RN2483_Silica Documentation

Release 0

Silica

Sep 19, 2017

Contents

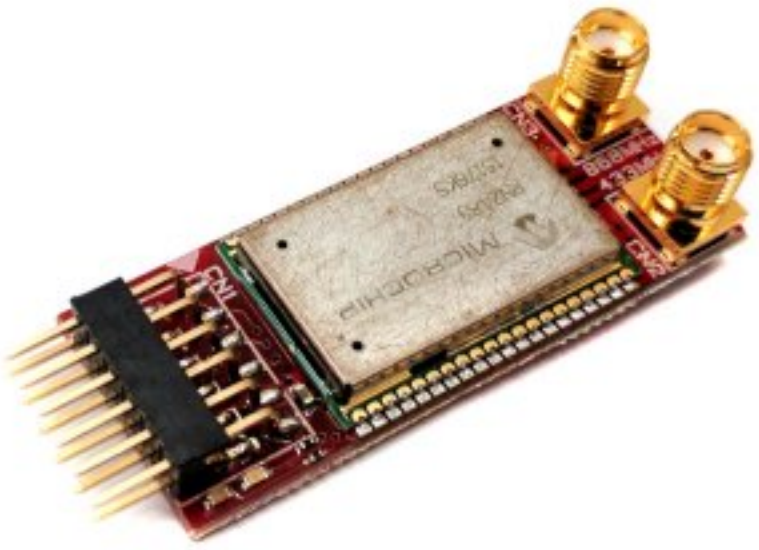
Version 1.00B

Date 6 Jul 2016

Reference name BAEPMOD-COM-003-A01

CHAPTER 1

Microchip’s Long Range Low Power End Node solution



PMOD-Lora

CHAPTER 2

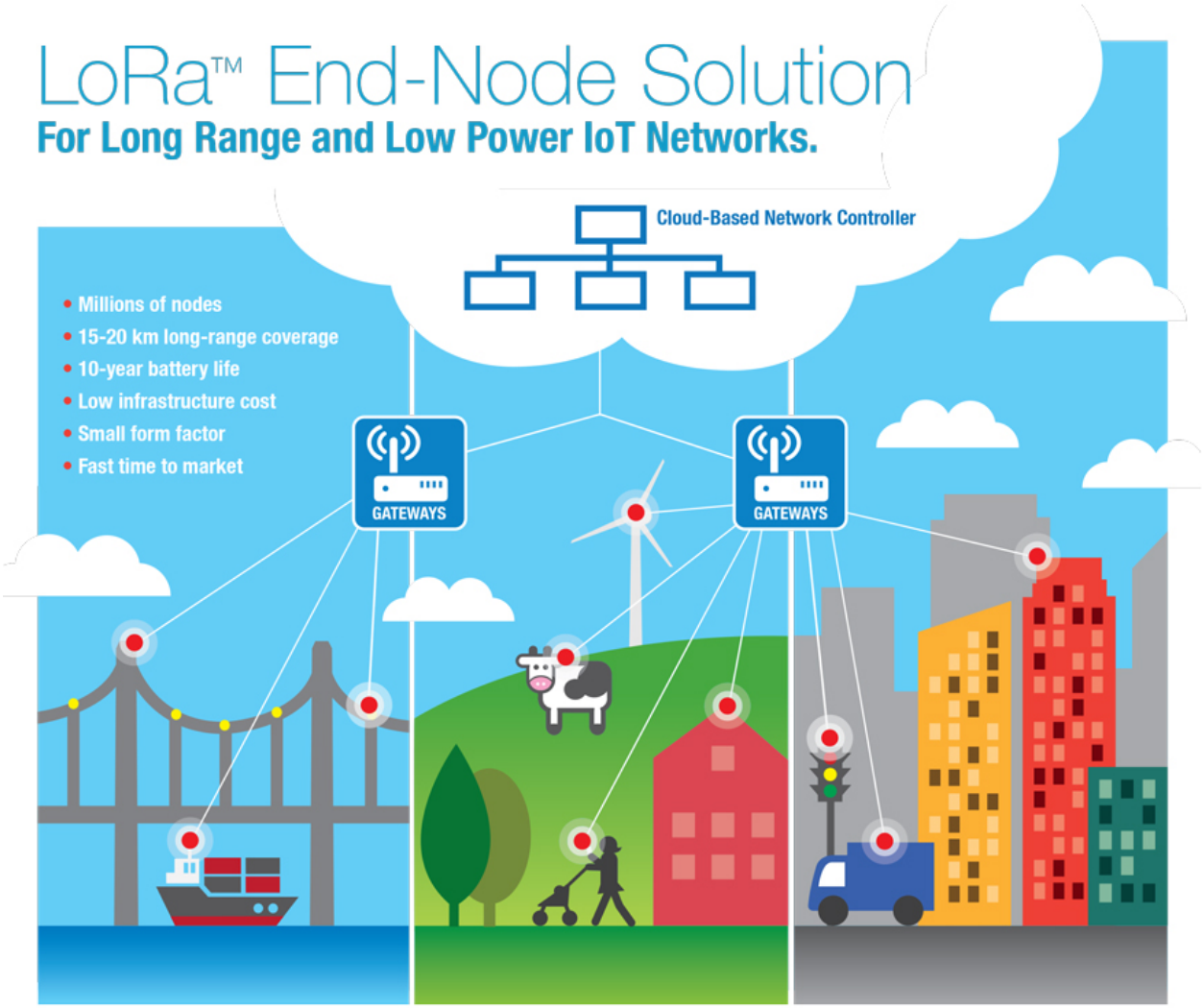
INTRODUCTION

Microchip's Long Range Low Power End Node solution

With the growing Internet of Things, Microchip has a LoRa® Technology wireless solution to address increasing demands on end-devices for long range connectivity, low power for battery operation, and low infrastructure cost for volume deployment.

Microchip's LoRa Technology solution is ready to run out-of-the box and with the complete LoRaWAN™ Protocol and certifications in place, it reduces time to market and saves development costs.

The RN2483 is a LoRa™-integrated modem with a range of more than 15 km (suburban), low power enabling a battery life greater than 10 years and the ability to connect millions of wireless sensor nodes to LoRa gateways and IoT-connected Cloud Servers. This robust system is due to LoRa's unique spread-spectrum based modulation that is capable of demodulation 20 dB below the noise level. This enables high sensitivity for ultra-long range, improved network efficiency and eliminates interference. The RN2483 modem operates over the 433 and 868 MHz license-free Industry Scientific and Medical (ISM) frequency bands and serves as the end-device in the LoRa network infrastructure.



The RN2483 is a fully-certified 433/868 MHz module based on wireless LoRa® technology.

The module's embedded LoRaWAN™ Class A protocol enables seamless connectivity to any LoRaWAN compliant network infrastructure, whether public or privately deployed. The module is specifically designed for ease of use, which shortens development time and speeds time to market. LoRa technology is ideal for battery-operated sensors and low power applications such as IoT, M2M, Smart City, Sensor networks, Industrial automation, and more.

Features:

- On-board LoRaWAN™ Class A protocol stack
- ASCII command interface over UART
- Compact form factor 17.8 x 26.7 x 3 mm
- Castellated SMT pads for easy and reliable PCB mounting
- Device Firmware Upgrade (DFU) over UART
- 14 GPIO for control, status, and ADC
- Highly integrated module with MCU, crystal, EUI-64 Node Identity Serial EEPROM, Radio transceiver with analog front end, and matching circuitry

- Environmentally friendly, RoHS compliant
- European R&TTE Directive Assessed Radio Module

Developement tools

Firmware developed using: NXP Codewarrior. For installation and configuration of the project, follow instruction inside *Quick start guide*

Document references

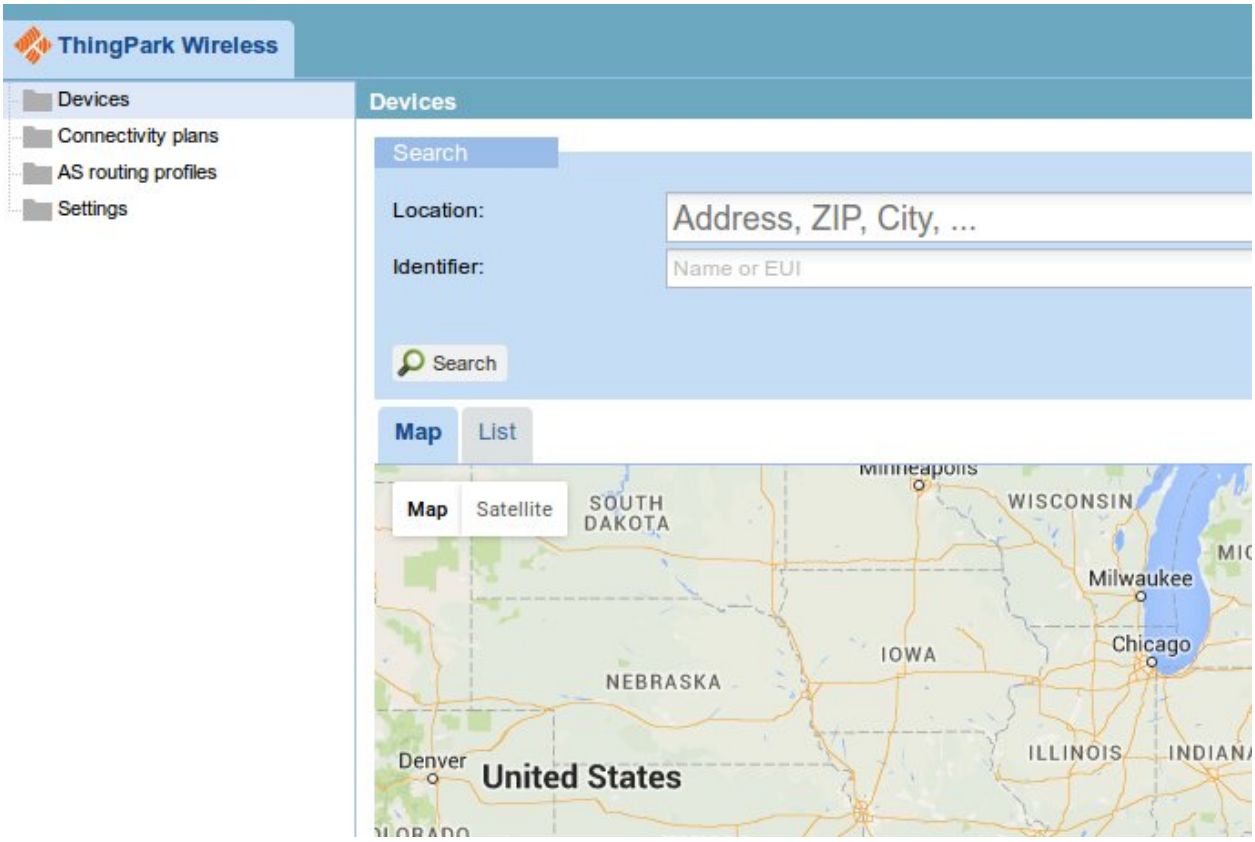
The board reference documentation is available on the architech-board website.

Contents:

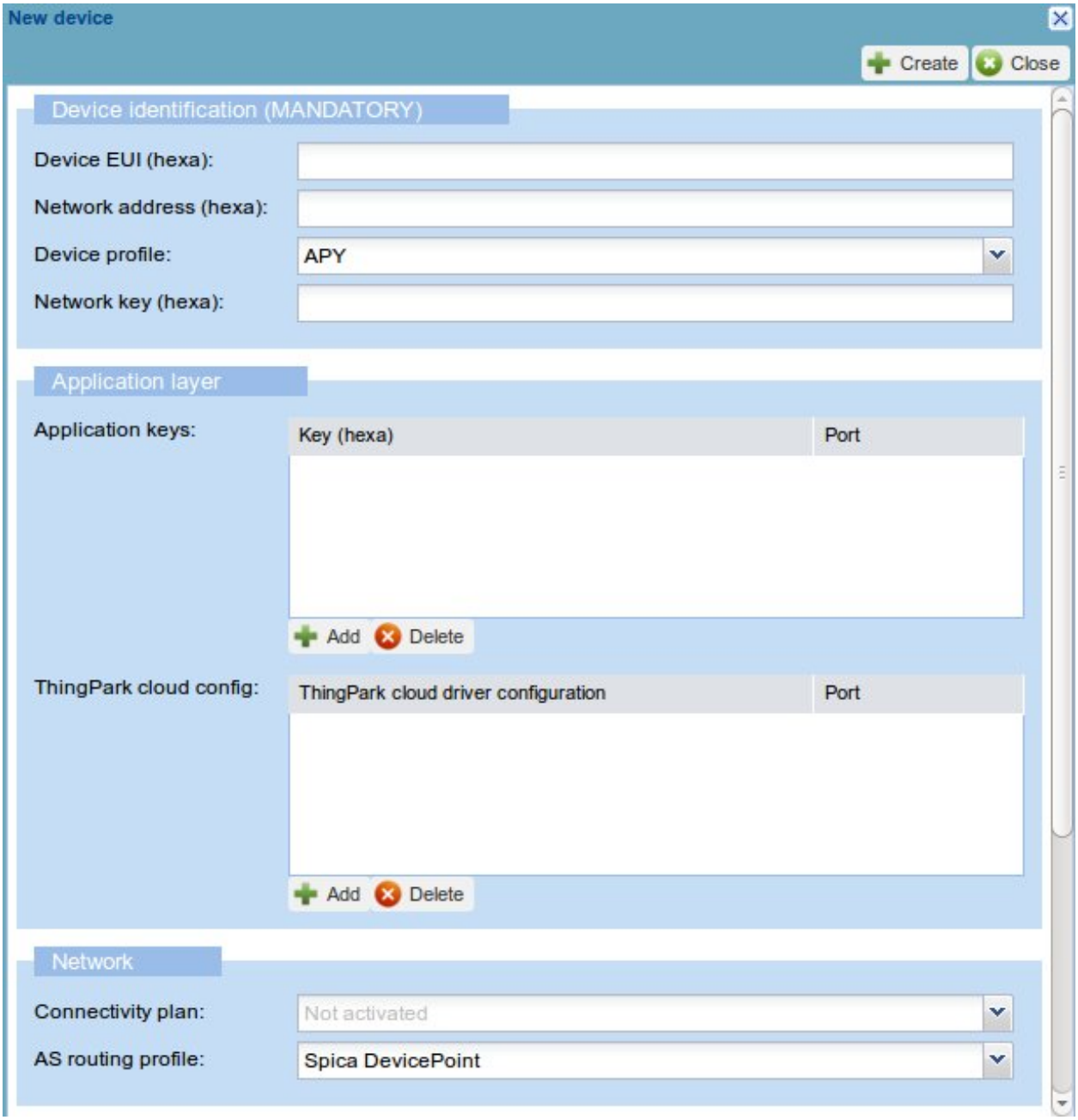
Quick start guide

In order to see working the PMOD-Lora you have to register the device on the activity website using your personal account. After the registration login and you will access the main page.

First up we have to register your Microchip RN2483 device, to do this, click on **Device Manager** arrow to open a new window.



Click the right mouse button on the **Devices** folder and select **+ Create**.



In this form you have to insert the data we have stored in the RN2483. The following fields are mandatory:

- **Device EUI**: the 16-hex identification key of the device. It is unique for every device, you can read it on the label of the board as in the image. For example the code can be **0004A30B001B9954**

- **Network address**: last 4 less significant bytes from the Device EUI key, for example if the code is **0004A30B001B9954** then the network address will be **001B9954**
- **Device profile**: **LoRaWAN 1.0 class A**
- **Network key (hexa)**: we have already saved it in the RN2834 device, it is the **Device EUI** key copied *twice*, for example if the device EUI is **0004A30B001B9954** then the **network key** you have to insert will be **0004A30B001B99540004A30B001B9954**.
- **Application keys**: Insert **AFBECD5647382910019283746 5FAEBDC**, port **4**. We have saved it in the RN2834 device. Press "Update" button to confirm.
- **Connectivity plan**: choose yours

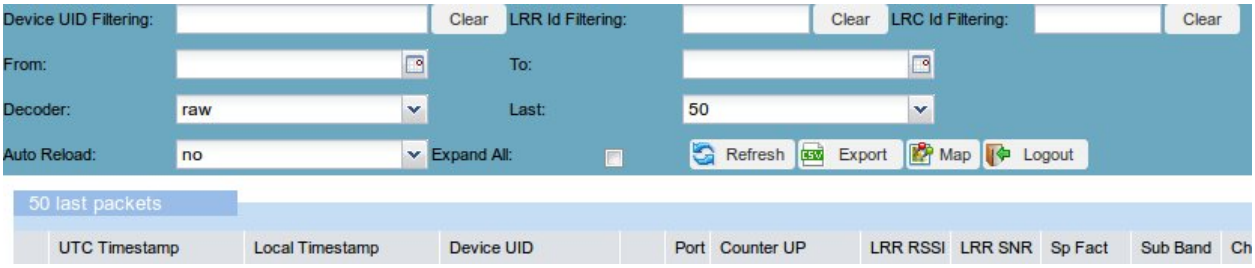
Optional field but useful:

- **Name**: insert a name just to recognize your device.

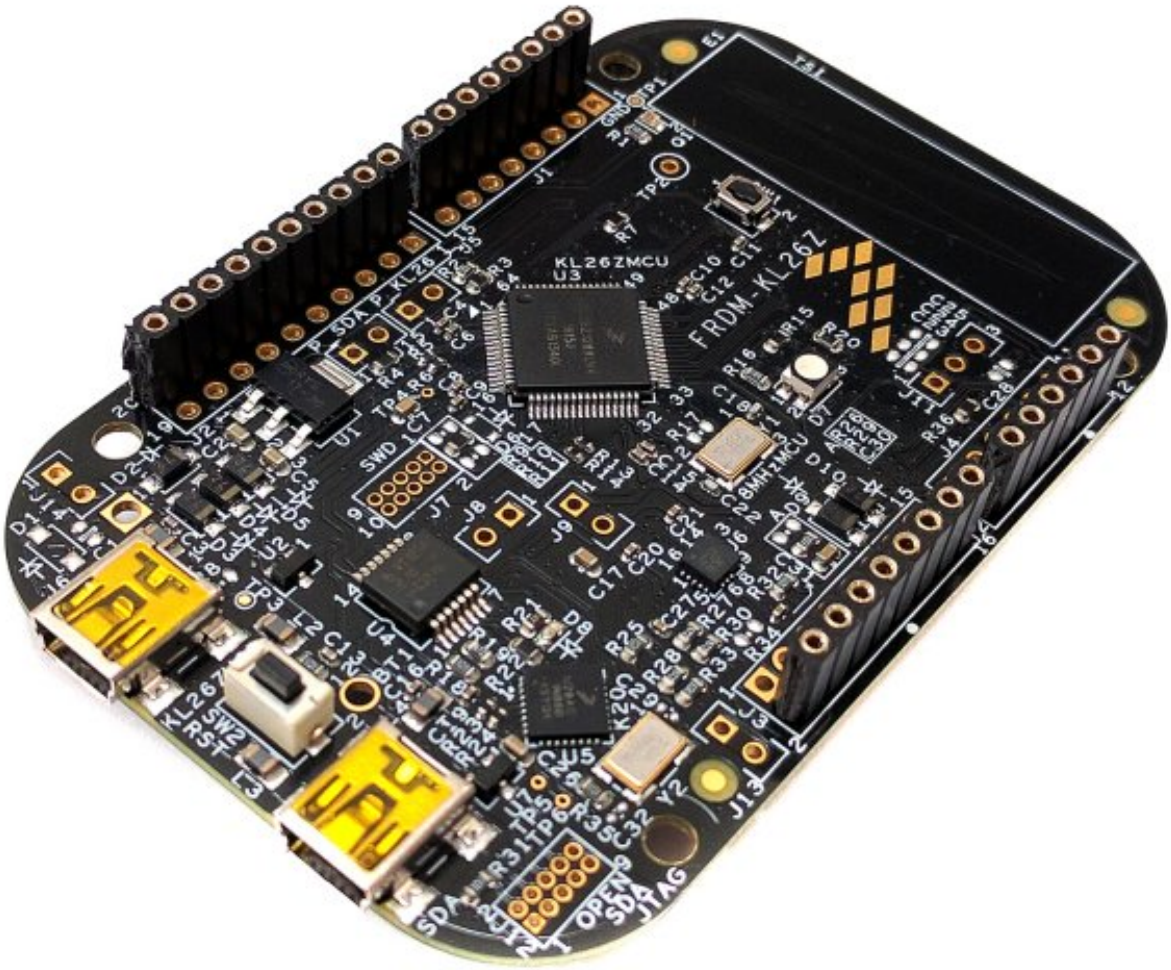
Then click on the top right **+ create**.

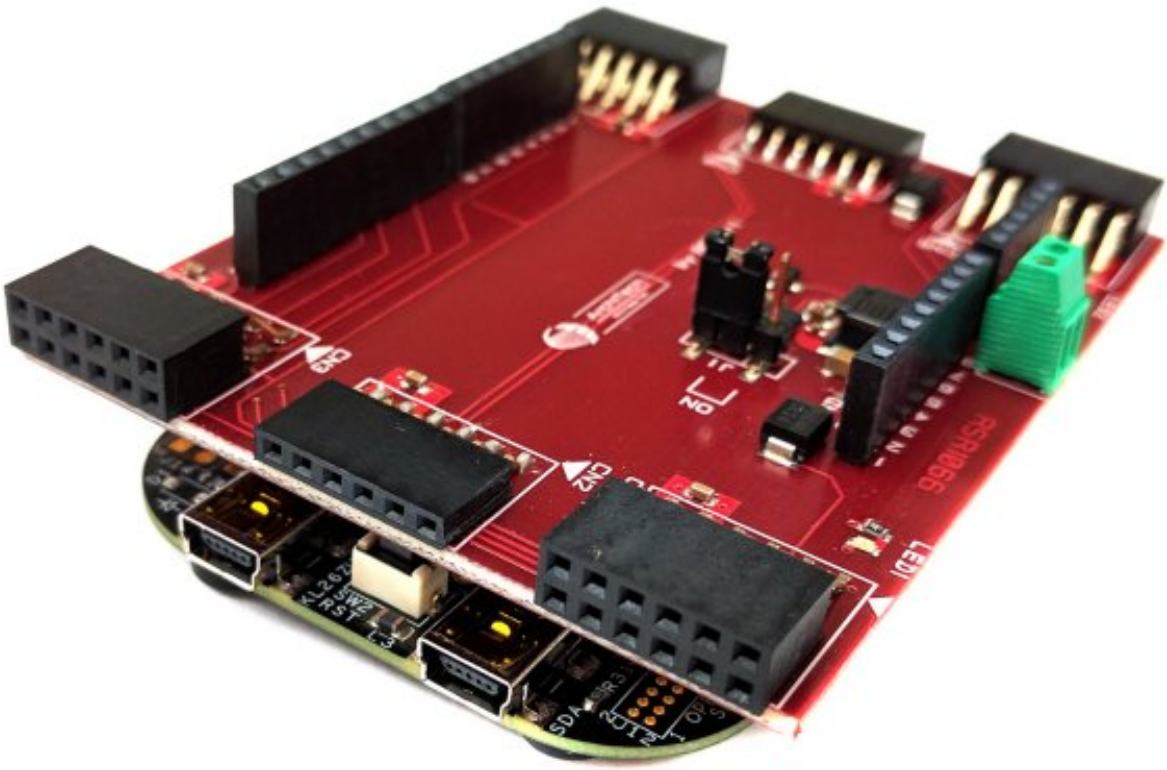
> **Warning:** If some key is wrong then the device will be unable to send data correctly in the Lora network.

After the registration, you can close the window device manager and on the main page go to **Logger**.



Here you will see all the messages sent by your device. Now take the FRDM-KL26Z board and connect it to the RSR1066 board. It is required mount the strip connectors:


KL26ZMCU
U3
FRDM-KL26Z
SWD
OPEN SDA JTAG
SW2
RST

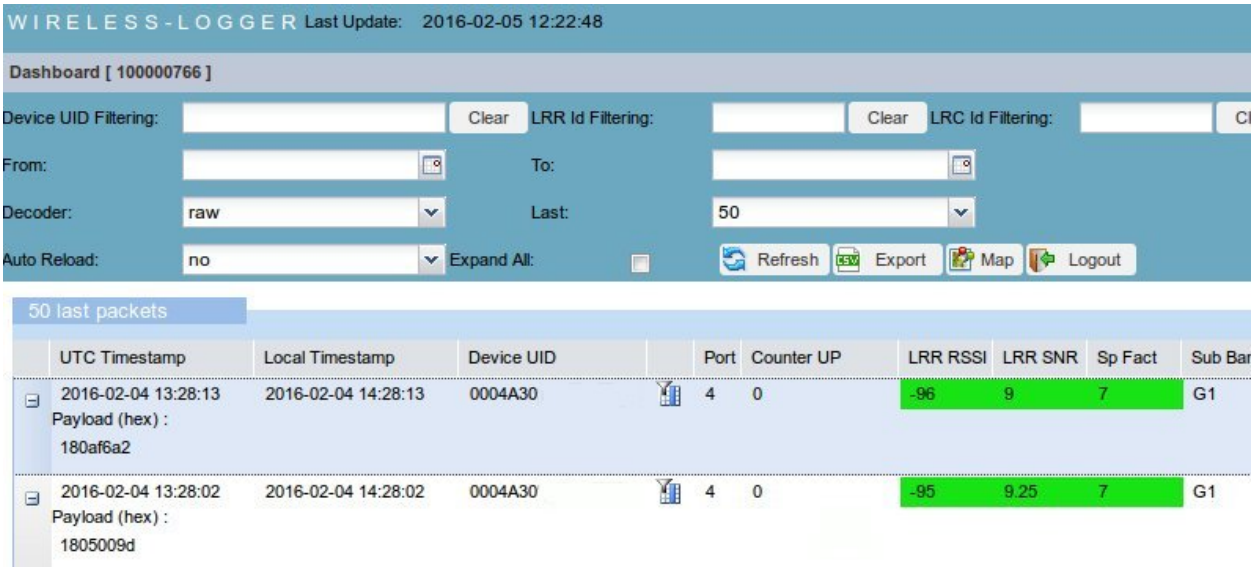


Connect the PMOD module to the CN7


by RGR-er
LED2
CN2
CN1
LED1

Then power supply the FRDM board via mini USB connection.

The board will send messages every 30 seconds. Now it's time to see the data sent. Power off the board. And in the logger window, you will have 2 rows, every row is a message received from the server. If you click on the **+** node you can see the unencrypted data received **Payload (hex):**. All messages start with the number 18, the other three number couples are the data read from the accelerometer mounted on the board. You will see the data changed when you have tilted the board and sent the second message.



Developing guide

This guide will provide instructions to install the development environment needed to compile and debug the demo firmware of the PMOD-Lora. The development system is built for Windows. The main steps are:

- Install Codewarrior Special Edition Software
- Import build & debug the source project

Hardware required:

- PMOD-Lora
- RSR1066 board
- FRDM-KL26Z board by Freescale
- Mini-USB cable
- PC with Windows

Install Codewarriror

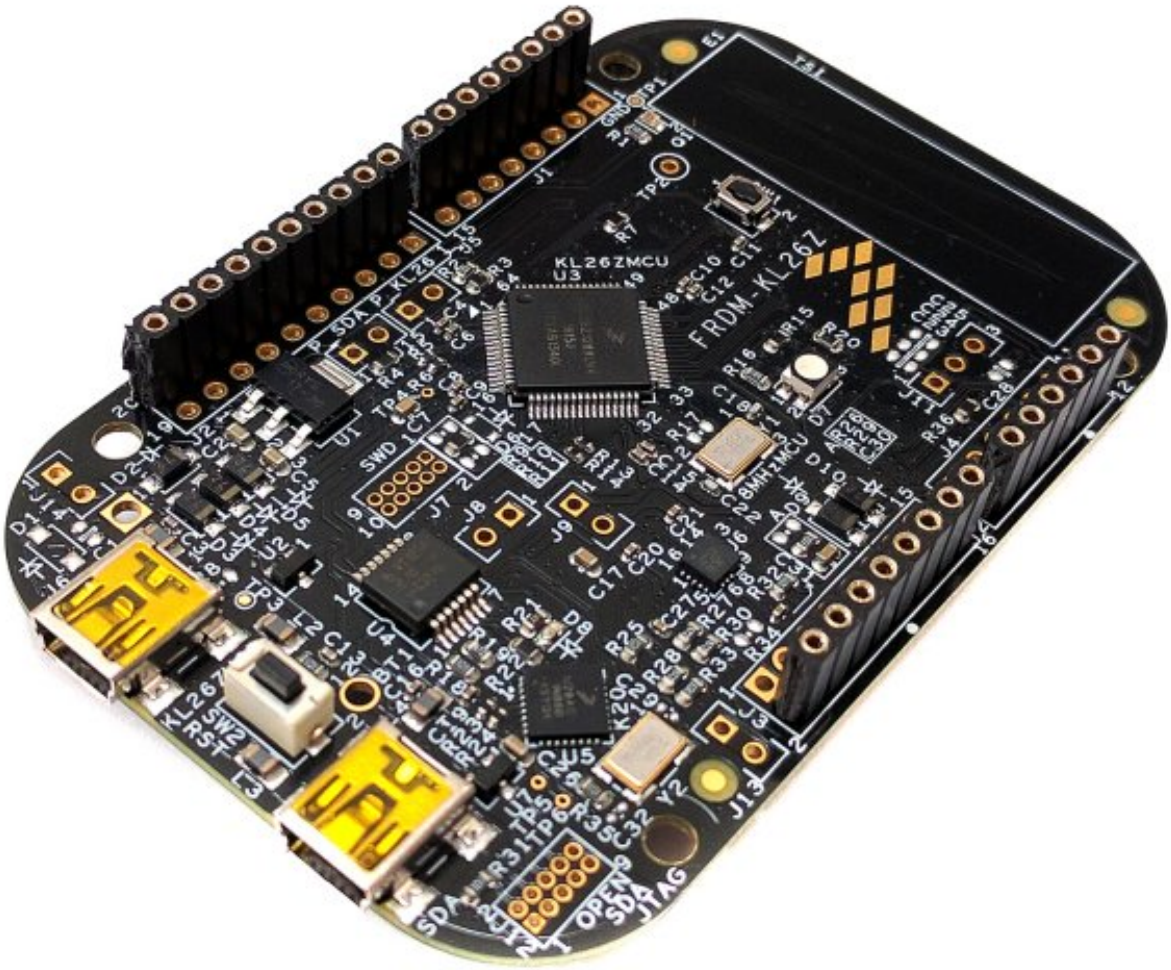
Special Editions are fully functional free download versions of the CodeWarrior Development Studio with code size restrictions on the build chain. Special Editions are pre-licensed, not bound to a single machine and are not time restricted. You do not need to register the tools or ask for a license.

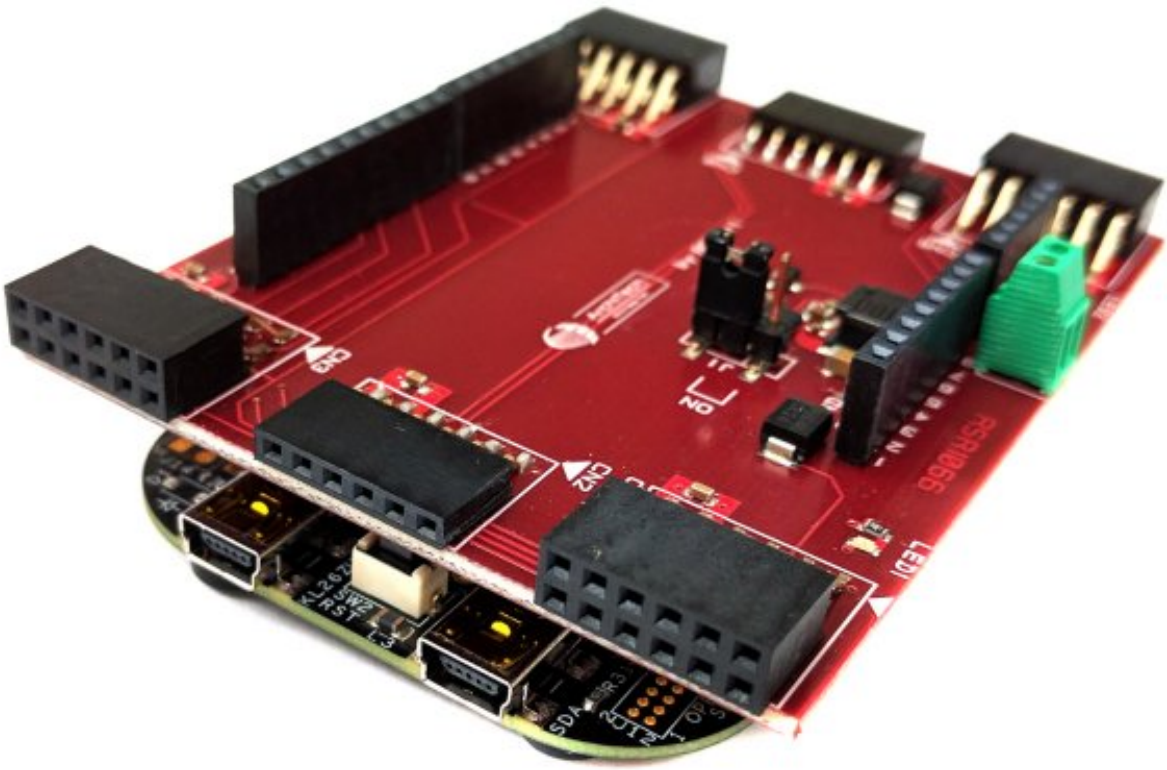
Download the IDE from this page, we used Codewarriror for Microcontrollers **v10.6.4**.

Next, launch the downloaded file **CW_MCU_v10.6.4_Special_Edition.exe** following all the default options and selecting **Kinetis** as platform. Once it is installed will be created its icon on the desktop.

Prepare the Hardware

Here you will see all the messages sent by your device. Now take the FRDM-KL26Z board and connect it to the RSR1066 board. It is required mount the strip connectors:





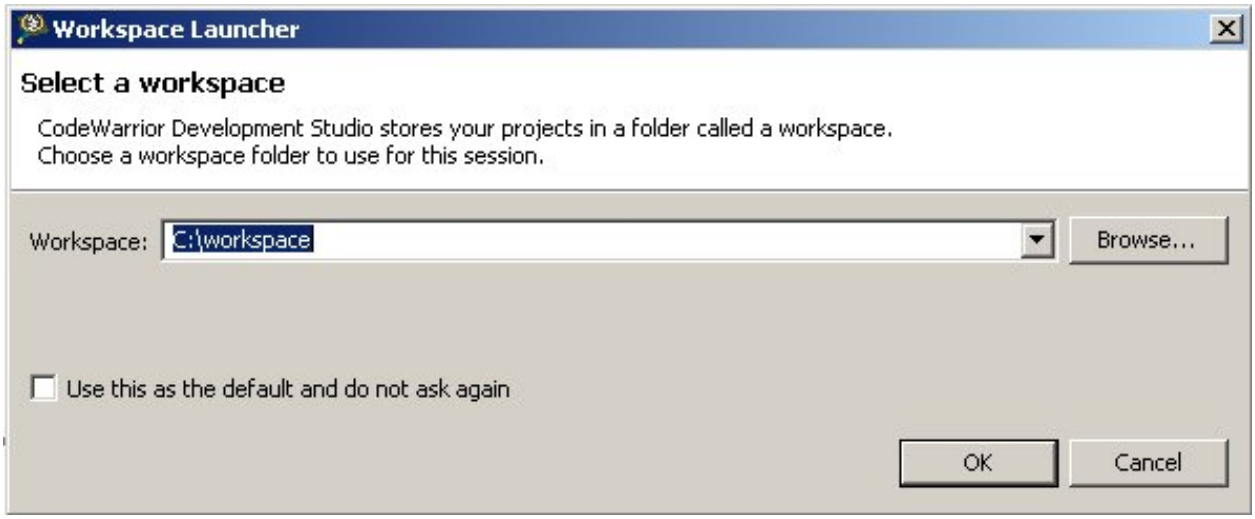
Connect the PMOD module to the CN7 then power supply the FRDM board via mini USB connection.

Configure the FRDM-KL26Z with OpenSDA interface

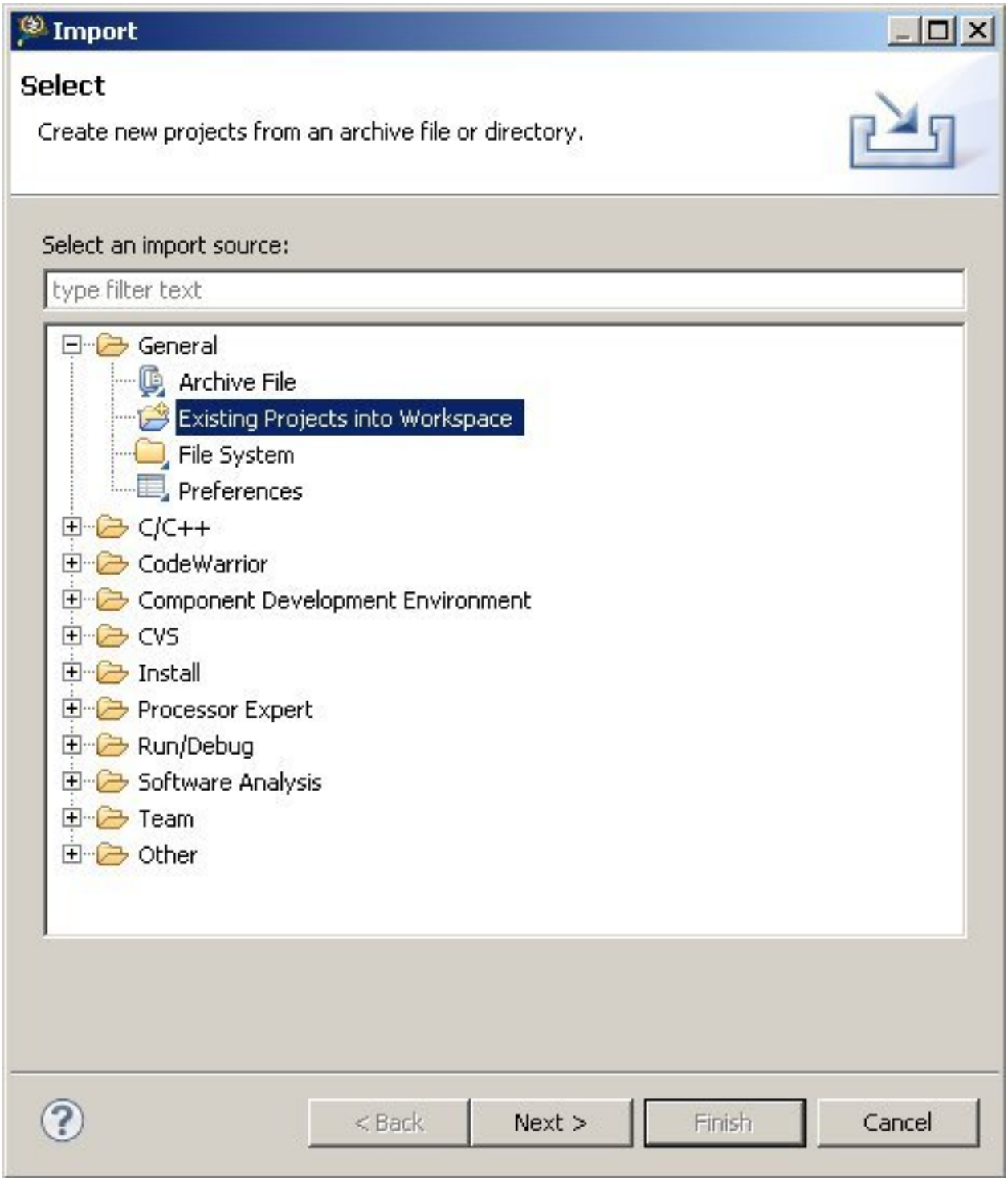
1. In order to install the lastest firmware go to the webpage OpenSDA Support.
2. Download and install **Windows USB Drivers Download PEDrivers_install.exe** from pemicro website. It is required to register in the website.
3. Then download the lastest Firmware Apps (.zip file).
4. Finally connect the FRDM-KL26Z board to the PC via mini-USB connector **OpenSDA**, remove the 1066 board and set the board in Bootloader mode (hold the Reset button down while connecting to USB, then release it). Your board will then be visible as a drive labelled **BOOTLOADER**. From the **Firmware Apps** zip copy into the BOOTLOADER disk the file **MSD-DEBUG-FRDM-KL26Z_Pemicro_vXXX.SDA** (where XX is the lastest version). Now unplag the USB cable.

Import Project

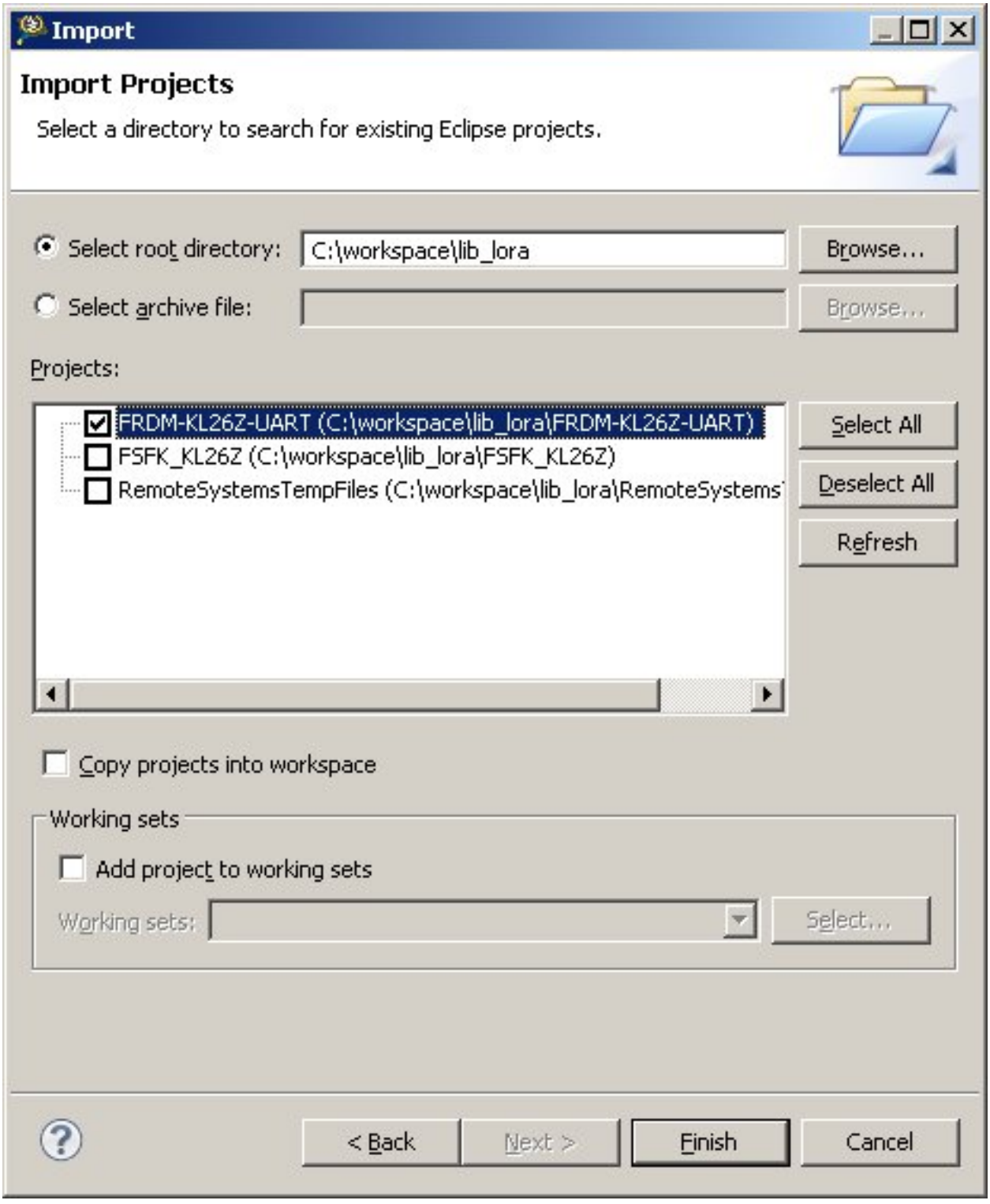
1. Create a folder named “workspace”
2. Download the project file form architechboards website and unzip it into the new folder.
3. Launch Codewarrior and select a folder for the workspace. Our project will be imported in this directory. In this guide we used this path:



4. Go to **File -> Import...**
5. Select **General -> Existing Projects into Workspace** and click on **Next >** button.



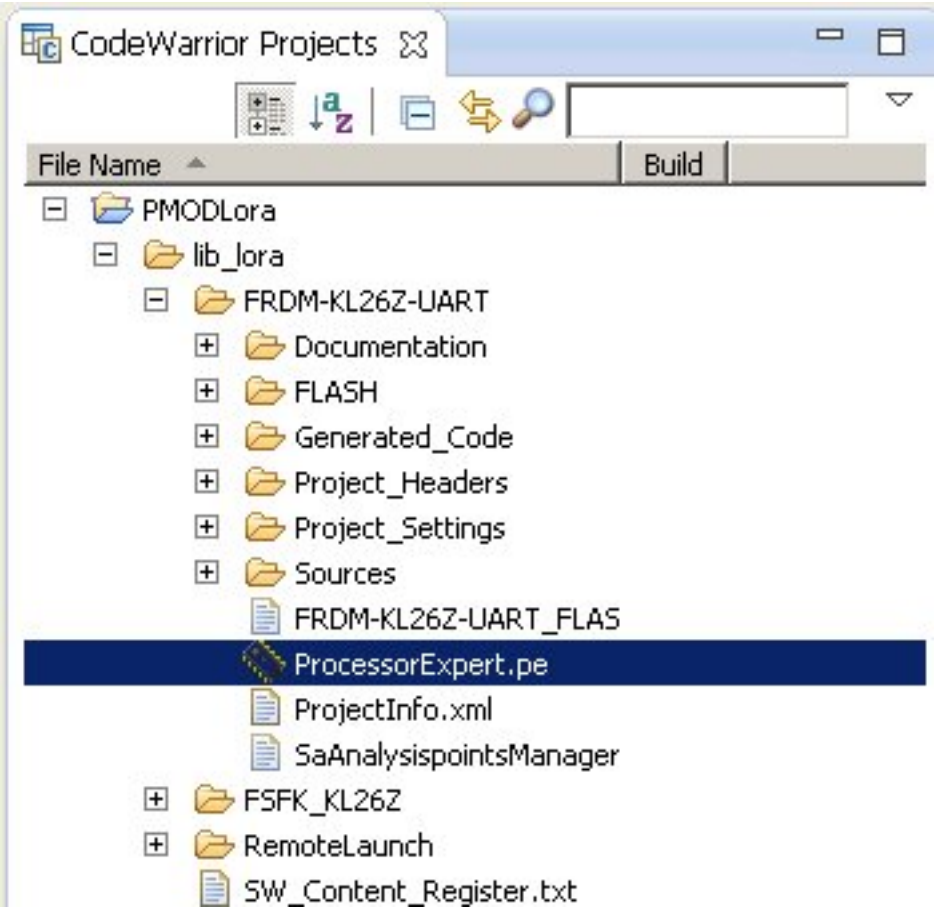
6. Select the folder where is locate the project **lib_lora** and select **FRDM-KL26Z-UART**. Then click on **Finish**.



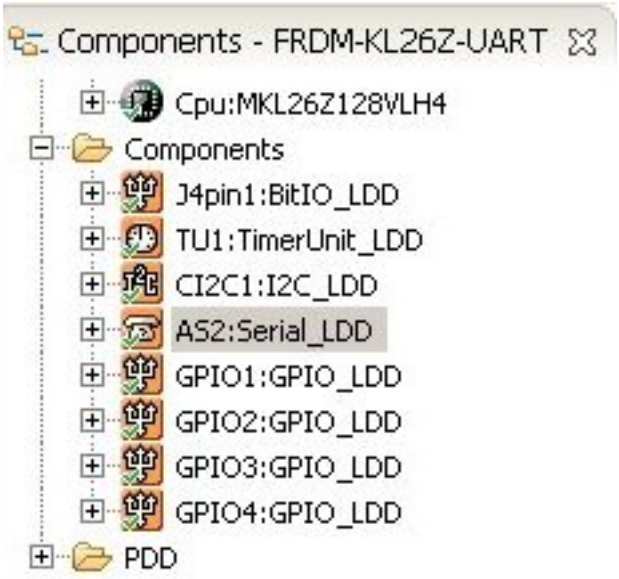
Build & Debug

If you want download the firmware in the board without debugging it go to step 5. In order to debug the code you have to change the UART port because **PTA1** and **PTA2** are used for debug purpose from the OpenSDA.

1. Now you have to open **Process Expert Window** double clicking on **ProcessorExpert.pe**



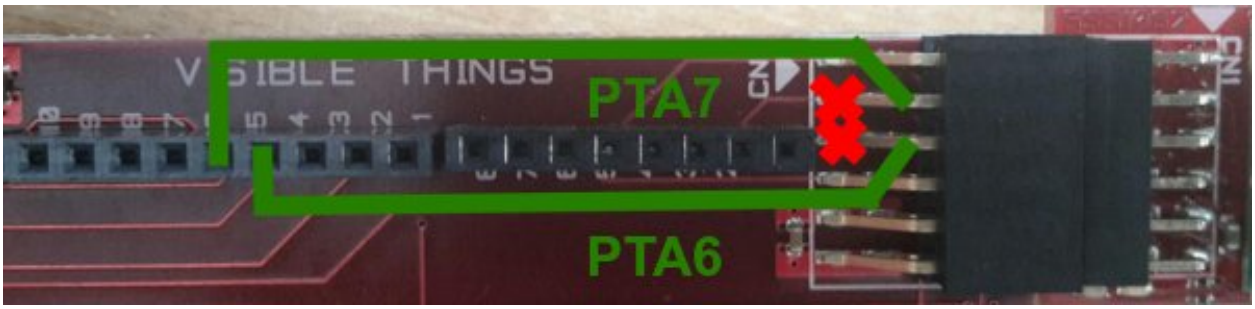
2. In **Components - FRDM-KL26Z-UART** tab select **AS2:Serial_LDD** node



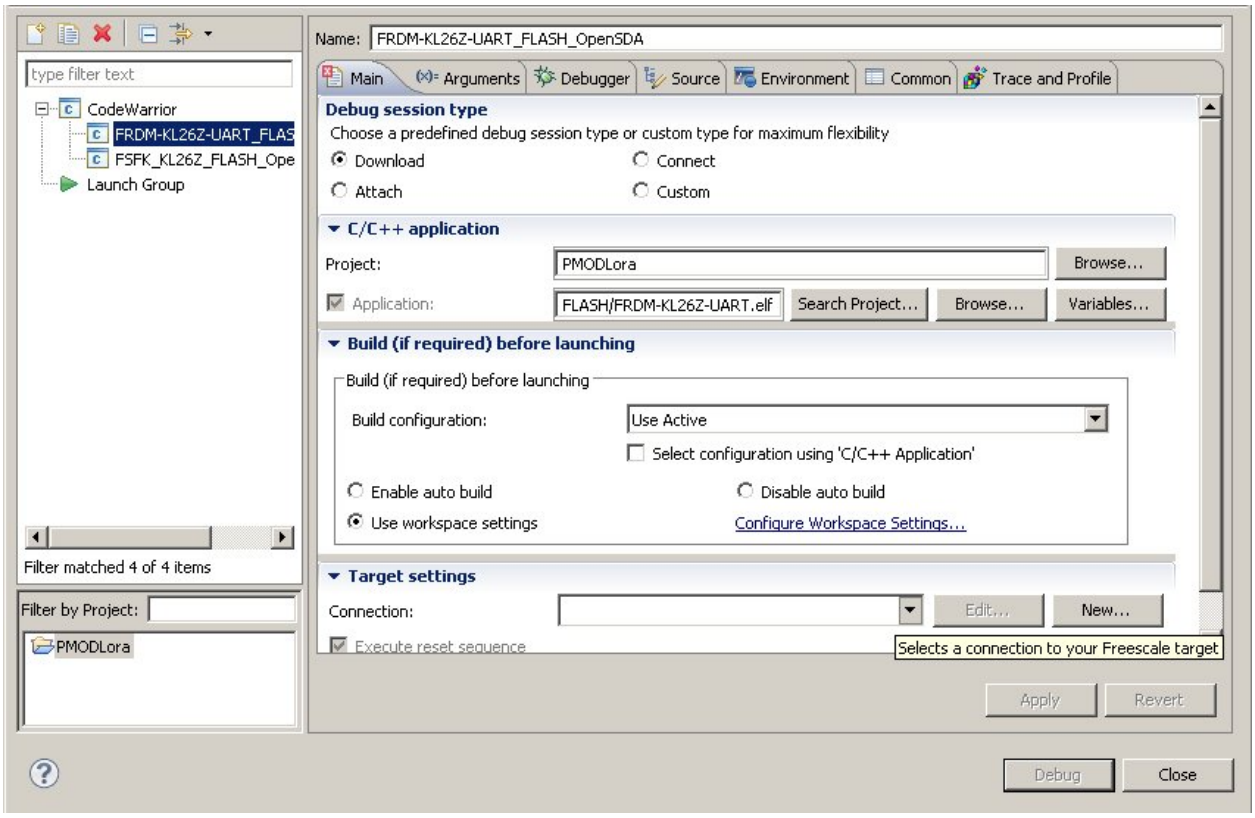
3. In **Component Inspector - A2** select RxD **PTD6** and TxD **PTD7**

⊟ **Receiver**	Enabled	
RxD	ADC0_SE7b/PTD6/LLWU_P15/SPI1_...	ADC0_SE7b/PTD6/LLWU_P15/SPI1_...
⊟ **Transmitter**	Enabled	
TxD	PTD7/SPI1_MISO/UART0_TX/SPI1_M...	PTD7/SPI1_MISO/UART0_TX/SPI1_M...

4. In order to debug you have to connect **PTD6** with **CN7 pin3** and **PTD7** with **CN7 pin2** as in figure. These pin must be disconnected from the board **1066**.



5. Now it's time to compile the sources code, go to **Project -> Build All**
6. Once compiling is finished connect the mini usb from the PC to the FRDM board. Then go to **Run -> Debug configurations...**
7. Finally select FRDM-KL26Z-UART_FLASH_OpenSDA and choose the type of connection **OpenSDA** than click on the **Debug** button.



Processor Expert

Processor Expert Software is a development system to create, configure, optimize, migrate, and deliver software components that generate source code for the microcontroller. For more information please go here,

Hardware Guide

The board is provided with:

- NXP FRDM-KL26Z Board
- Microchip RN2483 Module
- RSR1066 board

The Microchip RN2483 module provides LoRaWAN™ protocol connectivity using a simple UART interface. The NXP **MKLS26Z** is connected to the Microchip module using the configuration 57600 8N1 without using RTS, CTS lines.

Configuration RN2483

The connection used by **RN2483** is **ABP** (Activation by Personalization). To use this connection it is required to setup the RN device only one time. Every Lora Sensor Node bought is already configured. The commands used were:

sys factoryRESET
sys get hweui
mac set deveui [hweui key read]
mac set devaddr [last less significant hexs of hweui key]
mac set appskey AFBECD5647382910019283746 5FAEBDC
mac set nwkskey [hweui key repeated two times]
mac save

After saving this setup is not required repeat the opeation of setup. In order to send data in the Lora network the two used commands are:

- **mac join abp**: used to join the Lora network
- **mac tx cnf 4 18AABBCC**: used to send the frame *"18AABBCC"* on the port 4

Datasheet and more

Please refer to architechboards website.

Index