

FEBRIS QUICK START GUIDE



Febris ist eine smarte, kabellose Sensorreihe von Sentinum zur Überwachung der Raumluftqualität und Umgebungsbedingungen. Je nach Modell misst er CO₂, Temperatur, Luftfeuchtigkeit, Luftdruck oder Oberflächentemperaturen. LEDs und ein Buzzer warnen bei Grenzwertüberschreitungen. Die Daten werden energieeffizient über MIOTY®, NB-IoT oder LoRaWAN® übertragen. Ein Doppeltippen löst eine Sofortmessung aus. Betrieben von 4x AA-Batterien, leicht austauschbar. Wandmontage empfohlen, aber auch Platzierung auf Tischen oder Schränken möglich.

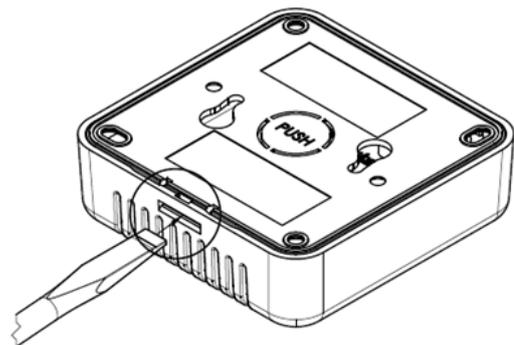
Bitte beachten Sie die Warnungen und Hinweise aus der Betriebsanleitung, um die Sicherheit für Sie, Ihr Umfeld und den Sensor zu gewährleisten!

MONTAGE UND INSTALATION

Installieren Sie den Sensor an einer Innenraumwand in einer Höhe zwischen 1,50m und 1,80m. Achten Sie darauf, dass die LEDs sich auf der rechten, unteren Seite liegen. Beachten Sie, dass der Sensor mindestens 20cm Abstand zu Personen hat und sich in einer Umgebung unterhalb von 2000m über dem Meeresspiegel befindet. Nutzen Sie für die Befestigung entweder Magnete oder geeignete Schrauben. Die detaillierten Schritte finden Sie in der Betriebsanleitung

INBETRIEBNAHME

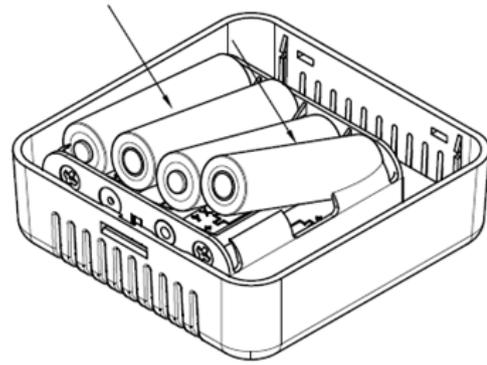
1. Öffnen Sie den Sensor an der dafür vorgesehenen Laschen (im Bild an der Unterkante). Benutzen Sie bei Bedarf vorsichtig einen stumpfen Gegenstand.



2. Legen Sie jetzt die Zellen ein. Um die vorgegebenen Laufzeiten und die in den Datenblätter angegebene Performance zu erreichen sind die folgenden Primärzellen zulässig:

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

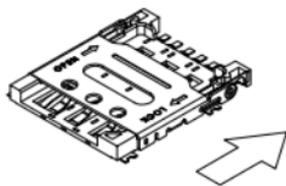
Es werden jeweils 4 Zellen pro Sensor benötigt.



SIM Karten Halter

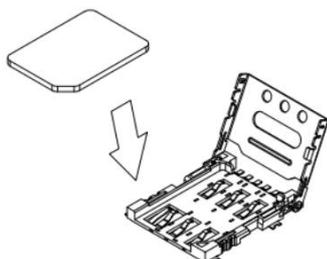
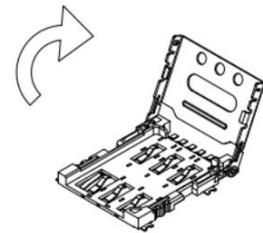


3. Falls Sie einen NB-IoT Sensor besitzen und Sie eine eigene SIM Karte einlegen wollen, legen Sie jetzt die SIM Karte ein.



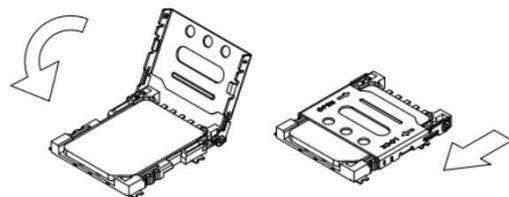
Entriegeln Sie durch Schieben nach oben den Metalldeckel.

Öffnen Sie diesen um die Nano SIM Karte einlegen zu können



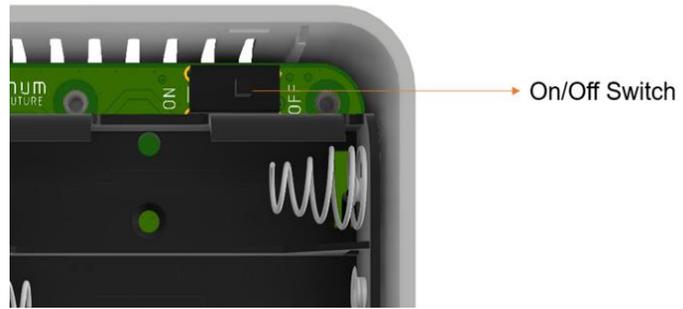
Legen Sie Ihre SIM Karte nun in der dafür vorgesehenen Halterung.

Schließen Sie die Vorrichtung und verriegeln Sie diese wieder durch Verschiebung des Metalldeckels nach unten.

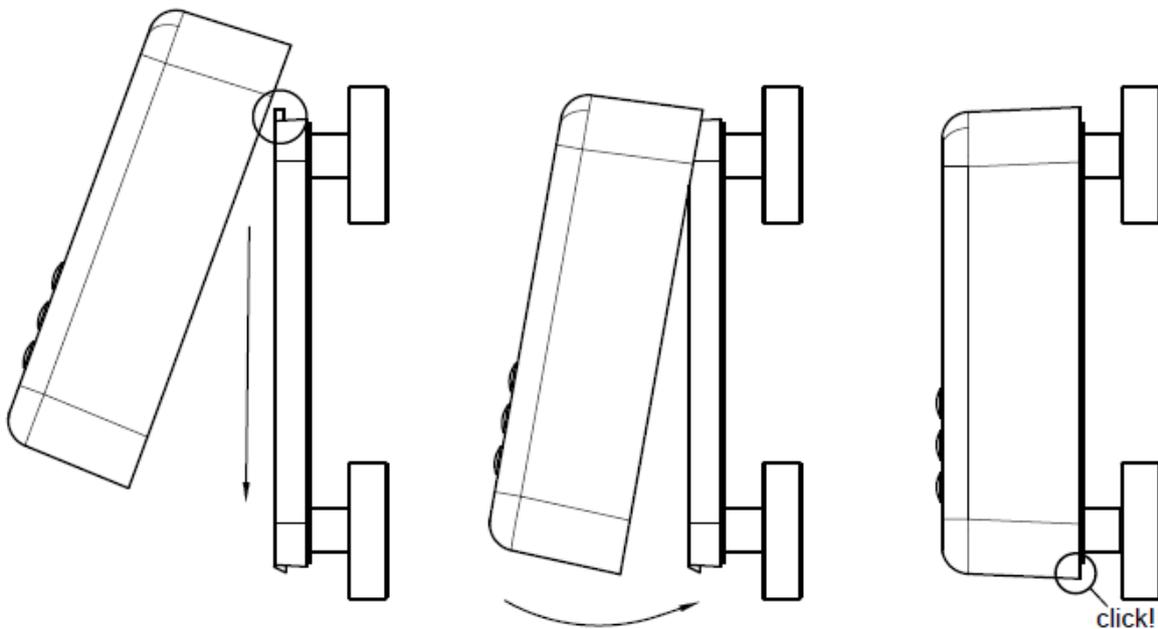


4. Montieren Sie die Rückseite als Wandhalterung mittels Magneten, Schrauben oder Klebstreifen an der gewünschten Stelle. Eine ausführliche Beschreibung der Anbringungsmöglichkeiten und Durchführung finden sie in der [Betriebsanleitung](#) unter 5.

5. Betätigen Sie den Schalter, um das Gerät in Betrieb zu nehmen. Nach Umlegen des Schalters startet der Sensor, die 3 LEDs auf der Vorderseite blinken auf und es ist eine Tonfolge zu hören. Danach führt der Sensor eine Messung durch. Die LED Ampel zeigt Ihnen das Ergebnis.

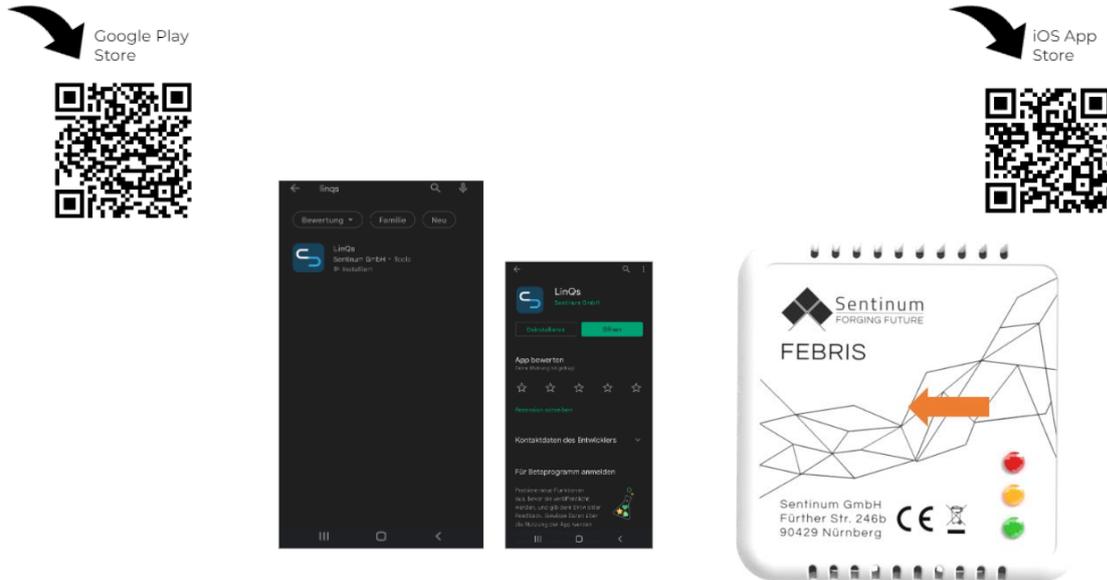


6. Führen Sie den Sensor von oben in die bereits befestigte Wandhalterung (Rückseite Sensor) ein, bis er in der oberen Halterung einrastet. Kippen Sie den Sensor nach hinten, sodass er sicher in der Halterung und dem Schnappverschluss einrastet. Stellen Sie sicher, dass der Sensor vollständig in der Halterung sitzt und fest anliegt.



NFC PARAMETRISIERUNG UND LAGE DES TAGS

Lokalisieren Sie zuerst den Tag auf dem Sensor und dann den Reader an Ihrem Endgerät. Die Lage des NFC Tags finden Sie an folgender Position:



PAYLOAD DECODER FEBRIS SERIE

```
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 = 'ttn';

        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
            enthalten 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
enthalten 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: []
};
}
```