A71CH Quick start guide for Windows Rev. 1.0 — 09 July 2018

492510

Application note COMPANY PUBLIC

Document information

Info	Content				
Keywords	Security IC, A71CH, Quick-start guide, Windows				
Abstract	This document gives information on how to get started with the A71CH Mini PCB board (OM3710A71CHPCB) for evaluation of A71CH features in a Windows computer.				



Revision history

Rev	Date	Description
1.0	20180709	First release

Contact information

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

AN12134

Application note COMPANY PUBLIC

1. Introduction

This document gives information on how to get started with the A71CH Mini PCB board (OM3710A71CHPCB) for evaluation of A71CH features in a Windows computer. It gives an overview of the required hardware and software options. It also gives step-by-step instructions for setting up the development environment as well as full directions for running the example application in a Windows-based environment.

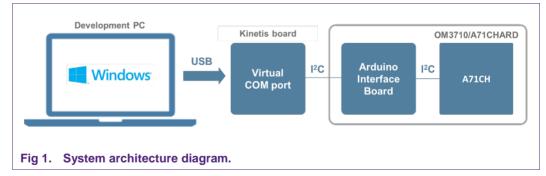
2. A71CH Overview

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 an outstanding level of security measures to protect the IC against many physical and logical attacks. It can be integrated in various host platforms and host 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, reference designs, documentation and IC samples for product evaluation.

3. System description

The A71CH evaluation setup presented in this document consists of an A71CH security IC connected to a Windows-based platform through a FRDM-K64F or FRDM-K82F board programmed to act as a virtual COM port and the A71CH Arduino compatible kit (OM3710A71CHARD). Fig 1 shows a basic diagram of the system architecture.



The document explains the following parts:

- Hardware overview: It describes the FRDM-K64F and the FRDM-K82F development platforms as well as the A71CH Arduino compatible kit (OM3710/A71CHARD).
- Hardware setup: It describes how to connect the A71CH Arduino compatible kit with the FRDM-K64F or the FRDM-K82F boards.
- **Software setup**: It describes how to configure the development environment and how to import the required software packages

- A71CH application examples execution: It describes how to run the A71CH Windows-based application examples contained in the A71CH Host software package for
- **RJCT server:** It describes how to use the RJCT server to communicate with A71CH from a remote or local client process.

In addition, the document includes the following appendix:

- USB to I²C bird: It describes how to use the USB to I²C bird Ascot adaptor (OM3710/B001) to interface a Windows platform as an alternative to a Kinetis board configured as a virtual COM port.
- VCOM driver installation: It describes how to troubleshoot VCOM driver installation issues.

4. Hardware overview

This setup uses a Windows laptop as a host MCU while the A71CH security IC acts as the protected storage module. The following material is needed:

- 1. A computer running Microsoft Windows 7 (32 or 64 bit) or later.
- 2. The A71CH Arduino compatible development kit (OM3710/A71CAHRD).
- 3. The FRDM-K64F or FRDM-K82F evaluation board.

4.1 A71CH Arduino compatible development kit (OM3710/A71CHARD)

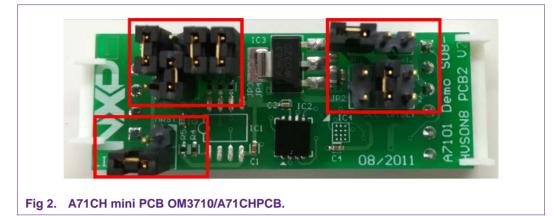
The OM3710/A71CHARD is an Arduino development kit containing two items as well:

- 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 employed. The figure shows the jumpers configuration that enables the use of the A71CH I²C interface.

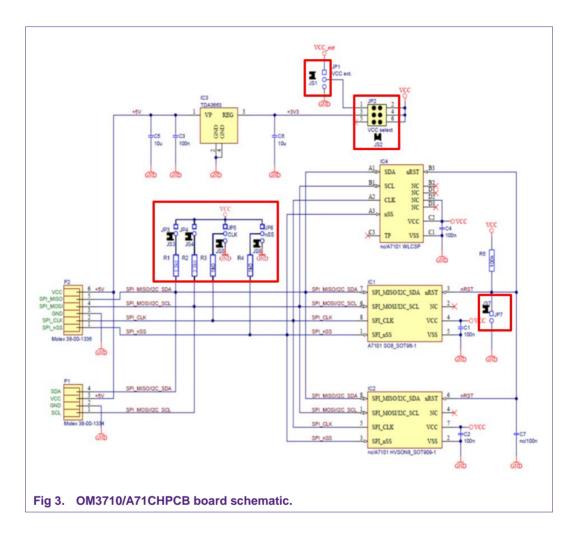


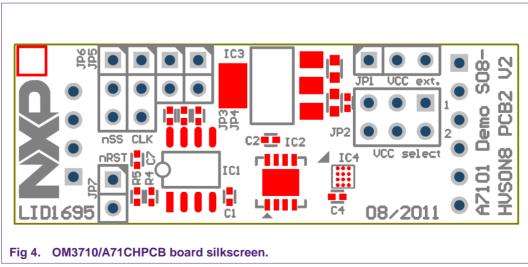
To enable the I²C communication protocol, it is necessary to configure JP5/6 according to Table 1. JP2 connects the A71CH to the on-board 3.3V voltage regulator on the MiniPCB board. The jumpers JP3 and JP4 enable the I²C SDA/SCL pull-up resistors. JP7 can be used to connect the A71CH reset signal.

Jumper	Setting	Usage
JP1	Not set	External VCC connection
JP2	3-4	Connect A71CH to 3.3V regulator on MiniPCB
JP3	Set	Connect I ² C SDA pull-up resistor
JP4	Set	Connect I ² C SCL pull-up resistor
JP5	1-2	Use I ² C address 0x92/0x93
	2-3 (Default)	Use I ² C address 0x90/0x91
JP6	1-2	Activate I ² C interface
JP7	Not set (Default)	A71CH operates
	Set	A71CH IC reset

Table 1. Default OM3710/A71CHPCB Jumper settings

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

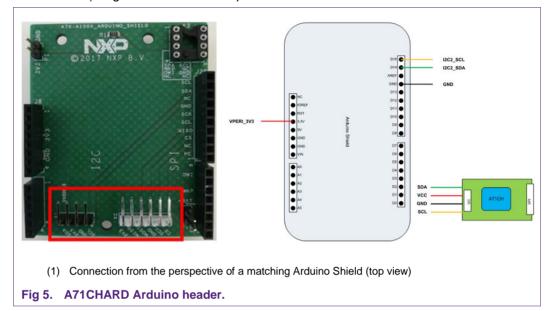




AN12134

4.1.2 Arduino interface board

The Arduino header board permits the user to interface the A71CH OM3710/A71CHPCB with other platforms with dedicated Arduino headers (e.g., FRDM-K64F and FRDM-K82F Kinetis boards). Fig 5 shows the board pinout.



4.2 Freedom development platforms for Kinetis

The section details the Freedom development platforms for Kinetis supported by the A71CH support package for Windows machines.

4.2.1 FRDM-K64F

The Kinetis FRDM-K64F [FRDM_K64F] development platform is a simple, yet sophisticated design, featuring a Kinetis K64 series microcontroller, built on the ARM® Cortex®-M4 core. The FRDM-K64F can be used to evaluate the K64, K63, and K24 Kinetis K series devices. It features the MK64FN1M0VLL12 MCU, which boasts the maximum operation frequency of 120 MHz, 1 MB of flash, 256 KB RAM, a full-speed USB controller, Ethernet controller, secure digital host controller, and analog and digital peripherals.

The FRDM-K64F hardware is form-factor compatible with the Arduino R3 pin layout, providing a broad range of expansion board options. The onboard interface includes a six-axis digital accelerometer & magnetometer, RGB LED, SDHC, add-on Bluetooth module, add-on RF module, Ethernet and OpenSDAv2, the NXP open-source hardware embedded serial and debug adapter running an open-source bootloader.

A71CH Quick start guide for Windows

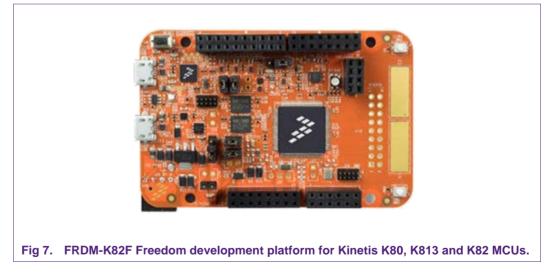


Fig 6. FRDM-K64F Freedom development platform for Kinetis K64, K63 and K24 MCUs.

4.2.2 FRDM-K82F

The Freescale Freedom K82 hardware [FRDM-K82F] is a simple yet sophisticated design featuring a Kinetis K series microcontroller built on the ARM® Cortex®-M4 core which features a floating-point unit (FPU).

The FRDM-K82F can be used to evaluate the K80, K81, and K82 Kinetis K series devices. The FRDMK82F board features the K82FN256VLL15 MCU, which boasts a maximum operation frequency of 150 MHz, 256 KB of flash, a 256 KB RAM, a full-speed USB controller with available crystal-less operation, and analog and digital peripherals.

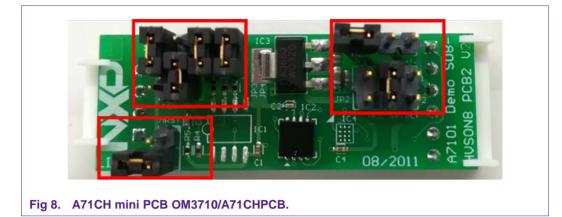


5. Hardware setup

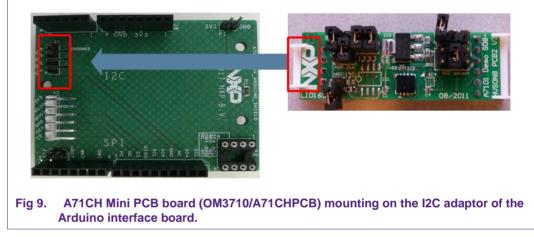
The hardware setup consists of mounting the different system parts together. Three simple steps are required. First, make sure the jumpers of the OM3710/A71CHPCB are properly configured.

AN12134

A71CH Quick start guide for Windows



Then, plug the A71CH Mini PCB board (OM3710/A71CHPCB) to the I²C adaptor of the Arduino interface board (Fig 9).

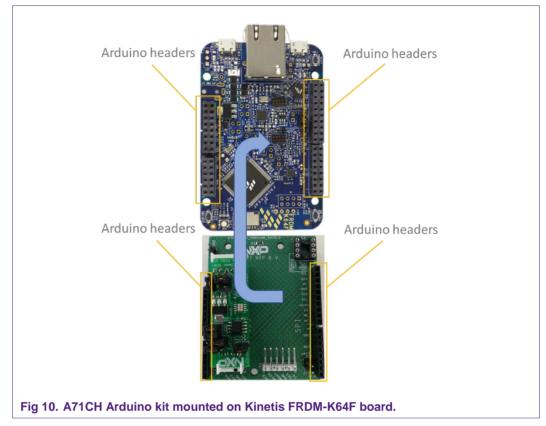


Finally, plug the A71CH into the Kinetis board using the Arduino adaptors (Fig 10). Please note the Arduino shield board comes with male connectors below.

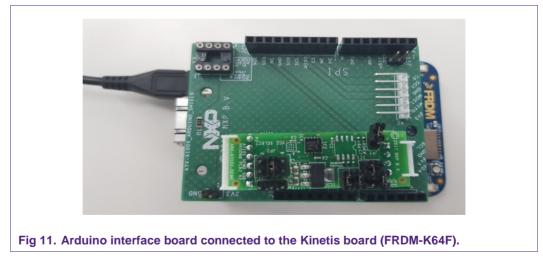
Application note

COMPANY PUBLIC

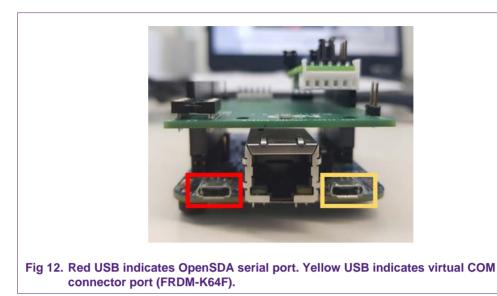
AN12134



Then, A71CH security IC is connected to the Kinetis board through the Arduino interface board (Fig 11).



As can be observed, there are two USB connectors in the FRDM-K64F and FRDM-K82F boards (Fig 12). The USB connector highlighted in red corresponds to OpenSDA serial port. This port is used by the development PC to configure the Kinetis board as a virtual COM port. The USB connector highlighted in yellow corresponds to the virtual COM connector port.



6. Software setup

This section details the required steps to complete the software setup for running the A71CH application examples in a Windows-based platform.

Note: This section details the software setup for running the A71CH application examples in Windows-based platforms based on A71CH HostLib v1.4.0. If you are using a different A71CH HostLib version, the screenshots or project names indicated in this section may differ.

6.1 MCUXpresso IDE installation

MCUXpresso IDE is a fully featured software development environment for NXP's ARMbased MCUs, and includes all the tools necessary to develop high-quality embedded software applications in a timely and cost-effective fashion.

MCUXpresso IDE is based on the Eclipse IDE and includes the industry standard ARM GNU toolchain. It brings developers an easy-to-use and unlimited code size development environment for NXP MCUs based on Cortex-M cores (LPC, Kinetis and i.MX RT). The IDE combines the best of the widely popular LPCXpresso and Kinetis Design Studio IDEs, providing a common platform for all NXP Cortex-M microcontrollers.

MCUXpresso IDE is a free toolchain providing developers with no restrictions on code or debug sizes. It provides an intuitive and powerful interface with profiling, power measurement on supported boards, GNU tool integration and library, multicore capable debugger, trace functionality and more. MCUXpresso IDE debug connections support Freedom, Tower, EVK, LPCXpresso and custom development boards with industry leading open-source and commercial debug probes including LPC-Link2, P&E and SEGGER.

The fully featured debugger supports both SWD and JTAG debugging, and features direct download to on-chip and external flash memory

The installation file of MCUXpresso can be found in [MCUXPRESSO_IDE]. The setup wizard will guide the user through the process of installing MCUXpresso correctly. Since MCUXpresso requires extra drivers during the installation, check all the items on the list to allow the installation of the drivers. Make sure the checkbox for installing the NXP debug drivers is activated (Fig 13).

57 X Setup - MCUXpresso IDE Optional debug driver selection These drivers are required when using the debug probes listed below. Windows may issue warnings when installing drivers that the IDE requires. These include drivers from Jungo Connectivity, PE Micro, SEGGER, and Ashling Microsystems Ltd as well as NXP. If prompted, please allow these drivers to be installed. NXP LPC-Link1 Debug drivers Red Probe Debug drivers http://www.nxp.com/mcuxpresso/ide Next > < Back Cancel Fig 13. MCUXpresso install wizard.

Note: Please, install MCUXpresso IDE version 10.2.0 or higher

6.2 OpenSDA configuration

OpenSDA is a serial and debug adapter built into the Kinetis board. It provides a bridge between the development PC and the Kinetis MCU, which can be used for debugging, flash programming and serial communication all over USB.

Note: This section explains how to install the correct OpenSDA bootloader firmware version to the Kinetis FRDM board. This needs to be done for debugging, flash programming, and serial communication over a single USB connection between a host and an embedded target processor. If this section is not followed carefully, it is possible that the examples will not be executed.

To configure OpenSDA into the Kinetis board, an OpenSDA bootloader (.bin file) should be downloaded from OpenSDA website [OPENSDA_FIRMWARE]. Scroll down the page to section '*Compatible Evaluation Boards*' and search for the target Kinetis board. For

AN12134

instance, Fig 14 depicts the OpenSDA bootloader version defined for the Kinetis FRDM K64: version 2.0.

Compatible Evaluation Boards	
The following eval boards have been tested with the J-Link OB firmware and are k SEGGER.	nown to be working. Other eval board may work as well but are not guaranteed to do so. In case of doubt, please consult
Evaluation Board	OpenSDA bootloader version
FRDM-K22F	2.1
FRDM-K28F	2.1
FRDM-K64F	2.0

Fig 14. OpenSDA bootloader version for the Kinetis FRDM-K64F.

Once the OpenSDA bootloader version is identified, click on the '*Downloads*' section, and scroll down until '*J-Link OpenSDA – Generic Firmwares*' appears and download the desired version. Fig 15 illustrates the process; in this case OpenSDA V2 Bootloader has been selected according to the compatible evaluation boards table previously mentioned.

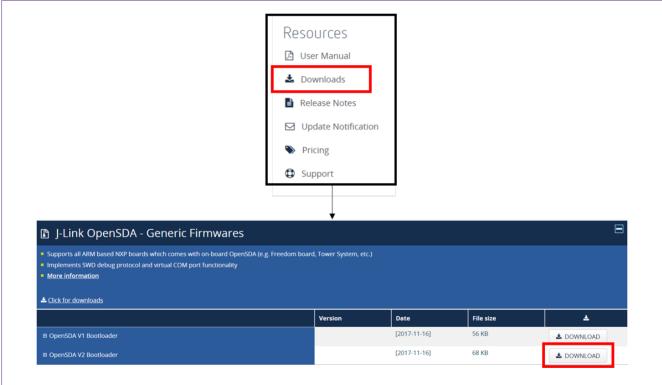
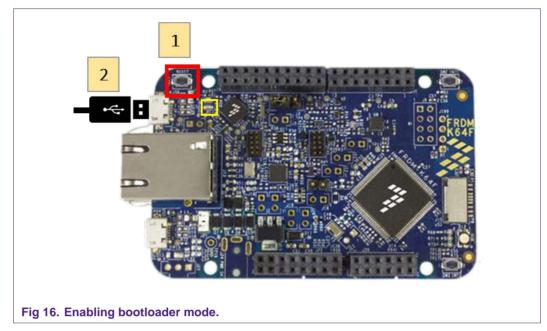


Fig 15. Desired firmware for the Kinetis FRDM-K64F.

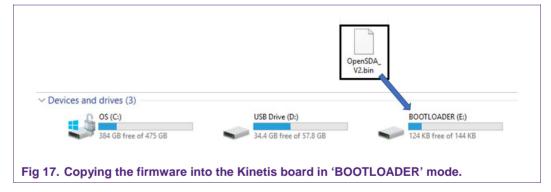
To write the downloaded firmware into the Kinetis board, the bootloader mode must be enabled. For this, press the Kinetis board '*Reset*' button and, while holding it down, connect a USB cable to the Kinetis board (e.g., FRDM K64, Fig 16).

AN12134



After connecting the USB cable to the Kinetis board, the green led located inside the yellow square will start blinking and the development PC will show a new drive called 'BOOTLOADER'.

Drag the downloaded firmware directly into the drive (Fig 17). Once the file is copied inside the '*BOOTLOADER*' drive, unplug the Kinetis board and plug it again. The green led remains still, thus indicating that the OpenSDA bootloader firmware has been configured correctly.



Application note

COMPANY PUBLIC

6.3 Kinetis SDK package for A71CH

To generate and download your customized SDK for your Kinetis FRDM board, you can enter the MCUXpesso SDKBuilder website [SDKBuilder] and follow these steps

1. Select your Kinetis FRDM board and click on '*Build MCUXpresso SDK*; in this case the selected board is the *FRDM-K64F* (Fig 18).

Search for your board or kit to get started.			
Search by Name		Hardware Details	
Search		Board	FRDM-K64F
		Device	MK64F12
Select a Device, Board, or Kit	14	Core Type / Max Freq	Cortex-M4F / 120MHz 1024 KB Flash
 Boards 		Memory Size	256 KB RAM
▼ Kinetis			
FRDM-K22F		Actions	
FRDM-K28F		Build MCUXpress	o SDK 2
FRDM-K64F	1		
FRDM-K66F		Explore selection	with Clocks tool
FRDM-K82F		Explore selection	with Pins tool
FRDM-KE02Z40M			
FRDM-KE04Z			
Name your SDK			
FROM-K64F			

In the next screen, select the software components (Fig 19):

- 1. Select 'Add software component'.
- 2. Select the middleware; choose options '*FatFS*', '*USB stack*', '*IwIP*', '*mbedtIs*', '*Secure Element*', '*Amazon-Freertos Kernel*' and '*AWS IoT*'.
- 3. Click on 'Save changes'.
- 4. Finally, click on 'Download SDK'.

Application note

COMPANY PUBLIC

SDK Builder	- CDK arabica far una with dealder MCI	Select Software Components	× 100	
Generate a downloadable	e SDK archive for use with desktop MCl	Select optional components from the list below to be added to your SDK.		
		7 items selected - 4		
Developer Environment Se	ittings	Selected Middleware		
	nd examples projects included in the SDK and Gener	AWS IoT, FalFS, Secure Element, USB stack, twiP, mbedtls, Amazon-FreeRTOS Kernel		FRDM-K64F
Host OS	Toolchain / IDE			MK64F12
Windows -	MCUXpresso IDE		req	Cortex-M4F / 120MHz
				1024 KB Flash
Select Optional Middleward	e			256 KB RAM
Add middleware, operating system	ms, and software libraries to your SDK.			
 Add software compone 	nt 3	Cancel Save ch	anges 5	KSDK 2.4.0 (released 2018-05-25)
		Host OS:		Windows
		Toolchain:		MCUXpresso IDE
This MCUXpresso SDK cor	nfiguration is available for direct download	Middleware	et)	FatFS, USB stack, IwIP
	Archive Name	Documenta	lian	
Download SDK 6	FRDM-K64F (3)			
	Don't use: <>	of your SDK Base SDK:		MCUXpresso SDK API Reference Manu

The downloaded SDK should be imported in MCUXpresso IDE. To import the SDK into MCUXpresso IDE, drag and drop the SDK file inside the red square ('*Installed SDKs*') and then click 'OK to confirm the operation (Fig 20).



6.4 Importing a project in MCUXpresso IDE

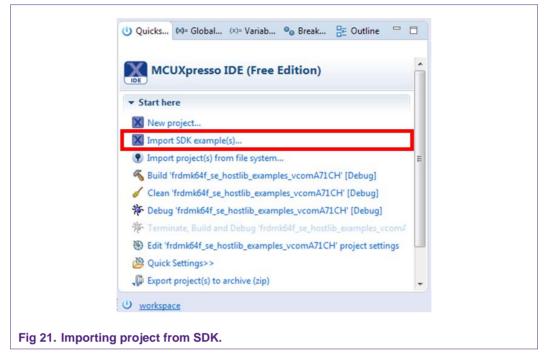
There are two possible ways to import A71CH project examples in MCUXpresso IDE, depending on if we are using the MCUXpresso project files bundled with the A71CH Host Software package installer or if the installed SDK package already contains the A71CH middleware:

- Importing the A71CH example projects from local drive (bundled with installer)
- Importing the A71CH example projects from the installed SDK.

6.4.1 Importing the A71CH example projects from the installed SDK

The first option is to import the A71CH example projects from the installed SDK:

1. Select '*Import SDK example(s)*'... to import available example projects to the workspace.



An SDK Wizard window will pop-up:

2. Select 'frdmk64f' from Available boards and then click the next button (Fig 22).

SDK Import Wizard				
Importing project(s) for device: M	K64FN1M0xxx12 using board: FRDM+K64F			
Board and/or Device	selection page			
▼ SDK MCUs	Available boards			1ª2 1%
MCUs from installed SDKs	Please select an available board for your project.			
NXP MK64FN1M0xxx12	Supported boards for device: MK64FN1M0xxx12			
	Land A			
Selected Device: MK64FN1M0xx	x12 using board: FRDM-K64F	SDKs for selected MCU		
Target Core: cortex-m4		Name		fest Ve Location
Description: K64_120: Kinetis® K	64-120 MHz, 256KB SRAM Microcontrollers (MCUs)	SDK_2.x_FRDM-K64F	2.4.0 3.2.0	Contraction Contractic Contr
based on ARM® Co	vrtex®-M4 Core			
0			< Back	Next > Finish Car

Fig 22. Importing project from SDK. Board selection page.

3. A list with different elements included in the SDK will appear; click on 'se-hostlibexamples' and 'Finish' (Fig 23).

Please select one or more examples to import			
Import projects			
Project name prefix frdmk64f_	2 Project name suffix		a.
Use default location			
Location: C:\Users\aleja\Documents\MCUXpressolDE_10.1.1_606\workspace\frdmk64f_			Browse
Project Type	Project Options		
○ C Project. ○ C++ Project ○ C Static Library ○ C++ Static Library	SDK Debug Console ⊚ Se ☑ Copy sources ☑ Import other files	mihost 🔘 UART	
Examples		2 1	
type to filter		Version	
> □ = dever, complets > □ = hop, complets > □ = mind-adl, complets □ = adl, complets			
Ø	٩	Back Next > Finish	Cancel

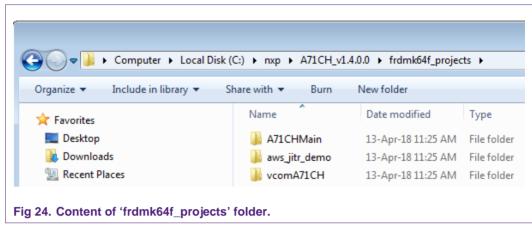
The imported examples will appear in the workspace window

6.4.2 Importing A71CH example projects from local drive (bundled with installer)

The second option is to use the project files bundled with the A71CH Host Software package installer. The A71CH Host Software Package can be downloaded from [A71CH_HOST_SW]. For instance, these are in

'*A71CH_v<libversion>/frdmk64f_projects*' in the case of the FRDM K64F board. The content of this folder is illustrated in Fig 24. As can be seen, there are three example projects:

- A71CH Host API usage project: demonstrates the usage of various functionalities of the A71CH in combination with mbedTLS cryptographic library.
- VCOM project: allows the Kinetis board to be used as a bridge between the PC and the A71CH and enables the execution of the A71CH Configure tool and other utilities from the development PC. It will be used in this document.
- AWS JITR demo project.



To import a project from file system, click on '*Import project(s) from file system...*' in the 'Quick start Panel' located in the bottom left (Fig 25).

MCUXpres	so IDE (Free Edi	tion)		
▼ Start here				_
New project				
Import SDK exa	mple(s)			
Import project(s)	s) from file system	1		
K Build " []				L
🖌 Clean '' []				
🎋 Debug " []				
🎋 Terminate, Build	d and Debug " []			

After clicking the import option, a new pop-up will open. In the '*Project directory* (*unpacked*)' field, browse and point to the correct project directory (Fig 26). Then, click on '*Next*'.

Import project(s) from the Select the examples archive	
project archive or root direct wish to import, and press <	in archives (.zip) or are unpacked within a directory. Select your ctory and press <next>. On the next page, select those projects you Finish>. en and 'legacy' examples are provided.</next>
Project archive (zip)	
Archive	Br
Project directory (unpack	ed)
Root directory C:\nxp\A	71CH_v1.4.0.0 Br
MCUXpresso IDE includes button in the Project arch	ended code base for Cortex-M based NXP LPC Microcontrollers. the LPCOpen packages which can be imported directly by pressing the B ive (zip) section, above, and navigating to the Examples/LPCOpen directo atton below to Browse the nxp.com website for latest resources. ces on nxp.com
?	< Back Next > Finish Car

Finally, select all the available A71CH example project and then click on 'Finish'.

Note: It is of most importance to uncheck the "Copy projects into workspace", as shown in Fig 27.

Application note

COMPANY PUBLIC

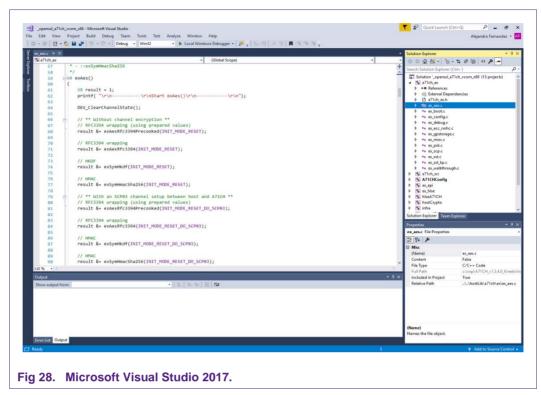
Select a directory to sear	n file system ch for existing Eclipse projects.	
Projects:		
🗸 aws_jitr_demo (C	nxp\A71CH_v1.4.0.0\frdmk64f_projects\A71CH :\nxp\A71CH_v1.4.0.0\frdmk64f_projects\aws, .nxp\A71CH_v1.4.0.0\frdmk64f_projects\vcom	jitr_demo)
Options Copy projects into w Working sets	orkspace 🔶 Uncheck	
Add project to work Working sets:	ing sets	▼) Sglect

6.5 Microsoft Visual Studio IDE installation

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as web sites, web apps, web services and mobile apps. It is only needed to build and run the Host API examples. The A71CH configuration tool is delivered as well as precompiled binary in the host library package.

Application note

COMPANY PUBLIC



The available A71CH projects for Microsoft Visual Studio support Microsoft Visual Studio 2010, 2012, 2015 and 2017 versions. The Microsoft Visual Studio IDE installation process can be found as part of the Microsoft online documentation.

7. A71CH application examples execution

The A71CH Host software package includes two application examples that can be executed in a Windows platform:

- A71CH Host API usage example: A sample project including a set of source code examples oriented to show the A71CH Host Library usage.
- A71CH Configure Tool: A command line tool that supports the injection of credentials into the A71CH.

Note: This section details the A71CH application examples execution based on A71CH HostLib v1.4.0. If you are using a different A71CH HostLib version, the screenshots or project names indicated in this section may differ.

The following steps are required to run the A71CH Windows-based applications:

- Load 'xxx_se_hostlib_examples_vcomA71CH' program into the Kinetis using MCUXpresso.
- Connect the Kinetis to the Windows platform over USB.
- Build the A71CH same projects with Microsoft Visual Studio.

• Start the built application from a Windows terminal, passing the virtual COM port address as input argument.

7.1 Set the Kinetis board as virtual COM port

Before running the A71CH example applications from a Windows platform, the Kinetis must be programmed as a virtual COM port. If the SDK has already been installed, and the project examples have been imported as explained in section 6.4, open MCUXpresso IDE and take the following steps (Fig 29):

- 1. Select 'frdmk64f_se_hostlib_examples_vcomA71CH' project.
- 2. Click on Debug '*frdmk64f_se_hostlib_examples_vcomA71CH* [Debug]' (Note that the name can be slightly different in future versions of the Host software package).
- 3. Select J-Link OpenSDA probe and click on 'OK'. Make sure the OpenSDA serial port is connected to the Windows platform (Fig 12, highlighted in red)
- 4. If a 'Terms of use' pop-up appears, check '*Do not show this message again for today*' box and click '*Accept*'.

• 🛄 🐚 🔁 • 🗞 • 📸 🎋 🏘 • 🔘 • 💁 • 🏟 🥭 🛷 •	●目目× ▶■■#XXの以目表 ●日単文の(#14)// ↓ ●日日	[]• φ φ•¢.•	Quick Access
Project E 52 👾 Peripher IIII Registers 🕴 Symbol 😁 🗖	Welcome 12		- 40
😑 🔯 🖶 🛛 🕶 🗸	🗇 🗇 🛢 🧬 file:///C:/nxp/MCUXpressoIDE_10.1.1_606/ide/pages/registeredFreeEdition.htm		
Grindfid (a p.) kotility, anarojak million) Grindfid (a p.) kotility, anarojak million) Individi (a p.) kotility, anarojak million) I Qurkis (b)- Global (b)- Variab % Break Coutine (Couting) Couting Qurkis (b)- Global (b)- Variab % Break Couting Couting Couting Qurkis (b)- Global (b)- Variab % Break Couting Couting Couting Qurkis (b)- Global (b)- Variab Pa. Break Couting Couting Couting Couting			
• Start here	Closed Concel C	Do not show this message again for today Decline Accept	
Tee project. Monor SIX-completion. Monor SIX-completion. Monor SIX-completion. Monor SIX-completion. Monor SIX-completion. Monor SIX-completion. Count Tednided up, hostile, examplete, ucronA21Cef (Debug) Monor SIX-completion. Six SIX-theodile, the statice exampletion (Debug) Six SIX-theodile, the statice exampletion (Debug) Monor SIX-completion. Six SIX-theodile, the statice exampletion (Debug) Monor SIX-completion. Six SIX-theodile, the statice exampletion (Debug) Monor SIX-completion. Six SIX-theodile, the statice (Debug) Monor SIX-completion. Six SIX-theodile, the statice (Debug) Monor SIX-completion. Six SIX-theodile. Six SIX		4	2

After that, the project will start to compile and execute automatically.

7.1.1 Connect the Kinetis board to the Windows platform over USB

Once the Kinetis board has been configured as a virtual COM port, connect it to the Windows-based platform through the USB port highlighted in yellow in Fig 12. If the Kinetis board is not recognized, check Appendix B for drivers troubleshooting.

7.2 Running A71CH Configure tool

In order to run the A71CH Configure tool, the Kinetis board is programmed to behave as a USB to I2C adapter and the A71CH Configure tool application is executed from a development PC running a Windows platform. Fig 30 illustrates the system architecture of this scenario.

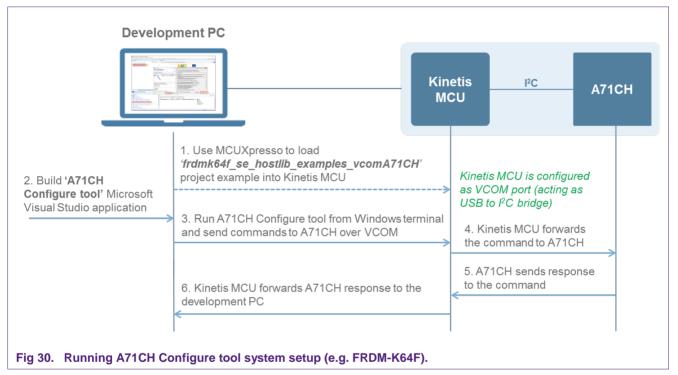
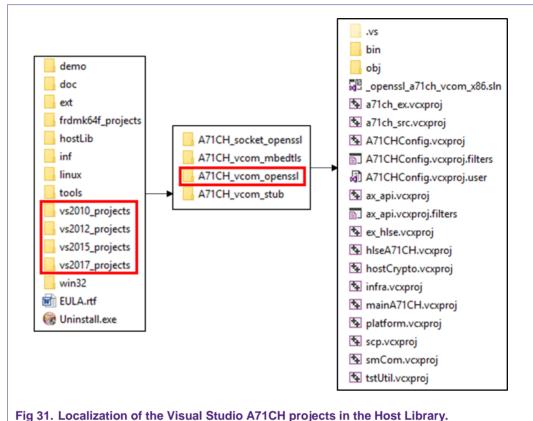


Fig 31 indicates the path to the A71CH Configure tool files (depending on the Microsoft Visual Studio installed version). As can be observed, there are '_vcom_' projects and '_socket_' projects. The first ones are meant to output the APDU commands through a virtual COM port, while the '_socket_' projects are meant to output the APDU commands through a TCP/IP socket (these will be further presented).

Open the '_vcom_' project by double-clicking '_openssl_a71ch_vcom_x86.sln'. Microsoft Visual Studio will be automatically opened.



3.

To configure the virtual COM address into the A71CHConfig project, do the following:

- 1. Right-click 'A71CHConfig' in the 'Solution Explorer' tab and open the 'Properties' window.
- 2. Then, in 'Configuration Properties Debugging' tab make sure the virtual COM address is set in the 'Command Arguments' field.
- 3. Once the VCOM port has been defined, invoke menu *Build Build Solution* to create the executable.

Fig 32 illustrates the above-mentioned steps to build the project in Microsoft Visual Studio. The '*Build Solution*' option has been highlighted in red, while the '*Properties*' and '*Command Arguments*' field have been highlighted in green. In addition, each step has been numbered in the figure.

AN12134

Image: Delayer to leasts: School Speers (2nd) Image: Delayer to least (2nd) School Speers (2nd) Imag	Redukt Solution Cara Solution Build full program databased for solution Build ACTORCordig Cana ACTORCordig Cana ACTORCordig Cana ACTORCordig Cana Solution Project Oordig Cana Solution Project Oordig Project Oordig Projec		Build Debug Team Tools Test	Ctrl+Shift+B 3 g	er = 🔎 💡						
s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-F13 s-	Colar Addato Bidd All grappin iddato fit for toulour No clock Addato re Subdato Bidd All Clock addato re Subdato Bidd All Clock addato re Subdato Const Andrée (Serie) Receited All Clock addato re Subdato Const Andrée (Serie) Receited All Clock addato re Subdato Const Andrée (Serie) Receited All Clock addato re Subdato Const Andrée (Serie) Receited All Clock addato re Subdato Const Andrée (Serie) Receited All Clock addato re Subdato Const Andrée (Serie) Receited All Serie Configuration Managerie Configuration Managerie Configuration Managerie Configuration Managerie Configuration Managerie Configuration Managerie State Bala (Serie) State Bala (Serie)<										
Part1 Part1 Part2 Part2 correct Active/Win327 Part2 co	bud at hoge at attacket in the goods Bud At TRECords Constraint Concords Reach Bud. Configuration Minagor. Configuration		Clean Solution								
Image: Debugger to learch. Image: Debugger to learch. Debugger to learch. Configuration Manage Debugger to learch. Statestable Debugger to learch. Configuration Manage Market. Statestable Market. Statestable Market. Statestable Market. Statestable Market. Statestable Debugger to learch. Statestable Command Agrice Statestable Market. Statestable Market. Statestable Market. Statestable Market. Statestable Market. Statestable Market. Statestable Debugger Note: Statestable Market. Statestable Debugger Note: Statestable	Nucleok Adaption Oddaton Multiple State ATTOR/Config Data Attor Project Doly Project Doly Project Doly Project Doly Configuration Managon Conf		Build full program database file for s	olution							
Partore: Active(Wn32) Partore: Active(Wn32) Configuration Manage Partore: Active(Wn32) Configuration Manage Configuration Configuration Manage Configuration	wide Alt CHCK only Selection (specific) Behad Alt CHCK only Image: Selection (specific) Configuration Measure Image: Selection (specific) Image: Selection (specific) Image: Selection (specific) </th <th></th>										
	Part Note of a Part		the Build A71CHConfin								_socket_x86" (13 projects)
Pattorn: Rod Number 0 Pattorn: Configuration Manage 0 Configuration Manage Configuration Manage 0 Configuration Manage Configuration Manage 0 Configuration Manage Configuration Manage 0 Construction Manage Configuration Manage 0 Connection Configuration Manage Control Configuration Manage 0 Connection Configuration Manage Control Configuration Manage 1 Connection Configuration Manage Control Configuration Configuration Manage 1 Control Configuration Configuration Configuration Manage Control Configuration Conf	Clear ATCHCordig Find Cold Analysis on ATCHCordig Project Only Find Cold Analysis Configuration Measage Find Configuration Programs Configuration Programs Find Configuration Programs Configuration Programs Find Configuration Programs Market Tell Number Tell Number Tell Find Configuration Programs Deconcer Generation Find Configuration Programs Number Tell Number Tell Number Tell Number Tell Number Tell Number Tell Societ Provide Number Tell Societ Provide Number Tell Societ Provide Number Tell Societ Provide Provide Tell Number Tell Societ Analysis Number Tell Societ Analysis Number Tell Societ Analysis Number Tell Societ Analysis Number Tell Cold Analysis Cold Ana										
image: state of the state o	Purpert Day 7 × Project Day 7 × Configuration Manager Platform Activit(MiXI2) Canfiguration Manager View wastCoppe View Winding Directory Manager Treet Note Nucleo Information Note Nucleo Information Note Nucleo Information Note Social Analysis Note										
Biology et lowers Construction Allow Allow Biology et lowers Configuration Manager Allow Biology et lowers Configuration Manager Allowers Biology et lowers Configuration Manager Allowers Command Agreements Configuration Manager Allowers Command Agreements Contact Data Allowers Command Agreements Contact Data Allowers Command Agreements Contact Data Allowers Debager by the Auto Source Contail Contact Data Debager by the Auto Source Contail Contact Data Source Contail Source Contail Source Contail Source Contail Fammet Biology Brance Manage Data Fammet	pt de tota biolo de la delagrandia de grande d							ata Build			
tor Partorne Active(Wn2)2	pt							Rebuild			
bi v Pattern Active(WnR2) v Cerfiguration Margar Point Constraints Accelerator Science Sci	Configuration Manager. Week Market (Viniz) Configuration Manager. Configuration Manager. Configuration Properties Configuration Manager. Market (Viniz) Configuration Manager. Vector Configuration Manager. Configuration Manager. Market (Viniz) Configuration Manager. Vector Configuration Manager. Configuration Manager. Market (Viniz) Configuration Manager. Vector Configuration Manager. Configuration Manager. Configuration Manager. Status (Market Manager.) Vector Configuration Manager. Configuration Manager. Configuration Manager. Status (Market Manager.) Vector Configuration Manager. Configuration Manager. Configuration Manager. Status (Configuration Manager.) Vector Status (Configuration Manager.) Net Market Manager. Status (Configuration Manager.) Vector Configuration Manager. Net Net Market Manager.) Status (Configuration Manager.) Vector Status (Configuration Manager.) Net Market Manager.) Net Status (Configuration Manager.) Vector Status (Configuration Manager.) Net Market Manager.) <td< td=""><td></td><td></td><td></td><td></td><td></td><td>? ×</td><td>Clean</td><td></td><td></td><td></td></td<>						? ×	Clean			
tion Properties bion Deckurger to launch: Local Window Deckurger Local Window Deckurger Local Window Deckurger Local Window Deckurger Local Window Deckurger Command Local Window Deckurger Command Local Window Deckurger Command Local Window Deckurger Local Cock Local	pt Configuration Properties Configuration Properties Contradiations Debugger Contradiations D				and and the second			View			
der opgerten sonnen progente sonnen son order projection progente sonnen sonnen die Sonnen die	pt		Configuration Manager			1	Configuration Manager	Analyze			
Inclail Window Debugger Conversed cpp Declaring Conversed Conversed cpp Conversed Conversed cpp Status 10 Conversed stbal Versition Signature Conversed stbal Signature Signature Conversed stbal March 10 Signature Conversed Debugger Ande Conversed Conversed Signature Conversed Conversed Conversed March 10 Conversed Conversed Signature Conversed Conversed Conversed <	Josef and Local Workshop Debugger Local Workshop Debugger rep rep Local Workshop Debugger Correnaid Local Workshop Debugger rep rep Correnaid Correnaid Local Workshop Debugger Restart Reputs rep Market Tool Market Tool Local Workshop Debugger Restart Reputs rep Market Tool Market Tool Local Workshop Debugger Restart Reputs rep Market Tool Market Tool Local Workshop Debugger Restart Reputs rep Market Tool Market Tool Local Workshop Debugger Restart Reputs rep Market Tool Market Tool Local Workshop Debugger Restart Reputs rep Market Tool Local Market Tool Advite Restart Reputs Restart Reputs Status Expenses Local Market Tool No Res Restart Reputs Restart Reputs Status Expenses Local Market Tool No Res Restart Reputs Restart Reputs Restart Reputs Status Expenses Local Market Tool No Res Res Res Res Res Res R				Debugger to launch:			Project Only			
ging Dictorions: a thol command Agraments Command Agraments Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command Command	put to the control of the segments to part to the application.				Local Windows Debugger		v)	Retarget Projects			
st Tool construction of opported to the design of the desi	put to the second secon				600,000,000,000,000,000,000			Overview			
nt hold counter Granter biologen biolo	put Command Arguments to parts to the application.							Scone to Thir			
at Tool Assch 16 of Carlos 16	b) Municit Tedi Writing Directory Birgeschi Birgeschi </td <td></td> <td></td> <td></td> <td></td> <td></td> <td>*</td> <td></td> <td></td> <td></td> <td></td>						*				
AddConstantial AddSon Add Add Add Add Add Add Add Add Add Ad	put										
International Environment Kano Gen Kano Seen Kano Se	put I Construction of the segment to part to										
Ruled Step oudyis: Merge Environment Vist So Clebageing No Amp Default Accelerator WARP schware accelerator WARP schware accelerator WARP schware accelerator	put Mage Exclusions 1/2 Ve. S. Cade Analysis S. Cade Analysis S. Cade Analysis D. Cade Analysis S. Cade Analysis D. Cade Analysis S. Cade Analysis D. Cade Analysis S. Cade Analysis D. Cada Analysis D. Cada Analysis D					Auto			mand		
SSL Debugging No Ang Default Accelerator WJAPP schware accelerator Source Control Partice Ang Default Accelerator WAPP schware accelerator Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Partice Par	put SCL Ochragong No SCL Ochragong No Amp Orlault Accelerator Source Control Source Contro					Max			Ctri+Shift+X		
Amp Default Accelerator WAPP schware accelerator WAPP schware accelerator X Core Core X Core X Core X Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannee Mannnee Mannee Mannee Mannee Mannee Mannee Mannee Mannn	put										
Source Control Source Control Paular Contro	put other of the spin of the s			p Code Analysis							
	put OK Cancel Apply - + X										
Polite Coli+1 E Mannove Del antig Project Propuries Inname A	put OK Cancel Apply - + 9 x										
X Parnove Del writig Project Properties □ Parame	put OK Cancel Apply + X									Sploter Team Expl	
Rename Santy Project repertues	put OK Cencel Apply V V V V										
E Rename 2	c s but application.									onfig Project Proper	ties
The face of the fa	put										
	put OK Cancel Apply - + X									- 1	
Rescan Solution A: ATICMConfig	e c command Arguments The command line arguments to pass to the application. put OK Cencel Apply.							Rescan Solution			ATICHConfin
	c c c c c c c c c c c c c c c c c c c							Open Folder in File Explorer		dencies	Artericoning
	e command line arguments to pass to the application. Red Managelee A7(cHConfig				Command Arguments		1	F Properties	Alt+Enter		
	put OK Cancel Apply				The command line arguments to pass	to the application.		-	Roo	t Namespace	
Recan Solution 0	put OK Cancel Apply C. 9 X							Rename Unload Project Rescan Solution C Open Folder in File Explorer	Alt+Enter) 1	
	the command line arguments to pass to the application. Control Apply OK Cancel Apply + 9.x				Command Arguments				Alt+Enter		
Command Arguments	ad OK Cencel Apply - 9 X			2.00	The command line arguments to pass	to the application.			Roo	t Namespace	A71CHConfig
Command Arguments The command line arguments to pass to the application. Root Numarplese ATI-ChrugUATICH_Full_VIAL Root Numarplese ATICHConfig				[s] 3]							
Command Arguments The command line arguments to pass to the application. Properties AN-Enter ChronyATTCH_Fall_v1A The command line arguments to pass to the application.		out				OK	Cancel Apply		Ψ×		
Command Arguments Computer Managements The command fine arguments to pass to the application. Root Nameputer >> ArcTiceConfig											
Command Arguments Computer Managements The command fine arguments to pass to the application. Root Nameputer >> ArcTiceConfig		an output notice									
Command Arguments Computers Computers Computers Computers Computers An-Toter 1 Computers Computers Computers ATICHConfig											
Command Arguments Computers Computers Computers Computers APTICNE (pully) ALL > The command fire arguments to pass to the application. Root Nameputer APTICNE (pully) ALL APTICNE (pully) ALL											
Command Arguments Computers Computers Computers Computers Computers An-Toter 1 Computers Computers Computers ATICHConfig											
Command Arguments The command Line arguments to pass to the application. Canced Apply Canced Apply • 4 x											
Command Arguments The command line arguments to pass to the application. → Computer to pass to the application. Concel Apply • 0 × Read									Specifi	ies the project name.	
Command Arguments The command line arguments to pass to the application. Canced Apply Canced Apply • 4 x											
Compared Arguments The command line arguments to pass to the application.											Add to Source Cor

Fig 32. Visual Studio 2017. Building the A71CHConfig project.

The resulting executable will be named '*A71CHConfig_vcom.exe*' and will be in directory '*tools*'. To run the obtained executable, open a PowerShell window in '*tools*' folder. This can be easily done by right clicking the folder while pressing the shift key. Once in the PowerShell window, it is possible to search and list all the existing .exe files (executables) with the following command:

dir *.exe

As shown in Fig 33, the previously built '*A71CHConfig*' project generated an executable file inside the 'tools' folder.

Direct	ory: C:\nxp\A7	1Сн_∨1.4.0.0∖	tools	
lode	Last	WriteTime	Length	
 -a -a	6/12/2018 6/12/2018 5/29/2018	11:51 AM	1513472	A71CHConfig_socket.exe A71CHConfig_vcom.exe mbedTLS_ax_ssl_client2_vcom.exe
a	5/29/2018		817664	mbedTLS_sa_ssl_server2.exe

Finally, the A71CH Configure tool can be run with the following command:

.\A71CHConfig_vcom.exe COM4 <input_arguments>

The A71CH Configure tool will be launched and the connection between the Windows platform and the A71CH over the Kinetis board will be established.

7.3 Running A71CH Host API usage example

The A71CH Host API usage examples are built with Microsoft Visual Studio:

- 1. In this case, right-click '*mainA71CH*' in the 'Solution Explorer' tab and open the '*Properties*' window.
- 2. Then, in '*Configuration Properties Debugging*' tab make sure the virtual COM address is set in the 'Command Arguments' field.
- 3. Once the VCOM port has been defined, invoke menu *Build Build Solution* to create the executable.

Finally, the A71CH Host API usage example can be executed in two ways.

7.3.1 Run the A71CH Host API usage executable from a terminal

An executable named 'mainA71CH.exe' will be generated in '/bin/Debug' folder

🔀 Windows Power				
PS C:\nxp\A71CH_v1.4.0.0\vs2017_projects\A71CH_vcom_openss1\bin\Debug> dir *.exe				
Directo	ry: C:\nxp\A71CH_v1.	4.0.0\vs2017_proj	ects\A71CH_vcom_openss1\bin\Debug	
Mode	LastWriteTi	me Length	Name	
 -a -a	6/12/2018 11:51 4/23/2018 12:30 6/12/2018 12:13	РМ 1642496	A71CHConfig.exe hlseA71CH.exe mainA71CH.exe	
Fig 34. main	A71CH executable file) .		

To launch it, open a PowerShell by right clicking the '*Debug*' folder while pressing the shift key. Then, start the executable with the following command:

.\mainA71CH.exe COM6

Where 'COM6' is the number assigned to the Kinetis configured as a virtual COM port. The A71CH Host API usage example will be launched and the connection between the Windows platform and the A71CH over the Kinetis board will be established (Fig 35).

PS a71 ***	C:\nxp\A71CH_v1.4.0.0\vs2017_projects\A71CH_vcom_openss1\bin\Debug> .\mainA71CH.exe COM6 ch HostLibrary example application (Rev 1.30:1.31)
Con	nect to A71CH-SM. Chunksize at link layer = 256.
UPE ATR	ning COM Port '\\.\COM6' =0xB8.04.11.01.05.04.B9.02.01.01.BA.01.01.BB.0C.41.37.31.30.78.43.48.32.34.32.52.31.BC.00
Hos	tLib Version : 0x0130
App	let Version : 0x0131 ureBox Version: 0x0000
sec	
===	======SELECT-DONE=======
Sta	rt exAes()
Sta	rt exAesRfc3394Precooked(Reset)
	chInitModule(Reset) et A71CH.
Res	et A/ICH.
A71	_setsymKey(0x00)
aes DR•	Ref[indexAesKey] (LEN=16): FE:E9:E3:B2:76:15:4D:67:F9:D8:4C:B9:35:54:56
	_SetSymKey(0x01) Ref[indexAesKey] (LEN=16):
00:	Ner[Indexaskey] (Len=10). 01:02:03:04:05:06:07:08:09:0A:0B:0C:0D:0E:0F
.71	C-+C
A/ 1 aes	_SetSymKey(0x02) Ref[indexAesKey] (LEN=16):
c0.	79:EF:82:CD:F7:12:F2:87:28:FD:18:ED:D7:F2:E4

Fig 35. Host API usage application execution from a Power Shell terminal.

7.3.2 Run the A71CH Host API usage from Visual Studio

Alternatively, the A71CH Host API usage application can be directly executed from the Microsoft Visual Studio. For this, just click on the '*Local Windows Debugger*' as highlighted in Fig 36. A new window will pop-up showing the execution. Also, the user can add or remove test points to debug the code.

Application note

COMPANY PUBLIC

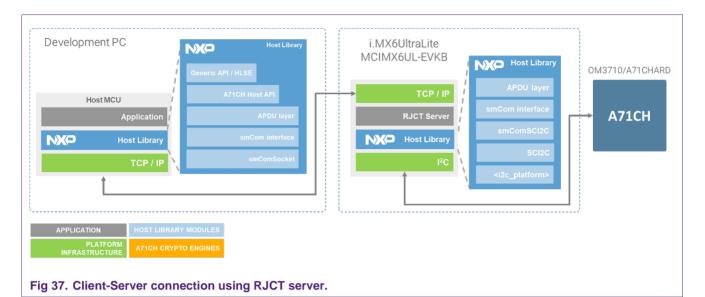
ainA71CH	 (Global Scope) Ø main(int argc, char** argv) 	 Solution Explorer O O Q H + O + S Ø IQ O ≠ 	- 1
1		Search Solution Explorer (Ctrl+)	
2	* Copyright(C) NXP Semiconductors, 2016 * All rights reserved.	Solution '_openssl_a71ch_vcom_x86' (13 projects)	
4	C:\nxp\A71CH v1.4.0.0\vs2017 projects\A71CH vcom openss\bin\Debug\mainA71CH.exe	- 0 X	
5	* Software that is described herein is for illustrative put * which provides customers with programming information rea7ich HostLibrary example application (Rev 1.30:1.31)	^	
7	A AT applies concerning the This experience and "AS I		
8	* warranties of any kind, and NXP Semiconductors and its 1 Opening COM Port '\\.\COM6'		
9	 all warranties, express or implied, including all implie ATR=0xB8.04.11.01.05.04.B9.02.01.01.BA.01.01.BB.0C.41.37.31.30.78. 	43.48.32.34.32.52.31.BC.00.	
10	* merchantability, fitness for a particular purpose and no HostLib Version : 0x0130 * intellectual property rights. NXP Semiconductors assume Applet Version : 0x0131		
12	 intellectual property rights. NXP Semiconductors assume Applet Version : 0x0131 or liability for the use of the software, conveys no lic SecureBox Version: 0x0000 		
13	* patent, copyright, mask work right, or any other intelle		
14	* or to any products. NXP Semiconductors reserves the righ		
15 16	* in the software without notification. NXP Semiconductors * representation or warranty that such application will be		1
10	specified use without further testing or modification. Start exAes()		
18	*		1
19	* Permission to use, copy and modify this software is here		1
20	* under NXP Semiconductors' and its licensor's relevant co * the software, without fee, provided that it is used in c		
21	* The software, without ree, provided that it is used in c. ***********************************		1
23	* must appear in all copies of this code		
24	Reset A71CH.		
25	A71 SetSymKey(0x00)		
	//=CC29MkEY(0X00) aesRef[indexAesKey] (LEN=16):		
ut	DB:FE:E9:E3:B2:76:15:40:67:F9:D8:4C:B9:35:54:56		
r output fro	m: Debug • 1 字 1 字 2 字 2 / 2 / 2 / 2 / 2 / 2 / 2 / 2 / 2		
	aesRef[indexAesKey] (LEN=16):		
	00:01:02:03:04:05:06:07:08:09:0A:08:00:0E:0F		
		~	6 - C
		*	
List Outp			

8. RJCT server

The A71CH Host software package includes a Remote Java Card Terminal server (RJCT). The RJCT server is a standalone process that can establish a communication session with a Secure Element on behalf of a client process. Therefore, the RJCT server can be used to communicate with A71CH from a remote or local client process.

Fig 37 illustrates the communication between client process running on a remote development PC and the A71CH connected to an embedded target. As an example, the i.MX6UltraLite will be used as the embedded platform. A quick start guide explaining how to setup the i.MX6UltraLite to start developing with the A71CH can be found in [QUICK_START_IMX6].

AN12134



The following steps must be followed to establish a communication between a Windows platform (remote development PC) and the i.MX6UltraLite (embedded platform) over the RJCT server. In this example, the A71CH Configure tool will be executed.

1. Open a Tera Term terminal to interact with the i.MX6UltraLite platform (Fig 38, 6.3 [QUICK_START_IMX6]). Make sure the i.MX6UltraLite is connected to the ethernet.

File Edit Setup Control Window Help
ALSA device list: #0: wm8960-audio EXT3-fs_(mmcblk1p2): error: couldn't mount because of unsupported optional featu
res (240) EXI2-fs (mmcblk1p2): error: couldn't mount because of unsupported optional featu
w0: wm0500-aud10 EXI3-fs (mmcDkLp2): error: couldn't mount because of unsupported optional featu res (240) EXI2-fs (mmcDkLp2): error: couldn't mount because of unsupported optional featu res (244) EXI4-fs (mmcDkLp2): recovery complete EXI4-fs (mmcDkLp2): mounted filesystem with ordered data mode. Opts: (null) VFS: Mounted root (ext4 filesystem) on device 179:2.
Freeing unused kernel memory: 432K (80b14000 - 80b80000) INII: version 2.88 booting
Starting udev udevd[125]: starting version 3.1.5
random: udevd urandom read with 14 bits of entropy available EXT4-fs (mmcblk1p2): re-mounted. Opts: (null) bootlogd: cannot allocate pseudo tty: No such file or directory
Populating dev cache tar: dev/disk/by-label/Bootx20imx6ul: Cannot stat: No such file or directory
tar: Existing with failure status due to previous errors udev-cache: update_failed!
INIT: Entering runleyel: 5
Configuring network interfaces fec 20b4000.ethernet eth0: Freescale FEC PHY d river [Micrel KS2008] or KS20091] (mii bus:phy_addr=20b4000.ethernet:01, irq=-1) [Pv6: ADDRCONF(NETDEV_UP): eth0: link Is not ready udhcpc (v1.24.1) started
udhopo (v1.24.1) started Sending discover
Sending discover Sending discover Sending discover No lease, forking to background
dope
Starting system message bus: dbus, Starting OpenBSD Secure Shell server: sshd done.
Starting rpcbind daemondone. starting statd: done Starting atd: OK
exportfs: can't open /etc/exports for reading
NFS daemon support not enabled in kernel Starting system log daemon0 Starting kernel log daemon0 Starting crond: OK
Starting crond: OK
Freescale i.MX Release Distro 4.1.15-2.0.3 imx6ulevk /dev/ttymxc0
imx6ulevk login: root root@imx6ulevk:~#

- 2. Navigate to the folder where the RJCT executable is located: root@imx6ulevk:~# cd axHostSw/linux
- 3. Check the IP address with the following command:

root@imx6ulevk:~# ifconfig

In this case, the i.MX6UltraLite address is 192.168.10.122

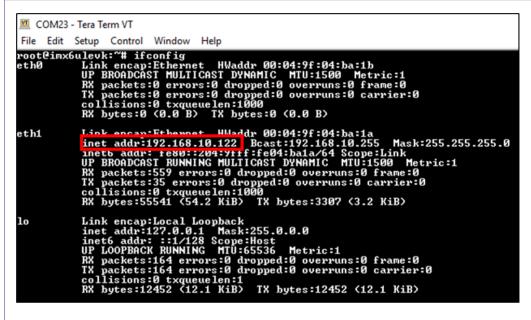


Fig 39. Check i.MX6UltraLite IP address.

4. Launch the RJCT server:

root@imx6ulevk:~# ./rjct_a71_i2c_imx

🔟 C	OM23 - Tera Term VT	
File	Edit Setup Control Window Help	
Remo A71 ××××	<pre>@imx6uleuk:~/axHostSw/linux# ./rjct_a71_i2c_imx teJCShell Terminal Server (supporting SCI2C) Rev 0.92 (I2C_CLK_MAX = 400 kHz) - Effective Master Clock depends on Host Platfon ************************************</pre>	ካጠ . (አአ
Serv	er: waiting for connections	
Fia 4	0. Launch RJCT server in the i.MX6UltraLite.	

The RJCT starts listening to incoming APDUs from the remote development PC. Once in the Windows platform:

- 1. Open the Visual Studio solution '_openssl_a71ch_socket_x86.sln'. Note that in this case we will be using the '_socket_' solution since the APDUs will be sent over TCP/IP to the i.MX6UltraLite.
- 2. Select the '*A71CHConfig*' project and set 'localhost:8050' in its properties, as shown is shown in Fig 32. Then, build the selected project.
- 3. An executable file called 'A71CHConfig_socket' will be generated in the 'tools' folder (Fig 33).
- 4. Open a Power Shell by right clicking the 'tools' folder while pressing the shift key. Then, start the executable with the following command:

.\A71CHConfig_socket.exe 192.168.10.122:8050 interactive

It will launch the A71CH Configure tool in interactive mode (Fig 41).

			<mark>ket.exe</mark> 192.168.10 size at link layer	0.122:8050 interactive
WSAStartup: OK	.10) connect t	O A/ICH. Chun	Size at Tink Tayer	- 250.
send: ATR 0000000400000100				
full recv: 0000	00153BFB180000813			
recv: 3BFB18000 send: 00A404000	08131FE45504C4143	45484F4C44455	2AB	
recv: 01319000	2012/21020900			
		.4c.41.43.45.4	8.4F.4C.44.45.52.	AB.
HostLib Version : Applet Version :				
SecureBox Version:				
========SELECT-[ONE			
HostLib Version	: 0x0130).		
Annlet-Rev: Secure	ox-Rev : 0x0131	:0x0000		

Fig 41. A71CH Configure tool executed in interactive mode.

As can be observed in Fig 42, the RJCT server (in the i.MX6UltraLite) establishes a connection with the development PC, receives the packed APDUs from the development PC and will re-direct these APDUs to the A71CH.

🔟 COM23 - Te	ra Term VT			_	
File Edit Setu	p Control Window He	elp			
RemoteJCShel 171 (I2C_CLI ********** Establish a /ten e.g	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	<pre>(supporting SCI20 Effective Master ************************************</pre>		Platfor	יח <u>-</u> אאי
Server: got [2CInit: ope [2C driver: [2C driver s [2C driver s	ing for connection connection from 19 ning /dev/i2c-1 PEC flag cleared upports plain i2c- upports Read Block 18.00.00.81.31.FE.	2.168.10.46 level commands.	5.48.4F.4C.44.45.52.AB		
	ection established w				

Then, the response from the A71CH will be sent to the development PC and printed in the PowerShell terminal. For instance, Fig 43 shows the command '*info all*' executed from the development PC. The response to this command is prompted in the terminal after being sent to the i.MX6UltraLite and consequently to the A71CH.

PS C:\nxp\A/1CH_v1.4.0 a/1chConfig (Rev 1.10)	<pre>D.O\tools> .\A71CHConfig_socket.exe 192.168.10.122:8050 interactive connect to A71CH. Chunksize at link layer = 256.</pre>
VSAStartup: OK	
send: ATR 0000000400000100	
full recv: 00000015	3BFB1800008131FE45504C414345484F4C444552AB
	31FE45504C414345484F4C444552AB
send: 00A4040005613 recv: 01319000	// 31636800
ATR=0x3B.FB.18.00.00.8	31.31.FE.45.50.4C.41.43.45.48.4F.4C.44.45.52.AB.
lostLib Version : 0x0	
Applet Version : 0x0 SecureBox Version: 0x0	
securebox version. oxc	000
======SELECT-DONE=	
HostLib Version Applet-Rev:SecureBox-R	: 0x0130
>>> info all	. 0.00151.000000
sena: 8091000300	
recv: 0131F32210009 send: 8091000100	000
	210800401198998905348129000
A71CH in Debug Mode Ve	ersion (SCP03 is not set up)
selectResponse: 0x01	
injectLockState: 0x	x03 (Transport Lock NOT YET set) x02 (Unlocked)
gpStorageSize: 40)96
id (LEN=18):	0 00 04 01 10 00 00 00 52 40 12
1/:90:/0:02:4/:91:12:1 certUid (LEN=10):	0:80:04:01:19:89:98:90:53:48:12
0:02:80:04:01:19:89:9	8:90:53
send: 8011000000	
recv: 6985 send: 8011010000	
recv: 6985	
send: 8011020000	
recv: 6985 send: 8011030000	
recv: 6985	
Public Keys from ECC k	ey pairs:
idx=0x00 n.a. idx=0x01 n.a.	
idx=0x01 h.a.	
idx=0x03 n.a.	

Fig 43. 'info all' command executed from the development PC.

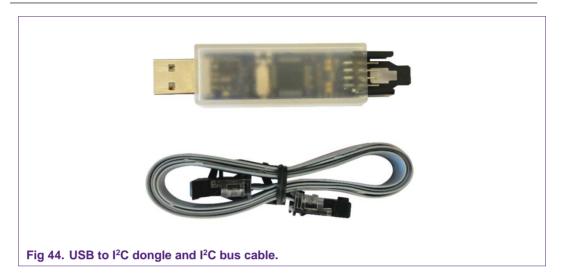
9. Appendix A: Running A71CH application examples with USB to I²C bird

The USB to I²C bird – Ascot adaptor (OM3710/B001) can be used to interface a Windows platform as an alternative to a Kinetis board configured as a virtual COM port. In this case, the development PC will be connected to the A71CH Mini PCB board (OM3710/A71CHPCB) through the USB to I²C bird (OM3710/B001). This appendix presents how to setup the involved hardware and software to execute A71CH applications using the USB to I²C bird (OM3710/B001).

9.1 USB/I²C bird (OM3710/B001)

The OM3710/B001 board is an I²C to USB converter enabling the connection of an A71CH via USB to a PC. The OM3710/B001 comes with a USB I²C Bird / Ascot adaptor and I²C bus cable.

AN12134



Note: For ordering information: please contact your local sales representative.

9.2 Hardware setup

Connect the A71CH mini PCB to the USB to I²C Bird as shown in the following Fig 45. Please note the connector polarity.



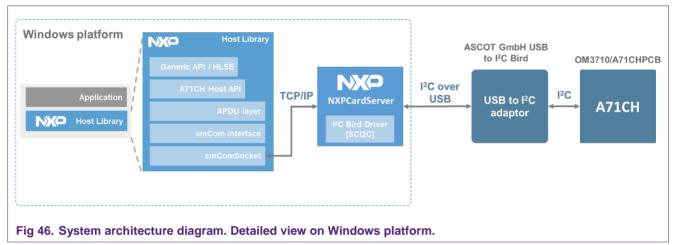
Fig 45. USB to I²C Bird – A71CH mini PCB interface connection.

Finally, connect the USB to I²C Bird to the Windows platform USB port.

9.3 Software setup

Fig 46 illustrates the system architecture, including a detailed view of the windows platform and illustrating the software layers involved when using the USB to I²C Bird solution.

A local server will be created to listen to the IP address where the A71CH application example APDUs are sent. This local server will be created using the *NXPCardServer* java application contained in the A71CH Host software package and will be in charge of waiting for requests from the IP address, capturing and prompting the APDU commands in a window, and re-directing these APDUs to the USB to I²C adaptor I²C address. As such, a TCP/IP address needs to be passed as an argument when running an A71CH example application.



The following steps are required to run the A71CH Windows-based applications using the USB to I^2C bird:

- Install USB to I²C adaptor drivers.
- Open a local server with NXPCardServer java application. Local server will listen to localhost:8050 port and will redirect data to I²C address 90h. This is a transparent operation for the user.
- Build the target A71CH application with Microsoft Visual Studio.
- Start the built application from a Windows terminal passing the *localhost:8050* address as input argument.

9.3.1 USB to I²C ASCOT adaptor – Drivers

To use the USB to I²C Ascot adaptor, it is necessary to install the required drivers. These can be found in [I2C_BIRD].

Install the executable for your processor (either 32 or 64 bits) and follow the setup wizard until it is finished. Once the driver has been correctly installed, the USB to I²C adaptor can be used to establish a communication between the Windows platform and the A71CH contained in the OM3710/A71CHPCB. Fig 47 shows a capture of the setup wizard.

1	URT 4.0.1 — — — ×
<u>_</u>	Select Installation Folder
_	
т	The installer will install URT 4.0.1 to the following folder.
T	To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".
	Eolder: C:\Program Files (x86)\ASCOT GmbH\URT 4.0.1\ Browse
	Disk Cost
	Install URT 4.0.1 for yourself, or for anyone who uses this computer:
	Everyone
	⊖ Just me
	Cancel < Back Next >
Fig 47. USB to I ² C A	Ascot adaptor. Drivers installation wizard.

9.3.2 NXPCardServer

NXPCardServer is a Java application that is called by using '*javaws*' command in the Command Prompt. This command launches Java Web Start, which executes Java applications or applets hosted on a network and creates a local server instance. This local server will be constantly listening to the *localhost:8050* IP address, and will re-direct the input commands to the I²C address of the USB to I²C Ascot adaptor.

The NXPCardServer Java application requires a 32-bit Java Runtime Environment (JRE). It is known to be compatible with Java6, Java7 and Java8.

All the required files for launching the NXPCardServer application can be found in *'ext/NXPCardServer'* folder of the A71CH Host SW package. This folder contains several windows batch files (.bat) and application extensions (.dll), as shown in Fig 50.

When working with the USB to I²C adaptor connected to a Windows platform, the '*NXPCardServer_I2C_Bird.bat*' file should be run to execute the '*javaws*' command and start the server.

This .bat file will call the '*NXPCardServer.bat*' executable with the I²C Bird configuration. '*NXPCardServer.bat*' will first check if Java Runtime environment is installed in the Windows device. Then, the following environment variable will be automatically defined:

• JRE_HOME=%PROGRAMFILES(X86)%\Java\jre_version

Where **jre_version** belongs to the installed JRE version. This variable defines where the JRE is installed and where the '*javaws*' is. Therefore, this must be correctly defined to call the '*javaws*' function and open the local server.

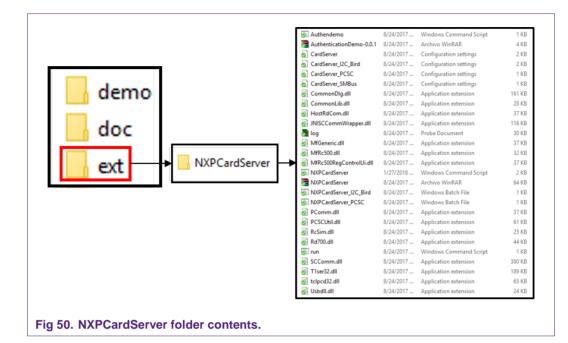
Variable	Value
HOME	C:\Temp
JAVA_BIN	C:\Program Files (x86)\Java\jre1.8.0_161\bin
JRE_HOME	C:\Program Files (x86)\Java\jre1.8.0_161
MOZ_PLUGIN_PATH	
OneDrive	C:\Users\aleja\OneDrive - MobileKnowledge
Path	C:\Users\aleja\AppData\Local\Microsoft\WindowsApps;
TEMP	C:\Users\aleia\AppData\Local\Temp
	New Edit Delete
ystem variables	
Variable	Value
OA_PLUGIN_PATH	%CDSROOT%\Share\oaPlugins
OPENSSL_CONF	C:\OpenSSL-Win32\bin\openssl.cfg
OS	Windows_NT
Path	C:\Program Files (x86)\Intel\iCLS Client\;C:\Program Files\Intel\iCL
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	
PROCESSOR IDENTIFIER	Intel64 Family 6 Model 78 Stepping 3. GenuineIntel
	New Edit Delete
	OK Cancel

The user should make sure that JRE_HOME variable has been correctly created or edited by opening the system environment variables editor of Windows. Fig 48 shows the system environment variables editor on Windows. As can be observed, JAVA_BIN and JRE_HOME variables are defined in the user variables window, while PATH variable includes the contents of JAVA_BIN (right figure).

Finally, the '*javaws*' command will be called and the NXPCardServer Java applet will be started. If NXPCardServer is successfully launched, the window presented in Fig 49 will be opened and the server will start to listen to the localhost:8050 IP address.

SmartCard' Server (v2.0.0.3); Reader Type: I2C (I2C-Addr. 90h)	-	Х
1. Server successfully opened, wait for request on host: 127.0.0.1, port: 8050		
Fig 49. NXPCardServer execution window.		

In case these environment variables are not properly defined, the '*javaws*' command will not be executed; thus, the server will not be started.



9.4 Running A71CH Configure tool

To run the A71CH Configure tool:

- 1. Open the Visual Studio solution '_openssl_a71ch_socket_x86.sln'. Note that in this case we will be using the '_socket_' solution since the APDUs will be sent over TCP/IP to the NXPCardServer application.
- 2. Right-click '*mainA71CH*' in the 'Solution Explorer' tab and open the '*Properties*' window.
- 3. Then, in 'Configuration Properties Debugging' tab make sure the *IP* address *localhost:8050* is set in the 'Command Arguments' field.
- 4. Once the IP address has been defined, invoke menu *Build Build Solution* to create the executable.
- 5. An executable file called 'A71CHConfig_socket' will be generated in the 'tools' folder.

Considering the NXPCardServer is running, open a PowerShell by right clicking the 'tools' folder while pressing the shift key. Then, start the A71CH Configure tool with the following command:

.\A71CHConfig_socket.exe localhost:8050 debug reset

And prompt the A71CH status information with:

.\A71CHConfig_socket.exe localhost:8050 info status

AN12134

	vs PowerShell							1	
			.\A71CHConfig_				reset		
71chCo	nfig (Rev 0.94)	conne	ect to A71CH. Chi	unksize a	it link layer =	256.			
			DESKTOP-2QFJ9KM.						
	Version Rev:SecureBox-R		(0130 (0131:0x0000						
			> .\A71CHConfig_	corket e	va localhoct:805	0 info	etatue		
71chCo	nfig (Rev 0 94)	conne	ect to A71CH. Chu	inksize a	t link laver =	256	status		
lient:	Client connect	ing to: D	DESKTOP-2QFJ9KM.	inter e	ie mini najen –				
ostLib	Version	: 0>	(0130						
	Rev: SecureBox-R	ev : 0x	(0131:0x0000						
	s Not enabled								
ey Pai	r status:								
	Index=0: Empty								
	Index=1: Empty								
	Index=2: Empty								
hlie	Index=3: Empty Key status:	open							
dorrie	Index=0: Empty	Open							
	Index=1: Empty								
	Index=2: Empty								
onfig	Key status:								
	Index=0: Empty	Open							
	Index=1: Empty								
	Index=2: Empty	Open							
ym Sec	ret status:								
	Index=0: Empty Index=1: Empty	Open							
	Index=1: Empty Index=2: Empty								
	Index=3: Empty								
	Index=3: Empty Index=4: Empty								
	Index=5: Empty								
	Index=6: Empty								
	Index=7: Empty								
ounter	status:								
	Index=0: Initi								
	Index=1: Initi								
eneral	Purpose Storag				0.0010	a	0.0000		
	Offset=0x0000: Offset=0x0080:		Offset=0x0020: Offset=0x00A0:		Offset=0x0040: Offset=0x00C0:		Offset=0x0060: Open Offset=0x00E0: Open		
	Offset=0x0100:		Offset=0x00A0:		Offset=0x00c0:		Offset=0x00E0: Open		
	Offset=0x0180:		Offset=0x0120:		Offset=0x0140:		Offset=0x01E0: Open		
	Offset=0x0200:		Offset=0x0220:		Offset=0x0240:		Offset=0x0260: Open		
	Offset=0x0280:		Offset=0x02A0:		Offset=0x02C0:		Offset=0x02E0: Open		
	Offset=0x0300:	Open	offset=0x0320:	Open	Offset=0x0340:	Open	Offset=0x0360: Open		
	Offset=0x0380:		Offset=0x03A0:		Offset=0x03c0:		Offset=0x03E0: Open		
	offset=0x0400: offset=0x0480:		Offset=0x0420:		Offset=0x0440:		Offset=0x0460: Open		
			Offset=0x04A0:		offset=0x04c0:		Offset=0x04E0: Open		

Fig 51. A71CH Configure tool execution using the USB to I^2C bird.

9.5 Running A71CH Host API usage example

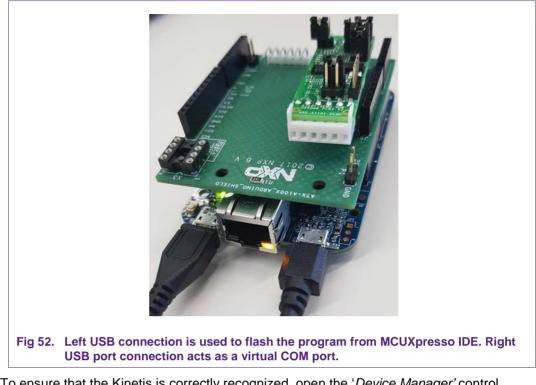
Similarly, the A71CH Host API usage example can be built with Microsoft Visual Studio and launched from a PowerShell terminal. The following command can be used to start it:

.\mainA71CH localhost:8050

10. Appendix B: VCOM driver installation troubleshooting

If the Kinetis board is connected (Fig 52) and has been configured to perform as an USB to I²C adaptor, the Windows platform should detect and assign it a virtual COM port number.

AN12134

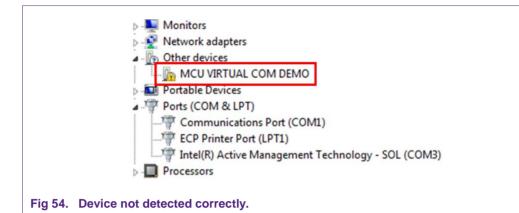


To ensure that the Kinetis is correctly recognized, open the '*Device Manager*' control panel. The Kinetis board should be detected and labeled as '...VCOM Port (COMX)' within the '*Ports* (COM & LPT)' drop-down (Fig 53).



If the Windows OS does not recognize the Kinetis board, it would appear labeled as '*MCU VIRTUAL COM DEMO*' within '*Other devices*' as shown in Fig 54. In this case, it means that the drivers need to be updated.

AN12134



In order to update the drivers, follow the next steps:

- 6. Right-click on 'MCU VIRTUAL COM DEMO'.
- 7. Click on Update Driver.
- 8. 'Browse my computer for driver software'.
- 9. 'Let me pick from a list of device drivers on my computer'.
- 10. Select Ports (COM & LPT).
- 11. Un-check 'Show compatible hardware'.
- 12. Select 'NXP' and 'LPC USB VCOM Port'.
- 13. Ignore the warning message and click on 'Yes'.

11. Appendix C: Running examples using Cygwin

This appendix specifies how to install Cygwin and use it to run the examples and shell scripts available in the A71CH support package from a Windows machine without the need of using Visual Studio IDE.

Cygwin is a large collection of GNU and Open Source tools which provide functionality like a Linux Distribution on Windows. It consists of two parts; a Dynamic Link Library (DLL) and an extensive collection of software tools and applications that provide a Unix-like environment.

11.1 Cygwin Installation

Cygwin programs are installed by running Cygwin's setup program. This program downloads the necessary packages from different repositories in the Internet. To download this setup program, go to [CYGWIN]. Choose the adequate option for your system (either 32-bit or 64-bit) and click on it so the download can start. Save it in your machine and run it at any time to update or install Cygwin packages.

Once the setup program is executed, it will guide the user through the installation process. The first step is to choose how to install the packages, which can be done downloading it from the Internet, installing it from a local directory, or even downloading the packets without installing them. Choose '*Install from the internet*' and click '*Next*'. Once an installation directory is chosen, it is necessary to choose the Internet

AN12134

Connection type. As can be observed in Fig 55, different options can be chosen depending on the what the user considers appropriate.

	net Connection know how you want settings below.	it to connect to	o the internet. C	noose	Ε
0	Use System Proxy Se Direct Connection Use HTTP/FTP Prox	-			
	Proxy Host Port 80				
			< Back	Next >	Cancel

Then, a list of sites from where to download the packages is provided. Either select one of them or define a known site in the corresponding box. After a site is chosen, or defined, the list of packages will appear, as in Fig 56.

Select Packages Select packages to install						E
View Pending ~ Search	Clear			🔘 Кеер	Best O Sync Sync	Test
Up To Date Curre Not Installed	Bin?	Src?	Categories	Size	Package	^
Picked	nja	nja	Games	10k	2048-cli: 2048 game fo	orterminal
Skip	njis	nja	Debug	29k	2048-cli-debuginfo: De	ebug info for
Skip	nja	nja	Games	1.371k	2048-qt: 2048 game fo	r Qt/KDE
Skip	nja	n/a	Debug	2,257k	2048-qt-debuginfo: De	bug info for
Skip	nja	nja	Debug	3,976k	4ti2-debuginfo: Debug	info for 4ti2
Skip	njis	nja	Audio	109k	AtomicParsley: Comma	and-line pros
Skip	nja	njla	Debug	281k	Atomic Parsley-debugin	fo: Debug i
Skip	nja	n/a	Libs, Math	103k	CUnit: CUnit is a Unit t	esting frame
Skip	nja	nja	Debug	76k	CUnit-debuginfo: Debu	ug info for C
Skip	nja	nja	Devel, Libs, System, I	694k	ELFIO: ELF file reader	and produc
Skip	nja	n/a	GNOME	307k	GConf2: GNOME conf	figuration da
Skip	nja	n/a	Debug	679k	GConf2-debuginfo: De	bug info for
Skip	njis	nja	Net	31k	GeoIP: GeoIP C library	,
Skip	nja	nja	Net	24.604k	GeolP-database: Geol	IP Lite (free)
G Skip	nja	n/a	Debug	118k	GeoIP-debuginfo: Deb	ug info for (
Skip	nja	nja	Graphics	874k	GraphicsMagick: Grap	hicsMagick
Skip	nja	nja	Debug	3,486k	GraphicsMagick-debu	ginfo: Debu
Skip	nja	nja	Graphics	146k	ImageMagick: Image p	processing £ v
<						>
Hide obsolete packages						
				< Ba	ick Next>	Cancel
				< 88	ICK IVEXL>	Cancel

In the View tab, the user can select which packages are shown (the full set of packages, those with updates available, the ones that have not been installed, etc.). To download a certain package, click on it and it will be selected for downloading.

To make use of Cygwin with A71CH, it is necessary to download the following packages: *gcc, make, Openssl* and *Openssl-devel.* Fig 57 shows the packages that must be downloaded.

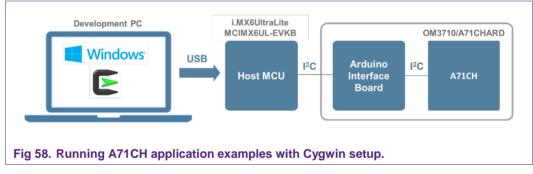
1.0.2o-1	Keep	nja	Net	4,997k open:	ssl-devel: A general purpose cryptography toolkit with TLS implementation (developme
4.2.1-2	🚯 Кеер	nja	Devel	449k make	: The GNU version of the 'make' utility
7.3.0-2	🚯 Кеер	nja	Libs	27k libgcc	1: GCC C runtime library
1.0.2o-1	Кеер	nja	Base, Libs	1,045k libope	enssl100: A general purpose cryptography toolkit with TLS implementation (runtime)
1.0.2o-1	Keep	nja	Base, Net	580k opens	ssl: A general purpose cryptography toolkit with TLS implementation

Fig 57. Package selection page.

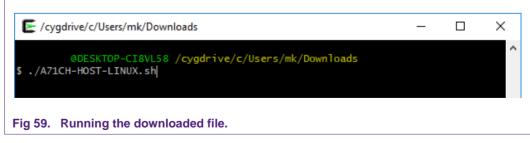
Once all the packages the user wants to download have been selected, click '*Next*'. Confirm the packages to be downloaded and click '*Next*' again. The setup program will start downloading the selected packages.

11.2 A71CH application examples execution

It has already been said that Cygwin allows working in a Unix-like environment. By using it, the examples available in the Host Library can be executed from a Windows machine without the need of running Visual Studio.



First, it is necessary to download the Bash Installer of the A71CH Host Software package [A71CH_HOST_SW]. An SH file named 'A71CH-HOST-LINUX' will be downloaded. Run the Cygwin terminal and go to the folder where the Bash Installer has been saved to run it (Fig 59).



This will create a folder named '*axHostSw*' where the Host Library will be located. Using Cygwin, move to the '*linux*' folder and build the A71CH binaries, as in Fig 60.

AN12134

AN12134



Once this script is executed, the following files will be installed in the 'linux' folder.

5 ls -1								
total 3412								
drwxrwxr-x+ 1	L fhe	erranzs	None	0	Jun	27	17:28	A71CH
-rw-rw-r+ 1	L fhe	erranzs	None	21862	Jun	27	14:19	A71CH_Engine.tgz
-rwxrwxr-x+ 1	L fhe	erranzs	None	655506	Jun	27	17:54	A71CH_socket_native.exe
-rwxrwxrwx+ 1	L Adn	ministrators	SYSTEM	0	Jun	27	17:28	a71ch_socket_native_dl.exe
-rwxrwxr-x+ 1	L fhe	erranzs	None	922654	Jun	27	17:28	a71chConfig_socket_native.exe
-rwxrwxr-x+ 1	L fhe	erranzs	None	760452	Jun	27	17:28	a71chHlse_socket_native.exe
-rwxrwxr-x+ 1	L fhe	erranzs	None	367507	Jun	27	17:28	A71CHlight_socket_native.exe
-rwxrwx+ 1	L fhe	erranzs	None	6390	Apr	5	16:21	buildA71CH.sh
-rwxrwxr-x+ 1	L fhe	erranzs	None	360281	Jun	27	17:28	libA71CH_socket_native.so
-rwxrwxr-x+ 1	L fhe	erranzs	None	360281	Jun	27	17:28	libA71CH_socket_native.so.1.3.0
-rwxrwx+ 1	L fhe	erranzs	None	16405	Jun	27	14:46	Makefile_A71CH
-rwxrwx+ 1	L fhe	erranzs	None	1409	Apr	5	16:21	packageA71chOpenSslEngine.sh

Fig 61. Files created with the buildA71CH.sh script.

This way, the user can use the A71CH Configure Tool, the Host API usage examples and the OpenSSL Engine examples from the Cygwin terminal. To do that, connect the i.MX6UltraLite board as explained in [QUICK_START_IMX6], check its IP address, as shown in Section 8, and run the Remote Java Card Terminal, also as explained in Section 8.

11.2.1 Running A71CH OpenSSL Engine examples

The OpenSSL Engine examples are available in the /hostLib/embSeEngine/ a71chDemo/scripts. For instance, the user can run the tlsPrepareClient.sh script to provision an A71CH security IC for TLS connection. In the Cygwin terminal, run the example specifying the IP address of the i.MX6UL evaluation board, as shown in Fig 62.

Application note

COMPANY PUBLIC

AN12134

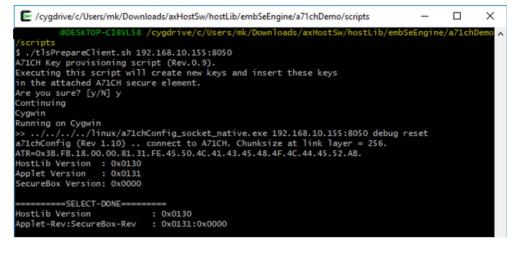


Fig 62. Running the A71CH provisioning example using Cygwin.

Once the script is run successfully, the A71CH will be provisioned and ready to establish a TLS connection to a server.

11.2.2 Running the A71CH Configure Tool

Using Cygwin, it is possible to use the A71CH Configure Tool without using Visual Studio. As an example, to run the '*info all*' command, just run the command shown in Fig 63 (with the appropriate IP address and port).

Application note

COMPANY PUBLIC

AN12134

/cygdrive/c/Users/mk/Downloads/axHostSw/linux	—		×
@DESKTOP-CI8VL58 /cygdrive/c/Users/mk/Downloads/axHostSw/linux			
./a71chConfig_socket_native.exe 192.168.10.155:8050 info all			
71chConfig (Rev 1.10) connect to A71CH. Chunksize at link layer = 256.			
TR=0x3B.FB.18.00.00.81.31.FE.45.50.4C.41.43.45.48.4F.4C.44.45.52.AB.			
ostLib Version : 0x0130			
pplet Version : 0x0131			
ecureBox Version: 0x0000			
======SELECT-DONE=======			
ostLib Version : 0x0130			
pplet-Rev:SecureBox-Rev : 0x0131:0x0000			
71CH in Debug Mode Version (SCPO3 is not set up)			
electResponse: 0x0131			
ransportLockState: 0x03 (Transport Lock NOT YET set)			
njectLockState: 0x02 (Unlocked)			
pStorageSize: 4096			
id (LEN=18): 7:90:70:02:47:91:12:10:80:04:01:02:95:98:90:53:48:12			
ertUid (LEN=10):			
0:02:80:04:01:02:95:98:90:53			
ublic Keys from ECC key pairs:			
idx=0x00 ECC_PUB (LEN=65):			
4:BF:42:B7:4F:FE:87:7C:5E:08:5E:FC:61:8D:9C:21:92:7B:2A:C5:08:8B:52:CA:D1	:41:EB:01:	5E:86:A	B:1F
2:5C:80:02:85:0E:9B:2A:0E:AB:A2:E6:F4:7D:A8:9F:5C:CE:2F:12:06:5B:E9:AB:E0	:02:65:E9:	32:6F:8	A:B9
0 idx=0x01 n.a.			
idx=0x01 h.a.			
idx=0x02 n.a.			
ublic Keys:			
idx=0x00 ECC_PUB (LEN=65):			
4:EE:15:80:C8:3B:64:32:A7:B9:19:C0:9A:70:DA:CF:D7:05:E8:F2:FC:A1:B3:DB:01	:03:F8:41:	D3:FF:9	8:F9
E:71:38:9A:B6:B3:85:36:B5:C7:74:E4:34:89:3A:31:D8:45:8E:9F:A7:A5:FB:DF:C0	· 1 / · DQ · 2E ·	11.71.0	0.36
	.44.09.22.	44.71.0	0.30
idx=0x01 n.a. idx=0x02 n.a.			
onotonic counter values:			
idx=0x00 0x0000000			
idx=0x00 0x00000000			
P Storage Data (128 segments from offset 0x0000):			
stan age state (and beginenes i for or bee on odd)			

11.2.3 Running A71CH Host API usage examples

Similarly, it is possible to run the Host API usage example by running the A71CH_socket_native.exe application. The information about the A71CH Host API usage examples can be found in [AN_A71CH_HOST_SW] and in the Doxygen documentation [A71CH_HOST_SW]. Fig 64 shows the result of this command in the Cygwin terminal.

Γ

A71CH Quick start guide for Windows

AN12134

E /cygdrive/c/Users/mk/Downloads/axHostSw/linux		×
<pre>@DESKTOP-CI8VL58 /cygdrive/c/Users/mk/Downloads/axHostSw/linux \$./A71CH_socket_native.exe 192.168.10.155:8050 a71ch HostLibrary example application (Rev 1.30:1.31)</pre>		
Connect to A71CH-SM. Chunksize at link layer = 256. ATR=0x3B.FB.18.00.00.81.31.FE.45.50.4C.41.43.45.48.4F.4C.44.45.52.AB. HostLib Version : 0x0130 Applet Version : 0x0131 SecureBox Version: 0x0000		
======================================		
Start exAes()		
Start exAesRfc3394Precooked(Reset)		
a71chInitModule(Reset)		
Reset A71CH.		
A71_SetSymKey(0x00) aesRef[indexAesKey] (LEN=16): DB:FE:E9:E3:B2:76:15:4D:67:F9:D8:4C:B9:35:54:56		
DB.FE.E9.E9.B2.70.11.40.07.F9.00.4C.B9.33.34.30		
A71_SetSymKey(0x01)		
aesRef[indexAesKey] (LEN=16): 00:01:02:03:04:05:06:07:08:09:0A:08:0C:0D:0E:0F		
A71_SetSymKey(0x02) aesRef[indexAesKev] (LEN=16):		
C0:79:EF:82:CD:F7:12:F2:87:28:FD:18:ED:D7:F2:E4		
A71_SetSymKey(0x03)		
aesRef[indexAesKey] (LEN=16):		
AA:BB:CC:DD:EE:FF:06:07:08:09:A0:B0:B1:B2:B3:B4		
A71_SetSymKey(0x04)		
<pre>aesRef[indexAesKey] (LEN=16):</pre>		
DB:FE:E9:E3:B2:76:15:4D:67:F9:D8:4C:B9:35:54:56		
A71_SetSymKey(0x05)		
ig 64. Running the A71CH Host API usage example using Cygwin.		

Application note COMPANY PUBLIC

12. References

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

[SCI2C]	SCI2C Protocol Specification Rev 1.5 or later (restricted to 1.x Revision, NOT compatible with Rev. 2.x)
[OPEN_SSL]	OpenSSL Cryptography and SSL/TLS Toolkit information www.openssl.org
[A71CH_APDU]	APDU Specification of A71CH Security Module - ds409420
[A71CH_HOST_SW]	A71CH Host Software Package (Windows Installer) – DocStore, document number sw4673xx ¹ , Version 01.03.00 (or later), available on <u>www.nxp.com/A71CH</u>
	A71CH Host Software Package (Bash installer) – DocStore document number sw4672xx ¹ , Version 01.03.00 (or later), available on <u>www.nxp.com/A71CH</u>
[FRDM_K64F]	Kinetis FRDM-K64F - https://www.nxp.com/products/processors-and- microcontrollers/arm-based-processors-and-mcus/kinetis- cortex-m-mcus/k-seriesperformancem4/k2x-usb/freedom- development-platform-for-kinetis-k64-k63-and-k24- mcus:FRDM-K64F
[FRDM_K82F]	Kinetis FRDM-K82F - https://www.nxp.com/products/processors-and- microcontrollers/arm-based-processors-and-mcus/kinetis- cortex-m-mcus/k-seriesperformancem4/k8x-secure/freedom- development-platform-for-kinetis-k82-k81-and-k80- mcus:FRDM-K82F
[MCUXPRESSO_IDE]	MCUXpresso IDE -https://www.nxp.com/support/developer- resources/software-development-tools/mcuxpresso-software- and-tools/mcuxpresso-integrated-development-environment- ide:MCUXpresso-IDE
[OPENSDA_FIRMWARE]	OpenSDA / OpenSDA V2 website - https://www.segger.com/products/debug-probes/j- link/models/other-j-links/opensda-sda-v2/
[MBED_TLS]	mbedTLS website - https://tls.mbed.org/
[SDKBUILDER]	MCUXPresso SBKBuilder website - https://mcuxpresso.nxp.com/en/select
[QUICK_START_IMX6]	AN12119 Quick start guide for OM3710A71CHARD i.MX6 – Application note, document number 4582**1
[I2C_BIRD]	http://www.ascot-gmbh.de/download/download.de.html
[AN_A71CH_HOST_SW]	AN12133 A71CH Host software package documentation – Application note, document number 4643**1
[CYGWIN]	Cygwin Setup download - https://cygwin.com/install.html

13. Legal information

13.1 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.

13.2 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 noninfringement, 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.

13.1 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.

13.2 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.

Application note

COMPANY PUBLIC

14. List of figures

Fig 1.	System architecture diagram3
Fig 2.	A71CH mini PCB OM3710/A71CHPCB5
Fig 3.	OM3710/A71CHPCB board schematic6
Fig 4.	OM3710/A71CHPCB board silkscreen6
Fig 5.	A71CHARD Arduino header7
Fig 6.	FRDM-K64F Freedom development platform for Kinetis K64, K63 and K24 MCUs8
Fig 7.	FRDM-K82F Freedom development platform for Kinetis K80, K813 and K82 MCUs8
Fig 8.	A71CH mini PCB OM3710/A71CHPCB9
Fig 9.	A71CH Mini PCB board (OM3710/A71CHPCB) mounting on the I2C adaptor of the Arduino interface board
Fig 10.	A71CH Arduino kit mounted on Kinetis FRDM- K64F board10
Fig 11.	Arduino interface board connected to the Kinetis board (FRDM-K64F)10
Fig 12.	Red USB indicates OpenSDA serial port. Yellow USB indicates virtual COM connector port (FRDM-K64F)
Fig 13.	MCUXpresso install wizard12
Fig 14.	OpenSDA bootloader version for the Kinetis FRDM-K64F13
Fig 15.	Desired firmware for the Kinetis FRDM-K64F. 13
Fig 16.	Enabling bootloader mode14
Fig 17.	Copying the firmware into the Kinetis board in 'BOOTLOADER' mode
Fig 18.	SDKBuilder website15
Fig 19.	Configuring the SDK to download16
Fig 20.	Installing the downloaded Kinetis SDK into MCUXpresso16
Fig 21.	Importing project from SDK17
Fig 22.	Importing project from SDK. Board selection page17
Fig 23.	Importing project from SDK. Select A71CH example projects
Fig 24.	Content of 'frdmk64f_projects' folder19
Fig 25.	Import project from file system19
Fig 26.	Project directory to import20
Fig 27.	Project directory to import21
Fig 28.	Microsoft Visual Studio 201722
Fig 29.	Configuration steps to debug using MCUXPresso IDE Console23
Fig 30.	Running A71CH Configure tool system setup (e.g. FRDM-K64F)24

Fig 31.	Localization of the Visual Studio A71CH projects in the Host Library25
Fig 32.	Visual Studio 2017. Building the A71CHConfig project
Fig 33.	A71CHConfig_vcom executable file26
Fig 34.	mainA71CH executable file27
Fig 35.	Host API usage application execution from a Power Shell terminal28
Fig 36.	Host API usage application execution from Visual Studio29
Fig 37.	Client-Server connection using RJCT server30
Fig 38.	i.MX6UltraLite boot sequence
Fig 39.	Check i.MX6UltraLite IP address
Fig 40.	Launch RJCT server in the i.MX6UltraLite31
Fig 41.	A71CH Configure tool executed in interactive mode
Fig 42.	Connection established with the RJCT server running in the i.MX6UltraLite32
Fig 43.	'info all' command executed from the development PC33
Fig 44.	USB to I ² C dongle and I ² C bus cable
Fig 45.	USB to I ² C Bird – A71CH mini PCB interface connection
Fig 46.	System architecture diagram. Detailed view on Windows platform35
Fig 47.	USB to I ² C Ascot adaptor. Drivers installation wizard
Fig 48.	Java environment variables37
Fig 49.	NXPCardServer execution window
Fig 50.	NXPCardServer folder contents
Fig 51.	A71CH Configure tool execution using the USB to I ² C bird
Fig 52.	Left USB connection is used to flash the program from MCUXpresso IDE. Right USB port connection acts as a virtual COM port40
Fig 53.	Drivers correctly installed40
Fig 54.	Device not detected correctly41
Fig 55.	Connection type selection page42
Fig 56.	Package selection page42
Fig 57.	Package selection page43
Fig 58.	Running A71CH application examples with Cygwin setup43
Fig 59.	Running the downloaded file43
Fig 60.	Running the build script44
Fig 61.	Files created with the buildA71CH.sh script44

Fig 62.	Running the A71CH provisioning example using Cygwin45
Fig 63.	Running the A71CH Configure Tool using Cygwin46
Fig 64.	Running the A71CH Host API usage example using Cygwin47

Application note COMPANY PUBLIC

15. List of tables

Table 2. References......48

16. Contents

1.	Introduction3
2.	A71CH Overview3
3.	System description3
4.	Hardware overview4
4.1	A71CH Arduino compatible development kit
	(OM3710/A71CHARD)4
4.1.1	A71CH Mini PCB board (OM3710/A71CHPCB).4
4.1.2	Arduino interface board7
4.2	Freedom development platforms for Kinetis7
4.2.1	FRDM-K64F
4.2.2	FRDM-K82F8
5.	Hardware setup8
6.	Software setup11
6.1	MCUXpresso IDE installation11
6.2	OpenSDA configuration12
6.3	Kinetis SDK package for A71CH15
6.4	Importing a project in MCUXpresso IDE
6.4.1	Importing the A71CH example projects from the
6.4.2	installed SDK16 Importing A71CH example projects from local
0.4.2	drive (bundled with installer)
6.5	Microsoft Visual Studio IDE installation
7.	A71CH application examples execution
7.1	Set the Kinetis board as virtual COM port23
7.1.1	Connect the Kinetis board to the Windows
	platform over USB
7.2	Running A71CH Configure tool
7.3	Running A71CH Host API usage example27
7.3.1	Run the A71CH Host API usage executable from
	a terminal27
7.3.2	Run the A71CH Host API usage from Visual
	Studio28
8.	RJCT server29

9.	Appendix A: Running A71CH application examples with USB to I ² C bird	22
9.1	-	
9.1	USB/I ² C bird (OM3710/B001)	
9.2 9.3	Hardware setup	
9.3 9.3.1	Software setup USB to I ² C ASCOT adaptor – Drivers	
9.3.1	NXPCardServer	
9.3.2 9.4	Running A71CH Configure tool	
9.4 9.5	Running A71CH Host API usage example	
10.	Appendix B: VCOM driver installation	20
	troubleshooting	
11.	Appendix C: Running examples using Cygw	
11.1	Cygwin Installation	
11.2	A71CH application examples execution	
11.2.1	Running A71CH OpenSSL Engine examples.	
11.2.2	Running the A71CH Configure Tool	
11.2.3	Running A71CH Host API usage examples	46
12.	References	48
13.	Legal information	49
13.1	Definitions	
13.2	Disclaimers	49
13.1	Licenses	49
13.2	Trademarks	49
14.	List of figures	50
15.	List of tables	52
16.	Contents	53

Please be aware that important notices concerning this document and the product(s) described herein, have been included in the section 'Legal information'.

© NXP B.V. 2018.

All rights reserved.

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

Date of release: 09 July 2018