ZIGBEE 3.0 TRAINING
K32W041AM

GETTING STARTED – INSTALLATION

FORREST SUN

PUBLIC USE

Company Public information is information that has been declared public knowledge by the owner and can freely be given to anyone inside or outside NXP without any possible damage to NXP

Public information can be shared with anyone, but only NXP spokespersons and nominated people can represent NXP publicly

Click here for more information on security awareness

Agenda

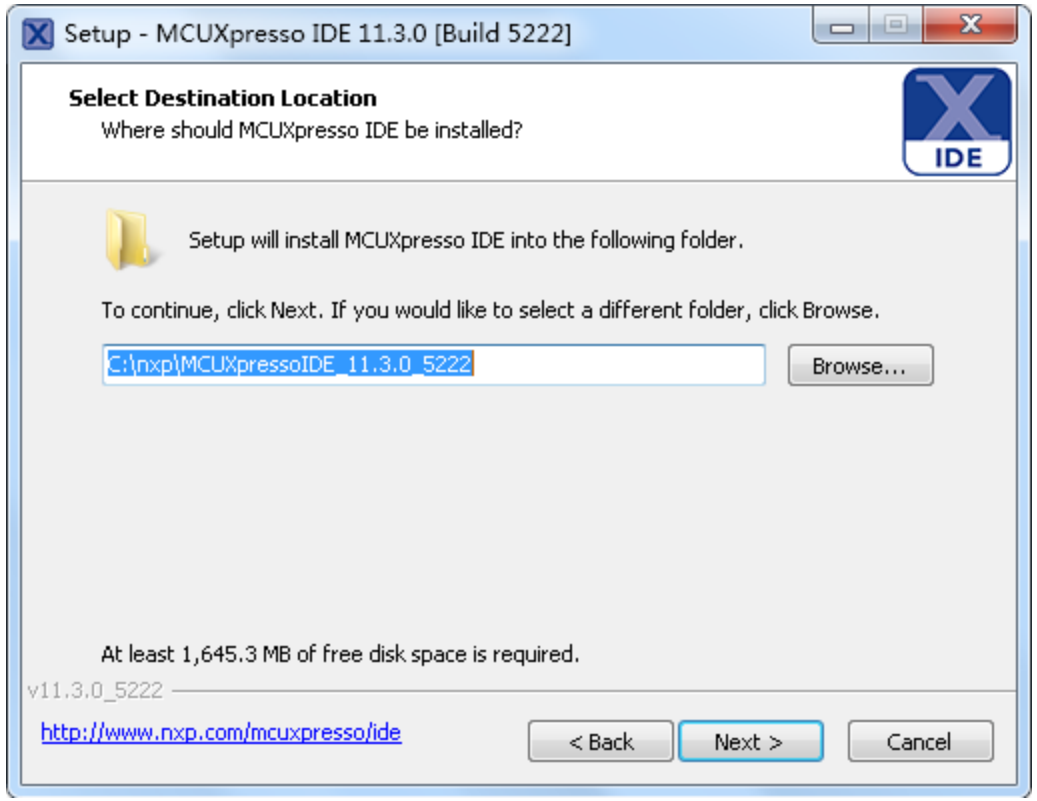
- **Objectives**
  - Describes the installation process to make use of the ZigBee 3.0 Application Notes
- **Audience**
  - Developers
- **Pre-requisites**
  - None
- **Reference Material**
  - JN-AN-1260 ZigBee 3.0 Getting Started
- **Duration**
  - 30 minutes

- **Contents**
  - Install MCUXpresso
  - Install SDK
  - Install ZB3.0 configuration tool
  - Install Python3.8
  - Install DK6 FlashProgrammer
  - Install Gawk(optional)
  - Install BeyondStudio(optional)

MCUXPRESSO

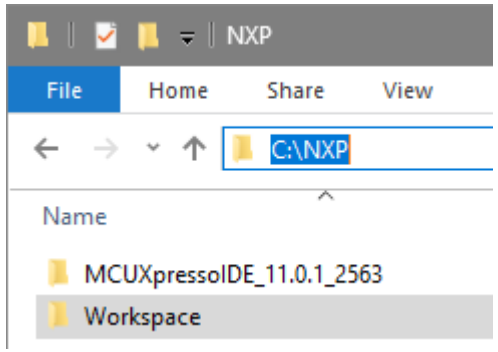
MCUXpresso – Installation

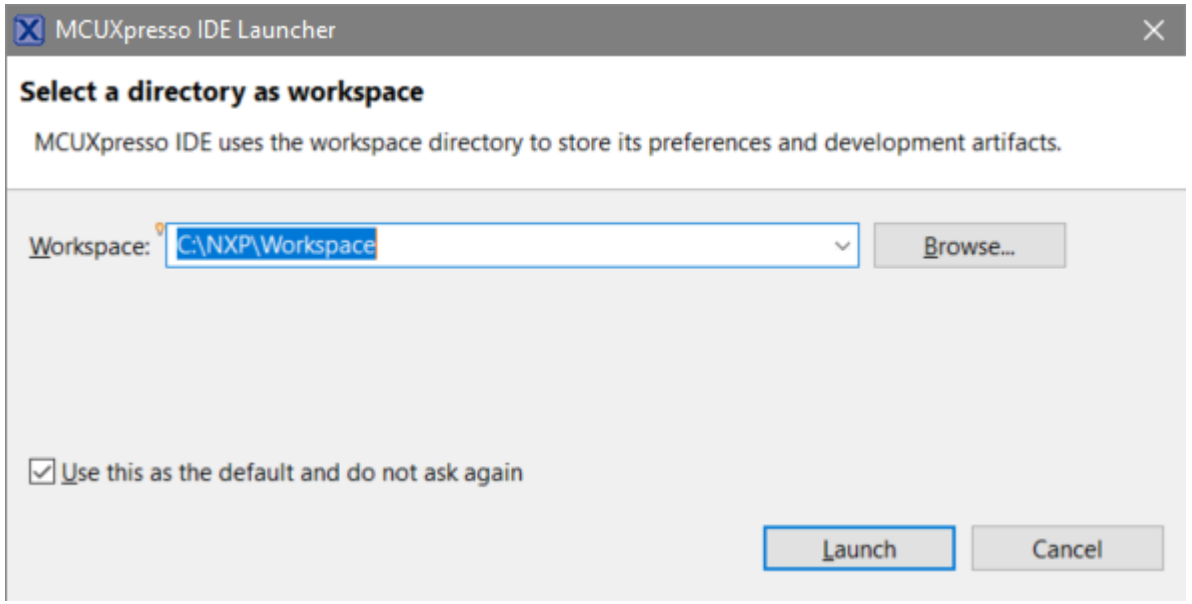
- MCUXpresso should be installed using the default settings
  - Will install to **C:\NXP\MCUxpressoIDE_11.3.0_5222**
  - Do not run until the Workspace folder has been created (next slide)



MCUXpresso – Workspace Folder

- Create a **C:\NXP\Workspace** folder
- Run MCUXpresso
  - When prompted use the **Browse** button to select the created Workspace folder
  - (Optional) Tick the checkbox to avoid future Workspace prompts





SDK

SDK Download from https://kex-stage.nxp.com/

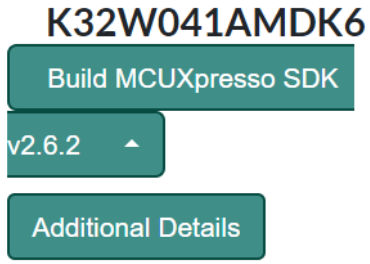
Step 1

Click *Select Development Board*

Step 2

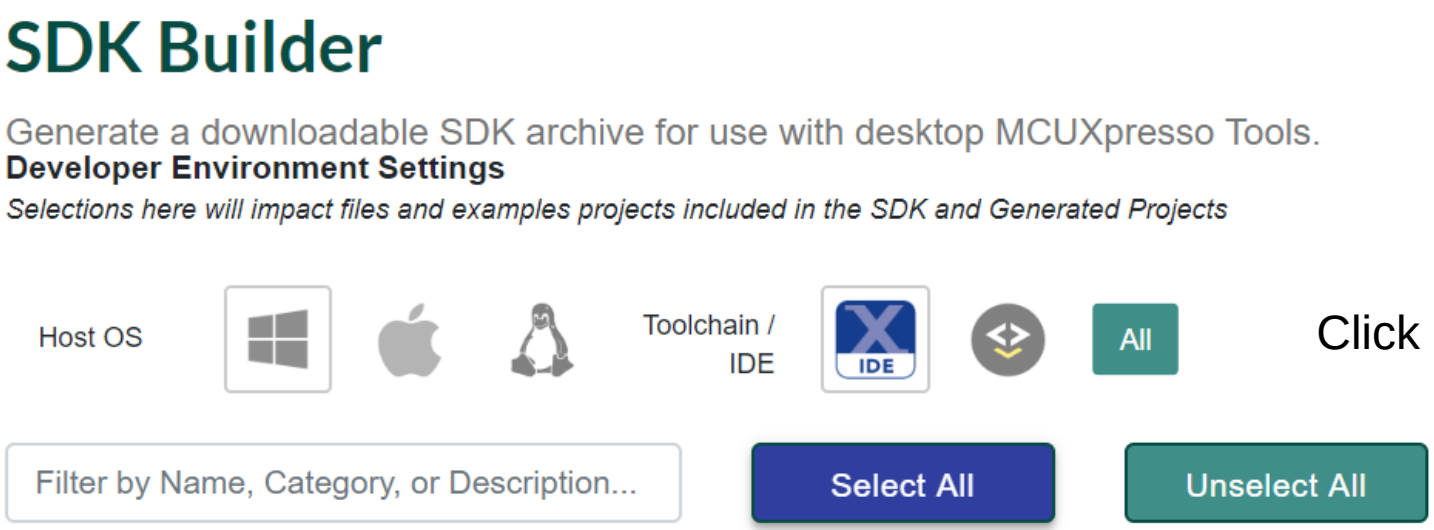
Choose *K32W041AMDK6*

Step 3



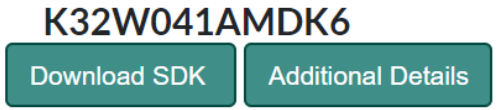
Choose the correct version *v2.6.3* then click *Build MCUXpresso SDK*

Step 4



Click *Select All*

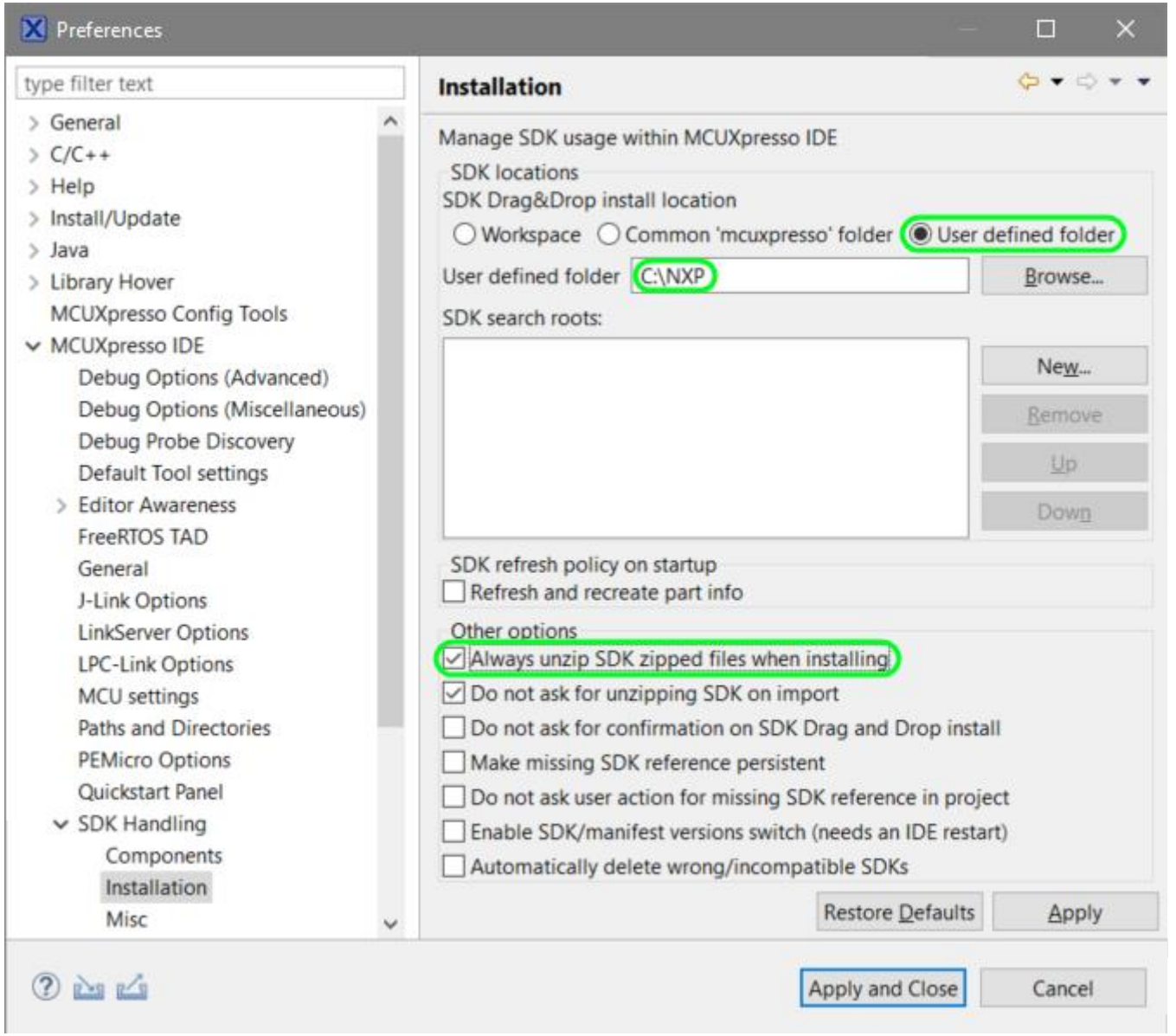
Step 5



Click *Download SDK to get SDK_2.6.3_K32W041AMDK6.zip*

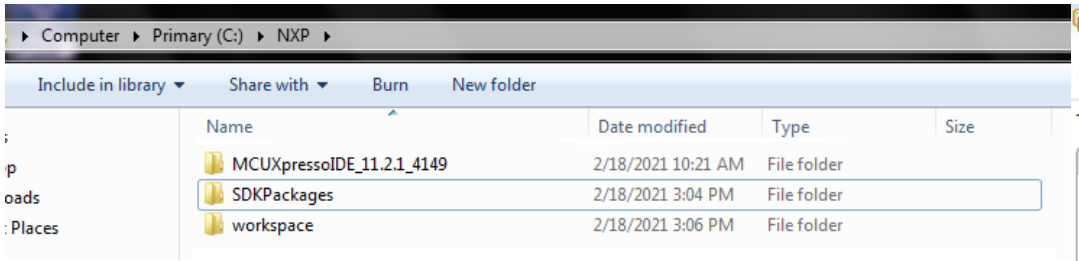
SDK – MCUXpresso SDK Installation Preferences

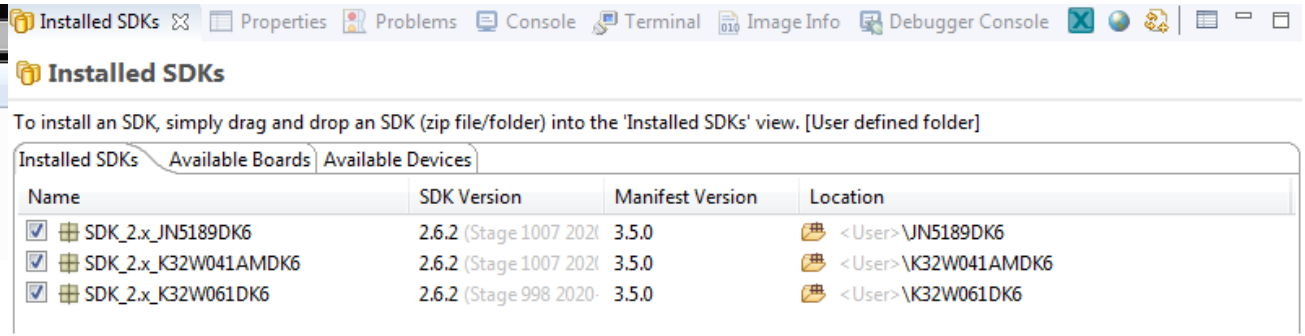
- In MCUXpresso select **Window > Preferences** from the menu bar
  - Expand and select MCUXpresso **IDE > SDK Handling > Installation**
    - Select the User defined folder radio button
    - Use the Browse button to select **C:\NXP**
    - Check **Always unzip SDK zipped files when installing**
    - Click **Apply and Close**



SDK – Installation

- Rename the SDK ZIP file to **K32W041AMDK6.zip**
  - The installation folder will be given the same name as the ZIP file
  - Makefiles in Zigbee application notes include the path to the SDK installation
- Select the Installed SDKs tab in the lower central pane of MCUXpresso
  - Drag and drop the SDK ZIP file into this pane
  - After clicking through the prompts the newly installed SDK will be shown in this pane

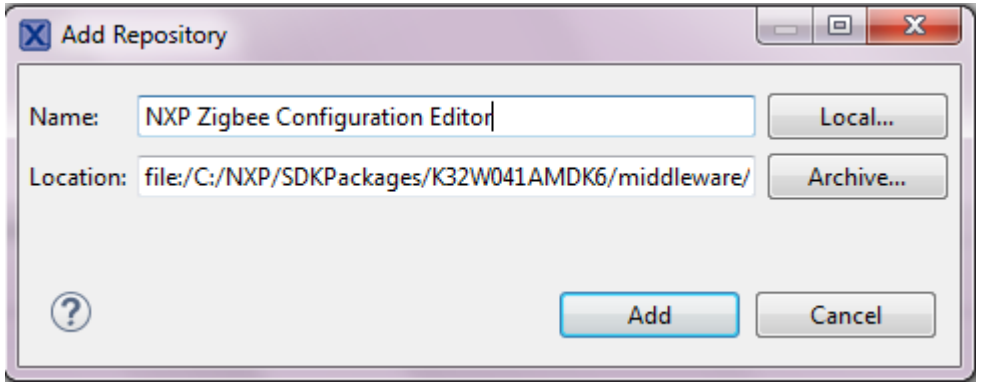




ZIGBEE 3.0 CONFIGURATION EDITOR

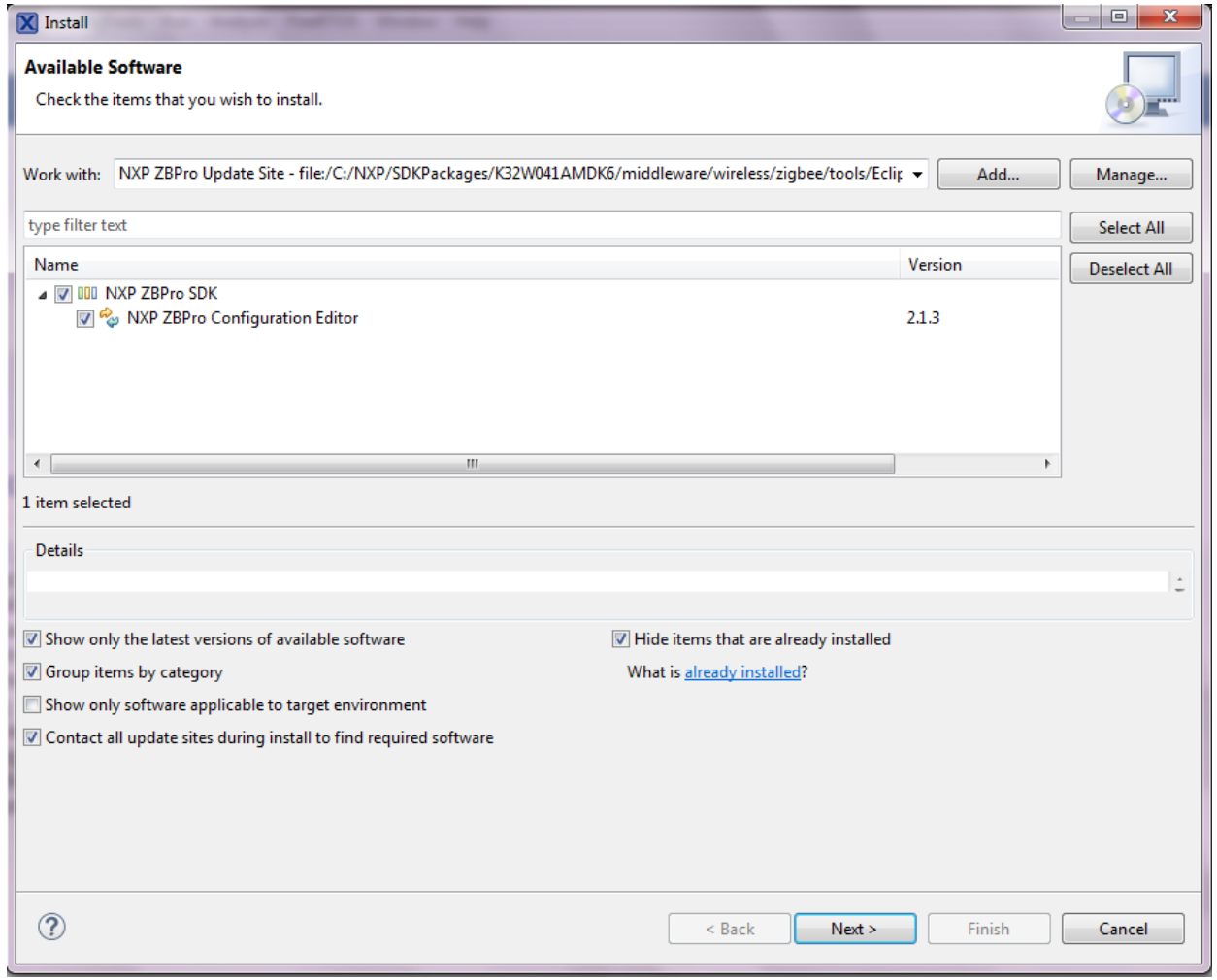
ZigBee 3.0 Configuration Editor – Installation (1)

- This plug-in provides a GUI to edit the ZigBee 3.0 Configuration settings
  - In MCUXPresso select **Help > Install New Software** from the menu bar
  - In the **Install** window that opens click the **Add** button in the upper right
  - In the **Add Repository** window that opens
    - Enter **NXP ZigBee Configuration Editor** in the **Name** edit box
    - Click the **Local** button and navigate to the folder below then click the **Select Folder** button: **C:\NXP\SDKPackages\<SDK>\middleware\wireless\zigbee\tools\Eclipse_plugins\com.nxp.sdk.update_site**
    - Click the **Add** button in the **Add Repository** window



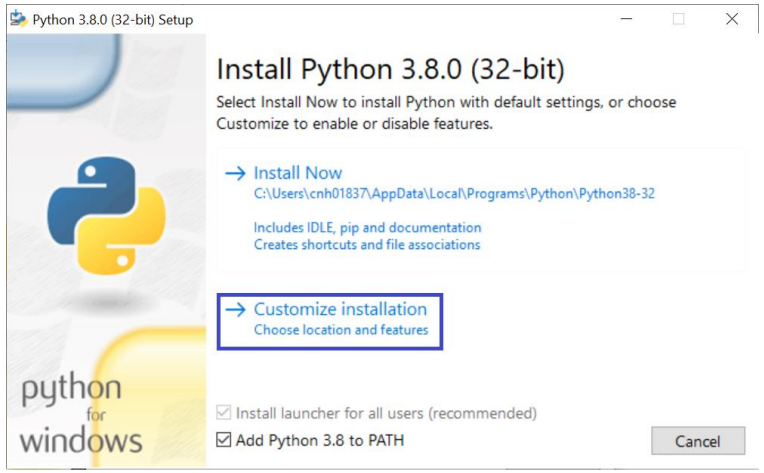
ZigBee 3.0 Configuration Editor – Installation (2)

- Back in the Install window:
  - Check the box next to NXP ZBPro SDK
  - Click the Next button to proceed with the installation
  - Click through the following windows
    - Accept the licence agreement when prompted
    - There may be a warning about unsigned content
    - MCUXpresso will need to be restarted to complete the plug-in installation
    - Latest version is 2.1.3



PYTHON 3.8

Python 3.x - Installation

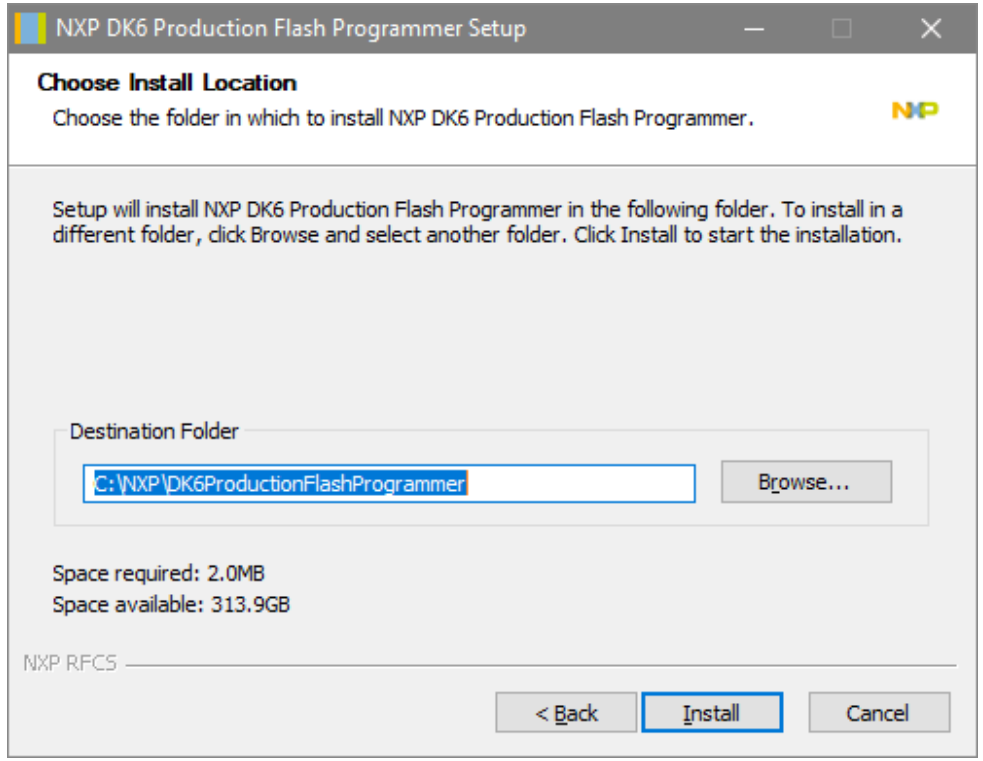


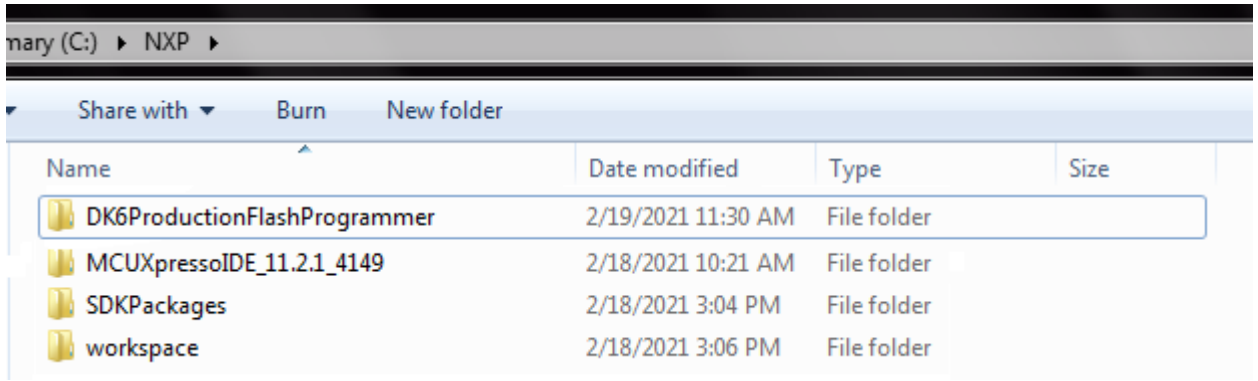
- Python 3.x is used to create an image signature during compilation
- Recommend the use of Python 3.8.0 from: https://www.python.org/downloads/release/python-380/
- Recommend to Select *customize installation*
- Select the option to add Python to the system's PATH environment variable during installation
- After installation a cryptography package must be installed using the command:
  - C:\Python38\Scripts>**pip install pycryptodome**

DK6 PROGRAMMER

DK6 Programmer - Installation

- The DK6 Production Flash Programmer [JN-SW-4407] is a command line programmer
  - Included in SDK zip *C:/NXP/SDKPackages/K32W041AMDK6/tools*
  - Install using the default options

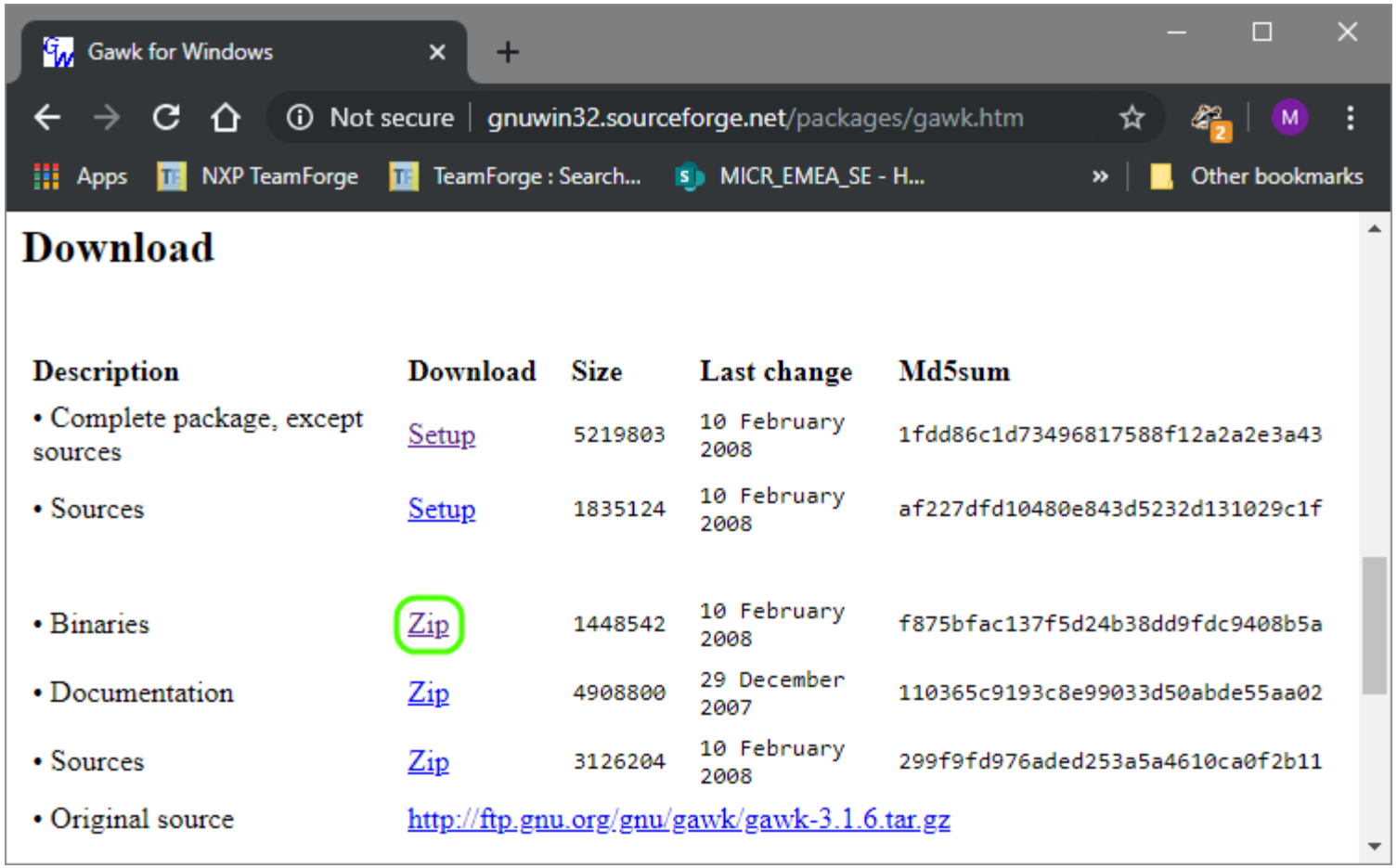


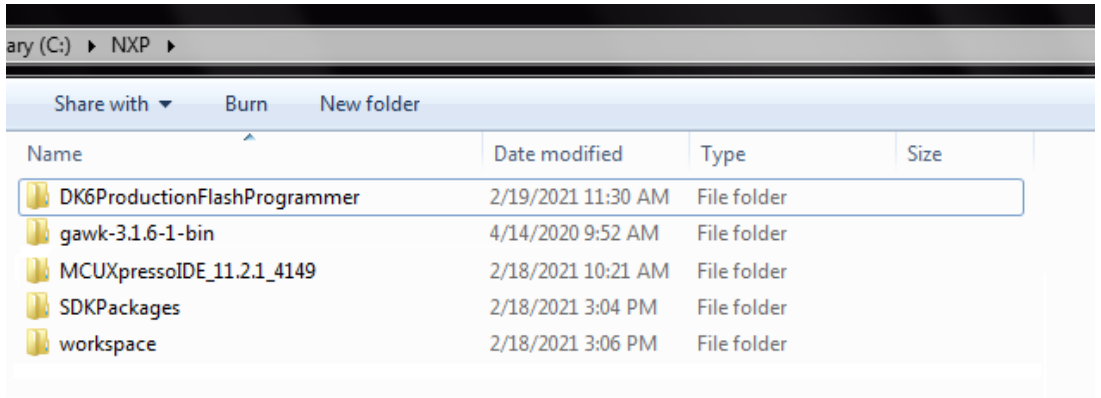


GAWK (OPTIONAL)

Gawk – Installation

- Used to create a detailed memory map during compilation
  - This compilation step is skipped if Gawk is not present
  - Download the binaries ZIP file from http://gnuwin32.sourceforge.net/packages/gawk.htm
  - Extract to **C:\NXP\gawk-3.1.6-1-bin** (retaining the ZIP file name as the folder name)





BEYOND STUDIO (OPTIONAL)

BeyondStudio – Installation

- Go to https://www.nxp.com/pages/jn516x-zigbee-home-automation:ZIGBEE-HOME-AUTOMATION

Home Automation Software

ZigBee PRO with the Home Automation profile is available for the JN5169 and JN5168 microcontrollers. The protocol, profile and associated support software are supplied in the following Software Developer's Kit (SDK):

- JN516x ZigBee Home Automation SDK

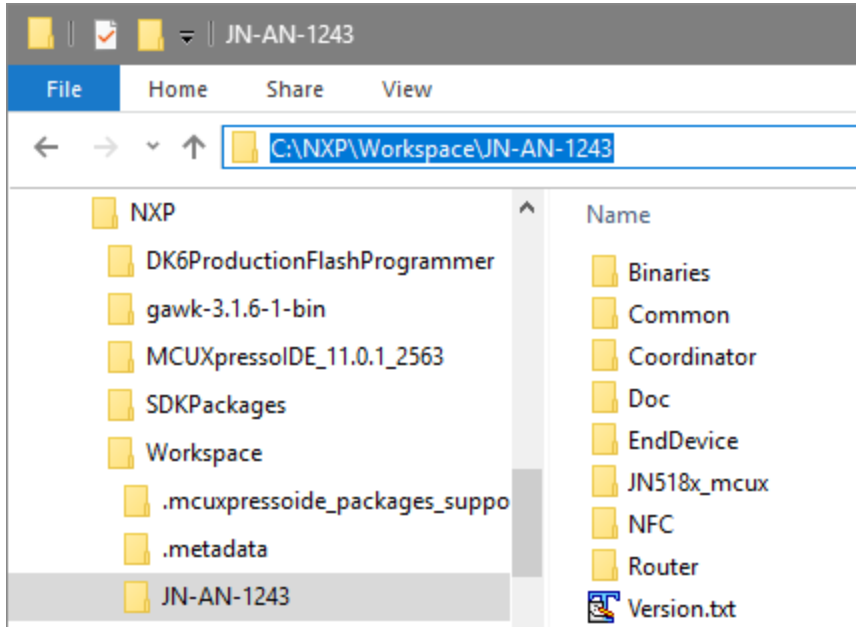
The above SDK must be installed on top of the BeyondStudio for NXP toolchain (JN-SW-4141). Installation instructions are provided in the BeyondStudio for NXP Installation and User Guide (JN-UG-3098).

ZigBee Home Automation demonstration applications are provided and described in Application Notes (see below). These demonstrations can be programmed into and run on the boards of the JN516x-EK001 or JN516x-EK004 Evaluation Kit.

APPLICATION NOTES

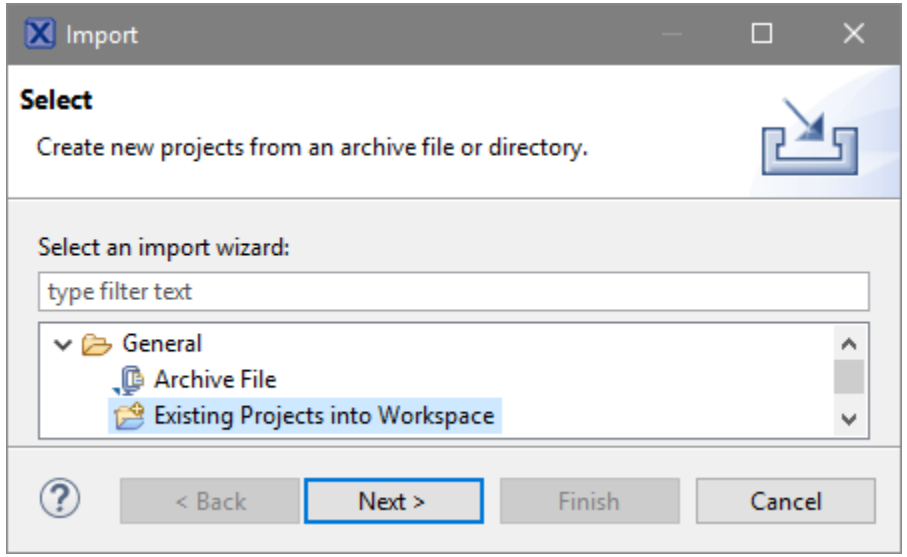
Application Notes - Installation

- Unzip the Application Note directly into the Workspace folder
  - The ZIP file includes a containing folder



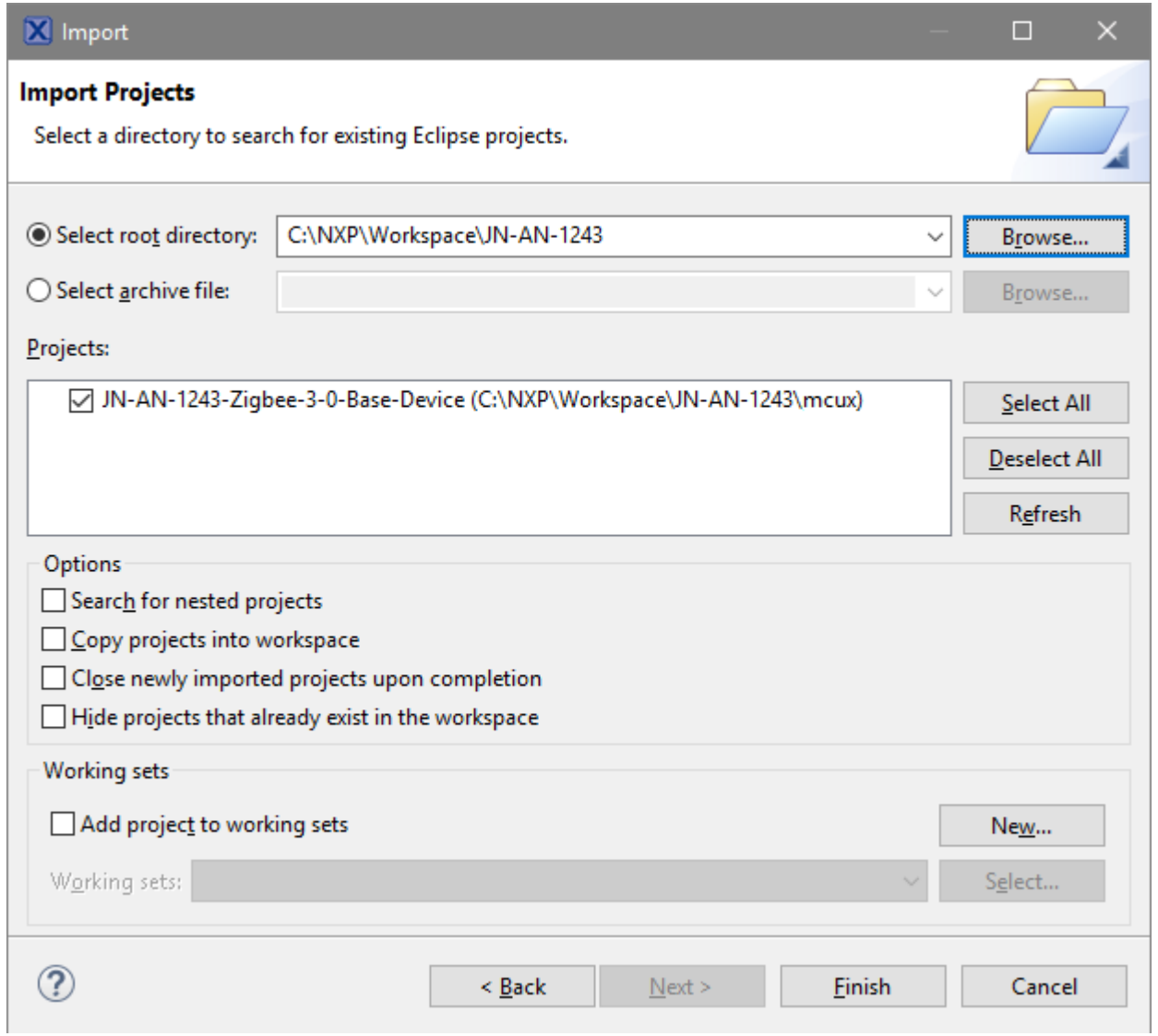
Application Notes – Import (1)

- In MCUXpresso select **File > Import** on the menu bar
  - Select **General > Existing Projects into Workspace** in the **Import** window



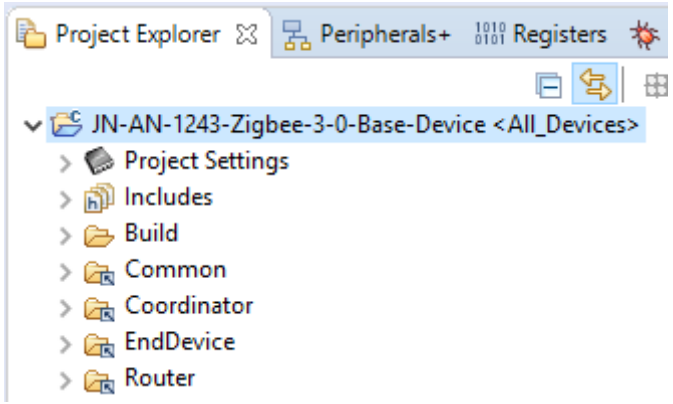
Application Notes – Import (2)

- In the second **Import** window:
  - Use the **Browse** button to select the path to the Application Note folder
  - Ensure the check box is ticked next to the Application Note project
  - Do not tick any of **Options** check boxes
    - The files are already in the correct locations
  - Click the **Finish** button to complete the import



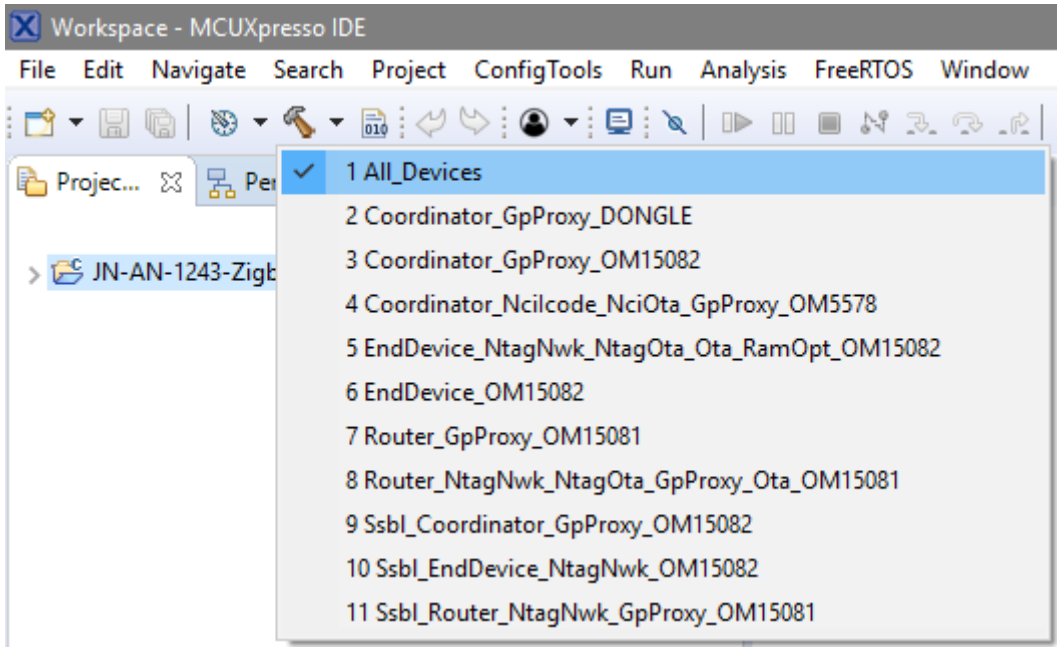
Application Notes – Import (3)

- The newly imported project is listed in the **Project Explorer** pane on the upper right of MCUXpresso



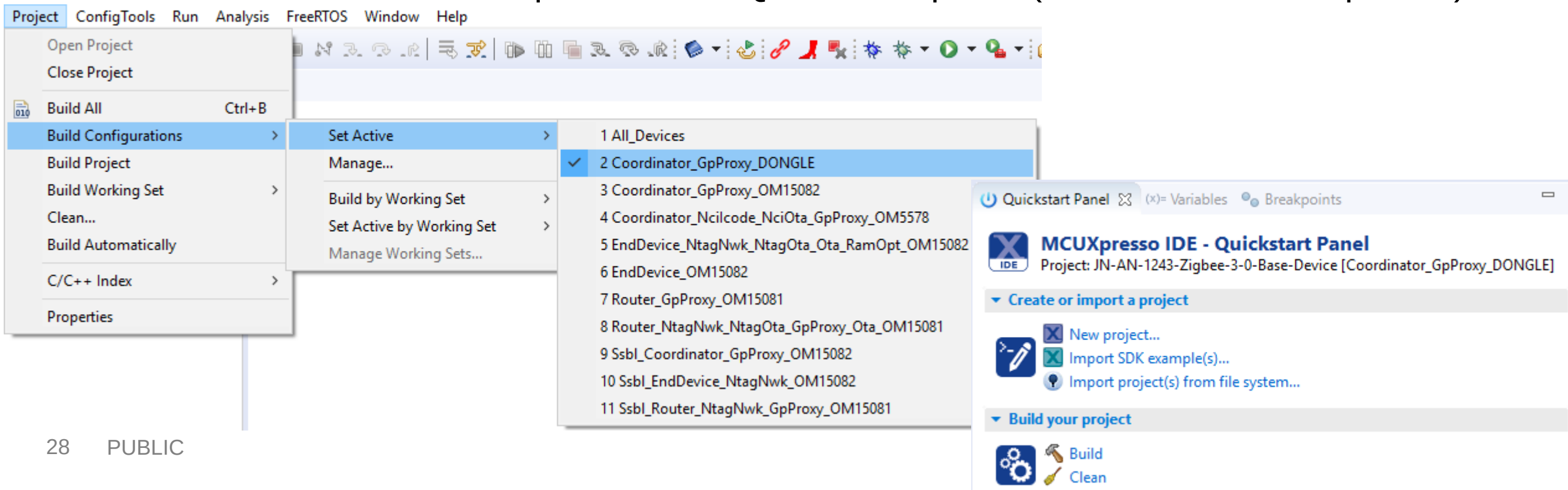
Application Notes – Compilation – All Devices

- To clean and build all devices in the Application Note
  - Click the build dropdown on the toolbar (the arrow next to the hammer icon)
  - Click the **All Devices** build configuration
  - This uses a makefile in the **mcux\Build** folder of the Application Note



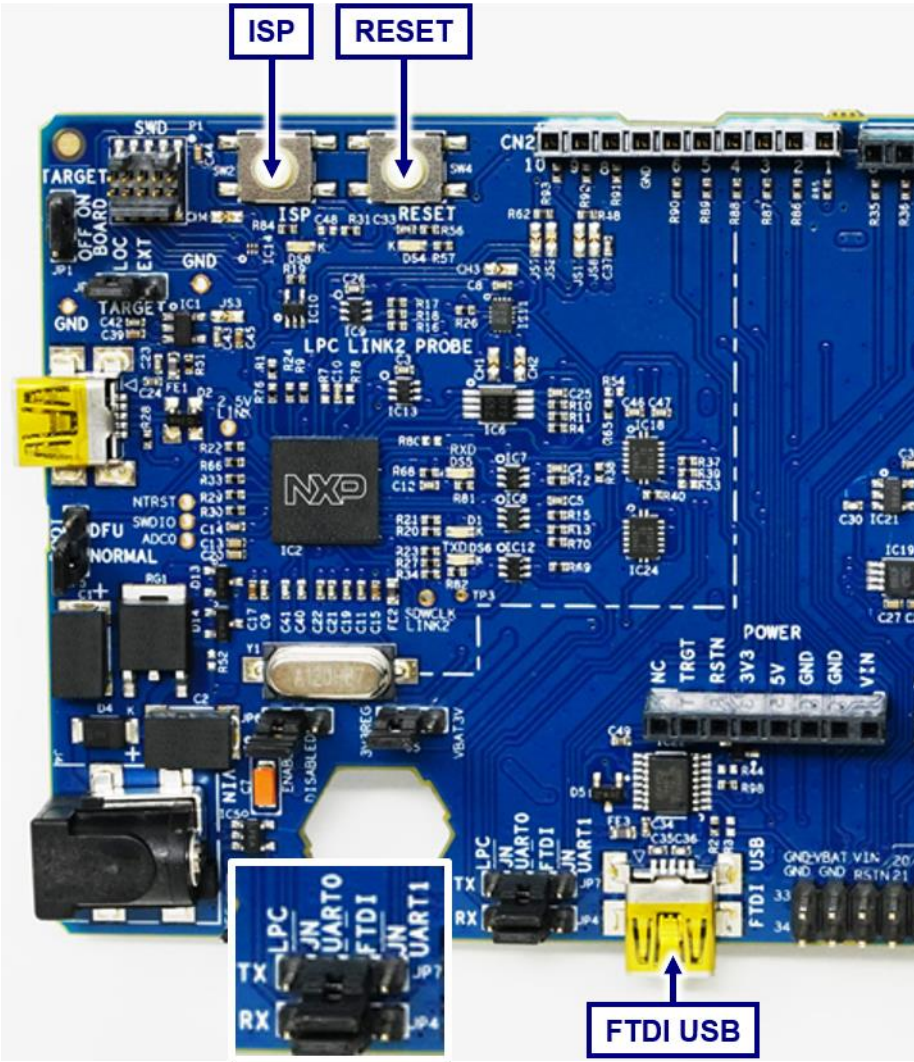
Application Notes – Compilation – Single Device

- To clean or build a single device in the Application Note
  - Select the project in the **Project Explorer**
  - Select **Project > Build Configuration > Set Active** then the required single **Build Configuration**
  - Use the **Build** and **Clean** options in the **Quickstart** panel (lower left of MCUXpresso)



Application Notes – Programming

- Compiled firmware is located in the **Binaries** folder of the Application Note
  - Windows batch (**.bat**) files that use the DK6 Programmer are located alongside the binary (**.bin**) files
- To program a binary file:
  - Connect a PC to the mini USB connector marked **FTDI USB**
  - Ensure the UART jumpers (to the left of **FTDI USB**) connect the centre two pins
  - Run the appropriate **.bat** file which will prompt for the COM port to use



SECURE CONNECTIONS
FOR A SMARTER WORLD