A guide for using IPMC
IOT Particle Monitoring Cube (version 19.02)

Dr. Suresh Dhaniyala
Clarkson University
2019

Table of Contents

Overview 3

1. Code flashing 3

1.1 Particle workbench 4

1.2 Downloading code 5

1.3 Setting up Wi-Fi 8

1.4 Solving Wi-Fi connection issues 9

1.5 Resetting SD card timestamp issues 11

2. Database 12

2.1 Device ID used in this project 12

2.2 Database info 13

Appendix 16

A.1 Microcontroller 16

A.2 Circuit Board 17

A.3 Sensors and electronic components 18

A.3.1 PM Sensor 18

A.3.2 Temp/Humidity Sensor 19

A.3.3 SD Card 19

A.3.4 Real-Time Clock 19

A.4 Assembling 20

Overview

This guide will help you understand how to operate, install, flash code and review data of an air quality monitoring unit. This unit is comprised of separate sensors. These sensors monitor temperature, humidity and particle concentrations. All of this data will be uploaded to a MYSQL server. Once the unit is operating, the unit takes continuous readings and sends them to the server. This project needs Wi-Fi or cellular reception to work properly. This unit will have the ability to send the data collected to a server and an SD card.

1. Code flashing

Before flashing any code, see the Appendix (A.4 Assembling) to access the Particle Argon board that you will be flashing code to.

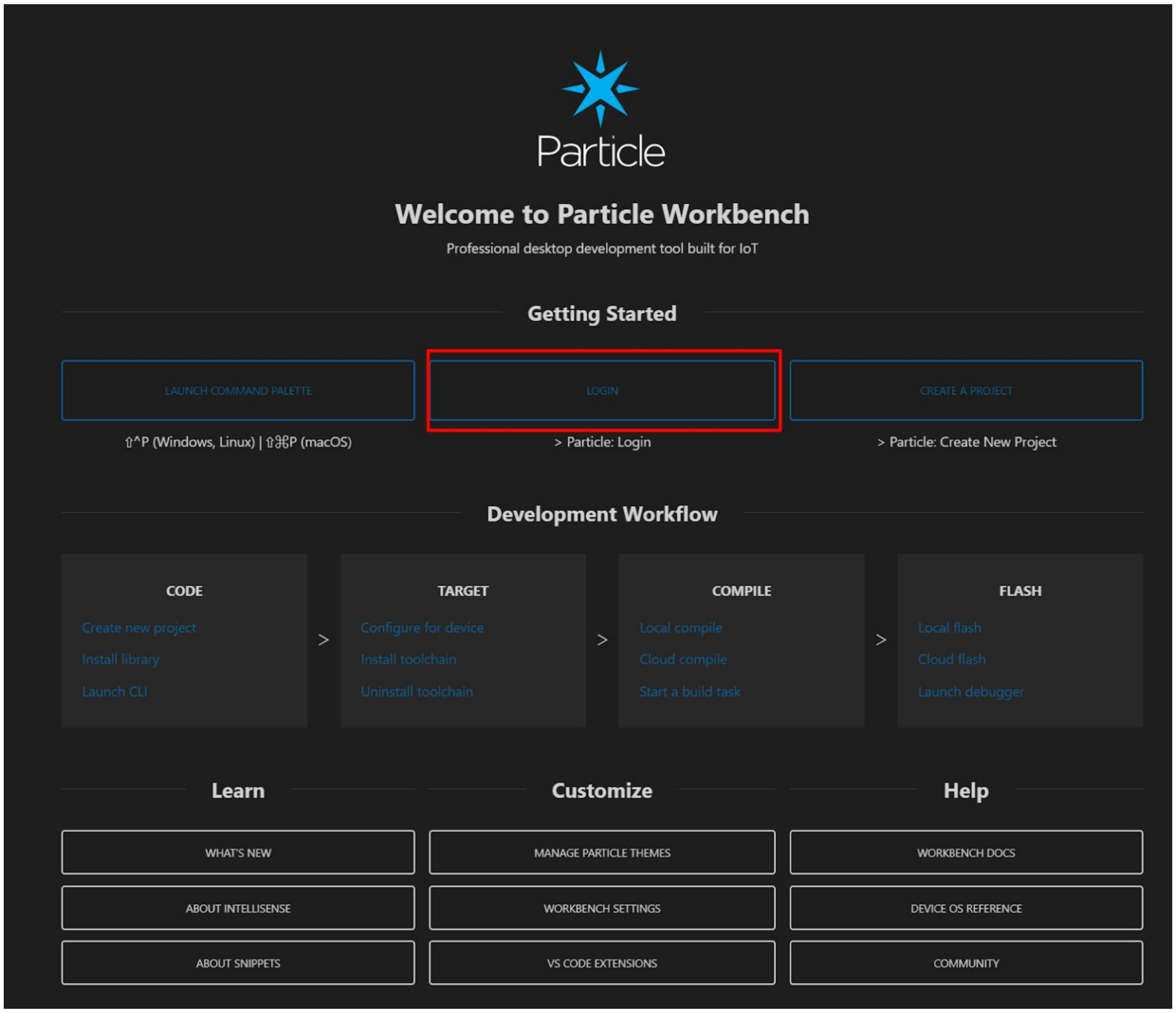
For the software, the Particle Workbench IDE will need to be downloaded with Visual Studio Code. The azure IOT add-on is not needed. This program can be run on ubuntu, Mac, and Windows.

VS Code is a popular and powerful source-code editor developed by Microsoft, and it has an open source license. It can be downloaded from here: https://code.visualstudio.com/

1.1 Particle workbench

In order to use the Particle board, you need to finish setting up the Particle workbench. It can be installed from here: https://www.particle.io/workbench/

Must Log-in after installing the workbench:

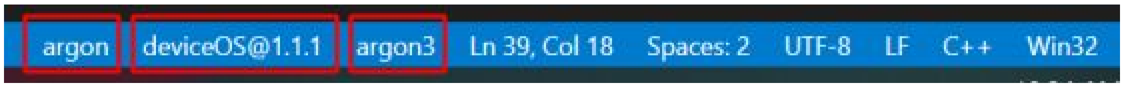


Username: airlab@clarkson.edu

Password: cuairlab@camp292

The Particle command line interface (CLI) can be opened by typing Ctrl+Shift+P. This will show a bar at the top of the screen. In the bar, the > means that you are typing in the Particle CLI. If the > is not there, you must type in the > before the command.

Begin by creating a project with the CLI command “Particle: Create New Project”. You must select a folder whose name has no spaces in its title or an error will appear and you will have to pick another folder for the project. Name the project “SD1”. For every project, you need to type “Particle: Configure Project for Device” in the Particle CLI. Next, type or select “deviceOS@1.1.0” to select the correct device OS. Select the target platform as “argon.” Type in the correct device ID according to your device; it will look like “argon##” where ## is your specific device number.



Check to make sure that the words boxed in red in the figure above are the same as your screen (except possibly the device ID number). If they aren’t the same, click on the one you want to choose and type in the correct information from the directions in the previous paragraph.

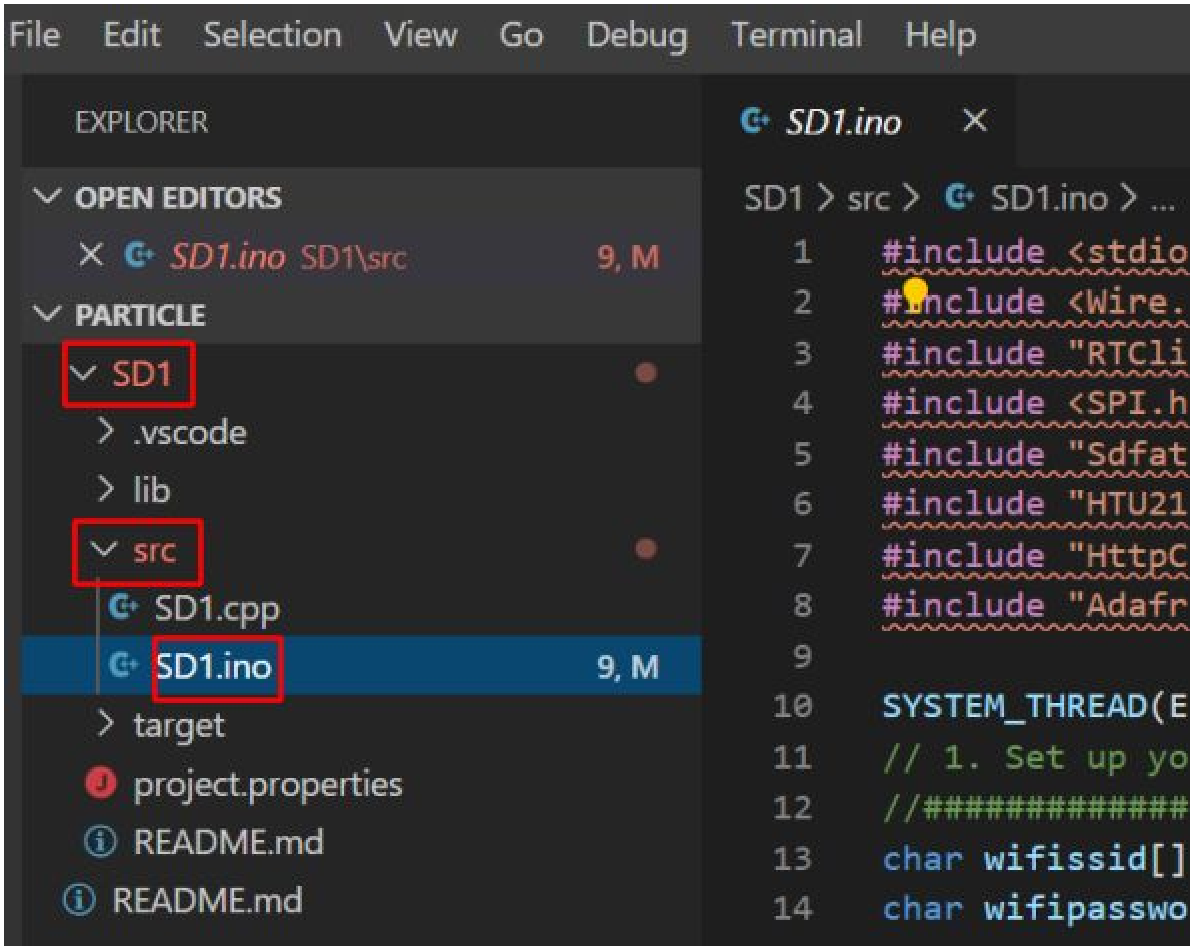
1.2 Downloading code

All the code files for this project are available at:

https://github.com/Potsdam-Sensors/Particle

1. On GitHub, go through this directory: SD1 // src // *SD1.ino*

2. Download SD1.ino and copy all text in the file by opening the file in Notepad and either highlighting it all or pressing Ctrl+A then Right click // Copy.
3. Click on the *src* folder from the left toolbar and open your *.ino* file



```
#include <stdio
#include <Wire.
#include "RTCli
#include <SPI.h
#include "Sdfat
#include "HTU21
#include "HttpC
#include "Adafr

SYSTEM_THREAD(E
// 1. Set up yo
//#############
char wifissid[]
char wifipasswo
```

4. Delete everything in the file and paste the code you copied from *SD1.ino*

Once done with this, you can start writing the program for the device. To install a library, you need use the Particle CLI to type “Particle: Install Library”. In the command line, type one library at a time exactly as they appear below:

- RTClibrary
- SdFat
- HTU21D
- HttpClient
- Adafruit_Si7021

After installing the needed libraries, you can start creating the code for the project. The Particle IDE uses C code to program the board. The main file is found as *projectname.ino*. The setup() function runs once then the loop() function runs and loops repeatedly. If you have experience with Arduino, this is exactly how the Particle program is structured. The first thing to do with the program is add the #include files at the top of the program. These should be added before the setup function. The #include files should be written like this: #include <filename.h>. A list of the filenames can be seen below:

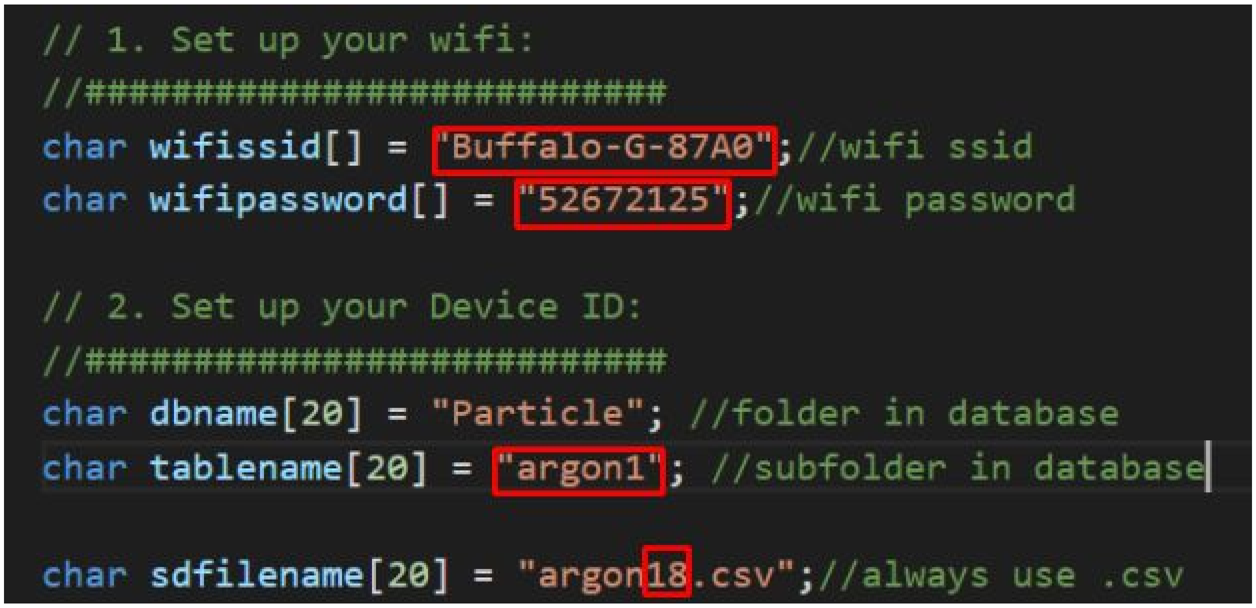
- stdio
- Wire
- RTClibrary
- SPI
- SdFat
- HTU21D
- HttpClient
- Adafruit_Si7021

1.3 Setting up Wi-Fi

After adding the libraries, it is good practice to type this command on the line after the #include files:

SYSTEM_THREAD(ENABLED);

The next step to do if you were using Wi-Fi is to put in the Wi-Fi credentials. This can be done by editing the *wifissid* (name of Wi-Fi) and *wifipassword* (password) as shown below:

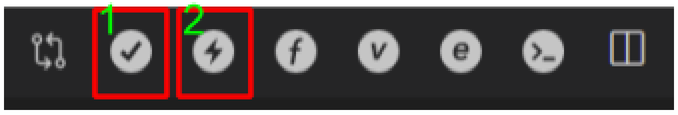


```
// 1. Set up your wifi:
//#############################
char wifissid[] = "Buffalo-G-87A0";//wifi ssid
char wifipassword[] = "52672125";//wifi password

// 2. Set up your Device ID:
//#############################
char dbname[20] = "Particle"; //folder in database
char tablename[20] = "argon1"; //subfolder in database

char sdfilename[20] = "argon18.csv";//always use .csv
```

Also, do not forget to change the Device ID (*tablename*) to the correct number so the data can go to the correct folder in the database.

The *sdfilename* should be edited so the file can be found easily when downloading data from the SD card.

Compile and flash the code to memory (check mark and lightning bolt in the top-right toolbar)



There may be some warnings after compiling related to the libraries, but that is normal.

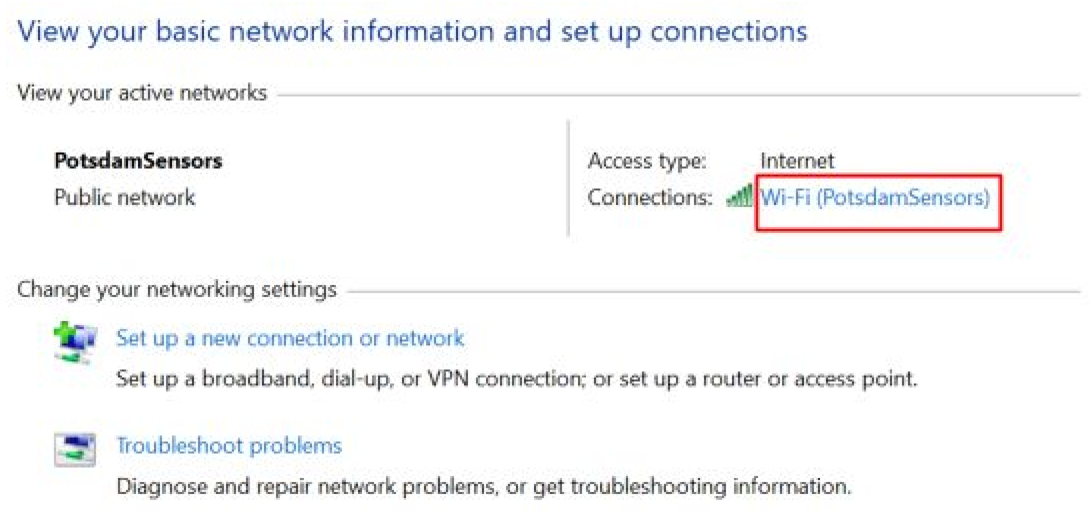
If this toolbar isn't there, open the Particle CLI and type "View: Show Particle Workbench" and under Development Workflow then Compile, select Local Compile. Next, under Flash, select Local Flash.

The code is now on the Particle Argon and ready to plug into the IPMC! See below for how to solve issues related to Wi-Fi connection and the SD card timestamp.

1.4 Solving Wi-Fi connection issues

If there are issues connecting to Wi-Fi, go to the *void setup()* function where the *WiFi.setCredentials* function is. The third parameter (the security type, usually *WPA2*) could be different for the Wi-Fi you are trying to connect to. To see which security type your Wi-Fi uses, use the following steps:

1. On a Windows computer connected to your Wi-Fi, press the Windows key, search "Control Panel" and click on the program.
2. Click on "Network and Internet"
3. Click on "Network and Sharing Center"
4. You should see your Wi-Fi as connected and click on the hyperlink "Wi-Fi (network_name)"


View your basic network information and set up connections

View your active networks

PotsdamSensors
Public network

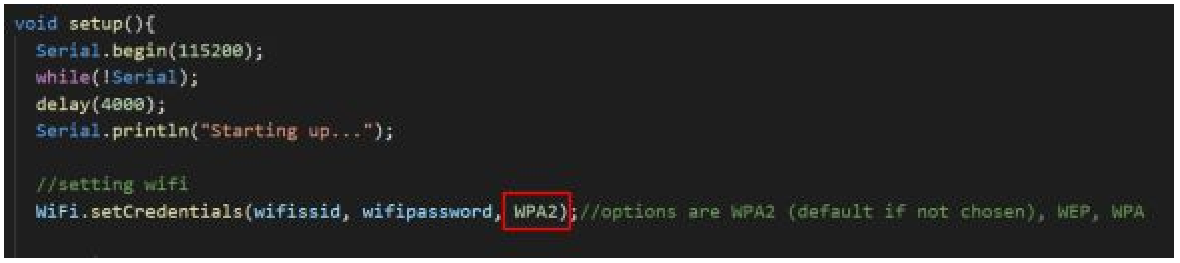
Access type: Internet
Connections: Wi-Fi (PotsdamSensors)

Change your networking settings

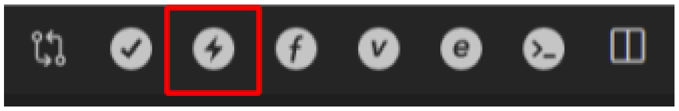
Set up a new connection or network
Set up a broadband, dial-up, or VPN connection; or set up a router or access point.

Troubleshoot problems
Diagnose and repair network problems, or get troubleshooting information.

5. Click on "Wireless Properties" then go to the "Security" tab
6. The Security type will show you some version of WPA2, WEP or WPA
    a. Note: do not change the Security type from drop down
7. Go back to the *void setup()* function and enter either WPA2, WEP or WPA in the third parameter in the *WiFi.setCredentials* function as seen below:


```
void setup(){
  Serial.begin(115200);
  while(!Serial);
  delay(4000);
  Serial.println("Starting up...");

  //setting wifi
  WiFi.setCredentials(wifissid, wifipassword, WPA2);//options are WPA2 (default if not chosen), WEP, WPA
```

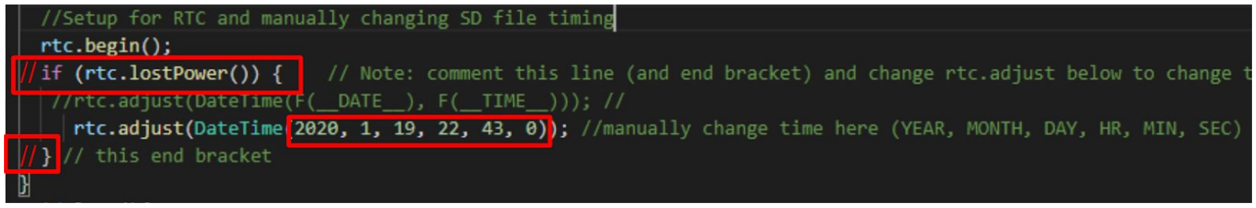
8. Flash the code to memory (lightning bolt in the top-right toolbar)



1.5 Resetting SD card timestamp issues

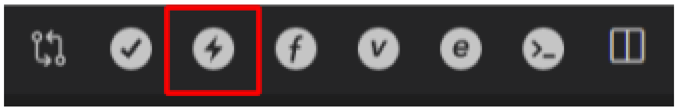
If you notice that the time/date in your SD file is incorrect, this can be solved by following the steps below:

1. At the end of the *void setup()* function there will be a section as shown below:



```
  //Setup for RTC and manually changing SD file timing
  rtc.begin();
//if (rtc.lostPower()) {    // Note: comment this line (and end bracket) and change rtc.adjust below to change t
   //rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //
    rtc.adjust(DateTime(2020, 1, 19, 22, 43, 0)); //manually change time here (YEAR, MONTH, DAY, HR, MIN, SEC)
//} // this end bracket
}
```

2. Comment out (entering "//") before the *if(rtc.lostPower()* statement as well as the first bracket ("}") on the line below the *rtc.adjust(* statement
3. Adjust the time in the *DateTime(* function according to (YEAR, MONTH, DAY, HR, MIN, SEC)
   a. Note: Set enough time for this to be flashed to the Particle board's memory, plugged back into the IPMC board and then plug in power ~10 seconds before that time.
4. Flash the code to memory (lightning bolt in the top-right toolbar)



5. Leave power on for about 30 seconds, then unplug
6. Uncomment out (delete "//") on the lines mentioned in step 2
7. Re-flash the code to your device
8. If possible, check files on SD card for timing accuracy

2. Database

2.1 Device ID used in this project

Device Name	Sensor	Board
argon6	Plantower	Argon (wifi)
argon15	Plantower	Argon (wifi)
argon16	Plantower	Argon (wifi)
argon17	Plantower	Argon (wifi)
argon21	Plantower	Argon (wifi)
argon23	Plantower	Argon (wifi)
argon27	Plantower	Argon (wifi)
argon33	Plantower	Argon (wifi)
argon36	Plantower	Argon (wifi)
argon38	Plantower	Argon (wifi)

2.2 Database info

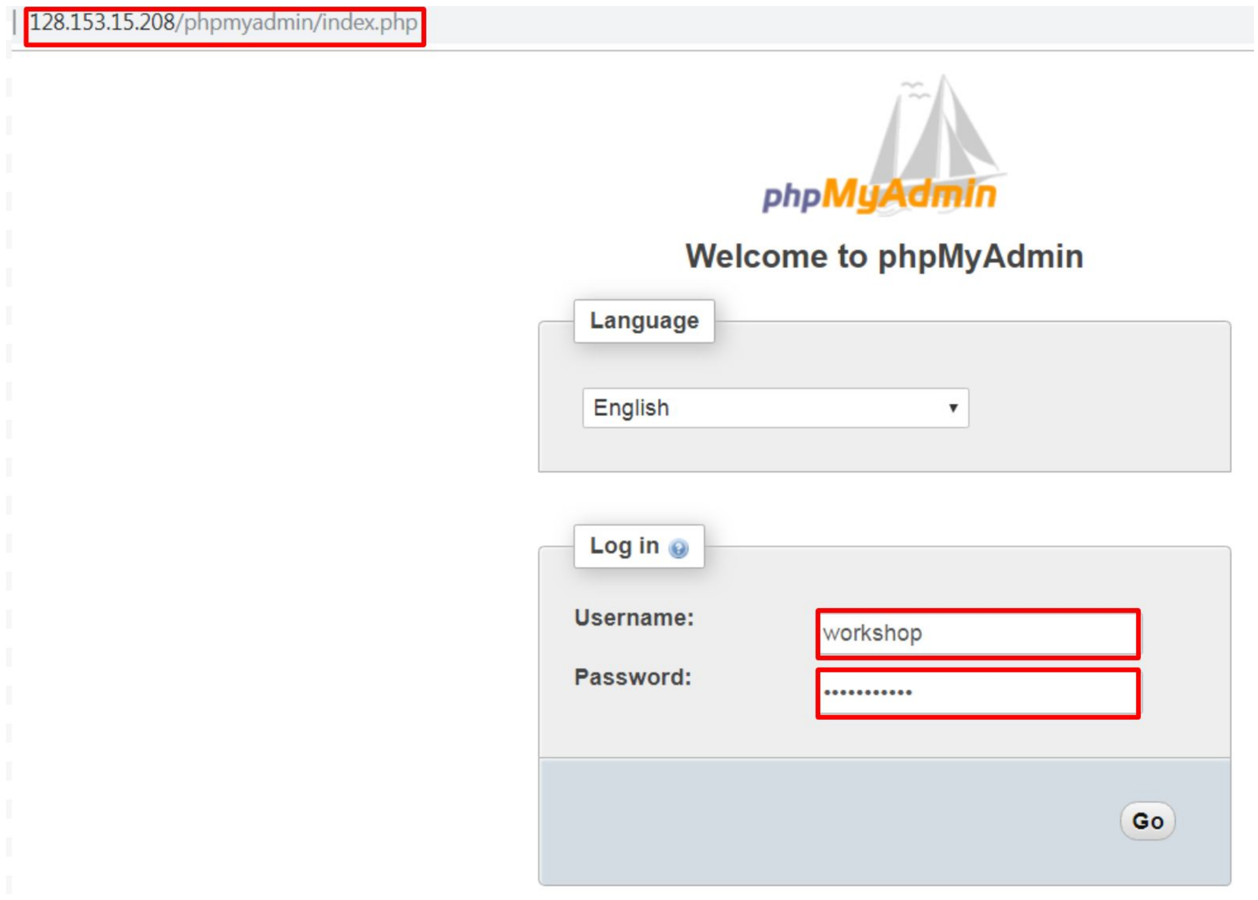
1. Go to the server login page in your web browser.
[Server Log in]

Mysql server login address:128.153.15.208/phpmyadmin

Log in to the server:

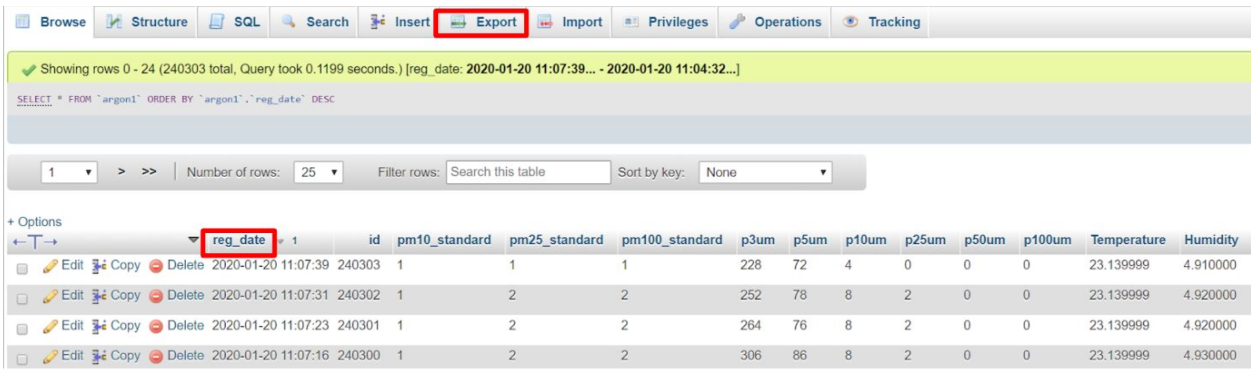
Username: workshop

Password: sensors2019


128.153.15.208/phpmyadmin/index.php

2. Select the database and table

At the left side of the page, you can see the database selection options, click on the "+" next to "Particle" then click on the device that corresponds to your argon number (will look like "argon##"). You will see the table show up with data, the header on the table already includes the information of each column. If you click on the "reg_date", you will be able to sort the data by most recent time. If you wish to **export the data** click on the "Export" button .

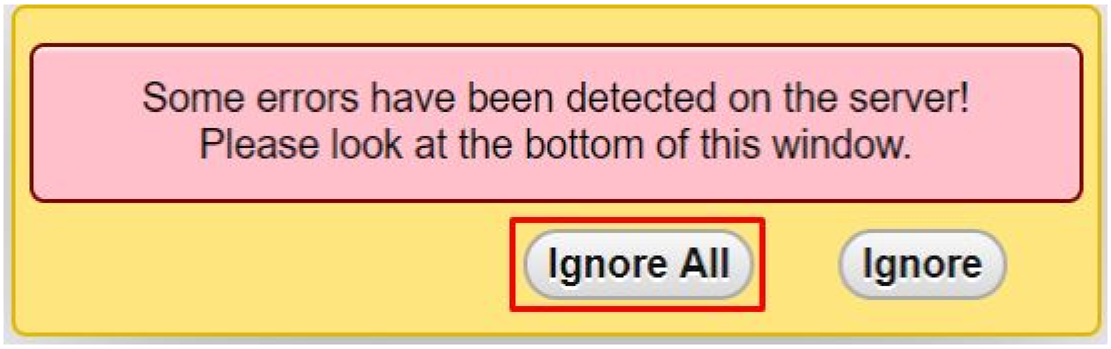


Showing rows 0 - 24 (240303 total, Query took 0.1199 seconds.) [reg_date: **2020-01-20 11:07:39... - 2020-01-20 11:04:32...**]

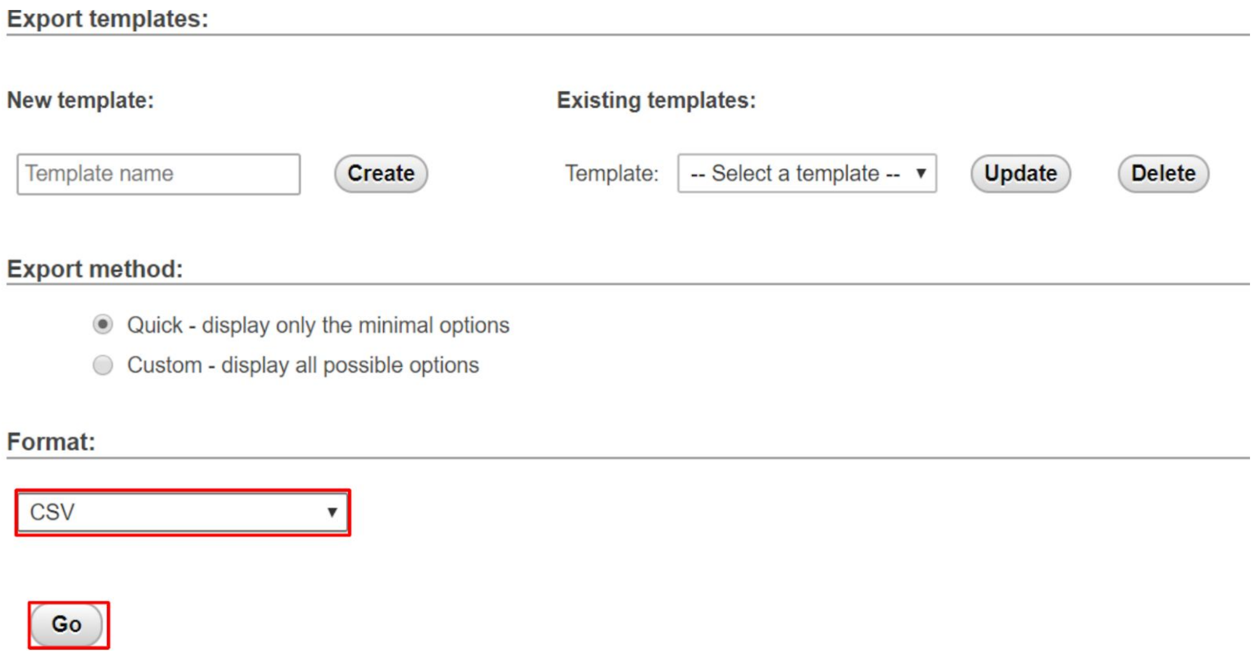
```
SELECT * FROM `argon1` ORDER BY `argon1`.`reg_date` DESC
```

	reg_date	id	pm10_standard	pm25_standard	pm100_standard	p3um	p5um	p10um	p25um	p50um	p100um	Temperature	Humidity
Edit Copy Delete	2020-01-20 11:07:39	240303	1	1	1	228	72	4	0	0	0	23.139999	4.910000
Edit Copy Delete	2020-01-20 11:07:31	240302	1	2	2	252	78	8	2	0	0	23.139999	4.920000
Edit Copy Delete	2020-01-20 11:07:23	240301	1	2	2	264	76	8	2	0	0	23.139999	4.920000
Edit Copy Delete	2020-01-20 11:07:16	240300	1	2	2	306	86	8	2	0	0	23.139999	4.930000

In the Export page, if there shows a warning, click on **"Ignore All"**


Some errors have been detected on the server!
Please look at the bottom of this window.

You can select 10 types of data under the “Format” drop-down menu (we recommend “CSV”), then click on “GO” to begin downloading the file.


Export templates:

New template:

Template name

Existing templates:

Template: -- Select a template --

Export method:

● Quick - display only the minimal options

○ Custom - display all possible options

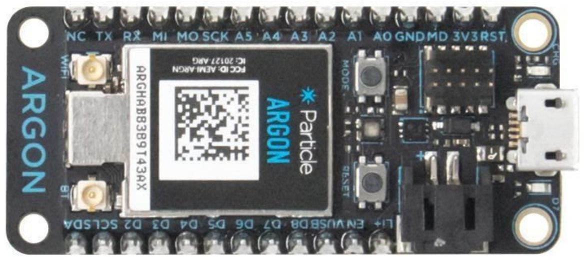
Format:

CSV

Appendix

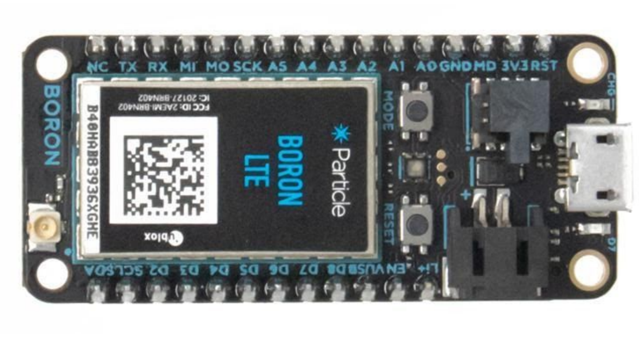
A.1 Microcontroller

Two types of Particle.io IOT development boards were used in this project: Argon and Boron. The Argon board uses Wi-Fi to transport data and Boron board uses Cellular to do so.



Argon Board

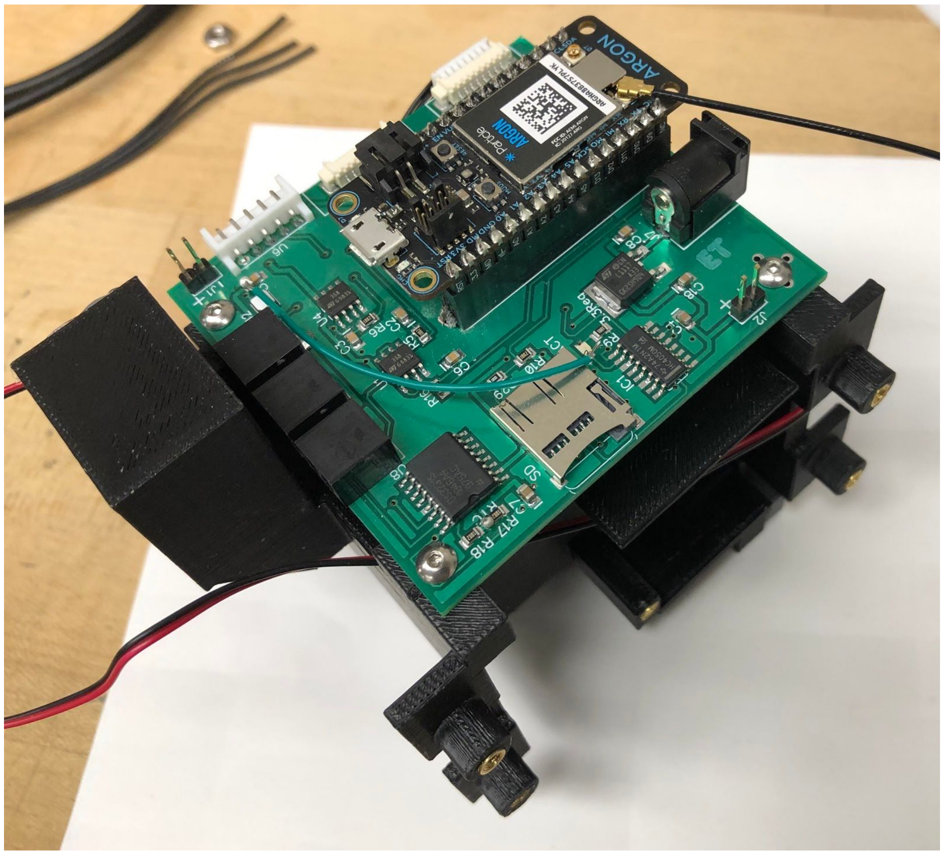
https://www.particle.io/wifi



Boron Board

https://www.particle.io/cellular

A.2 Circuit Board



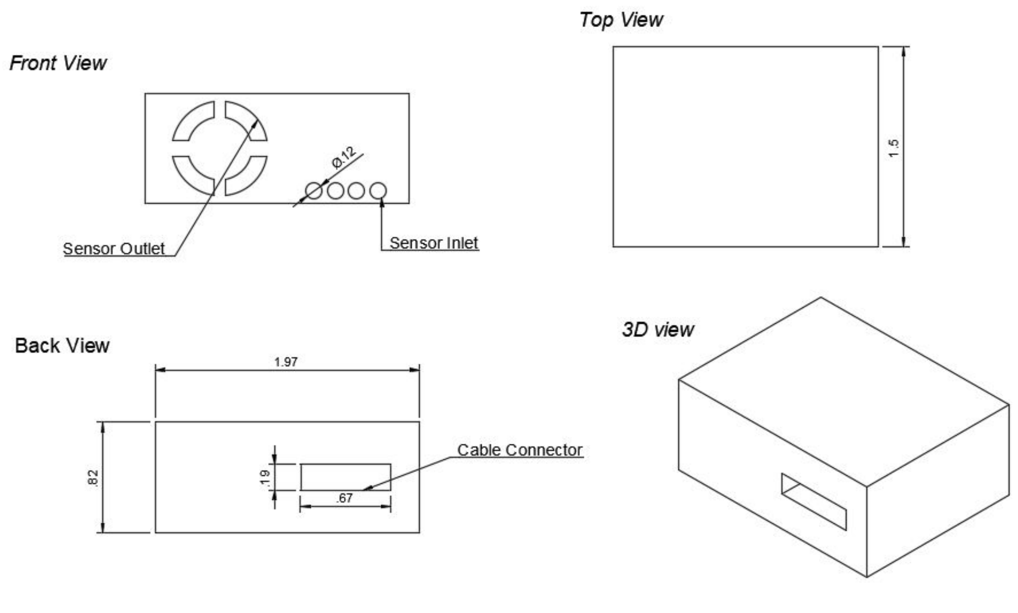
The circuit board has power supply system for each electronic components, it takes 5V DC power from 2.1 mm Power jack and drop the voltage to 3.3V.

The Particle.io board can be inserted on the board by pins soldered on it. PM sensor connected to the system by a 1.25mm molex header.
Circuit board was installed on a 3d printed holder by 4*40 screws.

A.3 Sensors and electronic components

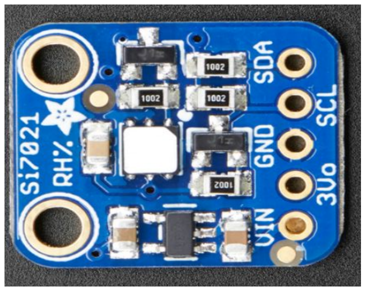
A.3.1 PM Sensor

Plantower PMS5003 was used in this system, PMS 5003, developed by Plantower, outputs six channels of particle number concentration data, representing 6 particle sizes: 0.3μm, 0.5μm, 1.0μm, 2.5μm, 5μm and 10μm, and three channels of mass concentration data: PM1.0, PM2.5 and PM10. The sensor uses a red laser light source and the illumination angle is 90 degrees. The light scattering signals get converted to particle concentration data by a built-in microprocessor. As shown in the figure below, a fan is built into the sensor to direct air flow through the sensor's detection chamber and be illuminated by the laser.



A.3.2 Temp/Humidity Sensor

Si7021



This temp/RH sensor used in this project has ± 3% relative humidity measurements with a range of 0–80% RH, and ±0.4°C temperature accuracy at a range of -10 to +85°C. Great for all of your environmental sensing projects. It uses I2C for data transfer so it works with a wide range of microcontrollers.

A.3.3 SD Card

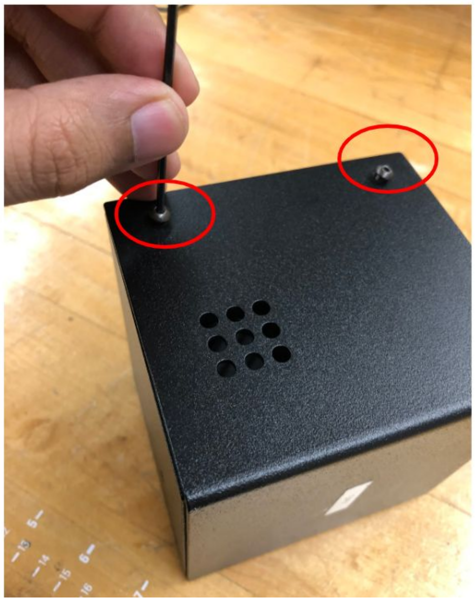
One mini SD card (8 GB) was installed on the circuit board, the file from that SD card is in the CSV format.

A.3.4 Real-Time Clock

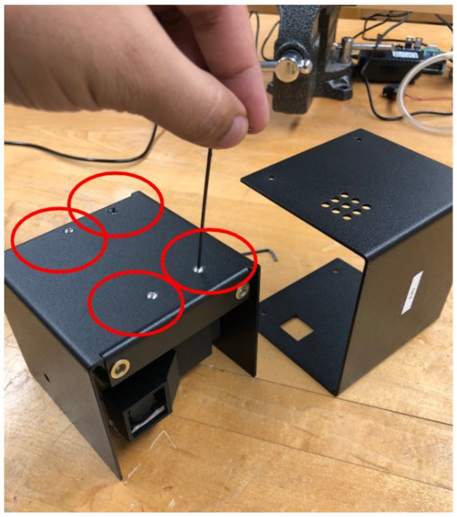
Adafruit DS3231 precision real time clock is used to record the time and add a timestamp to the data, and this unit is powered with a 5 V Power supply and also communicates with Argon/Boron through the I2C port

A.4 Assembling

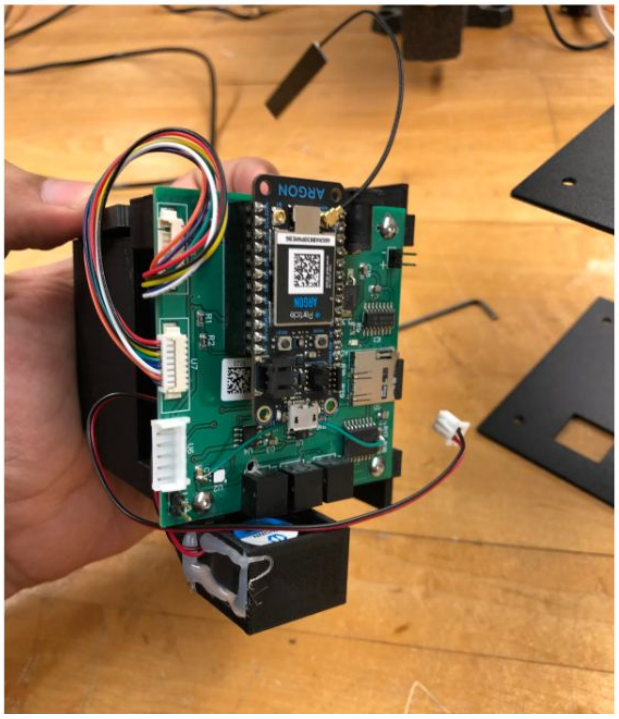
This unit also has a 3D-printed mounting system that was used to keep all the sensors stable and ventilated. An aluminum housing was used to cover the entire unit. The aluminum unit has holes where flow is needed and where openings are needed to input power.



For access to the argon board or SD card, the unit has to be opened. First, unscrew the side screws on the side of the enclosure.



Next, unscrew the screws located at the bottom of the unit.



After unscrewing the bottom screws, the s3 printed part can be accessed.

Note: the particle board must be removed from circuit board before uploading any code. DO NOT POWER IPMC BOARD AND UPLOAD CODE AT THE SAME TIME.