JN518x MCUXpresso Installation and User Guide

JN-UG-3136
Revision 1.0
20-Jun-2018

Contents

Preface

This manual describes the use of the NXP MCUXpresso Integrated Development Environment (IDE) to develop applications for the NXP JN518x family of wireless microcontrollers.

Note: MCUXpresso is fully detailed in its own documentation set, available from the NXP web site (See Support Resources on page 5).

This manual (JN-UG-3136) provides additional information needed to develop JN518x applications within MCUXpresso.

Organisation

This manual consists of three chapters and four appendices, as follows:

- Chapter 1 introduces MCUXpresso and provides installation instructions for the plug-ins that are required for developing JN518x applications
- Chapter 2 describes how to import JN518x application in MCUXpresso starting from an NXP Application Note
- Chapter 3 describes how to debug a running JN518x application via MCUXpresso
- The Appendices contain information and procedures that may be useful in using MCUXpresso

Conventions

Files, folders, functions and parameter types are represented in **bold** type.

Function parameters are represented in *italics* type.

Code fragments are represented in the Courier typeface.

This is a **Tip.** It indicates useful or practical information

This is a **Note.** It highlights important additional information.

*This is a **Caution**. It warns of situations that may result in equipment malfunction or damage.*

Acronyms and Abbreviations

Definitions of acronyms and abbreviations should follow the following format:

IDE	Integrated Development Environment
SDK	Software Developer's Kit

Related Documents

The list of related documents should follow the following format:

JN518x Datasheet (JN-DS-JN5189)

MCUXpresso IDE User Guide

MCUXpresso Installation and Licencing Guide

MCUXpresso IDE SWO Trace Guide

Support Resources

To access online support resources such as SDKs, Application Notes and User Guides, visit the Wireless Connectivity area of the NXP web site:

www.nxp.com/products/interface-and-connectivity/wireless-connectivity

All NXP resources referred to in this manual can be found at the above address, unless otherwise stated.

MCUXpresso and its documentation are available from:

www.nxp.com/mcuxpresso

For information on obtaining and installing the appropriate version of MCUXpresso to develop applications for JN518x chips, refer to Section 1.2

Trademarks

1 Introduction and Installation

The NXP MCUXpresso Integrated Development Environment (IDE) is employed as a platform for the development of wireless network applications to be run on NXP's JN518x family of wireless microcontrollers. MCUXpresso is an Eclipse-based IDE that provides editing, compiling, debug and Flash programming functionality.

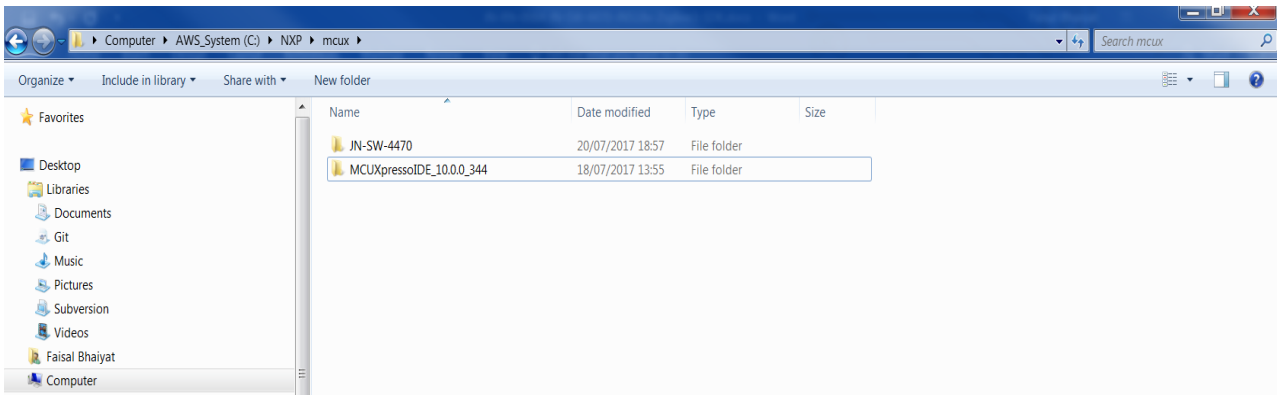
For information on how to obtain and install MCUXpresso along with the JN518x stack libraries, refer to Section 1.1.

1.1 MCUXpresso for JN518x Installation

It is recommended that before first installing MCUXpresso the user first installs the JN518x stack libraries SDK. This is so that the required workspaces are created.

Installation steps:

1. Download the JN518x stack libraries SDK installer (for example the **JN-SW-4470 JN518x Zigbee 3.0 vXXXX.exe**). These SDK's are distributed as a self-extracting executable.
2. Launch the JN518x stack libraries SDK executable in to a convenient location, such as C:\NXP\mcux. This SDK contains the device drivers, framework, connectivity stack along with example projects.
3. Download **MCUXpresso** – see Section1.2. MCUXpresso toolchain must be installed at the same location as the JN518x stack libraries SDK. Launch the **MCUXpresso** installer and follow the on-screen instructions to install.
4. After installing the SDK and MCUXpresso IDE you should have the directory structure shown below:



5. To add JN518x support to **MCUXpresso** it is necessary to add a number of files to the installation. These files are provided within the SDK which was installed during step 2. These can be found in the following folder

   'C:\NXP\mcux\JN-SW-4470\tools\lpcxpresso\bin'

6. Copy the '**C:\NXP\mcux\JN-SW-4470\tools\lpcxpresso\bin'** folder to '**C:\NXP\mcux\MCUXpressoIDE_10.1.1_606\ide'**. There should already be a 'bin' folder there and the copied version should be merged with it. If using Windows Explorer to copy the folder across, Windows will complain that it is

there already: this is correct, so click ‘Yes’ and/or ‘Copy and Replace’ as required.

> ***Caution:*** *Note that using the command line to copy the files may incorrectly set the access permissions whereas CTRL-C and CTRL-V in Windows Explorer work fine*

7. Run ‘mcuxpressoide.exe’ and select a folder to use as Workspace. This should be the folder where the SDK was installed

   ‘C:\NXP\mcux\JN-SW-4470\workspace’.

8. JN518x bootloader requires an image signature to verify the validity of the image. The Binary image generated is signed after the image is built in a two-stage process. The image signing tool is implemented in python. This requires an installation of python to exist. Python 2.7 is required.

   Python can be downloaded from the following web page:

   https://www.python.org/downloads/

   Once python is installed, it should be added to the windows path.

9. The python file is now using a crypto module; therefore, you may have to install the module to compile successfully:

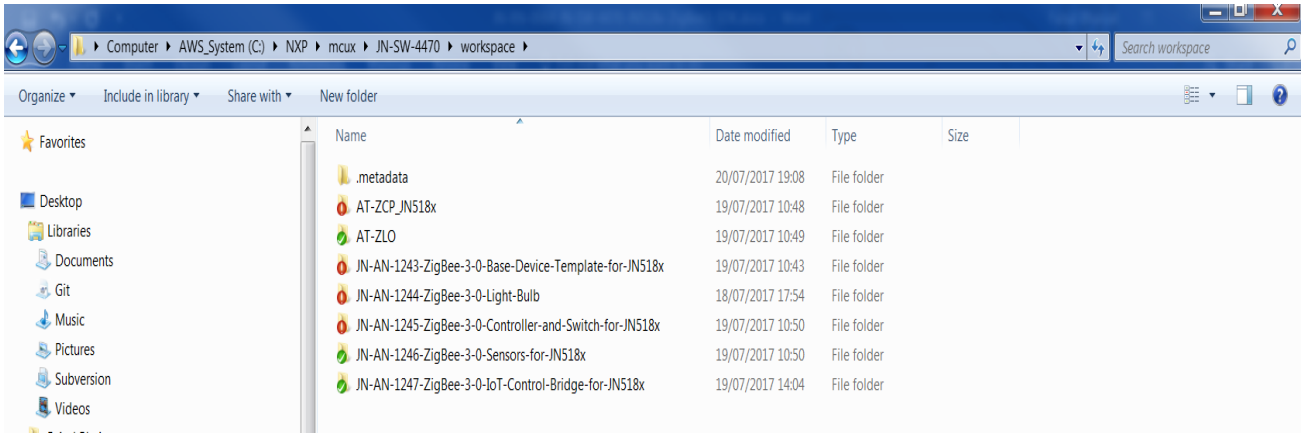
   C:\Python27\Scripts>pip install pycryptodome

   (python 2.7.13 (https://www.python.org/downloads/release/python-2713/) is required to have the pip tool)

10. The JN518x Application Notes are not provided as part of the JN518x stack libraries SDK. They are provided separately as ZIP files. The Application Notes should be extracted into the workspace created in step 7.

    ‘C:\NXP\mcux\JN-SW-4470\workspace’

    The directory structure may look like this:



1.2 MCUXpresso Installation

MCUXpresso can be obtained from the following NXP web page:

http://www.nxp.com/mcuxpresso/ide

Caution: *The recommended version of MCUXpresso to be used with an SDK can be found in the JN518x SDK Release Notes. This is the version with which the libraries within the SDK were compiled and verified. Other versions of MCUXpresso may not be compatible with the contents of the SDK and cannot be guaranteed to work or be supported with the JN518x devices.*

To obtain **MCUXpresso** and install it on your development machine:

1. If you do not already have a web account with NXP, navigate to www.nxp.com and create an account.
2. Sign in to your NXP web account.
3. Navigate to the page www.nxp.com/mcuxpresso/ide
4. Select the **Downloads** tab and then click the **Download** button.
5. Check whether the displayed version is the recommended version indicated above:

- If it is the recommended version, download it.
- If it is not the recommended version, click **Previous** and then select the recommended version and download it.

Full installation details are provided in the **MCUXpresso** *IDE Installation and Licensing Guide*, available on the **Documentation** tab of the above web page.

Tip: To develop JN518x applications without limitation we recommend that you purchase the Pro edition of MCUXpresso, however, with the free edition you can develop with applications up to 256KB

1.3 Installation of ZPSConfig plugins:

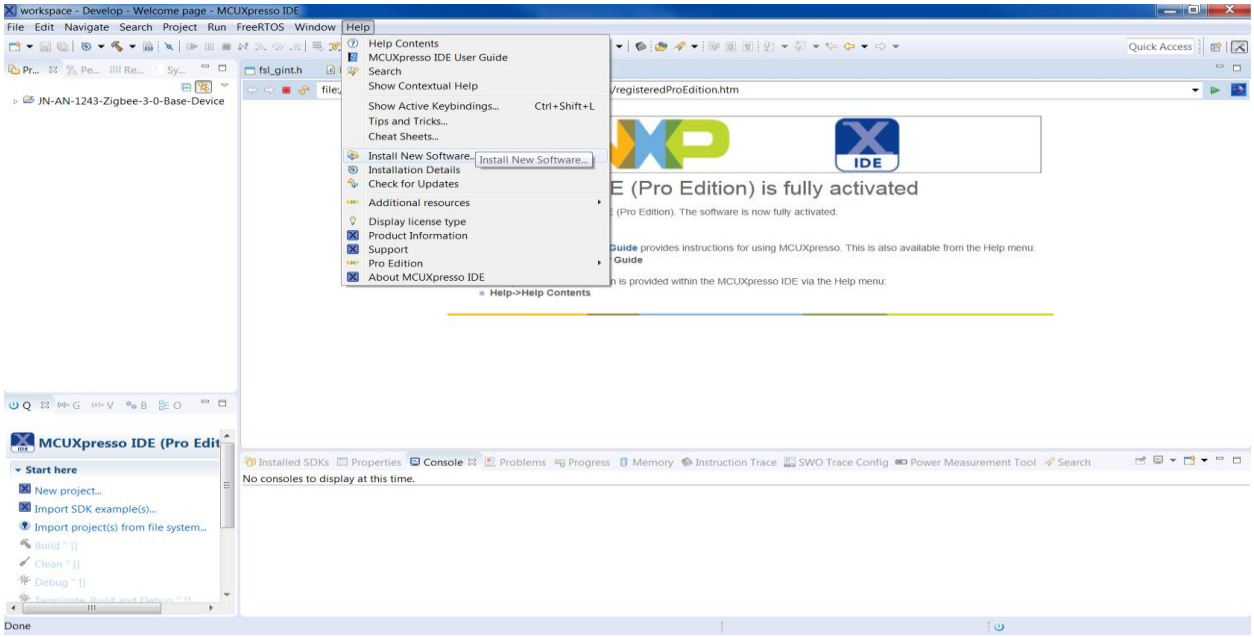
The Zigbee 3.0 stack configuration plugins have been updated to be hosted as part of the MCUXpresso IDE.

The plugins and features can be found located in the SDK folder structure as below:

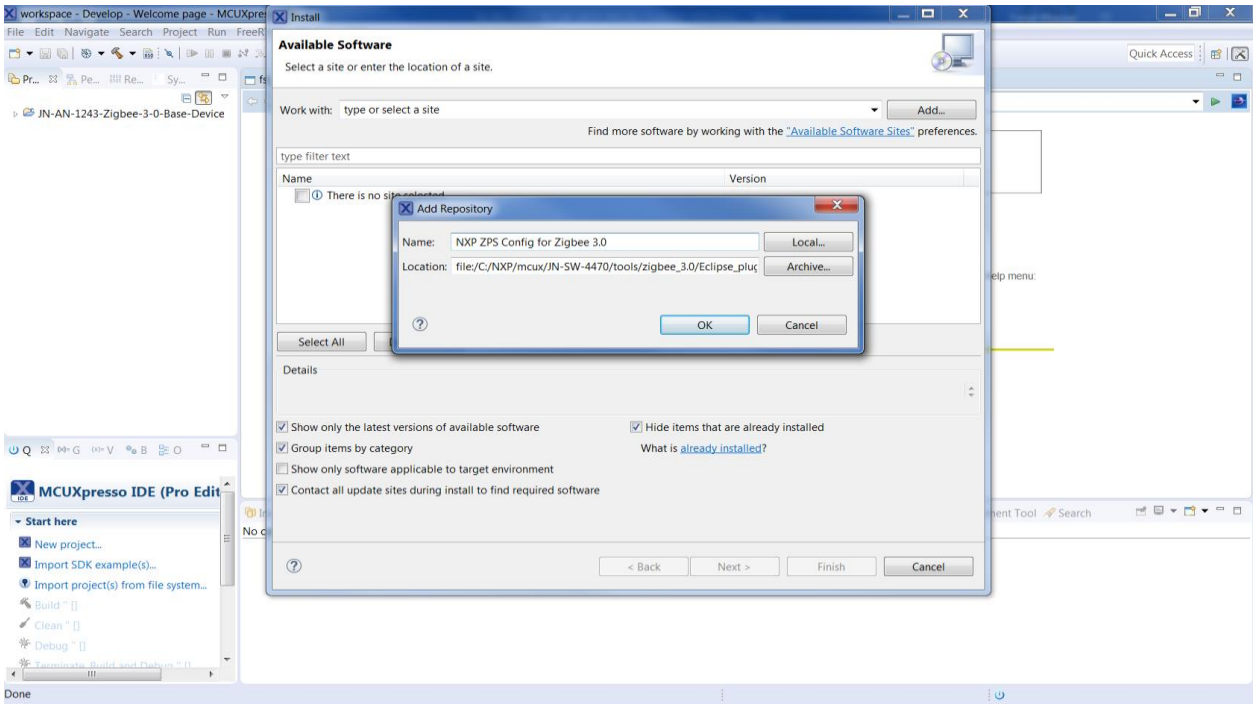
C:\NXP\mcux\JN-SW-4470\tools\zigbee_3.0\Eclipse_plugins\com.nxp.sdk.update_site

To add the plugins

1. On the top menu pane of the MCUXPresso select the *Help->Install New Software* option.



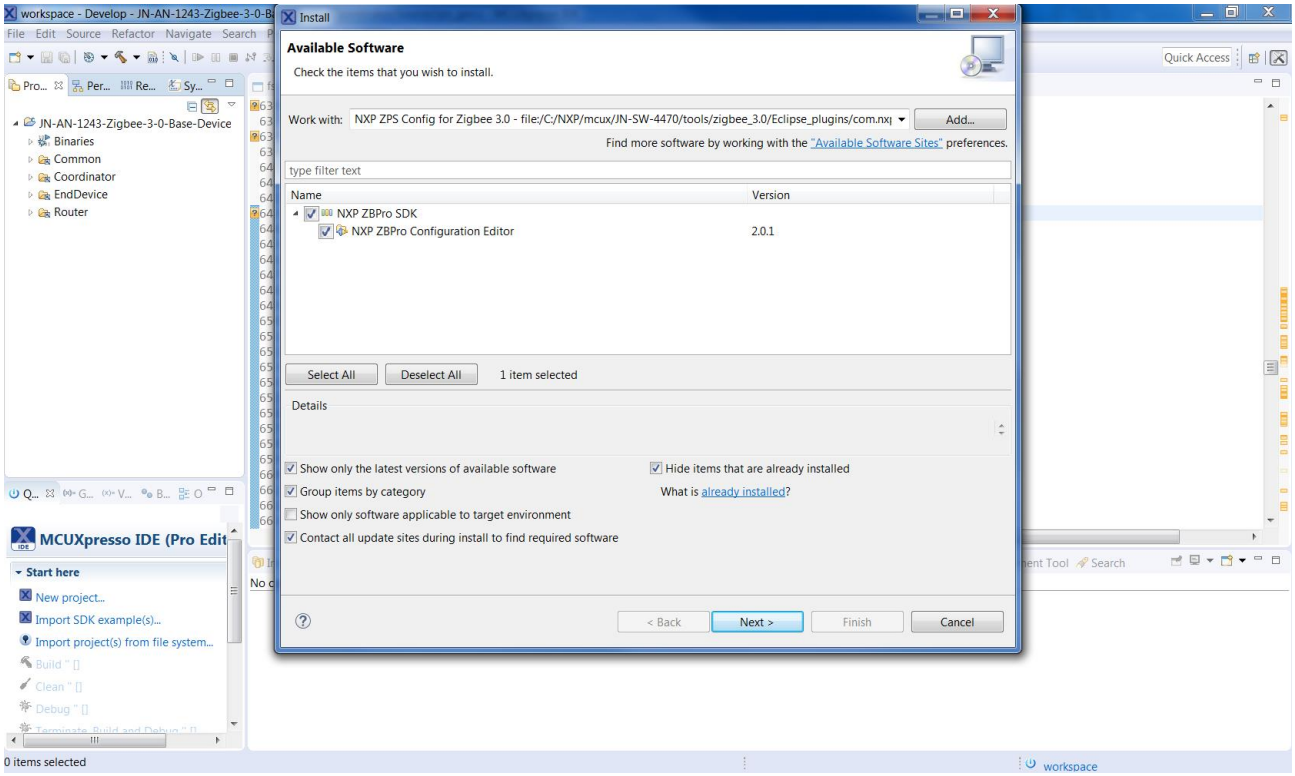
2. This should open a pop up menu selection as below:



Give the plugins a Name. This can be anything. Select the local button and browse to the location of the plugins which should be

C:/NXP/mcux/JN-SW-4470/tools/zigbee_3.0/Eclipse_plugins/com.nxp.sdk.update_site/

3. This should pull the available features to install;

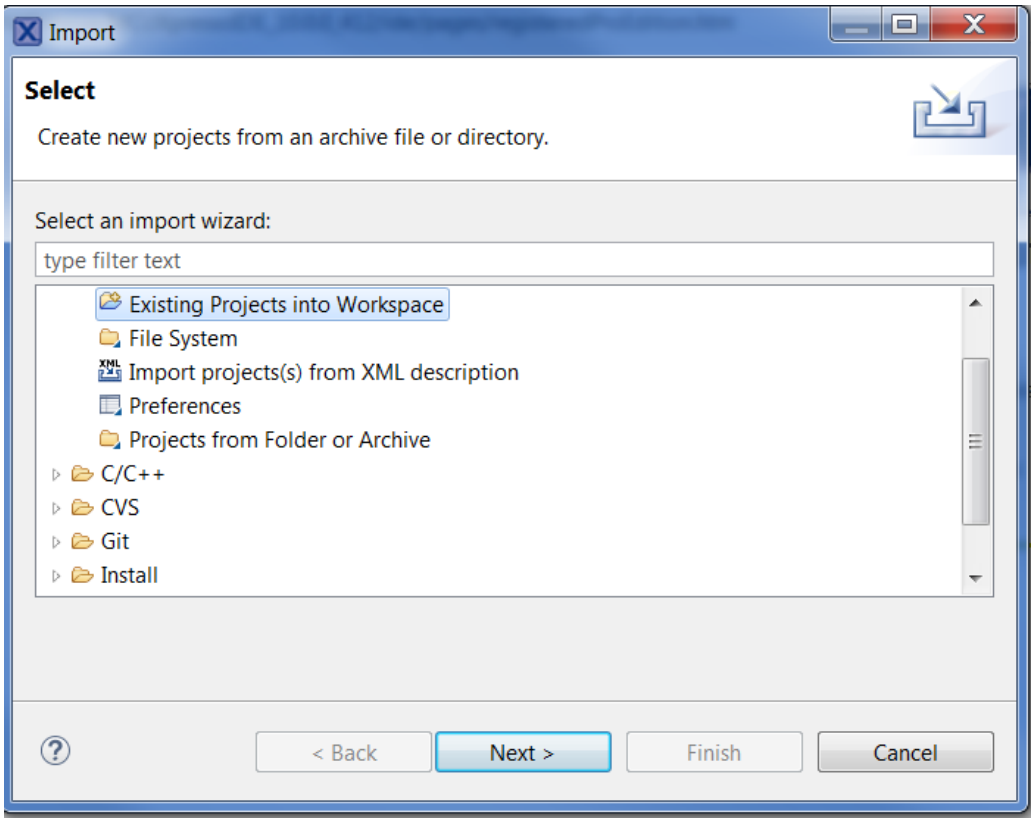


4. Select the features listed and then press the “Next” button. Accept the subsequent Licence agreement and warning of Signature validation. Then press the “Finish” button and let MCUXpresso restart.
5. If the plugins have been successfully installed it will be listed in the “already installed” software list.

2 Importing Projects into MCUXpresso

To import the example projects into MCUXpresso, click *File -> Import…* This will bring up a dialogue box.

Select *General -> Existing Projects into Workspace*, then press '*Next*':



The next dialogue box allows you to select the location that the projects can be found at. '*Select root directory*' will already be selected.

Click the '*Browse….*' Button to the right and then navigate to the folder where the SDK was installed. Navigate further into '**C:\NXP\mcux\JN-SW-44XX\workspace'** then press '*OK'*.

Back in the original dialogue box ensure that '*Copy projects into workspace*' is not selected, then press '*Finish'*;

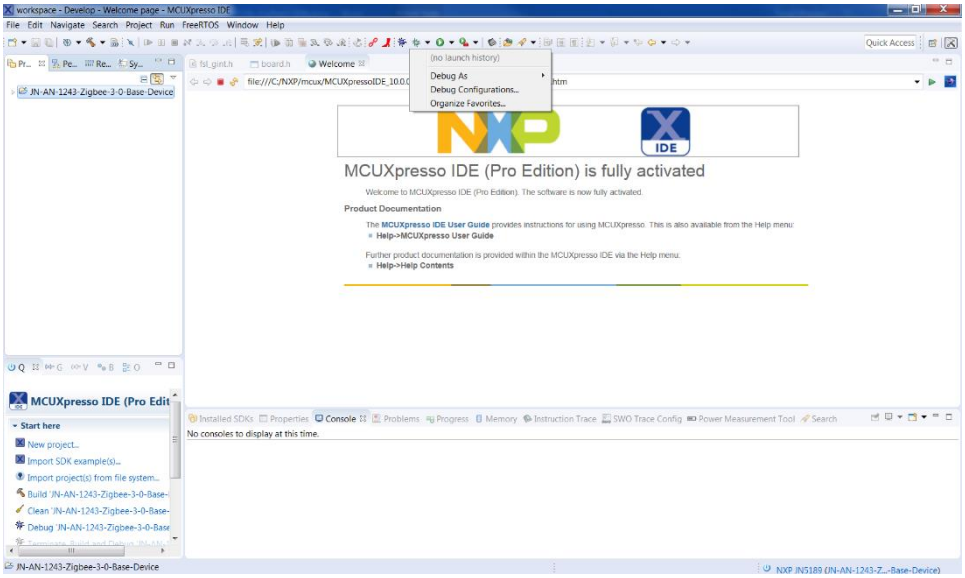
The projects will now appear in the Projects Explorer panel. To build one, select it and press the build button on the toolbar (looks like a hammer).

3 Debug Configuration

It is possible to flash the image for the JN518x by using the SWD (Single Wire Debug) functionality available in MCUXpresso.

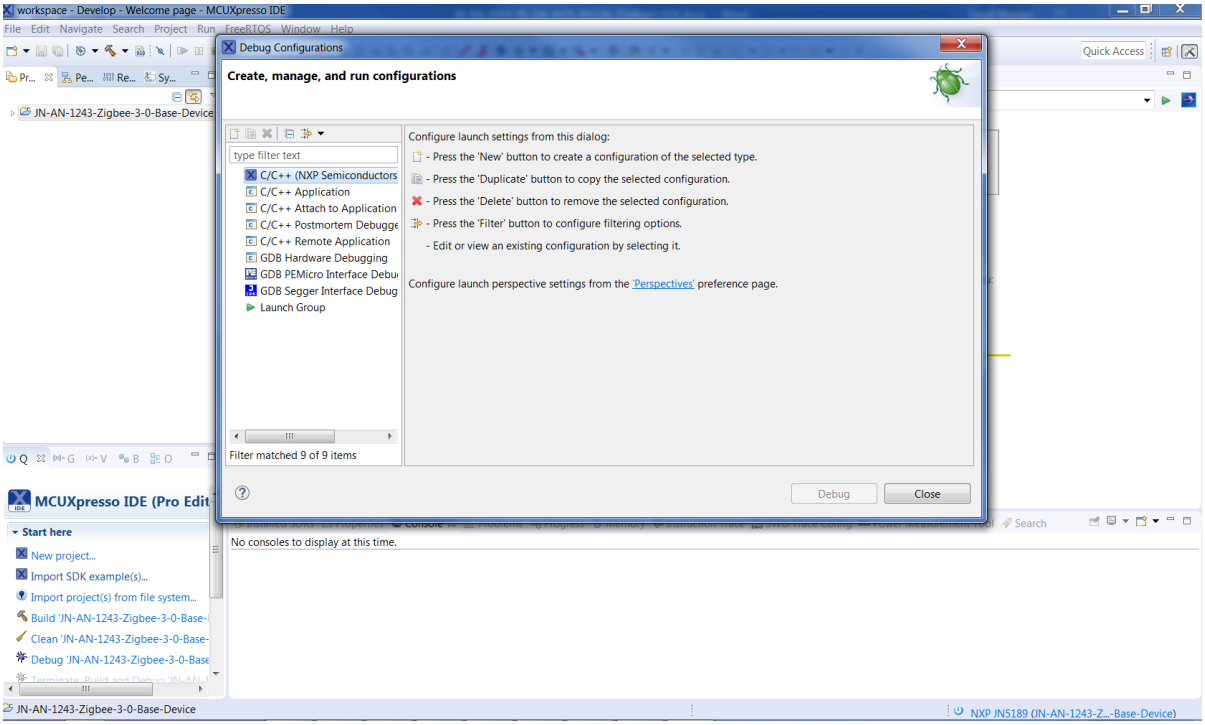
Select the dropdown menu next to the “bug” symbol on the selection panel of MCUXpresso IDE.

Select “Debug Configuration”.



The selection would bring up a pop up configuration box.

On the left-hand pane select the C/C++ NXP MCU Application and right click. That should provide drop down options, from the list select “*New*”.



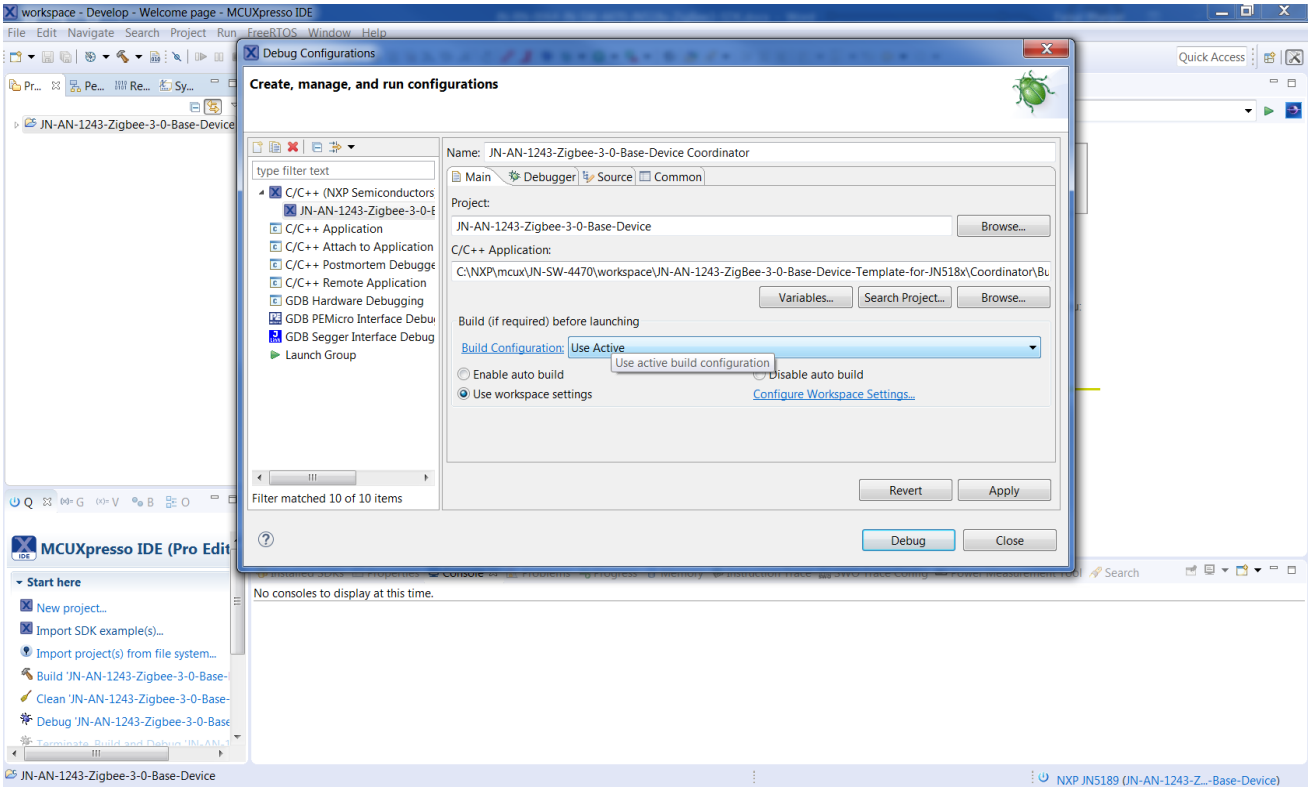
The subsequent pop up configuration box allows creating a debug configuration for the current build configuration.

In the Name box type a name of your choice for your current configuration. E.g. JN-AN-1243-Zigbee-3-0-Base-Device Coordinator

For the C/C++ Application box, use the browse button to find the ".axf" file for the image you want to debug/Flash.

e.g. **C:\NXP\mcux\JN-SW-4470\workspace\JN-AN-1243-ZigBee-3-0-Base-Device-Template-for-JN518x\Coordinator\Build\jn518x_mcux\Coordinator_JN5180_DONGLE.axf**

For the build configuration, use the dropdown menu to select "Use Active". Then save the changes by pressing the "Apply" button.



In the Debug Configuration pop up window, the Debugger needs to be configured.

Select the "Debugger" Tab.

This should update the existing Debug configuration pop up window with the possible configurations for the Debugger.

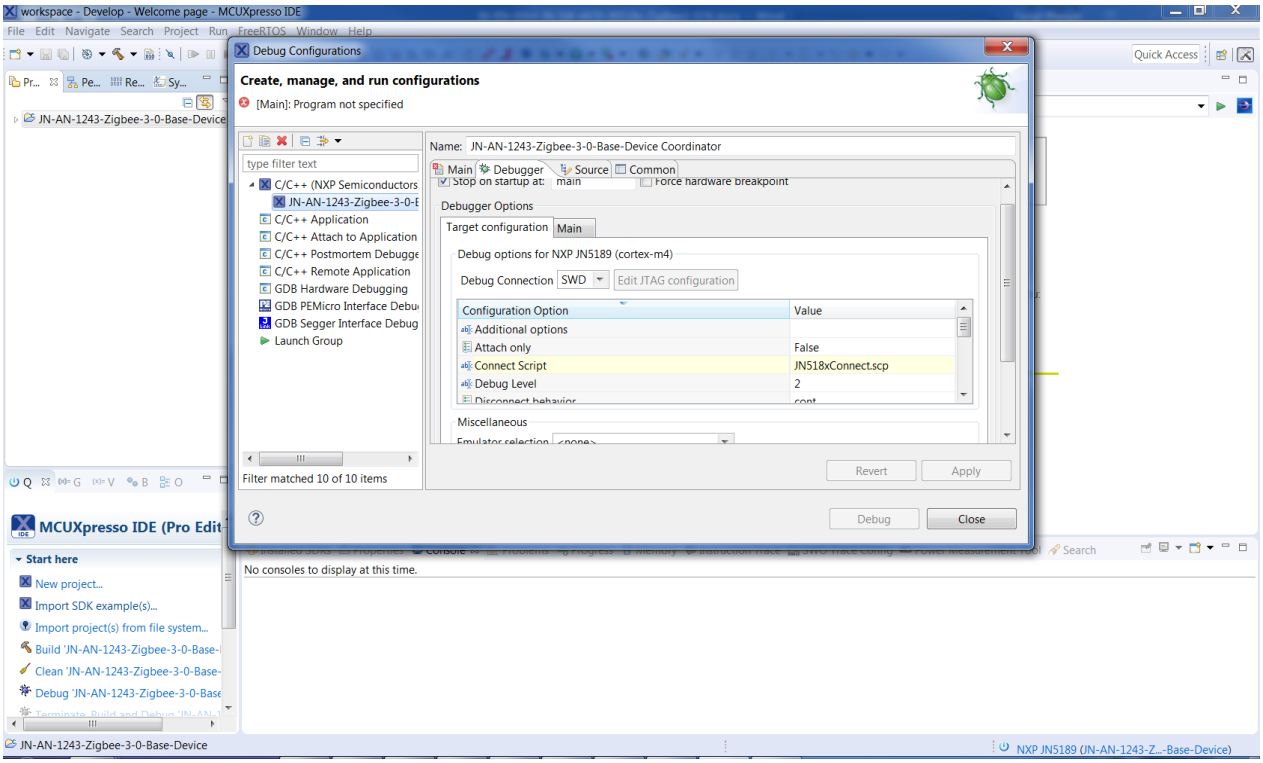
Tick the option for "Stop on startup at" and in the box type "main".

Update the "Connect script" to be "JN518xConnect.scp".

The "Emulator Selection" needs to be "LinkServer" and the "Debug options template" is "NXP JN5189 (*)"

To save these settings press "Apply".

Now just press "Debug" and connect your board if not already connected. The board will be programmed with the image and this should allow a debug session to be started.



By default, there are no debug symbols present so it's possible that no source code would be presented when running a debug session.

To build with debug symbols, the following needs to be added to the Makefile "DEBUG=HW" and a clean build of the application is now required.

Revision History

Version	Date	Description
1.0	20-06-2018	First release

Important Notice

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.

NXP Semiconductors

For the contact details of your local NXP office or distributor, refer to:

www.nxp.com

