Contents

List of Figures

List of Tables

1. Introduction

This document describes a general program flow to set up ZMOD4510 gas sensor modules for outdoor air quality (OAQ) gas measurements. In addition, it describes the function of an example code provided as a C file by IDT for the ZMOD4510, which can be operated using its evaluation kit (EVK).

Recommendation: Before using this document, read the *ZMOD4510 Datasheet* available on IDT.com.

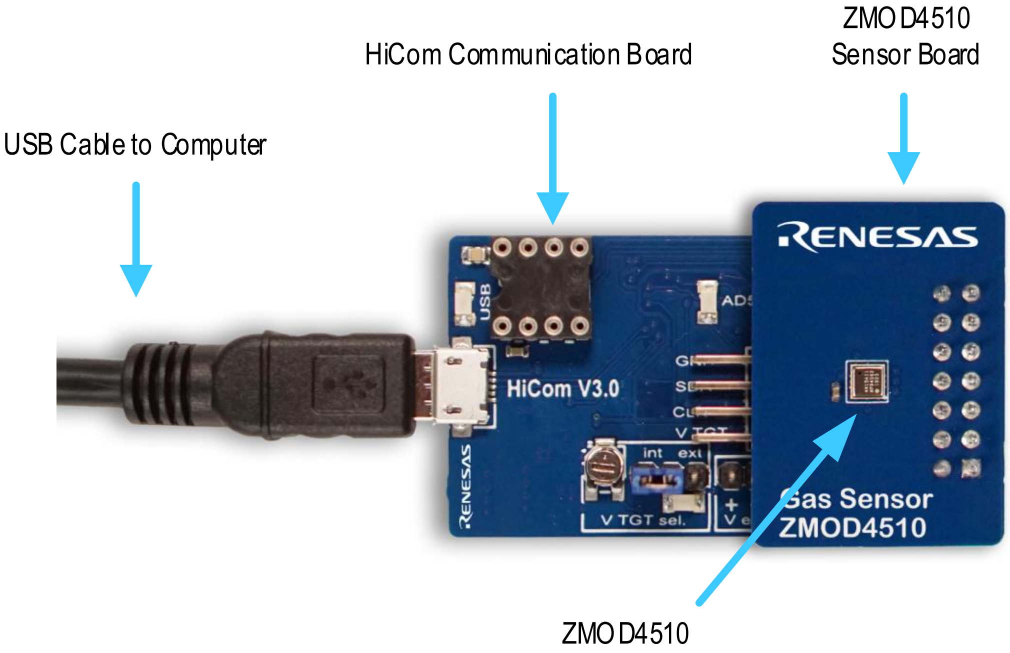
2. Hardware Setup for the ZMOD4510

To implement the ZMOD4510 sensor with the user's program, specific hardware is needed. The following example shows the ZMOD4510 EVK, which consist of three components:

- HiCom Communication Board
- ZMOD4510 Sensor Board (Daughter Board) with the ZMOD4510 Gas Sensor Module
- Micro-USB cable

For instructions on assembly, connections, and installation of hardware and software, refer to the *ZMOD4510 Evaluation Kit Description*. Figure 1 shows the assembled evaluation kit.

Figure 1. ZMOD4510 Evaluation Kit



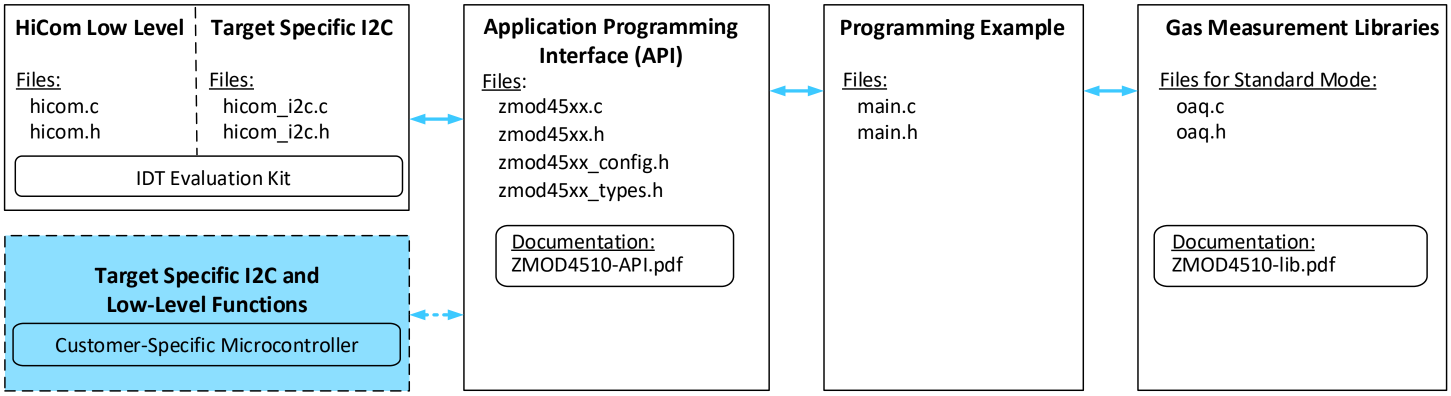
3. General Program Flow for Setting up ZMOD4510 Gas Measurements

To operate the ZMOD4510 in the hardware and use its full functionality, four code blocks are needed as displayed in Figure 2:

- The "Target Specific I2C and Low-Level Functions" block is the hardware-specific implementation of the I2C interface; it contains read and write functions to communicate with the ZMOD4510. If the IDT EVK is used, files for the HiCom Communication Board are provided with this manual. Custom microcontrollers can be used to establish I2C communication. Using the user's own microcontroller requires implementing the user's own target-specific I2C and low-level functions (highlighted in light blue in Figure 2).
- The "Application Programming Interface (API)" block contains the functions needed to work with the ZMOD4510. A detailed description of the API can be found in the document *ZMOD4510-API.pdf*, which is included with the download of the code file.
- The "Programming Example" block provides a code example that is used to initialize the ZMOD4510 gas sensor module and display the data for the Air Quality Index (AQI) based on the rating of the US Environmental Protection Agency (EPA)[1]. Further details can be found in "Description of the Programming Example Code."
- The "Libraries" block contains the functions and data structures needed to calculate the AQI. The library is described in more detail in the document *ZMOD4510-lib.pdf*, and is available for download with the code files from the IDT website.

To avoid naming conflicts, all function names start with the prefix "`zmod45xx_`" in the ZMOD4510 code. This naming applies to all operation methods of the ZMOD4510.

Figure 2. File Overview for ZMOD4510



[1] Source: https://airnow.gov/index.cfm?action=aqibasics.aqi

4. Description of the Programming Example Code

This section describes the example code and how to use the ZMOD4510 Outdoor Air Quality Gas Sensor Module. In the example, the ZMOD4510 is initialized and configured for operation, and then it displays measured air quality index (AQI) values. The example is intended to work on a Windows® (trademark of Microsoft, Inc.) computer in combination with the IDT gas sensor EVK. However, the example can be adjusted to operate on other platforms such as ARM® (trademark of ARM, Ltd.) and Linux® (trademark of Linus Torvalds).

To run the example using the EVK without further configuration, run *ZMOD45xx_example.exe,* which is included in the kit software.

4.1 *main* Files

The *main.c / main.h* files contain the main program flow.

The ZMOD4510 Outdoor Air Quality (OAQ) algorithms are configured by setting the parameters according to the ZMOD4510 libraries, then the target-specific initialization are performed. The ZMOD4510 is configured by reading the Final Module Test parameters from the sensor's nonvolatile memory (NVM) and initializing it to run at its operation temperature.

An endless loop continuously checks the status of the ZMOD4510 and reads its data. The raw data is subsequently processed, and the OAQ algorithm for Air Quality Index (AQI) is calculated. The values are shown in a command line window. To stop the loop, press any key, which releases the hardware and stops the program.

4.2 Program Flow to Operate the Sensor

For more information, refer to the example code.

Table 1. Program Flow

Note: In the following table, lines that are shaded blue can be run in an endless loop with polling or interrupt usage.

Line	Program Actions	Notes	API Functions
1	Reset the sensor.	Before configuring the sensor, reset the sensor by powering it off/on or toggling the reset pin.	–
2	Read device parameters from the nonvolatile memory (NVM).	This step is required to select the correct configuration for the sensor.	`zmod45xx_read_sensor_info`
3	Initial initialization.	This function must be called after every startup.	`zmod45xx_init_sensor`
4	Initialize the sensor for OAQ measurements.	Initialize the sensor.	`zmod45xx_init_measurement`
5	Start the measurement.	Start the measurement.	`zmod45xx_start_measurement`
6	Initialize the algorithm	Initializes the OAQ algorithm.	–
7	Read the status register.	Wait until the measurement is done. This will also be signaled by an interrupt.	`zmod45xx_read_status`
8	Get the ADC values.	Read ADC values from the sensor.	`zmod45xx_read_adc_results`
9	Start the measurement.	Start the measurement.	`zmod45xx_start_measurement`
10	Calculation.	Calculate Mox resistances.	`zmod45xx_calc_rmox`
11	Calculation algorithm.	Calculate Air Quality Index (AQI) from present sample.	–

5. Using the Example on a Different Hardware Platform

To incorporate this programming example into a different hardware platform, it is necessary to set the device's struct pointers *read*, *write,* and *delay_ms*. The type definitions of the function pointers can be found in *zmod45xx_types.h* (see Figure 2). The functions *read* and *write* should point to the I2C implementation of the hardware used.

In addition, IDT provides precompiled algorithm libraries that are hardware-platform dependent. IDT offers a download of these libraries for x86/x64 (Linux/Windows) as well as for ARM® Cortex®-M Series, MSP430 Series, 8051, and RL78 (GCC) microcontrollers.

5.1 Adaptation for the Target System

IDT's ZMOD4510 C API is located between the application and the target hardware.

Figure 3. System Hierarchy

<table>
<tr><td colspan="2">Application</td></tr>
<tr><td colspan="2">Application-Specific ZMOD4510 Configuration</td></tr>
<tr><td colspan="2">ZMOD4510 API and Libraries (Algorithm)</td></tr>
<tr><td>Low-Level I2C Communication</td><td>Low-Level Hardware Functions</td></tr>
<tr><td colspan="2">Target Hardware</td></tr>
</table>

The low-level I2C functions are implemented in the file *hicom_i2c.c* (see Figure 2) for the EVK hardware running on a Windows-based computer and the HiCom Communication Board.

5.2 Error Codes

Most of the API functions return a code to indicate the success of the operation. If no error occurred, the return code is 0. In the event of an error, a number not equal to zero is returned. The API has predefined symbols `ZMOD45XX_ERROR_*` for the error codes defined in *zmod45xx_types.h*.

6. Revision History

Revision Date	Description of Change
September 4, 2019	Initial release.

Corporate Headquarters
6024 Silver Creek Valley Road
San Jose, CA 95138
www.IDT.com

Sales
1-800-345-7015 or 408-284-8200
Fax: 408-284-2775
www.IDT.com/go/sales

Tech Support
www.IDT.com/go/support

DISCLAIMER Integrated Device Technology, Inc. (IDT) and its affiliated companies (herein referred to as "IDT") reserve the right to modify the products and/or specifications described herein at any time, without notice, at IDT's sole discretion. Performance specifications and operating parameters of the described products are determined in an independent state and are not guaranteed to perform the same way when installed in customer products. The information contained herein is provided without representation or warranty of any kind, whether express or implied, including, but not limited to, the suitability of IDT's products for any particular purpose, an implied warranty of merchantability, or non-infringement of the intellectual property rights of others. This document is presented only as a guide and does not convey any license under intellectual property rights of IDT or any third parties.

IDT's products are not intended for use in applications involving extreme environmental conditions or in life support systems or similar devices where the failure or malfunction of an IDT product can be reasonably expected to significantly affect the health or safety of users. Anyone using an IDT product in such a manner does so at their own risk, absent an express, written agreement by IDT.

Integrated Device Technology, IDT and the IDT logo are trademarks or registered trademarks of IDT and its subsidiaries in the United States and other countries. Other trademarks used herein are the property of IDT or their respective third party owners. For datasheet type definitions and a glossary of common terms, visit www.idt.com/go/glossary. All contents of this document are copyright of Integrated Device Technology, Inc. All rights reserved.