

FEBRIS QUICK START GUIDE



Febris is a smart, wireless sensor series from Sentinum for monitoring indoor air quality and ambient conditions. Depending on the model, it measures CO₂, temperature, humidity, air pressure or surface temperatures. LEDs and a buzzer warn if limit values are exceeded. The data is transmitted energy-efficiently via MIOTY®, NB-IoT or LoRaWAN®. A double tap triggers an instant measurement. Powered by 4x AA batteries, easily replaceable. Wall mounting recommended, but can also be placed on tables or cabinets.

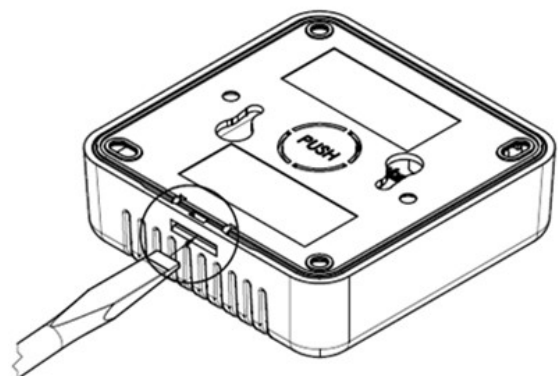
Please observe the warnings and instructions in the operating instructions to ensure the safety of you, your surroundings and the sensor!

ASSEMBLY AND INSTALLATION

Install the sensor on an interior wall at a height of between 1.50m and 1.80m. Ensure that the LEDs are on the lower right-hand side. Ensure that the sensor is at least 20 cm away from people and is located in an area below 2000 m above sea level. Use either magnets or suitable screws for mounting. The detailed steps can be found in the operating instructions.

COMMISSIONING

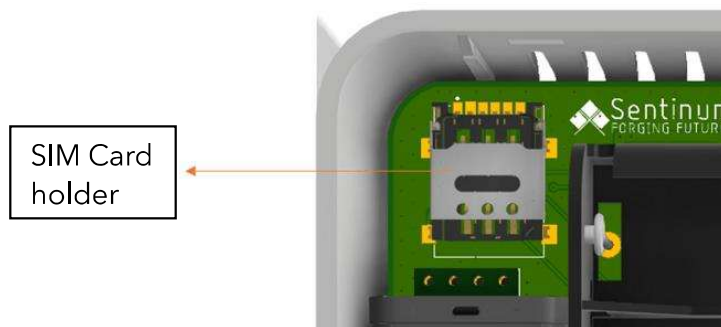
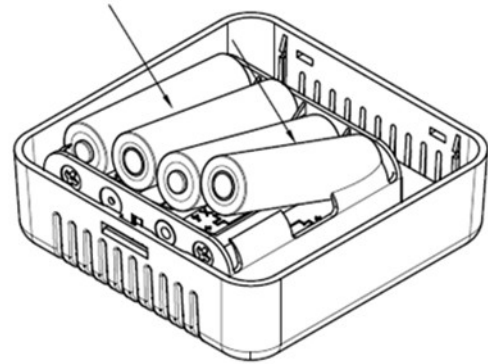
1. Open the sensor using the tabs provided (on the bottom edge in the picture). If necessary, carefully use a blunt object.



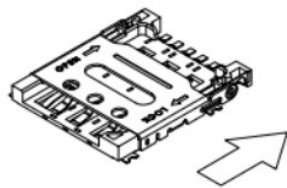
2. Now insert the cells. To achieve the specified runtimes and the performance specified in the data sheets, the following primary cells are permitted:

- Energizer® Ultimate Lithium Batteries - AA
- VARTA ULTRA LITHIUM Mignon AA

4 Cells are required per sensor.

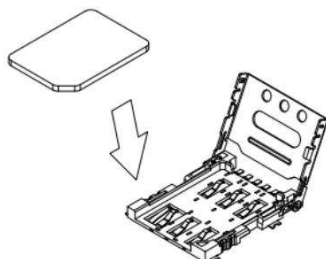
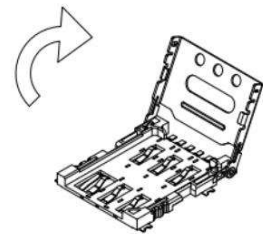


3. If you have an NB-IoT sensor and you want to insert your own SIM card, insert the SIM card now.



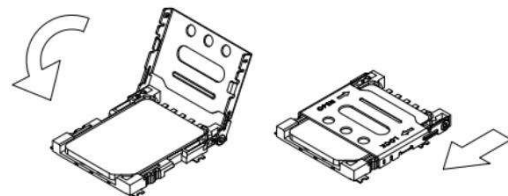
Unlock the metal lid by sliding it upwards.

Open it to insert the Nano SIM card



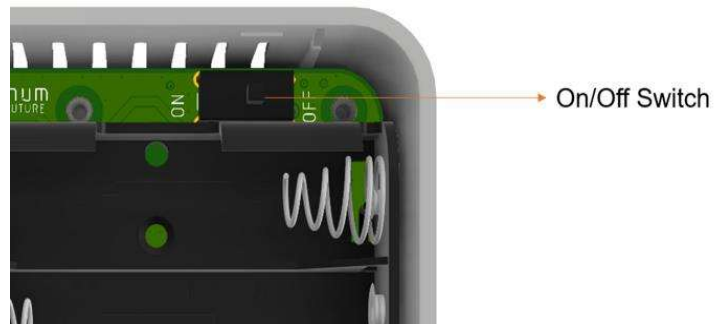
Now place your SIM card in the holder provided.

Close the device and lock it again by sliding the metal cover downwards.

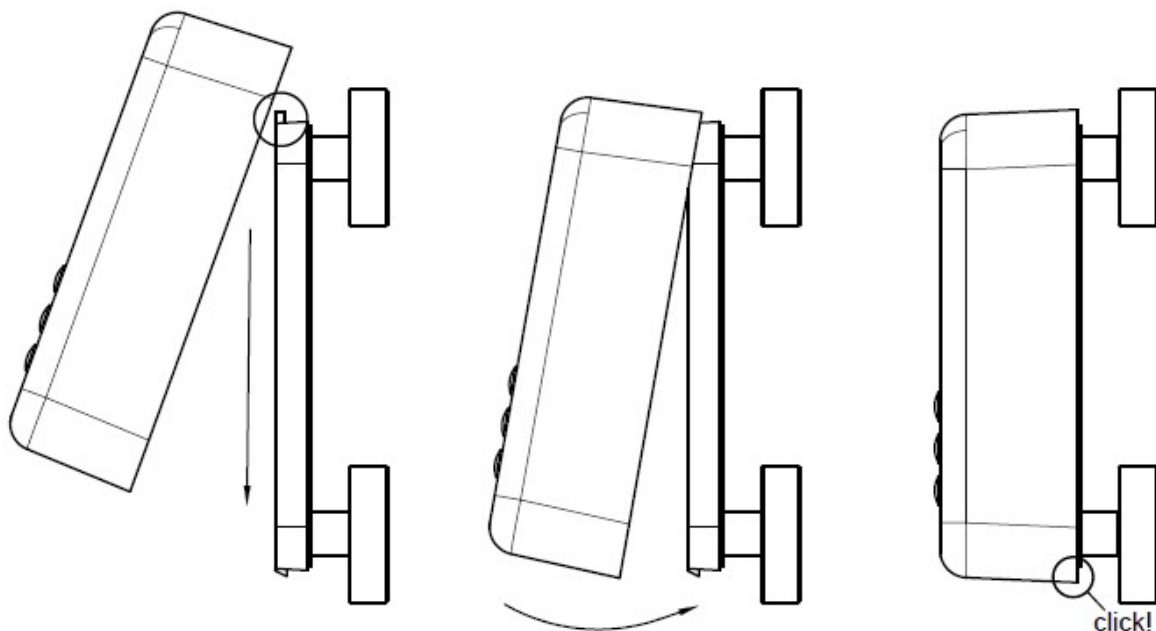


4. Mount the back as a wall bracket at the desired location using magnets, screws or adhesive strips. You will find a detailed description of the mounting options and procedure in the [operating instructions](#) under 5.

5. Press the switch to put the device into operation. After flipping the switch, the sensor starts, the 3 LEDs on the front flash and a tone sequence can be heard. The sensor then carries out a measurement. The LED traffic light shows you the result.

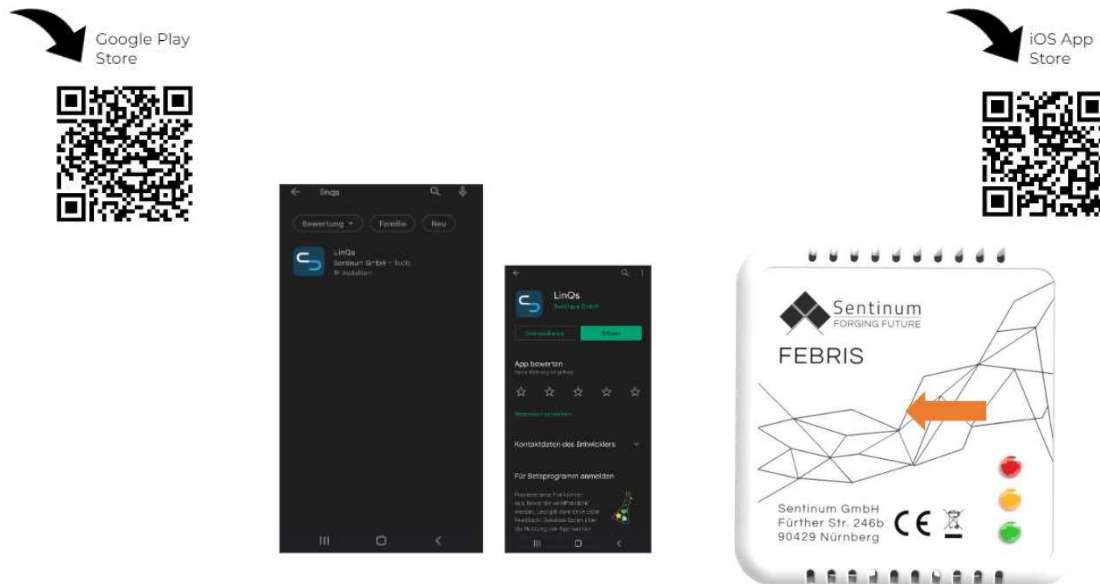


6. Insert the sensor from above into the already attached wall bracket (rear of sensor) until it snaps into the upper bracket. Tilt the sensor backwards so that it snaps securely into the bracket and the snap lock. Ensure that the sensor is fully seated in the bracket and is firmly attached.



NFC PARAMETERIZATION AND POSITION OF THE TAG

First locate the tag on the sensor and then the reader on your end device. You will find the location of the NFC tag at the following position:



Activation takes place via an NFC app. A smartphone is required for this. The app can be downloaded from the respective app stores. Simply search for "Sentinum LinQs" and download the LinQs app.

PAYLOAD DECODER FEBRIS SERIES

```
function decodeUplink(input) {

    var decoded = {};
    var bytes = input.bytes;

    if (input.fPort == 1) { //TELEMETRY

        //decode header
        decoded.base_id = bytes[0] >> 4;
        decoded.major_version = bytes[0] & 0x0F;
        decoded.minor_version = bytes[1] >> 4;
        decoded.product_version = bytes[1] & 0x0F;
        decoded.up_cnt = bytes[2];
        decoded.battery_voltage = ((bytes[3] << 8) | bytes[4]) / 1000.0;
        decoded.internal_temperature = ((bytes[5] << 8) | bytes[6]) / 10 -
100;

        decoded.networkBaseType = 'lorawan';
        decoded.networkSubType = 'tti';

        var it = 7;

        if(decoded.minor_version >= 3){
            it = 7;

            //Luftfeuchte ist bei allen Varianten enthalten
    }
```

```

        decoded.humidity = bytes[it++];

        if (decoded.product_version & 0x01) { // Co2 und Druck sind
enthaltene wenn subversion bit0 = 1, andernfalls 0
            decoded.pressure = (bytes[it++] << 8 | bytes[it++]);
            decoded.co2_ppm = (bytes[it++] << 8 | bytes[it++]);
        } else {
            it += 4; //Werte sind 0 aus kompatibilitäts Gründen, daher
überspringen
        }

        decoded.alarm = bytes[it++]; //Alarm-Level, entspricht grün,
gelb, rot

        //FIFO Werte wegwerfen (1 byte fifo size, 1 byte period, 7
bytes pro fifo eintrag)
        it += 2 + bytes[it] * 7;

        decoded.dew_point = ((bytes[it++] << 8) | bytes[it++]) / 10 -
100;

        // Wandtemperatur und Feuchte enthalten wenn subversion bit 2 =
1
        if (decoded.product_version & 0x04) {
            decoded.wall_temperature = ((bytes[it++] << 8) |
bytes[it++]) / 10 - 100;
            decoded.therm_temperature = ((bytes[it++] << 8) |
bytes[it++]) / 10 - 100;
            decoded.wall_humidity = bytes[it++];
        }

    } else {
        it = 7;

        //Luftfeuchte ist bei allen Varianten enthalten
        decoded.humidity = bytes[it++];

        if (decoded.product_version & 0x01) { // Co2 und Druck sind
enthaltene wenn subversion bit0 = 1, andernfalls 0
            decoded.pressure = (bytes[it++] << 8 | bytes[it++]);
            decoded.co2_ppm = (bytes[it++] << 8 | bytes[it++]);
        } else {
            it += 4; //Werte sind 0 aus kompatibilitäts Gründen, daher
überspringen
        }

        decoded.alarm = bytes[it++]; //Alarm-Level, entspricht grün,
gelb, rot

        //FIFO Werte wegwerfen (1 byte fifo size, 1 byte period, 7
bytes pro fifo eintrag)
        it += 2 + bytes[it] * 7;

        //Taupunkt seit minor version 2 bei alle Varianten enthalten
(ausnahme früher versionen subversion 2, daher byte prüfen)
        if (decoded.minor_version >= 2 && bytes[it] ) {

            decoded.dew_point = bytes[it++] - 100;

```

```
    }

    // Wandtemperatur und Feuchte enthalten wenn subversion bit 2 =
1
    if (decoded.product_version & 0x04) {
        decoded.wall_temperature = bytes[it++] - 100;
        decoded.therm_temperature = bytes[it++] - 100;
        decoded.wall_humidity = bytes[it++];
    }

    }

    return {
        data: decoded,
        warnings: [],
        errors: []
    };
}
```