Note: *This document has been automatically translated using DeepL.com. As a result, some content may not be fully accurate or correctly displayed.*

Blockly code:

Ein Bild, das Text, Screenshot, Schrift, Rechteck enthält.

Automatisch generierte Beschreibung

Alternatively as Arduino code:

Part 1

#include <senseBoxIO.h>

#include <SPI.h>

#include <Wire.h>

#include <Adafruit\_GFX.h> // http://librarymanager/All#Adafruit\_GFX\_Library

#include <Adafruit\_SSD1306.h> // http://librarymanager/All#Adafruit\_SSD1306

#include <bsec.h> // http://librarymanager/All#BSEC\_Software\_Library

float bmeTemperature;

float bmeHumidity;

double bmePressure;

float bmeIAQ;

float bmeIAQAccuracy;

int bmeCO2;

float bmeBreathVocEquivalent;

#define SCREEN\_WIDTH 128

#define SCREEN\_HEIGHT 64

#define OLED\_RESET -1

Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &Wire, OLED\_RESET);

Bsec iaqSensor;

void checkIaqSensorStatus(void)

{

if (iaqSensor.bsecStatus != BSEC\_OK) {

if (iaqSensor.bsecStatus < BSEC\_OK) {

for (;;)

errLeds(); /\* Halt in case of failure \*/

}

}

if (iaqSensor.bme68xStatus != BME68X\_OK) {

if (iaqSensor.bme68xStatus < BME68X\_OK) {

for (;;)

errLeds(); /\* Halt in case of failure \*/

}

}

}

void errLeds(void)

{

pinMode(LED\_BUILTIN, OUTPUT);

digitalWrite(LED\_BUILTIN, HIGH);

delay(100);

digitalWrite(LED\_BUILTIN, LOW);

delay(100);

}

Part 2:

void setup() {

display.begin(SSD1306\_SWITCHCAPVCC, 0x3D);

display.display();

delay(100);

display.clearDisplay();

Wire.begin();

iaqSensor.begin(BME68X\_I2C\_ADDR\_LOW, Wire);

checkIaqSensorStatus();

bsec\_virtual\_sensor\_t sensorList[13] = {

BSEC\_OUTPUT\_IAQ,

BSEC\_OUTPUT\_STATIC\_IAQ,

BSEC\_OUTPUT\_CO2\_EQUIVALENT,

BSEC\_OUTPUT\_BREATH\_VOC\_EQUIVALENT,

BSEC\_OUTPUT\_RAW\_TEMPERATURE,

BSEC\_OUTPUT\_RAW\_PRESSURE,

BSEC\_OUTPUT\_RAW\_HUMIDITY,

BSEC\_OUTPUT\_RAW\_GAS,

BSEC\_OUTPUT\_STABILIZATION\_STATUS,

BSEC\_OUTPUT\_RUN\_IN\_STATUS,

BSEC\_OUTPUT\_SENSOR\_HEAT\_COMPENSATED\_TEMPERATURE,

BSEC\_OUTPUT\_SENSOR\_HEAT\_COMPENSATED\_HUMIDITY,

BSEC\_OUTPUT\_GAS\_PERCENTAGE

};

iaqSensor.updateSubscription(sensorList, 13, BSEC\_SAMPLE\_RATE\_LP);

checkIaqSensorStatus();

}

void loop() {

if (iaqSensor.run()) {

bmeTemperature = iaqSensor.temperature;

bmeHumidity = iaqSensor.humidity;

bmePressure = iaqSensor.pressure;

bmeIAQ = iaqSensor.iaq;

bmeIAQAccuracy = iaqSensor.iaqAccuracy;

bmeCO2 = iaqSensor.co2Equivalent;

bmeBreathVocEquivalent = iaqSensor.breathVocEquivalent;

} else {

checkIaqSensorStatus();

}

display.setCursor(0,0);

display.setTextSize(1);

display.setTextColor(WHITE,BLACK);

display.println(bmeTemperature);

display.display();

}