SENSORS SOFTWARE DEVELOPMENT KIT

USER MANUAL

VERSION 1.5

JANUARY 28, 2021

Revision history

Manual version	SW version	Notes	Date
1.0	1.0	• Initial version of this document	August 2019
1.1	1.1	• Updated sensors supported and software history • Changed the wiringPi install method as recommended by it's author.	November 2019
1.2	1.2	• Updated sensors supported and software history	March 2020
1.3	1.3	• Updated sensors supported and software history	September 2020
1.4	1.5	• Updated sensors supported and software history	December 2020
1.5	1.5	• fixed link to raspbian	January 2021

⋆ For SDK version history see chapter `Software history`

Abbreviations and abstract

Abbreviation	Name	Description
ADC	Analog to Digital converter	
ASIC	Application specific integrated circuit	
GPIO	General Purpose Input Output	
IDE	Integrated development environment	A set of tools for software development
I^2C	Inter-integrated circuit	Serial communication interface and protocol
MCU	Microcontroller	
SDK	Software development kit	Software development tool to integrate various Würth Elektronik eiSos products
SPI	Serial peripheral interface	Serial communication interface and protocol

Contents

1 Introduction

The Würth Elektronik eiSos range of sensors provide advanced sensing capabilities to any embedded application. The built-in digital interface enables easy integration with any of the most commonly used host MCUs through industry standard serial communication interfaces like I^2C. This also allows complete configuration and control of the sensor via software running on the host MCU.

The Sensors SDK is a set of software tools that enable quick software integration of Würth Elektronik eiSos sensors to application software on the host MCU. It consists of a set of platform-independent drivers for the sensors and sample applications developed on the Raspberry Pi platform written in C.

1.1 Motivation

A typical sensor from Würth Elektronik eiSos consists of the sensing element along with an ASIC which implements the control logic. All configurations of the sensors are done by writing specific values to a set of control registers on board. One or more status registers can be monitored to get the current status of the sensor and finally, the output is stored in the output registers. These sensors not only deliver the raw values of the physical parameter being sensed, but also perform several advanced functions like filtering, FIFO-storage, threshold detection, interrupt generation etc.

To use the complete feature set of such a sensor, control of all the necessary registers have to be implemented on the host MCU. This involves considerable effort for the user. The aim of the Sensors SDK is to minimize the effort required to integrate the sensor control software to the host application.

It contains the implementations of all the available functions of the sensors in pure C-code. In order to integrate any Würth Elektronik eiSos sensor to the application, the user has to simply port the corresponding C-code to his host MCU. This significantly reduces the time needed for developing the software interface to the sensor.

The steps for porting are explained in more detail in chapter 3.

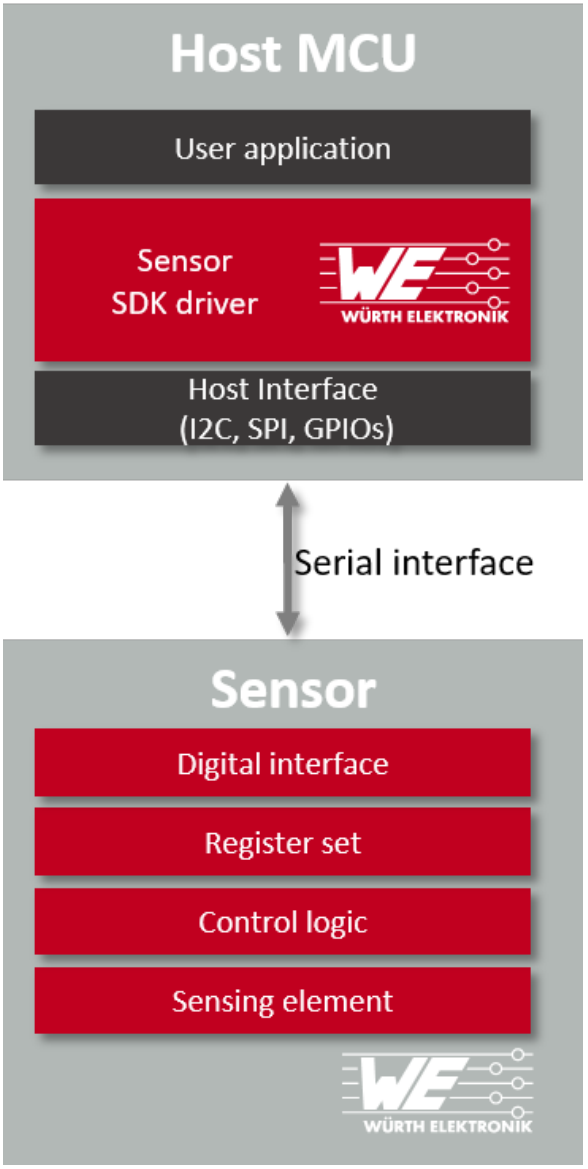

Figure 1: Sensors SDK driver as part of the end product

2 Sensors SDK overview

In this chapter, constituents of the Sensors SDK are described in details.

2.1 Platform-independent sensor drivers

This part of the Sensors SDK consists of a set of C header and source file per sensor. Every source file contains implementations for configuration and control of the sensors that is exposed to the application via functions and type declarations in the corresponding header file. Being platform independent and written in C enables easy porting of the drivers to any custom MCU platform.

2.2 Sample applications

In order to kick-start the application development, several sample applications that includes typical use-cases of the sensors are made available as a part of the Sensors SDK. These examples are developed on the Raspberry Pi platform. The Sensors SDK contains the complete source code for all the sample applications including the project files for the Code::Blocks IDE.

The sample applications were tested on Raspberry Pi 3 Model B with Raspbian Stretch and Raspberry Pi 3 Model B+ with Raspbian Buster. Code::Blocks version 17.12 was used as the IDE

These sample applications allow the user to take the sensor into operation. Additionally, the availability of the source code allows the user to try out several configurations and decide on the ones suitable for the application with minimal effort.
The sensors supported by the different versions of the Sensors SDK are shown in table 1.

SDK version	Sensor	Interface	Sensor type
1.0 and higher	WSEN-PADS (2511020213301)	I^2C	Absolute pressure sensor
1.1 and higher	WSEN-ITDS (2533020201601)	I^2C	Acceleration sensor
1.1 and higher	WSEN-PDUS (25131308XXX01)	I^2C	Differential pressure sensors
1.2 and higher	WSEN-TIDS (2521020222501)	I^2C	Temperature sensor
1.3 and higher	WSEN-HIDS (2523020210001)	I^2C and SPI	Humidity sensor
1.4 and higher	WSEN-PADS (2511020213301)	SPI	Absolute pressure sensor
1.4 and higher	WSEN-ITDS (2533020201601)	SPI	Acceleration sensor

Table 1: Sensor support in the Sensors SDK

The Differential pressure sensors with matchcode WSEN-PDUS work with 5V logic and should not be directly connected to Raspberry Pi. Hence, no sample applications (example directory) for this sensors are currently implemented for the Raspberry Pi platform.

2.3 Contents of the Sensors SDK

The Sensors SDK is delivered as a compressed zip-file. All code related to a sensor supported by the Sensors SDK are placed under a sub directory named after the corresponding sensor.

Each sensor directory contains two sub-directories, **drivers** and **examples**. The sub-directory **drivers** contains the platform-independent code that can be used to port to the custom MCU.

The **examples** folder contains sample applications that run on the Raspberry-Pi. The sub-directory **platform** contains the platform-specific implementation for the Raspberry-Pi.

The current version of the Sensors SDK is specified in the **version.h** file.

```
/
├─platform ..............................Platform specific code for the Raspberry Pi
│  ├─platform.c
│  └─platform.h
├─version.h ..............................................Sensors SDK version file
├─...
├─WSEN_PADS_2511020213301 .....Contains all the code related to sensor 2511020213301
│  ├─drivers ....................................Code to be ported to custom hosts
│  │  ├─WSEN_PADS_2511020213301.h ..................Function and type declarations
│  │  └─WSEN_PADS_2511020213301.c ........Implementation of the sensor functionality
│  └─examples ....................................Sample application WSEN-PADS
│     ├─...
│     ├─quickStart ..........................................Quick start example
│     │  ├─main.c
│     │  └─quickStart.cbp
│     └─...
├─WSEN_ITDS_2533020201601 .....Contains all the code related to sensor 2533020201601
│  ├─drivers ....................................Code to be ported to custom hosts
│  │  ├─WSEN_ITDS_2533020201601.h ..................Function and type declarations
│  │  └─WSEN_ITDS_2533020201601.c ........Implementation of the sensor functionality
│  └─examples ..................................... Sample application WSEN-ITDS
│     ├─...
│     ├─quickStart ..........................................Quick start example
│     │  ├─main.c
│     │  └─quickStart.cbp
│     └─...
└─...
```

3 Host integration

As described in chapter 2 the Sensors SDK has a platform independent driver component and Raspberry-pi specific examples which demonstrate the use of these drivers. In the following, the steps involved in porting the drivers to a custom platform is described.

The contents of the **drivers** directory has to be directly integrated into the custom project. For example, in case of WSEN_PADS, include the WSEN_PADS_2511020213301.h and WSEN_PADS_2511020213301.c files to the custom project.

The file **platform.h** declares platform specific functions that has to be implemented by the user on the custom host. For example, the access to I²C interface of the host is declared with the following functions.

```
/**
* @brief   Initialize   the I2C Interface
* @param I2C address
* @retval Error Code
*/
int8_t  I2CInit(int address);

/**
* @brief   Read data starting from the addressed register
* @param -RegAdr : the register address to read from
*          -NumByteToRead : number of bytes to read
*          -pointer Data : the address store the data
* @retval Error Code
int8_t  ReadReg(uint8_t RegAdr, int NumByteToRead, uint8_t *Data);

/**
* @brief  Write data  strarting  from the addressed register
* @param -RegAdr : Address to write in
*          -NumByteToWrite : number of bytes to write
*          -pointer Data : Address of the data to  be written
* @retval Error Code
*/
int8_t  WriteReg(int RegAdr, int NumByteToWrite, uint8_t *Data);
```

Code 1: Code snippet for I²C interface

Here the definition of these functions, depending on the host peripherals, has to be implemented by the user. The function **I2CInit()** should initialize the I²C peripheral on the custom MCU and the functions **ReadReg()** and **WriteReg()** should enable reading/writing data bytes over the I²C bus.

Please refer to the user manual of the corresponding sensor for a detailed description of the characteristics of the digital interface being used.

The register read and write functions shall be implemented to read/write one byte at a time. Burst read/write operations can be implemented based on the host MCU peripheral and the sensor used.

The existing **platform.c** file, can be removed from the project as it contains the corresponding implementation for the Raspberry Pi.

On porting the aforementioned functions onto the custom MCU, the driver can be used seamlessly for the application development. The sample applications can be used as a starting point for further development.

4 Running Sensors SDK sample application on the Raspberry Pi

This chapter explains the steps involved in running the sample applications of the Sensors SDK on the Raspberry Pi platform.

4.1 Hardware connections

For creating custom applications on the basis of the Raspberry Pi, connect the pins of the sensor evaluation board to corresponding pins on the Raspberry Pi (power supply, ground, I^2C). Figure 2 gives an overview of the pins of the Raspberry Pi used in the driver application examples. Refer to the respective user manuals of the evaluation boards for further details.

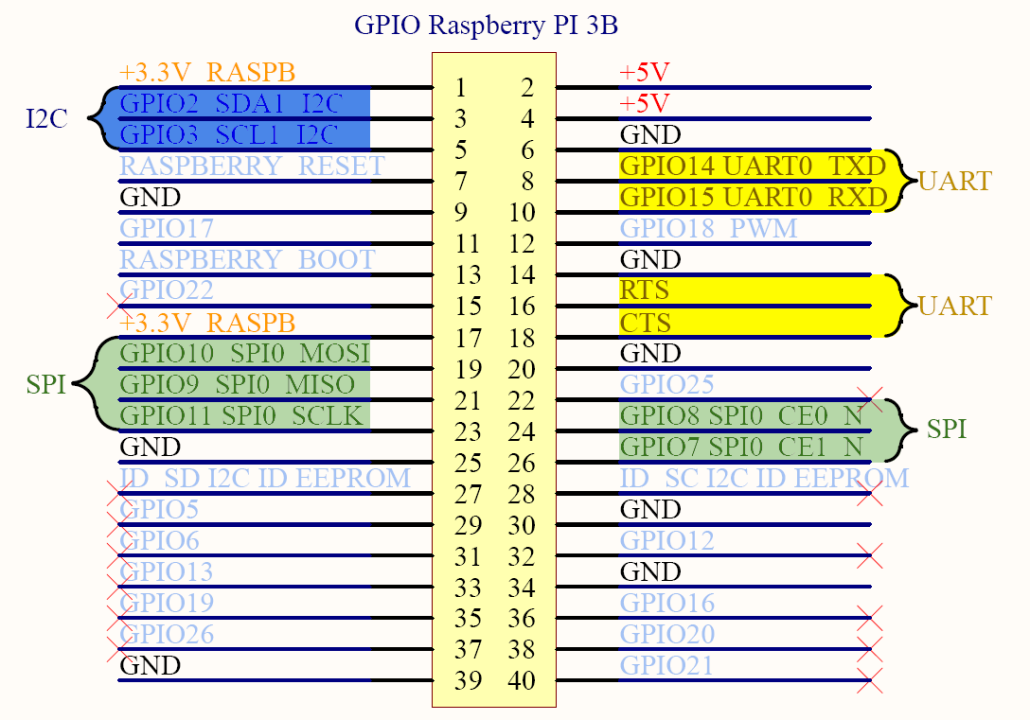

Figure 2: Pinout of 40-pin GPIO header of Raspberry Pi 3B

For example, figure 3 shows a block diagram for connecting a sensor via I^2C interface to Raspberry Pi.

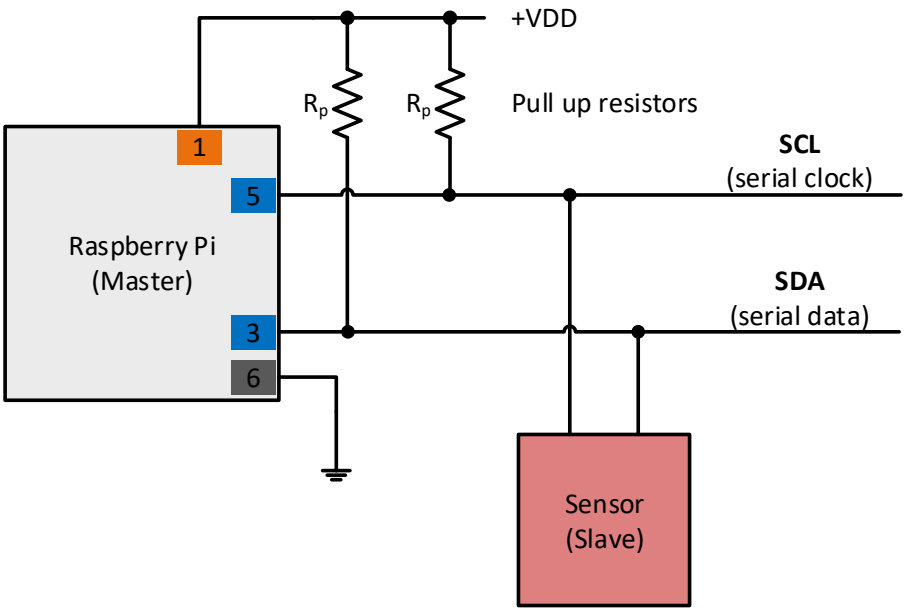

Figure 3: Connecting an I²C sensor to Raspberry Pi 3B

Alternatively, Würth Elektronik eiSos Amber-Pi, an add-on module for the Raspberry Pi can be used to connect the sensor evaluation board directly to the Raspberry Pi.

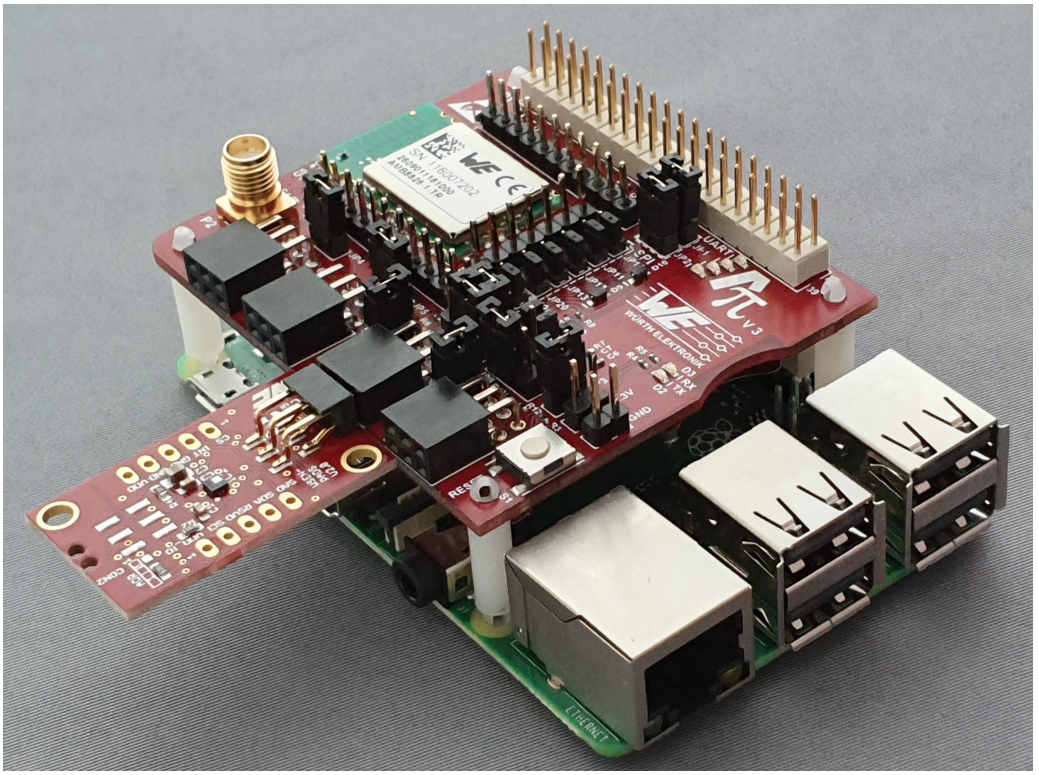

Figure 4: Sensor evaluation board with Amber-Pi

4.2 Install the Raspbian OS on the Raspberry Pi

1. First of all the Raspberry Pi has to be installed and configured.
   a) Download the latest version of Raspbian with GUI from the download section of *https://www.raspberrypi.org*
   b) Install the Raspbian OS by writing its image on your SD-card. On a Windows machine the Win32DiskImager tool can be used, as described here *www.raspberrypi.org/documentation/installation/installing-images/windows.md*
2. After installing the image on the SD card, insert it into the Raspberry Pi's SD card slot, connect your monitor, mouse and keyboard. Now the Raspberry Pi is ready to boot up. Please start it by powering it up.
3. After booting the Raspberry Pi, switch off the Bluetooth® interface by clicking on the Bluetooth® button on the right upper corner of the screen (see figure 5).
4. Then turn on the WiFi for connecting to the internet by clicking on the WiFi button on the right upper corner of the screen and selecting the WiFi of your choice.

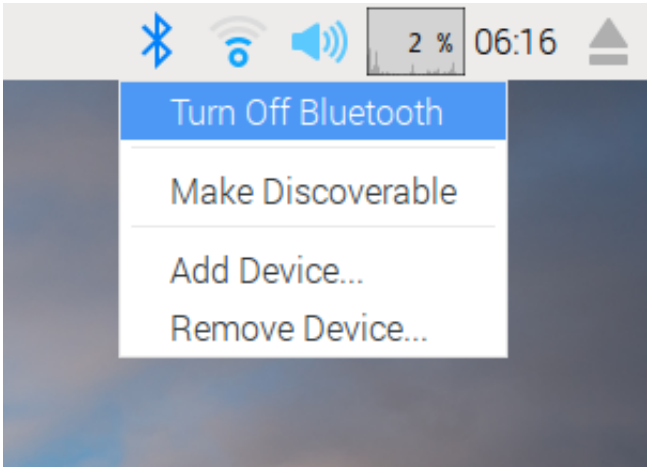

Figure 5: Switch off the Bluetooth® and connect to internet via WiFi

5. After connecting to the internet make sure your Raspberry Pi is up to date with the latest versions of Raspbian OS. To update the system open a terminal by clicking on the terminal symbol in the left upper corner (see figure 6).

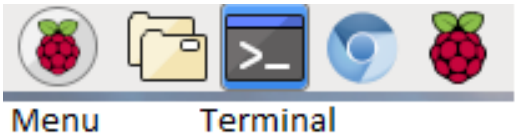

Figure 6: Terminal button

6. Then upgrade the Raspbian OS by typing in terminal:

```
sudo apt-get update
sudo apt-get upgrade
```

4.2.1 Configuring the peripherals

1. Next, the peripherals have to be enabled. To do so open the menu by clicking on the Raspberry Pi symbol on the left upper corner of the screen and open the **Preferences** → **Raspberry Pi Configuration** window (see figure 7). Enable the SPI, I²C and SERIAL interface.

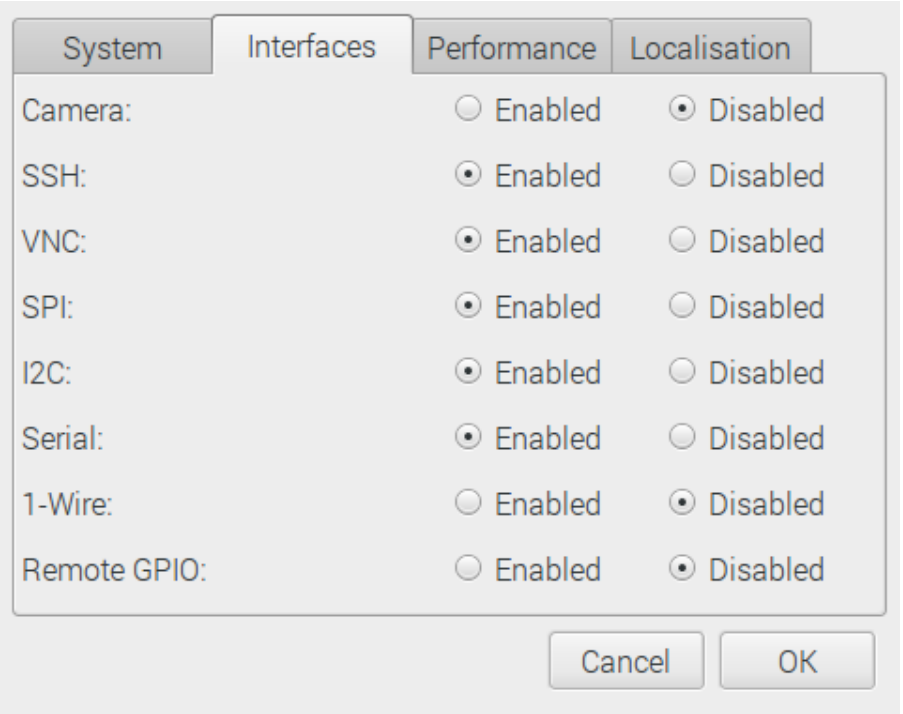

Figure 7: Raspberry Pi interface configuration

2. After enabling the interfaces a dialog should appear asking for a reboot to apply the changes. If no dialog appears reboot by clicking on the Raspberry symbol on the left upper corner of the screen and select **Shutdown**.

3. In order to use the peripherals as a non-root user, the local user has to be a member of the peripheral group. In order to check this, type in the following in the terminal,

```
groups
```

   If the output contains GPIO, SPI and I²C, then skip the next step.

4. Add the current user to the groups by typing in the following commands in the terminal,

```
sudo adduser pi gpio
sudo adduser pi i2c
sudo adduser pi spi
```

   Logout and login to update the user group settings.

4.3 Install the wiringPi library

The wiringPi library is used to easily access the peripherals of the Raspberry Pi.

1. First check if wiringPi is already installed. In a terminal type:

   ```
   gpio -v
   ```

   If you get the version number, then you have it installed already. In this case continue with 4.4.
   If you want to work on a Raspberry Pi 4B make sure you have wiringPi version 2.52 (or newer) installed.

2. If it is not installed, install wiringPi as described in *http://www.wiringpi.com/download-and-install/* or *http://wiringpi.com/wiringpi-updated-to-2-52-for-the-raspberry-pi-4b/*.

4.4 Install the Sensors SDK

The Sensors SDK was developed in the Code::Blocks development environment.

1. Thus first download and install the software Code::Blocks. Therefore open a terminal and type:

   ```
   sudo apt-get install codeblocks
   ```

2. Now download the Sensors SDK driver as zip file from (*www.we-online.de/wcs-software*) to the location ~/**Downloads**

3. The file is going to be extracted to the folder ~/**Projects**. If the folder does not exist create it by typing in terminal:

   ```
   mkdir ~/Projects
   ```

4. Now extract the Sensors SDK to ~/**Projects** by typing in terminal:

   ```
   unzip ~/Downloads/WSEN_SDK.zip -d ~/Projects
   ```

5. Then start the desired project via Code::Blocks by typing in terminal. For example,

   ```
   codeblocks ~/Projects/WSEN-SDK/WSEN_PADS_2511020213301/
       examples/quickStart/quickstart.cbp &
   ```

6. Now include the wiringPi libraries into Code::Blocks by opening the global linker settings in **Settings** → **Compiler** → **Linker Settings** and adding the library /**usr/lib/libwiringPi.so** to the **Link libraries** field (see figure 8).

7. Additionally add **-pthread** in the **Other linker options** field. Close the linker settings again.

All necessary libraries are also linked in the projects linker settings to not run into trouble in case they have not been linked in the global linker settings.

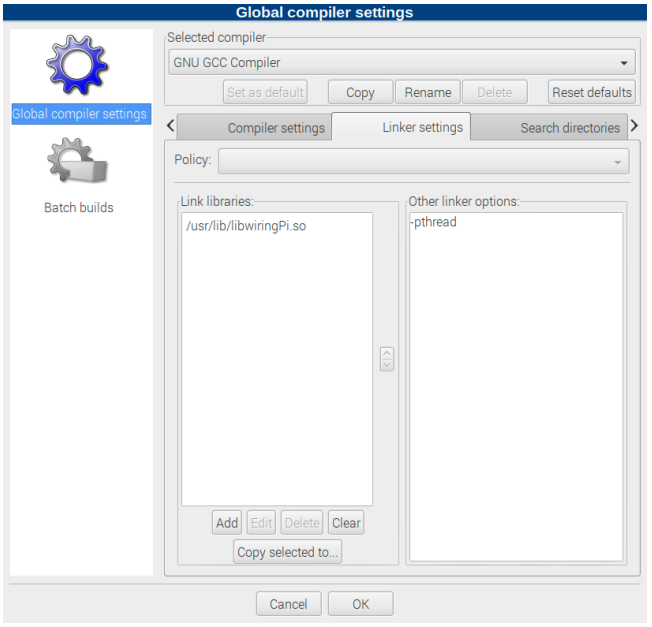

Figure 8: Code::Blocks linker settings

8. Then press **Build** → **Rebuild** to build the project (see figure 9).

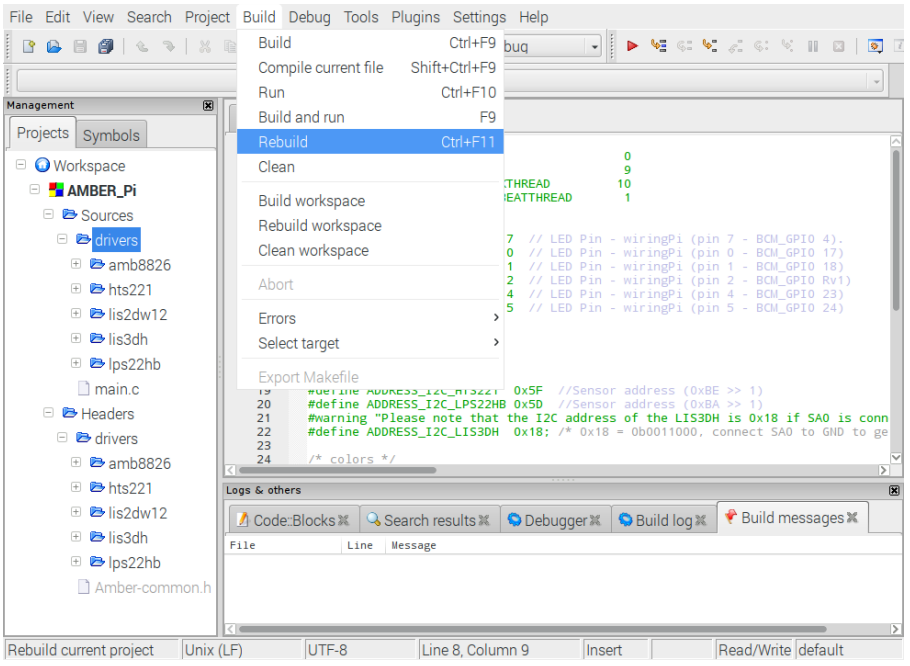

Figure 9: Rebuild the application

9. If it builds without errors the Raspberry Pi, WiringPi and Sensors SDK setup succeeded.

10. In case of further question, please contact our technical support at *https://we-online.com/wireless-connectivity/support*

5 Software history

Version 1.0.0 "Release"

- First released version of the SDK.
- PADS driver and example.

Version 1.1.0 "Release"

- Added driver support for ITDS and PDUS sensors.
- Added example for ITDS.

Version 1.2.0 "Release"

- Added driver support for TIDS sensor.
- Added temperature readout functions to ITDS drivers.
- Updated the quick start example of the PADS sensor.

Version 1.3.0 "Release"

- Added driver support for HIDS sensor.
- Added quick start example for HIDS sensor.
- Updated the Raspberry Pi specific platform files to support the SPI interface.

Version 1.4.0 "Internal"

Version 1.5.0 "Release"

- Added SPI support for PADS and ITDS sensor.

6 Important notes

The following conditions apply to all goods within the sensors product range of Würth Elektronik eiSos GmbH & Co. KG:

6.1 General customer responsibility

Some goods within the product range of Würth Elektronik eiSos GmbH & Co. KG contain statements regarding general suitability for certain application areas. These statements about suitability are based on our knowledge and experience of typical requirements concerning the areas, serve as general guidance and cannot be estimated as binding statements about the suitability for a customer application. The responsibility for the applicability and use in a particular customer design is always solely within the authority of the customer. Due to this fact, it is up to the customer to evaluate, where appropriate to investigate and to decide whether the device with the specific product characteristics described in the product specification is valid and suitable for the respective customer application or not. Accordingly, the customer is cautioned to verify that the documentation is current before placing orders.

6.2 Customer responsibility related to specific, in particular safety-relevant applications

It has to be clearly pointed out that the possibility of a malfunction of electronic components or failure before the end of the usual lifetime cannot be completely eliminated in the current state of the art, even if the products are operated within the range of the specifications. The same statement is valid for all software and software parts contained in or used with or for products in the sensor product range of Würth Elektronik eiSos GmbH & Co. KG. In certain customer applications requiring a high level of safety and especially in customer applications in which the malfunction or failure of an electronic component could endanger human life or health, it must be ensured by most advanced technological aid of suitable design of the customer application that no injury or damage is caused to third parties in the event of malfunction or failure of an electronic component.

6.3 Best care and attention

Any product-specific data sheets, manuals, application notes, PCN's, warnings and cautions must be strictly observed in the most recent versions and matching to the products revisions. This documents can be downloaded from the product specific sections on the wireless connectivity and sensors homepage.

6.4 Customer support for product specifications

Some products within the product range may contain substances, which are subject to restrictions in certain jurisdictions in order to serve specific technical requirements. Necessary information is available on request. In this case, the field sales engineer or the internal sales person in charge should be contacted who will be happy to support in this matter.

6.5 Product improvements

Due to constant product improvement, product specifications may change from time to time. As a standard reporting procedure of the Product Change Notification (PCN) according to the JEDEC-Standard, we inform about major changes. In case of further queries regarding the PCN, the field sales engineer, the internal sales person or the technical support team in charge should be contacted. The basic responsibility of the customer as per section 6.1 and 6.2 remains unaffected.
The sensor driver software ¨Sensor SDK¨ and it's source codes are not subject to the Product Change Notification information process.

6.6 Product life cycle

Due to technical progress and economical evaluation we also reserve the right to discontinue production and delivery of products. As a standard reporting procedure of the Product Termination Notification (PTN) according to the JEDEC-Standard we will inform at an early stage about inevitable product discontinuance. According to this, we cannot ensure that all products within our product range will always be available. Therefore, it needs to be verified with the field sales engineer or the internal sales person in charge about the current product availability expectancy before or when the product for application design-in disposal is considered. The approach named above does not apply in the case of individual agreements deviating from the foregoing for customer-specific products.

6.7 Property rights

All the rights for contractual products produced by Würth Elektronik eiSos GmbH & Co. KG on the basis of ideas, development contracts as well as models or templates that are subject to copyright, patent or commercial protection supplied to the customer will remain with Würth Elektronik eiSos GmbH & Co. KG. Würth Elektronik eiSos GmbH & Co. KG does not warrant or represent that any license, either expressed or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, application, or process in which Würth Elektronik eiSos GmbH & Co. KG components or services are used.

6.8 General terms and conditions

Unless otherwise agreed in individual contracts, all orders are subject to the current version of the "General Terms and Conditions of Würth Elektronik eiSos Group", last version available at *www.we-online.com*.

7 Legal notice

7.1 Exclusion of liability

Würth Elektronik eiSos GmbH & Co. KG considers the information in this document to be correct at the time of publication. However, Würth Elektronik eiSos GmbH & Co. KG reserves the right to modify the information such as technical specifications or functions of its products or discontinue the production of these products or the support of one of these products without any written announcement or notification to customers. The customer must make sure that the information used corresponds to the latest published information. Würth Elektronik eiSos GmbH & Co. KG does not assume any liability for the use of its products. Würth Elektronik eiSos GmbH & Co. KG does not grant licenses for its patent rights or for any other of its intellectual property rights or third-party rights.

Notwithstanding anything above, Würth Elektronik eiSos GmbH & Co. KG makes no representations and/or warranties of any kind for the provided information related to their accuracy, correctness, completeness, usage of the products and/or usability for customer applications. Information published by Würth Elektronik eiSos GmbH & Co. KG regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof.

7.2 Suitability in customer applications

The customer bears the responsibility for compliance of systems or units, in which Würth Elektronik eiSos GmbH & Co. KG products are integrated, with applicable legal regulations. Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of Würth Elektronik eiSos GmbH & Co. KG components in its applications, notwithstanding any applications-related in-formation or support that may be provided by Würth Elektronik eiSos GmbH & Co. KG. Customer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences lessen the likelihood of failures that might cause harm and take appropriate remedial actions. The customer will fully indemnify Würth Elektronik eiSos GmbH & Co. KG and its representatives against any damages arising out of the use of any Würth Elektronik eiSos GmbH & Co. KG components in safety-critical applications.

7.3 Usage restriction

Würth Elektronik eiSos GmbH & Co. KG products have been designed and developed for usage in general electronic equipment only. This product is not authorized for use in equipment where a higher safety standard and reliability standard is especially required or where a failure of the product is reasonably expected to cause severe personal injury or death, unless the parties have executed an agreement specifically governing such use. Moreover, Würth Elektronik eiSos GmbH & Co. KG products are neither designed nor intended for use in areas such as military, aerospace, aviation, nuclear control, submarine, transportation (automotive control, train control, ship control), transportation signal, disaster prevention, medical, public information network etc. Würth Elektronik eiSos GmbH & Co. KG must be

informed about the intent of such usage before the design-in stage. In addition, sufficient reliability evaluation checks for safety must be performed on every electronic component, which is used in electrical circuits that require high safety and reliability function or performance. By using Würth Elektronik eiSos GmbH & Co. KG products, the customer agrees to these terms and conditions.

8 License terms for Würth Elektronik eiSos GmbH & Co. KG sensor product software and source code

This License terms will take effect upon the purchase and usage of the Würth Elektronik eiSos GmbH & Co. KG sensor products. You hereby agree that this license terms are applicable to the product and the incorporated software, firmware and source codes (collectively, "Software") made available by Würth Elektronik eiSos in any form, including but not limited to binary, executable or source code form.
The software included in any Würth Elektronik eiSos sensor product is purchased to you on the condition that you accept the terms and conditions of this license terms. You agree to comply with all provisions under this license terms.

8.1 Limited license

Würth Elektronik eiSos hereby grants you a limited, non-exclusive, non-transferable and royalty-free license to use the software and under the conditions that will be set forth in this license terms. You are free to use the provided software only in connection with one of the products from Würth Elektronik eiSos to the extent described in this license terms.
You are entitled to change or alter the source code for the sole purpose of creating an application embedding the Würth Elektronik eiSos sensor product. The transfer of the source code to third parties is allowed to the sole extent that the source code is used by such third parties in connection with our product or another hardware provided by Würth Elektronik eiSos under strict adherence of this license terms. Würth Elektronik eiSos will not assume any liability for the usage of the incorporated software and the source code.
You are not entitled to transfer the source code in any form to third parties without prior written consent of Würth Elektronik eiSos.
You are not allowed to reproduce, translate, reverse engineer, decompile, disassemble or create derivative works of the incorporated software and the source code in whole or in part.
No more extensive rights to use and exploit the products are granted to you.

8.2 Usage and obligations

The responsibility for the applicability and use of the Würth Elektronik eiSos sensor product with the incorporated software in a particular customer design is always solely within the authority of the customer. Due to this fact, it is up to you to evaluate and investigate, where appropriate, and to decide whether the device with the specific product characteristics described in the product specification is valid and suitable for your respective application or not.
You are responsible for using the Würth Elektronik eiSos sensor product with the incorporated software in compliance with all applicable product liability and product safety laws. You acknowledge to minimize the risk of loss and harm to individuals and bear the risk for failure leading to personal injury or death due to your usage of the product.
Würth Elektronik eiSos' products are not authorized for use in safety-critical applications, or where a failure of the product is reasonably expected to cause severe personal injury or death. Moreover, Würth Elektronik eiSos' products are neither designed nor intended for

use in areas such as military, aerospace, aviation, nuclear control, submarine, transportation (automotive control, train control, ship control), transportation signal, disaster prevention, medical, public information network etc. You shall inform Würth Elektronik eiSos about the intent of such usage before design-in stage. In certain customer applications requiring a very high level of safety and in which the malfunction or failure of an electronic component could endanger human life or health, you must ensure to have all necessary expertise in the safety and regulatory ramifications of your applications. You acknowledge and agree that you are solely responsible for all legal, regulatory and safety-related requirements concerning your products and any use of Würth Elektronik eiSos' products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by Würth Elektronik eiSos. YOU SHALL INDEMNIFY WÜRTH ELEKTRONIK EISOS AGAINST ANY DAMAGES ARISING OUT OF THE USE OF WÜRTH ELEKTRONIK EISOS' PRODUCTS IN SUCH SAFETY-CRITICAL APPLICATIONS.

8.3 Ownership

The incorporated Software created by Würth Elektronik eiSos is and will remain the exclusive property of Würth Elektronik eiSos.

8.4 Disclaimer of warranty

THE SOFTWARE AND IT'S SOURCE CODE IS PROVIDED "AS IS". YOU ACKNOWLEDGE THAT WÜRTH ELEKTRONIK EISOS MAKES NO REPRESENTATIONS AND WARRANTIES OF ANY KIND RELATED TO, BUT NOT LIMITED TO THE NON-INFRINGEMENT OF THIRD PARTIES' INTELLECTUAL PROPERTY RIGHTS OR THE MERCHANTABILITY OR FITNESS FOR YOUR INTENDED PURPOSE OR USAGE. WÜRTH ELEKTRONIK EISOS DOES NOT WARRANT OR REPRESENT THAT ANY LICENSE, EITHER EXPRESS OR IMPLIED, IS GRANTED UNDER ANY PATENT RIGHT, COPYRIGHT, MASK WORK RIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT RELATING TO ANY COMBINATION, MACHINE, OR PROCESS IN WHICH THE WÜRTH ELEKTRONIK EISOS' PRODUCT WITH THE INCORPORATED SOFTWARE IS USED. INFORMATION PUBLISHED BY WÜRTH ELEKTRONIK EISOS REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE A LICENSE FROM WÜRTH ELEKTRONIK EISOS TO USE SUCH PRODUCTS OR SERVICES OR A WARRANTY OR ENDORSEMENT THEREOF.

8.5 Limitation of liability

Any liability not expressly provided by Würth Elektronik eiSos shall be disclaimed.
You agree to hold us harmless from any third-party claims related to your usage of the Würth Elektronik eiSos' products with the incorporated software and source code. Würth Elektronik eiSos disclaims any liability for any alteration, development created by you or your customers as well as for any combination with other products.

8.6 Applicable law and jurisdiction

Applicable law to this license terms shall be the laws of the Federal Republic of Germany. Any dispute, claim or controversy arising out of or relating to this license terms shall be

resolved and finally settled by the court competent for the location of Würth Elektronik eiSos registered office.

8.7 Severability clause

If a provision of this license terms are or becomes invalid, unenforceable or null and void, this shall not affect the remaining provisions of the terms. The parties shall replace any such provisions with new valid provisions that most closely approximate the purpose of the terms.

8.8 Miscellaneous

Würth Elektronik eiSos reserves the right at any time to change this terms at its own discretion. It is your responsibility to check at Würth Elektronik eiSos homepage for any updates. Your continued usage of the products will be deemed as the acceptance of the change.
We recommend you to be updated about the status of new software, which is available on our website or in our data sheet, and to implement new software in your device where appropriate.
By ordering a sensor product, you accept this license terms in all terms.

List of Figures

List of Tables

WÜRTH ELEKTRONIK

more than you expect

Internet of Things

Monitoring & Control

Automated Meter Reading

Contact:
Würth Elektronik eiSos GmbH & Co. KG
Division Wireless Connectivity & Sensors

Max-Eyth-Straße 1
74638 Waldenburg
Germany

Tel.: +49 651 99355-0
Fax.: +49 651 99355-69
www.we-online.com/wireless-connectivity