

# Kinetis MKW2xD and MCR20A IEEE®

## 802.15.4 Software

### Quick Start Guide

This document is a brief presentation of the Kinetis IEEE® 802.15.4 MAC/PHY Software for the MKW2xD wireless microcontrollers and the MCR20A 2.4 GHz wireless transceiver, version 5.0.5. This software package is built using the Kinetis Software Development Kit (KSDK) version 2.0. This document covers installation of the software packages, hardware setup, build and usage of the provided demo applications.

#### Contents

1	Installation .....	2
1.1	IEEE® 802.15.4 Software Installation .....	2
2	Cloning a project .....	5
3	Building the Binaries .....	8
3.1	Building and Flashing the Kinetis IEEE 802.15.4 Software Demo Applications using Kinetis Design Studio .....	9
3.2	Building and Flashing the Kinetis IEEE 802.15.4 Software Demo Applications using IAR Embedded Workbench .....	13
3.3	Flashing a Binary Image File Without Using an IDE .....	15
4	Hardware Setup .....	19
5	Example: Running the MyWirelessApp Demo Application .....	21



# 1 Installation

This section covers the steps for a successful installation of the connectivity software.

## 1.1 IEEE® 802.15.4 Software Installation

Execute the installer and follow the steps presented in the example.

---

### Kinetis MKW2xD IEEE® 802.15.4 Software Installation Example

---

The first page is just a preamble for the installation. Choose next to continue.

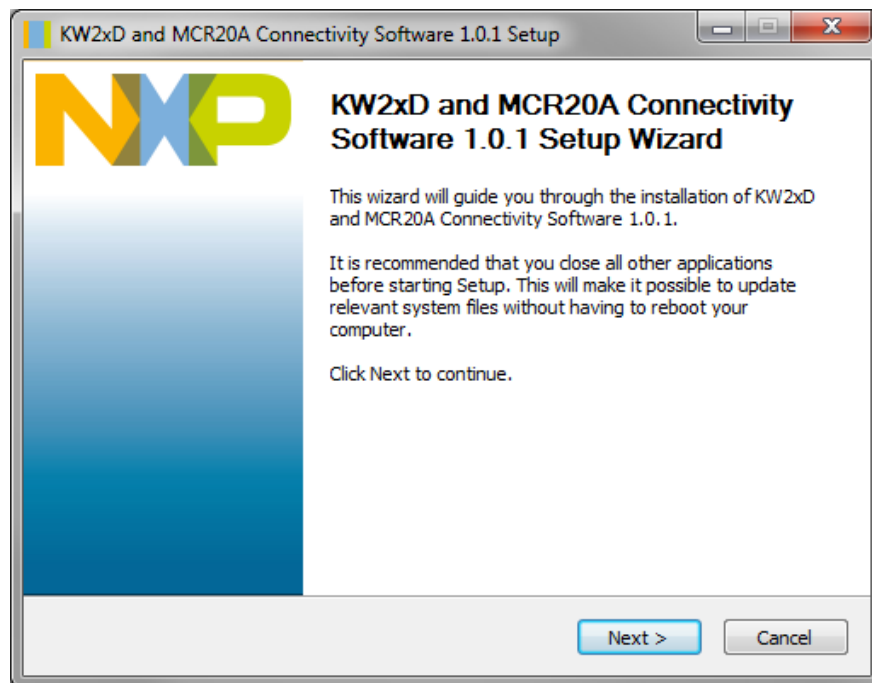
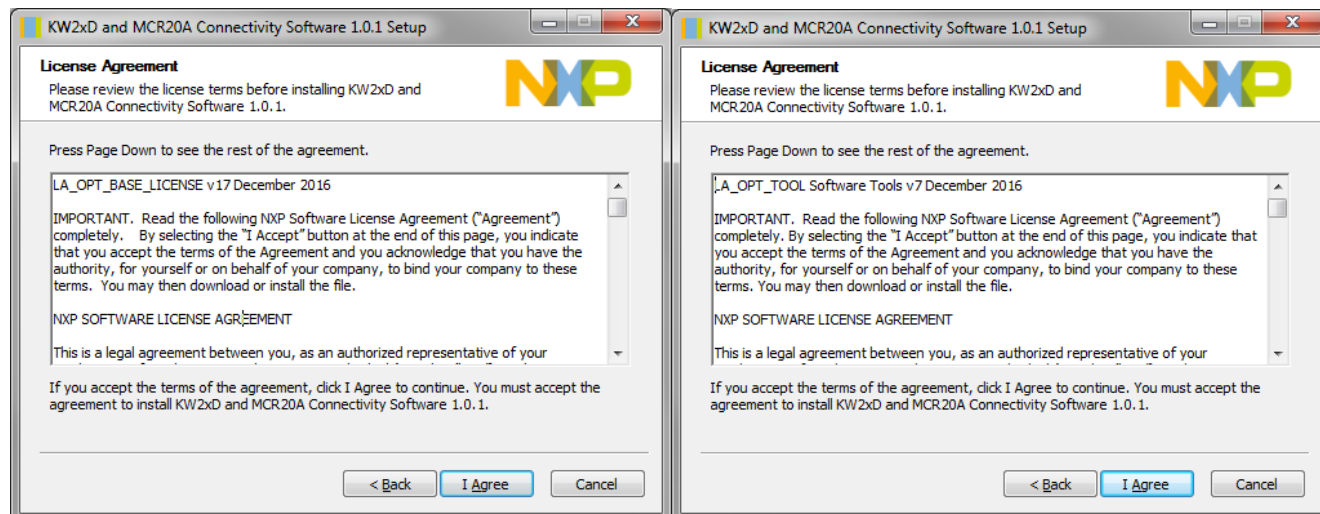


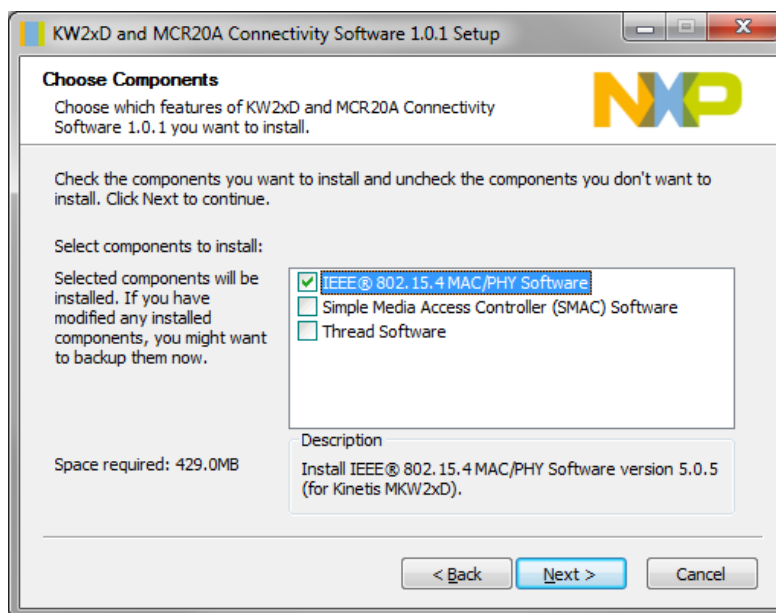
Figure 1: IEEE 802.15.4 software install wizard first screen

The next two pages represent the license agreement. If you accept the terms and conditions please select “I Agree” to continue the installation.



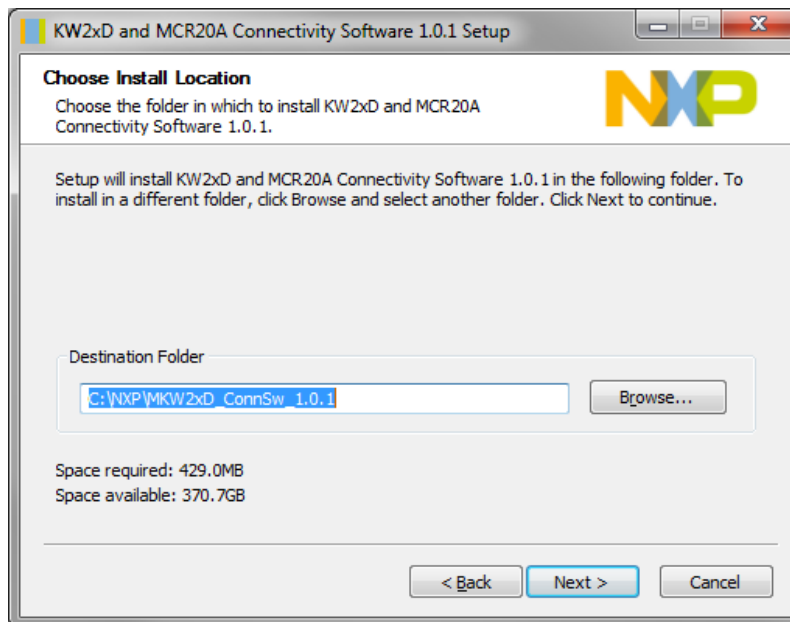
**Figure 2: IEEE 802.15.4 software install wizard license screen**

In the next step you have to check the components you want to install and uncheck the components you don't want to install. In order to install IEEE® 802.15.4 MAC/PHY Software be sure that the corresponding checkbox is enabled as shown in the next figure.



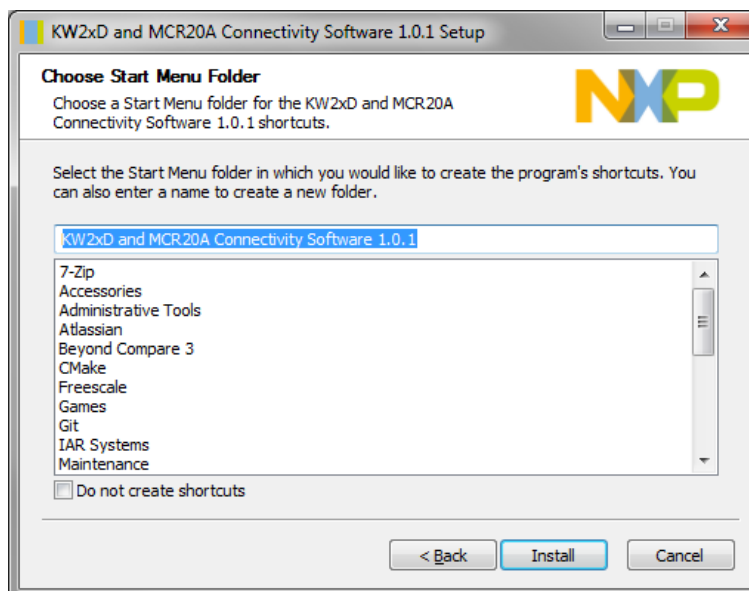
**Figure 3: IEEE 802.15.4 MAC/PHY Software checkbox enable**

The next step is to select the install location for the Kinetis IEEE® 802.15.4 MAC/PHY software. By default the installer uses “C:\NXP”, but this may be changed, depending on your needs.



**Figure 4: IEEE 802.15.4 software install wizard location selection**

The last step is to choose whether you wish to create shortcuts for the IEEE® 802.15.4 MAC/PHY software installation.



**Figure 5: IEEE 802.15.4 MAC/PHY software install wizard creates shortcuts**

## NOTE

The MKW2xD software installer will also contain the KSDK version 2.0 drivers and RTOS enablement for KW24D/KW22D/KW21D, KL46Z and K64F.

## 2 Cloning a project

Navigate to the KW2xD Connectivity Software installation folder and run the Project Cloner application (**MKW2xD\_ConnSw\_1.0.1\tools\wireless\ProjectCloner\ProjectCloner.exe**).

At the first run, the Project Cloner will search for the location of the KW2xD Connectivity Software installation folder. This location can be modified at any time.

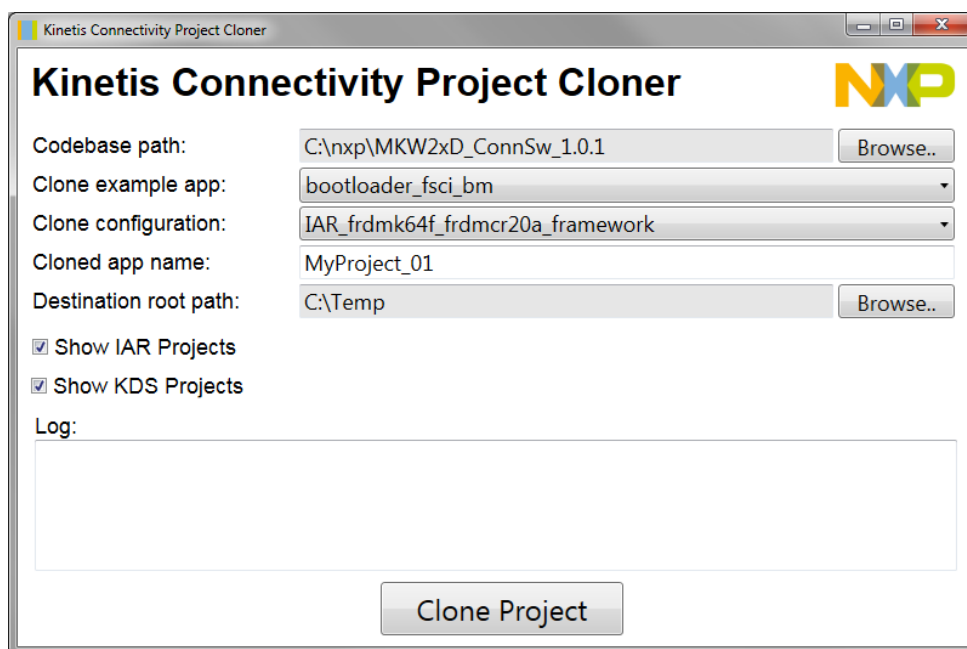


Figure 6: Select the KW2xD Connectivity Software installation path

After the **Codebase path** has been selected correctly, the Project Cloner will display all the Connectivity Software example applications.

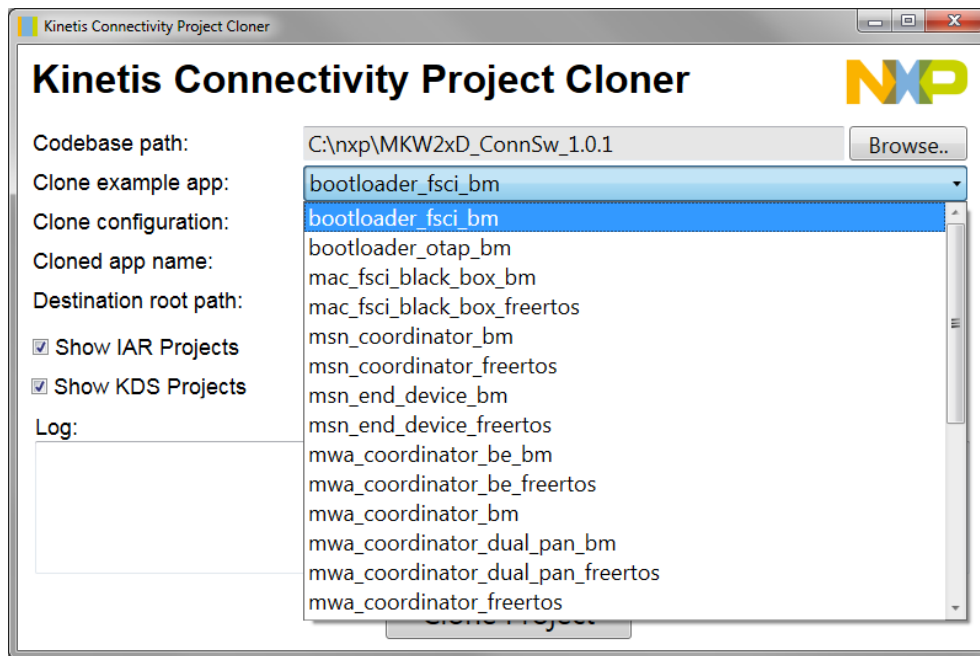


Figure 7: Connectivity Software example applications

Next select the example application to be cloned (**Clone example app**), and the desired configuration (**Clone Configuration**).

By default both IAR and KDS projects are shown in the **Clone configuration** dropdown menu. This can be modified by deselecting one of the two corresponding checkboxes.

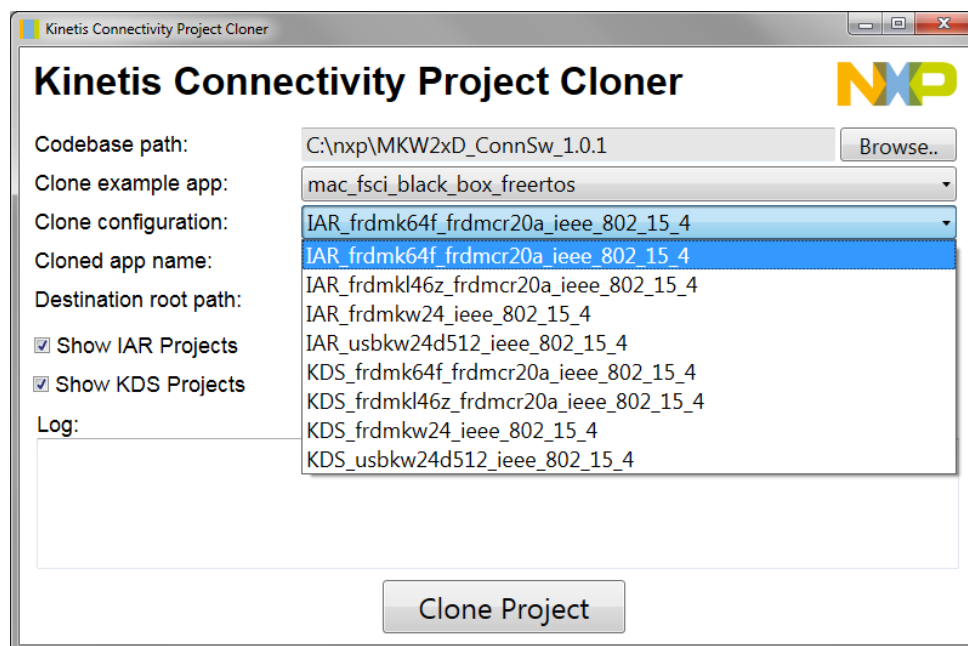


Figure 8: Available configurations for the selected example application

After this, the **Cloned app name** text box will contain a default name for the selected application. This name can be modified to any value.

The default **Destination root path** for the cloned application is the “C:\Temp” folder. To change this path click the **Browse** button to select a new location.

Now press the **Clone Project** button to start the cloning process. The log window will display “Cloning completed” when the process ends.

The Project Cloner will clone all files referenced by the demo application project. The cloned application can be relocated anywhere since it has no dependencies of the original codebase folder.

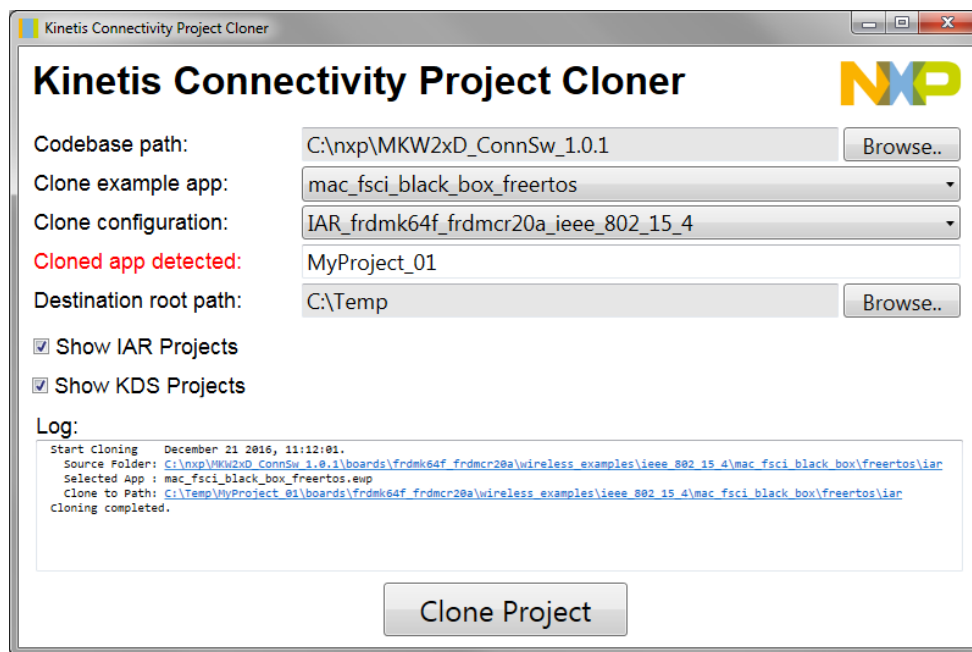


Figure 9: Application cloning done

### 3 Building the Binaries

This section details the required steps for obtaining the binary files for usage with the boards.

#### NOTE

The examples from below have been build using the IAR Embedded Workbench for ARM® version 7.80.1 or higher and Kinetis Design Studio Integrated Development Environment version 3.2.0 or higher. This connectivity software package does not include support for any other toolchains.

The packages must be built with the debug configuration in order to enable debugging information.

This package includes various demo applications that can be used as a starting point.

The next section presents the steps required for building the *mac\_fsci\_black\_box*. All applications can be found using the following placeholders for text:

- <installation\_path> : represents the root path of the cloned application, or the root path for the IEEE 802.15.4 software package.
- <board> : represents the target board for the demo app, can be “frdmkw24d”, “usbkw24d512”, “frdmk64f\_frdmcr20a” or “frdmkl46z\_frdmcr20a”
- <RTOS>: represents the scheduler or RTOS used by the app, can be “bm” or “FreeRTOS”
- <demo\_app> : represents the demo app name
- <IDE>: represents the integrated development studio used to build projects and can be “iar” or “kds”

The demo applications general folder structure is the following:

```
<installation_path>\boards\<board>\wireless_examples\ieee_802_15_4\<demo_app>\<RTOS>\<IDE>\
```

---

#### Kinetis IEEE 802.15.4 Software Demo Application Build Example

---

Selected app: mac\_fsci\_black\_box

Board: frdmkw24d

RTOS: FreeRTOS

Resulting location:

```
<installation_path>\boards\frdmkw24d\wireless_examples\ieee_802_15_4\mac_fsci_black_box\freertos\<IDE>
```



### 3.1 Building and Flashing the Kinetis IEEE 802.15.4 Software Demo Applications using Kinetis Design Studio

#### Step 1:

Open the KDS IDE and create a new workspace.

#### Step 2:

Import the project into Workspace: File -> Import -> General -> Existing Projects into Workspace.

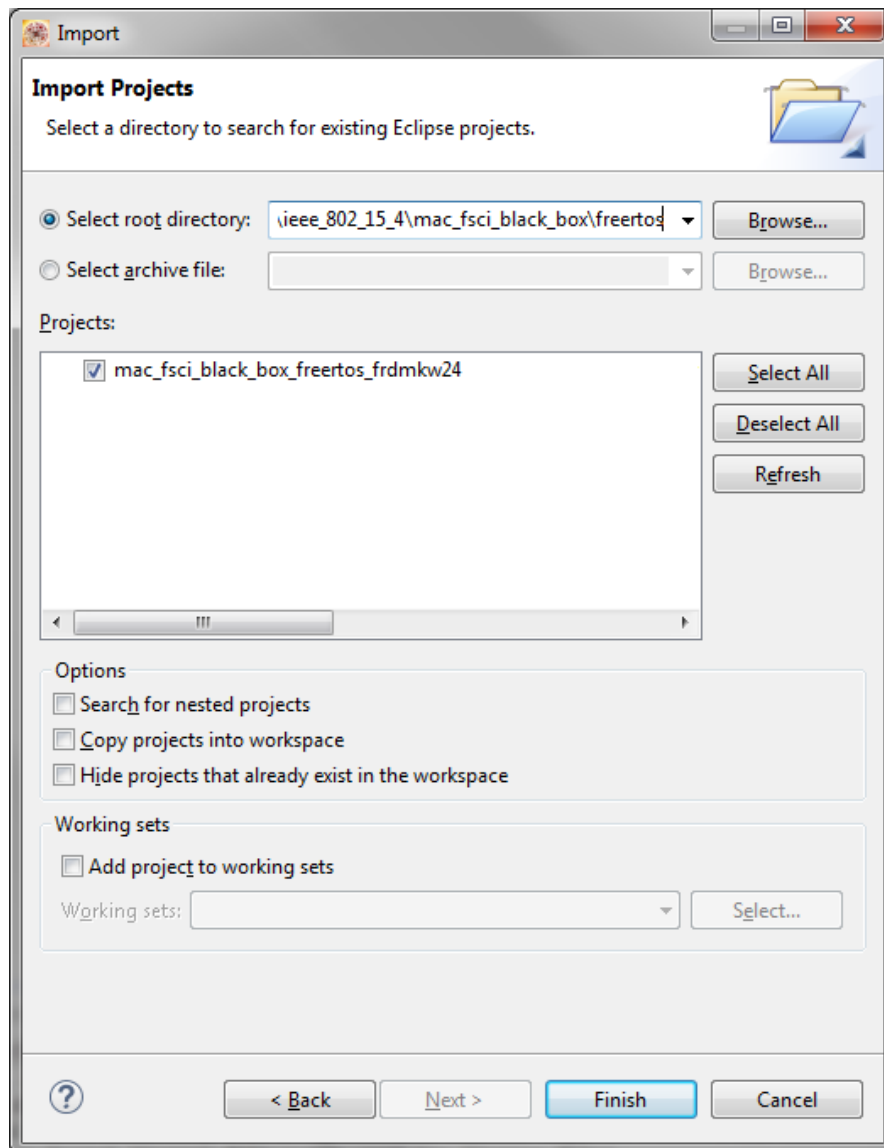


Figure 10: “mac\_fsci\_black\_box” import project

### Step 3:

Select the mac\_fsci\_black\_box project.

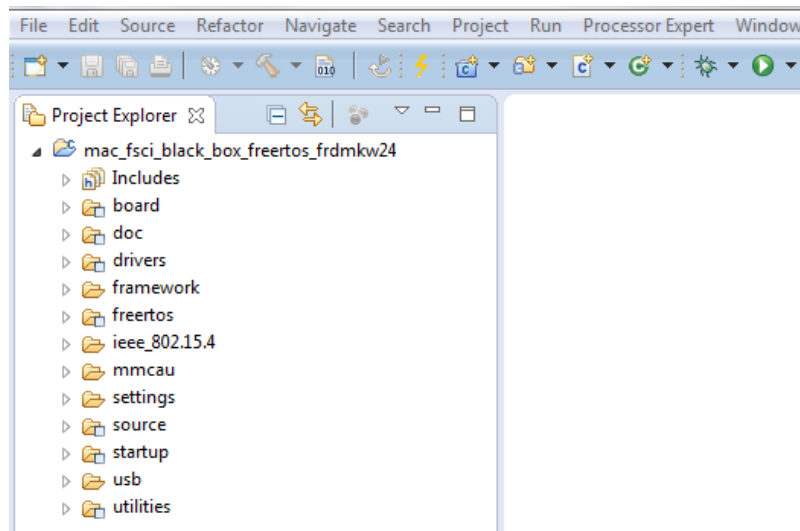


Figure 11: “mac\_fsci\_black\_box” FreeRTOS KDS project

### Step 4:

Build the mac\_fsci\_black\_box project.

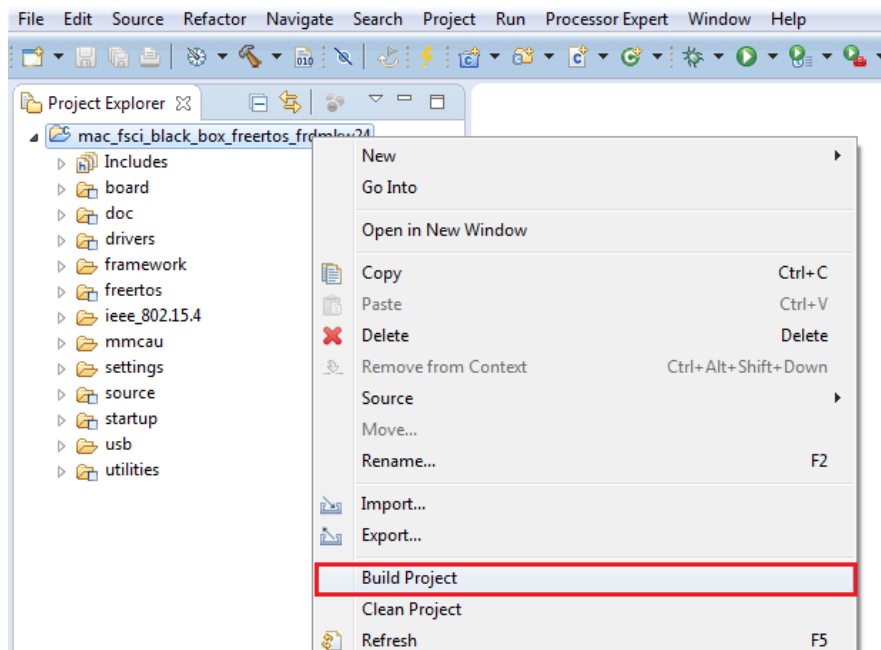
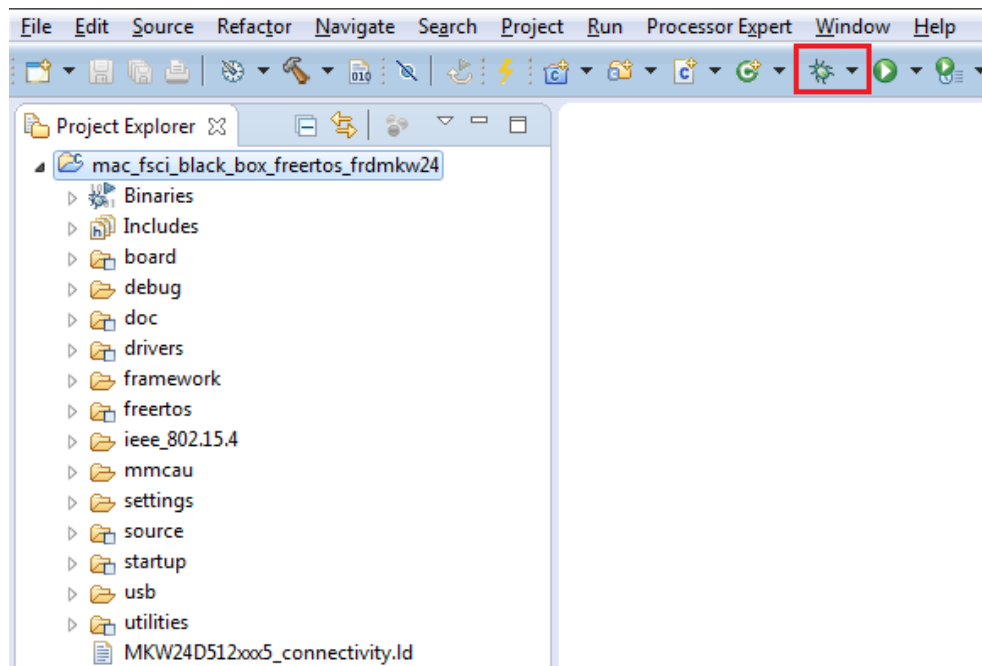


Figure 12: “mac\_fsci\_black\_box” FreeRTOS build

### Step 5:

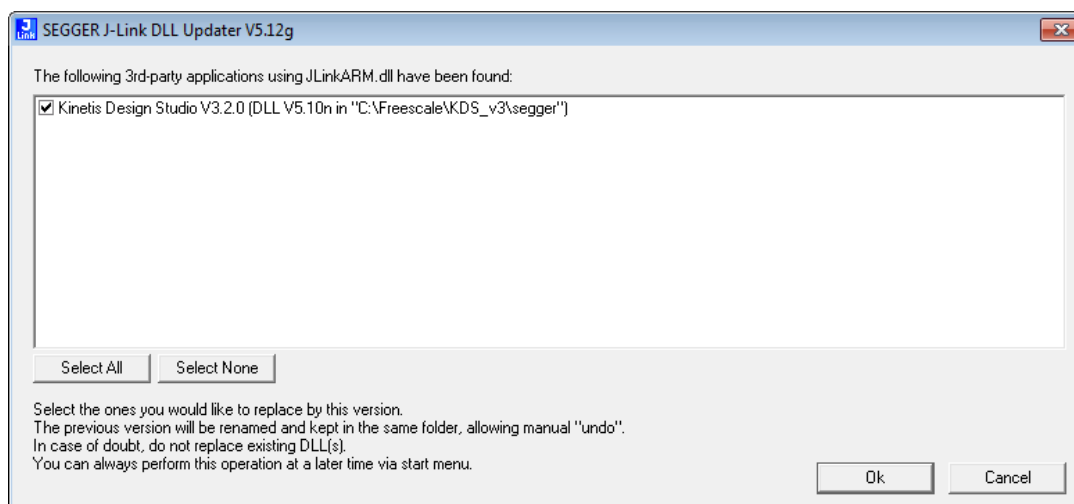
Click the “Debug” button to flash the executable onto the board.



**Figure 13: “mac\_fsci\_black\_box” Debug**

### NOTE

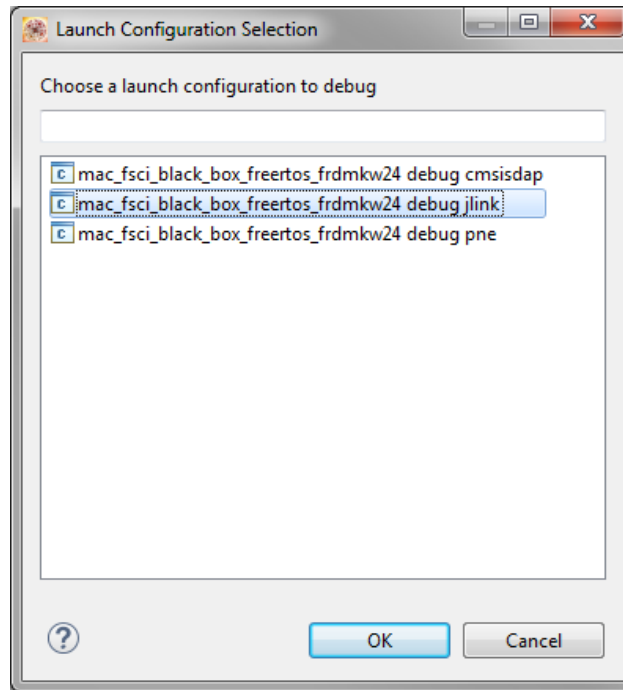
Please make sure that you install the latest J-Link driver and link it with KDS. To do this, download the driver from <https://www.segger.com/jlink-software.html> and install it. When you are asked to associate the applications which use J-Link driver with this version of driver you only have to check the box near Kinetis Design Studio and click OK.



**Figure 14: Update KDS J-Link Driver**

### Step 6:

Select the J-Link debug configuration option when asked for Launch Configuration.



**Figure 15: “mac\_fsci\_black\_box” debug configuration**

### NOTE

The projects are configured to use “CMSIS-DAP” as the default debugger. Please make sure that your board’s OpenSDA chip contains a J-Link firmware or that the debugger selection corresponds to the physical interface used to interface to the board. See the section 2.3 for more information.

## 3.2 Building and Flashing the Kinetis IEEE 802.15.4 Software Demo Applications using IAR Embedded Workbench

### Step 1:

Navigate to the resulting location in either the connectivity software installation directory or the cloned application root directory.

### Step 2:

Open the highlighted IAR workspace file (\*.eww file format):

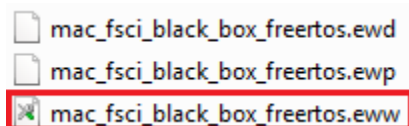


Figure 16: “mac\_fsci\_black\_box” demo project location

### Step 3:

Select the mac\_fsci\_black\_box project.

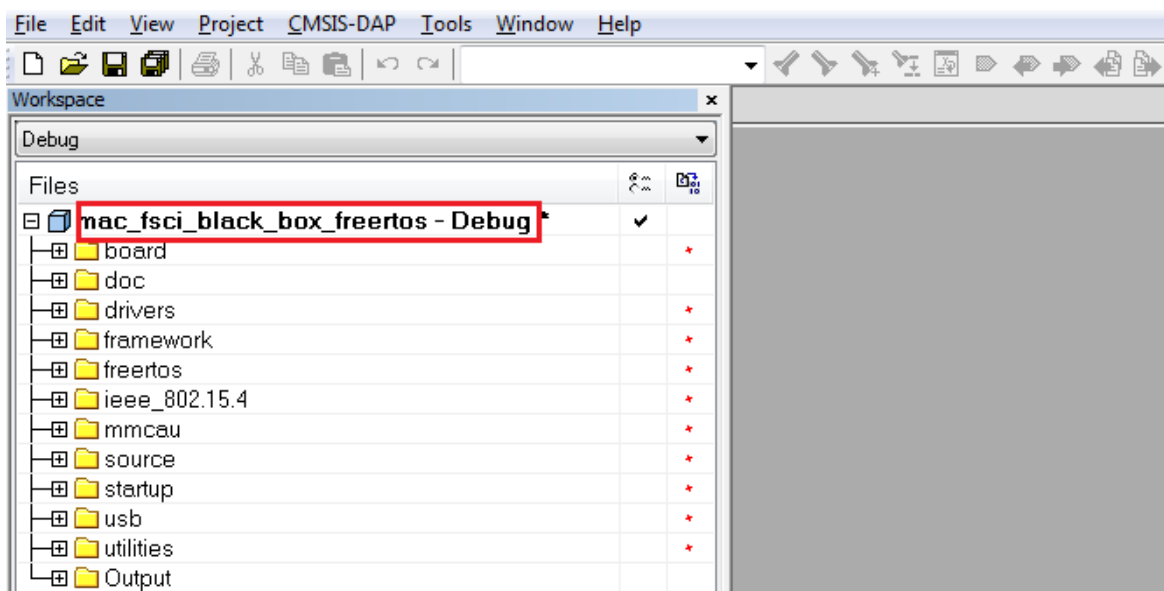


Figure 17: mac\_fsci\_black\_box FreeRTOS IAR project

#### Step 4:

Build the mac\_fsci\_black\_box project.

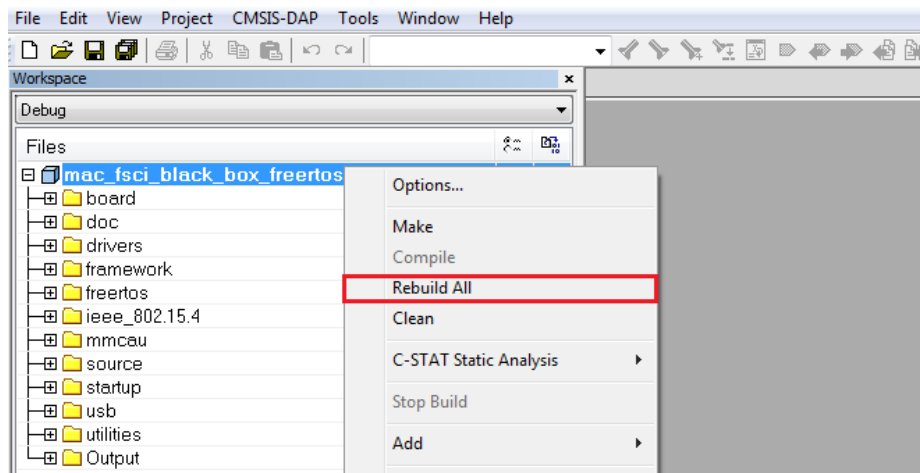


Figure 18: mac\_fsci\_black\_box build

#### Step 5

Make the appropriate debugger settings in the project options window:

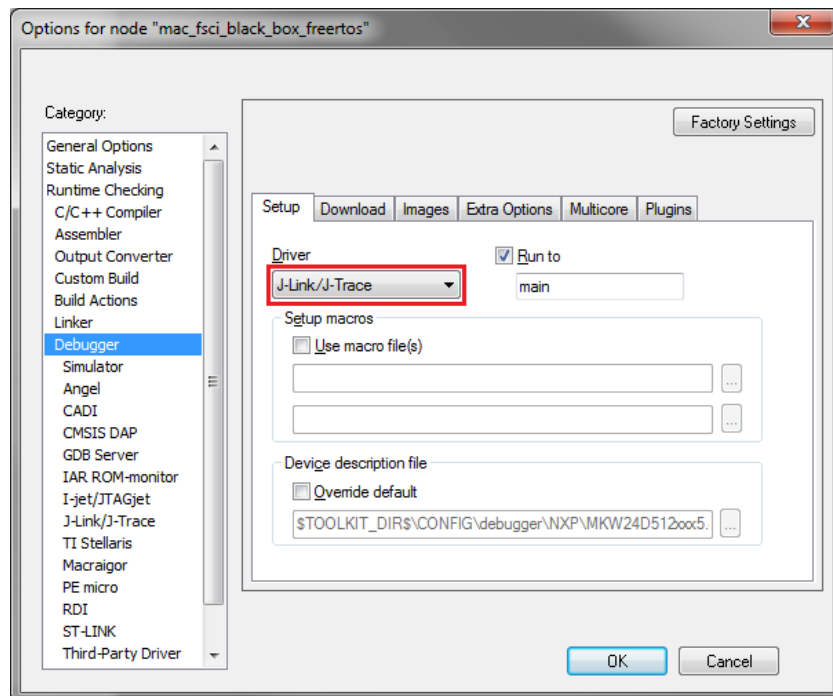
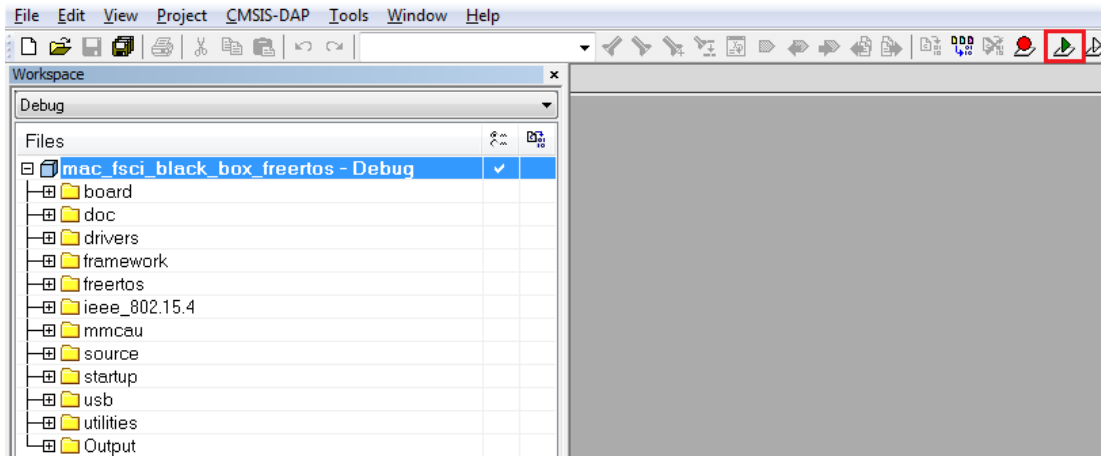


Figure 19: Debugger Settings

## Step 6:

Click the “Download and Debug” button to flash the executable onto the board.



**Figure 20:** mac\_fsci\_black\_box Download and Debug

### NOTE

The projects are configured to use “J-Link / J-Trace” as the default debugger. Please make sure that your board’s OpenSDA chip contains a J-Link firmware or that the debugger selection corresponds to the physical interface used to interface to the board. See the section below for more information.

## 3.3 Flashing a Binary Image File Without Using an IDE

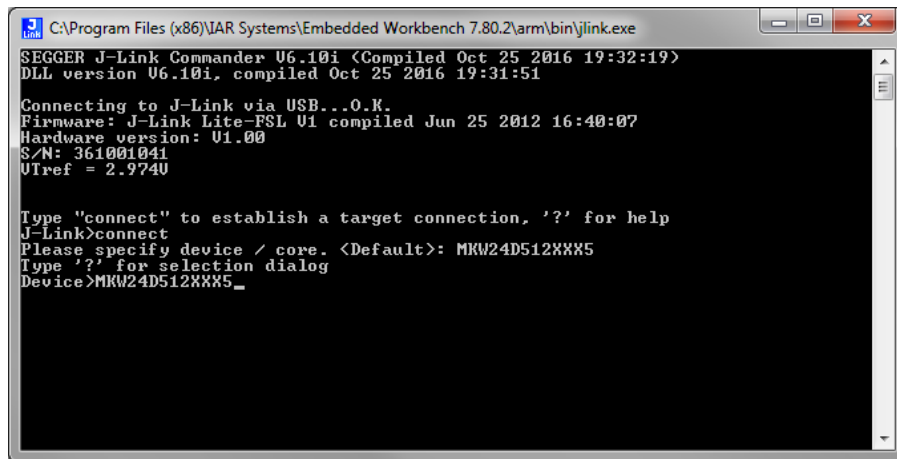
The MKW2xD connectivity software package contains in the `<installation_path>\tools\wireless\binaries` folder a series of pre-compiled binary applications that can be flashed onto a development board.

In order to flash the corresponding binaries to the FRDM-KW24D board, the best approach is to use the OpenSDA on-board interface J-Link Mass Storage Device functionality, by simply dragging and dropping the binary image in the mass storage drive exposed by this OpenSDA firmware.

In order to flash the firmware on the USB-KW24D512, a J-Link probe is needed along with the latest J-Link software from [www.segger.com](http://www.segger.com).

Run the `jlink.exe` executable provided in the J-Link software installation follow the steps below for flashing the image on the microcontroller. Make sure that the binary file is in the same folder with the `jlink.exe` executable, or specify the absolute path to the file.

Step 1: Select MKW24D512XXX5 device.



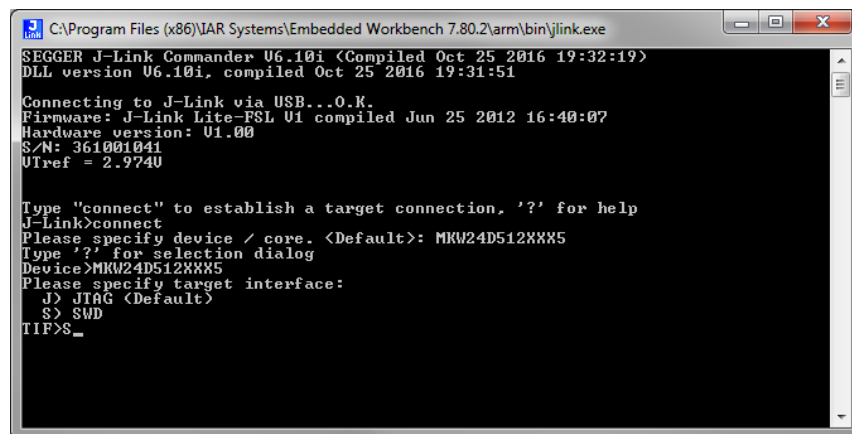
```
C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
SEGGER J-Link Commander V6.10i (Compiled Oct 25 2016 19:32:19)
DLL version V6.10i, compiled Oct 25 2016 19:31:51

Connecting to J-Link via USB...O.K.
Firmware: J-Link Lite-FSL V1 compiled Jun 25 2012 16:40:07
Hardware version: V1.00
S/N: 361001041
UTref = 2.9740

Type "connect" to establish a target connection, '?' for help
J-Link>connect
Please specify device / core. <Default>: MKW24D512XXX5
Type '?' for selection dialog
Device>MKW24D512XXX5_
```

Figure 21: MKW24D512xxx5 device selection

Step 2: Select SWD target interface.



```
C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
SEGGER J-Link Commander V6.10i (Compiled Oct 25 2016 19:32:19)
DLL version V6.10i, compiled Oct 25 2016 19:31:51

Connecting to J-Link via USB...O.K.
Firmware: J-Link Lite-FSL V1 compiled Jun 25 2012 16:40:07
Hardware version: V1.00
S/N: 361001041
UTref = 2.9740

Type "connect" to establish a target connection, '?' for help
J-Link>connect
Please specify device / core. <Default>: MKW24D512XXX5
Type '?' for selection dialog
Device>MKW24D512XXX5
Please specify target interface:
J> JTAG <Default>
S> SWD
TIF>S_
```

Figure 22: SWD interface selection

Step 3: Press “Enter” to select the default interface speed.



```

C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
SEGGER J-Link Commander V6.10i (Compiled Oct 25 2016 19:32:19)
DLL version V6.10i, compiled Oct 25 2016 19:31:51

Connecting to J-Link via USB...O.K.
Firmware: J-Link Lite-FSL V1 compiled Jun 25 2012 16:40:07
Hardware version: V1.00
S/N: 361001041
URef = 2.9740

Type "connect" to establish a target connection, '?' for help
J-Link>connect
Please specify device / core. (Default): MKW24D512XXX5
Type '?' for selection dialog
Device>MKW24D512XXX5
Please specify target interface:
J) JTAG (Default)
S) SWD
TIF>S
Specify target interface speed [kHz]. (Default): 4000 kHz
Speed>_

```

Figure 23: SWD interface speed selection

```

C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
Device>MKW24D512XXX5
Please specify target interface:
J) JTAG (Default)
S) SWD
TIF>S
Specify target interface speed [kHz]. (Default): 4000 kHz
Speed>_
Device "MKW24D512XXX5" selected.

Found SWD-DP with ID 0x2BA01477
Found SWD-DP with ID 0x2BA01477
AP-IDR: 0x24770011, Type: AHB-AP
Found Cortex-M4 r0p1, Little endian.
FPUnit: 6 code (BP) slots and 2 literal slots
CoreSight components:
ROMTbl 0 @ E00FF000
ROMTbl 0 [0]: FFF0F000, CID: B105E00D, PID: 000BB000 SCS
ROMTbl 0 [1]: FFF02000, CID: B105E00D, PID: 003BB002 DWT
ROMTbl 0 [2]: FFF03000, CID: B105E00D, PID: 002BB003 FPB
ROMTbl 0 [3]: FFF01000, CID: B105E00D, PID: 003BB001 ITM
ROMTbl 0 [4]: FFF41000, CID: B105900D, PID: 000BB9A1 TPIU
ROMTbl 0 [5]: FFF42000, CID: B105900D, PID: 000BB925 ETM
Cortex-M4 identified.
J-Link>

```

Figure 24: Cortex-M4 identified

Step 4: Type **loadbin app.bin 0** in order to flash the binary file (assuming application name is *app.bin*).

```

C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
Device>MKW24D512XXX5
Please specify target interface:
J) JTAG (Default)
S) SWD
TIF>S
Specify target interface speed [kHz]. (Default): 4000 kHz
Speed>_
Device "MKW24D512XXX5" selected.

Found SWD-DP with ID 0x2BA01477
Found SWD-DP with ID 0x2BA01477
AP-IDR: 0x24770011, Type: AHB-AP
Found Cortex-M4 r0p1, Little endian.
FPUnit: 6 code (BP) slots and 2 literal slots
CoreSight components:
ROMTbl 0 @ E00FF000
ROMTbl 0 [0]: FFF0F000, CID: B105E00D, PID: 000BB000 SCS
ROMTbl 0 [1]: FFF02000, CID: B105E00D, PID: 003BB002 DWT
ROMTbl 0 [2]: FFF03000, CID: B105E00D, PID: 002BB003 FPB
ROMTbl 0 [3]: FFF01000, CID: B105E00D, PID: 003BB001 ITM
ROMTbl 0 [4]: FFF41000, CID: B105900D, PID: 000BB9A1 TPIU
ROMTbl 0 [5]: FFF42000, CID: B105900D, PID: 000BB925 ETM
Cortex-M4 identified.
J-Link>loadbin app.bin 0

```

Figure 25: Load binary file

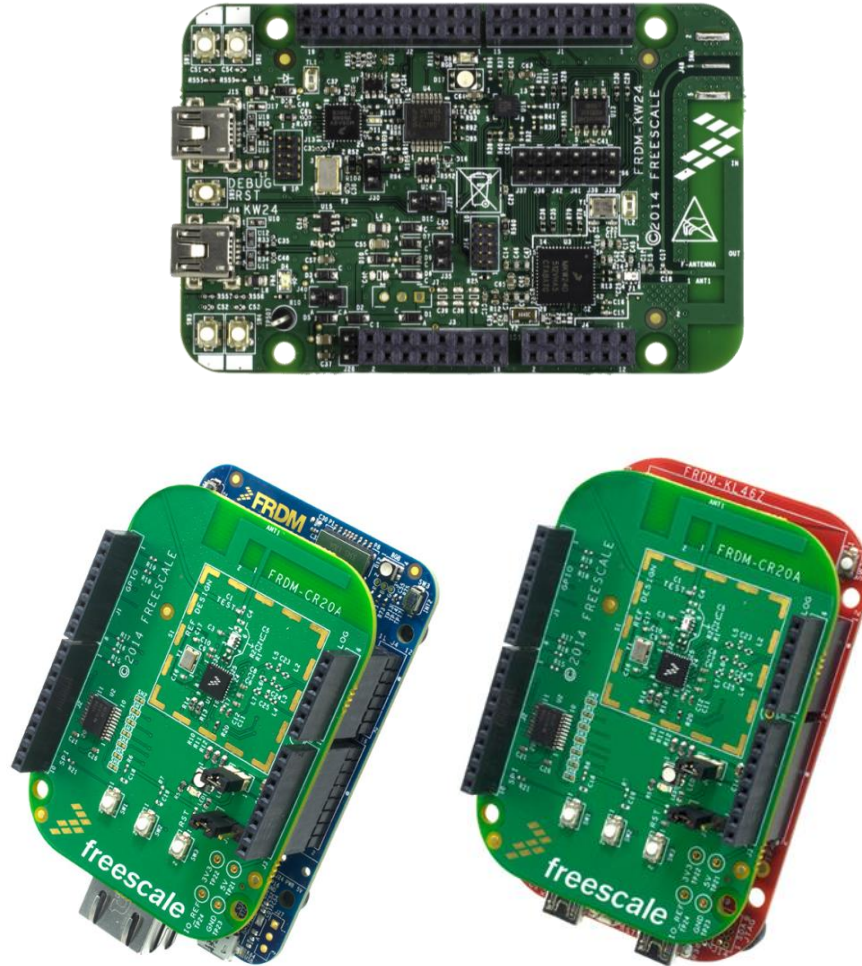
```
C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.80.2\arm\bin\jlink.exe
Device "MKW24D512XXX5" selected.

Found SWD-DP with ID 0x2BA01477
Found SWD-DP with ID 0x2BA01477
AP-IDR: 0x24770011, Type: AHB-AP
Found Cortex-M4 r0p1, Little endian.
FPUnit: 6 code (BP) slots and 2 literal slots
CoreSight components:
ROMTbl 0 @ E00FF000
ROMTbl 0 [0]: FFF0F000, CID: B105E00D, PID: 000BB000 SCS
ROMTbl 0 [1]: FFF02000, CID: B105E00D, PID: 003BB002 DWT
ROMTbl 0 [2]: FFF03000, CID: B105E00D, PID: 002BB003 FPB
ROMTbl 0 [3]: FFF01000, CID: B105E00D, PID: 003BB001 ITM
ROMTbl 0 [4]: FFF41000, CID: B105900D, PID: 000BB9A1 TPIU
ROMTbl 0 [5]: FFF42000, CID: B105900D, PID: 000BB925 ETM
Cortex-M4 identified.
J-Link>loadbin app.bin 0
Halting CPU for downloading file.
Downloading file fapp.binl...
J-Link: Flash download: Flash programming performed for 1 range (2048 bytes)
J-Link: Flash download: Total time needed: 0.189s (Prepare: 0.095s, Compare: 0.0
05s, Erase: 0.013s, Program: 0.061s, Verify: 0.001s, Restore: 0.011s)
O.K.
J-Link>
```

Figure 26: Download completed successfully

## 4 Hardware Setup

The hardware setup in this example uses a FRDM-KW24 development platform. This platform as well as FRDM-CR20A connected to FRDM-K64F and FRDM-KL46Z are shown in the figure below:



**Figure 18: FRDM-CR20A connected to FRDM-K64F and FRDM-KL46 and FRDM-KW24**

The FRDM-KW24, FRDM-K64F and FRDM-KL64Z boards should have their OpenSDA USB ports connected via mini and micro-USB cables respectively to a Windows PC. The OpenSDA chip on the motherboards should have appropriate firmware flashed, with debugging and virtual serial COM port capabilities. For more information on OpenSDA please refer to the following webpage:


[www.nxp.com/opensda](http://www.nxp.com/opensda).

Variants of embedded firmware for the OpenSDA chip can be downloaded from:

<https://github.com/mbedmicro/CMSIS-DAP>

<https://www.segger.com/opensda.html>

<http://www.pemicro.com/opensda/>



CMSIS-DAP is the default interface selected in the IDE projects for FRDM-KW24 and FRDM-K64F with MCR20A included in this release.

## 5 Example: Running the MyWirelessApp Demo Application

The MAC “MyWirelessApp” demo application requires a serial terminal program to connect to the boards. For this example, [Tera Term](#) was chosen.

### Step 1:

Load the applications on the boards using IAR Embedded Workbench for ARM®. This demo has two configurations: a “coordinator” and an “end device”.

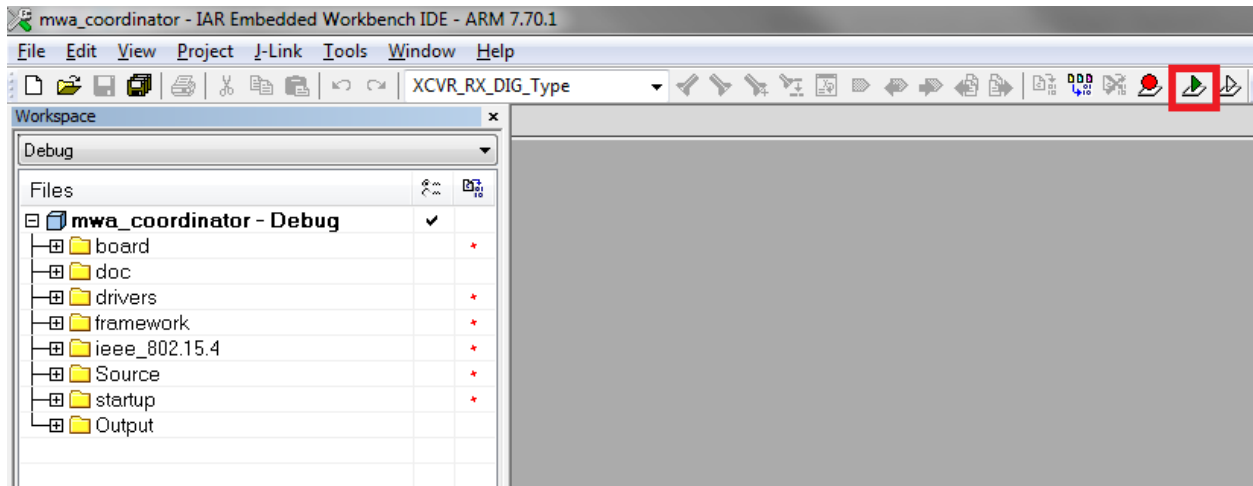


Figure 27: MyWirelessApp coordinator project

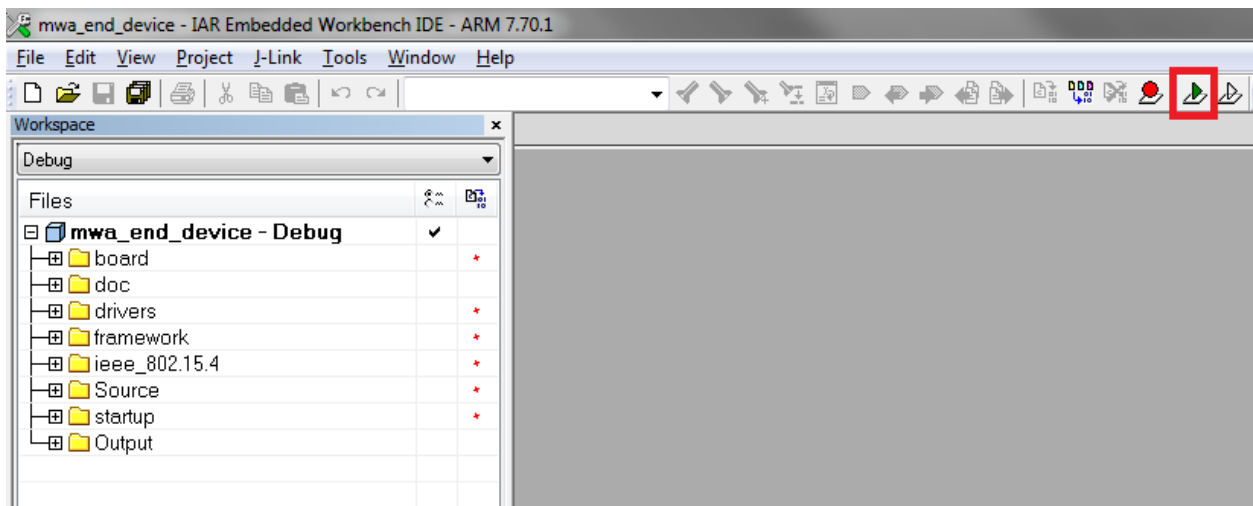


Figure 28: MyWirelessApp end device project

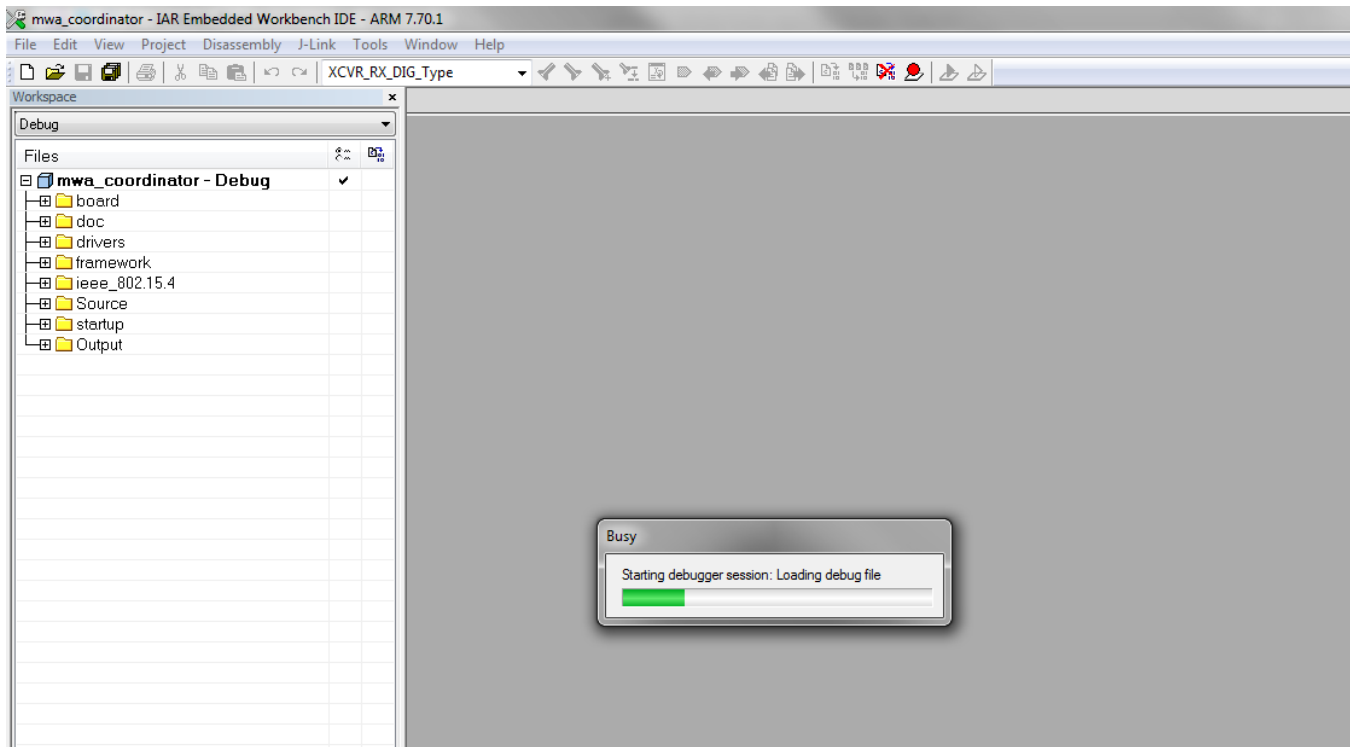


Figure 29: MyWirelessApp Coordinator loading stage example

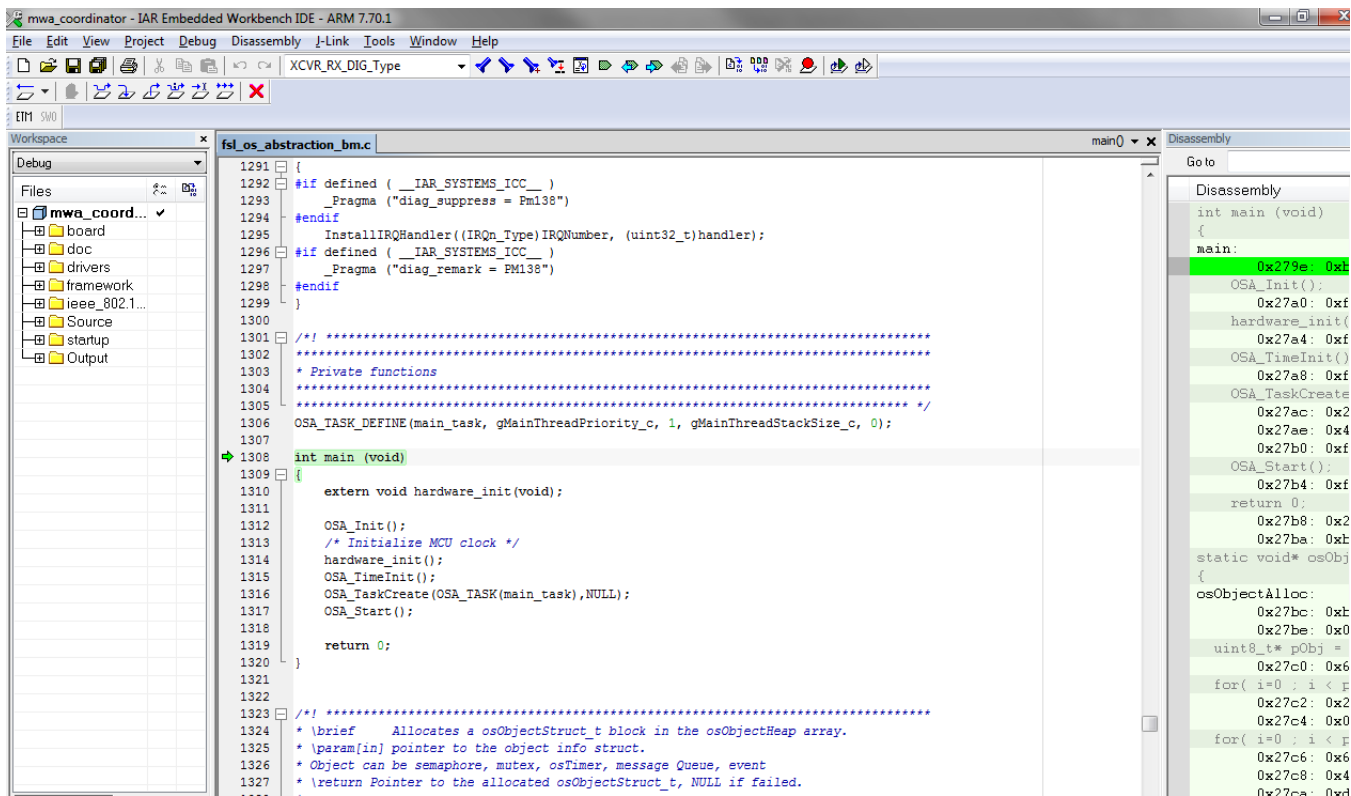
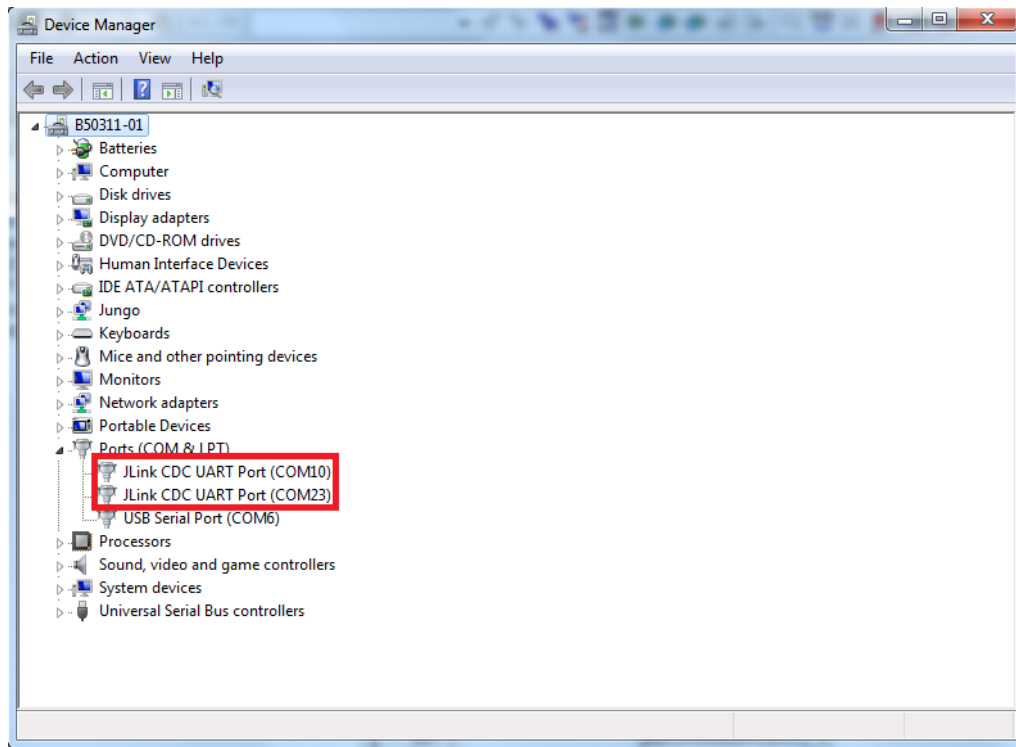


Figure 30: MyWirelessApp Coordinator application loaded

### Step 2:

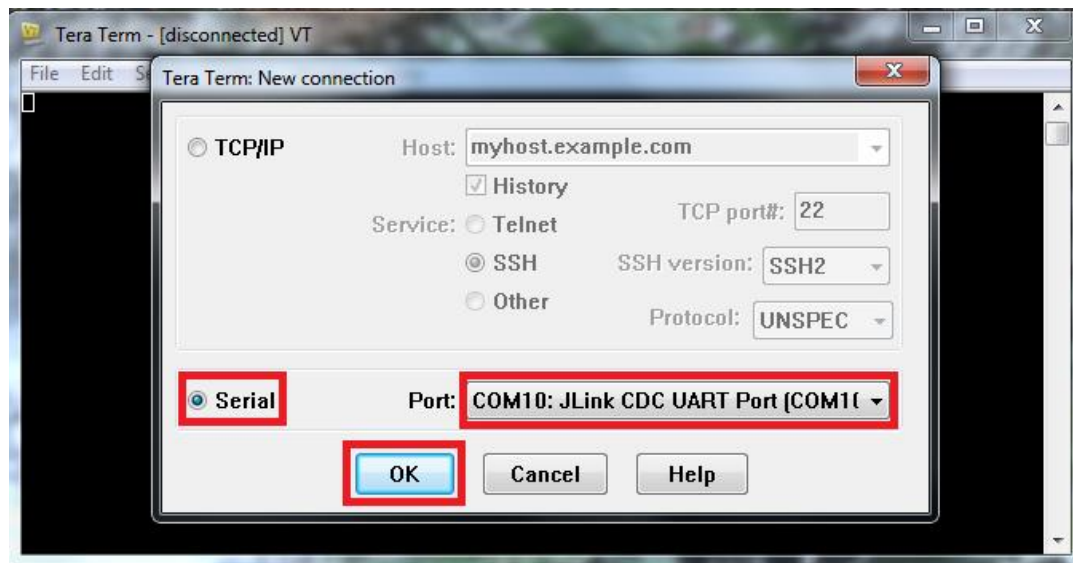
After loading both application check “Device Manager” to get the serial ports numbers. These should appear with the prefix “JLink”.



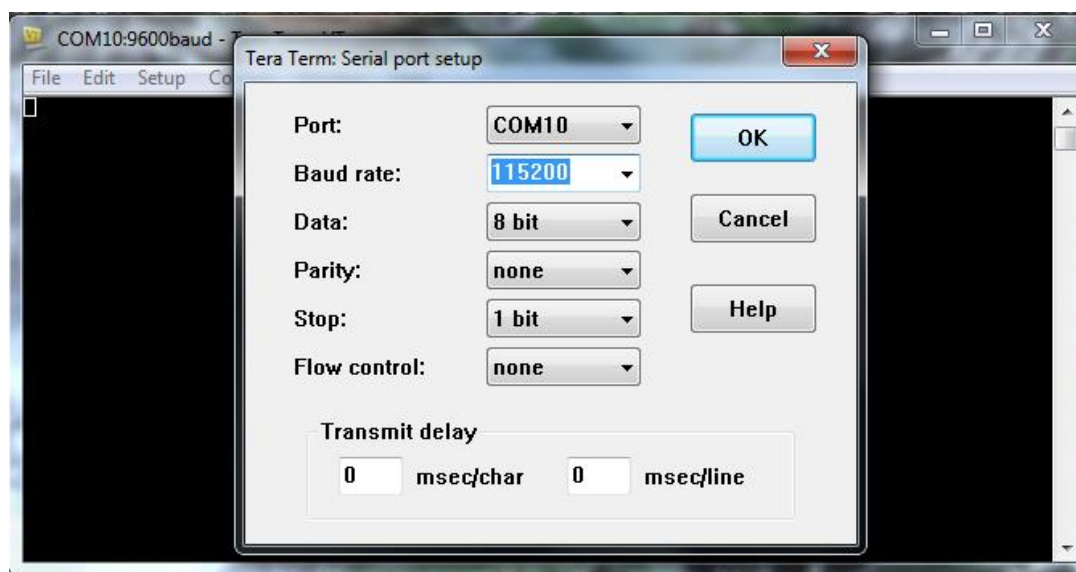
**Figure 31: Device Manager serial port look up**

### Step 3:

Using the port numbers specified in Device Manager, open two Tera Term instances and connect to the devices using the 115200 baud rate. To change the baud rate of the terminal go to “Setup-> Serial Port” menu.



**Figure 32: Select JLink serial connection COM port**



**Figure 33: Setting correct baud rate**

#### **Step 4:**

Start the applications by pressing any available key on the FRDM boards: first the coordinator and then the end device.



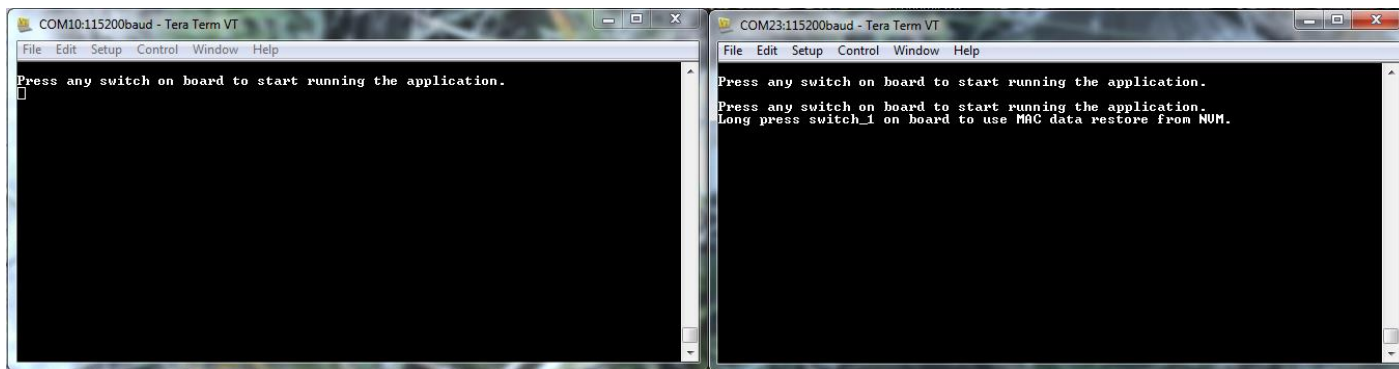


Figure 34: Both applications after a reset

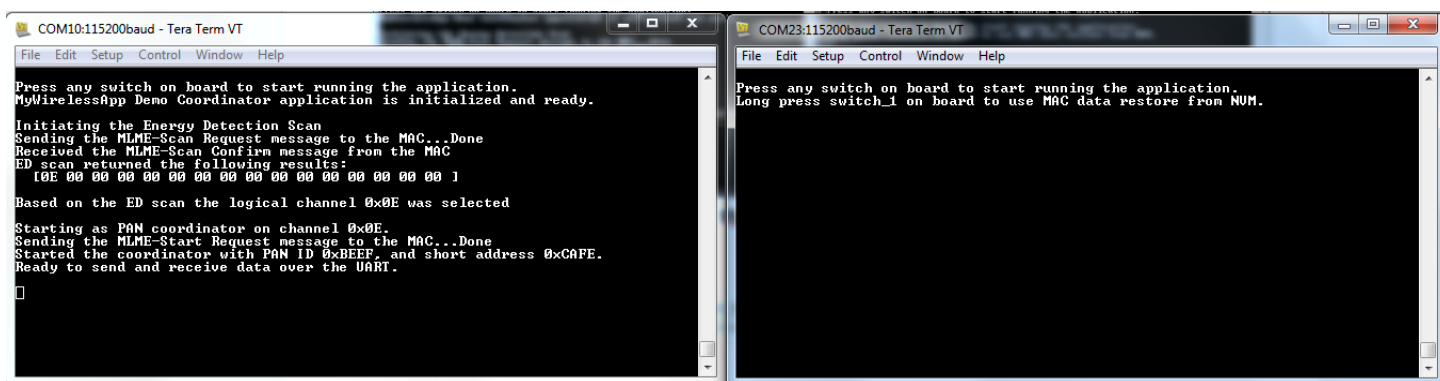


Figure 35: MyWirelessApp coordinator started

The coordinator performs an energy detect scan to determine the least occupied channel and then selects it to start the network. The end device performs an active scan and after receiving a beacon from a coordinator, it issues an association request.

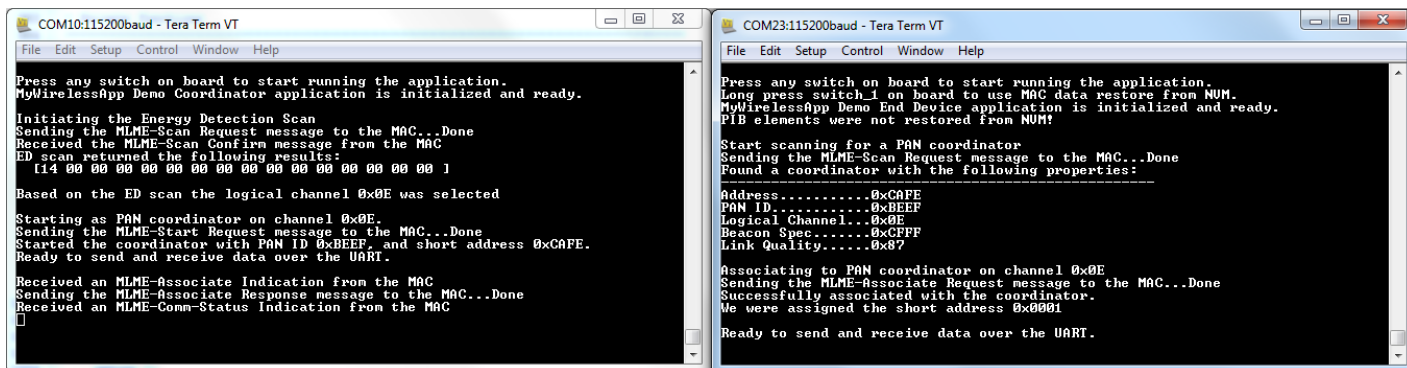
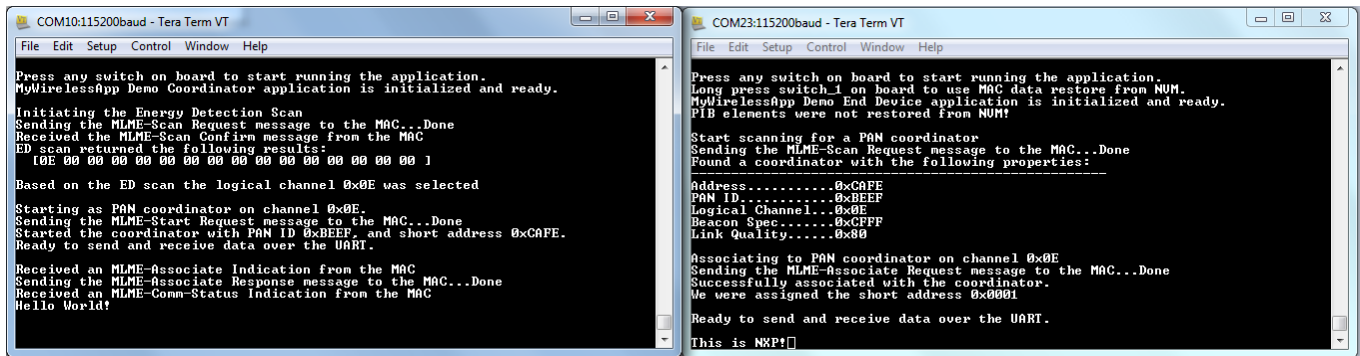


Figure 36: MyWirelessApp end device associated to coordinator

After the association procedure completes, in the two terminal windows messages can be written to be exchanged by the two wireless nodes.



The previous section demonstrates the basic steps to run a demo application. For detailed information about the demo applications, please refer the Demo Applications User's Guide included in the installer (*IEEE 802.15.4 MAC Demo Applications User's Guide.pdf*).

---

**How to Reach Us:**

**Home Page:**

[www.nxp.com](http://www.nxp.com)

**Web Support:**

[www.nxp.com/support](http://www.nxp.com/support)

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: [nxp.com/SalesTermsandConditions](http://nxp.com/SalesTermsandConditions).

Freescale, the Freescale logo, and Kinetis are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. IEEE 802.15.4 is a trademark of the Institute of Electrical and Electronics Engineers, Inc. (IEEE). This product is not endorsed or approved by the IEEE. All other product or service names are the property of their respective owners. ARM, the ARM powered logo, and Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. ZigBee is a registered trademark of ZigBee Alliance, Inc. All rights reserved.

© 2016 Freescale Semiconductor, Inc.

Document number: MKW2xD802154SW5050QSG  
Rev. 0  
01/2017

