

SGP30 Project Instructions

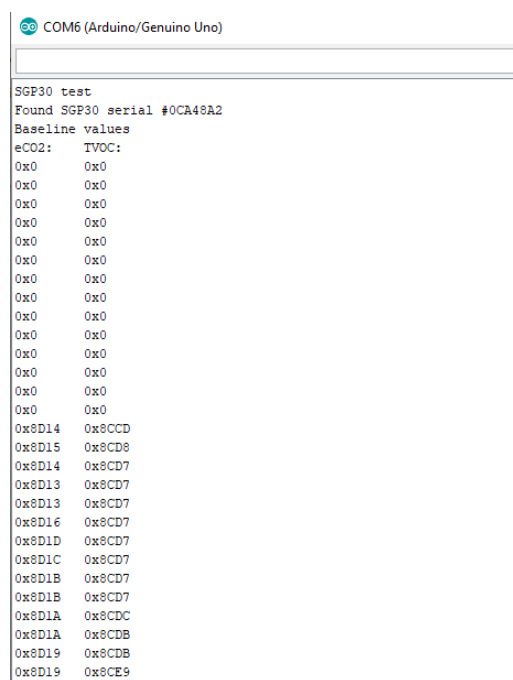
Fabiola Marin

March 2020

1 Documents

1.1 Baseline

In order to use the sensor you have to let it run for 12 hours so you can obtain the baseline values, this have to be done for each specific place. When you run the Baseline_arduino.ino code and open the monitor seria you will see, fig 1.



```
COM6 (Arduino/Genuino Uno)

SGP30 test
Found SGP30 serial #0CA48A2
Baseline values
eCO2:    TVOC:
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x0      0x0
0x8D14   0x8CCD
0x8D15   0x8CD8
0x8D14   0x8CD7
0x8D13   0x8CD7
0x8D13   0x8CD7
0x8D16   0x8CD7
0x8D1D   0x8CD7
0x8D1C   0x8CD7
0x8D1B   0x8CD7
0x8D1B   0x8CD7
0x8D1A   0x8CDC
0x8D1A   0x8CDB
0x8D19   0x8CDB
0x8D19   0x8CE9
```

Figure 1: Caption

1.2 Data

Once you have the baseline values you can run the TVOC_arduino.ino code, first you have to set the values that you obtained before, fig. 2, this value will be updated every hour, fig. 3. You shouldn't power off the sensor, see datasheet.

```
// Set baseline values obtained before|
sgp.setIAQBaseline(0x96B1, 0x93CA);

void loop() {
  time = millis();
  // If you have a temperature / humidity sensor,
  //you can set the absolute humidity to enable the humidity compensation for the air quality signals
  float temperature = 23; // [°C]
  float humidity = 26; //[%RH]
  sgp.setHumidity(getAbsoluteHumidity(temperature, humidity));

  if (! sgp.IAQmeasure()) {
    Serial.println("Measurement failed");
    return;
  }
  Serial.print(time); Serial.print(" \t ");
  Serial.print(sgp.TVOC); Serial.print(" \t \t ");
  Serial.println(sgp.eCO2);
}
```

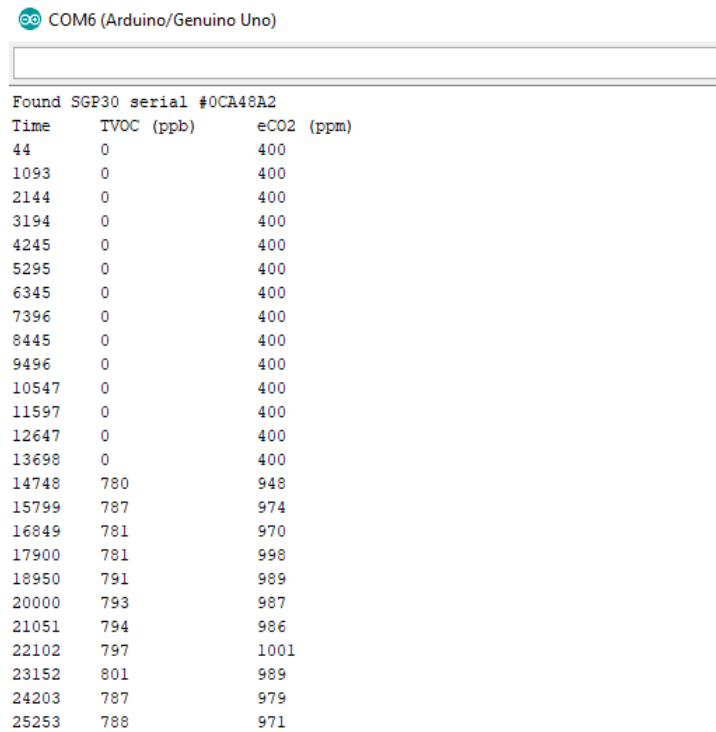
Figure 2: Baseline, temperature, and relative humidity values set

```
current_time = time / (1000/0.000277777);
if (last_time != current_time ) {
  last_time = current_time;
  uint16_t TVOC_base, eCO2_base;
  if (! sgp.getIAQBaseline(&eCO2_base, &TVOC_base)) {
    Serial.println("Failed to get baseline readings");
    return;
  }

  sgp.setIAQBaseline(eCO2_base, TVOC_base ); //Sets the baseline value every
```

Figure 3: Baseline update

Once you run the monitor serial you will see something like in fig 4.



Time	TVOC (ppb)	eCO2 (ppm)
44	0	400
1093	0	400
2144	0	400
3194	0	400
4245	0	400
5295	0	400
6345	0	400
7396	0	400
8445	0	400
9496	0	400
10547	0	400
11597	0	400
12647	0	400
13698	0	400
14748	780	948
15799	787	974
16849	781	970
17900	781	998
18950	791	989
20000	793	987
21051	794	986
22102	797	1001
23152	801	989
24203	787	979
25253	788	971

Figure 4: Monitor serial, data

1.3 CSV file

The Arduino doesn't allow to save the data that comes from the monitor serial, you have to either copy-paste or to use another program to save all the port information. I wrote a python program that takes all the information and saves it in a csv file.

1.3.1 Baseline

If you run the CSV_baseline python code you will get fig XXX, this is all the information that comes through the port, once you clean the file you should get

1.3.2 Data

You have to run the CSV_data, you will get ,and once you clean the file you will have .

Change the names of the file every time you want to create a new file of data, otherwise you will write always on the same file.