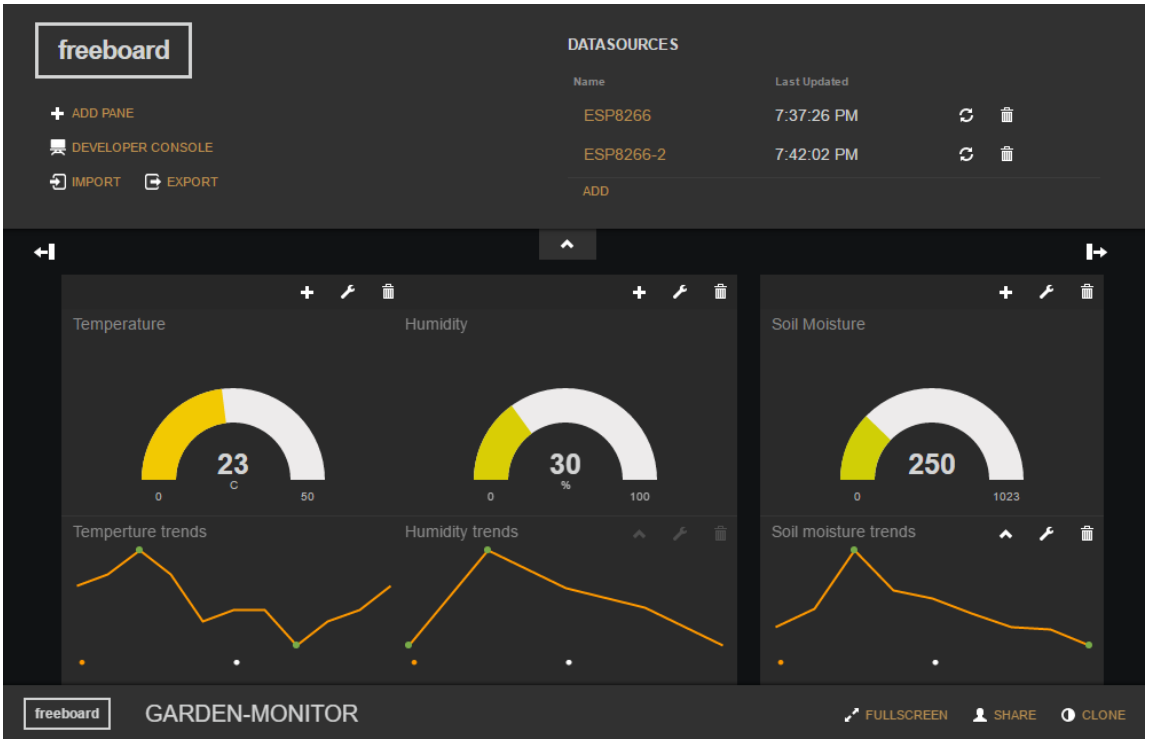

freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
Name
Last Updated
ESP8266
7:26:28 PM
ADD
Temperature
21
C
0
50
Humidity
30
%
0
100
Temperture trends
Humidity trends


DATASOURCES
Name
Last Updated
ESP8266
7:26:28 PM
ADD

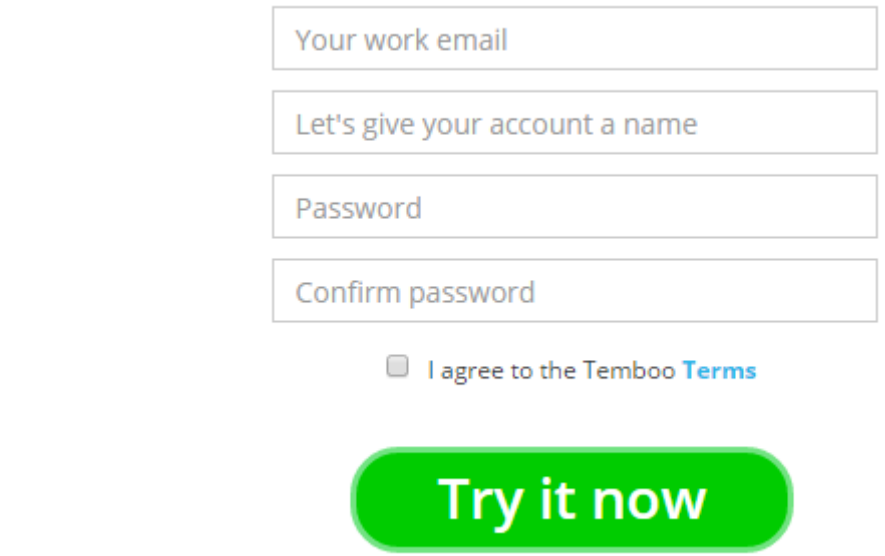

DATASOURCE
A datasource for connecting to things at dweet.io.
TYPE
Dweet.io
NAME
ESP8266-2
THING NAME
garden-monitor-11448
Example: salty-dog-1
KEY
If the thing is not locked, you can ignore this field
SHOW FULL PAYLOAD
NO
If on, gives access to the full Dweet payload (used to obtain timestamp). If not, only the Content object is captured
SAVE
CANCEL

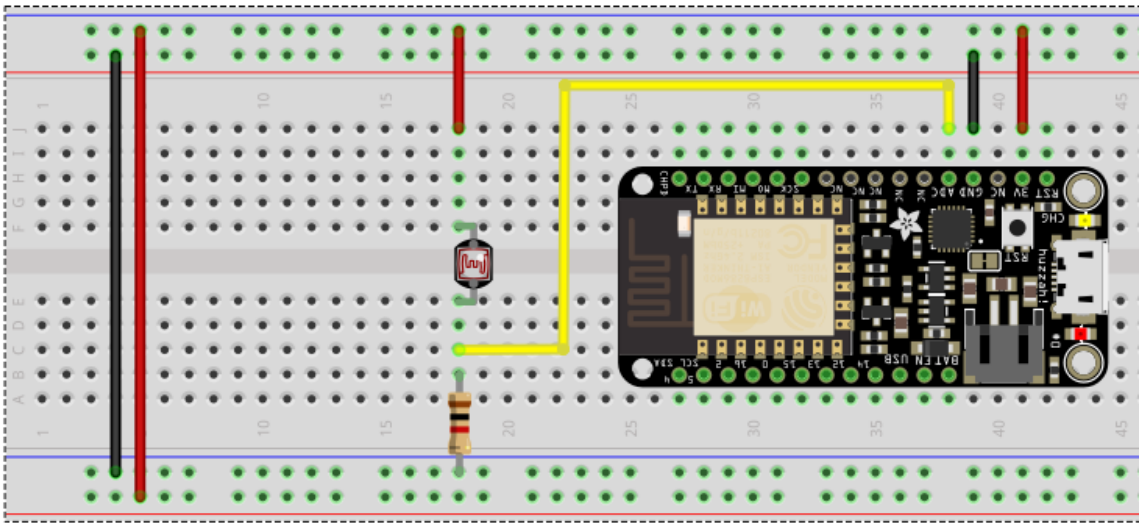

WIDGET
TYPE
Gauge
TITLE
Soil Moisture
VALUE
datasources["ESP8266-2"]["moisture"]
DATASOURCE
JS EDITOR
UNITS
MINIMUM
0
MAXIMUM
1023
SAVE
CANCEL


WIDGET
TYPE
Sparkline
TITLE
Soil moisture trends
VALUE
datasources["ESP8266-2"]["moisture"]
DATASOURCE
JS EDITOR
ADD
INCLUDE LEGEND
YES
SPARKLINE LABELS
,
Input comma-separated text to name each sparkline (e.g. sparkline 1, sparkline 2)
SAVE
CANCEL


freeboard
ADD PANE
DEVELOPER CONSOLE
IMPORT
EXPORT
DATASOURCES
Name
Last Updated
ESP8266
7:37:26 PM
ESP8266-2
7:42:02 PM
ADD
Temperature
23
C
0
50
Humidity
30
%
0
100
Soil Moisture
250
0
1023
Temperture trends
Humidity trends
Soil moisture trends
freeboard
GARDEN-MONITOR
FULLSCREEN
SHARE
CLONE

Chapter 6: Interacting with Web Services





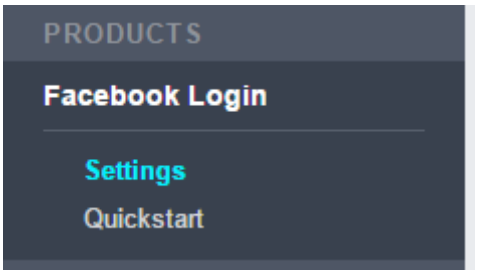
Your Access Token

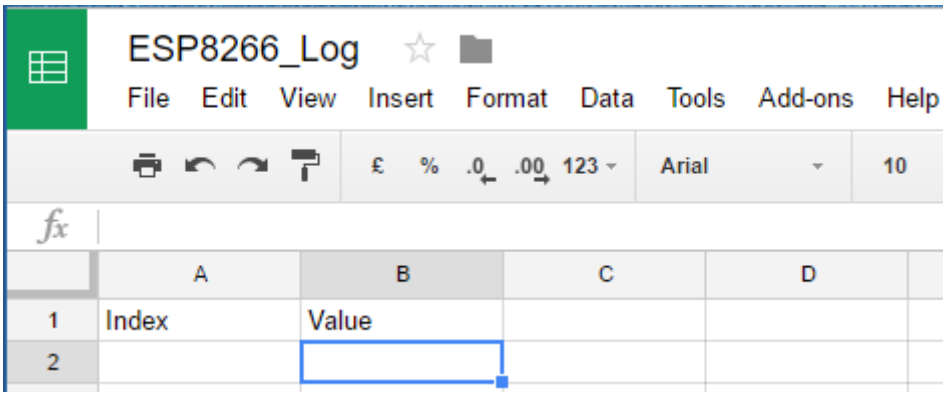
You haven't authorized this application for your own account yet.

By creating your access token here, you will have everything you need to make API calls right away. The access token generated will be assigned your application's current permission level.

Token Actions

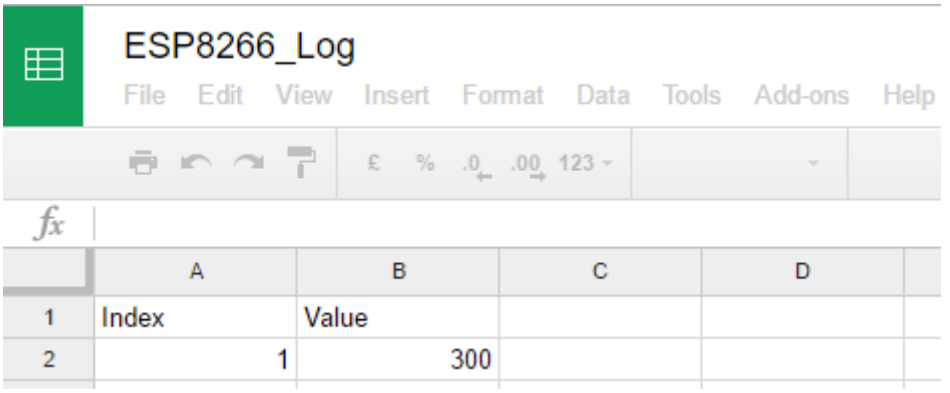
Create my access token

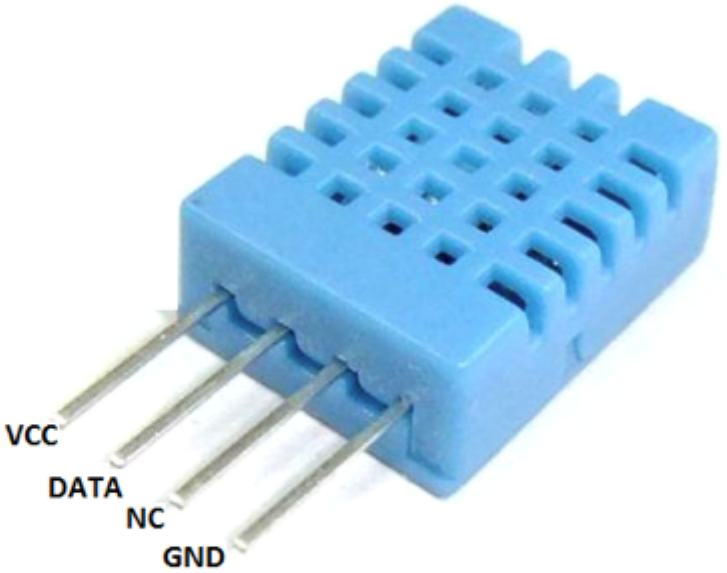


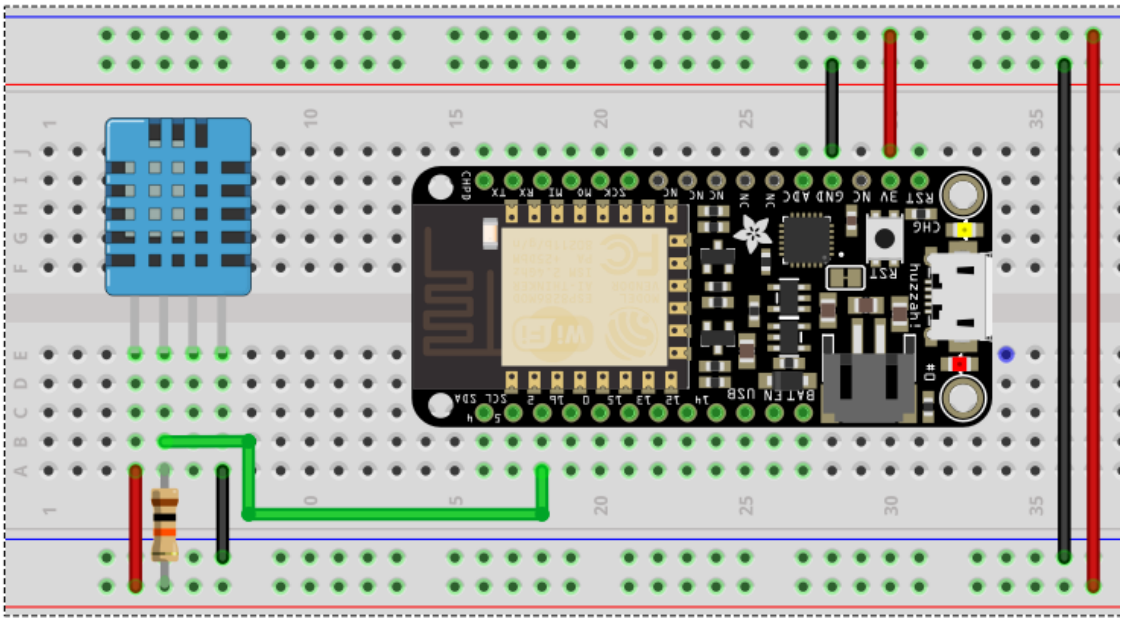


OAuth 2.0 client IDs

Name	Creation date	Type	Client ID
ESP8266_Log	9 Feb 2017	Web application	974405649




VCC
DATA
NC
GND



if this then that

if then that

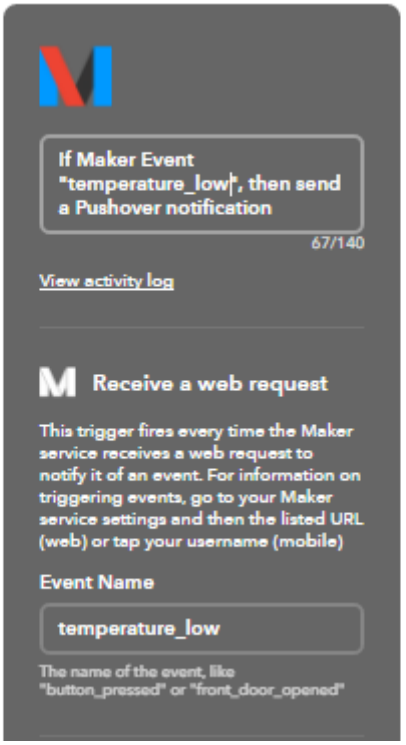
Account Info

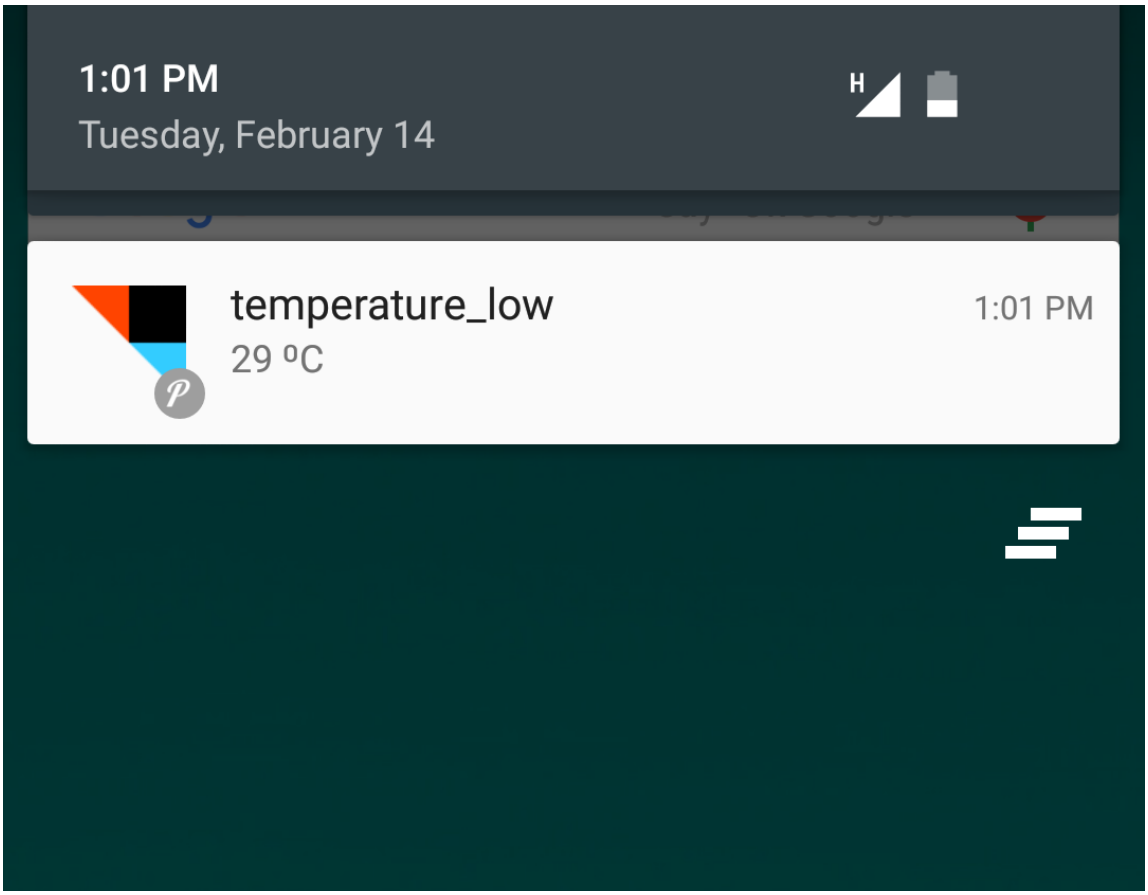
Connected as: yhtomit

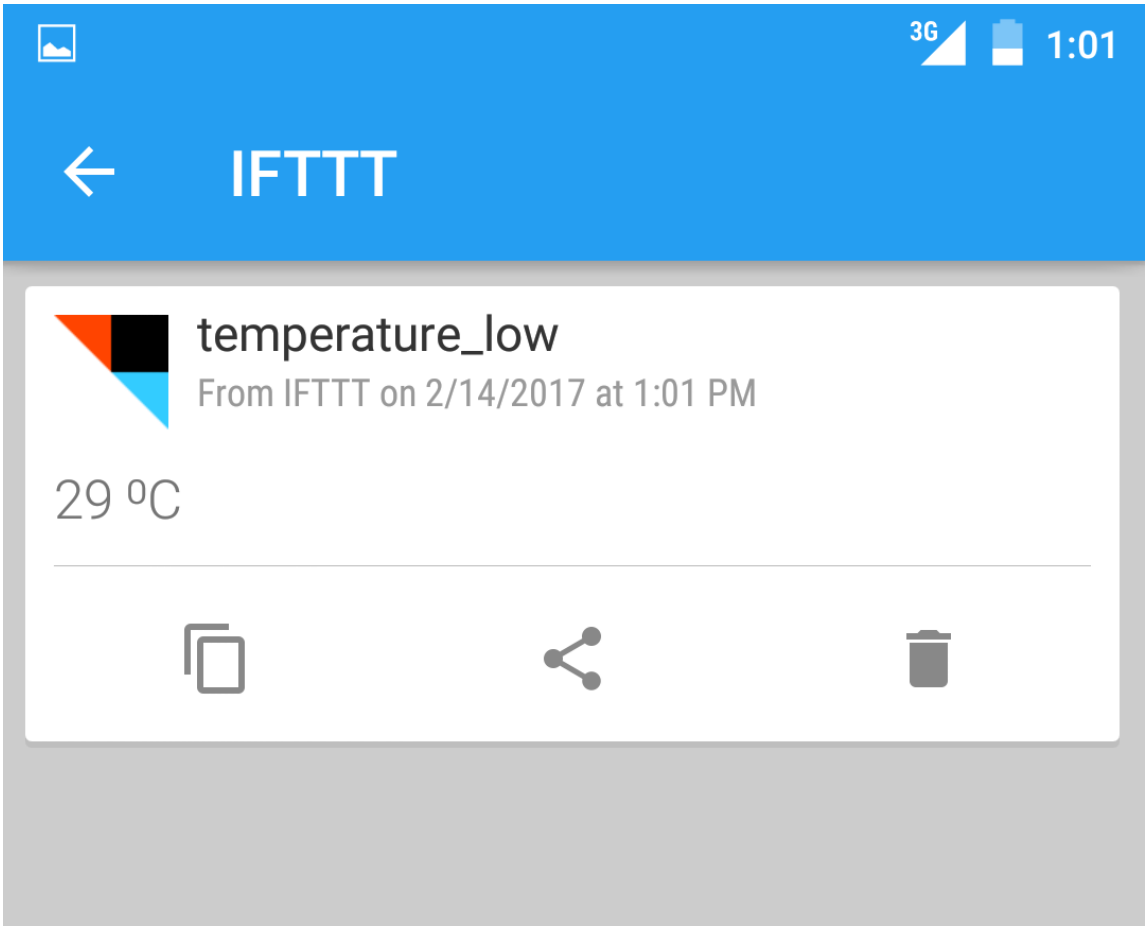
URL: https://maker.ifttt.com/use/[redacted]

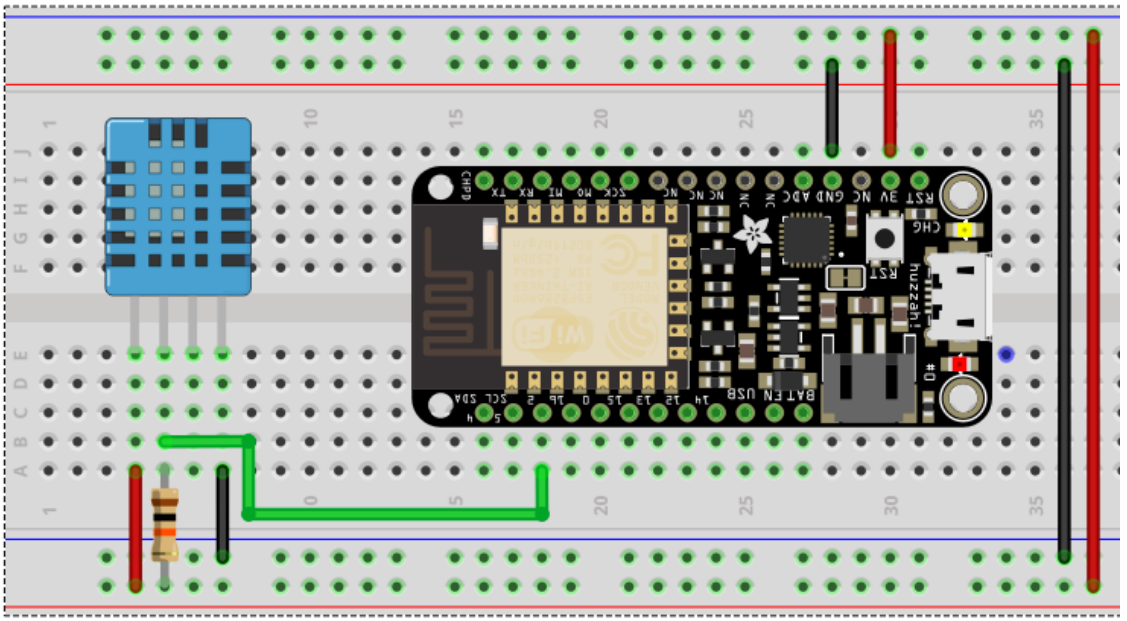
Status: active

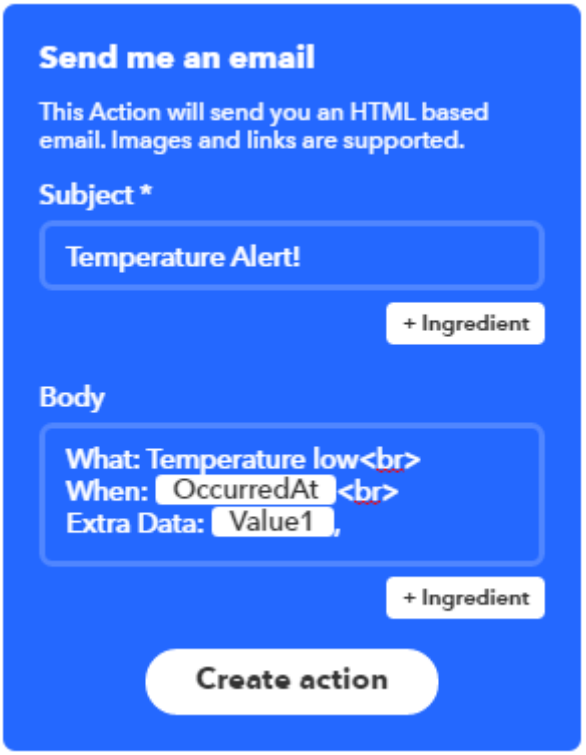
Edit connection


If Maker Event "temperature_low", then send a Pushover notification
67/140
View activity log
Receive a web request
This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)
Event Name
temperature_low
The name of the event, like "button_pressed" or "front_door_opened"


1:01 PM
Tuesday, February 14
temperature_low
29 ºC
1:01 PM


3G
1:01
IFTTT
temperature_low
From IFTTT on 2/14/2017 at 1:01 PM
29 ºC




Send me an email
This Action will send you an HTML based email. Images and links are supported.
Subject *
Temperature Alert!
+ Ingredient
Body
What: Temperature low
When: OccurredAt
Extra Data: Value1 ,
+ Ingredient
Create action

Temperature Alert!

Maker via IFTTT
What: Temperature low When: February 14, 2017...

Maker via IFTTT
to me

12:35 PM

What: Temperature low
When: February 14, 2017 at 12:35PM
Extra Data: 24,

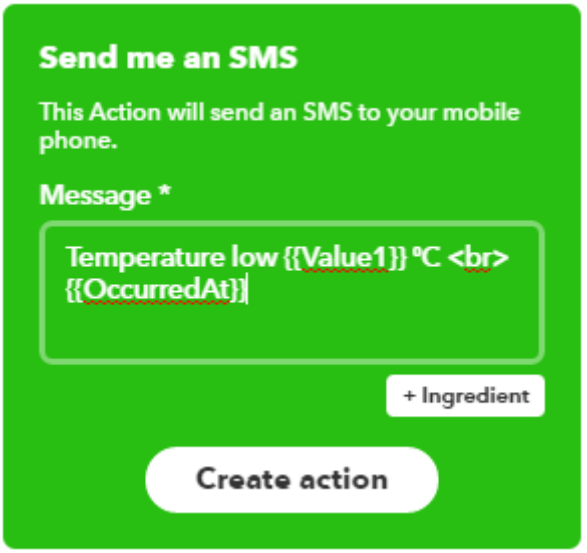
Connect SMS

Enter your cell phone number to receive an activation PIN via SMS. All US carriers are supported. Some carriers outside of the US* are not supported yet. *For non-US numbers, include the leading "00" and country code. If you do not receive the PIN, your carrier may not be supported.

Your phone number

+254

Send PIN

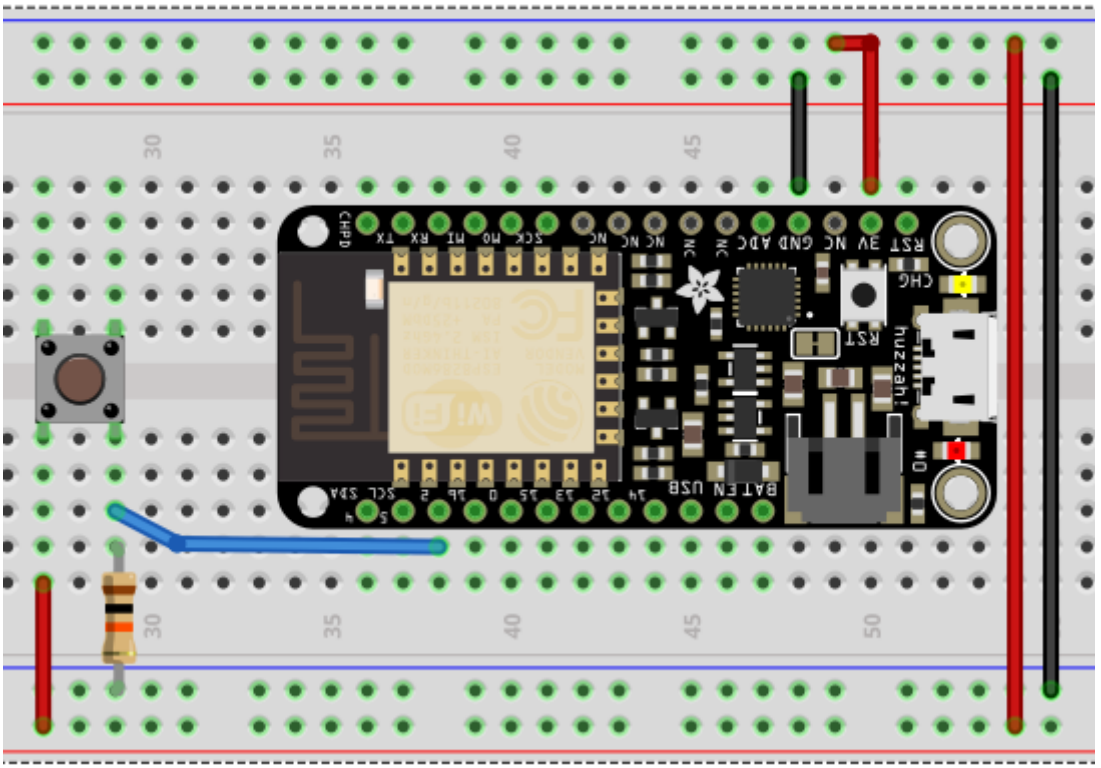


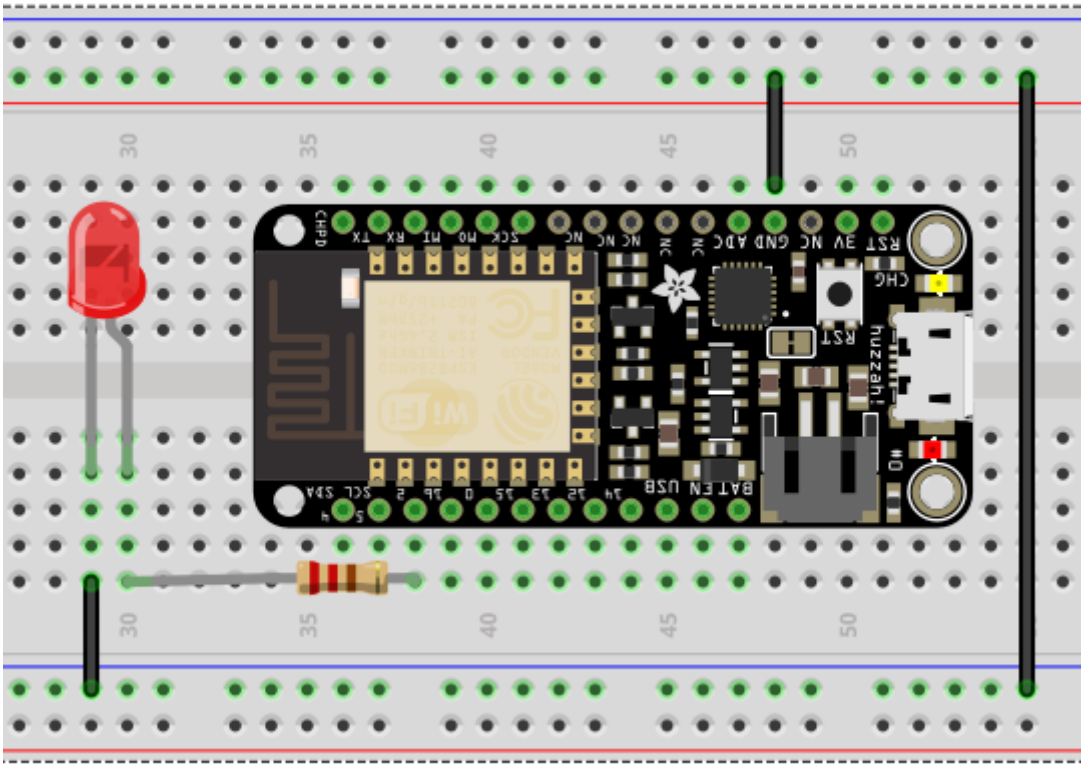
Temperature low 24 ?C
February 14, 2017 at 01:56PM

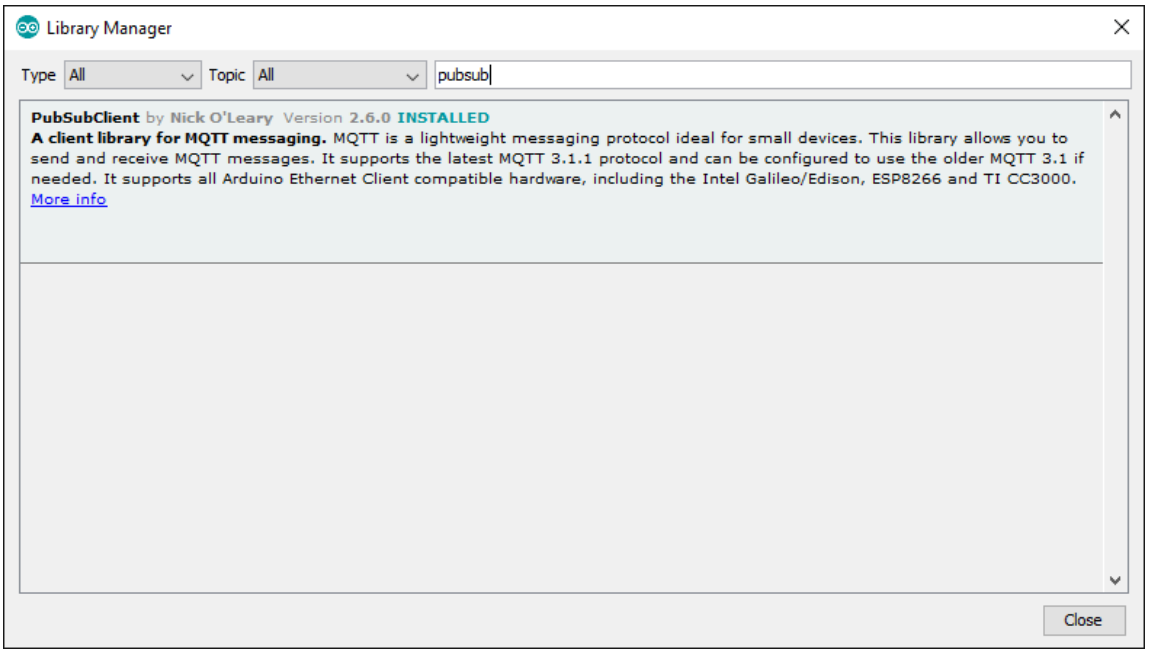
Now via SMS

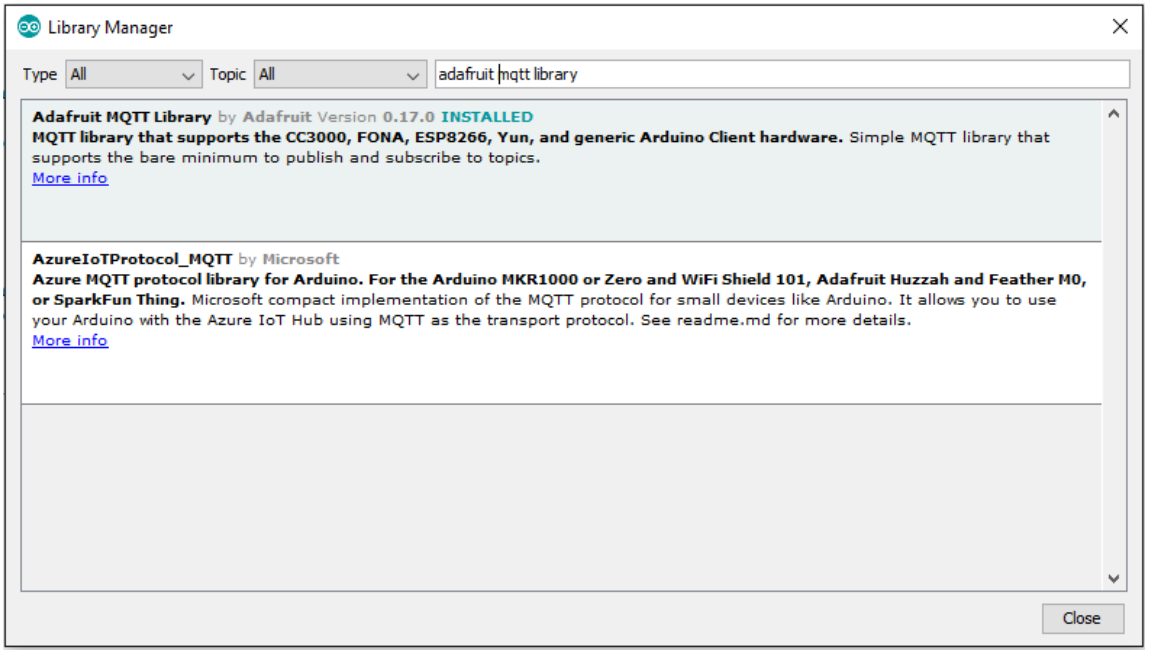
Send an SMS

Chapter 7: Machine to Machine Interactions



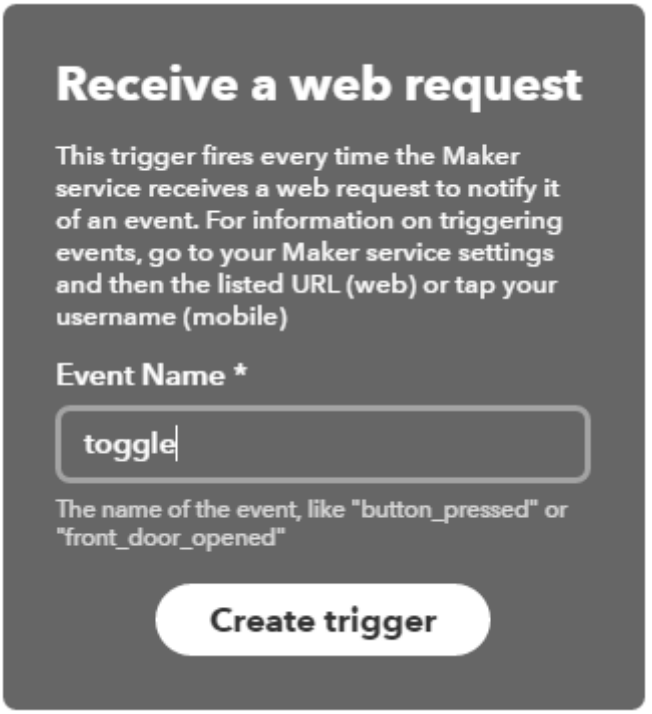

30
35
40
45
50
CHPD
TX
RX
M1
M0
SCK
NC
NC
NC
NC
ADC
GND
3V
NC
RST
CHG
RST
BAT EN USB
SCL SDA
ON


Library Manager
Type All Topic All pubsub
PubSubClient by Nick O'Leary Version 2.6.0 INSTALLED
A client library for MQTT messaging. MQTT is a lightweight messaging protocol ideal for small devices. This library allows you to send and receive MQTT messages. It supports the latest MQTT 3.1.1 protocol and can be configured to use the older MQTT 3.1 if needed. It supports all Arduino Ethernet Client compatible hardware, including the Intel Galileo/Edison, ESP8266 and TI CC3000.
More info
Close


Library Manager
Type All Topic All adafruit mqtt library
Adafruit MQTT Library by Adafruit Version 0.17.0 INSTALLED
MQTT library that supports the CC3000, FONA, ESP8266, Yun, and generic Arduino Client hardware. Simple MQTT library that supports the bare minimum to publish and subscribe to topics.
More info
AzureIoTProtocol_MQTT by Microsoft
Azure MQTT protocol library for Arduino. For the Arduino MKR1000 or Zero and WiFi Shield 101, Adafruit Huzzah and Feather M0, or SparkFun Thing. Microsoft compact implementation of the MQTT protocol for small devices like Arduino. It allows you to use your Arduino with the Azure IoT Hub using MQTT as the transport protocol. See readme.md for more details.
More info
Close


yhtomit / Feeds
Actions
Name
Key
Last Value
statetoggle
statetoggle
No Data Available


if this then that


Receive a web request
This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)
Event Name *
toggle
The name of the event, like "button_pressed" or "front_door_opened"
Create trigger

Send data to Adafruit IO

This Action will send data to a feed in your Adafruit IO account.

Feed name *

moistureLog

The name of the feed to save data to.

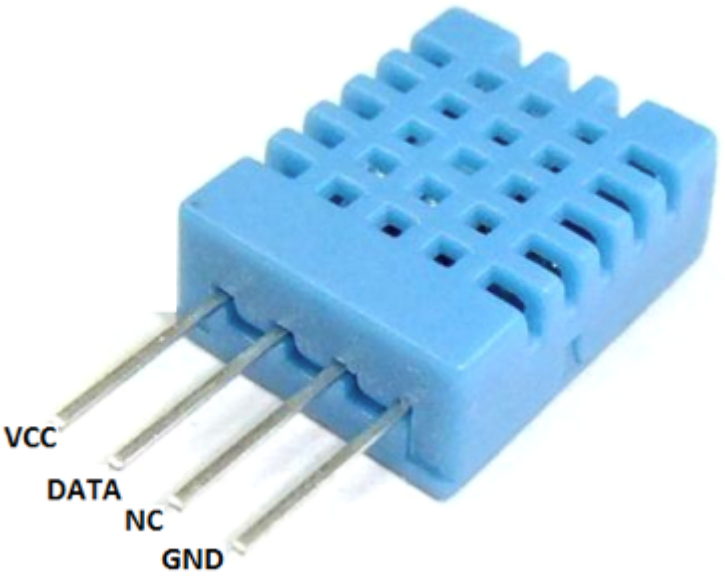
Data to save *

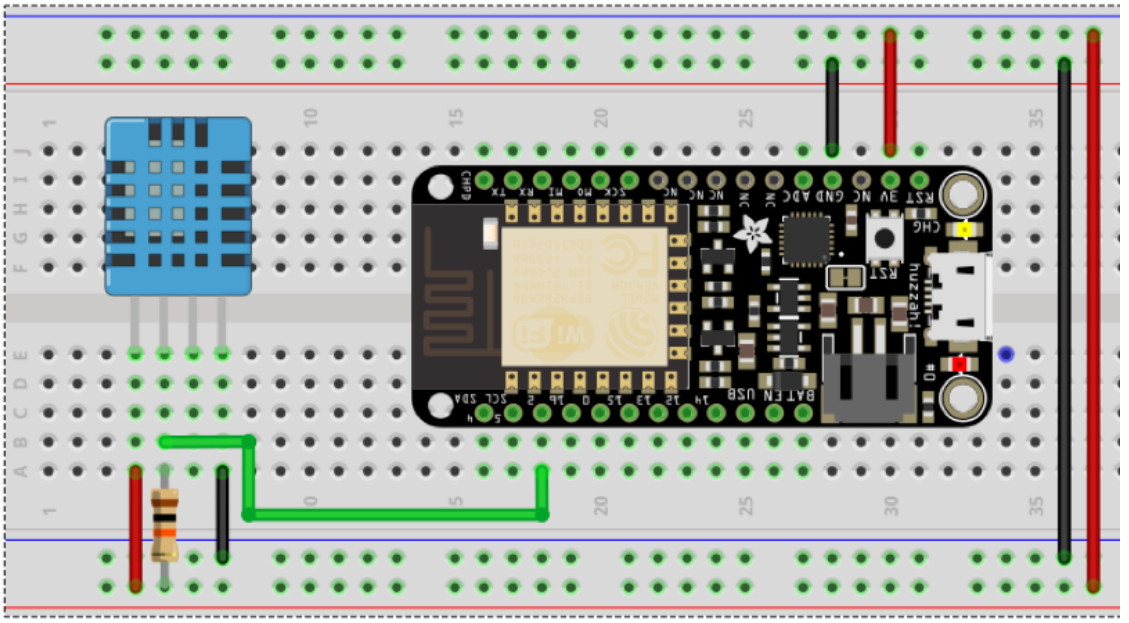
{{Value1}}

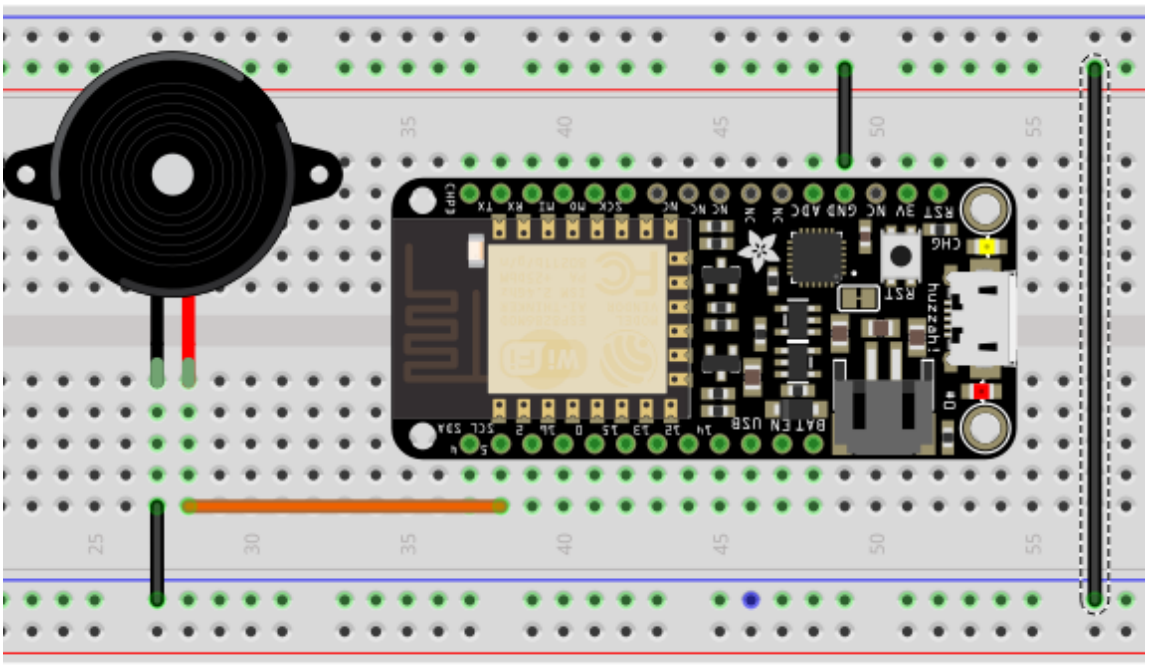
The data to be saved to your feed.

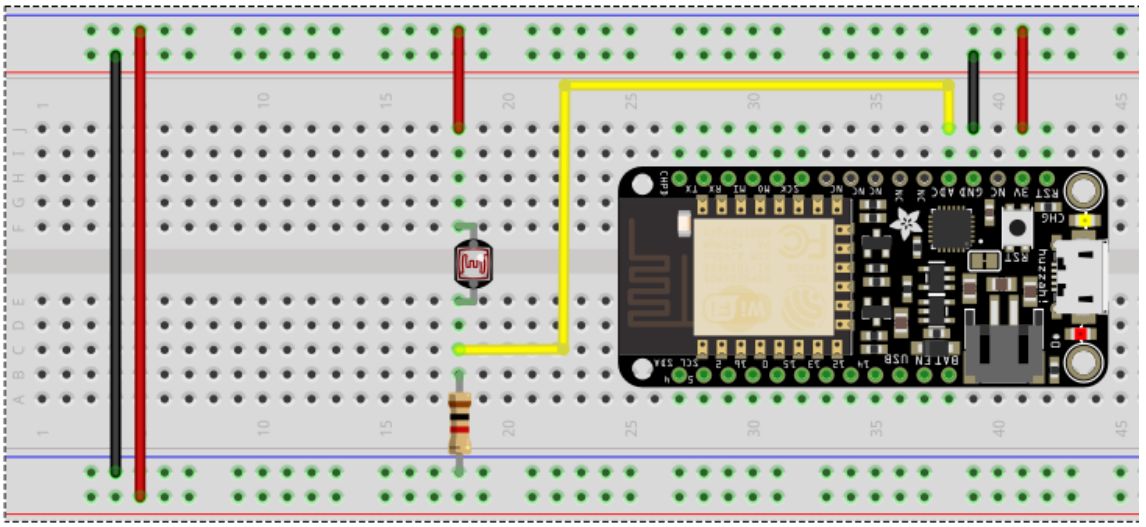
+ Ingredient

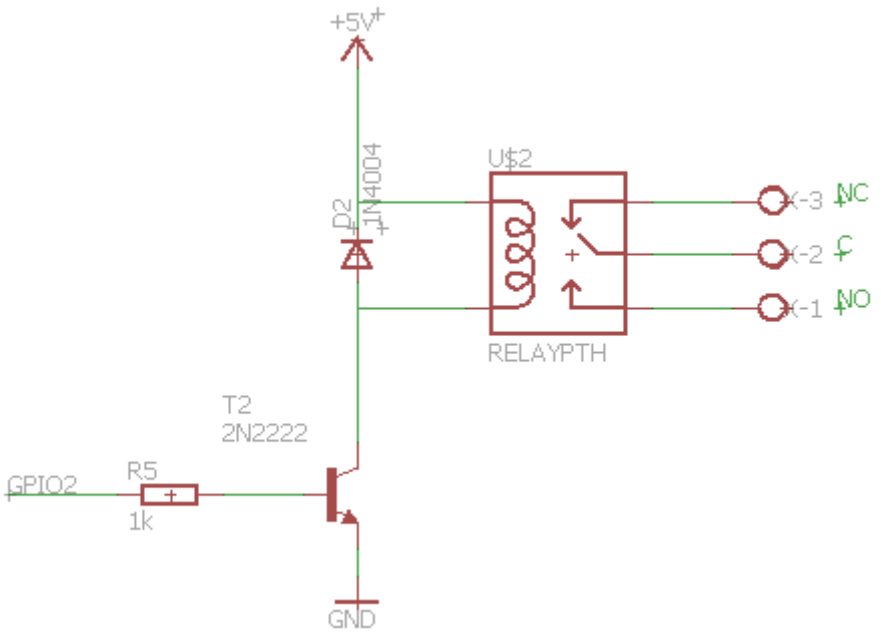
Create action

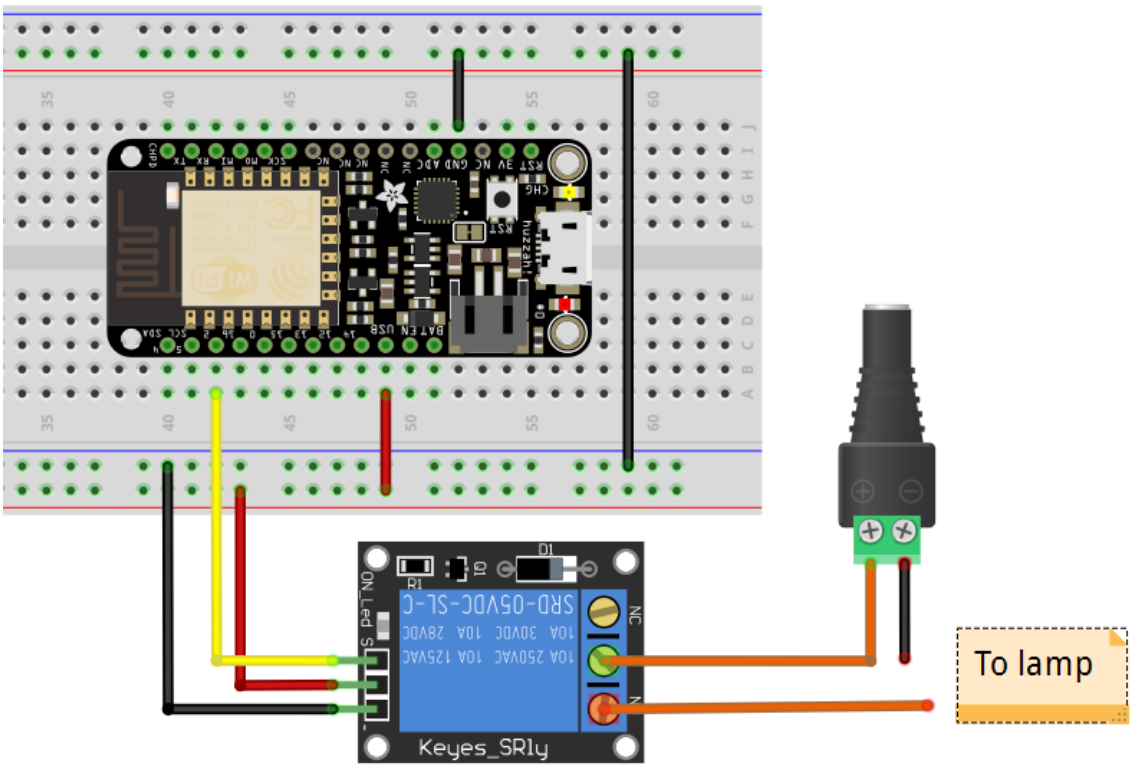

VCC
DATA
NC
GND

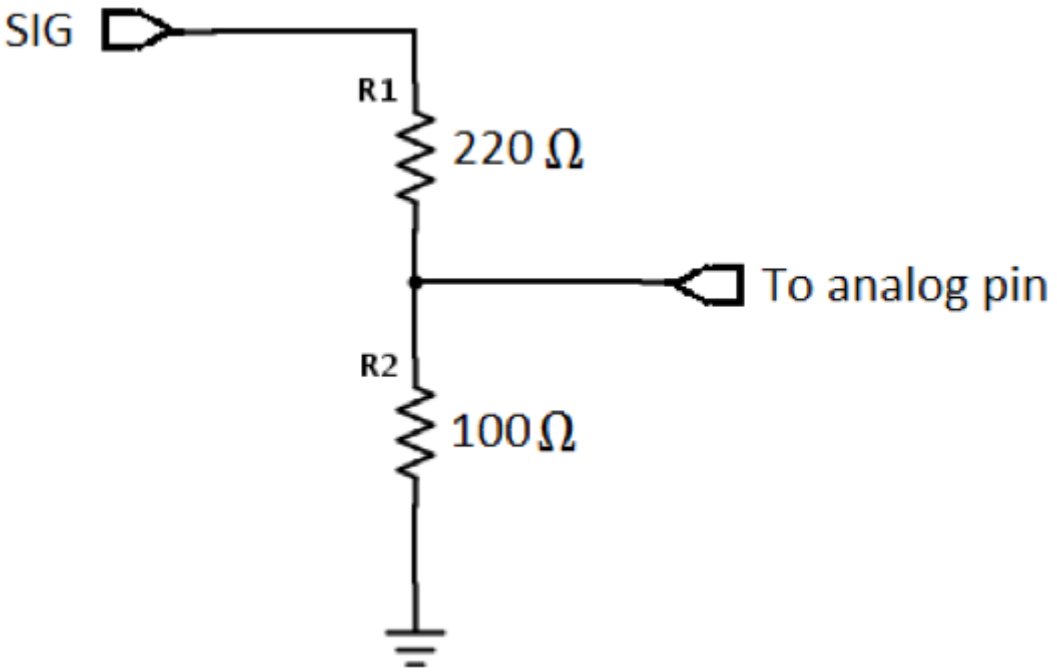


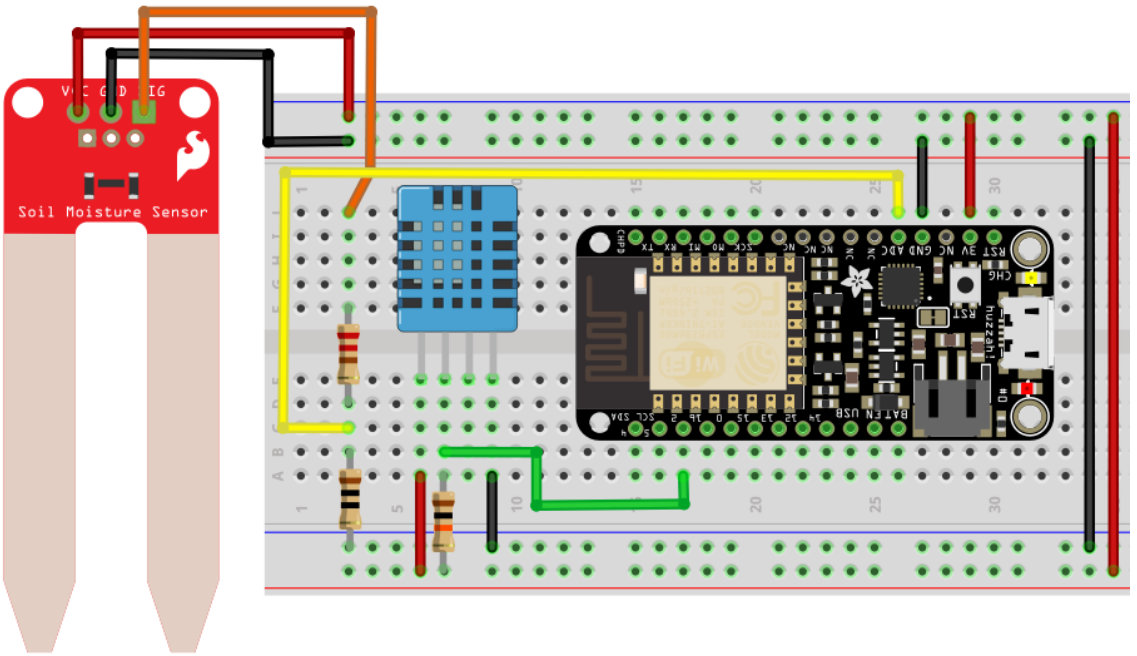


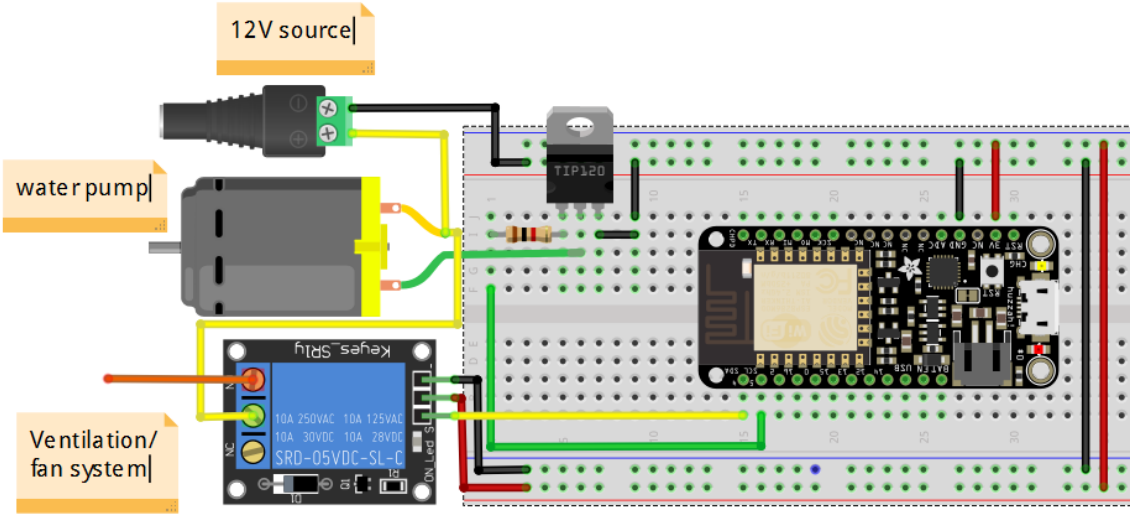



+5V
D2
1N4004
U$2
X-3
NC
X-2
C
X-1
NO
RELAYPTH
T2
2N2222
GPIO2
R5
1k
GND

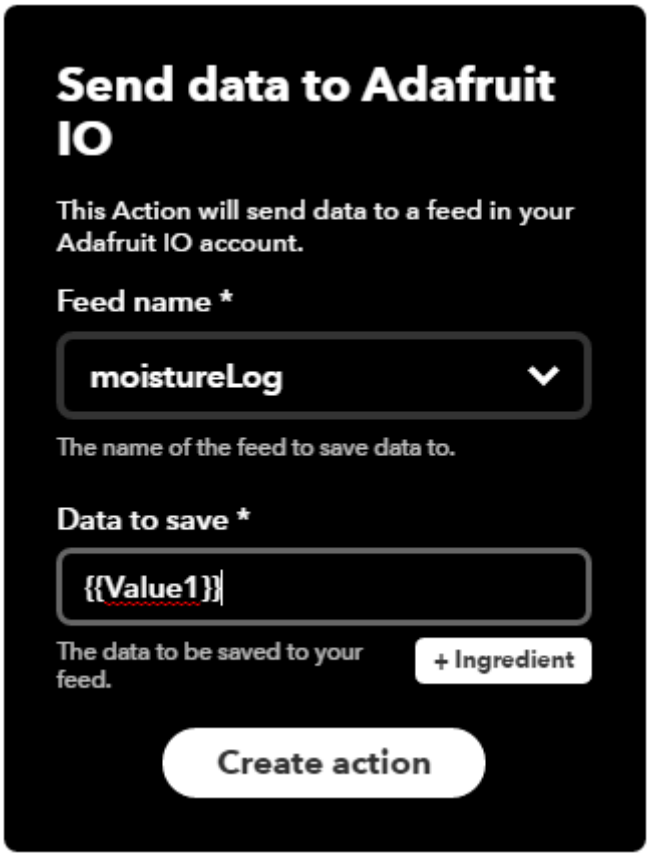

SRD-05VDC-SL-C
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
Keyes_SR1y
To lamp


SIG
R1
220 Ω
To analog pin
R2
100 Ω


Soil Moisture Sensor


12V source
water pump
Ventilation/
fan system
TIP120
Keyes_SRly
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
SRD-05VDC-SL-C


Receive a web request
This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)
Event Name *
moistureLog
The name of the event, like "button_pressed" or "front_door_opened"
Create trigger


Send data to Adafruit IO
This Action will send data to a feed in your Adafruit IO account.
Feed name *
moistureLog
The name of the feed to save data to.
Data to save *
{{Value1}}
The data to be saved to your feed.
+ Ingredient
Create action