TLE9893_2QTW62S_AES_GCM

About this document

Scope and purpose

The aim of this guide is to present the scope, the implementation, the algorithm and a demonstration of the **TLE9893_2QTW62S_AES_GCM** example code for the TLE989x Infineon Embedded Power ICs based on Arm® Cortex® M3. This example code can be found in the Keil µVision Pack Installer.

The full functionalities and characteristics of the embedded power devices are described in the datasheets and user's manual. Please refer to these documents for more detailed information. Furthermore, a low level (line-by-line) description of the code is not the aim of this document, although occasionally some codeblocks might be reported if necessary to the comprehension.

Note: *The following information is given as a hint for the implementation of the system only and shall not be regarded as a description or warranty of a certain functionality, condition or quality of the referred devices or presented software example.*

Intended audience

Design engineers, system engineers, embedded power designers

Table of contents

1 Introduction

The example computes the AES GCM tag for the given data, initialization vector and authenticated data. The calcuted tag is printed on UART. Figure 1 shows output. Note: Only 96bits of IV is supported in the current example.

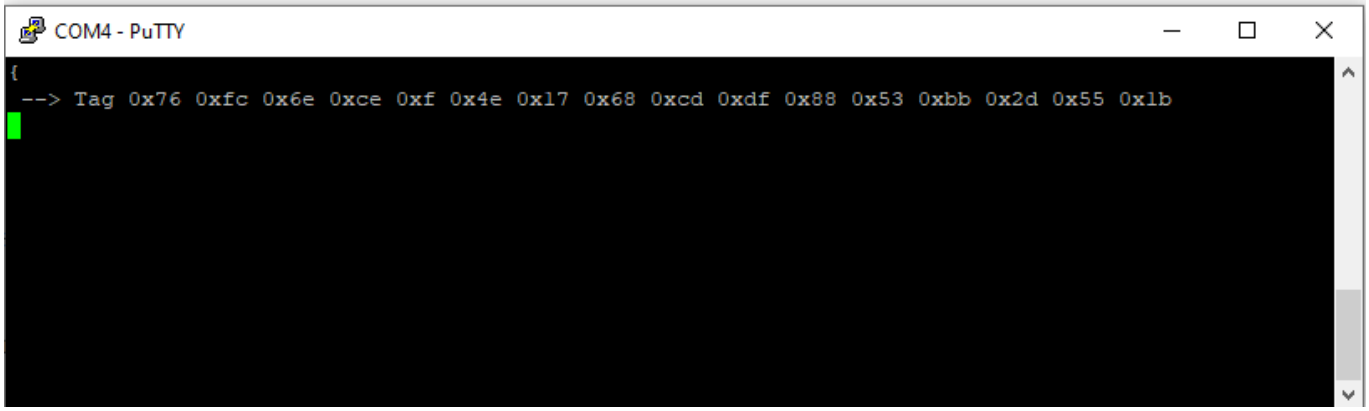

Figure 1 UART output

Note:
Above Tag is computed using the **test vector 16** from **gcm-spec** document in __documents folder of this example. And it is not computed using the default key but with one mentioned in the test vector.
This key can writen to the device with the example **SECURITY_WRITE_KEY_EXAMPLE_TLE989X.**

2 Hardware

This chapter shows how to run the TLE9893_2QTW62S_AES_GCM example with the TLE988x/TLE989x evaluation board. For this the project must be opened and compiled.

Figure 2 shows the TLE988x/TLE989x evaluation board. Theapplication code must be loaded via a debugger (e.g. ULINK or J-Link) to the board. The board must be powered with 12V (red and black connections).

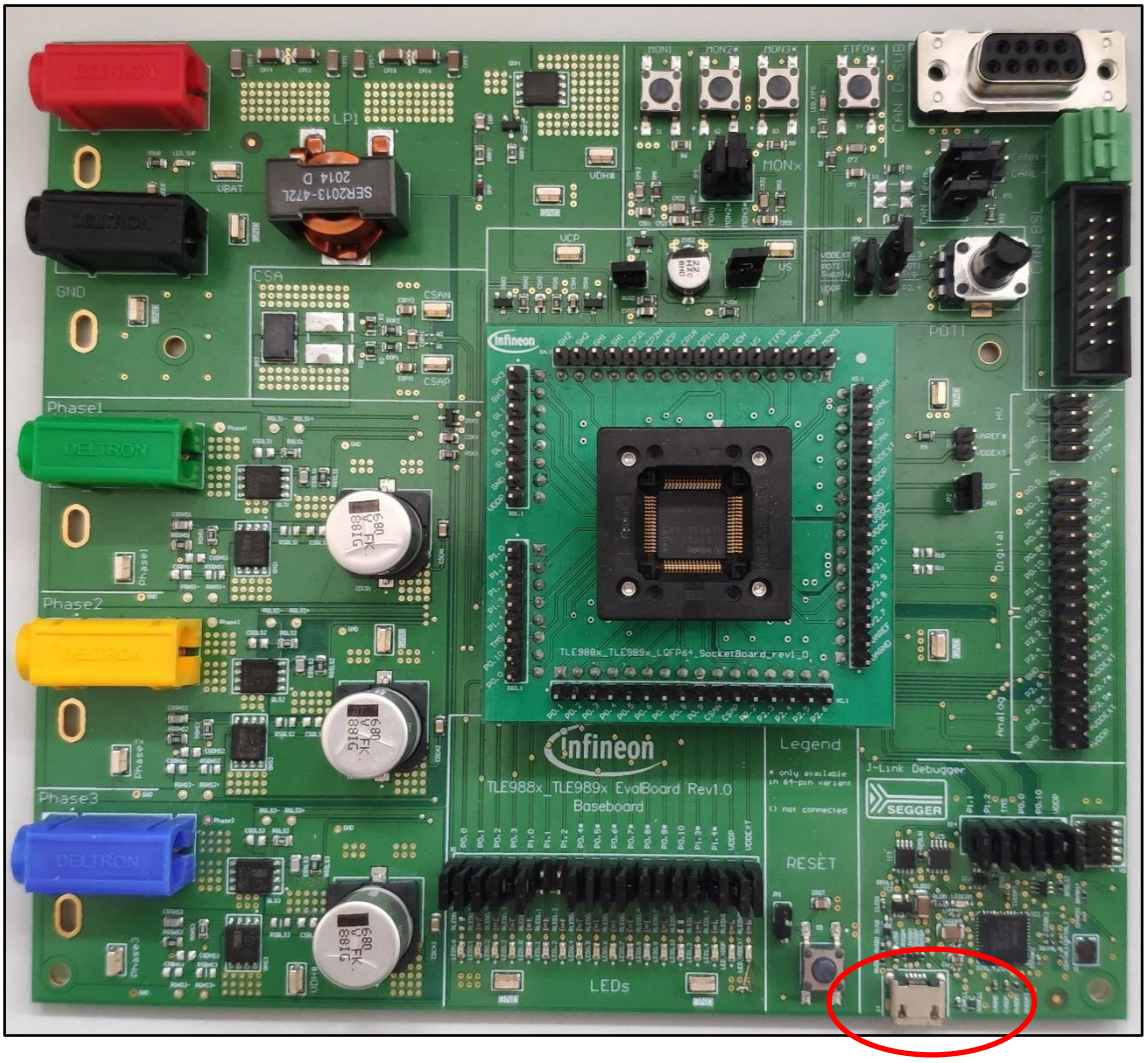

Figure 2 TLE988x/TLE989X evaluation board

Alternatively, a USB connection can be established to a local PC, which emulates a virtual COM port. The relevant COM device number can be identified via the Device Manager on Windows systems or the dmesg tool on Unix based operating systems

In order to show the output on a command console, free tools like Putty or TeraTerm can be used. The UART1 in this example is configured with:

- a transmission baud rate of 115200,
- 8 data bits,
- 1 stop bit,
- no parity and no flow control.

3 Implementation

This chapter shows the process to follow to get a working secure access simple example.

3.1 Get the example via the Pack Installer for Keil

Open the Pack Installer within the Keil IDE.

Choose the appropriate device (here TLE9893_2QKW62S) on the left-hand side. On the right-hand side, select the tab Examples, where you can access the TLE98932QKW62S_AES_GCM example.

Clicking on “Copy” will copy the example on your computer and open it.

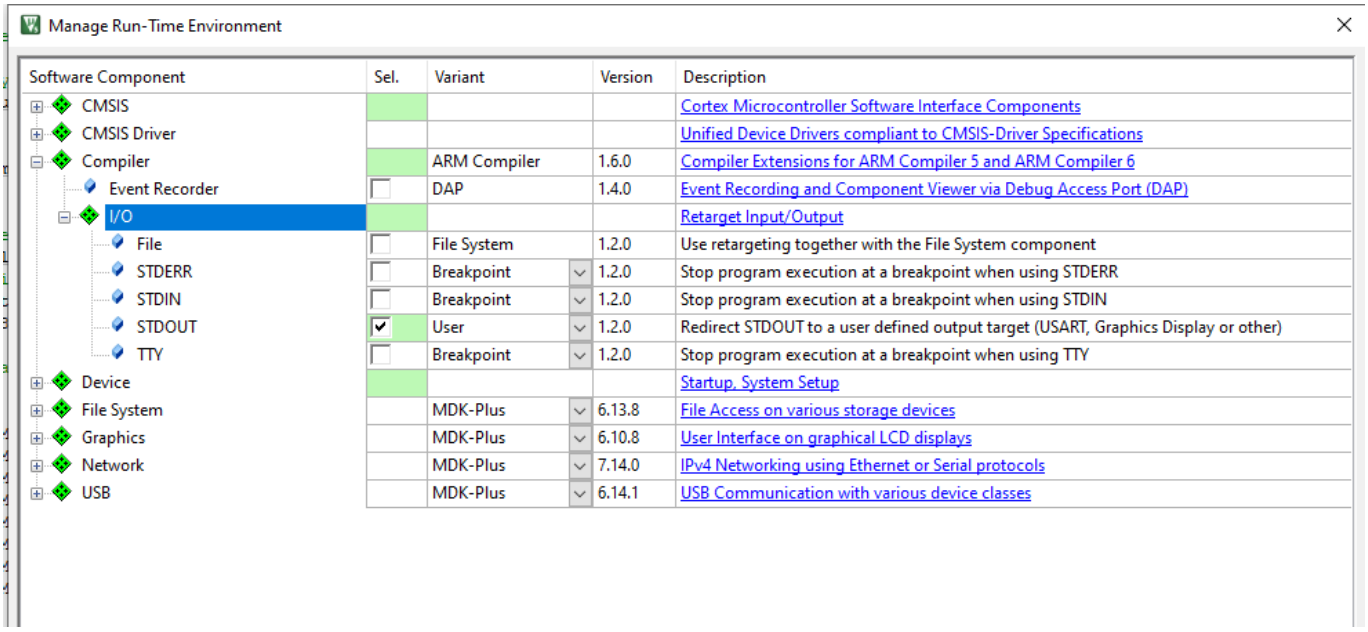

Figure 3 RTE settings for stdout and stdin

In order to redirect the stdout functions - the printf call in the example, adjust the runtime environment setting for the compiler within the Keil IDE. Select the option “User” under Compiler -> I/O -> STDOUT (see Figure 4).

3.2 Configuration

In order to configure the UART module for the TLE9893_2QKW62S_SECURE_ACCESS example, select the UART tab. Enable the UART1 module. Next, select the 8-bit UART mode with variable baudrate. The baudrate is set to 115200 in the blue box Baudrate Generator Settings. This is one of the common speed settings for the UART. In the pink box Transmission Settings, select the pin P1.1. See Figure 4 Config Wizard, module UART for more details

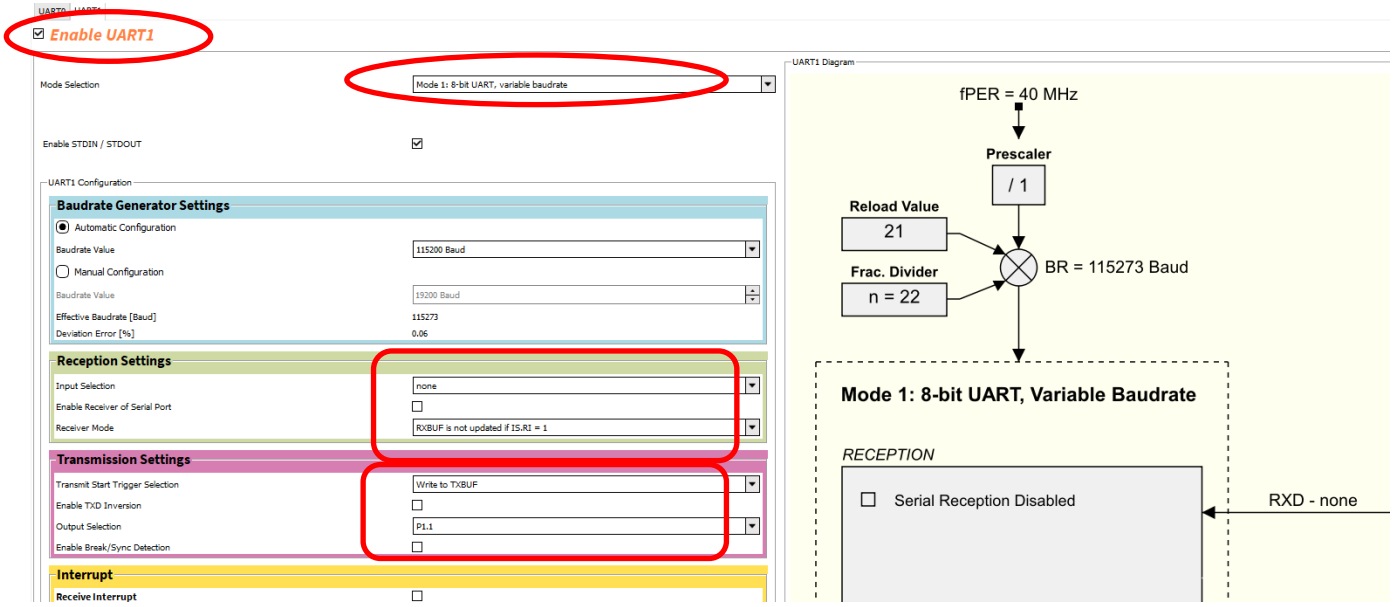

Figure 4 Config Wizard configuration

Finally, save your configuration to take these changes into account (File -> Save).

3.3 Sample code explanation for AES-GCM

The steps for performing AES-GCM operation is as follows.

Step 1: Compute the hash key using *execAES* function, H=E(k,0^128).

Step 2: Use the initialization vector as the input to the counter and increment it by 1.

Step 3: Check if the input data length is multiple of 16, if not make it multiple of 16.

Step 4: Perform AES encryption for the counter value.

Step 5: Perfrom xor operation of input data and Step 4 out put. This is the AES-GCM ecrypted output.

Step 6: Increment the counter and repeat Step 4, Step 5 and Step 6 untill all input data are encrypted.

Step 7: Compute the GHASH value.

Step 8: Reset the counter value with IV, then increment the counter. Perform AES encryption using *execAES* function.

Step 9: Calculate the authenticated Tag by xoring the ghash value and Step 8 value.

References

See the code examples at **www.infineon.com**

Revision history

Document version	Date of release	Description of changes
1.0	2021-11-04	Initial version
1.1	2022-10-13	Editorial changes

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2022-10-13

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

IMPORTANT NOTICE

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.