

The weather station project (Level 1)

Worksheet for students

Team:.....

Aim: Measure the temperature, pressure and humidity values

Where the measurement of temperature, pressure and humidity can be important? Search for information online and write your answers below.

How the temperature and pressure can be measured? Search for information online and write your answers below.

Review several scenarios, how weather station can be build, and choose one to demonstrate. Sketch your weather station and list the crafting materials that you may need.

Area for sketches

Bill of materials

Time for circuit making!

During this project, the sensors with grove standard will be used. This standard allows to connect elements easily. This is really important when sensors are sensitive to damage.

First, connect the Arduino UNO/Mega 2560 to *Grove Base Shield*/*Grove Mega Shield* as is shown in Fig. 1. This shield allows to connect elements in grove standard to Arduino board.

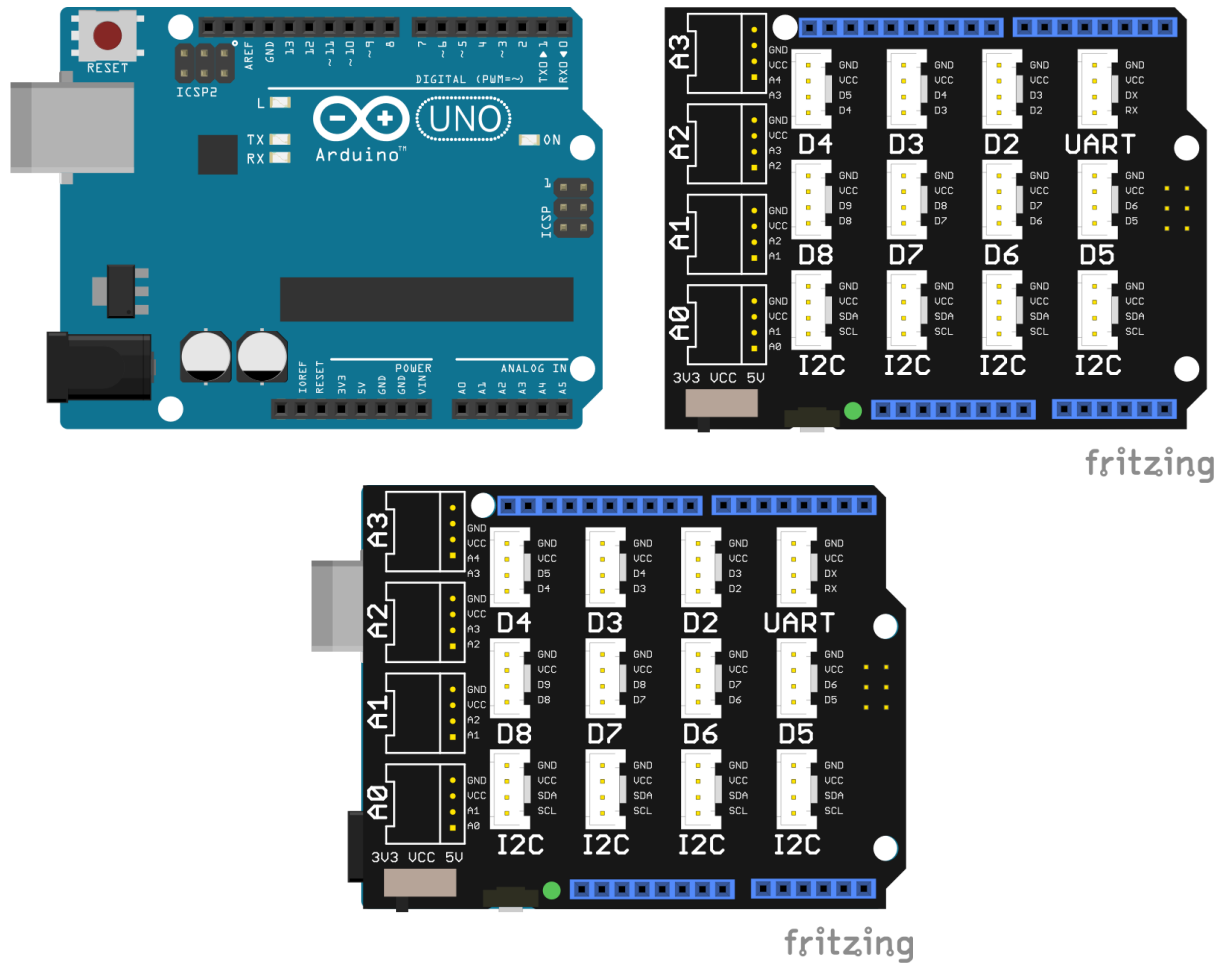


Figure 1: Top figure: Arduino Uno board and Grove Base Shield separately. Bottom figure: Grove Base Shield connected with Arduino Uno board.

Second, the BME280 sensor should be connected to the shield as is shown in Fig. 2. This sensor module consists of temperature, humidity and pressure sensors.

Time for hands-on practice!

Let's create the circuit using your Arduino board and the corresponding electrical components.

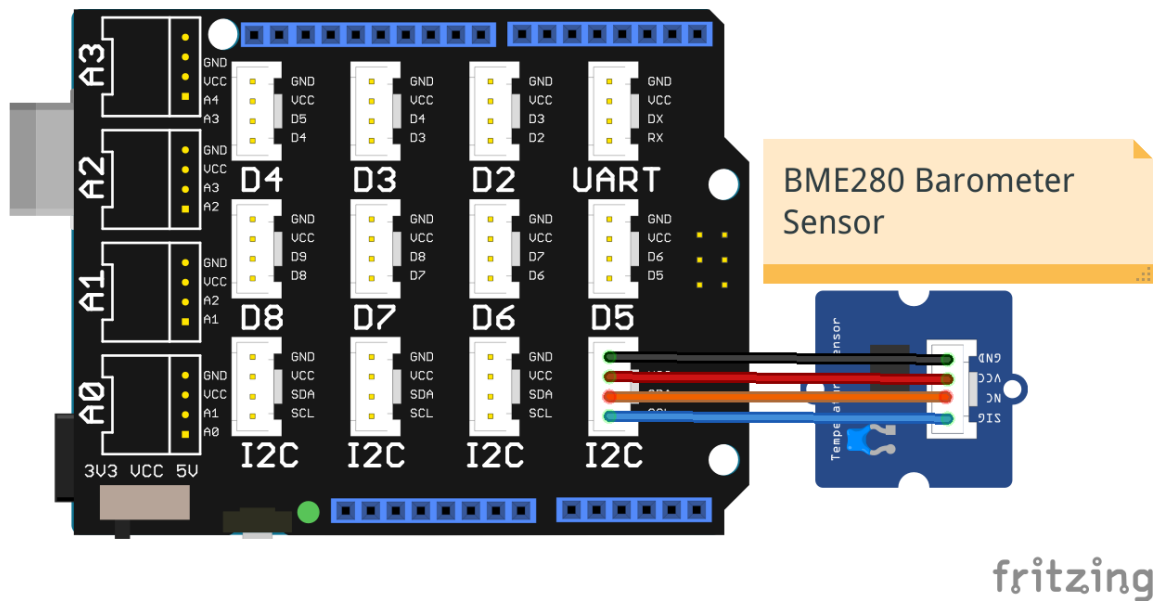


Figure 2: The BME280 sensor module connected to the Grove shield.

Time for programming!

Connect your Arduino to USB and open Arduino IDE software.

Most sensors have already prepared dedicated libraries, which can be easily downloaded from manufacturer's website or repository (e.g. github). The BME280 has such library available on page: https://github.com/Seeed-Studio/Grove_BME280 . If you do not have installed library, download library and save to *libraries* folder inside of Arduino IDE folder. Than, unzip library and run again Arduino IDE.

Open the example from library by choosing *File* → *Examples* → *Grove - Barometer Sensor BME280* → *bme280_example*. All programs consist of two functions: *setup* and *loop*. First function runs only one. Therefore, all commands inside *setup* function will execute only once. The *loop* function is running continuously. Therefore, all commands inside the *loop* function will be repeated until Arduino board is powered off. Try to analyze and understand the example.

Write your own code based on the example, which will read temperature, pressure and humidity and display their values on serial monitor. Use tabulator ("`\t`") as a separator. When you finish, open the serial monitor and look on values. Do they look reasonably?

Useful functions:

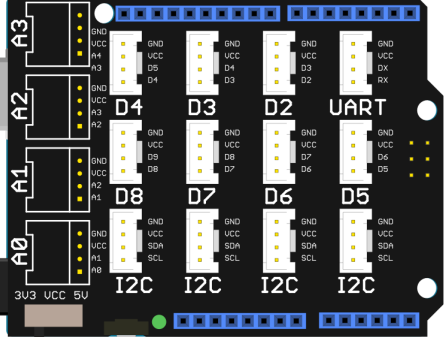
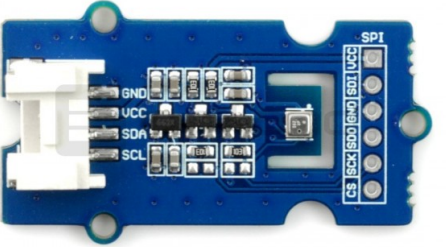
- `#include name_of_library` - this line allows to add library to the script,
- `Serial.begin(baud rate)` - initialization of Serial monitor with chosen baud rate,
- `Class_name object_name` (e.g. `BME280 bme280`) - create object of class, in which special function can be called,
- `init` - function of BME280 class, which initializes and configures the sensor,
- `print` - function of Serial class, which print values or strings in the serial monitor,
- `println` - function of Serial class, which print values or strings in the serial monitor and add a new line character at the end of line,
- `getTemperature` - function of BME280 class, which read and return temperature value from BME280 sensor,
- `getPressure` - function of BME280 class, which read and return pressure value from BME280 sensor,
- `getHumidity` - function of BME280 class, which read and return humidity value from BME280 sensor,
- `delay(time_in_ms)` - function, which stop execution of script for chosen period specified in milliseconds.

Time for crafting!

Start working on the design of the weather station using the available crafting materials; bear in mind that your model will be revised several times and adjustments/improvements will be introduced.

Electrical components

The following table is an index containing all the components that need to be implemented for accomplishing the present activity.

 <p>A Fritzing diagram of a Grove Base Shield. The board is black with white and blue components. It features four I2C headers (labeled I2C) at the bottom, each with pins for GND, VCC, SDA, and SCL. There are also four digital headers (labeled D4, D3, D2, D5) and one UART header. On the left side, there are three analog headers (labeled A0, A1, A2, A3) with pins for GND, VCC, and A0-A3. The top edge has a header with pins for GND, VCC, and RX. The bottom edge has a header with pins for 3V3, VCC, and 5V. The logo 'fritzing' is visible at the bottom right of the diagram.</p>	<p>Grove Base Shield (for Arduino UNO) or Grove Mega Shield (for Arduino Mega 2560)</p>
 <p>A photograph of a Grove Barometer Sensor (BME280) on a blue PCB. The board has a white I2C header on the left with pins for GND, VCC, SDA, and SCL. On the right, there is a white SPI header with pins for CS, MISO, GND, MOSI, and VCC. The sensor chip is visible in the center of the board.</p>	<p>Grove Barometer Sensor (BME280)</p>

ROBOSCIENTISTS PROJECT

Motivating secondary school students towards STEM careers through robotic artefact making

Erasmus+ KA2 2018-1PL01-KA201-051129

Creator

Angelika Tefelska (WUT)

Declaration

This report has been prepared in the context of the ROBOSCIENTISTS project. Where other published and unpublished source materials have been used, these have been acknowledged.

Copyright

© Copyright 2018 - 2021 the Roboscientists Consortium
All rights reserved.



This document is licensed to the public under a Creative Commons Attribution- NonCommercial-ShareAlike 4.0 International License.

Funding Disclaimer

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.