MKW40ZBLE114SWQSG
Rev. 0, 11/2015

MKW40Z Bluetooth® Low Energy Software Quick Start Guide

1 Introduction

This document is a brief presentation of the Freescale Bluetooth® Low Energy Software for the KW40Z wireless microcontroller platforms version 1.1.4. This software package contains Kinetis Software Development Kit (KSDK) sources. This document covers installation of the software package, hardware setup, build and usage of the provided demo applications.

2 Installation

This section covers the steps for a successful installation of the required software packages: connectivity and Kinetis SDK.

Contents

The first step is to download the “KW40Z_Connectivity_Software_1.0.1.exe” installer.

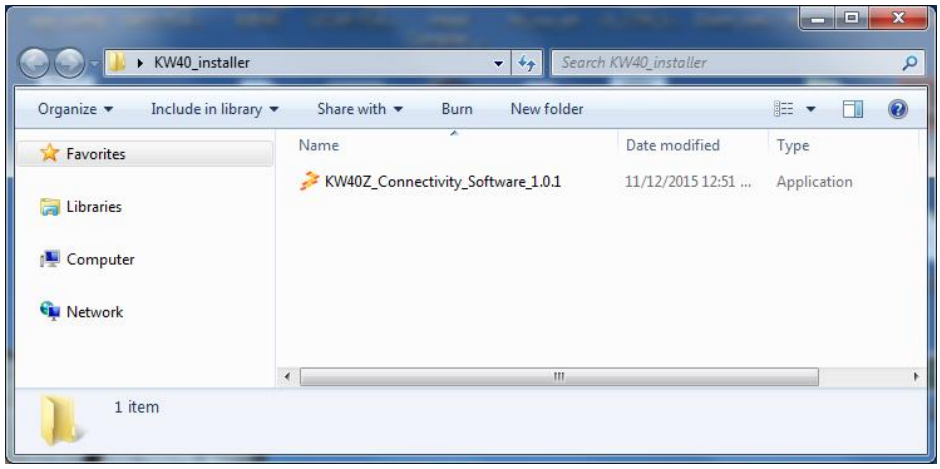

Figure 1: The KW40Z Connectivity Software Installer

On the main screen, press the *Next* button.

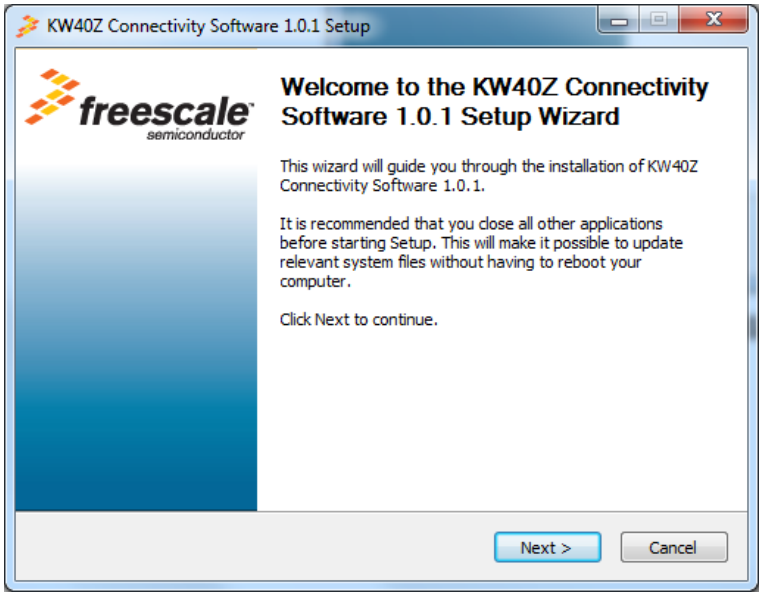

Figure 2: KW40Z Installer main screen

On the *License Agreement* screen press the *I agree* button to accept the license agreement.

Figure 3: License agreement screen

On the next screen click *Browse* to select another destination folder for the KW40Z Connectivity Software installation or click the *Next* button to continue.

Figure 4: Destination folder selection screen

On the following screen uncheck the first two options and press the *Next* button.

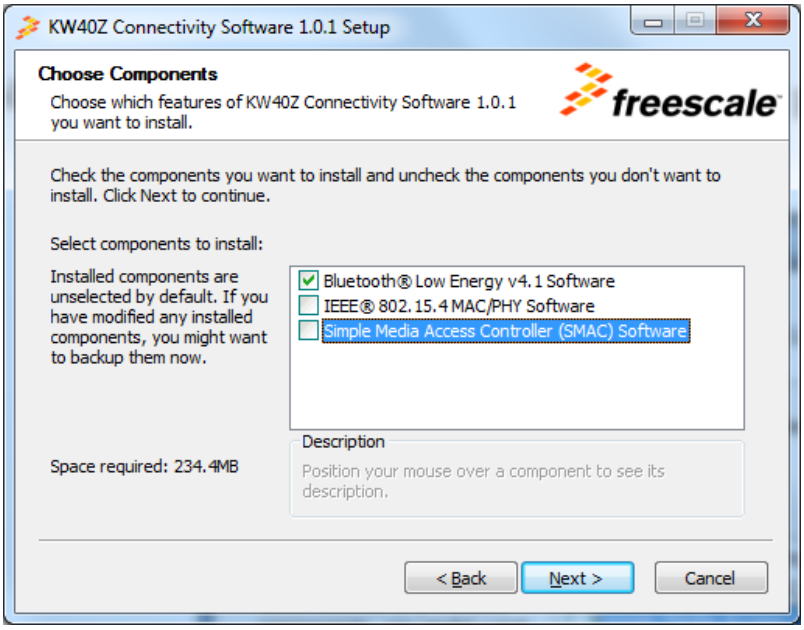

Figure 5: Component selection screen

Select a *Start Menu* folder and press the *Install* button.

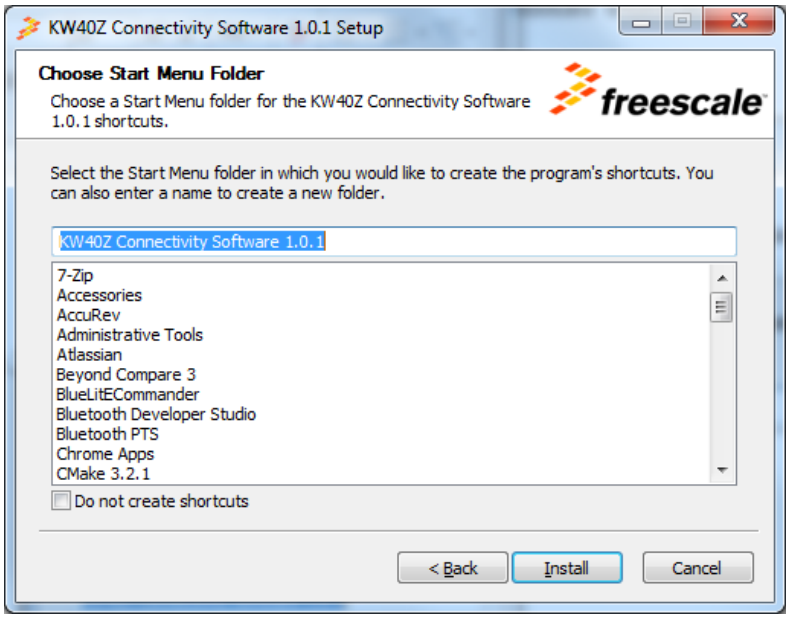

Figure 6: KW40Z installer Start Menu Folder selection screen

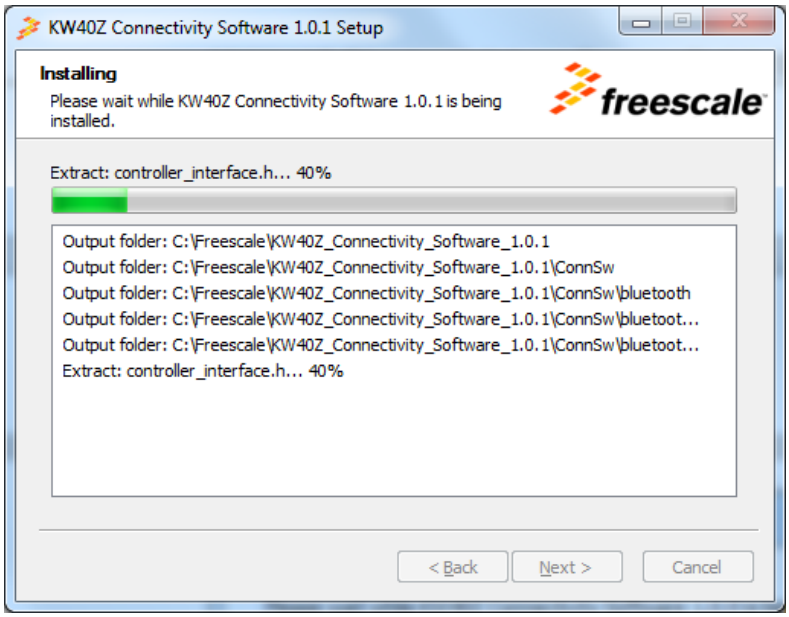

Figure 7: KW40Z installation process

Click *Finish* to close the installer.

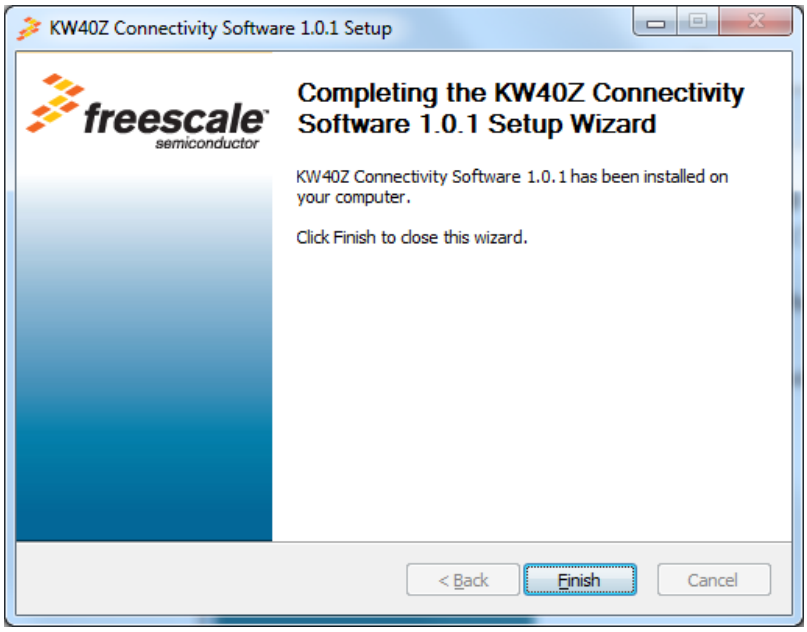

Figure 8: KW40Z Connectivity Software installation complete

The installer automatically creates or updates the *KSDK13_FWK524_PATH* environment variable required by the KW40Z BLE projects. Once the above steps are performed, you can start using the Bluetooth® Low Energy Demo Applications.

3 Cloning a project

Navigate to the KW40Z Connectivity Software installation folder and run the Project Cloner application (**ConnSw\tools\project_cloner\project_cloner.exe**).

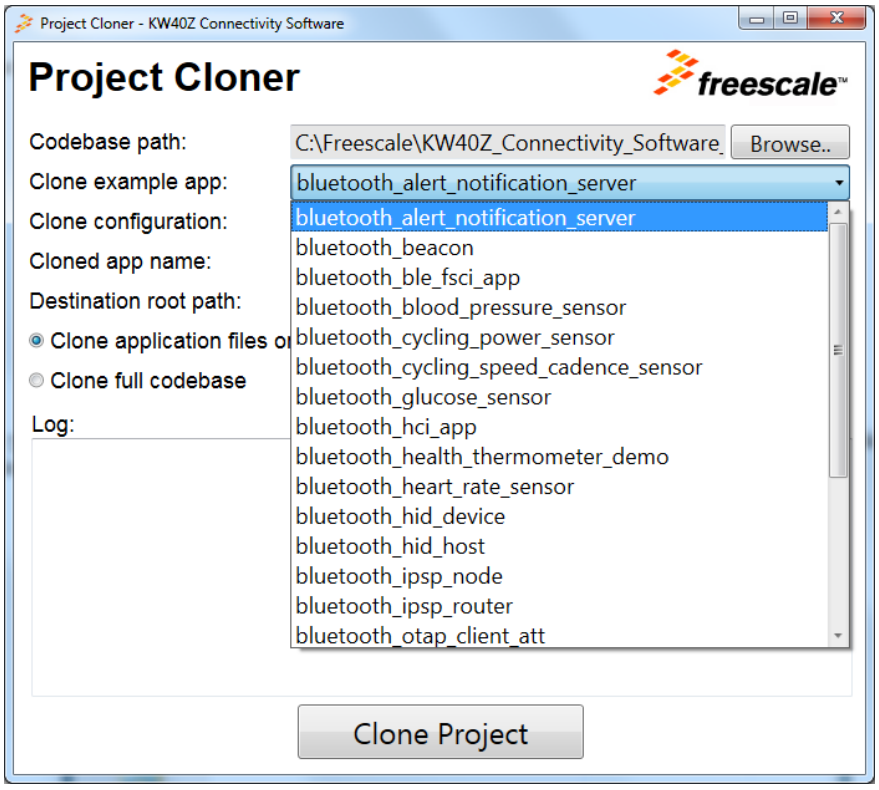

Figure 9: Connectivity Software example applications

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

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 Documents folder of the current user. To change this path click the **Browse** button to select a new location.

By default the Project Cloner will clone only application files (board specific files and example app files). To clone all the files, select the **Clone full codebase** radio button.

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

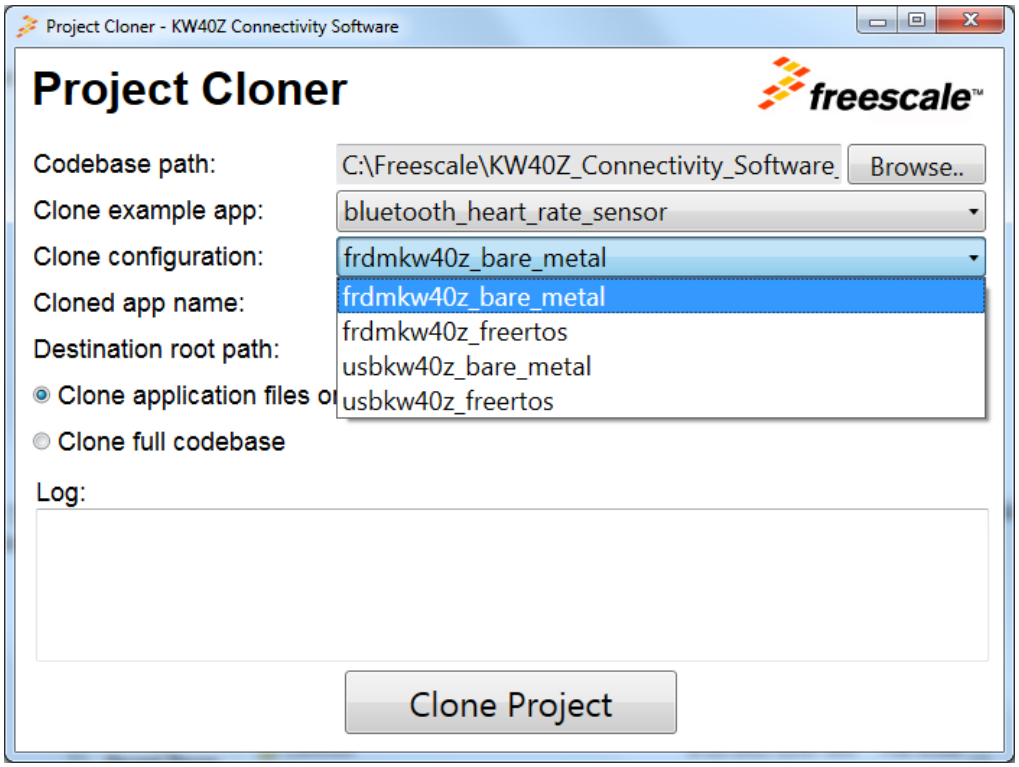

Figure 10: Available configurations for the selected example application

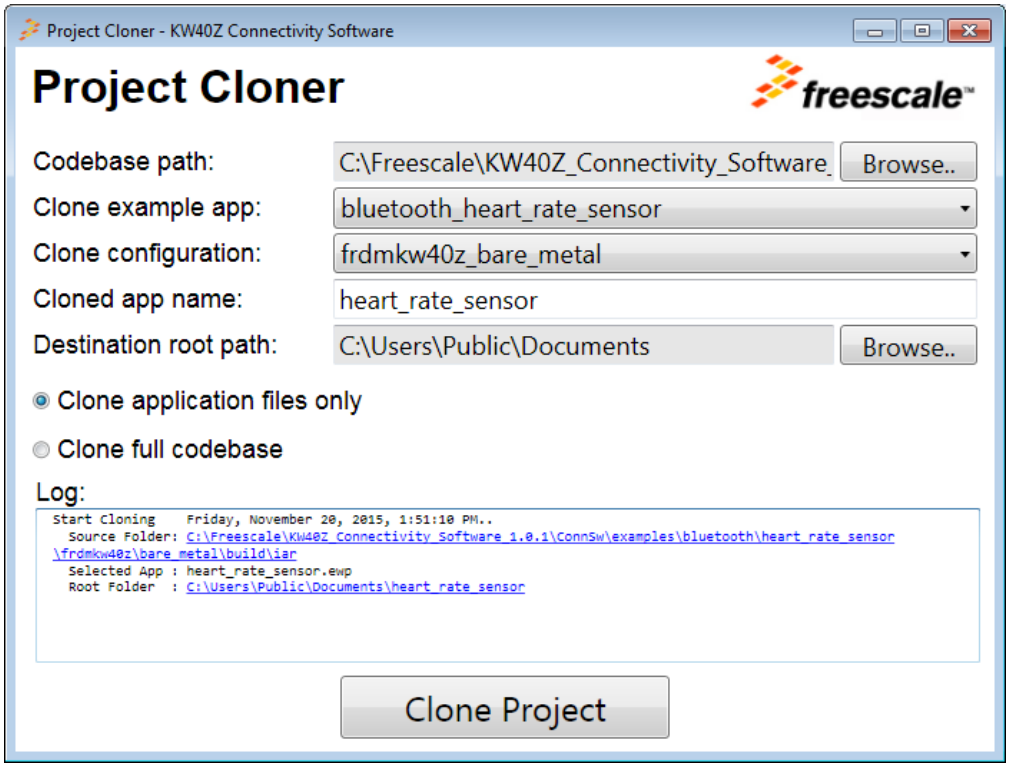

Figure 11: Application cloning done

In the unlikely case that the cloner does not detect the location of the codebase, the following notification will show.

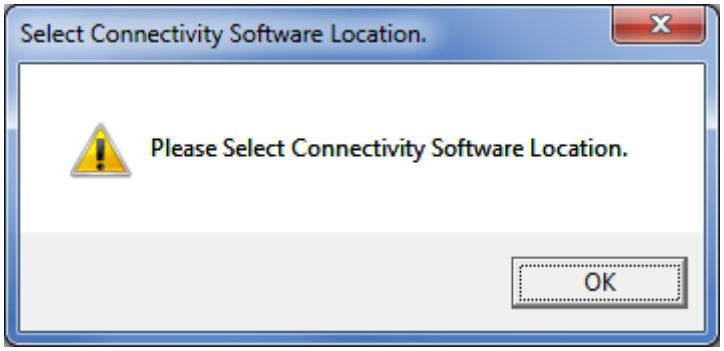

Figure 12: First run time notification

Then you must specify the location of the KW40Z Connectivity Software installation folder. This location can be modified at any time.

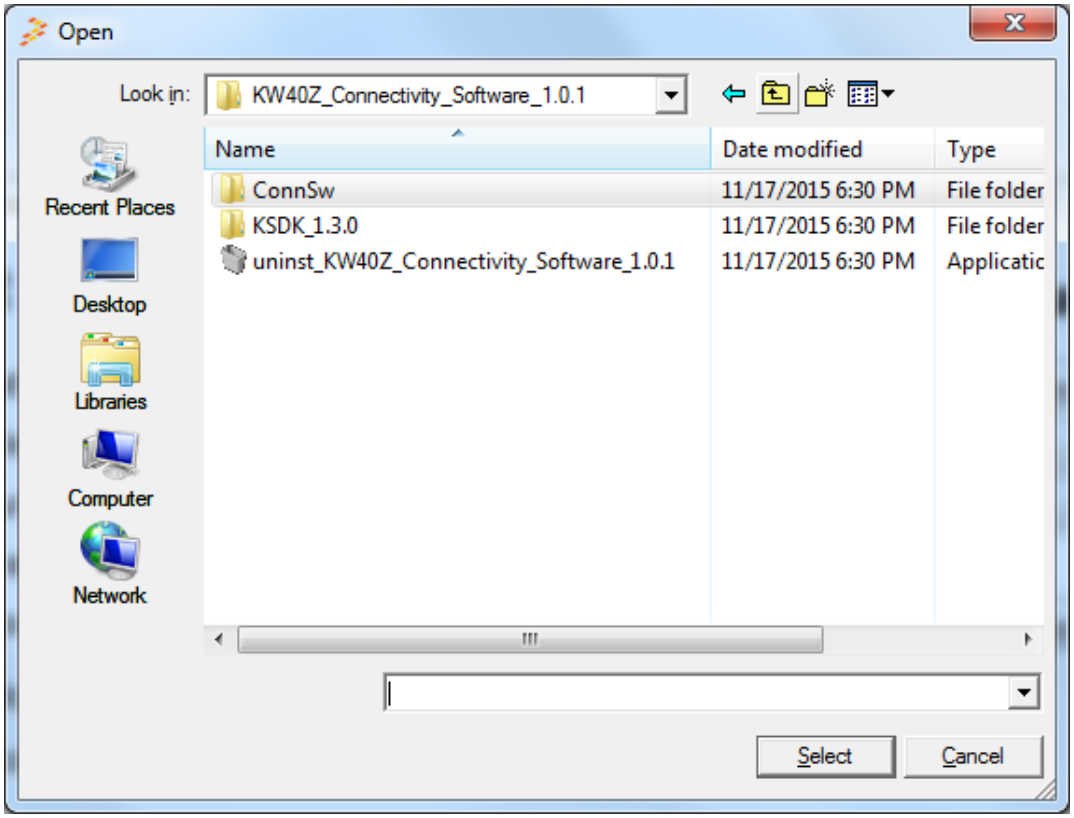

Figure 13: Select the KW40Z Connectivity Software installation path

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

4 Building the Binaries

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

NOTE

In order to be able to build any of these packages you need a copy of the IAR Embedded Workbench for ARM® version 7.40.2 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.

Building the KSDK Libraries

This release supports all development platforms based on the KW40Z wireless microcontroller. The RTOS support includes the FreeRTOS kernel, and a bare metal scheduler. The KSDK platform libraries are RTOS dependent, so appropriate libraries must be built for the selected RTOS.

For any connectivity application, the following Kinetis SDK libraries must be built with the IAR Embedded Workbench for ARM® in order to enable the complete board support and RTOS kernel support:

- FreeRTOS or bare metal Platform drivers library

The location of the KSDK platform projects is described using the following placeholders for text:

- <ksdk_path> : represents the root path of the KSDK installation folder
- <device> : represents the board MCU: KW40Z4
- <board> : represents the board: frdmkw40z

Using the placeholders, these are the required Kinetis SDK v1.3.0 projects locations:

- <ksdk_path>\lib\ksdk_freertos_lib\iar\<device>\ksdk_freertos_lib.eww
- <ksdk_path>\lib\ksdk_platform_lib\iar\<device>\ksdk_platform_lib.eww

NOTE

The IAR projects for KSDK libraries are included in the IAR workspaces corresponding to the BLE demonstration applications and it is recommended to access them this way.

Building and Flashing the Freescale BLE Software Demo Applications

The package contains various demo applications that can be used to get a first feel for the software.

In this section you will be guided through building the Heart Rate Sensor demo application. After you select the configuration you want to clone, locate the destination folder and follow the steps bellow.

NOTE

If your FRDM-KW40Z board is configured for the buck or boost modes of the DCDC converter inside the KW40Z microcontroller, please note that the following defines need to be set: *gDCDC_Enabled_d* to 1 and *APP_DCDC_MODE* to *gDCDC_Mode_Buck_c* or *gDCDC_Mode_Boost_c* respectively, in the *app_preinclude.h* header file.

Freescale BLE Software Demo Application Build Example

Selected app: Heart Rate Sensor

Board: frdmkw40z

RTOS: bare-metal scheduler

Destination root path: C:\Users\Public\Documents\

Resulting location: C:\Users\Public\Documents\heart_rate_sensor\

Step 1:

Navigate to the resulting location.

Step 2:

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

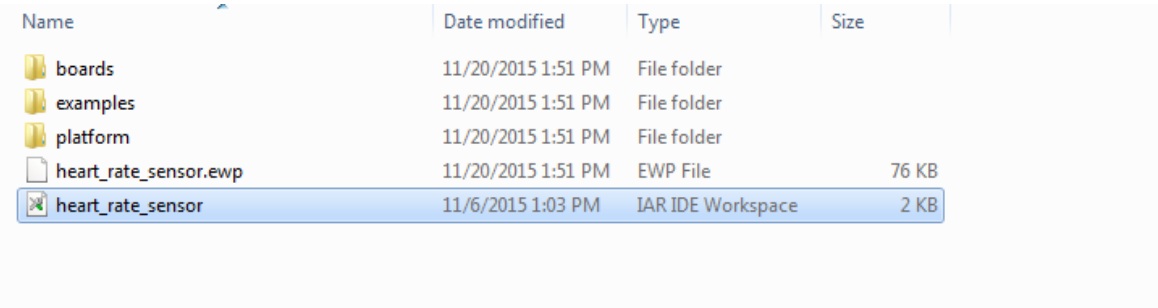

Figure 14: Heart Rate Sensor Demo Application - Project Location

Step 3:

Select the KSDK platform (bare-metal) library project.

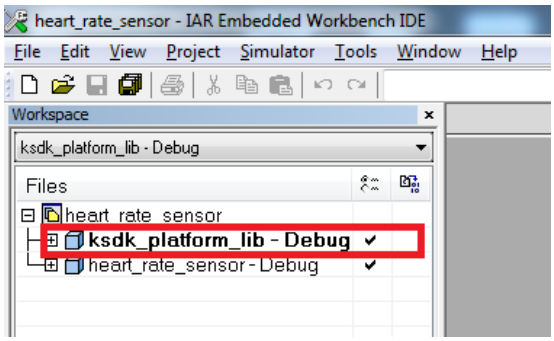

Figure 15: IAR project for KSDK platform (bare-metal) library

Step 4:

Build the KSDK platform library project.

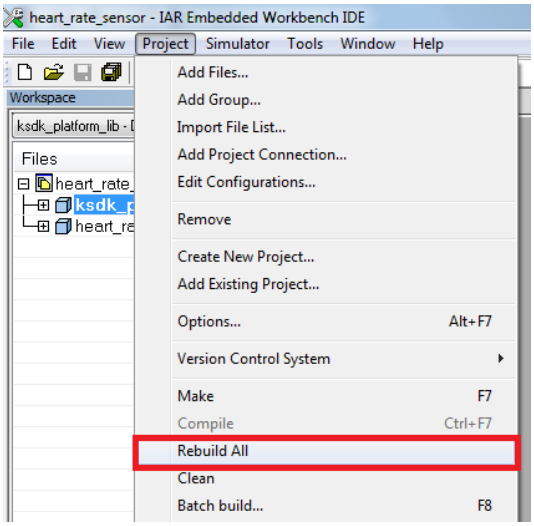

Figure 16: KSDK platform library build

Step 5:

Select the Heart Rate Sensor project.

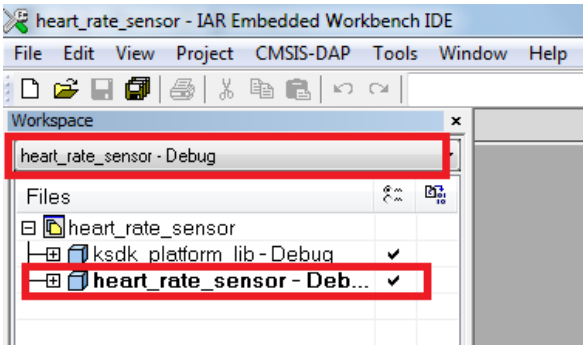

Figure 17: IAR project for Heart Rate Sensor with bare-metal scheduler

Step 6:

Build the Heart Rate Sensor project.

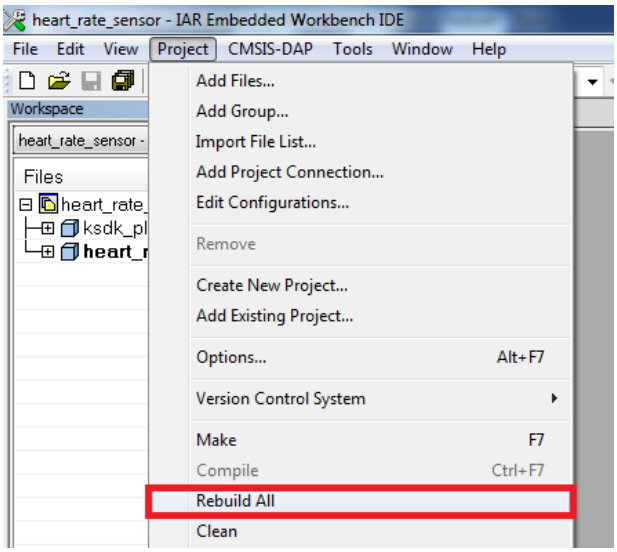

Figure 18: Heart Rate build

Step 7:

Make the appropriate debugger settings in the project options window:

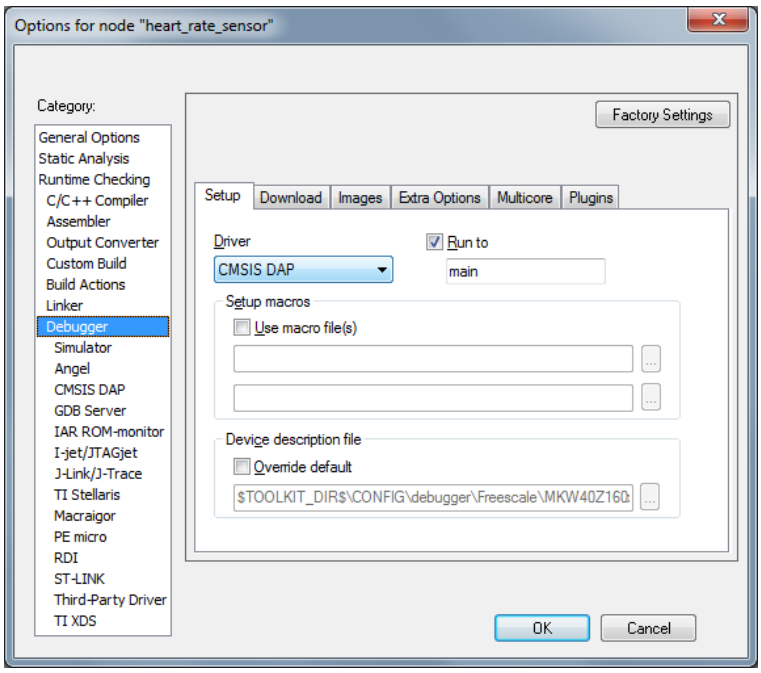

Figure 19: **Debugger Settings**

Step 8:

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

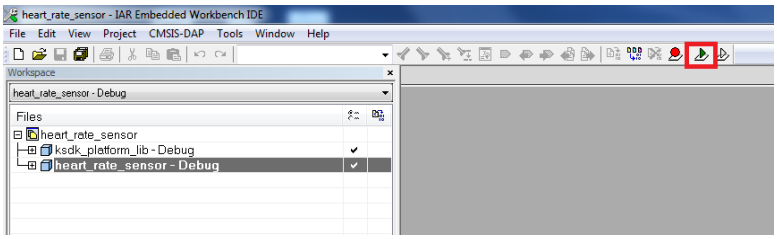

Figure 20: **Heart Rate Sensor Download and Debug**

NOTE

The projects are configured with “CMSIS-DAP” firmware as the default debug configuration. Please make sure that your board’s OpenSDA chip contains a CMSIS-DAP firmware or that the debugger selection corresponds to the physical interface used to connect to the boards. See the section below for more information.

Flashing a Binary Image File Without Using an IDE

The KW40Z connectivity software package contains in the ConnSw\tools\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-KW40Z board, the best approach is to use the OpenSDA on-board interface CMSIS-DAP Mass Storage Device functionality, by simply dragging and dropping the binary image in the mass storage drive exposed by this OpenSDA firmware. For more information, see the CMSIS-DAP firmware github project: https://github.com/mbedmicro/CMSIS-DAP.

In order to flash the firmware on the USB-KW40Z, for either the KW40Z or the K22F silicon on board, a J-Link probe is needed along with the latest J-Link software from www.segger.com.

Run the *jlink.exe* executable provided in the J-Link software installation and type the commands 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.

```
unlock kinetis
device mkw40z160xxx4
loadbin BLE_HRS_frdmkw40z.bin 0
```

5 Hardware Configurations

This section describes how to set up the Freescale Freedom FRDM-KW40 platform for the Bluetooth® Low Energy applications.

Freescale Freedom FRDM-KW40 Platform Introduction

5.1.1 Freescale Freedom FRDM-KW40 Platform Features

The target platform is the FRDM-KW40 board based on the KW40 wireless, dual mode SoC, which incorporates an ARM® Cortex®-M0+ core configured to operate at 32 MHz frequency. It has 160 KB of Flash and 20 KB of RAM. For detailed information about the board, see the appropriate board user's guide.

The board features a composite USB device called OpenSDA which serves as debugger interface and as a USB-to-serial converter via a virtual COM port application. Several firmware images can be programmed on the OpenSDA device, such as these:

- developer.mbed.org/handbook/CMSIS-DAP
- segger.com/opensda.html
- www.pemicro.com/opensda/

5.1.2 Freescale Freedom FRDM-KW40 Platform Overview

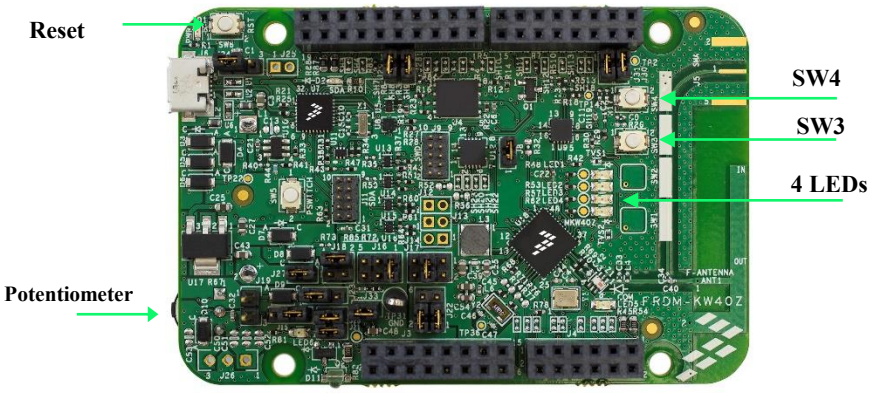

Figure 21: Freescale Freedom FRDM-KW40 platform

The FRDM-KW40Z board can be configured via jumpers to be in the two modes of the DCDC converter inside the KW40Z microcontroller or to bypass it entirely, as shown in the figure below:

Power Configuration

	PWR_CFG J18	PSW_CFG J16	DCDC_CFG J17	REG_CFG J22
Bypass (1.8V - 3.6V)	1-2	1-2	3-4	1-3 2-4
Buck (1.8V - 4.2V) - Coin Cell	2-4	5-6	3-4	3-5
Buck Auto Mode (1.8V - 4.2V)	2-4	3-4	3-4	3-5
Boost (0.9V - 1.8V) - Single battery	2-4	3-4	1-2 5-6	3-5

Figure 22: FRDM-KW40Z Jumper Configuration for DCDC Modes

For more details, please consult the FRDM-KW40Z board schematic.

6 Revision history

This table summarizes revisions to this document.

Revision History		
Revision number	**Date**	**Substantial changes**
0	11/2015	Initial release

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: freescale.com/SalesTermsandConditions.

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: MKW40ZBLESW114QSG
Rev. 0
11/2015