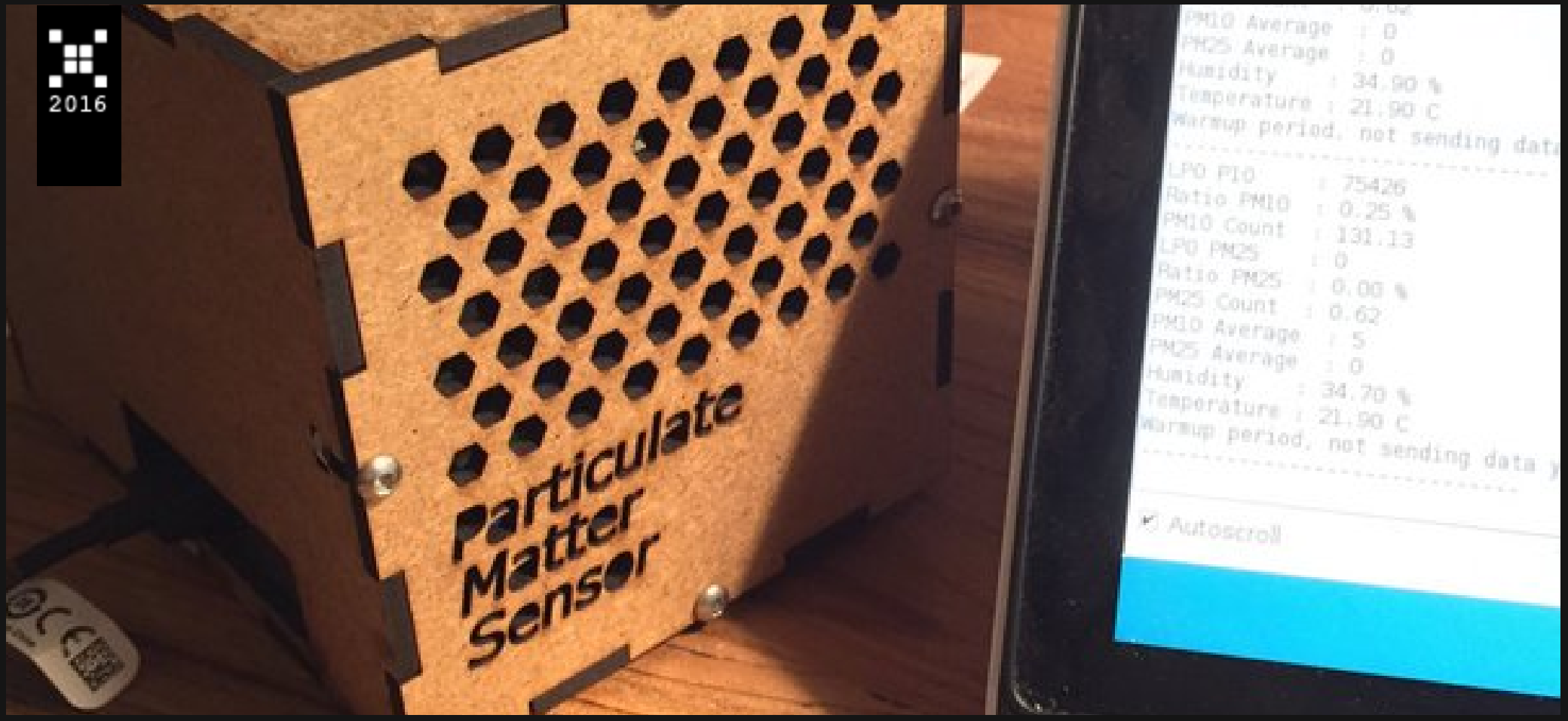
Particulate Matter Sensor Network

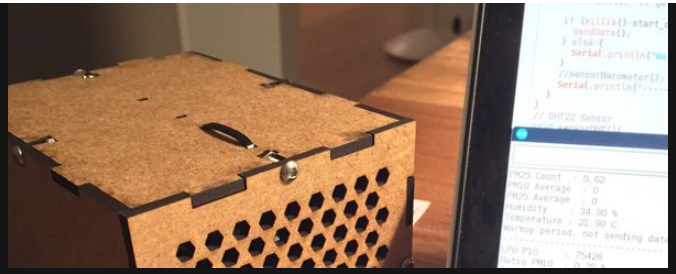
Gathering more data on PM in our environment

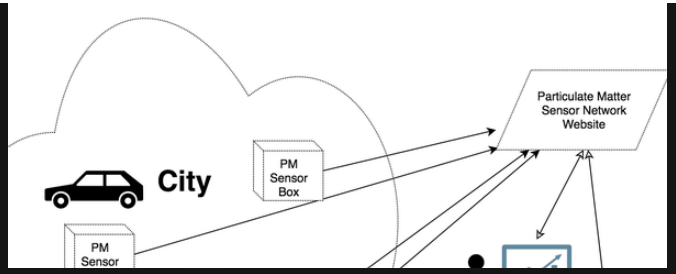
tiefpunkt

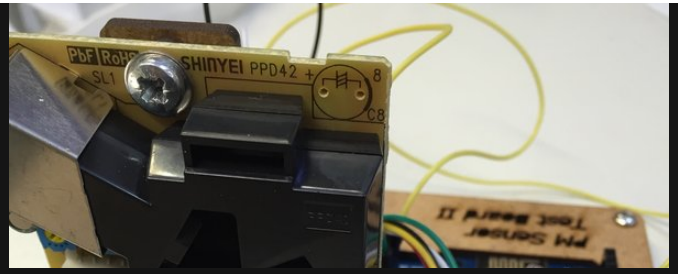
594 views **4** comments **67** followers **5** likes

DESCRIPTION DETAILS FILES (0) COMPONENTS (3) LOGS (4) INSTRUCTIONS (0) DISCUSSION (4)









View Gallery

TEAM (1)

tiefpunkt

Join this project's team

Repository for HW and SW

HARDWARE

ONGOING PROJECT

2016HACKADAYPRIZE AIR QUALITY PARTICULATE MATTER ESP8266 WIFI IOT CITIZEN SENSING POLLUTION

THIS PROJECT IS SUBMITTED FOR

- The 2016 Hackaday Prize
- Get Started: Design Your Concept
- Anything Goes
- Citizen Scientist

This project was created on 04/24/2016 and last updated 5 months ago.

DESCRIPTION

Living in a city, there's a lot of traffic and industry around. And as it is, these create quite a bit of pollution. But how much "dirt" is really in the air that we breath

Now there is a bit of public data on pollution out there, but that is based off of only a hand full of measuring stations in a city, and most of the time, even that is hard to come by. So let's make our own measurements

In order to do that, the idea is to create a cheap sensor unit people can place outside on their balcony or just on their window ledge. The sensor unit will measure PM 2.5 and PM 1.0 values, as well as possibly some other things, and send them to a central service for people to track pollution all over their city.

DETAILS

The project is open source, both hard- and software, so people can build their own units or improve them as

much as they like. At the same time, I'm hoping to produce a few runs of sensor units so people without all the necessary production means can get a unit or two.

Check out the GitHub repository for the source code. I'm using the Arduino IDE to program the ESP8266 WiFi microcontroller. The README is a bit out of date, and there aren't any hardware design files in the repo yet, as it's all still just put together on a WeMos D1 arduino compatible board.

COMPONENTS

1 × ESP8266 Board (I use the Wemos D1)
Processing and wifi connectivity are the key

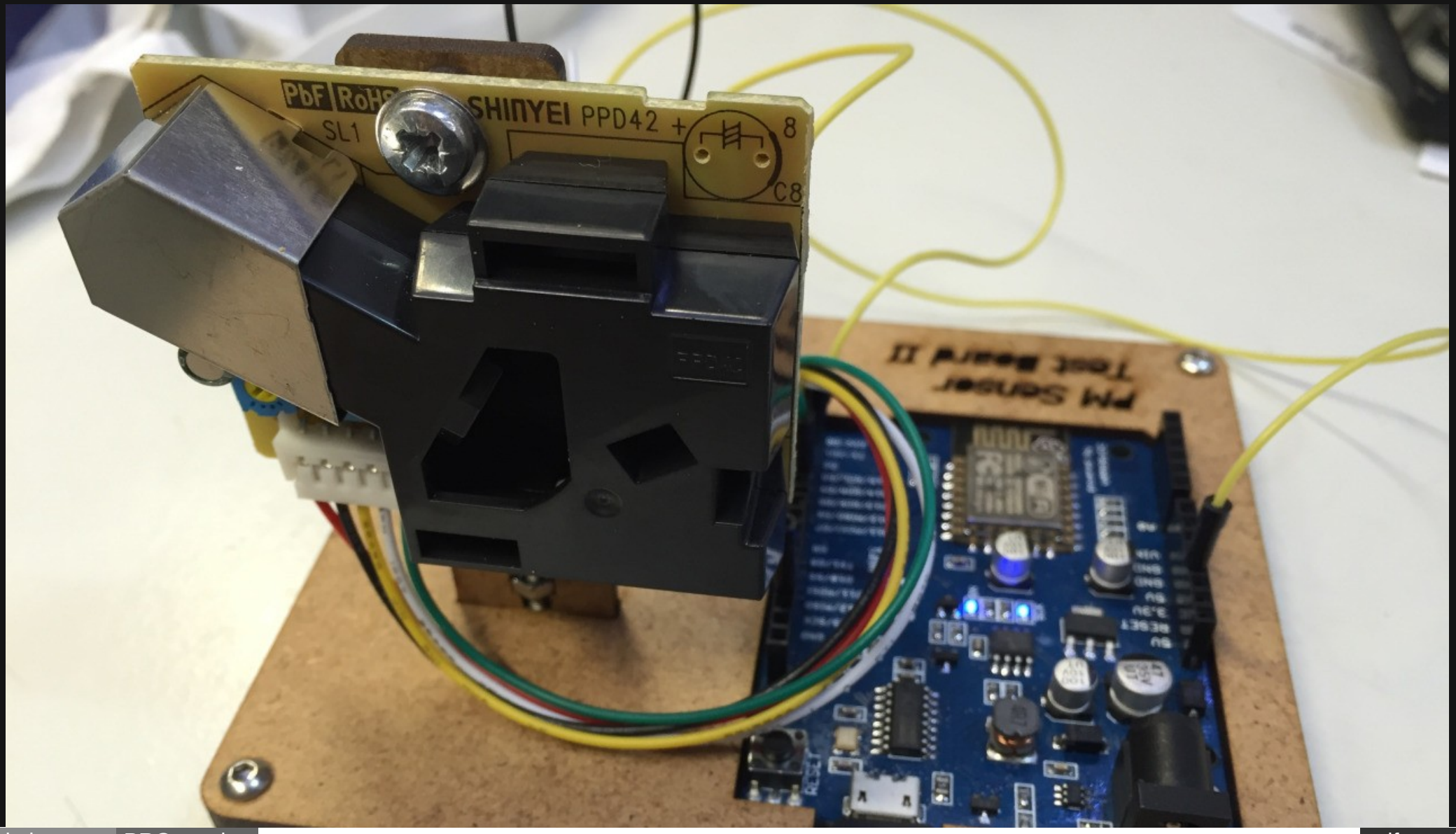
1 × Shinyei PPD42NS
Particulate Matter sensor

1 × DHT22
Temperature and humidity sensor

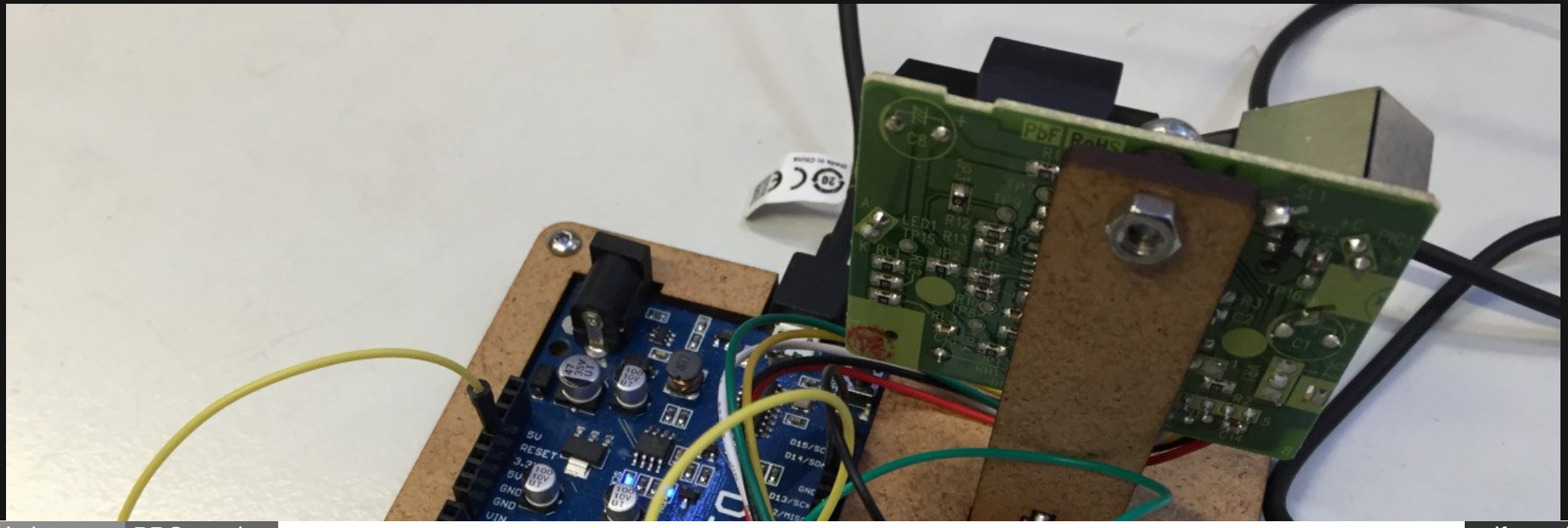
PROJECT LOGS

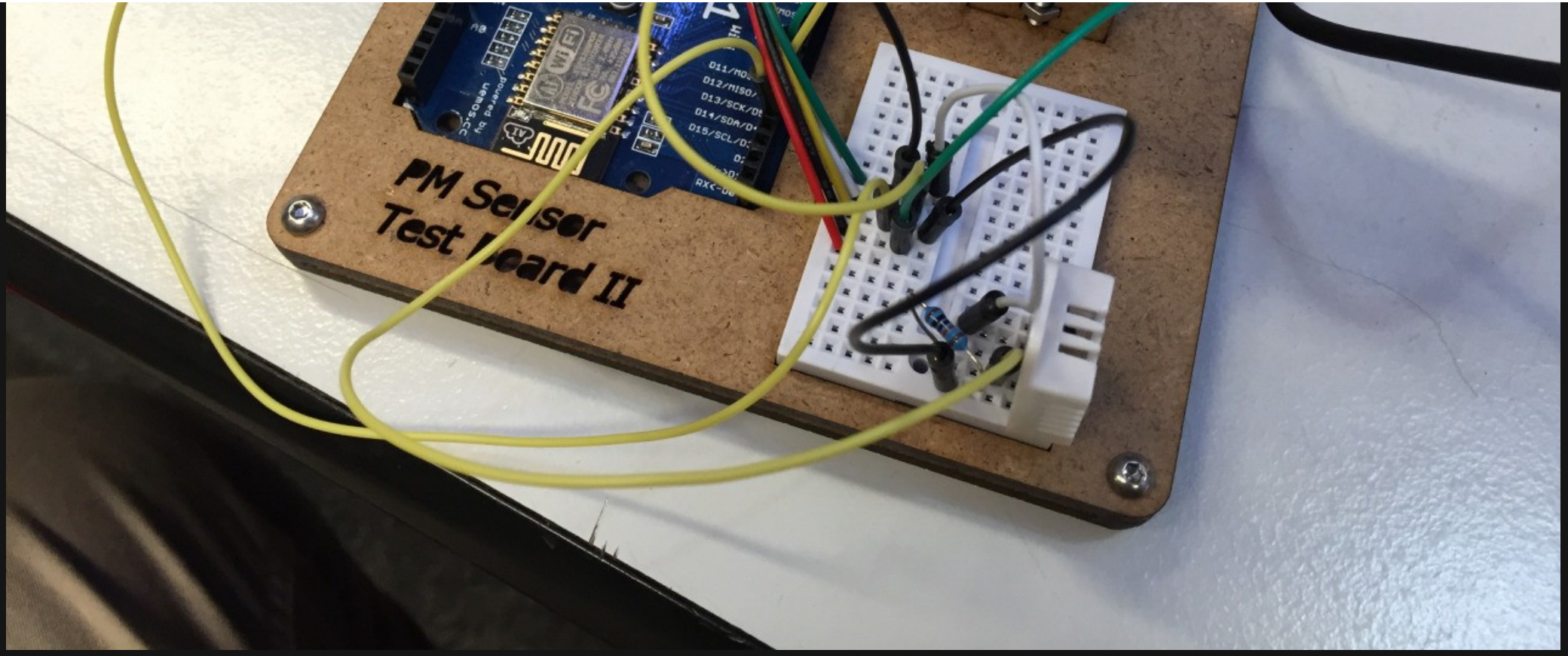
Getting more data from the PPD42NS

tiefpunkt • 05/27/2016 at 06:09 • 0 comments

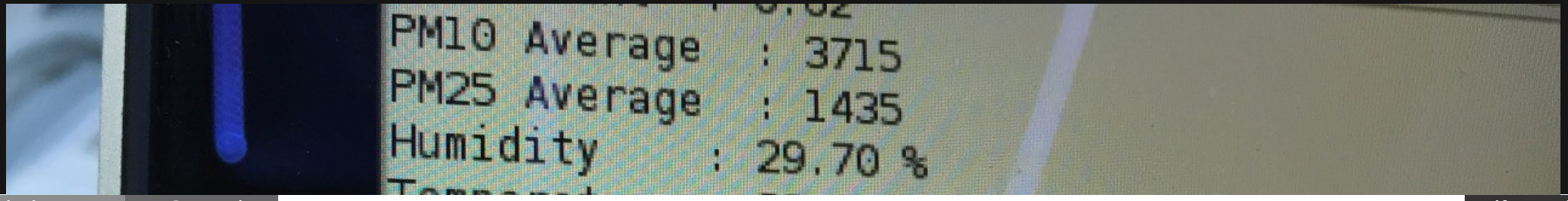


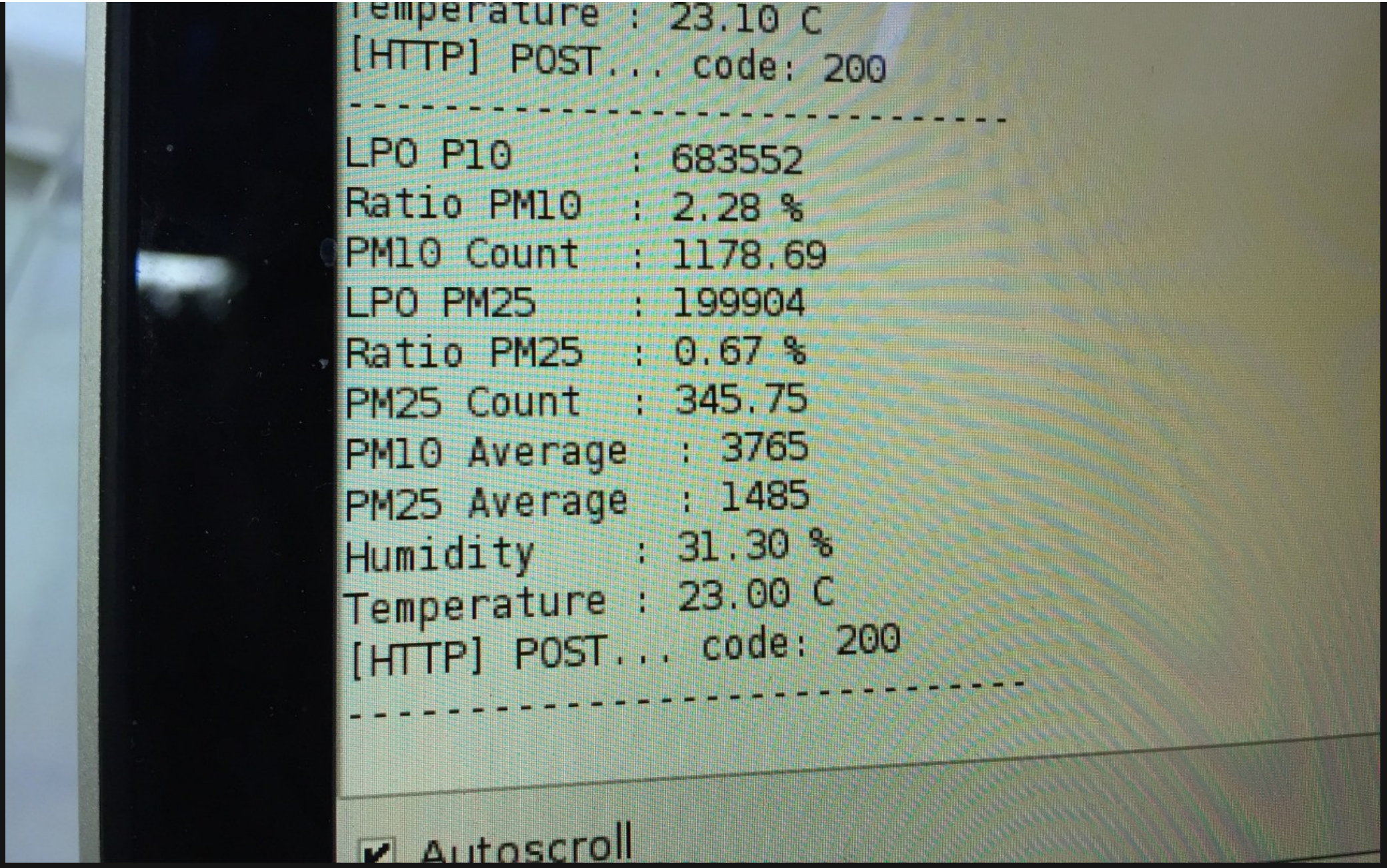
Having decided on the Shinyei PM sensor, I took a closer look at the sensor. Turns out, the PPD42NS can differentiate between PM1.0 and PM2.5, but unfortunately, the version in the Grove kit only measures PM1.0. The sensor has two outputs, but the adapter cable that comes with the sensor only connects one of these outputs to the Grove connector, even though the connector allows for two digital pins to be connected (are you reading this, Seeed Studio?)





Now I wanted to gather data from both outputs, so I decided to get the "normal" (non-Grove) version of the sensor. After receiving it, I built a similar platform as for the Sharp sensor and started collecting data again. Again, I'm using the same online platform to collect and display the data. I also added a DHT22 again, to figure out whether having a fan, as in the first prototype, or not, as this version, influences my measurements.

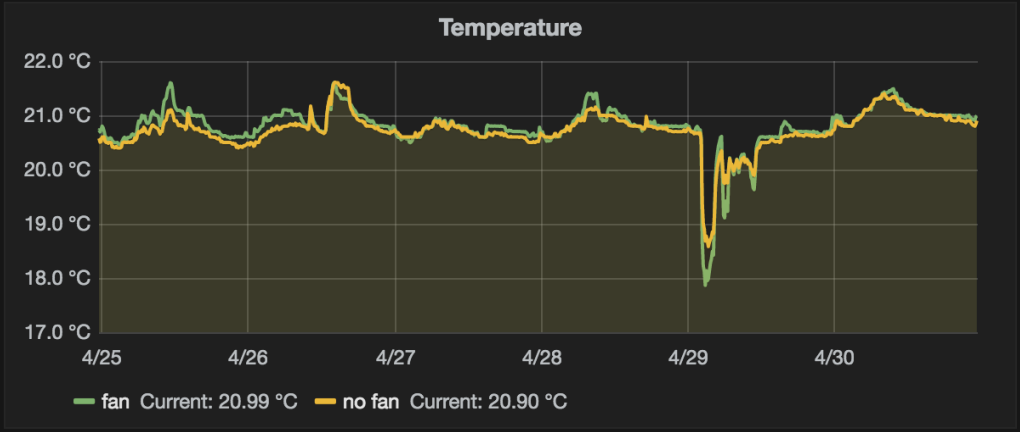


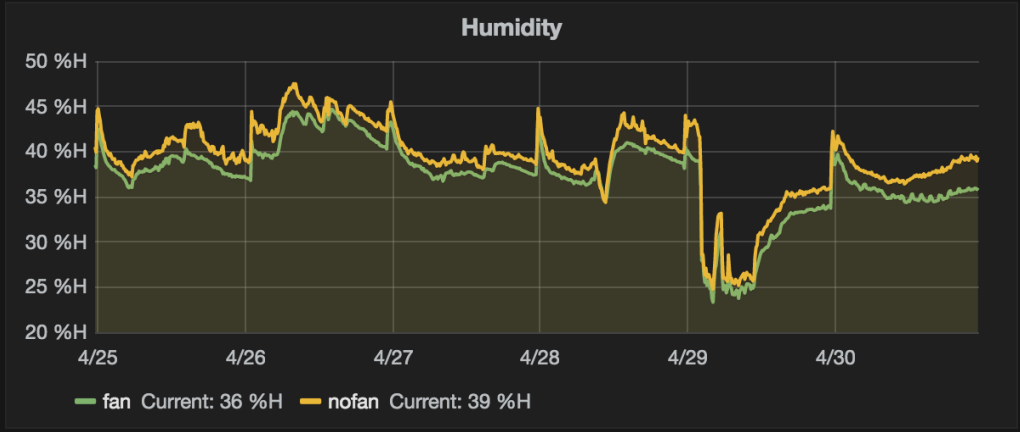


Well, there doesn't seem to much difference between the fan and no-fan version for the PM1.0, which is a pretty neat finding.

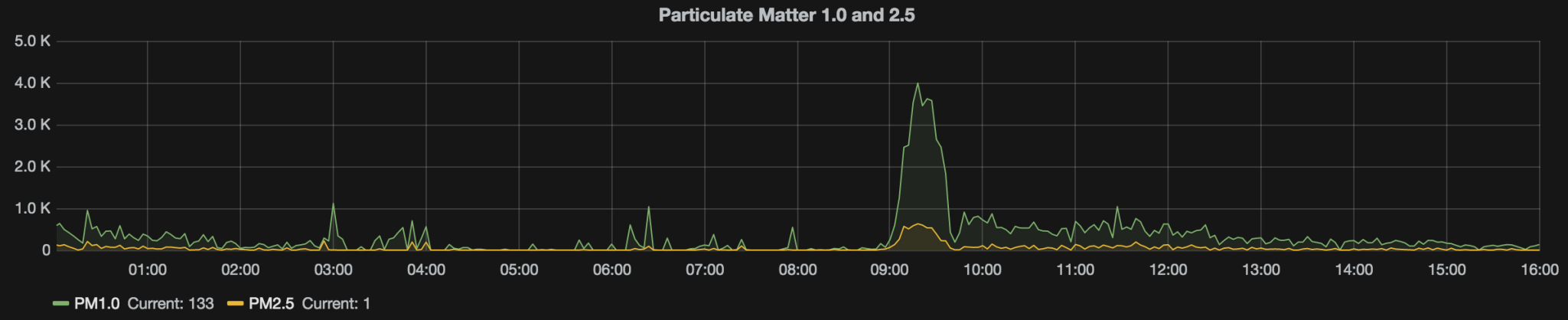
Temperature doesn't seem to be to much different, but the humidity readout is about 1-

3 percentage points lower on the version with the fan than on the non-fan version. That makes sense, but could also just be an inaccuracy in the sensors.





Finally, since the new sensor also has PM2.5, here's that (together with PM1.0 measurements):

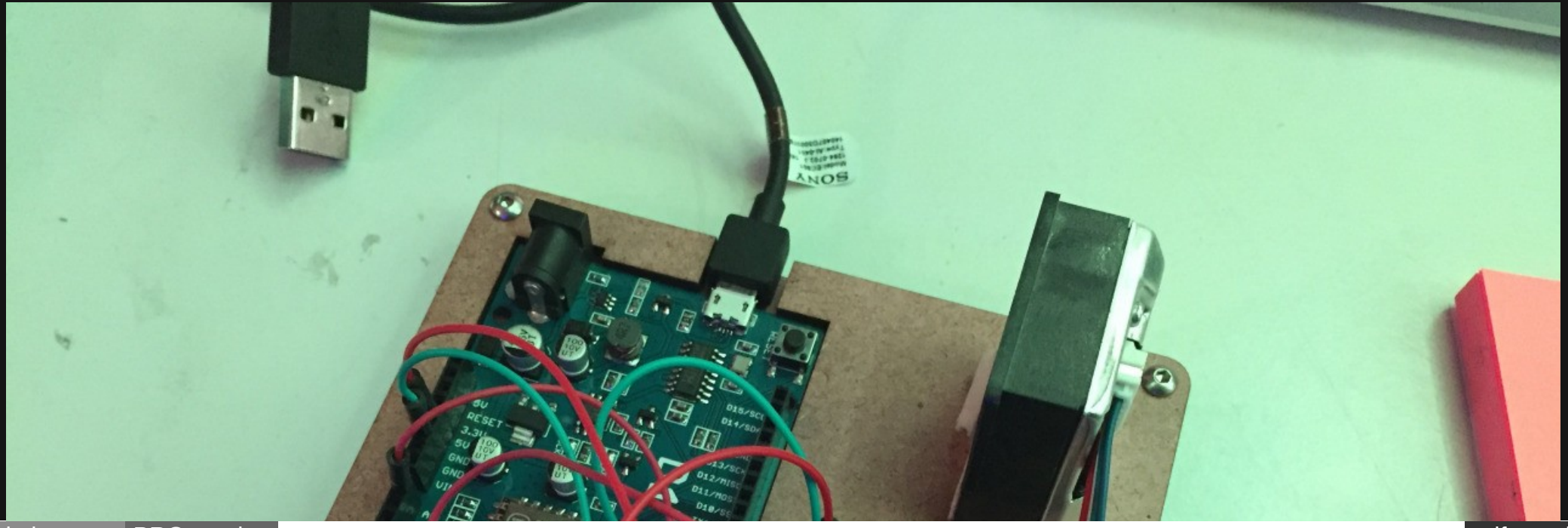


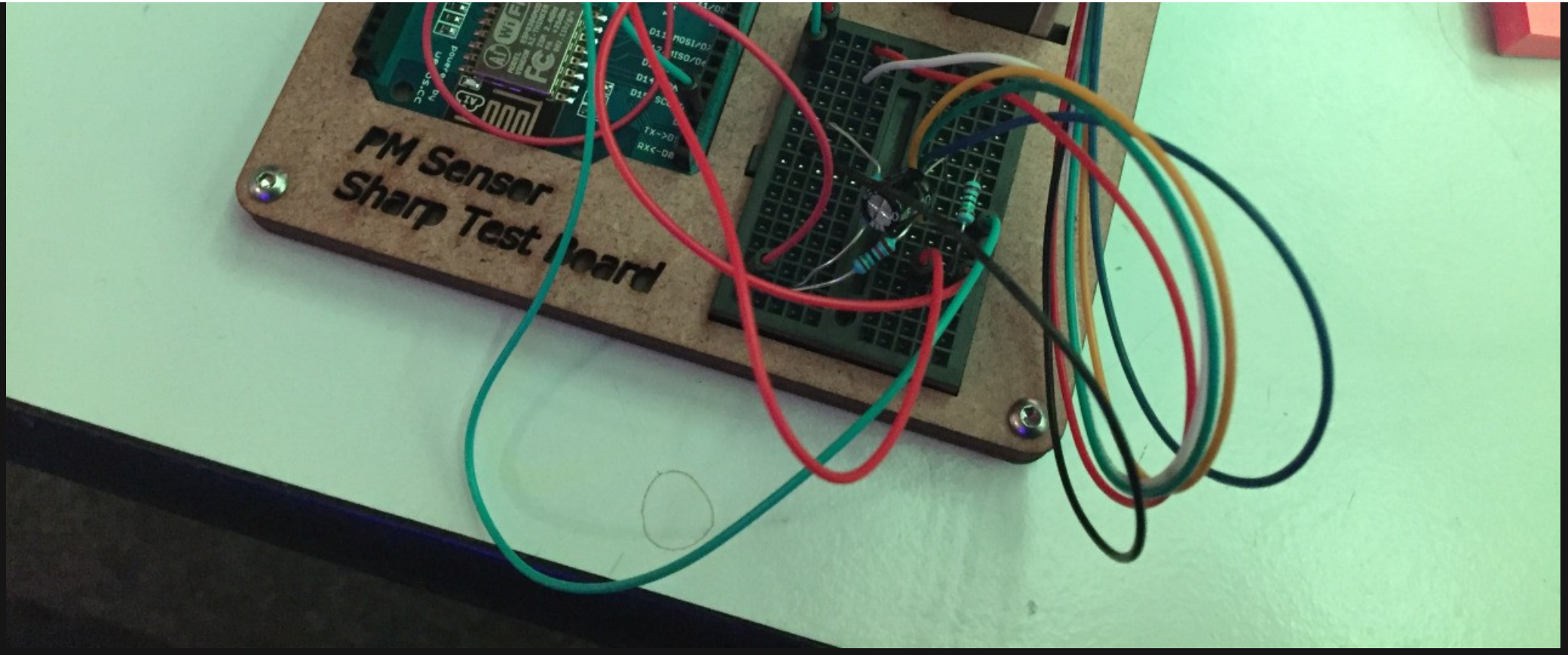
Now the two seem to be correlating, but does it have to? Not quite sure, but it does somewhat make sense to me, especially in the home environment that I currently have these two sensors set up in. Maybe it'll be different once they are in the outside world, but I guess it's time now to do a small batch of sensors and get them out into the world. But more on that later.

Different sensors?

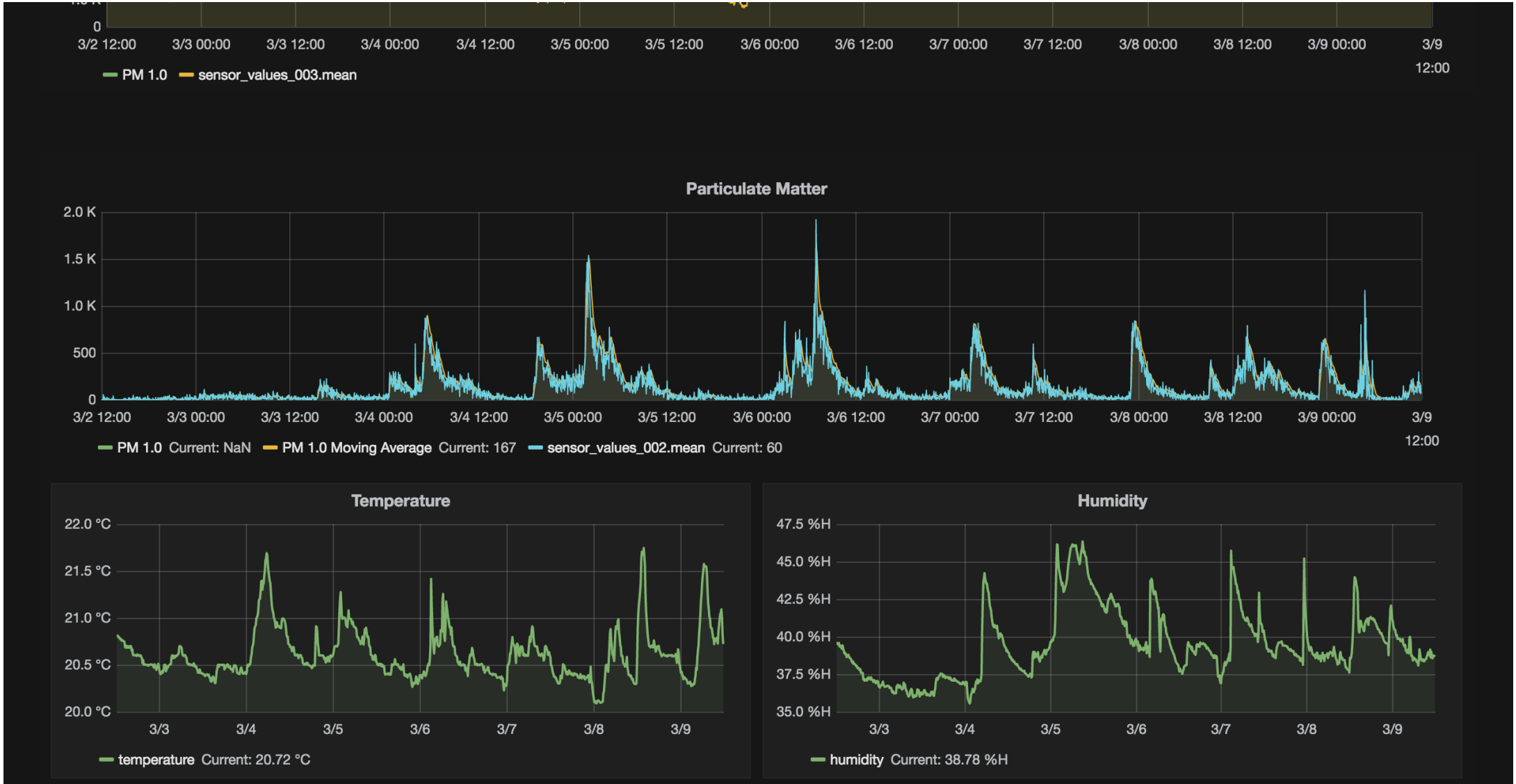
tiefpunkt • 05/23/2016 at 04:57 • 0 comments

So far, I've used the Shinyei sensor for measuring PM. There are a bunch of other sensors out there, but most are fairly hard to get, at least if you're in Germany like I am. There's one other one that's fairly common and available, the Sharp GP2Y1010AU0F. It's in about the same price range as the Shinyei Sensor from the Grove Kit, so I decided to give this one a try as well.





I once again used a WeMo D1 ESP8266 board as microcontroller. The sensor needs some resistors and stuff, so I added a small breadboard, and lasercut a baseboard so it doesn't look like a bunch of tangled wires. The software setup was similar, the sensor gathered the data, and sent it over WiFi to a webservice.



Unfortunately, the measurements I got didn't really show anything meaningful. If you look at the screenshots above, the top shows the measurements of the Sharp sensor, while the bottom screenshot shows the measurements of the Shinyei sensor during the exact same timeframe. The bottom one also includes temperature and humidity measurements (the test board had no DHT22 on it). As you can see, there are quite a few big spikes in the bottom screenshot, which relate to things I did in the apartment during that time, such as

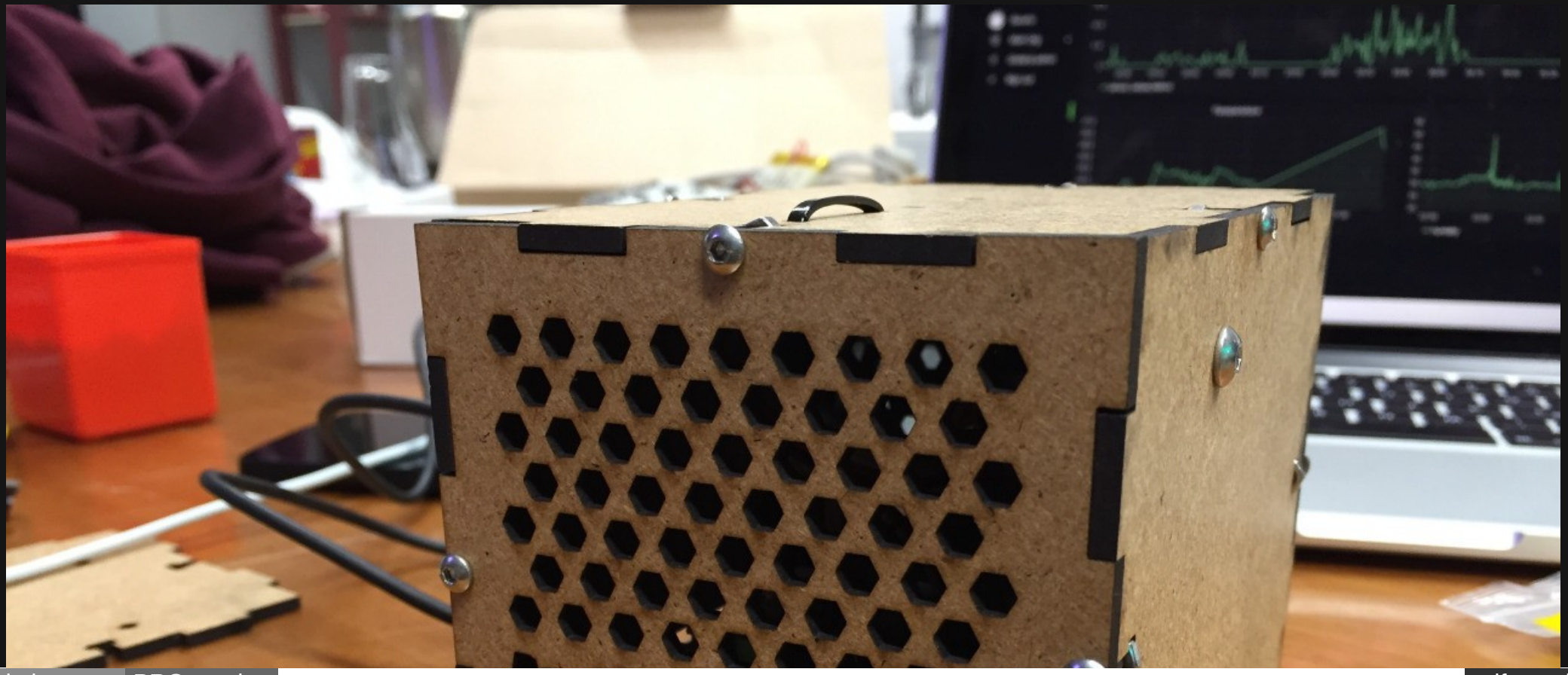
cooking or vacuuming. The Sharp sensor did not show any of these spikes, which seemed unrealistic to me.

Long story short, I decided to stay with the Shinyei sensor. On the long run, I will probably check out other sensors again, but for a first version, I'll just use that.

(Edit: added measurements and some explanations)

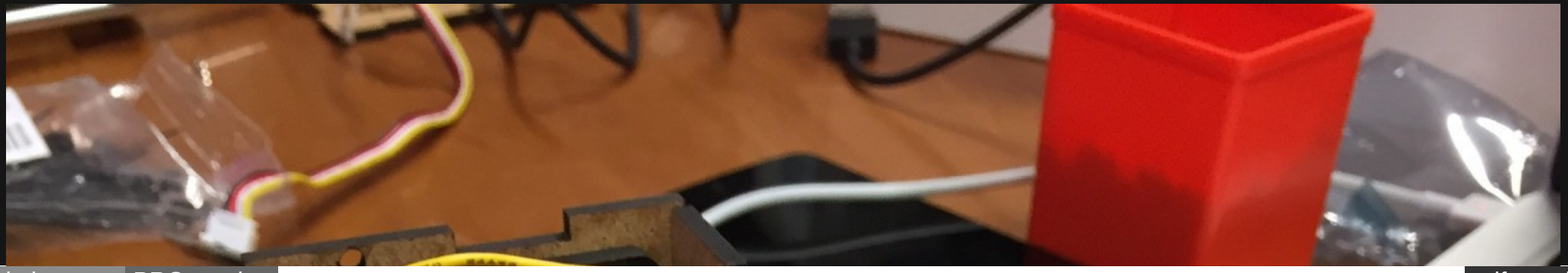
Loosing the laptop

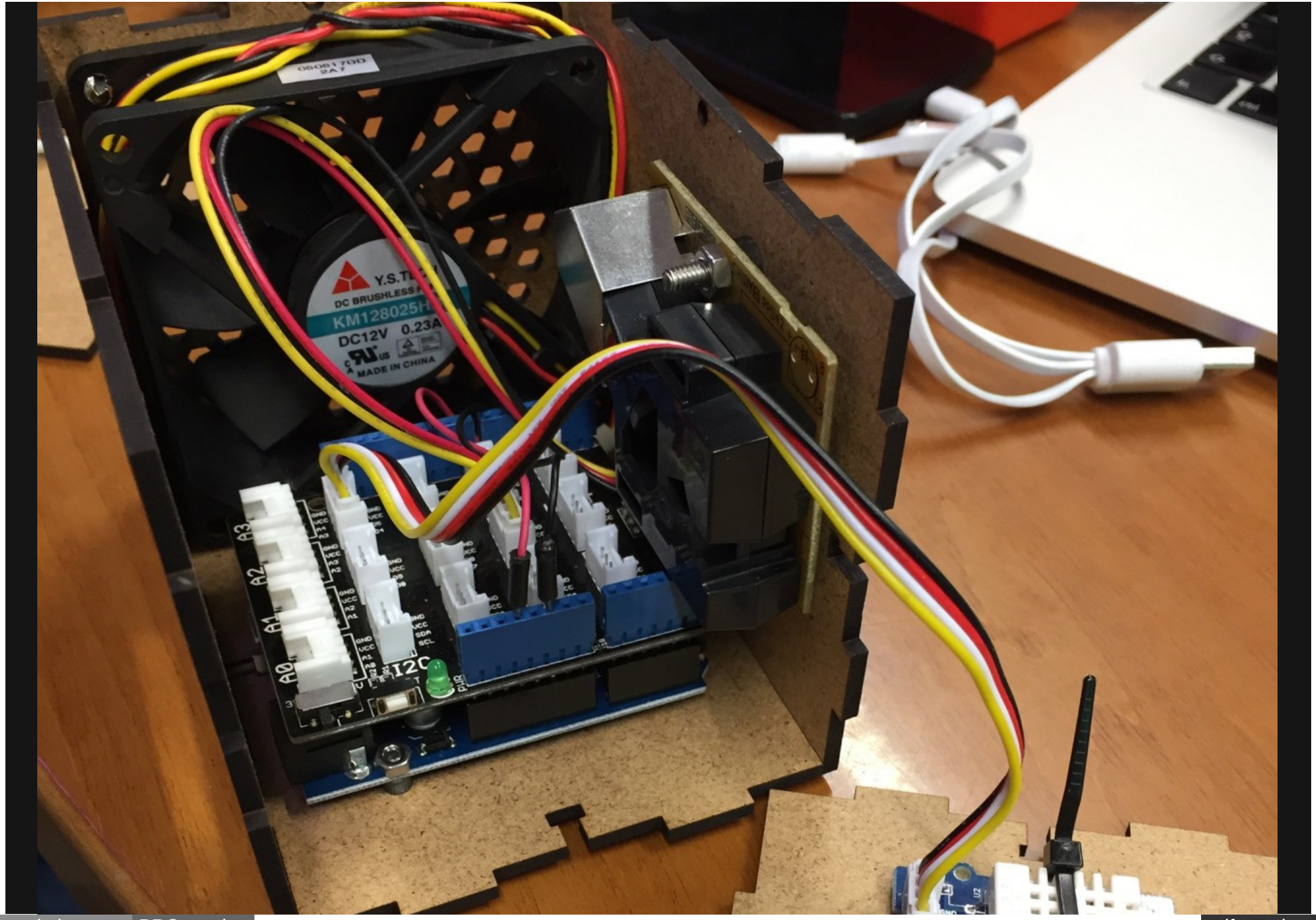
tiefpunkt • 05/15/2016 at 18:22 • 0 comments



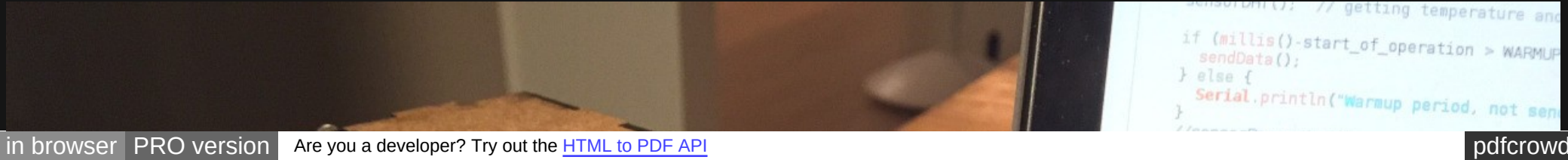
The first prototype from CCCamp used a laptop to connect to the internet. Now that seemed like a bit too much to have for a sensor station, so that was the first thing to go.

I went through a bit of iterations, trying out different hardware along the way. Since the PPD42NS (PM sensor) I had came as part of the LinkIt One Grove Kit, and I had gotten a free LinkIt One, I tried to set that up first, without any success. So it came down to using the ESP8266 for the job (using a WeMo board as an adapter).





Playing around with software and different methods to get the measurements from the sensor, I finally built another prototype. Using only two different sensors this time, a PPD42NS for PM, and a DHT22 for temperature and humidity, I built a small box to put it all in. I also added a fan to get air flowing through the thing. I'm not entirely sure how that influences the measurements, so there will be some more tests without a fan, but for now that seemed sensible.



The ESP8266 outputs the measurements on its serial interface for debugging, but also sends it to a webservice. I

didn't use OpenSenseMap this time, because I wanted more control over the data. Instead, I set up my own little thing using InfluxDB and Grafana, to log all the data. I had it running in my kitchen for a while, which seemed like a better testing environment than putting it straight outside, since I have more control over what is happening.

You can find the code and some more on the GitHub Repository, if you want to give it a try yourself: https://github.com/tiefpunkt/airqualitystation. Note that the readme isn't really up to date, need to get that fixed once I find the time.

View all 4 project logs

ENJOY THIS PROJECT?

Share

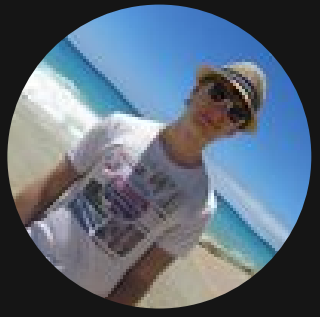
DISCUSSIONS

Log In or become a member to leave your comment

Log In/Sign up to comment

Armagan C. wrote 08/13/2016 at 23:31

i just purchased the same dust detection module to upgrade my current air quality sensor, is the fan necessary?

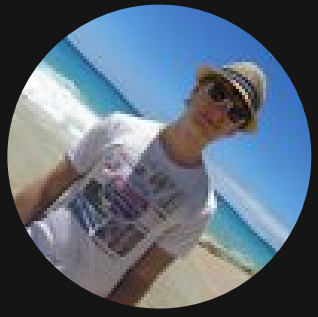


tiefpunkt wrote 08/15/2016 at 16:45

To be honest, I'm not exactly sure. Because the case is kinda small, I thought it would be. But I'll do some trials once I have more sensors, and then I can make a comparison. There were some articles that suggested the precision get better with a fan, but that's kinda hard to tell, at least with my capabilities and gear.

∫(Diego F. G)dx wrote 08/13/2016 at 21:49

Hi dude awsome project , could you give me some advices? , because I'm very keen on make this project but using a Bluetooth conexion with mobile phone.



tiefpunkt wrote 08/15/2016 at 16:46

I suppose you could use a standard Arduino and add a bluetooth shield? The Aircasting folks did something in that direction, maybe that might be a good starting point

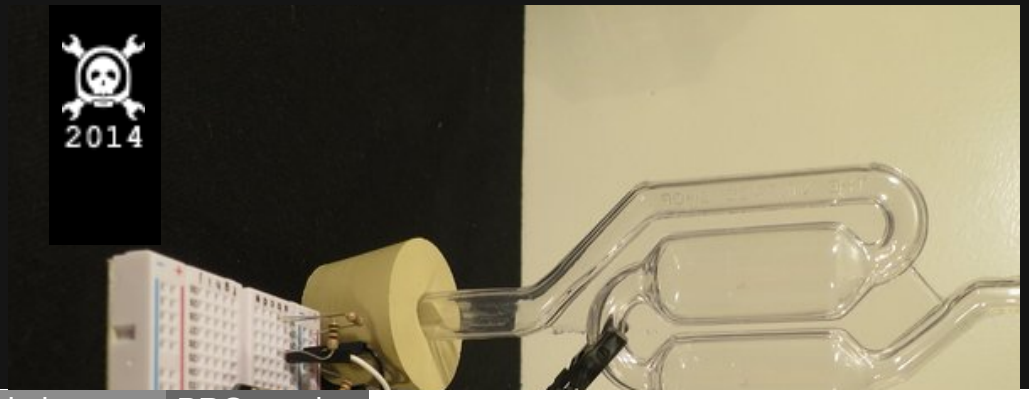
Similar Projects

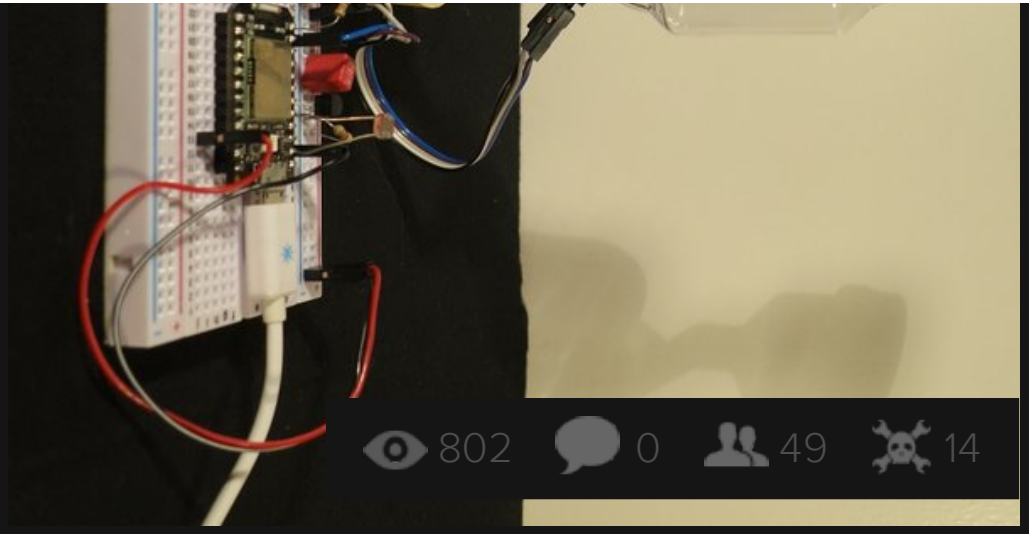
The Cave Pearl Project

by Edward Mallon

Peril-Sensitive Sunglasses

by Minimum Effective Dose





Heimdalr, Multi-Mead-Watcher

by Alex

Going up?

About Us Contact Hackaday.io Give Feedback Terms of Use Privacy Policy Hackaday API