

```
////////////////////////////////////////
// Angst+Pfister Sensors & Power AG      //
// PFLOW3008 communication protocol      //
// Frey Lars                             //
// 09.05.2025 - Revision 0.1             //
////////////////////////////////////////

#include <Wire.h>

// define I2C_Address: 0x01
int address_PFLOW = 0x01;

// buffer for received data
uint16_t Readbyte[18];

//Calculate the crc8 of one byte
uint8_t calculateCRC8(uint8_t data) {

    uint8_t crc = 0;

    crc ^= data;
    for (int j = 0; j < 8; j++) {
        if (crc & 0x80) {
            crc = (crc << 1) ^ 0x07; // CRC-8 polynomial
        }
        else {
            crc <<= 1;
        }
    }
    return crc;
}

// Print serial number of sensor bitwise
// Every third byte is a crc checksum and can
// be used for validation
void print_serial_number(int addr) {
    Wire.beginTransmission(addr);
    Wire.write(0x00);
    Wire.write(0x30);
    Wire.endTransmission(false);

    Wire.requestFrom(addr, 18);
    for(int i = 0; i < 18; i++) {
        Readbyte[i] = Wire.read();
        Serial.print(i);
        Serial.print(": ");
        Serial.println(Readbyte[i], HEX);
    }
}
```

```
// Change I2C address of the sensor
void change_i2c_address(int old_addr, int new_addr) {
    new_addr = new_addr << 1;

    Wire.beginTransmission(old_addr);
    Wire.write(0x00);
    Wire.write(0xA4);

    Wire.write(0x00);
    Wire.write(new_addr);

    uint8_t crc = calculateCRC8(new_addr);

    Wire.write(crc);

    Wire.endTransmission();
}

// Print flow data of sensor bitwise and as calculated flow
// Every third byte is a crc checksum and can
// be used for validation
void read_flow(int addr) {
    Wire.beginTransmission(addr);

    Wire.write(0x00);
    Wire.write(0x3A);
    Wire.endTransmission(false);

    Wire.requestFrom(addr, 6);

    for(int i = 0; i < 6; i++) {
        Readbyte[i] = Wire.read();
        Serial.println(Readbyte[i]);
    }

    long Flow = (long)Readbyte[0] << 24 | (long)Readbyte[1] << 16 |
(long)Readbyte[3] << 8 | (long)Readbyte[4];
    Flow = Flow / 1000.0;

    Serial.print("Flow data: ");
    Serial.println(Flow);
}
```

```
// Print baudrate of sensor (default value 3)
void read_Baudrate(int addr) {
    Wire.beginTransmission(addr);

    Wire.write(0x00);
    Wire.write(0x82);

    Wire.endTransmission();

    Wire.requestFrom(addr, 2);
    for(int i = 0; i < 2; i++) {
        Readbyte[i] = Wire.read();
    }
    Serial.println(Readbyte[1]);
}

// Print filter depth of sensor (default value 3)
void read_Filter(int addr) {
    Wire.beginTransmission(addr);

    Wire.write(0x00);
    Wire.write(0x8C);

    Wire.endTransmission();

    Wire.requestFrom(addr, 2);
    for(int i = 0; i < 2; i++) {
        Readbyte[i] = Wire.read();
    }
    Serial.println(Readbyte[1]);
}

// Setup: I2C clock 100kHz, baudrate 38400 baud/s
void setup() {
    Wire.begin();
    Wire.setClock(100000);
    Serial.begin(38400);
    while (!Serial);
}

void loop()
{
    //Put the function(s) to run here
}
```

We are here for you. Addresses and Contacts.

Headquarter Switzerland:

Angst+Pfister Sensors and Power AG
Thurgauerstrasse 66
CH-8050 Zurich
Phone +41 44 877 35 00
sensorsandpower@angst-pfister.com

Office Germany:

Angst+Pfister Sensors and Power Deutschland GmbH
Edisonstraße 16
D-85716 Unterschleißheim
Phone +49 89 374 288 87 00
sensorsandpower.de@angst-pfister.com

Office North America:

Angst+Pfister North America Inc.
10391 Brecksville Rd.
US-Brecksville, OH 44141
Phone +1 440 375-5212
info.apus@angst-pfister.com

Scan here and get an overview of personal contacts!



<https://sensorsandpower.angst-pfister.com/en/>
