

ÉMIS Communauté PM Sensor



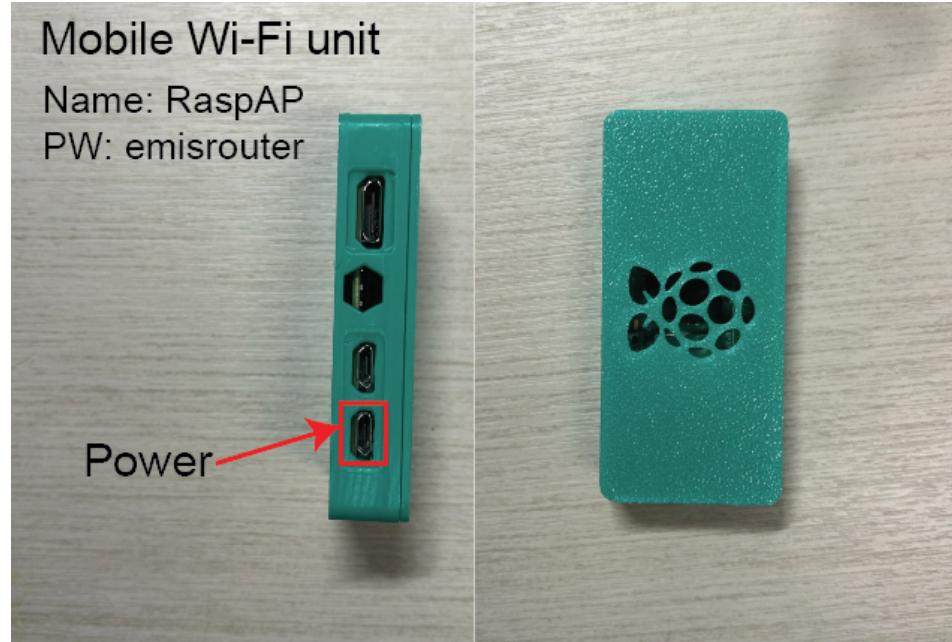
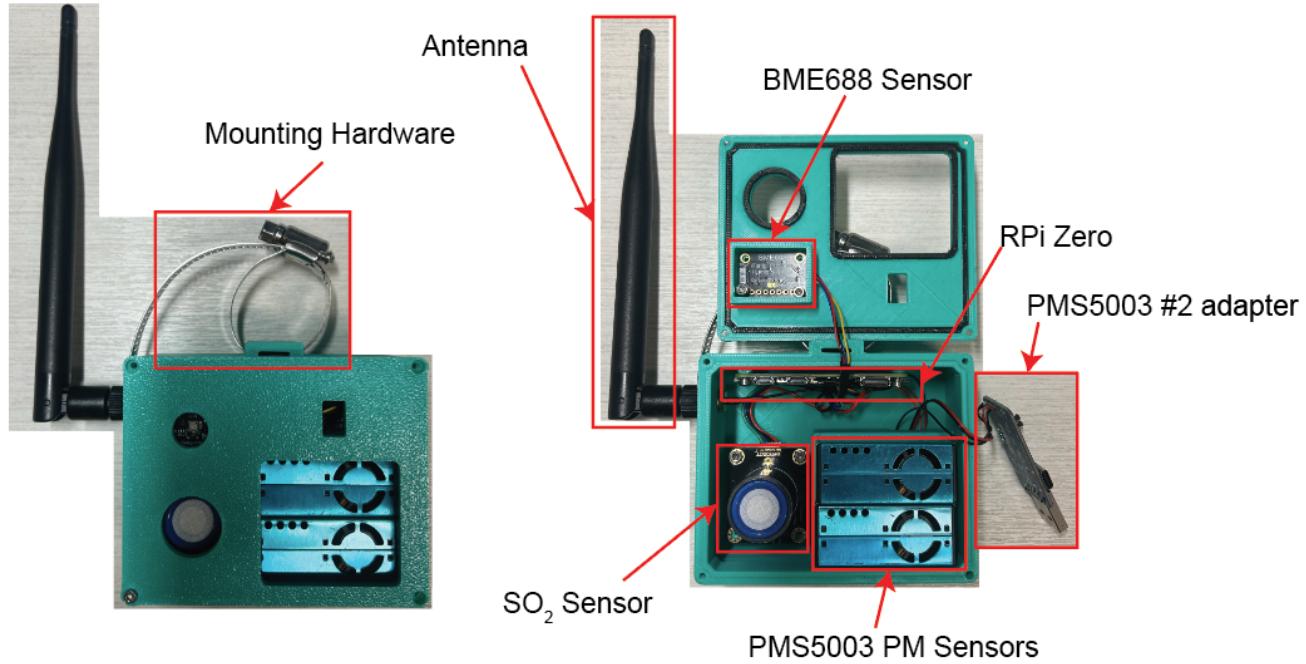
This document introduces the basic components of the Environmental Monitoring for Industrial Sites (ÉMIS) Communauté sensor and how to use it. Please contact emmet.norris@umontreal.ca if you have questions not answered by this document or are interested in modifications or other project ideas.

Table of contents

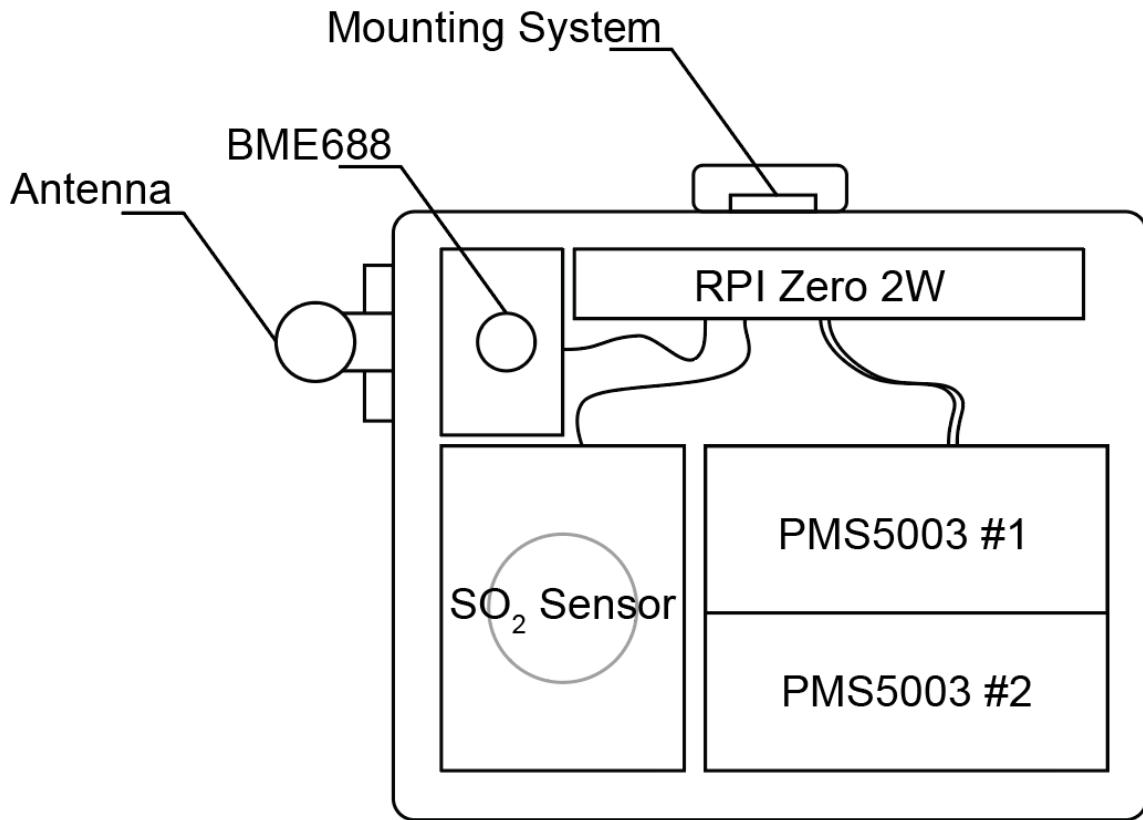
1. Sensor Components	2
1.1 Annotated unit images	2
1.2 Schematic diagram and function explanation	3
1.2.1 RPi Zero 2W	3
1.2.2 BME688	3
1.2.3 SO2 Sensor	4
1.2.4 PMS5003	4
1.2.5 Antenna	4
1.2.6 Mounting System	4
2. Software Structure	5
2.1 Overview	5
3. Installation and Initialization	5
3.1 Mounting	5
3.2 Powering	6
3.3 Connecting to Wi-Fi	6
4. Downloading Data	7
4.1 Via SSH	7
4.2 Other methods to download data	7

1. Sensor Components

1.1 Annotated unit images



1.2 Schematic diagram and functions



1.2.1 RPi Zero 2W

The Raspberry Pi (RPI) Zero 2 W is a compact, low-power single-board computer designed for embedded and portable applications. It runs a full Linux operating system and features a quad-core 64-bit processor, onboard Wi-Fi and Bluetooth, and a 40-pin GPIO header for connecting sensors and external hardware. The system is powered by **Raspberry Pi OS Bookworm**, using a **6.12.47 Linux kernel** tailored for the Pi Zero 2 W platform (armv7), released on **16 September 2025**. This is a ‘headless’ system, meaning there is no desktop environment, and is controlled entirely through the command line. We have modified the Wi-Fi unit to connect to an external antenna.

The Zero 2 W is well suited for continuous, low-power data collection while maintaining the flexibility of a general-purpose computer. It supports connection to Wi-Fi and Bluetooth, as well as optional peripherals such as a monitor (via mini-HDMI port) and keyboard or accessories (via micro-USB port). Read more about the [RPi Zero 2W here](#).

In this system, it serves as the brain, or primary sensor and data processing unit for the sensor.

1.2.2 BME688

The Adafruit BME688 is a multi-sensor unit that measures temperature, humidity, pressure and gas (VOCs). It was chosen because of its reliability, previous use in similar devices, and because it can be

connected to the RPi via SPI or I2C. In this application we are connecting via I2C using one of the 4 pin STEMMA QT connectors available. Read more about the sensor specifications and manual [online here](#).

1.2.3 SO₂ Sensor

We currently use a Gravity factory calibrated sulfur dioxide (SO₂) sensor from DFRobot. It is attached to a signal conversion board to take raw analog outputs and convert them to I2C, which is how we connect to it. It has a detection range of 0~20ppm in a working temperature of -20~50°C. The lifespan is ~2 years. Read more about the sensor [online here](#).

1.2.4 PMS5003

The PMS5003 by Plantower is a compact optical particulate matter (PM) sensor widely used in portable and outdoor monitoring applications. It measures airborne particles using a laser-scattering method and reports size-resolved concentrations for PM1.0, PM2.5, and PM10 in real time. The sensor includes an internal fan and air channel to draw air through the sensing chamber, ensuring consistent sampling. It operates at low power, communicates via a simple serial interface, and provides reliable particulate measurements suitable for long-term ambient monitoring in a small, robust package. Read more about the sensor [online here](#).

In the ÉMIS, we include two PMS5003 sensors in parallel to guard against potential unit failure and to increase confidence in measurements by averaging their outputs.

1.2.5 Antenna

We have modified the onboard Wi-Fi antenna on the RPi Zero 2 W so that it connects to an external antenna. This provides increased wireless range and improves signal strength when the device is installed outdoors or far from a router.

The device uses a simple whip antenna tuned for the 2.4–2.5 GHz frequency band, which is compatible with most standard home and mobile Wi-Fi networks. Because the antenna only supports 2.4 GHz, **do not connect the device to a 5 GHz Wi-Fi network.**

1.2.6 Mounting System

The mounting system is composed of a built-in sleeve with vertical and horizontal openings and metal hardware. This allows for the device to be attached to different materials, including flat objects, such as walls or wooden posts, or cylindrical poles. Read section 3.1 for a more detailed explanation of how to mount the device.

2. Software Structure

2.1 Overview

The device is running a simple linux system, that contains several python codes (.py) to read the sensor data, process and store it. The file structure is:

```

emis/
└── code/
    ├── collect_data.py      # Main loop: reads sensors + PMS agreement + writes daily CSV
    ├── daily_writer.py     # DailyWriter + COLUMNS (CSV header/order)
    └── sensor_status.py   # status/diagnostic CLI

    └── sensors/
        ├── bme.py           # BME688 reader (temp/rh/pressure/voc_ohm)
        ├── pms.py            # PMS5003 reader (serial frames)
        └── so2.py             # SO2 reader (DFRobot Gravity, I2C)

    └── utils/
        └── timekeeping.py  # now_utc(), utc_to_local(), isoformat_* helpers

    └── config/
        └── node.yaml         # node_id, timezone, tick_seconds, sensor ports/bus/address

    └── data/
        └── daily/
            └── Node7_YYYY-MM-DD.csv # Daily CSV files

    └── logs/
        └── emis.log          # Runtime logs

    └── README.md           # Project overview / run instructions

```

The directory contains the main program that reads the sensor data output (`collect_data.py`) along with helper files that handle saving data.

Sensor-specific logic is kept in the `sensors/` folder (for example, files for the PMS particle sensors, BME688, OPC-N3, and SO₂), while shared helper functions such as time handling live in `utils/`. Device settings—such as the node ID, sampling interval, and which sensors are enabled—are defined in a single human-readable configuration file (`config/node.yaml`). Measurements from all sensors are combined and written automatically as separate daily CSV files to `data/daily/`. Runtime messages or errors are recorded in the `logs/` directory. This structure keeps configuration, code, and data clearly separated, making the system easy to understand, modify, and maintain.

3. Installation and Initialization

3.1 Mounting

The ÉMIS sensor should be installed well away from noticeable background particulate matter sources such as exhaust vents, common smoking areas, or construction sites (unless you are intentionally measuring those emissions). Place the unit where air flow is as natural as possible and not blocked or

redirected by buildings, fences, or trees. Ideally, the sensor should sit higher than nearby objects so it can sample air that reflects the surrounding environment. For most installations, mounting the unit on a pole at least 2 metres above the ground — and preferably above or on top of a building — is recommended.

Use the hose clamp (for mounting on a pole) or the aluminum bracket (for mounting on a flat surface) to secure the ÉMIS unit in place. If using the hose clamp, no additional hardware is required. Tighten the clamp around the pole using a socket tool, adjustable wrench, or flathead screwdriver. If using the aluminum bracket, you will need two (2) screws and a screwdriver to fasten the ÉMIS securely to the chosen flat surface.

3.2 Powering

To power the unit, **plug** the supplied power adapter into a standard electrical outlet (120 V AC). Small multi-colored LED lights will illuminate and the PMS5003 fans will start rotating when powered correctly.

If the outlet is located outdoors, ensure the adapter and its connections are protected from rain and moisture. You may need to seal exposed joints with tape or a weatherproof cover to prevent water from entering.

If the power supply is interrupted, the unit will automatically restart once power returns. However, for reliable long-term operation, choose an outlet or power source that is unlikely to be bumped, unplugged, or affected by repeated outages.

3.3 Connecting to Wi-Fi

The ÉMIS unit is pre-configured to connect to the mobile Wi-Fi hotspot provided (Figure X). This will enable a user to also connect to the Wi-Fi hotspot and access the RPi and data via SSH (see section 4.1).

You can change the Wi-Fi network using either of the following methods:

1. Via SSH access the RPi Zero 2W to access command line or
2. Connect a monitor and keyboard directly to the RPi Zero 2W

Method 1 (via SSH) is preferred, as it causes the least disruption to the deployed unit.

Once you have access to the RPi command line, run: `sudo raspi-config`. Then navigate to: **System Options → S1 Wireless LAN** and enter the new Wi-Fi network name and password.

Connecting to the mobile Wi-Fi unit

1. **Plug** the mobile Wi-Fi device into any USB power source.
2. **Wait** approximately 30 seconds for it to start up.
3. On your computer or tablet, look for the Wi-Fi network named “**RaspAP**”.
4. **Connect** using the password **emisrouter** (all lowercase).
5. Use your preferred **SSH** tool to connect to the RPi.

For **Node 7**, the default SSH IP address is: 10.3.141.131

4. Downloading Data

Sensor data is saved in the RPi folder `/home/emis/emis/data/daily` with files name in the format: `Node#_YYYY_MM_DD.csv`

Below are several methods to access and download the data. The RPi is password protected. To enter, when prompted the **username/login is: emis**, and the **password is: emis**.

4.1 Via SSH

To connect to the RPi and download data files the preferred method is to control the RPi remotely via a secure shell protocol (SSH) from a separate computer. To establish a SSH connection the second computer must be connected to the same Wi-Fi as the RPi (see section 3.3).

On Windows, a convenient option to SSH is using the program **WinSCP**, which provides a user-friendly interface to browse the RPi folders and download files directly to a computer. It is also possible to copy files using a command-line tool that supports SCP (e.g., PowerShell, Terminal, or Git Bash).

Via powershell on a windows computer typical command to download data looks like:

```
scp emis@10.3.141.131:/path/to/file path/on/your/computer
```

Replace `/path/to/file` with the file you want to copy and update the destination path accordingly.

4.2 Other methods to download data

It is also possible to retrieve data using alternative methods, such as removing the SD card and inserting it into a compatible computer, or by connecting a USB drive to the RPi Zero 2W and copying files through the command line. These options require additional steps and technical knowledge, so they are not the preferred methods—but they are available if needed.

A planned improvement for future versions is to partition the SD card so that data is stored in a file system readable by Windows and Mac. This would allow the user to remove the SD card, place it into a standard card reader, and copy files to the desired location directly without using SSH or command-line tools.

4.3 Data columns

Below are the columns names you will find the in the daily files:

timestamp_utc — Time the measurement was recorded, in *Coordinated Universal Time (UTC)*
timestamp_local — Same time converted to the *local timezone* where the sensor is deployed
node_id — Unique identifier for the ÉMIS sensor unit collecting the data

[Environmental Measurements \(BME688\)](#)**temp_c** — Ambient air temperature in *degrees Celsius (°C)***rh_pct** — Relative humidity in *percent (%)***pressure_hpa** — Atmospheric pressure in *hectopascals (hPa)***voc_ohm** — Volatile Organic Compound resistance value from the BME688 sensor in *ohms (Ω)***bme_status** — Operational status of the BME sensor[PMS5003 #1 Particulate Matter](#)**pm1_atm_pms1** — PM1.0 concentration measured by PMS #1 in $\mu\text{g}/\text{m}^3$ **pm25_atm_pms1** — PM2.5 concentration from PMS #1 in $\mu\text{g}/\text{m}^3$ **pm10_atm_pms1** — PM10 concentration from PMS #1 in $\mu\text{g}/\text{m}^3$ **pms1_status** — Status flag for PMS #1[PMS5003 #2 Particulate Matter](#)**pm1_atm_pms2** — PM1.0 concentration measured by PMS #2 in $\mu\text{g}/\text{m}^3$ **pm25_atm_pms2** — PM2.5 concentration from PMS #2 in $\mu\text{g}/\text{m}^3$ **pm10_atm_pms2** — PM10 concentration from PMS #2 in $\mu\text{g}/\text{m}^3$ **pms2_status** — Status flag for PMS #2[Derived PMS Pair Metrics](#)**pm25_pms_mean** — Average PM2.5 concentration from PMS #1 and PMS #2 ($\mu\text{g}/\text{m}^3$).

- Calculated only when both sensors are reporting valid values.

pm25_pms_rpd — Relative Percent Difference between PMS #1 and PMS #2 PM2.5 readings (%)**pm25_pair_flag** — Quality flag based on the agreement between PMS sensors.**pm25_suspect_sensor** — Indicator if a PMS sensor is unreliable based on pair comparison.[SPEC SO₂ Gas Sensor](#)**so2_ppm** — Sulfur dioxide concentration in *parts per million (ppm)***so2_raw** — Raw analog reading prior to calibration (unitless sensor output)**so2_byte0** — First data byte from the SO₂ sensor (*low-level communication value*)**so2_byte1** — Second data byte from the SO₂ sensor**so2_error** — Error indicator (0 = *OK*, nonzero may reflect bad reading)**so2_status** — General operational status for the SO₂ module (0 = *OK*)