QMC6310 3-Axis Hall Effect Sensor Arduino Library Documentation

Overview

This library provides a simple interface for communicating with the QMC6310 Hall sensor over I2C. The library is designed to be used with the Teensy 4.0 microcontroller, but it should be compatible with any microcontroller that supports the Arduino Wire library for I2C communication.

Usage

```
#include "QMC6310.h"

QMC6310 hallSensor;

void setup() {
 if (!hallSensor.begin()) {
  Serial.println("Failed to initialize QMC6310 sensor!");
  while (1);
 }
}

void loop() {
 int16_t x, y, z;
 if (hallSensor.readData(x, y, z)) {
  Serial.print("X: ");
  Serial.print(x);
  Serial.print(" Y: ");
  Serial.print(y);
  Serial.print(" Z: ");
  Serial.println(z);
 } else {
  Serial.println("Failed to read data from QMC6310 sensor!");
 }
}
```

Functions

bool begin()

This function initializes the sensor and sets it to continuous mode with an output data rate of 200Hz and an oversampling rate of 512. The function returns **true** if the sensor was initialized successfully, and **false** otherwise.

bool readData(int16_t &x, int16_t &y, int16_t &z)

This function reads the magnetic field data from the sensor. The function takes three references to **int16_t** variables and fills them with the X, Y, and Z magnetic field data. The function returns **true** if the data was read successfully and is within the expected range, and **false** otherwise.

Future Considerations

This is a basic library and does not include all possible features. Depending on your specific needs, you may want to add additional functions to configure the sensor, read and write other registers, etc.

Limitations

This library does not support changing the I2C address of the sensor. If your sensor has a different I2C address, you will need to update the **QMC6310_ADDRESS** define in the library.

The library does not include any functions for handling errors or exceptions. If an error occurs during communication with the sensor, the library functions will simply return **false**. It is up to the user of the library to decide how to handle these errors.

The library assumes that the Wire library has been initialized with **Wire.begin()** before calling **QMC6310::begin()**. If the Wire library is not initialized, the **QMC6310::begin()** function will not work correctly.