|  |
| --- |
| **2024 ALTIS SW 개인 활동 보고서** |

|  |  |
| --- | --- |
| **활동 개요** | BMP390 및 BMI270 데이터 시트 정리 및 BMP390 raw data 읽기 |
| **일시** | 2024년 05월 18일 19시 |
| **작성자** | 문기준 |
| **활동 내용** | 1. BMP390 및 BMI270 데이터 시트 정리  |  |  | | --- | --- | | BMP390 | BMI270 | | Vin - 3.3V 입력 (전압 레귤레이터 탑재-5V 입력 가능)  3Vo - 3.3V 출력  GND - 그라운드 (-)  SCL – 동기 통신  SDO - 시리얼 출력  SDA - 시리얼 입력  CS - 칩 선택 (SPI 통신용)  INT - 오류 출력 | 3V3 - 3.3V 입력  GND - 그라운드 (-)  SDA - 시리얼 입력  SCL - 시리얼 출력  ADR - 데이터 읽어올 주소 (i2c 통신) / 미리 정의된 주소(pin) 중 하나를 선택  \*하나의 센서에서 여러 I2C 주소 이용  CS - SPI 통신용 칩 통신 핀  ASDx - 외부 센서 연결  ASCx - 외부 센서 연결  \*센서와 센서간 통신  하나의 I2C주소로 여러 센서를 구동 가능  OCS - 외부 컨트롤러 연결 (OIS)  OSDO - 외부 컨트롤러 연결 \*카메라 모듈 연결 시 손떨림 방지 기능 구현 가능  INT1 - 오류 출력1  INT2 - 오류 출력2 |  1. BMP390 raw data 읽어오기  |  | | --- | | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*    This is a library for the BMP3XX temperature & pressure sensor    Designed specifically to work with the Adafruit BMP388 Breakout    ----> http://www.adafruit.com/products/3966    These sensors use I2C or SPI to communicate, 2 or 4 pins are required    to interface.    Adafruit invests time and resources providing this open source code,    please support Adafruit and open-source hardware by purchasing products    from Adafruit!    Written by Limor Fried & Kevin Townsend for Adafruit Industries.    BSD license, all text above must be included in any redistribution   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #include <Wire.h>  #include <SPI.h>  #include <Adafruit\_Sensor.h>  #include "Adafruit\_BMP3XX.h"  #define BMP\_SCK 13  #define BMP\_MISO 12  #define BMP\_MOSI 11  #define BMP\_CS 10  #define SEALEVELPRESSURE\_HPA (1013.25)  Adafruit\_BMP3XX bmp;  void setup() {    Serial.begin(115200);    while (!Serial);    Serial.println("Adafruit BMP388 / BMP390 test");    //if (!bmp.begin\_I2C()) {   // hardware I2C mode, can pass in address & alt Wire    //if (! bmp.begin\_SPI(BMP\_CS)) {  // hardware SPI mode    if (! bmp.begin\_SPI(BMP\_CS, BMP\_SCK, BMP\_MISO, BMP\_MOSI)) {  // software SPI mode      Serial.println("Could not find a valid BMP3 sensor, check wiring!");      while (1);    }    // Set up oversampling and filter initialization    bmp.setTemperatureOversampling(BMP3\_OVERSAMPLING\_8X);    bmp.setPressureOversampling(BMP3\_OVERSAMPLING\_4X);    bmp.setIIRFilterCoeff(BMP3\_IIR\_FILTER\_COEFF\_3);    bmp.setOutputDataRate(BMP3\_ODR\_50\_HZ);  }  void loop() {    if (! bmp.performReading()) {      Serial.println("Failed to perform reading :(");      return;    }    Serial.print("Temperature = ");    Serial.print(bmp.temperature);    Serial.println(" \*C");    Serial.print("Pressure = ");    Serial.print(bmp.pressure / 100.0);    Serial.println(" hPa");    Serial.print("Approx. Altitude = ");    Serial.print(bmp.readAltitude(SEALEVELPRESSURE\_HPA));    Serial.println(" m");    Serial.println();    delay(2000);  } |  * 1. 통신 오류 및 해결 I2C 통신 시도 시 BMP390을 찾지 못하는 오류가 발생했다. 나머지 통신 방식을 시도한 결과, 하드웨어적 SPI 통신 역시 같은 오류가 발생하였다. 소프트웨어적 SPI 통신만이 오류 없이 BMP390의 값을 읽을 수 있었다.   2. Raw data 오류 및 해결책 센서의 위치 및 각도가 변화할 때 특히 데이터들이 갑자기 튀는 현상이 발생했다. 이때 값이 튄 상태를 몇 초간 유지하다 연결이 끊어지는 것으로 보아, 이를 센서와 아두이노 간의 접촉 불량으로 판단하였다. 센서 핀 및 아두이노 케이블을 다시 연결하여 통신을 시도한 결과 정상 적인 값을 얻을 수 있었다. |
| **활동 사진** | <그림1: 시리얼 모니터>  텍스트, 회로, 전자 공학, 스크린샷이(가) 표시된 사진  자동 생성된 설명  <그림2: 회로도> |
| **활동 결과** | 1. BMP390 및 BMI270의 기능 파악 2. BMP390의 통신 오류 및 접지 불량 가능성 발견과 해결 |
| **참고 문헌** |  |

2024년 5월 18일

폰트, 상징, 화이트, 서예이(가) 표시된 사진

자동 생성된 설명

작성자 : 문기준 (인)