AN12119

A71CH Quick start guide for OM3710A71CHARD and i.MX6UltraLite

Rev. 1.1 — 7 March 2018
458211

Application note
COMPANY PUBLIC

Document information

Info	Content
Keywords	Security IC, i.MX6UltraLite, OM3710A71CHARD, MCIMX6UL-EVKB
Abstract	This document provides a detailed guide for getting started with OM3710A71CHARD and i.MX6UltraLite board.

Revision history

Rev	Date	Description
1.0	20180220	First release
1.1	20180302	Updated figure 5

Contact information

For more information, please visit: http://www.nxp.com

1. Introduction

This document explains how to get started with the OM3710A71CHARD development kit and the i.MX6UltraLite board. It gives an overview of the hardware and describes the board configuration options. It also gives step by step instructions to set up the software development environment as well as full directions to run the example application in a Linux platform using the i.MX6UltraLite evaluation board (MCIMX6UL-EVKB).

2. A71CH overview

The A71CH is a ready-to-use solution enabling ease-of-use security for IoT device makers. It is a secure element capable of securely storing and provisioning credentials, securely connecting IoT devices to public or private clouds and performing cryptographic device authentication.

The A71CH solution provides basic security measures protecting the IC against many physical and logical attacks. It can be integrated with various host platforms and operating systems to secure a broad range of applications. In addition, it is complemented by a comprehensive product support package, offering easy design-in with plug & play host application code, easy-to-use development kits, documentation and IC samples for product evaluation.

3. System description

The A71CH evaluation setup presented in this document consist of an A71CH security IC connected to an i.MX6UltraLite host MCU (MCIMX6UL-EVKB) through the OM3710/A71CHARD Arduino compatible kit. The i.MX6UltraLite is controlled by a development PC over USB, and the A71CH IC communicates with the i.MX6UltraLite over I²C protocol Fig 1 shows a basic diagram of the system architecture.

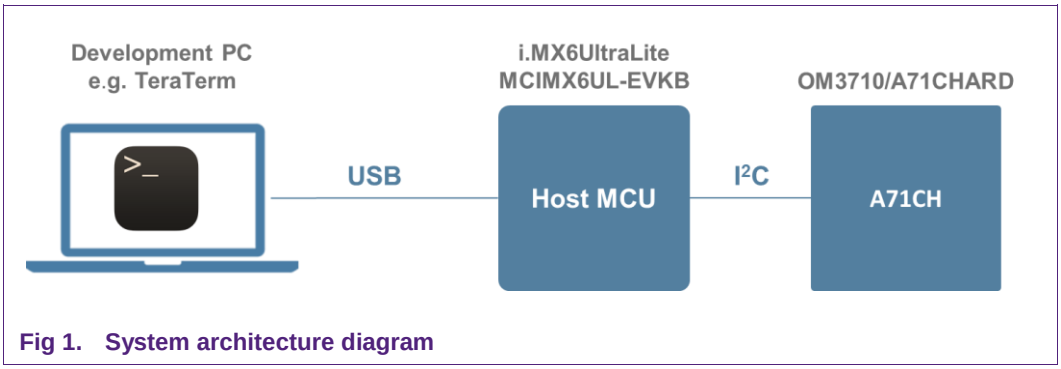


Fig 1. System architecture diagram

This getting started guide is divided in four parts:

- **Hardware overview and setup**: It describes the i.MX6UltraLite host MCU evaluation board (MCIMX6UL-EVKB), the A71CH Arduino compatible kit (OM3710/A71CHARD) and how to mount them together.
- **Software setup**: It describes the preparation of the i.MX6UltraLite Linux image, drivers and terminal application.
- **A71CH application examples execution**: It describes how A71CH application examples included in the A71CH Host software package can be run.

4. Hardware overview

This setup uses the i.MX6UltraLite contained in the MCIMX6UL-EVKB as a host MCU while the A71CH security IC acts as the protected storage module. The following two boards are needed:

1. A71CH Arduino compatible development kit (OM3710/A71CHARD).
2. i.MX6UltraLite evaluation board (MCIMX6UL-EVKB).

4.1 A71CH Arduino compatible development kit (OM3710/A71CHARD)

The OM3710/A71CHARD is an Arduino development kit containing two items as well:

1. An A71CH Mini PCB board (OM3710/A71CHPCB)
2. An Arduino interface board, allowing the user to connect the A71CH to any host featuring an Arduino compatible header (e.g. many LPC, Kinetis and i.MX boards in the industry).

4.1.1 A71CH Mini PCB board (OM3710/A71CHPCB)

The OM3710/A71CHPCB board is a small PCB containing the A71CH solution and a set of jumpers for the I²C or SPI host interface selection (Note that only the I²C driver is available, SPI support might be added in future revisions).

Fig 2 shows an image of the MiniPCB. It features two connectors that can be used depending on which communication interface is used. The figure shows the jumpers configuration that enables the use of the A71CH I²C interface.

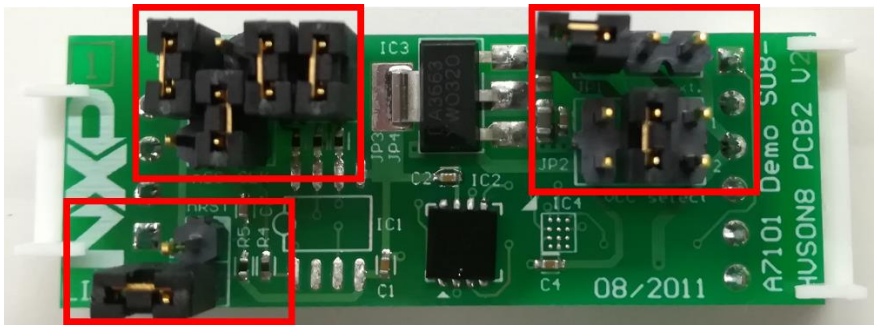


Fig 2. A71CH mini PCB OM3710/A71CHPCB

To enable the I²C communication protocol, it is necessary to configure JP5/6 according to Table 1. JP2 connects the A71CH to the on-board 3.3V voltage regulator on the MiniPCB board. The jumpers JP3 and JP4 enable the I²C SDA/SCL pull-up resistors. JP7 can be used to connect the A71CH reset signal.

Table 1. Default OM3710/A71CHPCB Jumper settings

Jumper	Setting	Usage
JP1	Not set	External VCC connection
JP2	3-4	Connect A71CH to 3.3V regulator on MiniPCB
JP3	Set	Connect I²C SDA pull-up resistor

Jumper	Setting	Usage
JP4	Set	Connect I²C SCL pull-up resistor
JP5	1-2	Use I²C address 0x92/0x93
	2-3 (Default)	Use I²C address 0x90/0x91
JP6	1-2	Activate I²C interface
JP7	Not set (Default)	A71CH operates
	Set	A71CH IC reset

The board schematic and layout are shown in Fig 3 and Fig 4.

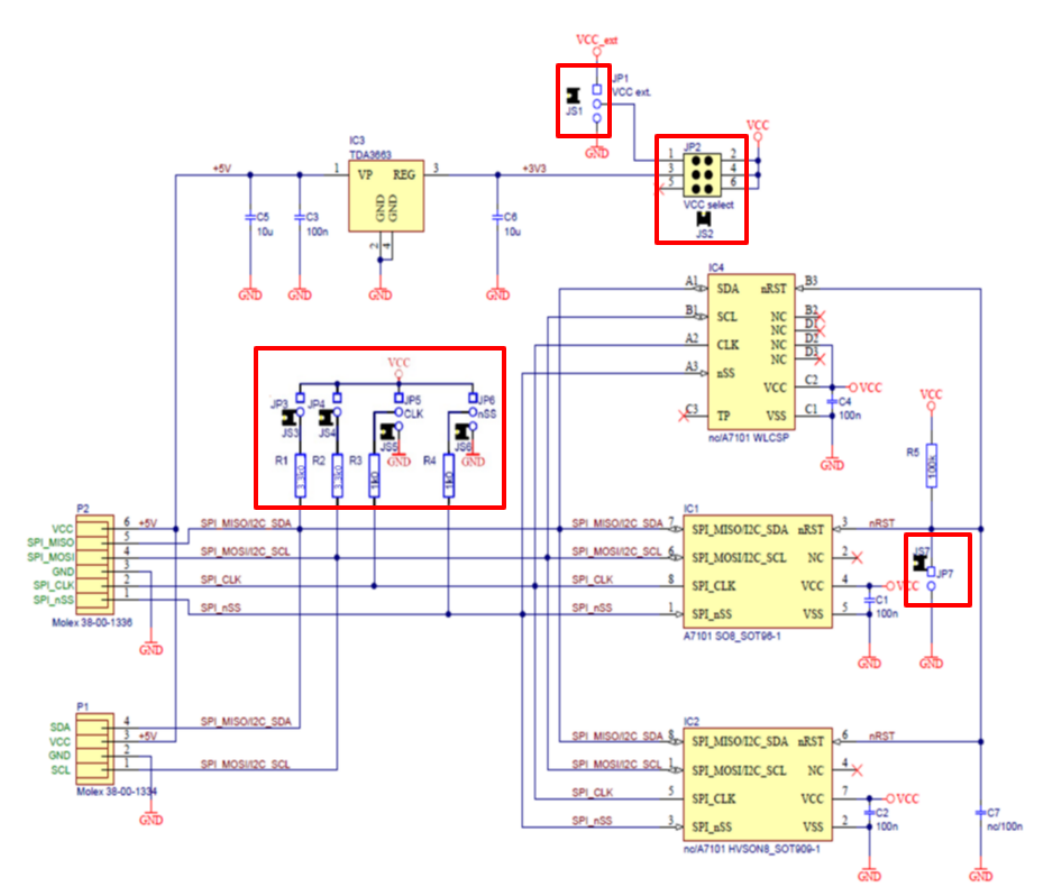


Fig 3. OM3710/A71CHPCB board schematic

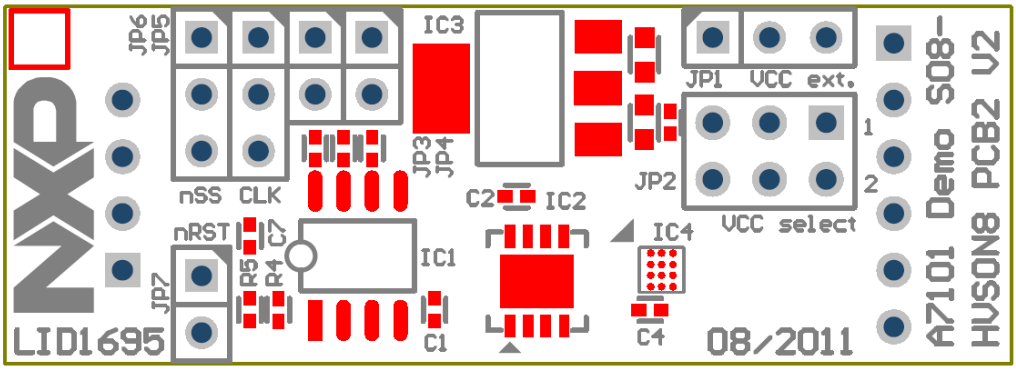
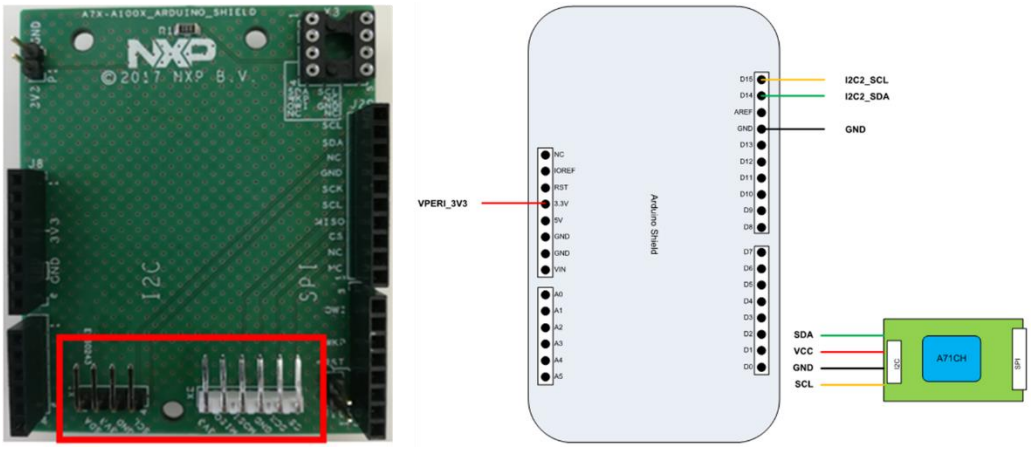


Fig 4. OM3710/A71CHPCB board silkscreen

4.1.2 Arduino interface board

The Arduino header board permits the user to interface the A71CH OM3710/A71CHPCB with the MCIMX6UL-EVKB i.MX6UL motherboard. Fig 5 shows the board pinout.



(1) Connection from the perspective of a matching Arduino Shield (top view)

Fig 5. A71CHARD Arduino header

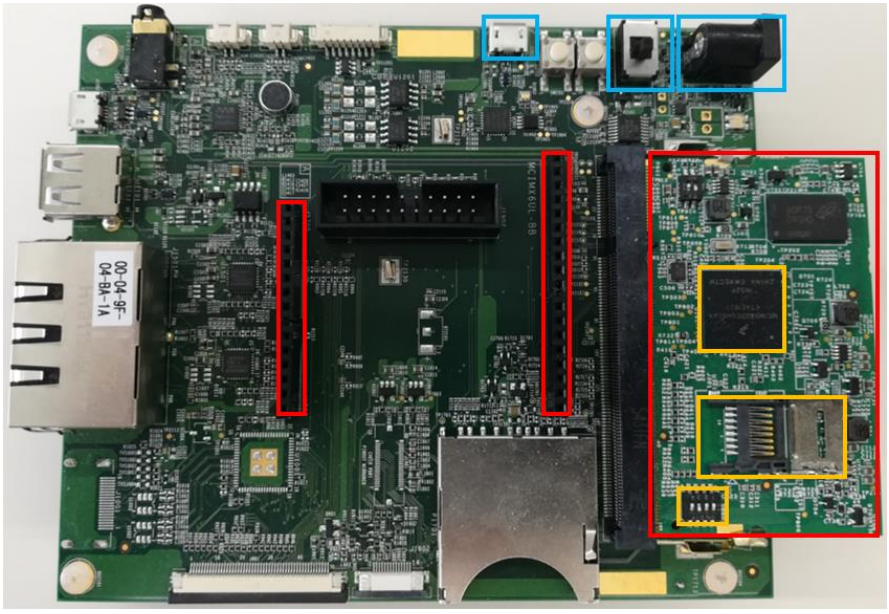
In addition, the A71CHARD provides dedicated male connectors to mount the A71CHPCB via I²C or SPI without any hardware modification.

4.2 i.MX6UltraLite evaluation kit (MCIMX6UL-EVKB)

The i.MX6UltraLite evaluation kit contains a MCIMX6UL-EVKB2 Motherboard and a MCIMXUL-EVKB Daughterboard. Fig 6 shows an image of the complete evaluation kit. The daughterboard is highlighted with a red rectangle. The Arduino header connector is also highlighted in red.

The motherboard provides the system with several communication interfaces (CAN, USB, Ethernet, I^2C, SPI), a set of sensors (gyroscope, accelerometer, magnetometer), peripherals (camera, microphone, headphone output, speaker out connectors) and a display board interface to connect either an LCD screen or an HDMI interface. Additionally, the motherboard features two debug interfaces: 20-pin standard JTAG connector and UART to micro-USB connector.

The daughterboard contains the i.MX6UltraLite MCU, a programmed microSD memory with Linux and RAM memory, and a set of switches to select the boot mode and boot device. These three elements are highlighted with yellow rectangles on the daughterboard. Finally, the debug USB, the 5V supply connector and power switch have been highlighted in blue. More information can be found in [MCIMX6UL_EVKB].



(1) Arduino header and daughterboard highlighted in red.

(2) i.MX6 UltraLite, microSD memory card slot and boot mode switches highlighted in yellow

(3) Power supply connector, power switch and debug USB port highlighted in blue

Fig 6. MCIMX6UL-EVKB i.MX6UL evaluation kit

5. Hardware setup

The hardware setup consists of mounting the different boards together. Two simple steps are required. First, plug the A71CH Mini PCB board (OM3710/A71CHPCB) to the I²C adaptor of the Arduino interface board.

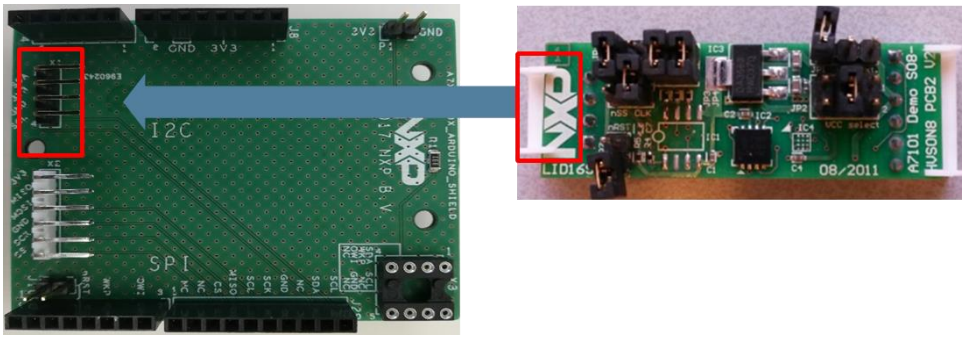


Fig 7. A71CH Mini PCB board (OM3710/A71CHPCB) mounting on the I2C adaptor of the Arduino interface board

Second, plug the A71CH into the i.MXUltraLite evaluation board (MCIMXUL-EVKB) using the Arduino adaptors.

Note: The Arduino shield board comes with male connectors below. In case the MCIMXUL-EVKB board does not come with the Arduino headers assembled by default, these could be easily soldered by the user.

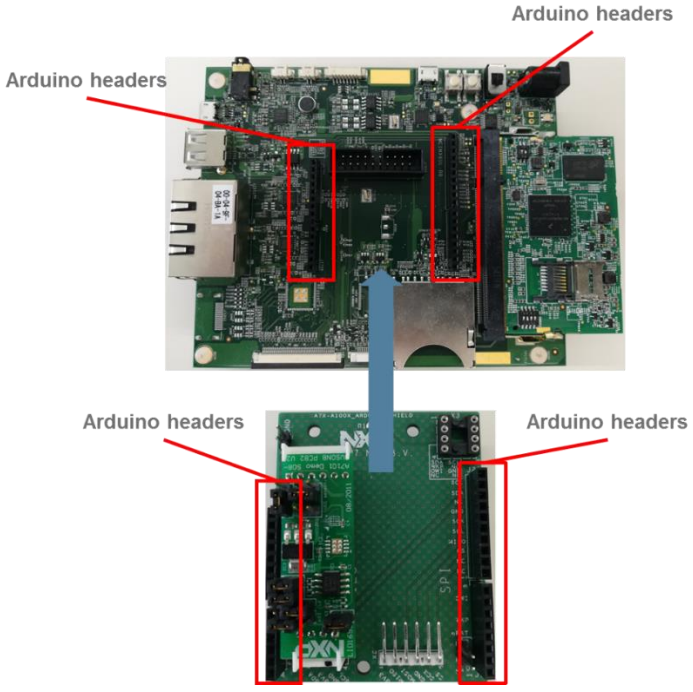


Fig 8. A71CH Arduino kit mounting on i.MXUltraLite board (MCIMXUL-EVKB)

The complete hardware setup mounting corresponds to the following figure:

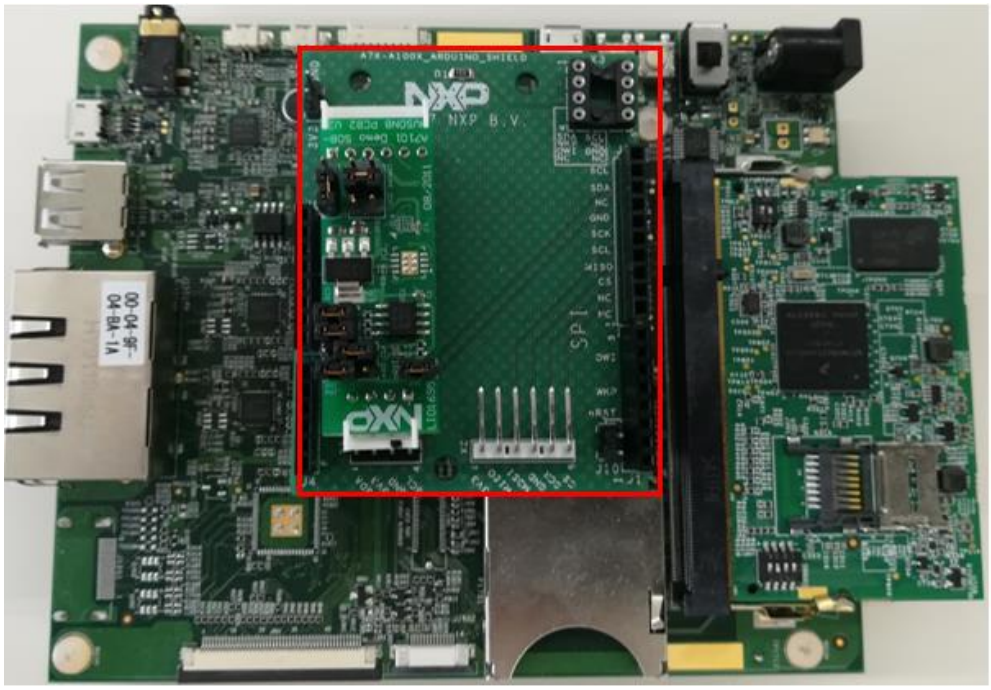


Fig 9. OM3710/A71CHARD mounted on the i.MXUltraLite board (MCIMXUL-EVKB).

6. Software setup

This section details the required to complete the software setup for A71CH security IC and i.MXUltraLite host MCU.

6.1 SD card preparation

The A71CH product support package includes a pre-compiled Linux image with the A71CH Host software package integrated in it [IMX6_LINUX_IMAGE]. The first step is to flash this Linux image in a microSD memory card and plug it in the i.MX6UltraLite evaluation board (MCIMX6UL-EVKB).

The Win32Disk Imager software is used in this document for this purpose. It is an open source program compatible with several versions of Windows. It can be downloaded from [WIN32DISKIMAGER]. The user shall select the directory of the Linux image in their laptop and click the write button (Fig 10).

Once the process is completed, the microSD memory card is ready to be inserted into the card slot of the i.MX6UL evaluation kit daughterboard (Fig 6).

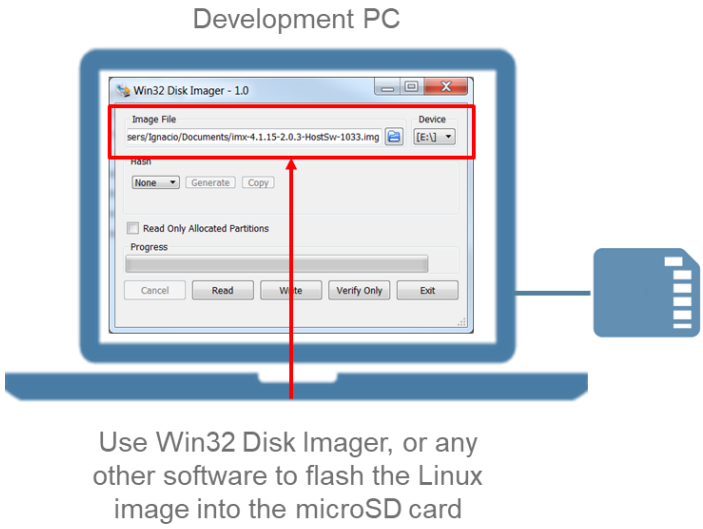


Fig 10. Win32 Disk Imager

6.2 Drivers

The i.MX6UltraLite evaluation board (MCIMX6UL-EVKB) is connected to the PC via USB interface. Also, the evaluation board should be powered by connecting the plug of the 5 V power supply to the DC power jack.

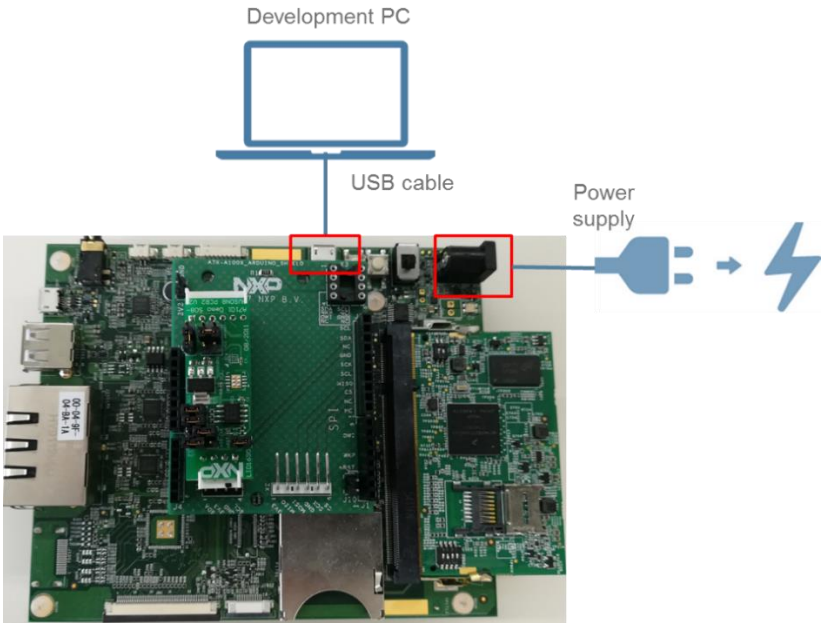


Fig 11. i.MX6UltraLite PC connection and power supply

It might be required to install the USB to UART Bridge VCOM Drivers in some laptops for driving the i.MX6UltraLite. These drivers can be downloaded from [SILAB_DRIVERS]. Install the executable for your processor (either 32 or 64 bits) and follow the setup wizard until it is finished.

Now, i.MX6UltraLite evaluation board (MCIMX6UL-EVKB) should be enumerated in the Device Manager as Silicon Labs CP210x USB to UART Bridge (COMxx) in the Ports (COM & LPT) category (Fig 12). Please note, the port number COMxx will be needed to configure the serial communication between the PC and the i.MX6UL evaluation kit in the next step.

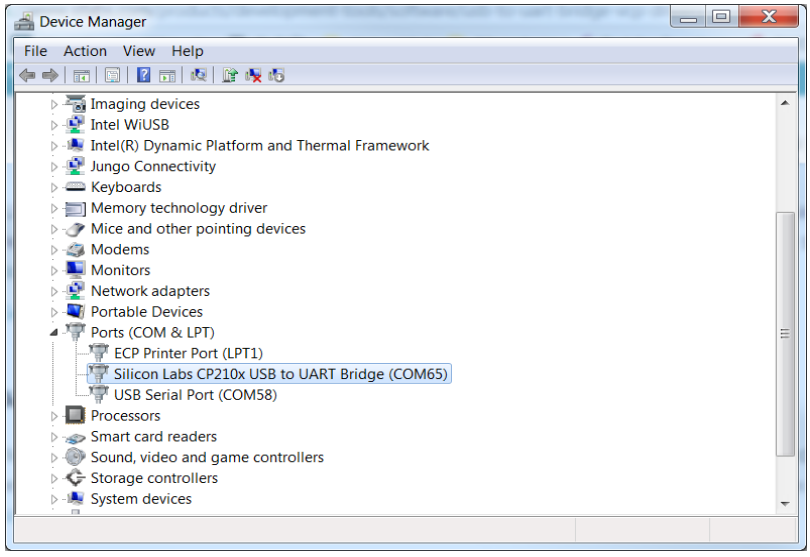


Fig 12. Device manager

6.3 Terminal setup

A terminal application shall be executed from the development PC to interact with the i.MX6UltraLite evaluation board (MCIMX6UL-EVKB). Any terminal supporting a serial port interface can be used.

In this document, Tera Term is used and it can be downloaded from [TERA_TERM]. The setup wizard will guide the user through the installation. The standard installation can be chosen for this purpose. Once it is finished, Tera Term can be started.

The first thing that should be configured is a new connection for the terminal (Fig 13). The user should choose a Serial connection and the port that was checked previously in the Device Manager.

Fig 13. Tera Term new connection

Then Setup-Serial port should be configured as shown in Fig 14 with the proper serial port that was already chosen.

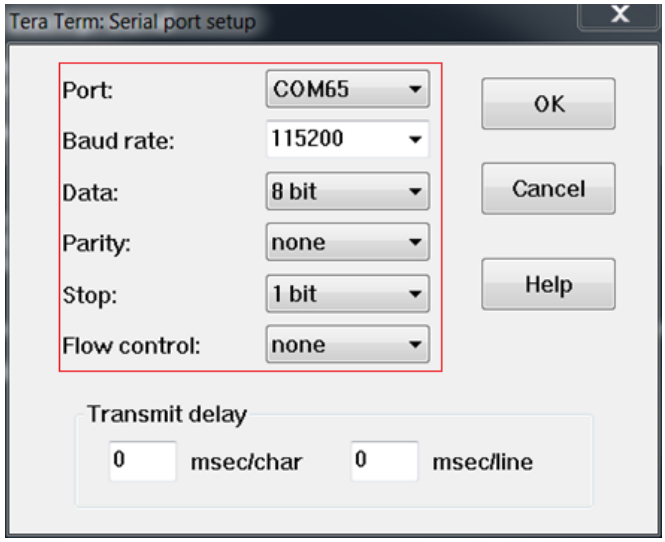


Fig 14. Serial port setup

6.4 Booting the system

Before booting the system, first insert the microSD with pre-compiled Linux image into the microSD card slot of the MCIMX6UL-EVKB daughterboard.

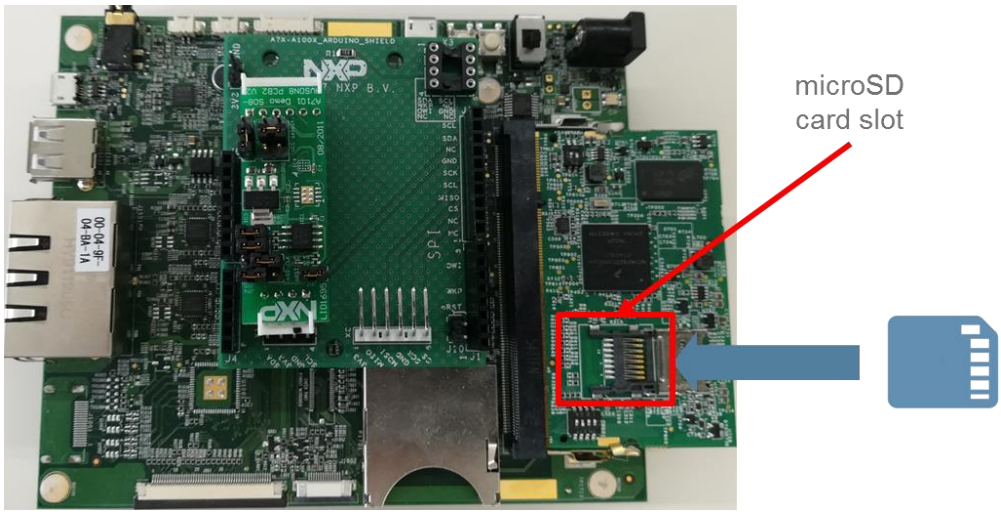


Fig 15. Insert microSD card with the pre-compiled Linux image in the microSD card slot

Second, the switches on the MCIMX6UL-EVKB daughterboard should be configured in the following way:

- Boot Mode Select Switch SW602: ON, OFF (from 1-2 bit)
- Boot Device Select Switch SW601: OFF, OFF, ON, OFF (from 1-4 bit)

Fig 16. i.MX6UltraLite (MCIMX6UL-EVKB) board switches

This configuration allows the i.MX6UltraLite (MCIMX6UL-EVKB) board to be booted from the microSD memory card.

Third, turn on the power supply switch to boot up the system.

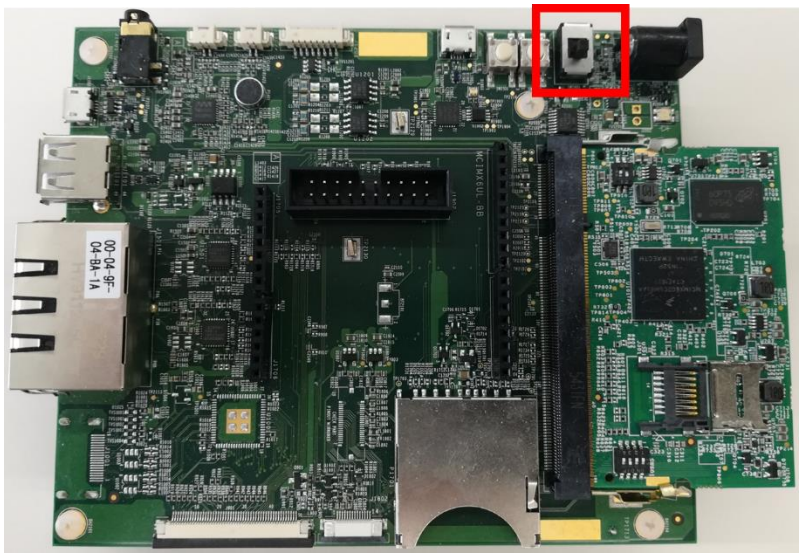


Fig 17. i.MX6UltraLite (MCIMX6UL-EVKB) power supply switch

During the boot process, the operating system status information will be prompted on the terminal (Tera Term) window. When the process is complete, the user can login with the following credentials (Fig 18):

- Account name: root
- Password: not required

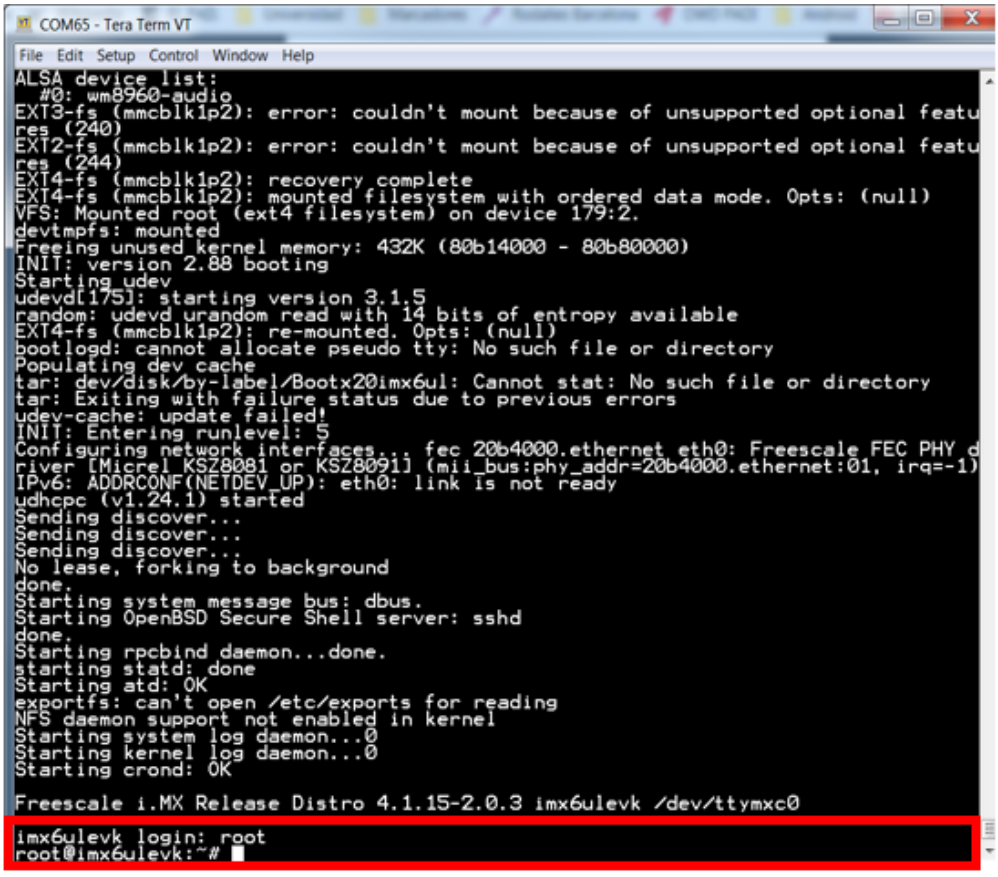


Fig 18. Boot sequence finished

7. A71CH application examples execution

Once the system has been initialized, the A71CH application examples can be executing by navigating through the menu with Linux terminal commands.

7.1 Running A71CH OpenSSL Engine examples

The A71CH OpenSSL Engine examples can be found in:

```
root@imx6ulevk:~# cd axHostSw/hostLib/embSeEngine/a71chDemo/scripts/
```

The following examples are contained in the **scripts** folder:

Table 2. A71CH SW examples

Example script	Description
a71chPrepareEcc	Generate ECC keys with OpenSSL libraries and inject them into the A71CH
a71chRandDemo	Request random numbers to the A71CH
a71chEccCsrDemo	Create a Certificate Signing Request and verify it
a71chEcDhKa	ECDH Key agreement using A71CH

Example script	Description
a71chEccSignDemo	Sign a file with the previously injected keys (it needs the file path as an argument) and verify the signature
a71chEccSignNegTest	ECDSA signature negative test using the wrong signature file
tlsCreateCredentialsRunOnClientOnce	Generation of all the necessary files for a TLS communication, e.g., client and server ECC key pairs and certificates
tlsPrepareClient	Injection of the client ECC key pair into the A71CH Security Module using the Configure Tool
tlsSeClient	Client connection start. The client will establish a TLS/SSL-based communication with a give IP address and port
tlsServer	Server connection start. The server will start to listen to TLS/SSL communication requests through a specified port.

For instance, **a71chPrepareEcc** example can be run using this command:

```
root@imx6ulevk:~/axHostSw/hostLib/embSeEngine/a71chDemo/scripts# ./a71chPrepareEcc.sh
```

It is recommended to run first the **a71chPrepareEcc** example to inject keys into the A71CH that will be used by other scripts included in the package.

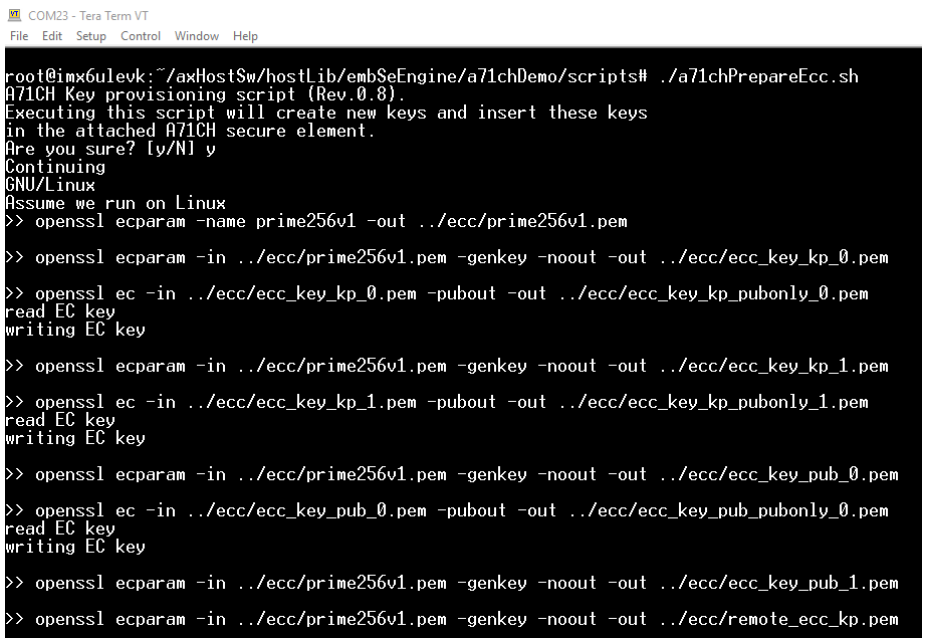


Fig 19. Execution of script a71chPrepareEcc.sh

When executing the examples, the terminal will show the process step by step with some information regarding the example. The scripts can also be opened as text files to examine the OpenSSL commands. Further information about A71CH OpenSSL Engine

examples can be found in [AN_A71CH_HOST_SW] and in the Doxygen documentation [IMX6_LINUX_IMAGE].

7.2 Running A71CH Host API usage examples

It is also possible to navigate to the A71CH Host API usage example application by typing the following:

```
root@imx6ulevk:~# cd axHostSw/linux/
```

Once in the **linux** folder, the A71CH Host API usage project can be launched with the following command:

```
root@imx6ulevk:~/axHostSw/linux# ./A71CH_i2c_imx
```

The A71CH Host API commands will be sequentially executed and prompted in the TeraTerm terminal as shown in Fig 20. Further information about the A71CH Host API usage examples can be found in [AN_A71CH_HOST_SW] and in the Doxygen documentation [IMX6_LINUX_IMAGE].

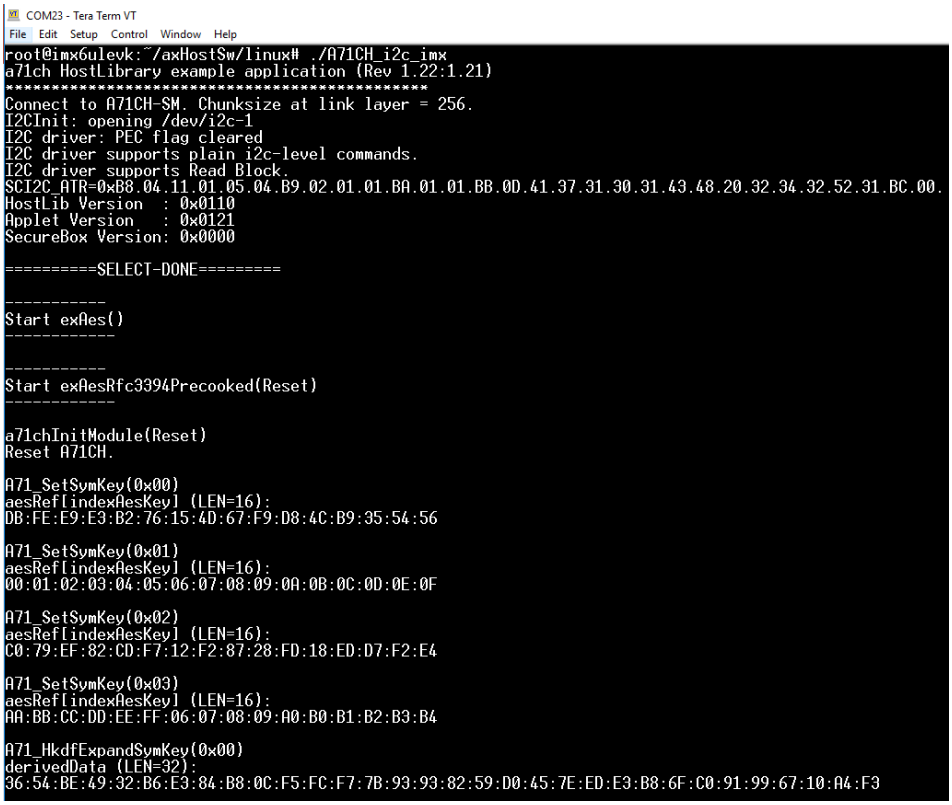


Fig 20. Execution of Host API usage application

7.3 Running A71CH Configure tool

Finally, the A71CH Configure Tool can also be executed from the **Linux** folder:

```
root@imx6ulevk:~/axHostSw/linux# ./a71chConfig_i2c_imx
```

Fig 21 and Fig 22 illustrate the use of the A71CH Configure Tool with debug reset and info status commands, respectively. In Fig 21 the A71CH is started in debug mode.

Fig 21. A71CH Configure Tool execution example

In Fig 22 the status of the A71CH, e.g., stored key pairs, public keys, monotonic counters, etc, is prompted.

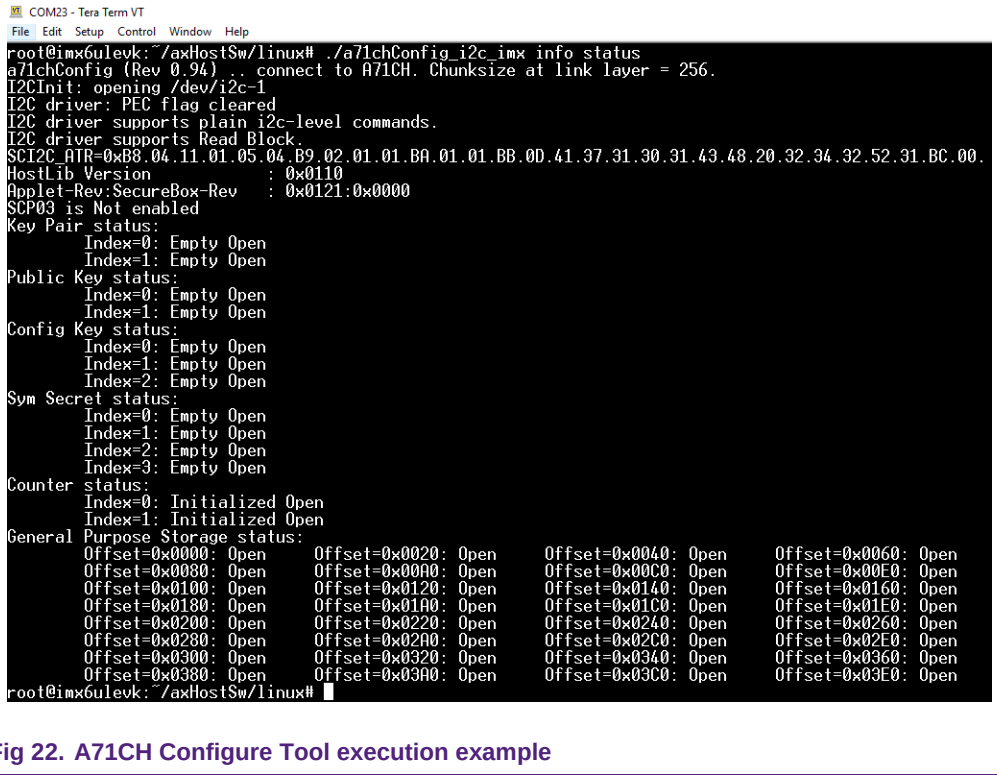


Fig 22. A71CH Configure Tool execution example

Further information about the A71CH Configure Tool can be found in [AN_A71CH_HOST_SW] and in the Doxygen documentation [IMX6_LINUX_IMAGE].

8. References

All the references contained in this document are listed in the following table:

Table 3. References

[SCI2C]	**SCI2C Protocol Specification** – Revision 1.x only, Docstore an1950**[1]
[A71CH_APDU]	**APDU Specification of A71CH Security Module -** DocStore ds4094**[1]
[IMX6_LINUX_IMAGE]	**A71CH Host Software IMX6UL-EVK Linux SD Card Image (Windows Installer)** – Docstore, document number sw4676**[1], REV 4.9.11_1.0.0_v01.03.00 (or later), available on www.nxp.com/A71CH **A71CH Host Software IMX6UL-EVK Linux SD Card Image (Bash Installer)** – Docstore, document number sw4674**[1], REV 4.9.11_1.0.0_v01.03.00 (or later), available on www.nxp.com/A71CH
[SCP03]	**Global Platform Card Specification v2.3** – Amendment D v1.1.1.
[A71CH_OPENSSL_ENGINE]	**A71CH OpenSSL Engine** – DocStore, document number um4334**[1]
[AN_A71CH_HOST_SW]	**AN12133 A71CH Host software package documentation** – Application note, document number 4643**[1]
[MCIMX6UL_EVKB]	**i.MX6UltraLite Evaluation Kit** - www.nxp.com/iMX6ULEVK
[WIN32DISKIMAGER]	**Win32 Disk Imager -** https://sourceforge.net/projects/win32diskimager/
[SILAB_DRIVERS]	**USB to UART Bridge VCP Drivers -** https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers
[TERA_TERM]	**Tera Term terminal** - https://osdn.net/projects/ttssh2/releases/

[1] **… document version number

9. Legal information

9.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

9.1 Licenses

ICs with DPA Countermeasures functionality

NXP ICs containing functionality implementing countermeasures to Differential Power Analysis and Simple Power Analysis are produced and sold under applicable license from Cryptography Research, Inc.

9.2 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

FabKey — is a trademark of NXP B.V.

I²C-bus — logo is a trademark of NXP B.V.

10. List of figures

11. List of tables

12. Contents

For more information, visit: http://www.nxp.com

Date of release: 7 March 2018 **458211**

Document identifier: AN12119