Product Document

Published by
ams OSRAM Group

Quick Start Guide

QG001029

AS733x SDK

Quick Start Guide

AS733x EVK Data Logger

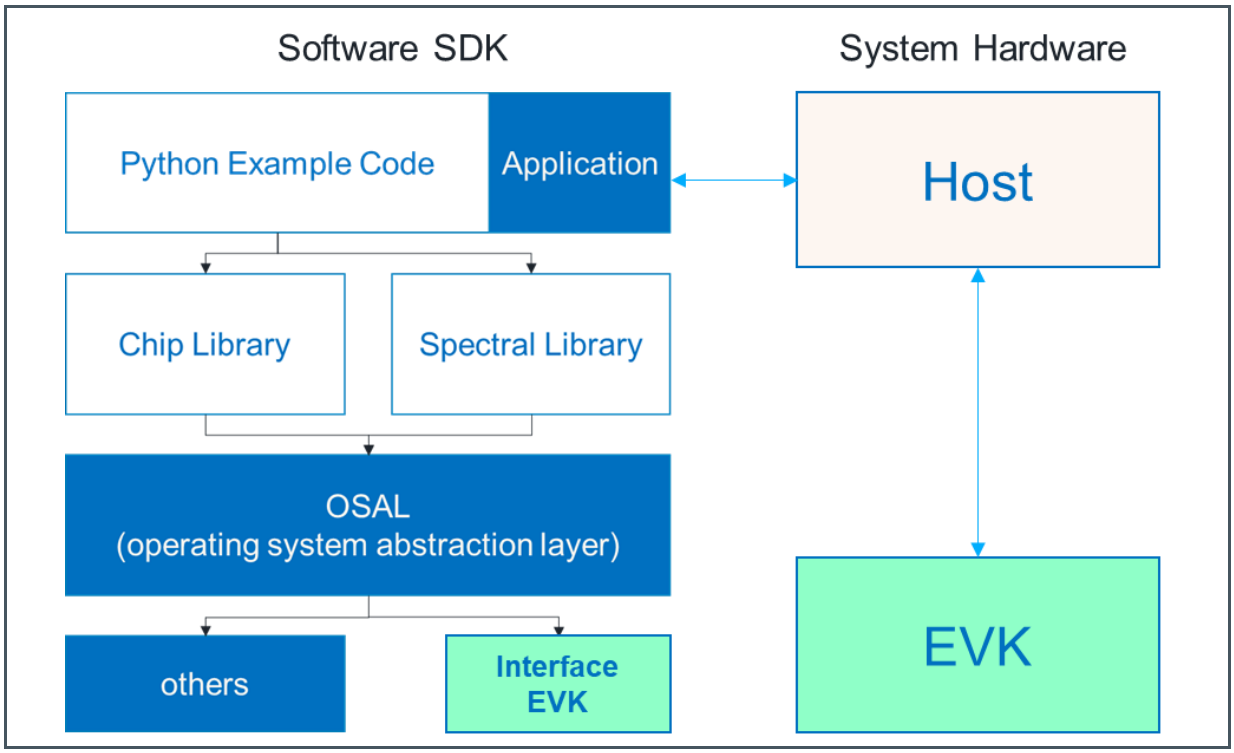
v1-00 • 2022-Aug-12

Content Guide

1 Introduction

The AS733x Software Development Kit SDK (called SDK here) is a collection of software components and libraries written in C programming language, allowing you to quickly prototype solutions using the AS733x sensors on a host, e.g. PC by using a Python example code and the AS733x Evaluation Kit (called EVK here).

Figure 1:
Structure of the AS733x Source Development Kit



The SDK shows an example of programming and controlling the AS733x sensor via ChipLib, which can be extended by a customer and application-specific software. A spectral library with pre-defined algorithms and data - to correct and map the sensor results into UV applications - is part of the SDK and can be the basis of the application-specific software.

A hardware extension of the SDK would be also possible by adapting the Operating System Abstraction Layer, OSAL (see chapter 7). The OSAL and SDK are prepared to work as a plug-and-play system in conjunction with the AS733x EVK and LEDs on the sensor board.

This document provides an overview of its software components and explains their usage.

2 System Requirement

An EVK or Demo Board with an AS733x sensor is required to use the AS733x Software Development Kit. Running the provided sample code requires a 64-bit Windows computer, which is connected to the board using a USB cable.

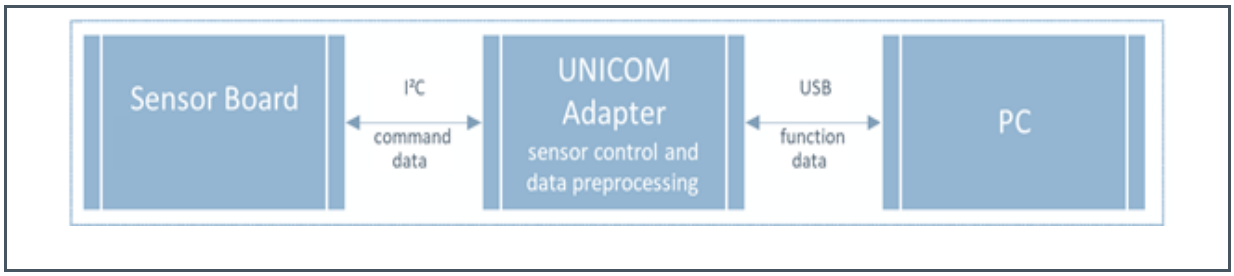
Other system components are possible, but require adaptations described later in this document (see chapter 7).

With the libraries and SDK, a concept is implemented that allows the sensor to be used with the libraries in any hardware. The user only has to adapt the OSAL and the source code to their target system and program their application. The libraries can be used identically everywhere and already contain all the functions of the sensor for control and many functions for correction and application.

2.1 AS733x EVK

The user manual [1] describes the AS733x Evaluation Kit. The datasheet [2] contains all the technical details for the sensor chip. It is recommended to study both documents before using the SDK, as this document only contains details on the SDK and not the sensor or EVK. The SDK includes hardware and software parts. The hardware consists of a sensor board, an interface board, together in a plastic case, connected via USB to a PC.

Figure 2:
AS733x SDK Hardware Solution



The SDK software tools are source codes a customer can modify.

2.2 AS733x SDK Software and Tools

The AS733x SDK includes the following components in the single subfolders:

- **bin**

This directory contains a collection of the tools necessary to convert and link the codes and libraries into an executable file (see chapter 4).

- **build**

This represents the `build` directory for the project results after compiling the source code with the libraries (see chapter 6).

- **chiplib**

The AS733x Chip Library implements a driver for AS733x devices. It handles communication with the device and is used to configure the device and perform measurements. In general, this software component is used as a software library and black box. The functional and interface description is part of the documents in the specified directories. The source code can be provided.

- **docu**

This directory contains this quickstart manual with other documents.

- **sample_code**

This current directory contains sample codes using the software components provided in the AS733x Software Development Kit. All examples use the AS733x Chip Library to acquire raw measurement data from the sensor. The example named “as733x_sample_code_chiplib” prints the raw data directly, while the example named “as733x_sample_code_spectral_lib” post-processes the data using the AS733x Spectral Library. The sample code programs are intended to be executed directly on a Windows computer. All examples are provided as source code and executables. When executing the sample code, all software components run on a computer - the microcontroller on the board is only used as a bridge for I²C communication and to signal interrupts from the sensor.

- **spectral_library**

The AS733x Spectral Library is a post-processing library that corrects the sensor results and calculates UV light intensity information based on raw data measured using an AS733x sensor and calibration data. This software component is provided as source code.

- **utilities**

This directory contains a helper code used by the other software components, such as error code definitions. It is provided as a source code.

Other project-oriented files are in the main directory.

For information on the interface of an individual software component, please refer to the documentation of the software component, which can be found in the directory of the software component in the AS733x Software Development Kit.

3 Operating System Abstraction Layer (OSAL)

The software components of the AS733x Software Development Kit are designed to be used on different platforms. Since the interface to access I²C buses, GPIO pins, and timer ticks differ per platform, each software component using such resources contains an Operating System Abstraction Layer (OSAL), which contains the platform-specific code to access these resources. When porting a software component to a custom platform, only the corresponding OSAL needs to be adopted. All other source codes of the software component can be left unmodified.

For the sample code provided, the AS733x Software Development Kit contains an OSAL library, which implements the OSAL for the AS733x Chip Library. The OSAL library, which is only available as a binary, uses the board as a bridge for I²C communication, and to signal interrupts from the sensor. It can be found in the `chiplib\bin\x86-64\windows` directory.

The OSAL description and functional dependencies can be found in the ChipLib documentation `\chiplip\as733x_chiplib_doc_v0.2.1.pdf`.

Chapter 7 shows an adaptation of the OSAL in the case of modifying custom-specific hardware.

4 Sample Codes

In the SDK are pre-designed sample codes in the `sample_code` directory. Each of the examples contained in the AS733x Software Development Kit is provided as pre-build executables. They can be found in the `sample_code\bin\x86-64\windows` directory as well as the CSV initialization files.

Figure 3:
sample_code\bin\x86-64\windows Directory

```
C:\tmp\as733x_sdk_v1.0.0_test\sample_code\bin\x86-64\windows>dir
 Volume in drive C has no label.
 Volume Serial Number is C229-FD67

 Directory of C:\tmp\as733x_sdk_v1.0.0_test\sample_code\bin\x86-64\windows

21.02.2022  14:05    <DIR>          .
21.02.2022  14:05    <DIR>          ..
15.02.2022  14:48           367.609 as733x_osal_corefw.dll
15.02.2022  14:48           439.986 as733x_sample_code_chiplib.exe
15.02.2022  14:48           442.947 as733x_sample_code_spectral_lib.exe
15.02.2022  14:47               278 as733x_sample_code_spectral_lib_appl_calib_template.csv
15.02.2022  14:47               545 as733x_sample_code_spectral_lib_sensor_calib_template.csv
               5 File(s)      1.251.365 bytes
               2 Dir(s)  251.677.569.024 bytes free
```

Each sample demonstrates the data and library, including definitions, global and local library functions, sensor configurations and initializations, as well as sensor measurements and alternative post-processing:

(A) `as733x_sample_code_chiplib.exe` shows how the AS733x Chip Library can be used to obtain ADC values from the sensor.

(B) `as733x_sample_code_spectral_lib.exe` shows how the AS733x Chip Library can be used to obtain ADC values from the sensor and how the acquired data can be post-processed using the AS733x Spectral Library. The required sensor correction and application-specific calibration parameters are read from CSV files.

The CSV initialization files include all the required sensor and application-specific parameters to correct sensor deviations (used in the ChipLib) or to map sensor results into application-specific units (used in the SpectralLib).

The following initialization files are defined as an example for the "*as733x_sample_code_spectral_lib*":

- `as733x_sample_code_spectral_lib_sensor_calib_template.csv` includes the sensor initialization and correction data.

Figure 4:
Example of the Initialization File for the Sensor Calibration

```
ref_temp,20000
ref_gain,2
ref_time,100
ref_divider_value,1
ref_divider_enable,1
poly_coeff_for_temp,5.3,4.1
poly_coeff_for_temp_counts,2.5,2.6,3.7,3.8,3.9,4.0,4.7,4.8,5.9,5.3,6.4,7.5,7
poly_coeff_for_temp_offset,1.3,1.1,1.2,1.3,1.4,2.8,2.5,2.6,2.8,3.8,3.4,3.3,4
offset_basic_counts,2.3,3.4,4.5
balancing_coeffs,3.1,3.2,3.3,4.4,4.5,5.6,6.7,6.8,7.9
gain_corr,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9,2.0,2.1,2.2,2.3,2.4,2.5,2.6,2.
```

- `as733x_sample_code_spectral_lib_appl_calib_template.csv` includes the application-specific calibration data.

Figure 5:
Example of the Initialization File for the Sensor Mapping

```
description, use case A
calib_matrix,1.1,1.2,1.3,1.4,1.5,1.6,2.7,2.8,2.9
poly_coeff_per_channel,2.1,2.2,2.3,2.4,3.4,3.5,4.6,4.7,5.7
poly_coeff_combined,2.3,2.3,2.5
channel_combination_coeff,1.7,1.4,2.5
threshold,2.1,4.1
application_coeff,6.1,6.6,7.8,8.9,9.3,10.6,12.3,13.5,15.1
```

Note that these initialization files are only templates and the suitable parameters must be determined individually for each sensor/LOT and application. It is recommended to make your application tests with the EVK to find out the optimized parameters for settings and mapping. Then, these parameters can be used in the initialization files and SDK. The used keywords and formats in these files are fixed. The meaning of the control words or the parameter lists is, in most cases, self-explanatory or is described in the library documents "*as733x_spectral_lib_doc_v1.0.0.pdf*" or "*as733x_chiplib_doc_v0.3.0.pdf*" (located in the chiplib or spectral_lib directories).

5 Executing the Sample Code in Windows

Figure 6 and Figure 7 show examples of calls with the parameters and results on the screen.

```
CALL Example (A)>as733x_sample_code_chiplib.exe --interface COM:COM47
```

Figure 6:
Sample (A) Call and Results on the Screen

```
C:\tmp\as733x_sdk_v1.0.0 test\sample_code\bin\x86-64\windows>as733x_sample_code_chiplib.exe --interface COM:COM47
Press Ctrl-C to stop the measurement.

ADC Values: 174, 113, 231; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.60 C
ADC Values: 172, 110, 229; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.65 C
ADC Values: 172, 110, 227; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.70 C
ADC Values: 170, 109, 226; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.70 C
ADC Values: 172, 110, 227; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 170, 108, 225; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 170, 108, 223; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.70 C
ADC Values: 168, 107, 222; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 169, 108, 224; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 170, 108, 224; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 168, 107, 223; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 169, 108, 224; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.75 C
ADC Values: 167, 107, 221; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.80 C
ADC Values: 167, 107, 222; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.80 C
ADC Values: 168, 107, 222; Gain: 2048x; Integration Time: 1024000 us; Temperature: 24.80 C
```

Each of the examples contained in the AS733x Software Development Kit is provided as pre-build executables. They can be found in the `sample_code\bin\x86-64\windows` directory, where the calibration files for sample (B) are also located.

```
CALL Example (B)>as733x_sample_code_spectral_lib.exe --interface COM:COM47 --
sensor_calib as733x_sample_code_spectral_lib_sensor_calib_template.csv --
appl_calib as733x_sample_code_spectral_lib_appl_calib_template.csv
```

Figure 7:
Sample (B) Call and Results on the Screen

```
U:\AMS\AMS products\AS7331 UV\02c_firmware\as733x_sdk_v1.0.0_test\sample_code\bin\x86-64\windows>as733x_sample_code_spectral_lib.exe --interface COM:COM47 --sensor_calib as733x
_sample_code_spectral_lib_sensor_calib_template.csv --appl_calib as733x_sample_code_spectral_lib_appl_calib_template.csv
Press Ctrl-C to stop the measurement.

Combined Regression: 5145350.71997, Regression Channel 0: 132894.05105, Regression Channel 1: 225544.96683, Regression Channel 2: 407606.17101, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.78508, Regression Channel 0: 132894.05274, Regression Channel 1: 225544.96969, Regression Channel 2: 407606.17617, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
Combined Regression: 5145350.84859, Regression Channel 0: 132894.05438, Regression Channel 1: 225544.97248, Regression Channel 2: 407606.18121, Application Output: 19.568787
```

To run the pre-built examples, follow these general steps:

1. Connect the AS733x EVK via USB to the PC.
2. Identify the serial port (also known as COM port) assigned to the board. This information can be found in the Device Manager. The board shows up as `USB Serial Device (COMx)` (with `x` being an integer) in the `Ports (COM & LPT)` section of the Device Manager. The text within the parentheses represents the serial port assigned to the board, for example `COM47`.

Information

The Device Manager can be opened by pressing the keys “Windows key+R” on the keyboard. In the newly opened window, type “devmgmt.msc” and click OK.

3. Open a Command Prompt and navigate to the `sample_code\bin\x86-64\windows` directory within the AS733x Software Development Kit.

Information

The Command Prompt can be opened by navigating to the extracted `sample_code\bin\x86-64\windows` directory in Explorer. Afterward, type `cmd` in the address bar of Explorer and press Enter. This opens a Command Prompt that has already navigated automatically to the required directory.

4. In the Command Prompt, insert the file name of the sample code executable you want to execute, followed by the arguments required by the program. Ensure to insert a space character before each argument. Afterwards, press Enter to run the sample code. All programs require the serial port of the board to be provided via an argument using the format --interface COM:COMx. For the “*as733x_sample_code_spectral_lib.exe*” program, the paths to the CSV file containing the sensor and application calibration data must be provided via the additional arguments “–-sensor_calib PATH1” and “--appl_calib PATH2” (replace PATH1 and PATH2 with the actual paths and names of the CSV files – see Figure 6 and Figure 7). The CSV files must be conform to the format required by the AS733x Spectral Library.
Template CSV files named “*as733x_sample_code_spectral_lib_sensor_calib_template.csv*” and “*as733x_sample_code_spectral_lib_appl_calib_template.csv*” can be found in the sample_code directory of the AS733x Software Development Kit.
Example: Assuming the “*as733x_sample_code_chiplib.exe*” is to be executed and the serial port COM47 has been assigned to the board, `as733x_sample_code_chiplib.exe --interface COM:COM47` must be entered in the Command Prompt.
5. Afterward, the output of the sample code is printed to the Command Prompt.
6. Lastly, press the keys Ctrl-C to stop the sample code.

6 Building the Sample Code

The AS733x Software Development Kit contains a Make file to build the sample code executable from the provided source code and libraries. A GCC-based build environment suitable for building Windows binaries needs to be installed on your computer.

There are multiple ways to install GCC on Windows systems. This section briefly explains how GCC can be installed using the Mingw-w64 project and how this GCC installation can be used to build the sample code. However, any other suitable GCC installation can alternatively be used to build the sample code.

1. To install GCC via Mingw-w64, download the Mingw-w64 installer.
2. Then, execute the installer. When prompted, select the following settings:

- Version: 8.1.0
- settings: Version: 8.1.0
- Architecture: x86_64
- Threads: posix
- Exception: seh
- Build revision: 0

3. After completing the installation, add the subdirectory "*mingw64/bin*" of your Mingw-w64 installation to the `PATH` environmental variable of your system or user.
   Example: If you select `C:\Program Files\mingw-w64\x86_64-8.1.0-posix-seh-rt_v6-rev0` as the destination folder during installation, the directory `C:\Program Files\mingw-w64\x86_64-8.1.0-posix-seh-rt_v6-rev0\mingw64\bin` must be added to the `PATH` environmental variable.

Information

Perform the following steps to add a PATH environmental variable:

a. Open the start menu, enter "Edit environment variables for your account" in the search bar, and select the same result.
b. In the upper part of the window ("User variables for x"), double-click on the "Path" entry.
c. Then, add the installation path using the "New" button.
d. Afterwards, close all the windows with the OK button.
e. Finally, close the command prompt and open it again.

4. To build the sample code, navigate to the root directory of the extracted AS733x Software Development Kit in a Command Prompt and run the command `mingw32-make CC=gcc all`. After building is completed successfully, the executables can be found in the newly created `build` directory of the AS733x Software Development Kit. The file names of all executables built from source code include the suffix "*_source*".
   Example: `mingw32-make CC=gcc AS733x_sample_code_uv` calls the compiler to make an executable file using the C++ compiler.
5. Start and test the build of the new executable file. Take a note of the given details from chapter 4 for executing, e.g. .directories, interfaces, and calibration files.

7 Customizing the OSAL and Example Code

When porting the AS733x Chip Library to a customer-specific platform (hardware and/or software), the corresponding OSAL needs to be implemented for the given platform.

For the AS733x Chip Library, a template named `AS733x_osal_chiplib_template.c` is provided in the `chiplib\src` directory of the AS733x Software Development Kit. The OSAL template file can be used as a starting point when implementing a custom OSAL. Anywhere the insertion of platform-specific code is required is marked with `TODO`.

7.1 Including the Chip Library in a Custom Application

To include the AS733x Chip Library with the customized OSAL in an application, the OSAL implementation and the following files of the AS733x Software Development Kit must be copied to the target project:

- All the sources files of "*chiplib\src*", except "*as733x_osal_chiplib_template.c*".
- All the header files of "*chiplib\inc*".
- All the header files of "*chiplib\inc\internal*".
- "*error_codes.h*" and "*std_inc.h*" of "*utilities\inc*".

When compiling the source files, the directories containing the header files must be added to the "include" search paths of the compiler. Additionally, the compiler must be used to define "NUM_SUPPORTED_DEVICES" to the number of AS733x Chip Library instances.

Example: If a single instance of the AS733x Chip Library is used, "NUM_SUPPORTED_DEVICES" will be defined as "1". When using GCC, this can be achieved by passing the option "DNUM_SUPPORTED_DEVICES=1" to the compiler.

7.2 Including the Spectral Library in a Custom Application

To include the AS733x Spectral Library in an application, the following files of the AS733x Software Development Kit must be copied to the target project:

- All the sources files of "*spectral_lib\src*".
- All the header files of "*spectral_lib\inc*".
- All the header files of "*spectral_lib\inc\internal*".
- All the sources files of "*utilities\src*".
- All the header files of "*utilities\inc*".

When compiling the source files, the directories containing the header files must be added to the "include" search paths of the compiler.

8 Additional Documents

The following list includes a selection of available documents with more technical details for the AS733x SDK. This list is not fixed and it is constantly changing. Ask ams OSRAM for new details.

For further information, please refer to the following documents:

1. ams-OSRAM AG, *AS7331 EVK Logger (UG001037)*, User Manual.
2. ams-OSRAM AG, *AS7331 Spectral UVA/B/C Sensor (DS001047)*, Datasheet.

Headquarters
ams-OSRAM AG
Tobelbader Strasse 30
8141 Premstaetten
Austria, Europe
Tel: +43 (0) 3136 500 0

Please visit our website at www.ams.com
Buy our products or get free samples online at www.ams.com/Products
Technical Support is available at www.ams.com/Technical-Support
Provide feedback about this document at www.ams.com/Document-Feedback
For sales offices, distributors and representatives go to www.ams.com/Contact
For further information and requests, e-mail us at ams_sales@ams.com