AN12396

EdgeLock™ SE050 Quick start guide with FRDM-K64F

Rev. 2.1 — 17 December 2019 Application note

534421

Document information

Information	Content
Keywords	EdgeLock SE050, EdgeLock SE050 Plug & Trust middleware, FRDM-K64F
Abstract	This document explains how to get started with EdgeLock SE050 Plug & Trust middleware using the OM-SE050ARD and FRDM-K64F MCU boards. It provides detailed instructions to run projects imported either from the FRDM-K64F SDK or the CMake-based build system included in the EdgeLock SE050 Plug & Trust middleware.

Revision history

Revision history

Revision number	Date	Description
1.0	2019-06-08	First document release
1.1	2019-06-20	Update of board figures
2.0	2019-11-25	Major update to incorporate details to import projects from FRDM-K64F SDK and CMake-based build system.
2.1	2019-12-17	Corrected OM-SE050ARD J14 jumper setting.

1 How to use this document

The EdgeLock SE050 Plug & Trust middleware includes a set of project examples that demonstrate the use of EdgeLock SE050 in the latest IoT security use cases. These project examples can be either:

- Imported from the MCUXpresso SDKs made available for FRDM-K64F MCU board.
- Imported from the CMake-based build system included in the EdgeLock SE050 Plug & Trust middleware package

This document provides detailed instructions to run EdgeLock SE050 project examples imported either from the FRDM-K64F SDK or the CMake-based build system. However, the FRDM-K64F SDK is recommended as it fastest way to import and run EdgeLock SE050 project examples. The CMake-based option is provided for developers familiar with it or willing to run exactly the same project example on PC/Windows/Linux and embedded targets. The main body of this document should be used in this sequence:

1. Order board samples. Section 2 contains the ordering details of the boards required in this document
2. Setup your boards. Section 3 describes how to setup the OM-SE050ARD and FRDM-K64F boards.
3. Run project examples. Go to Section 4 for instructions to import projects from the FRDM-K64F MCUXpresso SDK or alternatively, go to Section 5 for instructions to import projects from the CMake-based build system.

Supplementary material has been provided in the appendices.

2 Hardware required

This guide provides detailed instructions to run the EdgeLock SE050 Plug & Trust middleware project examples using the hardware described below. However, you could use other MCU boards supported by EdgeLock SE050 Plug & Trust middleware for this purpose as well.

1. OM-SE050ARD development kit:

Table 1. OM-SE050ARD development kit details

Part number	12NC	Content	Picture
OM-SE050ARD	935383282598	EdgeLock SE050 development board	

2. FRDM-K64F board:

Table 2. FRDM-K64F details

Part number	12NC	Content	Picture
FRDM-64F	935326293598	Freedom development platform for Kinetis K64, K63 and K24 MCUs	

3 Boards setup

This section explains how to prepare the OM-SE050ARD and FRDM-K64F boards to run the EdgeLock SE050 Plug & Trust middleware project examples. This consists of:

1. Update FRDM-K64F with DAPLink firmware
2. Hardware setup for FRDM-K64F
3. OM-SE050ARD and FRDM-K64F board connection.

3.1 Update FRDM-K64F board with DAPLink firmware

Arm Mbed DAPLink is an open-source software project that enables programming and debugging application software running on Arm Cortex CPUs. DAPLink runs an open-source bootloader and enables developers with drag-and-drop programming, a serial port and CMSIS-DAP based debugging.

Note: *To debug MCUXpresso project examples, we need to flash FRDM-K64F with DAPLink firmware. If your FRDM-K64F board already includes DAPLink firmware, you can skip these steps.*

To flash DAPLink firmware, follow these steps:

1. Go to NXP OpenSDA site
2. Scroll down and select FRDM-K64F board from the ***Download - OpenSDA bootlader and application*** drop down list as indicated in Figure 1:

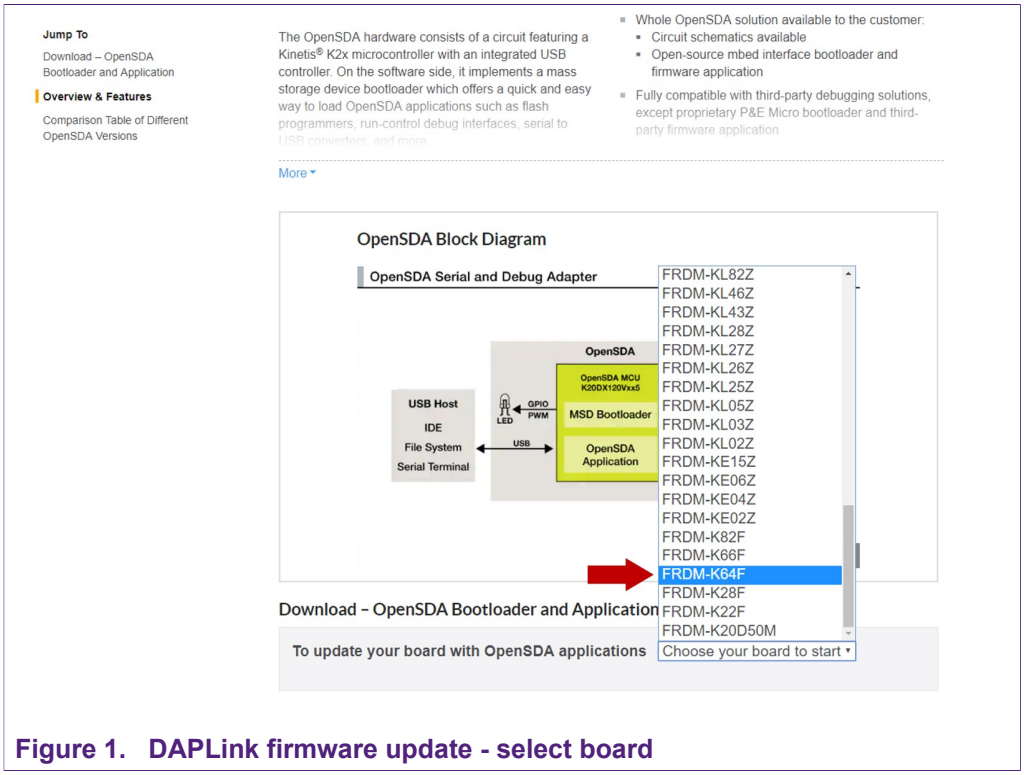

Figure 1. DAPLink firmware update - select board

3. Download the latest DAPLink firmware version as shown in Figure 2:

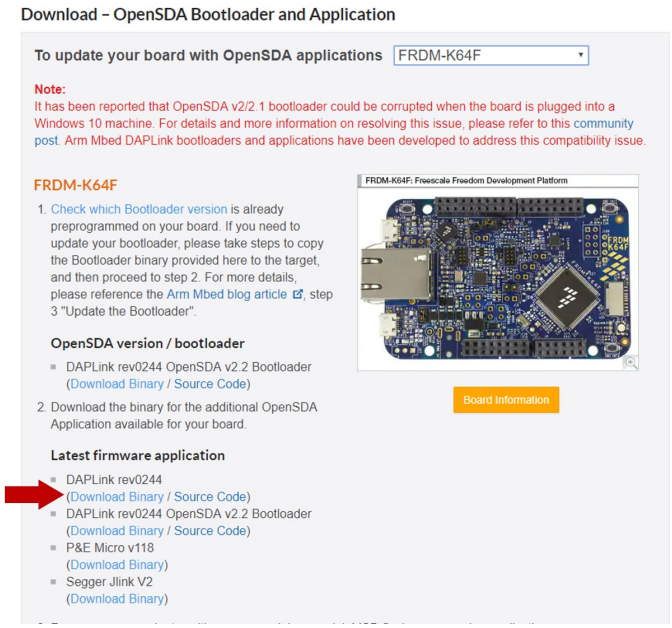

Figure 2. DAPLink firmware update - select board

4. Start the board's bootloader mode. To do so, (1) keep reset button pressed while (2) connecting the USB cable to the SDA USB port and release it after 1s (Figure 3):

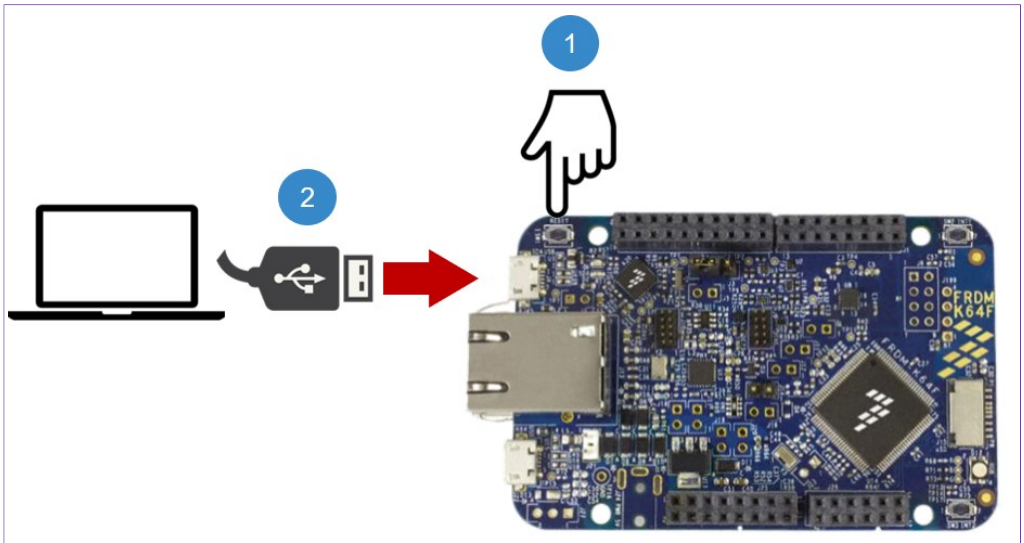

Figure 3. Enter bootloader mode

5. Drag and drop or copy and paste the binary file into the BOOTLOADER drive from your computer file explorer as shown in Figure 4. The FRDM-K64F will automatically un-mount after the drag and drop operation.

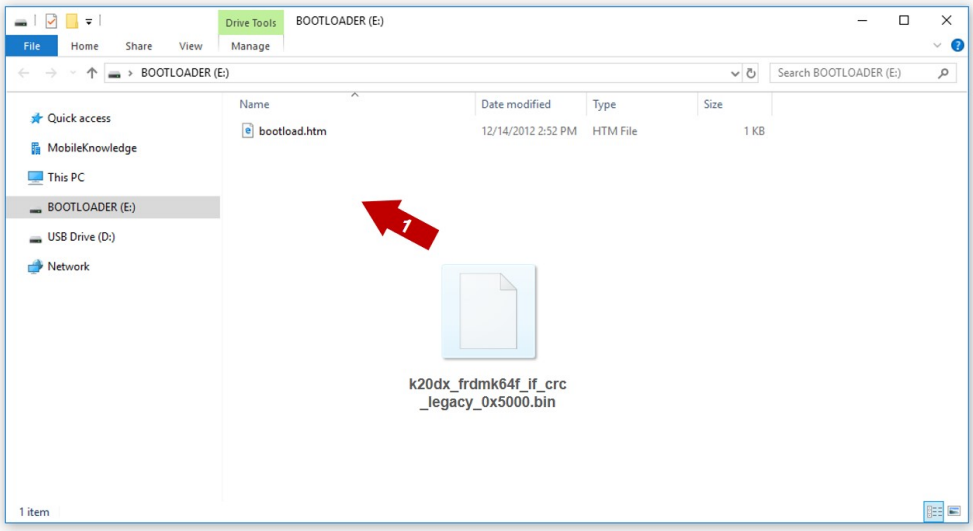

Figure 4. Enter bootloader mode

6. Un-plug and re-plug the USB cable from the SDA USB port ***without*** keeping reset button pressed.
7. Check the category Ports (COM & LTP) from your computer Device Manager to ensure that new devices have been properly detected and their driver correctly installed by your computer OS.

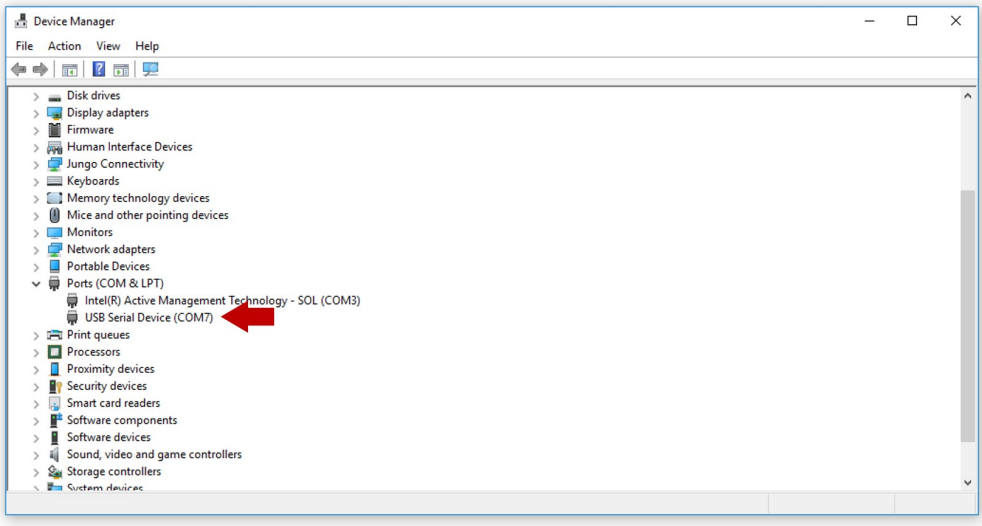

Figure 5. Enter bootloader mode

3.2 OM-SE050ARD jumper configuration

The OM-SE050ARD board has jumpers that allow you to interface the EdgeLock SE050 I^2C interface via the Arduino header. Configure the jumper settings as shown in Figure 6 to enable this option.

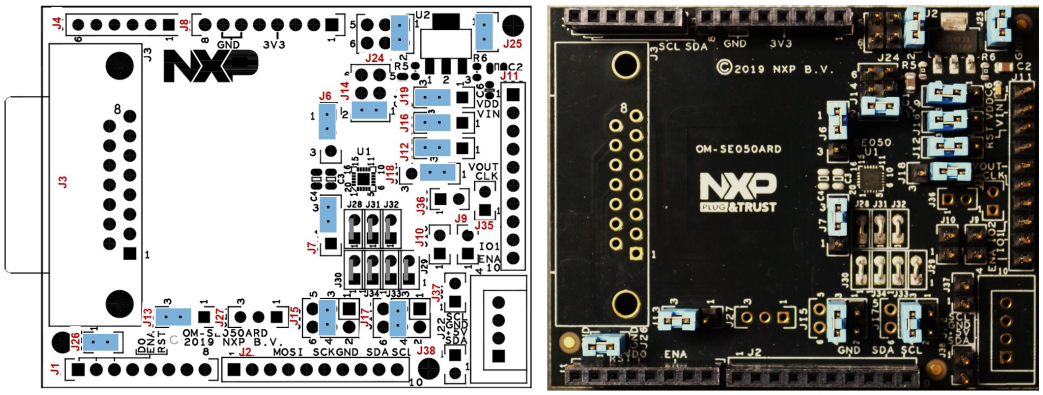

Figure 6. Jumper configuration for FRDM-K64F

For more information about the OM-SE050ARD jumper settings, refer to AN12395 OM-SE050ARD hardware overview.

3.3 OM-SE050ARD and FRDM-K64F board connection

The OM-SE050ARD and FRDM-K64F boards can be directly connected using the Arduino connectors of both boards. The OM-SE050ARD comes with male connectors while the FRDM-K64F board comes with female headers.

Mount the OM-SE050ARD board on top of the FRDM-K64F as shown in Figure 7:

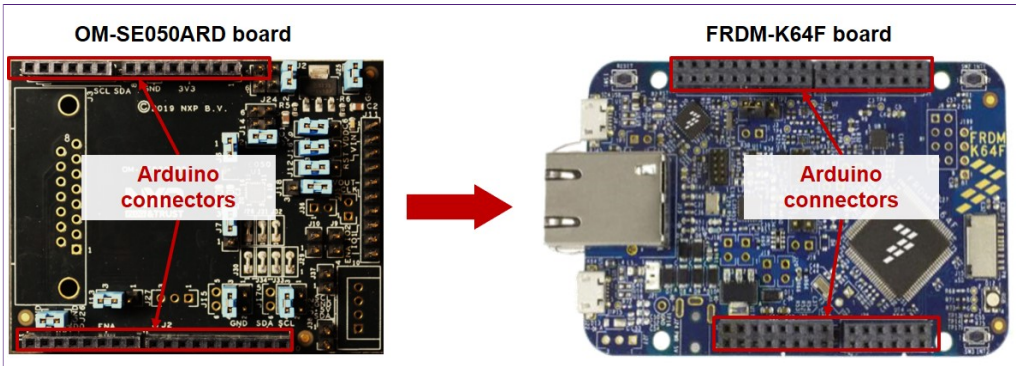

Figure 7. Arduino connectors of OM-SE050ARD and FRDM-K64F boards

Double check that the two boards are connected as shown in Figure 8:

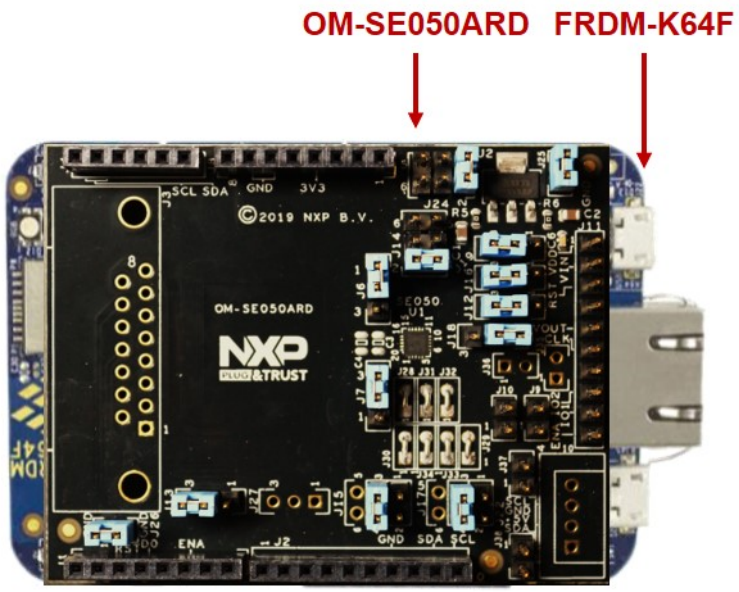

Figure 8. OM-SE050ARD mounted in FRDM-K64F board

Note: *Refer to Figure 6 for OM-SE050ARD jumper configuration.*

4 Import project examples from FRDM-K64F SDK

This section explains how to run the EdgeLock SE050 projects importing them from the FRDM-K64F SDK.

4.1 Prerequisites

The following steps are required to run a project imported from the MCUXpresso SDK:

1. MCUXpresso IDE. Check Section 6 for detailed installation instructions
2. TeraTerm (or an equivalent serial application). You can download and run TeraTerm installer from this link.

4.2 Download FRDM-K64F SDK

The project examples for the EdgeLock SE050 are included as part of the FRDM-K64F SDK. First, download the FRDM-K64F SDK, publicly available from the NXP website.

4.3 Install FRDM-K64F SDK

After downloading the FRDM-K64F SDK, we need to install it into our MCUXpresso workspace. To install the SDK, (1) drag and drop the FRDM-K64F SDK zip file in the ***Installed SDKs*** section in the bottom part of the MCUXpresso IDE and (2) click ***OK*** as shown in Figure 9:

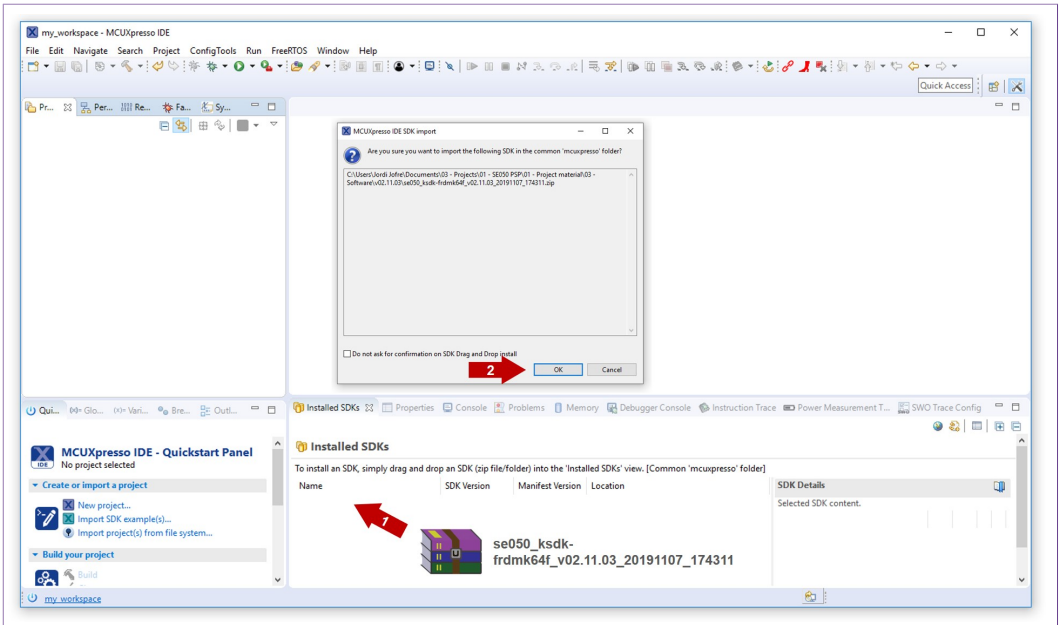

Figure 9. Import FRDM-K64F board SDK into MCUXpresso environment

If the SDK is successfully imported, you should see it listed in the ***Installed SDK*** window as shown in Figure 10:

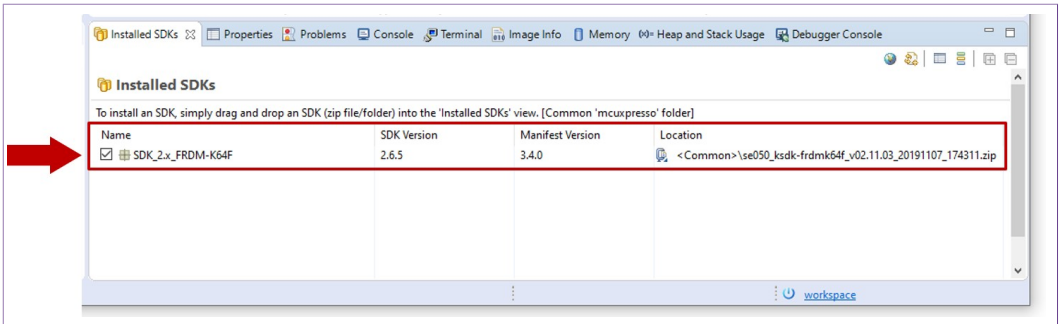

Figure 10. Imported FRDM-K64F SDK

4.4 Import project example in MCUXpresso

After importing the FRDM-K64F SDK in our MCUXpresso workspace, follow these instructions to import a project:

1. Click *Import SDK example*s from file system in the MCUXpresso IDE quick start panel as shown in Figure 11

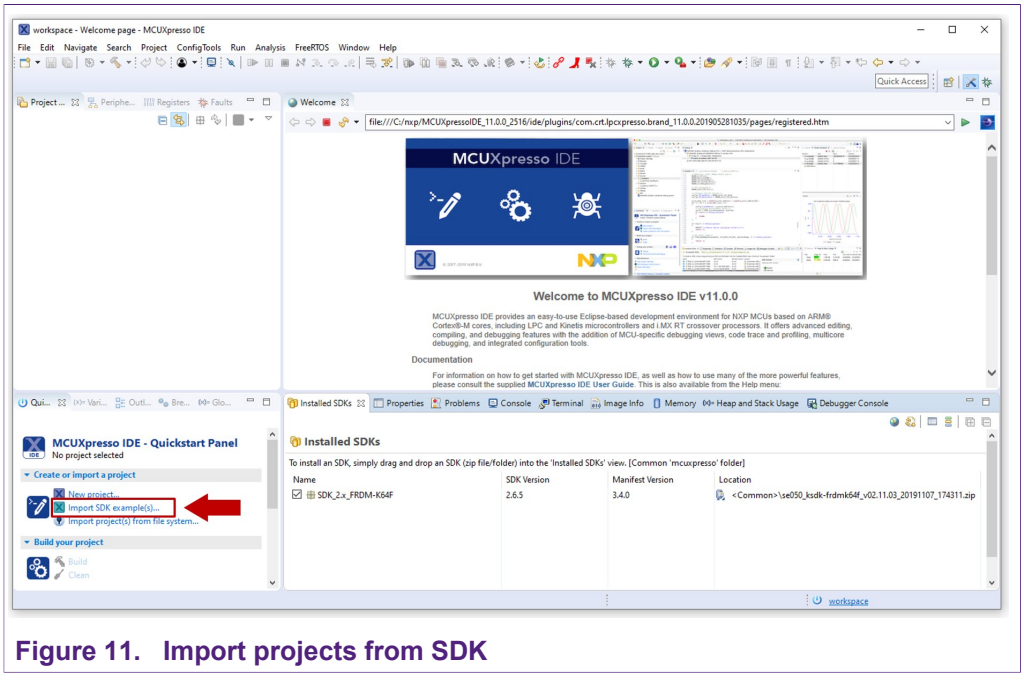

Figure 11. Import projects from SDK

2. The SDK import wizard will be opened. You should see a figure of an FRDM-K64F board with an SE050 orange label. Select the board and click *Next* button as shown in Figure 12:

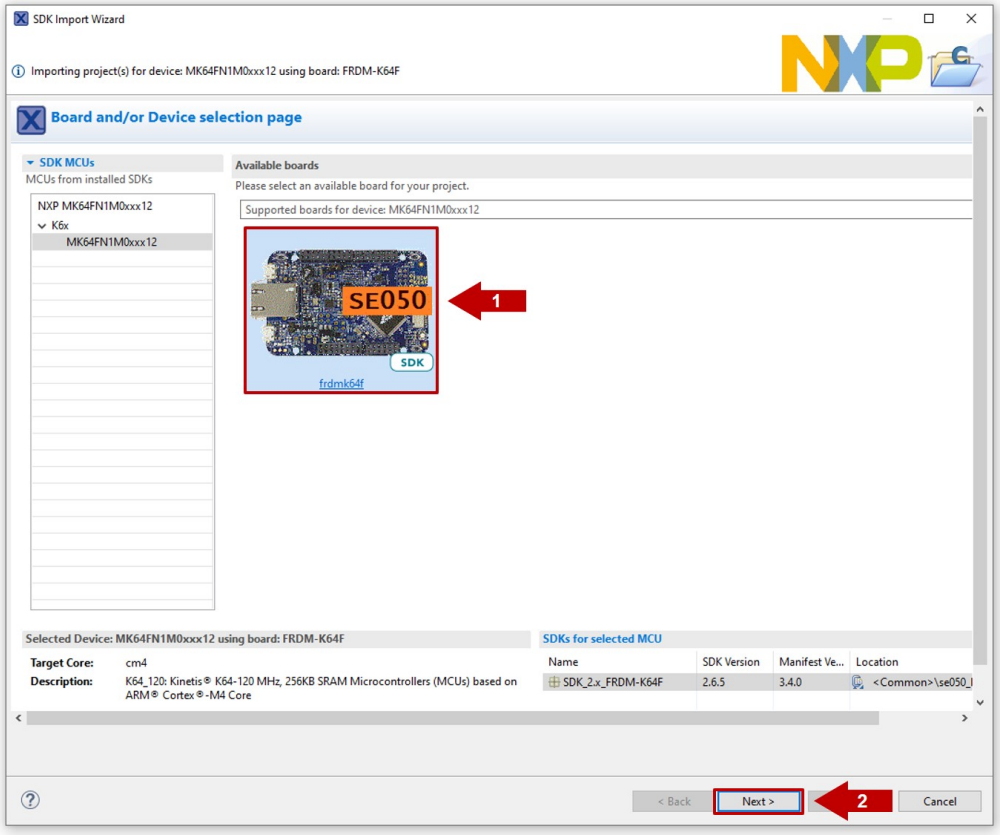

Figure 12. SDK import wizard

3. Under the `se_hostlib_examples` drop down list, you have the list of supported project examples for the FRDM-K64F. Select the number of examples you would like to import in your MCUXpresso workspace and click *Finish* button as shown in Figure 13. In this case, we select the `se_hostlib_se05x_minimal` project as an example. The same process can be done with the rest of them.

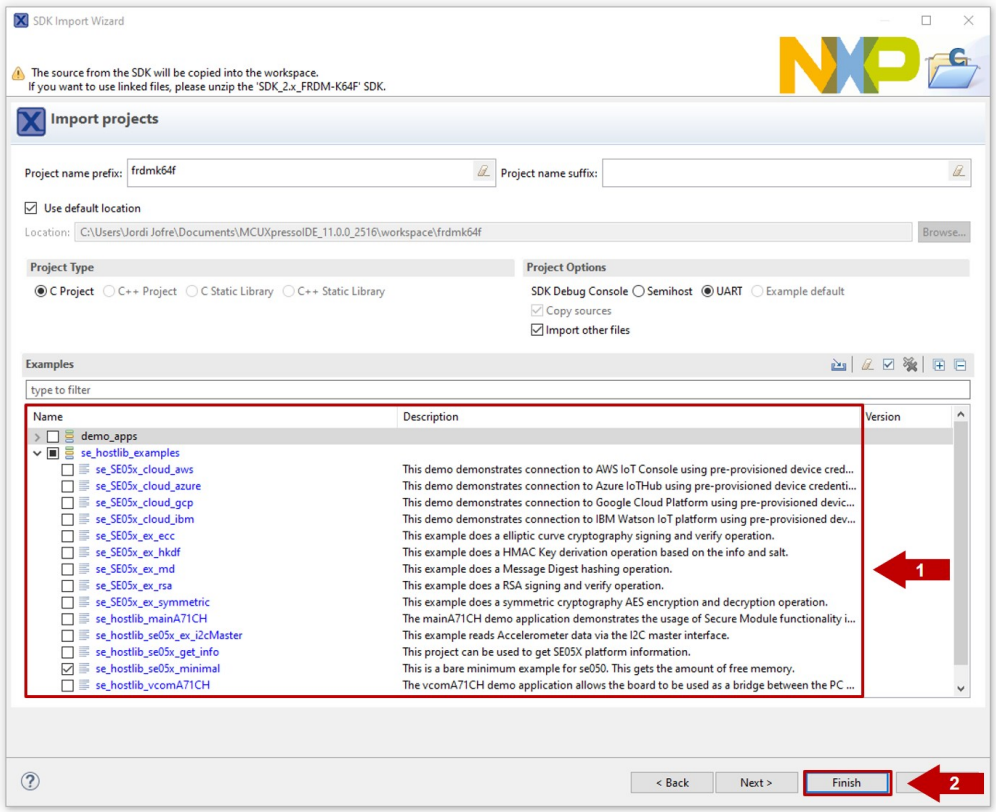

Figure 13. Select projects to import

4. The projects you selected should now be visible in your MCUXpresso workspace as shown Figure 14:

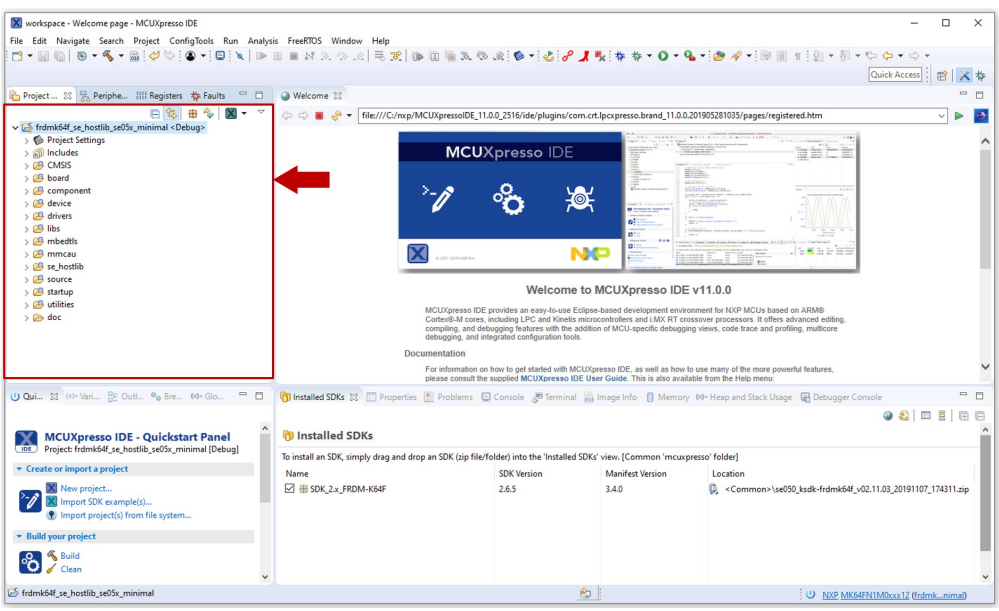

Figure 14. Imported projects in MCUXpresso workspace

4.5 Build, run and debug project example

After importing project examples in our MCUXpresso workspace, follow these instructions to build, run and debug a project:

1. Attach a USB cable from the computer to the K64F OpenSDA debug USB connector as shown in Figure 15.

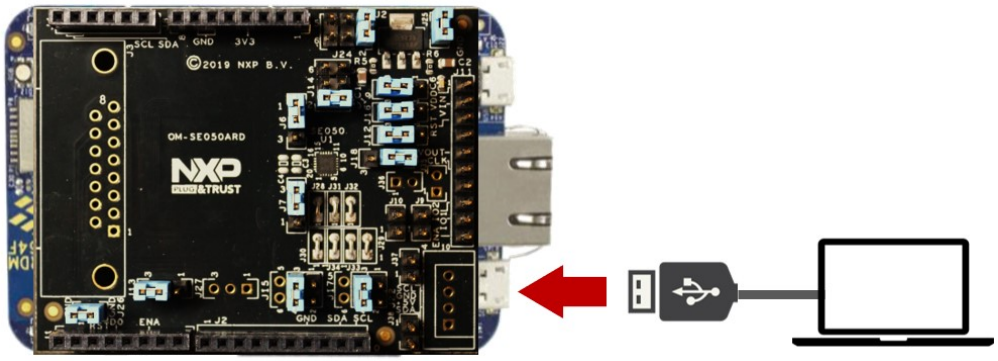

Figure 15. Connect boards to the laptop

2. Launch and setup TeraTerm application as shown in Figure 16:
   a. Click *Serial* option and select from the drop down list the COM port number assigned to your FRDM-K64F board
   b. Go to Setup > Serial Port and configure the terminal to 115200 baud rate, 8 data bits, no parity and 1 stop bit and click OK.

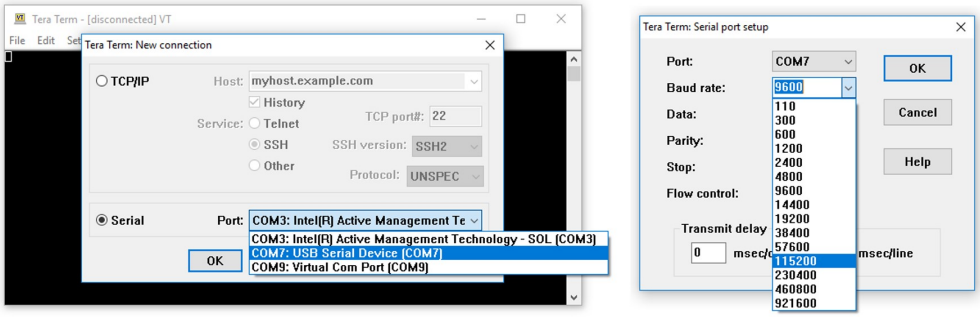

Figure 16. TeraTerm setup

3. Go to the MCUXpresso Quickstart Panel and click *Build* button as shown in Figure 17. Wait a few seconds and check that the build process has finished successfully in the MCUXpresso console window.

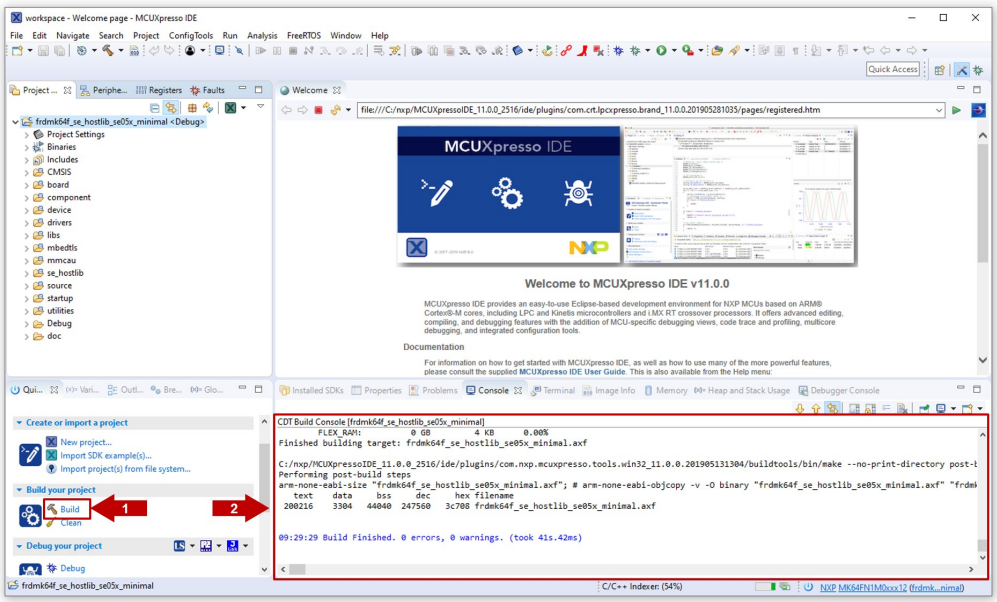

Figure 17. Build projects in MCUXpresso workspace

4. Go to the MCUXpresso Quickstart Panel and click *Debug* button as shown in Figure 18. If there is more than one probe attached, you have to the select CMSIS-DAP debug probe from the list. Wait a few seconds until the project executes

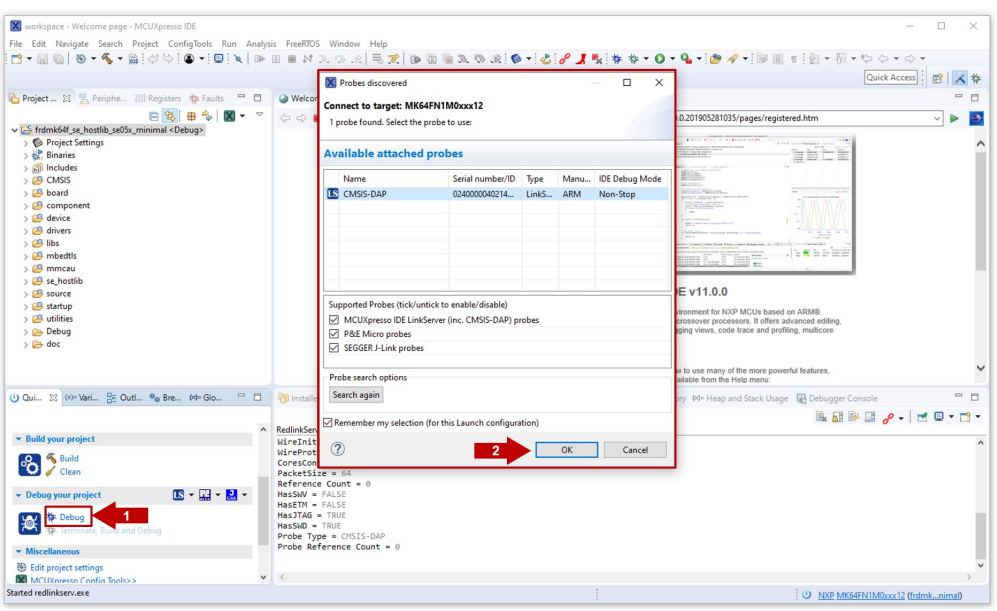

Figure 18. Debug projects in MCUXpresso workspace

5. When it executes, it will automatically stop in a breakpoint. Click on Resume to allow the software to continue its execution as shown in Figure 19.

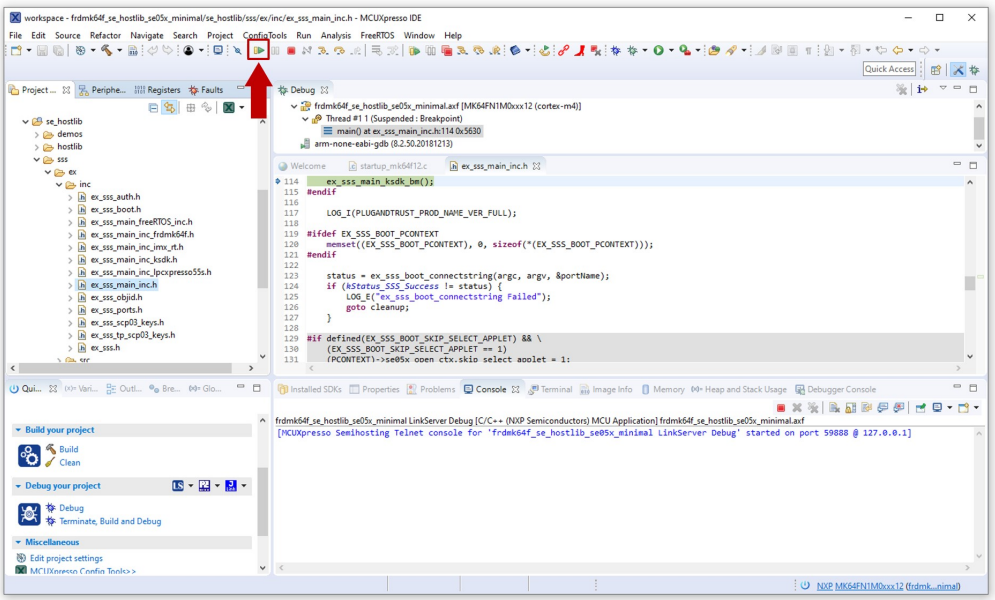

Figure 19. Run projects in MCUXpresso workspace

6. Once the program execution begins, logs are printed on the terminal application indicating the execution status. For the `se05x_minimal` project example, the logs should indicate the available memory in EdgeLock SE050 (in this case, 592) as can be seen in Figure 20

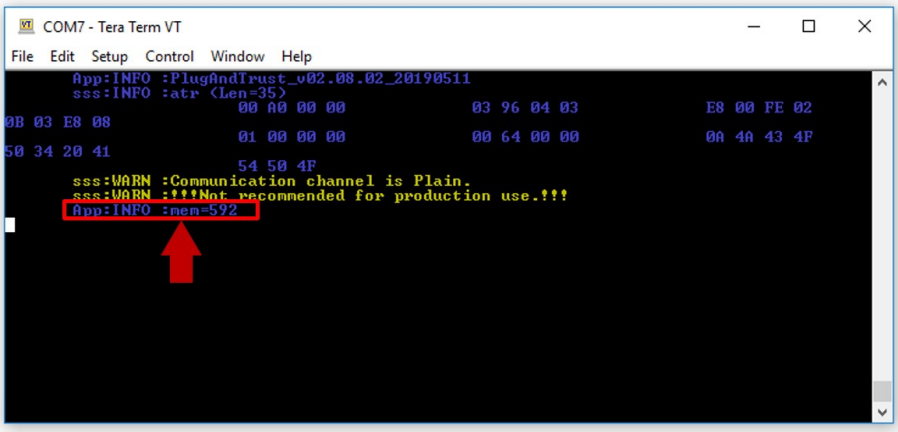

Figure 20. TeraTerm logs - `se05x_minimal` project example

7. The same operation can be repeated with any of the other EdgeLock SE050 Plug & Trust middleware project examples.

5 Import project examples from CMake-based build system

This section explains how to run EdgeLock SE050 projects using the CMake-based build system.

5.1 Prerequisites

The following tools are required to run project imported from the MCUXpresso SDK:

1. MCUXpresso IDE. Check Section 6 for detailed installation instructions.
2. CMake. Check Section 7 for detailed installation instructions.
3. Python 3.7.x 32 bit version. Check Section 8 for detailed installation instructions.
4. TeraTerm (or an equivalent serial application). You can download and run TeraTerm installer from this link.

5.2 Download EdgeLock SE050 Plug & Trust middleware

Follow these steps to download the EdgeLock SE050 Plug & Trust middleware in your local machine:

1. Download EdgeLock SE050 Plug & Trust middleware from the NXP website.
2. Create a folder called ***se050_middleware*** in C: directory as shown in Figure 21:

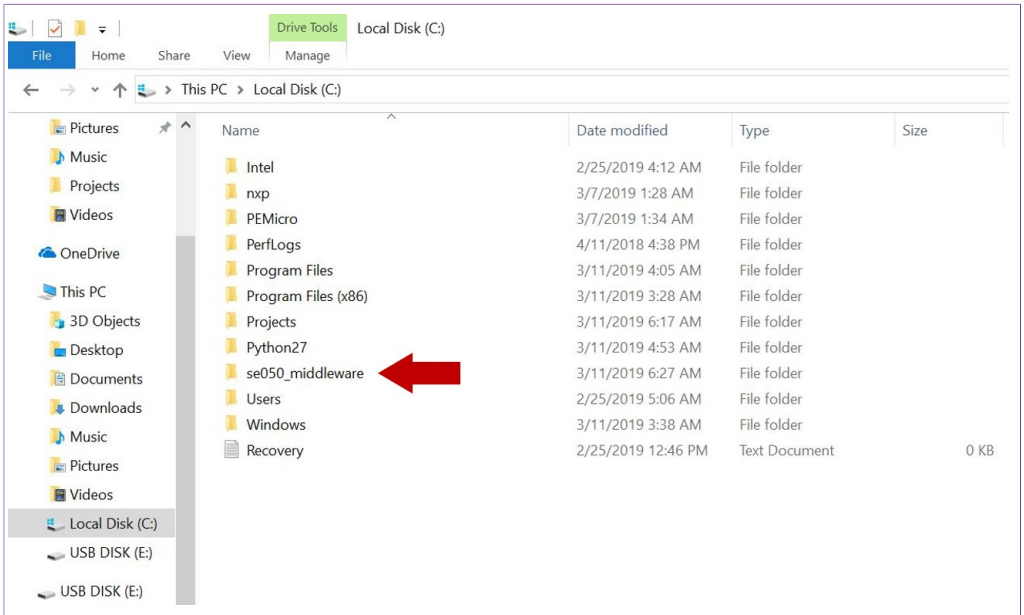

Figure 21. Create se050_middleware folder

3. Unzip the EdgeLock SE050 Plug & Trust middleware inside the *se050_middleware* folder. After unzipping, you will see a folder called ***simw-top*** created. The contents of the ***simw-top*** directory should look as shown in Figure 22:

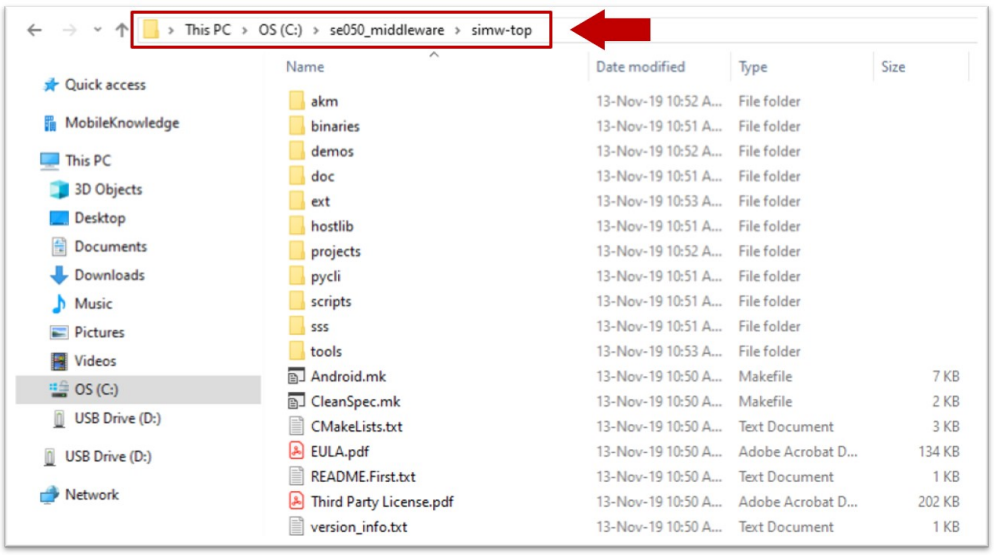

Figure 22. Unzip se050 middleware

Note: *It is recommended to keep* `se050_middleware` *with the* ***shortest*** *path possible and* ***without spaces*** *in it. This avoids some issues that could appear when building the middleware if the path contains spaces.*

5.3 Build EdgeLock SE050 Plug & Trust middleware project examples

The EdgeLock SE050 Plug & Trust middleware uses CMake for building the project examples into your local machine. To build EdgeLock SE050 Plug & Trust middleware, open a Command Prompt and use the following steps as shown in Figure 23:

1. Go to folder with the unzipped SE050 middleware:
   (1) Send >> `cd C:\se050_middleware\simw-top\scripts`
2. Define the environment:
   (2) Send >> `env_setup.bat`
3. Generate the EdgeLock SE050 Plug & Trust middleware project examples:
   (3) Send >> `create_cmake_projects.py`
   Note*: This command may take a few seconds to complete.*

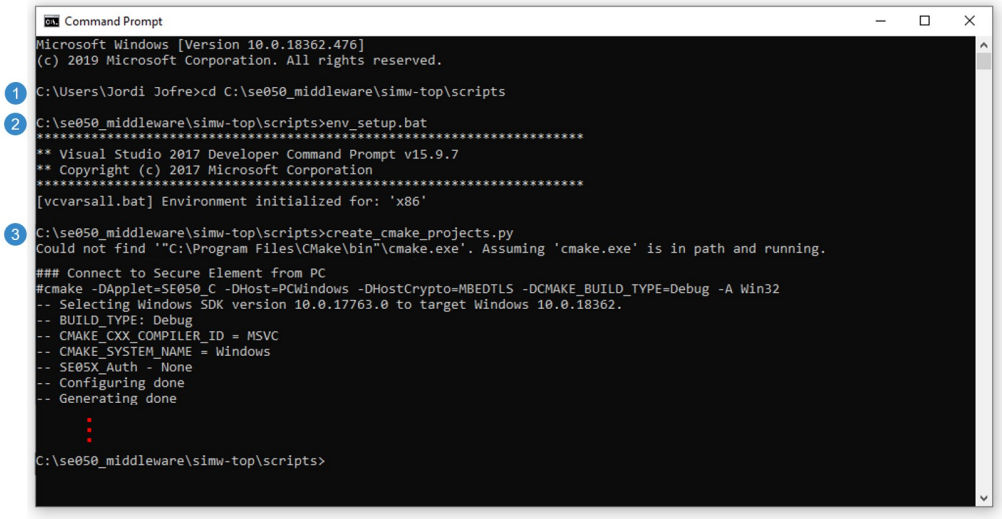

Figure 23. Generate EdgeLock SE050 Plug & Trust middleware project examples

4. Your project directory should now contain two folders: a (1) `simw-top` folder and a (2) `simw-top_build` folder as shown in Figure 24:

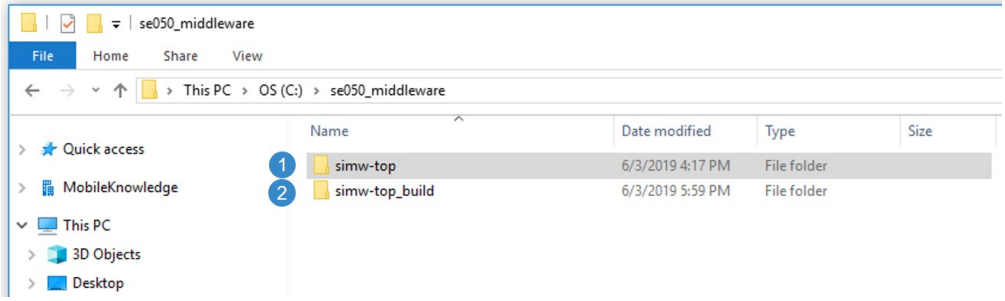

Figure 24. SE050 middleware project structure

5.4 Import *PlugAndTrustMW* project example in MCUXpresso workspace

After generating the projects in your local machine using the `create_cmake_projects.py` script, we need to import the *PlugAndTrustMW* project example in our MCUXpresso workspace. Follow these steps to import a project:

1. Go to *File* → *Import* using the top bar menu as shown in Figure 25. **Note**: In this case, do not use the MCUXpresso Quickstart Panel to import project.

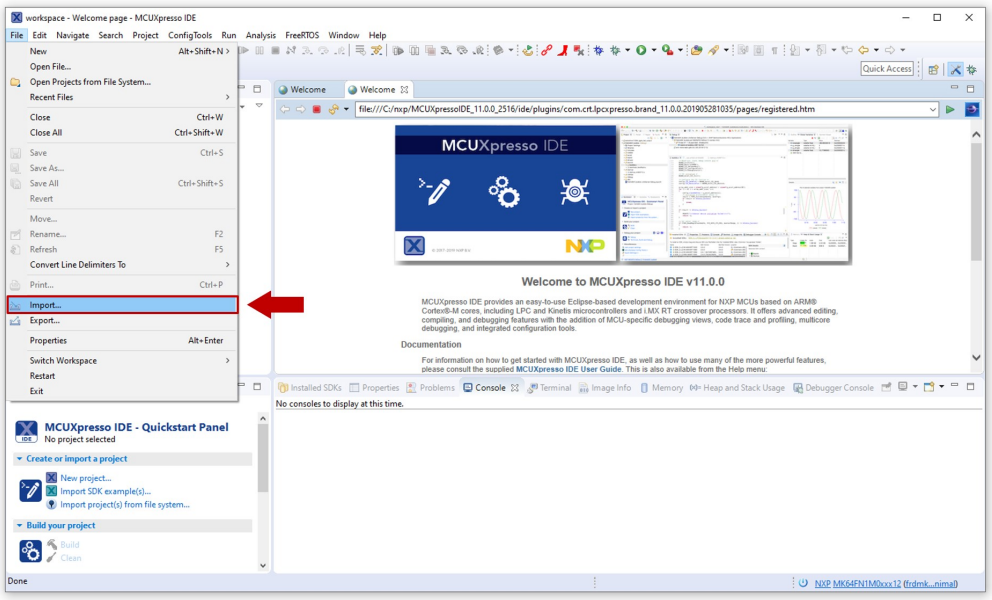

Figure 25. Import a project wizard

2. In the import wizard menu, select import ***"Existing Projects into Workspace"*** from the *General* folder as shown in Figure 26:

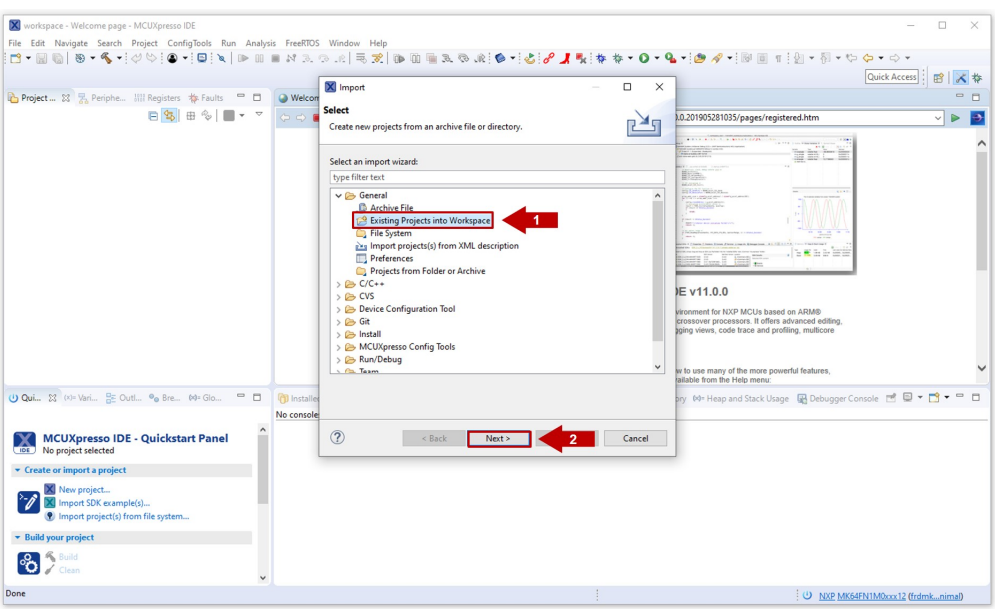

Figure 26. Import a project wizard (II)

3. First, we need to import EdgeLock SE050 Plug & Trust middleware project in MCUXpresso. For that, in the *Select root directory* option, browse to *C:\se050_middleware\simw-top_build* or browse the location of your EdgeLock SE050 Plug & Trust middleware directory and click *Select folder* as shown in Figure 27:

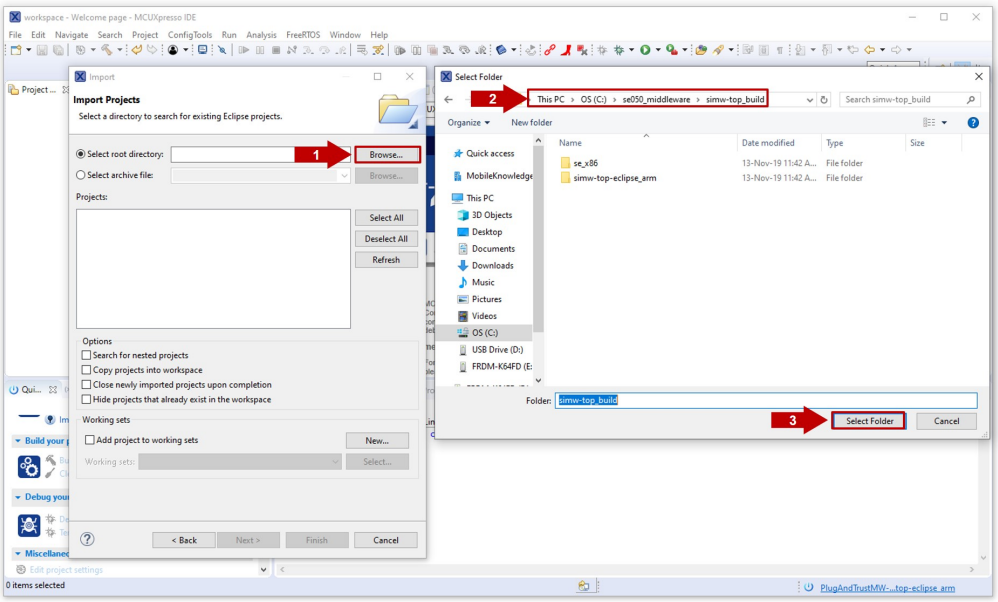

Figure 27. Select EdgeLock SE050 Plug & Trust middleware build folder

4. After selecting *C:\se050_middleware\simw-top_build* folder, a project called *PlugAndTrustMW* should be visible in the "projects" area. Click *Finish* button to import this project into your worskpace as shown in Figure 28:

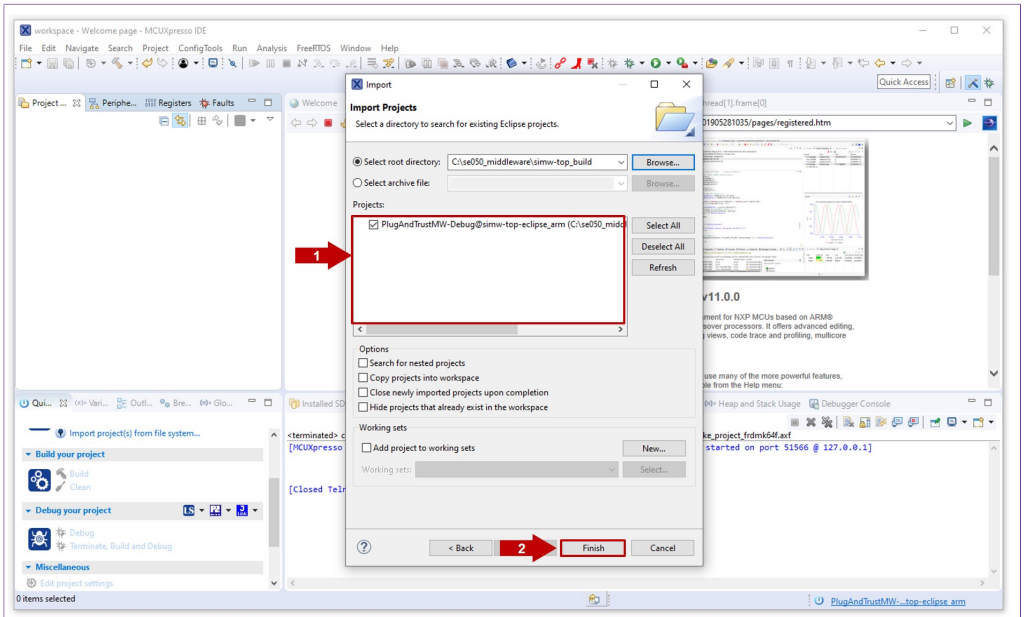

Figure 28. Import EdgeLock SE050 Plug & Trust middleware

5. The *PlugAndTrustMW* project should now be imported in your workspace as shown in Figure 29:

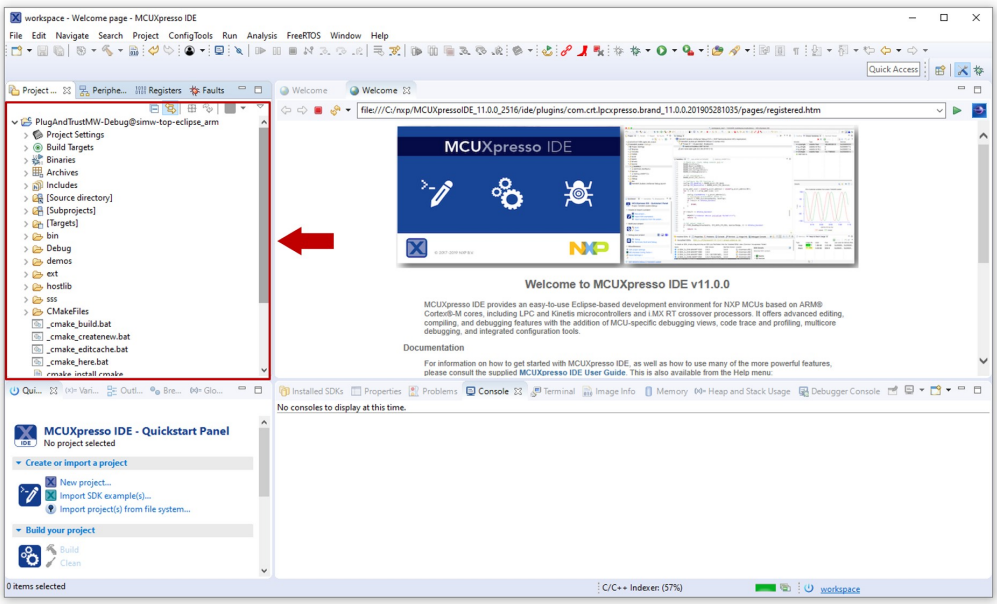

Figure 29. EdgeLock SE050 Plug & Trust middleware imported in workspace

5.5 Import *cmake_projects_frdm64f* project example in MCUXpresso workspace

After importing the *PlugAndTrustMW* project example in MCUXpresso, we need to import the *cmake_projects_frdm64f* project example. Follow these steps:

1. Go to *File* → *Import* using the top bar menu as shown in Figure 25. **Note**: In this case, do not use the MCUXpresso Quickstart Panel to import project.

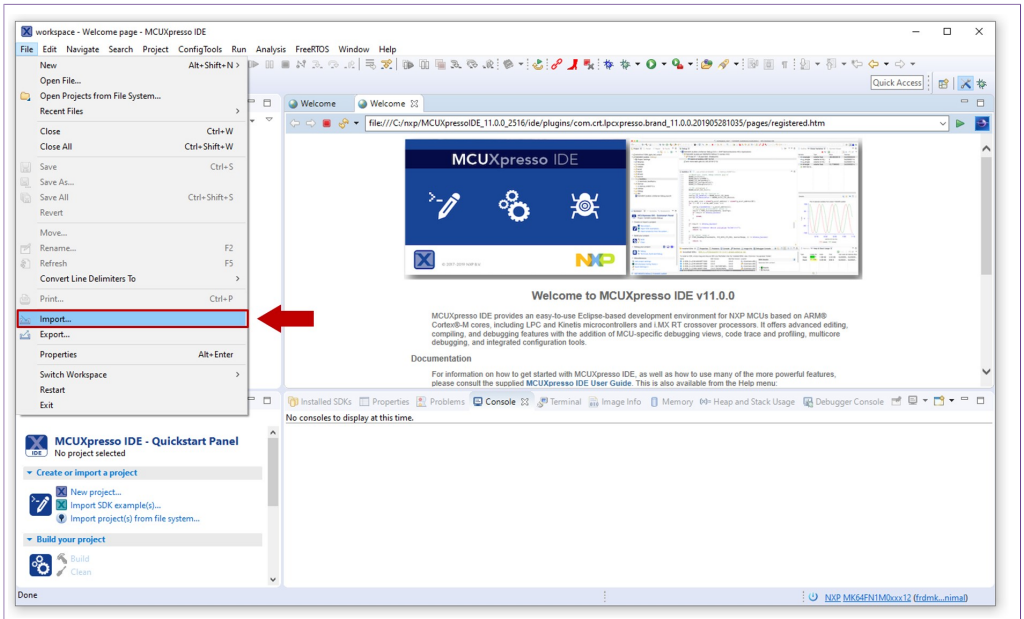

Figure 30. Import a project wizard

2. In the import wizard menu, select import ***"Existing Projects into Workspace"*** from the *General* folder as shown in Figure 26:

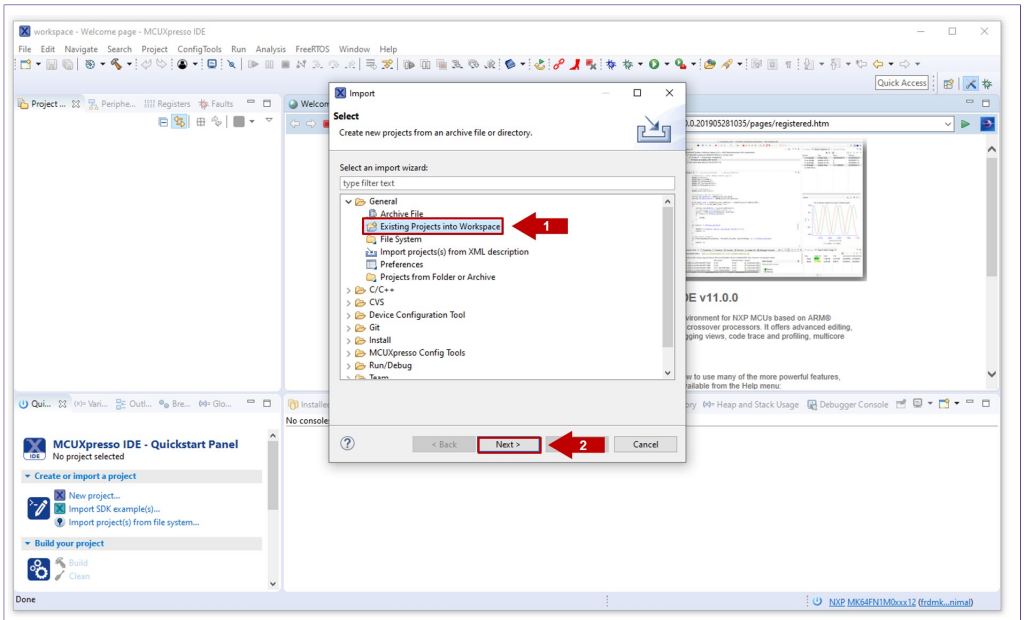

Figure 31. Import a project wizard (II)

3. In the *Select root directory* option, browse to *C:\se050_middleware\simw-top \projects* or browse the location of your FRDM-K64F projects directory. Choose the *cmake_projects_frdm64f* project and click *Select folder* as shown in Figure 32:

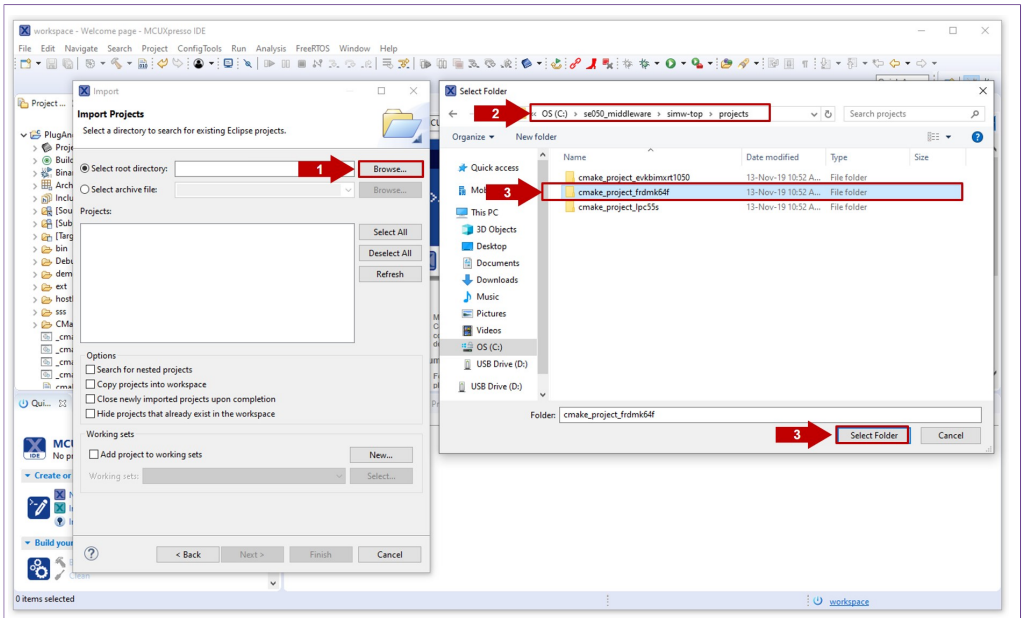

Figure 32. Select FRDM-K64F projects folder

4. After selecting *C:\se050_middleware\simw-top\projects* folder, the *cmake_projects_frdm64f* project should be visible in the "projects" area. Click *Finish* button to import this project into your worskpace as shown in Figure 33:

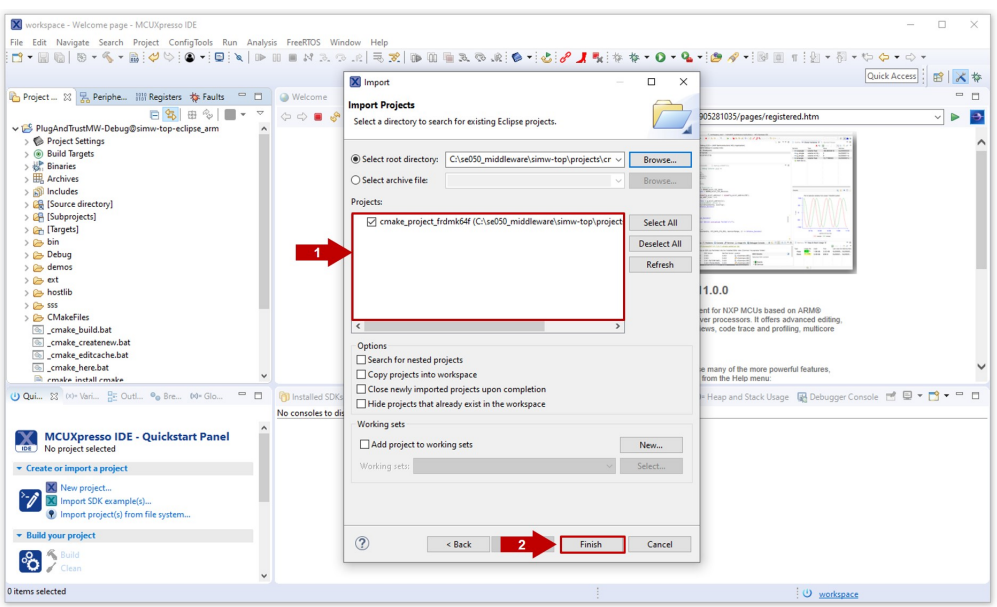

Figure 33. Import FRDM-K64F in workspace

5. Both The *PlugAndTrustMW* and *cmake_projects_frdm64f* projects should now be imported in your workspace as shown in Figure 34:

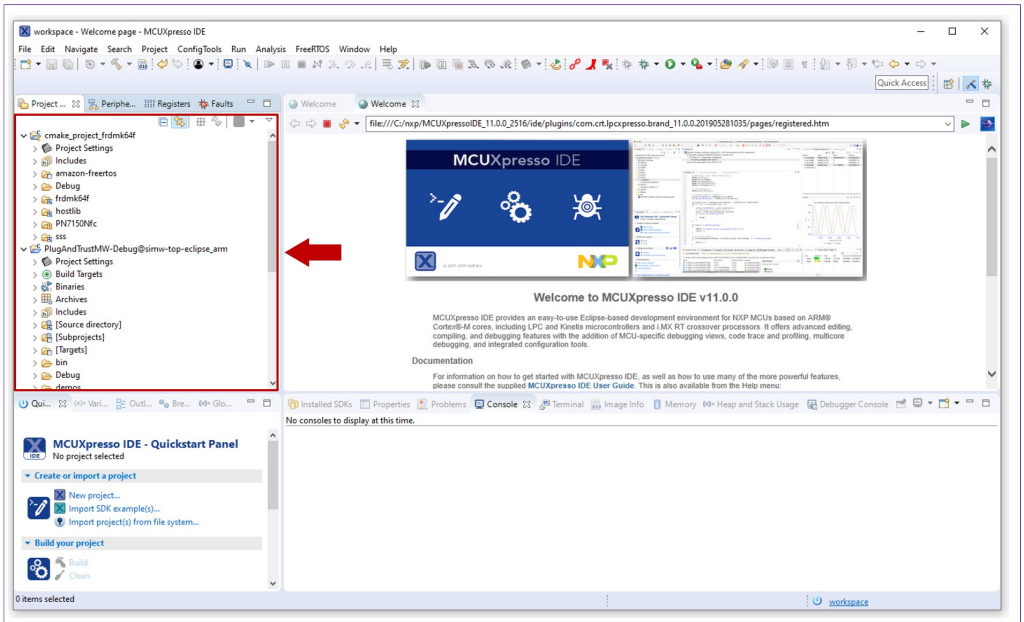

Figure 34. FRDM-K64F imported in workspace

5.6 Run EdgeLock SE050 Plug & Trust middleware test examples

This section explains how to list, edit and execute project examples using the CMake build system. It includes the following sections:

- List the EdgeLock SE050 Plug & Trust middleware test examples.
- Edit EdgeLock SE050 Plug & Trust middleware test example CMake options.
- Execute one EdgeLock SE050 Plug & Trust middleware test example.

5.6.1 List the EdgeLock SE050 Plug & Trust middleware test examples

The EdgeLock SE050 Plug & Trust middleware comes with several test examples used to verify atomic EdgeLock SE050 security IC features. To get the list of test examples, follow these steps:

1. Select the *cmake_project_frdmk64f* project example and click on the arrow on the "hammer" icon in the top bar menu of the MCUXpresso.
2. Select ***3 help (Print help)*** option. Wait a few seconds until the operation is completed.
3. The MCUXpresso console display the list of EdgeLock SE050 Plug & Trust middleware test examples as shown in Figure 35.

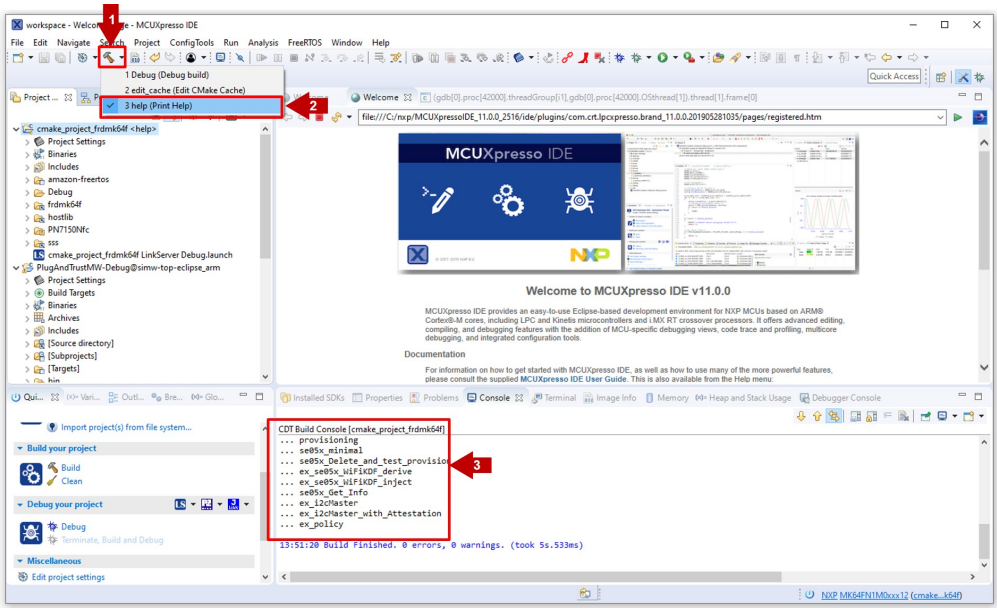

Figure 35. EdgeLock SE050 Plug & Trust middleware test examples

5.6.2 Edit EdgeLock SE050 Plug & Trust middleware test example CMake options.

The EdgeLock SE050 Plug & Trust middleware is delivered with the CMake files that include the set of directives and instructions describing the project's source files and targets. In addition, it includes the CMake configuration files used to enable or disable several features, portability and setting flags to generate the build files for your platform and native build environment. To edit the CMake options, follow these steps:

1. Click on the arrow on the "hammer" icon in the top bar menu of the MCUXpresso.

2. Select ***2 edit_cache (Edit CMake Cache)***.
3. The CMake GUI window will open in your laptop as shown in Figure 36. Using this GUI, you could change the CMake options (if needed). Leave out the **default** pre-selected options and close the CMake GUI window.
   Note: *In case you want to change any of the default pre-selected CMake options, you need to click on Configure or Generate buttons before closing the CMake window.*

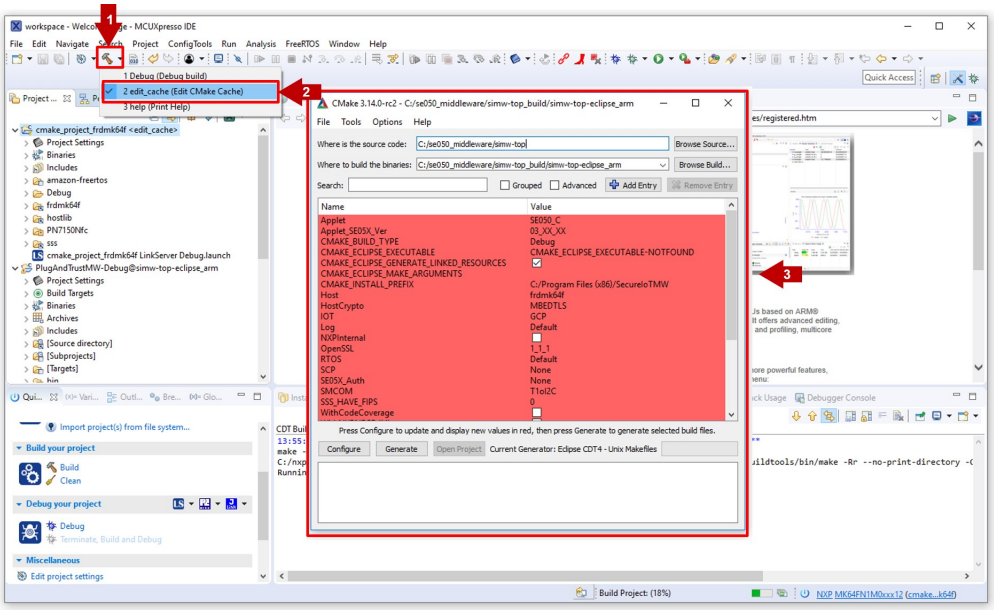

Figure 36. Configure CMake options of EdgeLock SE050 Plug & Trust middleware test examples.

5.6.3 Build and run an EdgeLock SE050 Plug & Trust middleware project example

This section explains how to build and run the EdgeLock SE050 Plug & Trust middleware test example called `se05x_minimal`. The `se05x_minimal` project outputs the memory left in EdgeLock SE050 security IC.

Note: *The execution of the `se05x_minimal` project is shown as an example. The steps detailed in this section can be replicated to run any other test example included as part of the EdgeLock SE050 Plug & Trust middleware.*

To execute the `se05x_minimal` project example, follow these steps:

1. Attach a USB cable from the computer to the K64F OpenSDA debug USB connector as shown in Figure 37.

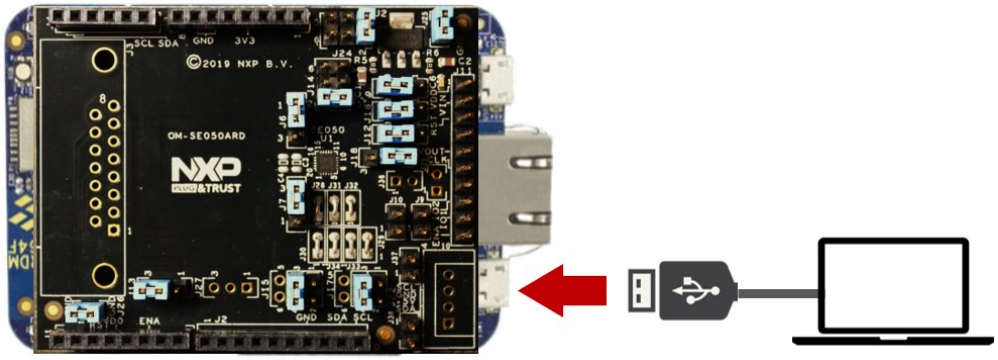

Figure 37. Connect boards to the laptop

2. Open TeraTerm. Click ***Serial*** option and select from the drop down list the COM port number assigned to your FRDM-K64F. Then go to Setup > Serial Port and configure the terminal to 115200 baud rate, 8 data bits, no parity and 1 stop bit and click OK as shown in Figure 38:

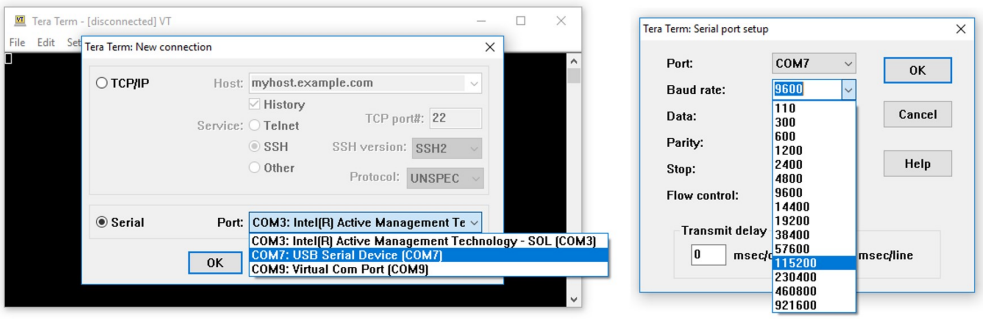

Figure 38. Configure TeraTerm

3. Select the `se05x_minimal` as the project to be executed. For that, follow the steps shown in Figure 39:
   a. In the Project Explorer window, go to ***Debug*** folder and open the ***Makefile*** file (under cmake_project_frdmk64f).
   b. The **BUILD_TARGET** contains the name of the project to be executed. Write `se05x_minimal` in the **BUILD_TARGET** variable
   c. Click on the arrow on the "hammer" icon in the top bar menu of the MCUXpresso.
   d. Select ***1 Debug (Debug build)***. Wait a few seconds until the build operation completes.

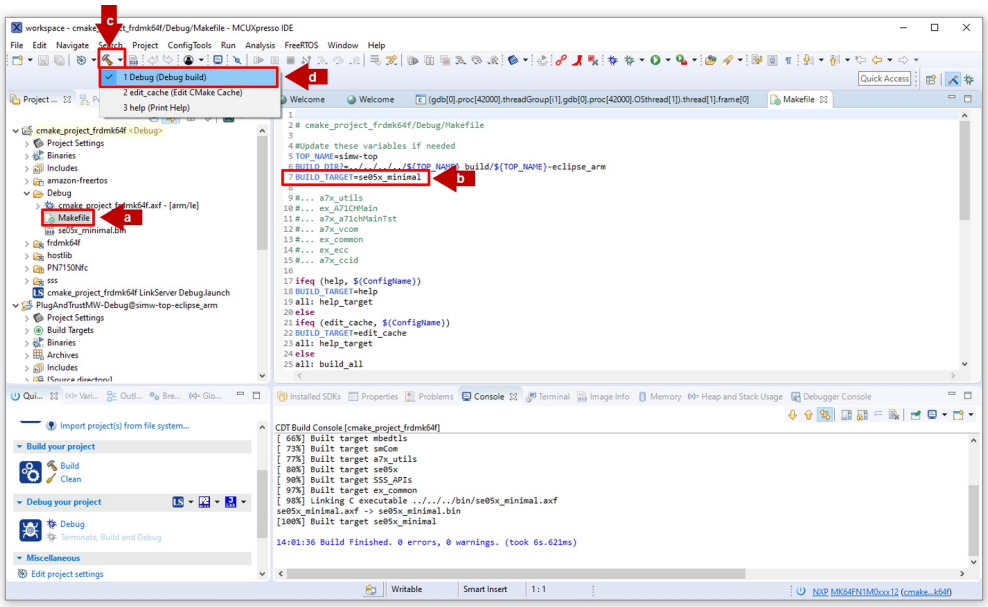

Figure 39. Debug EdgeLock SE050 Plug & Trust middleware `se05x_minimal` project example

4. Go to the MCUXpresso Quickstart Panel and click ***Debug*** button as shown in Figure 40. If there is more than one probe attached, you have to the select CMSIS-DAP debug probe from the list. Wait a few seconds until the project executes:

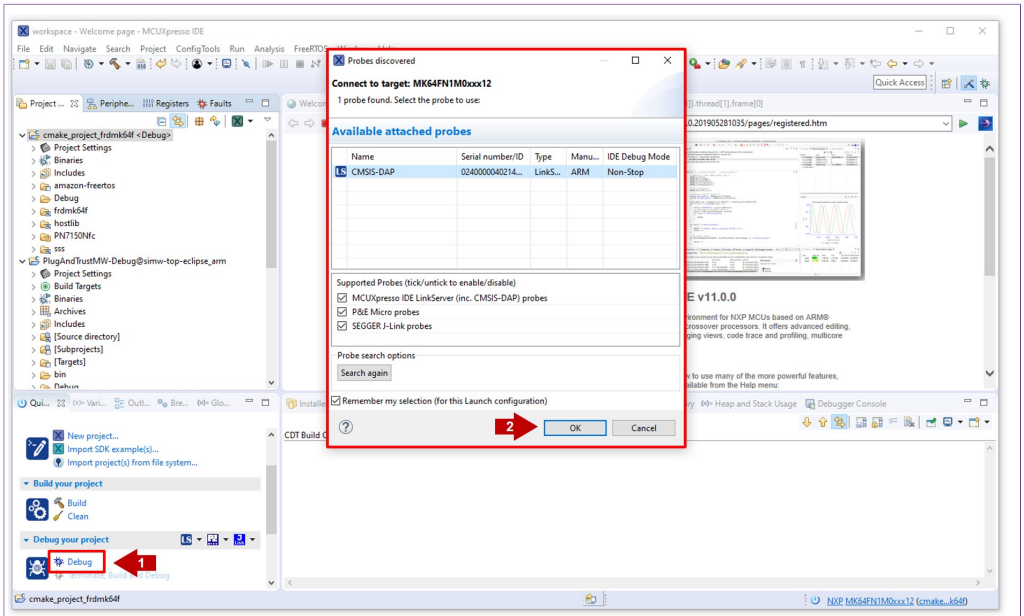

Figure 40. Debug `se05x_minimal` project example

5. When it executes, it will automatically stop in a breakpoint. Click on ***Resume*** to allow the software to continue its execution as shown in Figure 41.

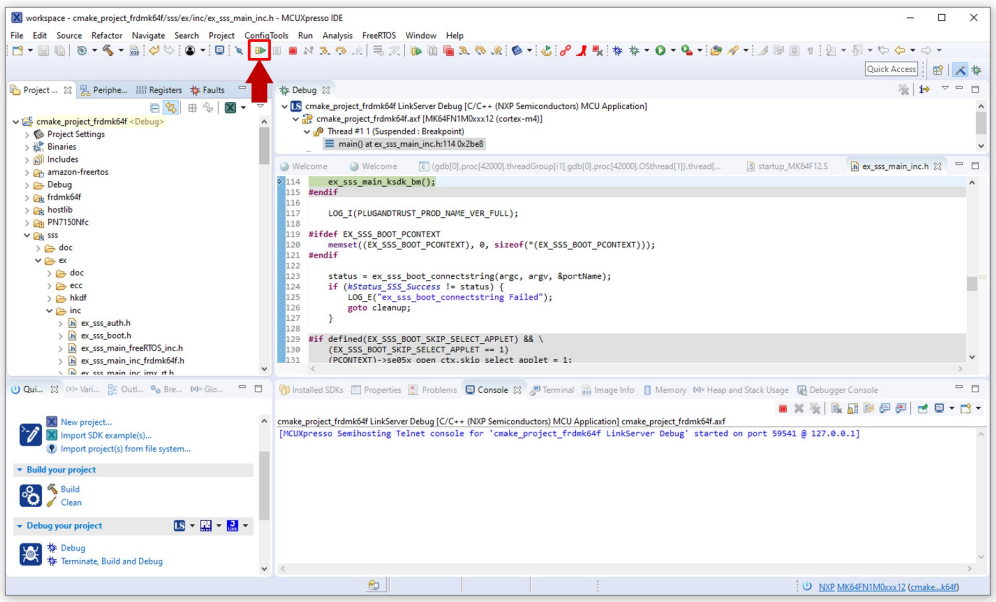

Figure 41. Resume `se05x_minimal` project example

6. The project example should now be running into your FRDM-K64F. If it is running successfully, the TeraTerm logs should indicate the available memory in SE050 security IC (in this case, 592) as can be seen in Figure 42 .

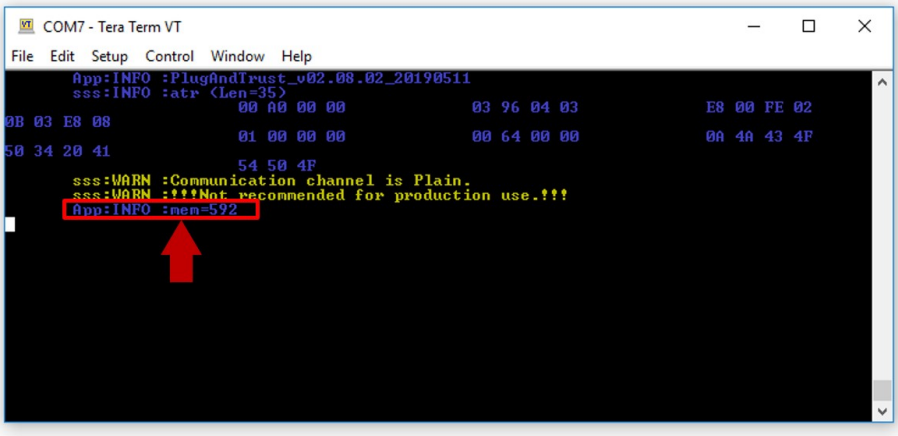

Figure 42. TeraTerm logs - `se05x_minimal` project example

7. The same operation can be repeated with any of the other EdgeLock SE050 Plug & Trust middleware project examples.

6 Appendix A: Install MCUXpresso IDE

MCUXpresso is a free-of-charge, code size unlimited, easy-to-use IDE for Kinetis and LPC MCUs, and i.MX RT crossover processors. To install it, do the following:

1. Go to MCUXpresso and click the download button as indicated in Figure 43:

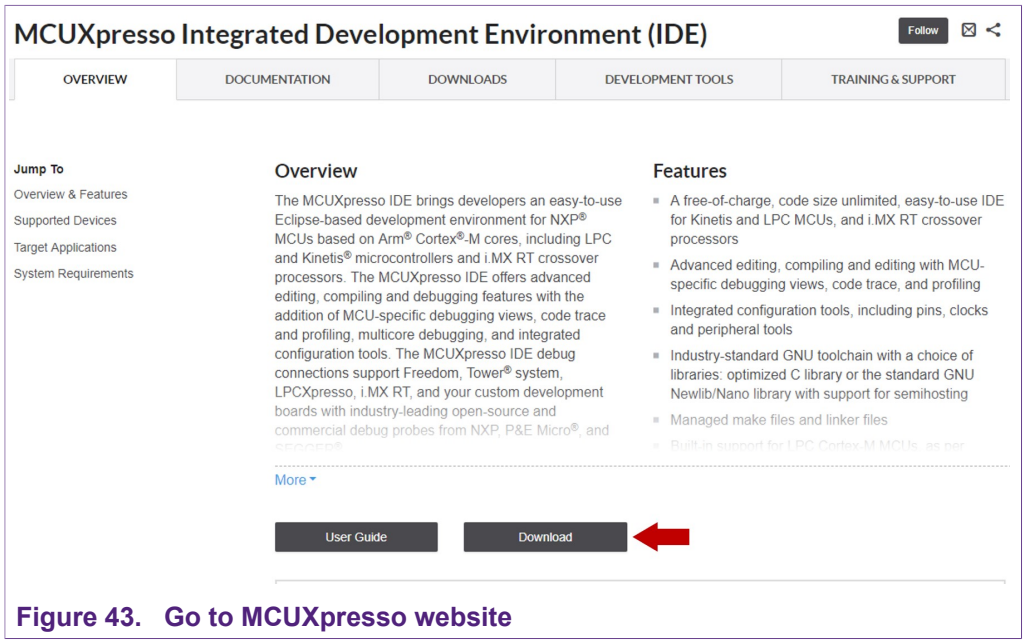

Figure 43. Go to MCUXpresso website

2. You will be asked to sign-in with your account at the NXP website. If you do not have an account, click on ***Register Now*** as shown in Figure 44:

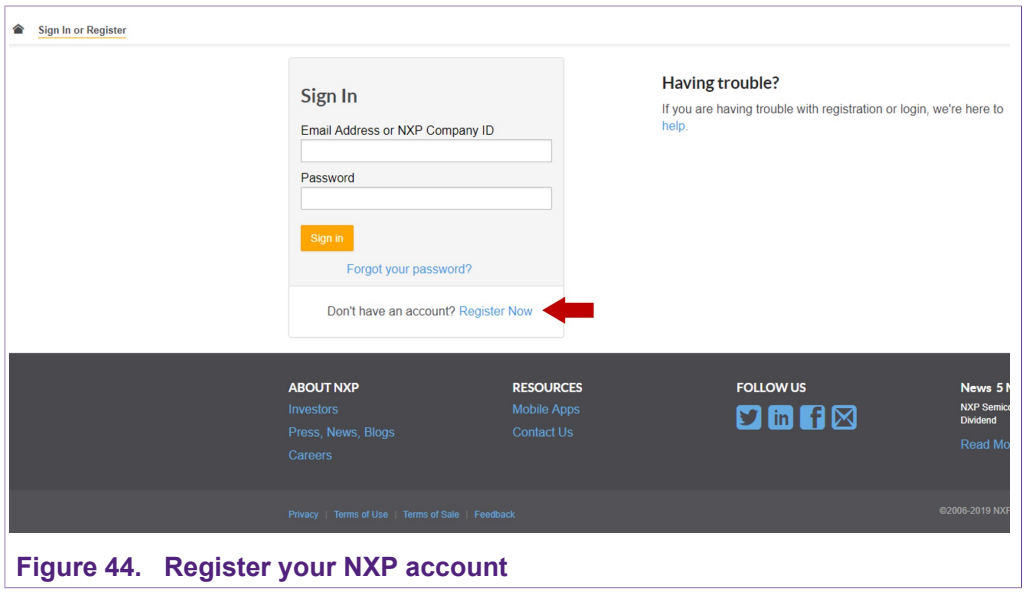

Figure 44. Register your NXP account

3. If you already have an account, you can directly type your (1) email address, (2) password and (3) click sign-in button as shown in Figure 45:

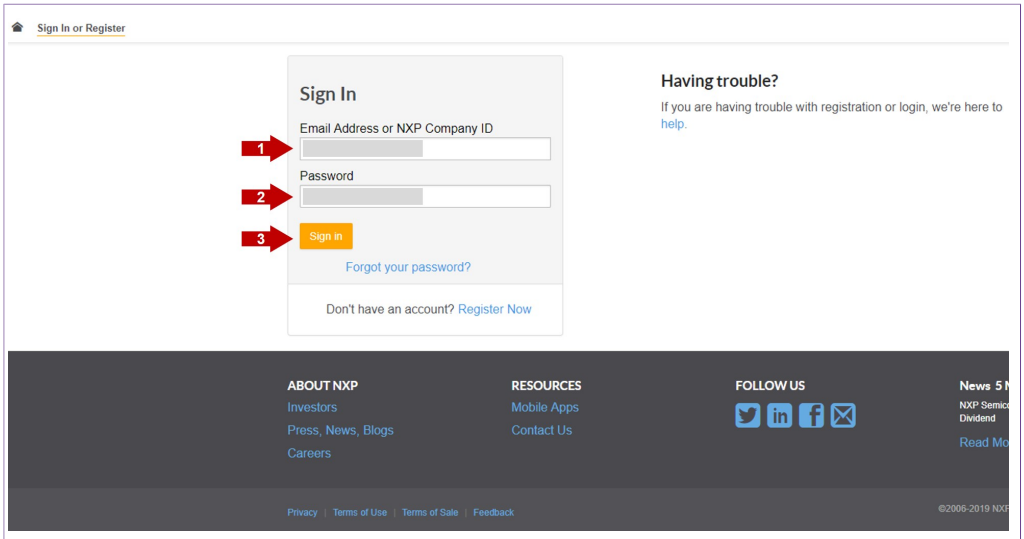

Figure 45. Sign-in in NXP website

4. Click on MCUXpresso IDE as shown in Figure 46:

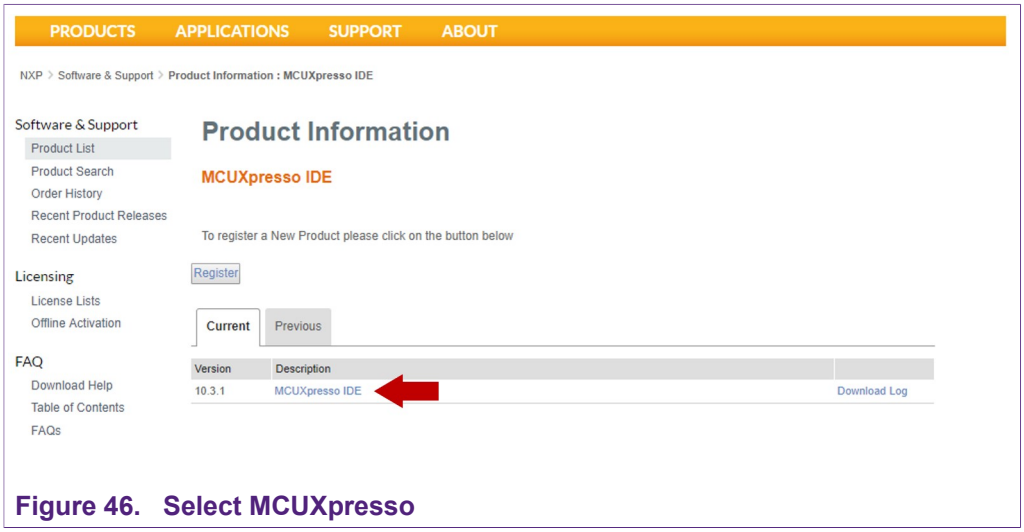

Figure 46. Select MCUXpresso

5. Accept software terms and conditions as shown in Figure 47:

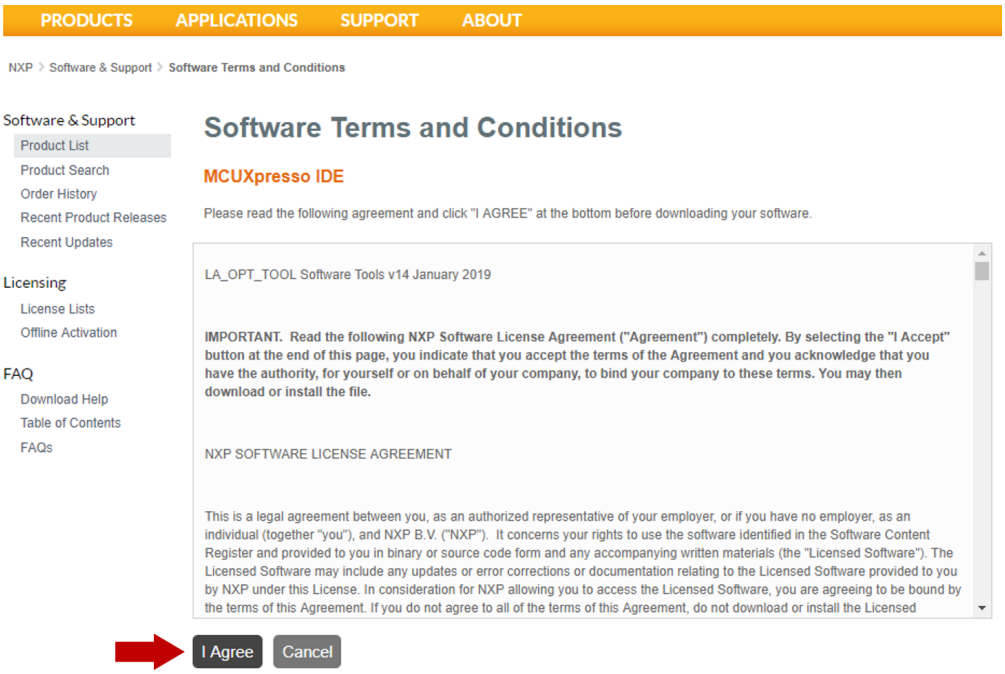

Figure 47. Accept software terms and conditions

6. Select your MCUXpresso product version and click on the corresponding ***File Name*** to start the download as shown in Figure 48:

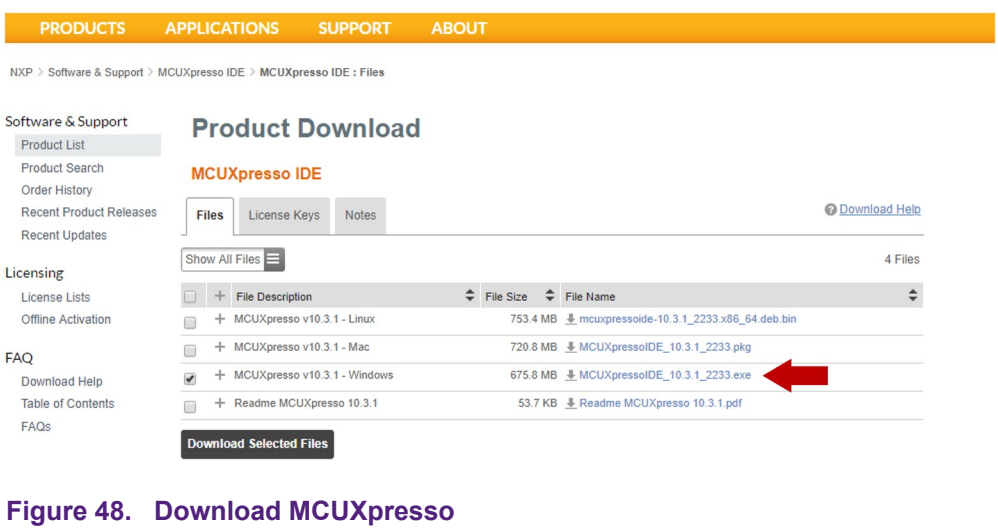

Figure 48. Download MCUXpresso

7. Double click on the installer file and follow the setup wizard until MCUXpresso installation is completed. Please, make sure you allow the installation of the additional drivers required by MCUXpresso during the installation process as shown in Figure 49, Figure 50, Figure 51 and Figure 52:

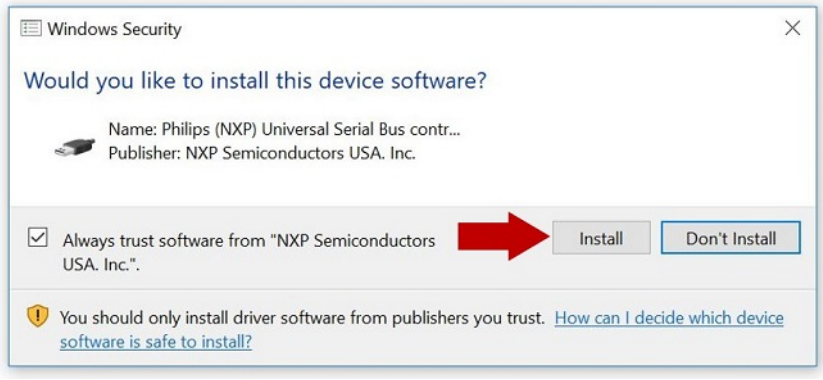

Figure 49. Install MCUXpresso required drivers I

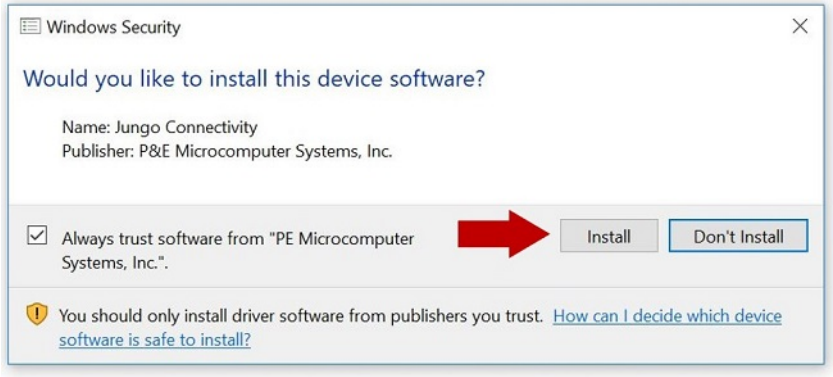

Figure 50. Install MCUXpresso required drivers II

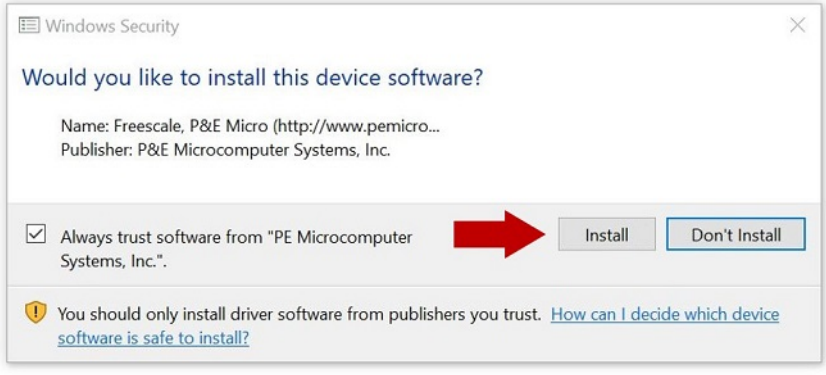

Figure 51. Install MCUXpresso required drivers III

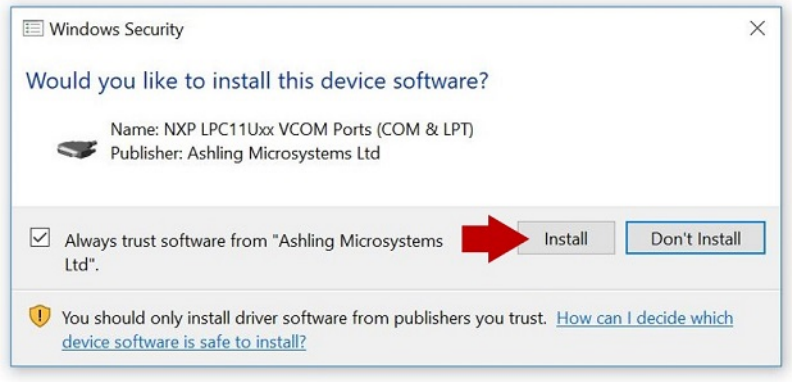

Figure 52. Install MCUXpresso required drivers IV

7 Appendix B: Install CMake

CMake is an open-source, cross-platform family of tools that helps you build C/C++ projects on multiple platforms using a compiler-independent method. It has minimal dependencies, requiring only a C++ compiler on its own build system. SE050 middleware leverages on CMake to generate native makefiles and workspaces that can be used in the compiler environment of your choice. To install CMake:

1. Go to CMake downloads page: https://cmake.org/download/
2. Scroll down and select your binary distribution. For this guide, the binary distribution is Windows as shown in Figure 53:

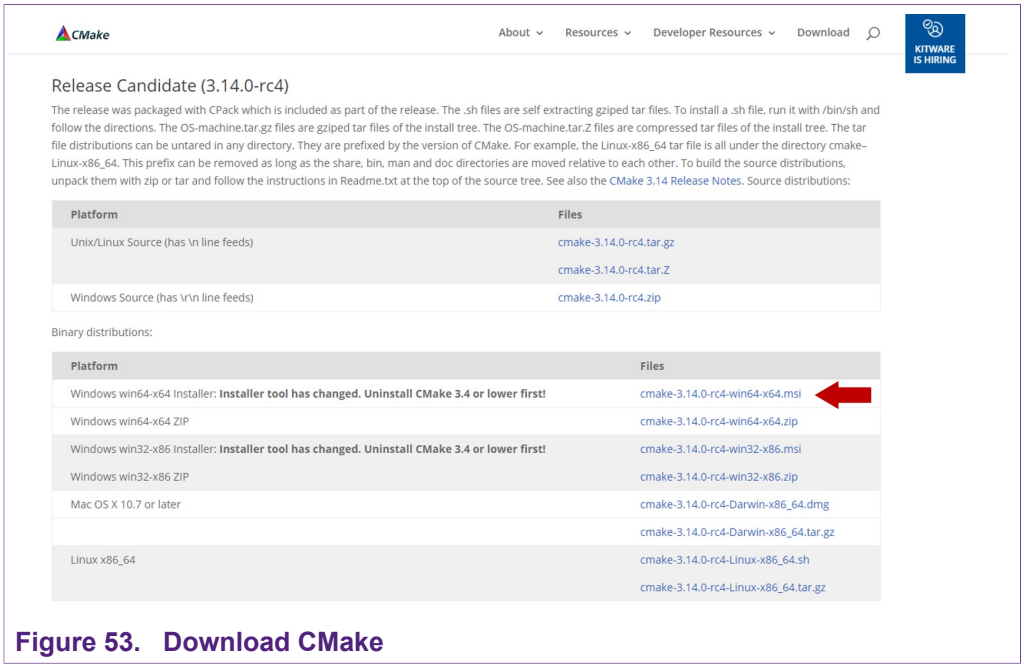

Figure 53. Download CMake

3. Double click on the downloaded installer file. Windows Defender SmartScreen might pop-up the wizard shown in Figure 54:

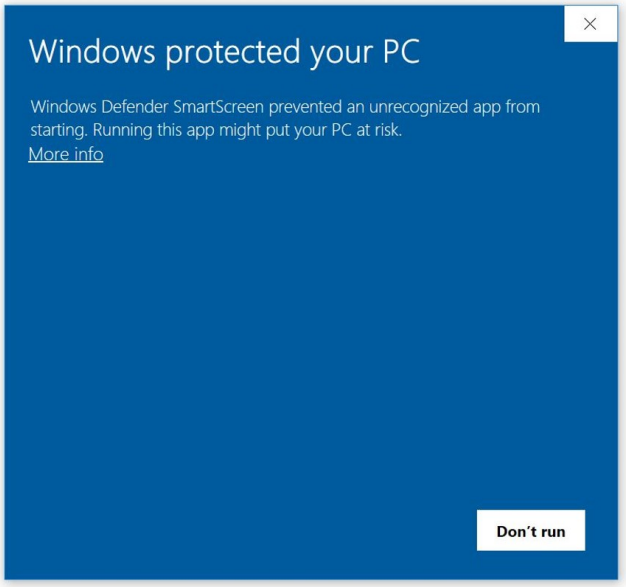

Figure 54. Execute CMake installer

4. If this is your case: Click (1) on ***More info*** and then (2) click on ***Run anyway*** as shown in Figure 55:

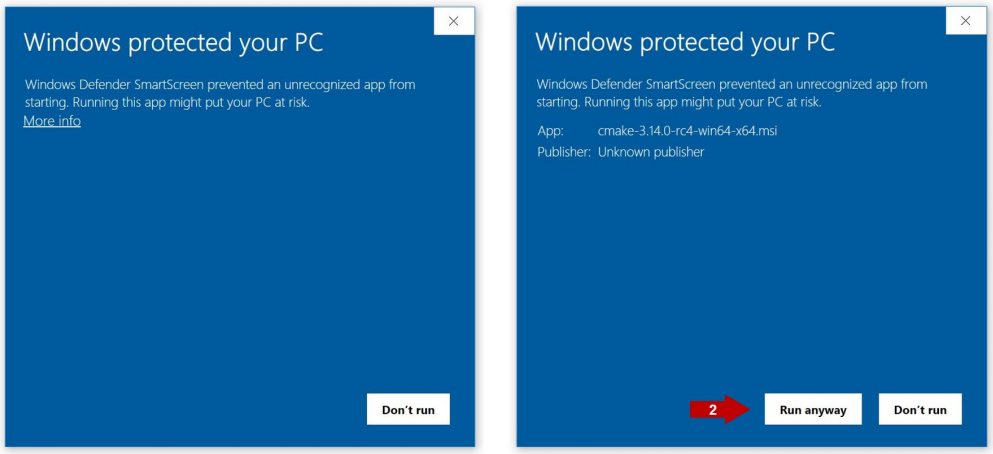

Figure 55. Run the CMake installer (II)

5. The CMake installation wizard will open. Click (1) ***Next*** and (2) **accept** the End-User License Agreement as shown in Figure 56:

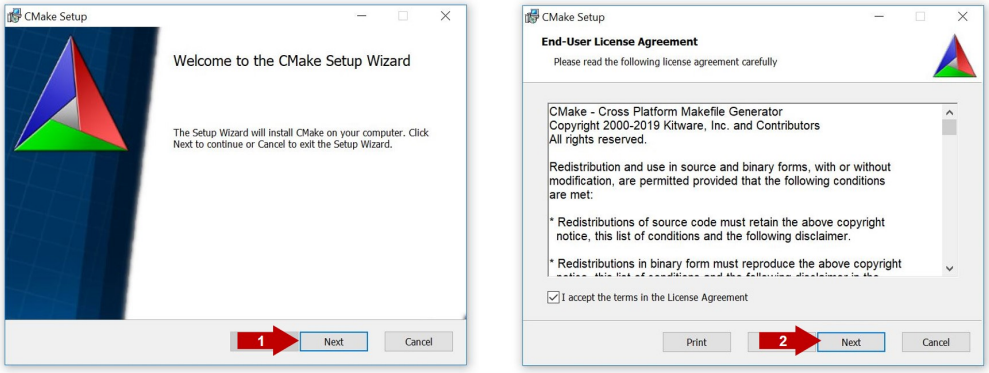

Figure 56. CMake installation wizard

6. As part of the CMake setup, (1) ***Add Cmake to the system PATH for all users*** and (2) click ***Next*** as shown in Figure 57:

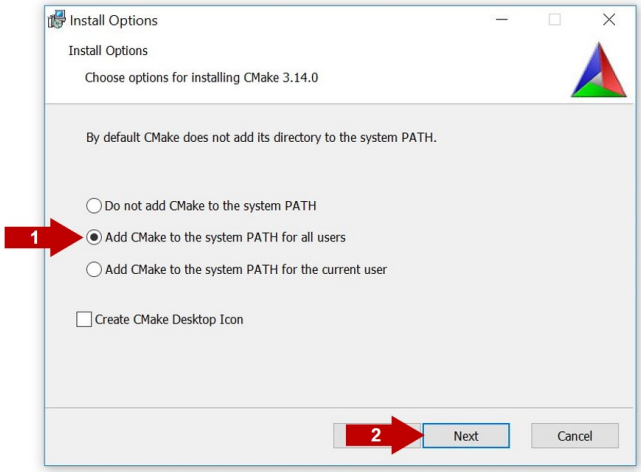

Figure 57. Add CMake path

7. Select a destination folder, (1) click ***Next*** and then (2) click ***Install*** as shown in Figure 58:

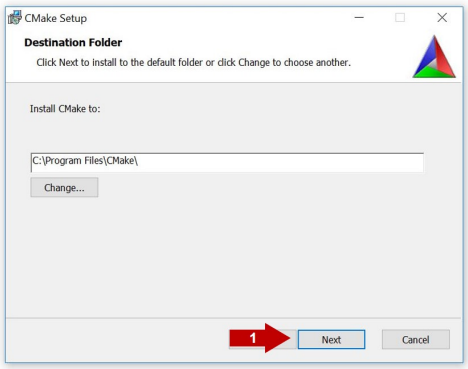

Figure 58. Install CMake

8. Wait a few seconds until the installation is completed and click ***Finish*** as shown in Figure 59:

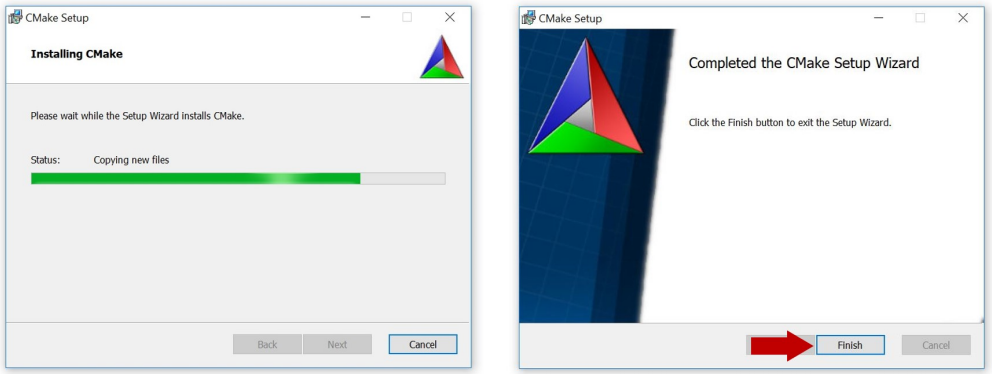

Figure 59. Complete CMake installation

8 Appendix C: Install Python 3.7.x 32-bit version

Use these screenshots to install Python 3.7.x in your Windows machine:

1. Go to https://www.python.org/downloads and download **Python v.3.7.x 32 bit version**. Make sure you download Python v3.7.x 32 bit version.

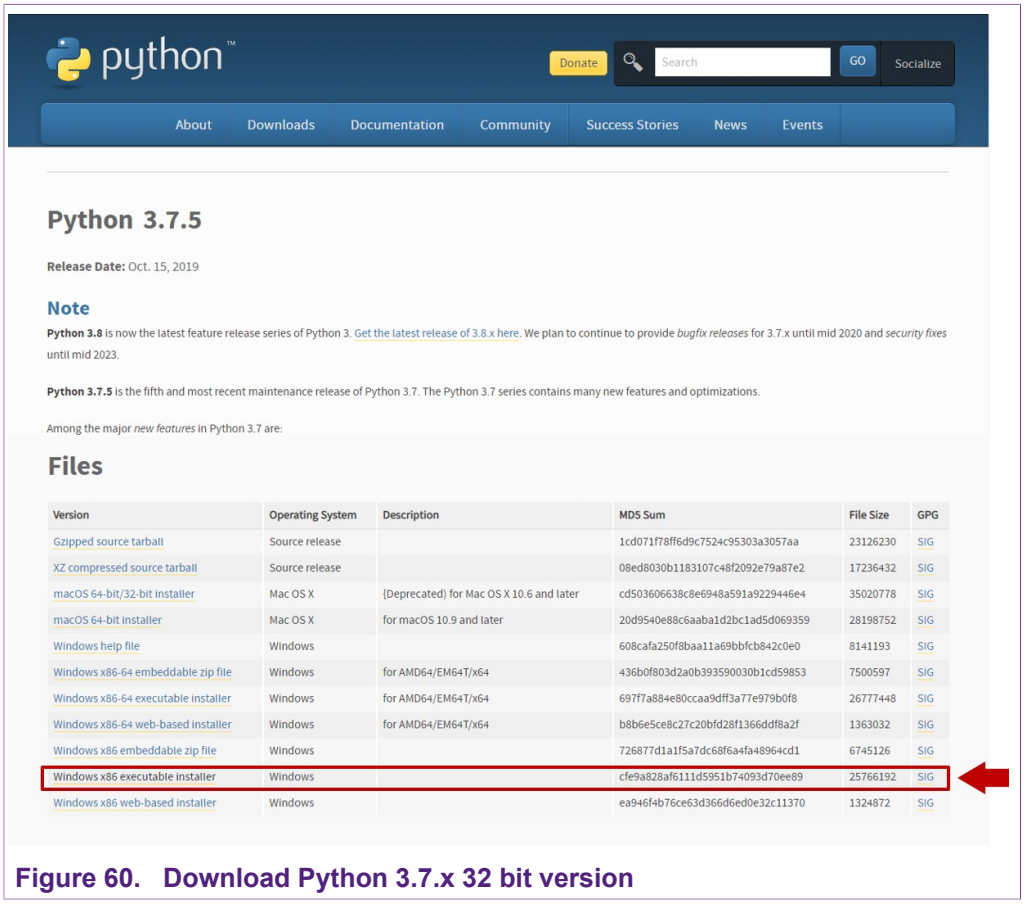

Figure 60. Download Python 3.7.x 32 bit version

2. Double click on the downloaded installer file. Select the "*Install launcher for all users*" and "*Add Python 3.7 to Path*" options and click *Install Now* as indicated in Figure 61:

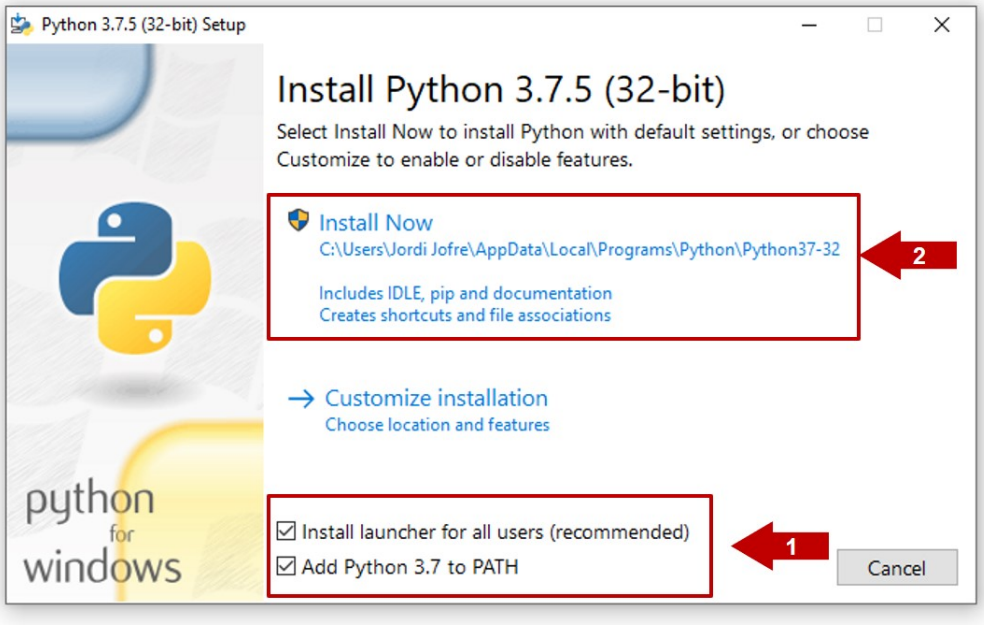

Figure 61. Install Python 3.5.x 32 bit for Windows

3. Wait a few seconds until the installation is completed as indicated in Figure 62

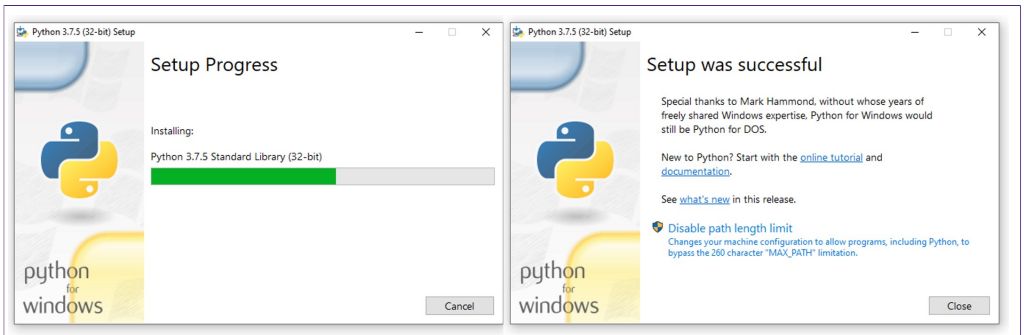

Figure 62. Python 3.5.x 32 bit installation completed

9 Legal information

9.1 Definitions

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

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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

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

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

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

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

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.

Security — While NXP Semiconductors has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP Semiconductors accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

9.3 Trademarks

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

Tables

Figures

Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 17 December 2019
Document identifier: AN12396
Document number: 534421