AN14077

Steps for Migrating from KW45x41x8xAxx (1 MB) to KW45x41x5xAxx (512 kB)

Rev. 2 — 27 October 2023 Application note

Document information

Information	Content
Keywords	AN14077, IAR IDE, KW45B41Z, MCUXpresso, flash memory, linker file
Abstract	This document describes the initial steps require to migrate from 1 MB flash to 512 kB flash. The document explains the KW45B41Z demonstrator package and the software configuration of the X-KW45BZ41 board.

1 Introduction

This document describes the initial steps require to migrate from 1 MB flash to 512 kB flash. The document explains the KW45B41Z demonstrator package and the software configuration of the X-KW45BZ41 board.

Note: *It is important to know which KW45 variant you have. This information helps to configure the software to use the flash properly.*

2 Working with SDK example projects

The following sections explain the steps to download and run the software examples.

2.1 Getting the SDK

NXP provides code examples to get you started. For more information, see the MCUXpresso SDK Builder page.

1. Access the MCUXpresso SDK Builder.
2. Log in with your NXP account and click "Select Board/Processor" on the left menu. Select the "KW45B41Z-EVK", as shown in Figure 1.

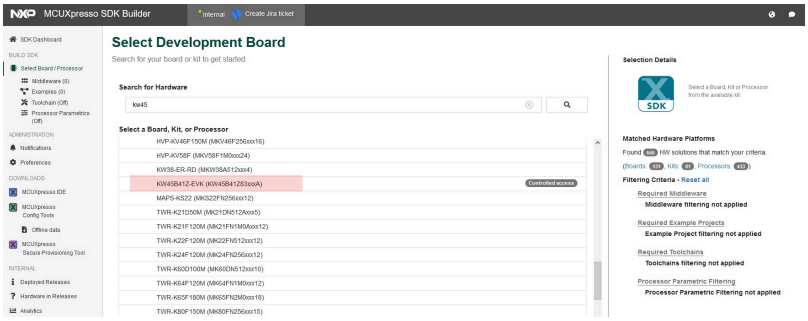

Figure 1. Select development board

3. Click "Build MCUXpresso SDK" on the right menu.

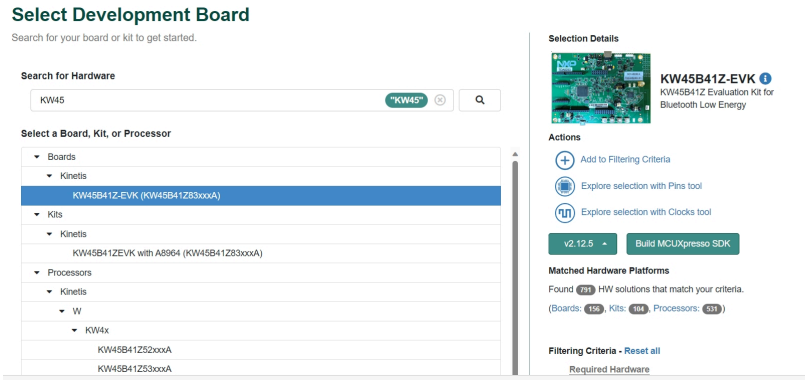

Figure 2. Build MCUXpresso SDK

4. Select the "Host OS" that you are working on and the "Toolchain/IDE" you want to develop. For this case, click "All". This option selects all the toolchains and middleware.
5. Click "Download SDK".

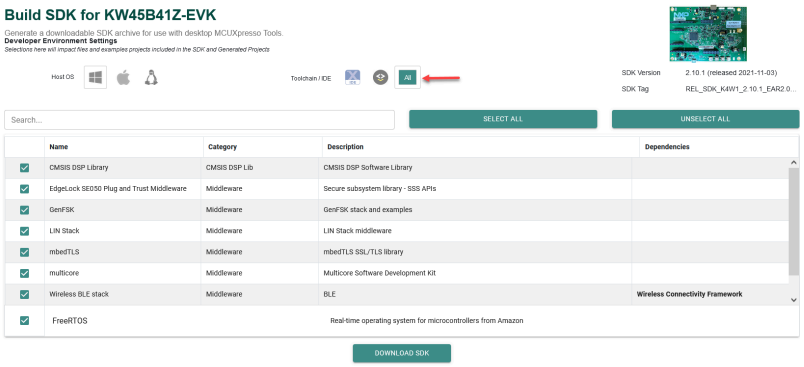

Figure 3. Select the toolchain and download the SDK package

2.2 MCUXpresso software setup

Depending on the KW45 flash, change the linker file; It can be 1 MB flash or 512 kB flash. To change the linker file, click "Project > Properties > MCU settings" and modify the flash size.

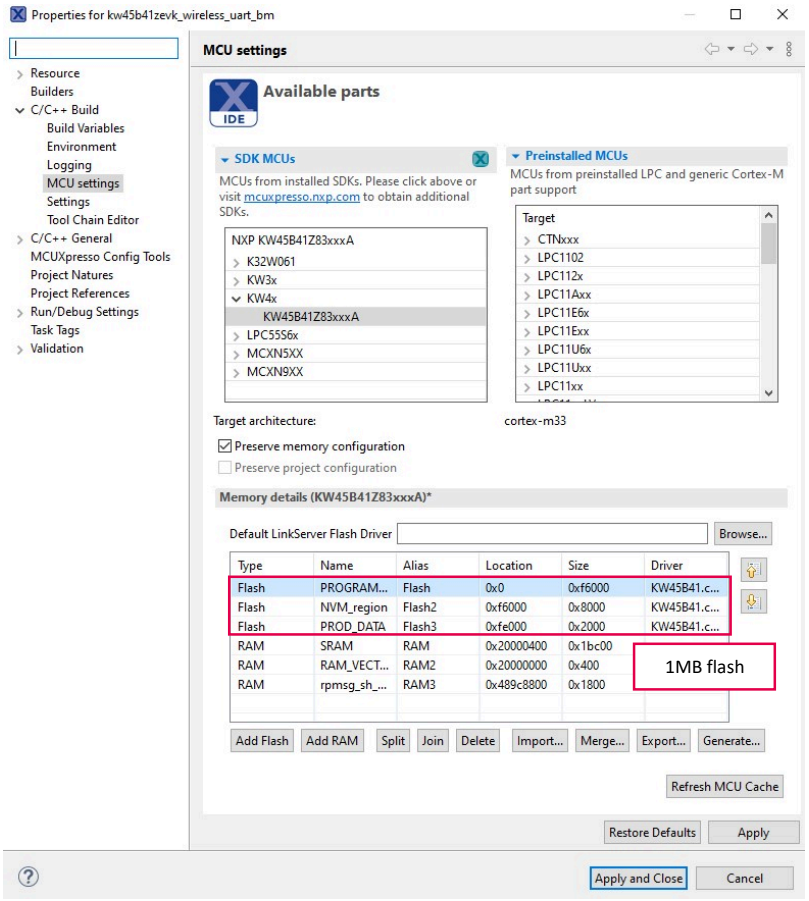

Figure 4. KW45 linker file 1 MB flash

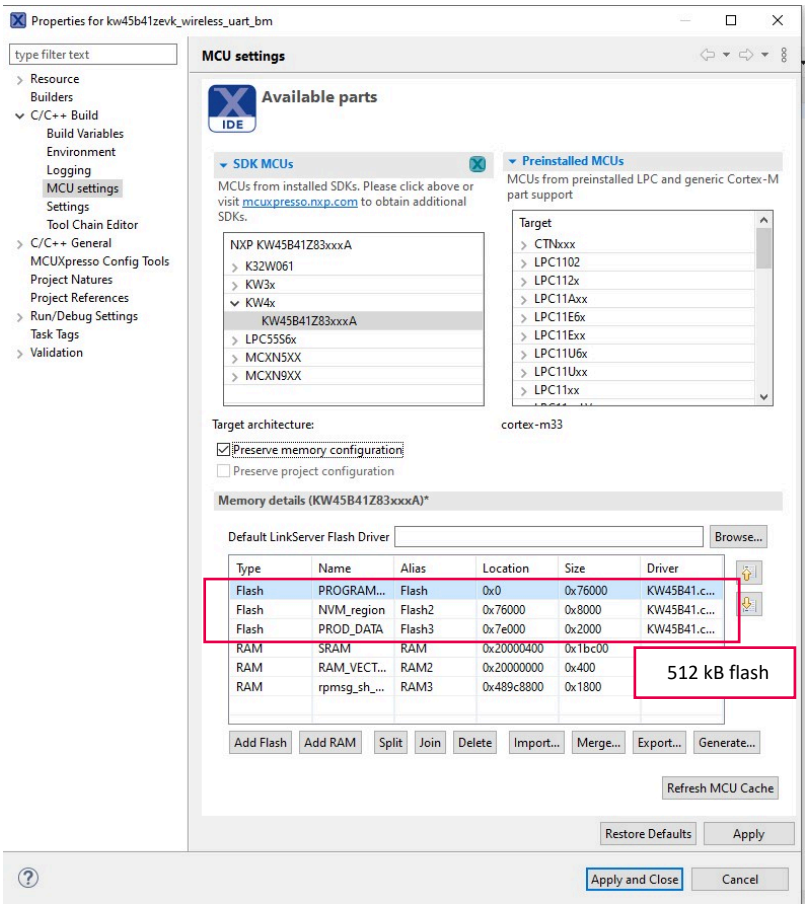

Figure 5. KW45 linker file 512 kB flash

2.3 Running the Hello World

To run the "Hello World" example with the MCUXpresso IDE, perform the following steps:

1. Open the MCUXpresso IDE and create or reuse one of your workspaces.

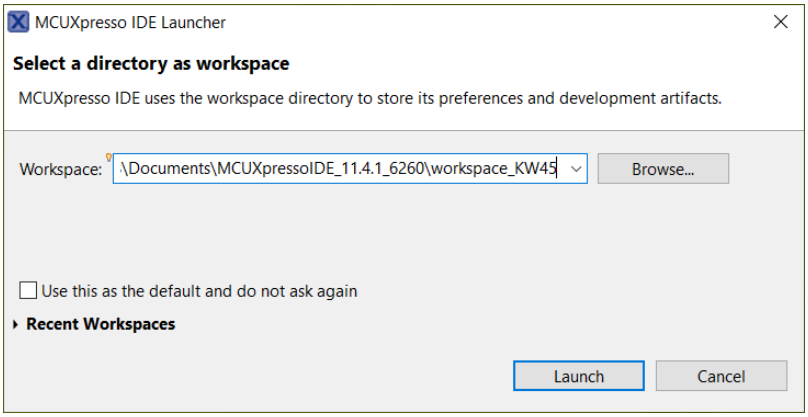

Figure 6. Select your MCUXpresso workspace

2. Select the Installed SDKs tab. Drag and drop the SDK zip file downloaded in the previous step.

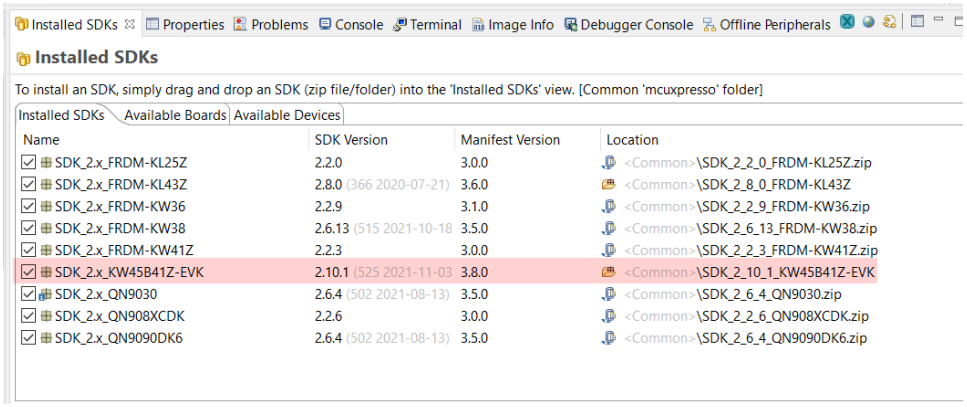

Figure 7. Drag and drop the SDK zip file

3. After completion of SDK installation, select the "Import SDK example(s)" option from the Quickstart Panel.

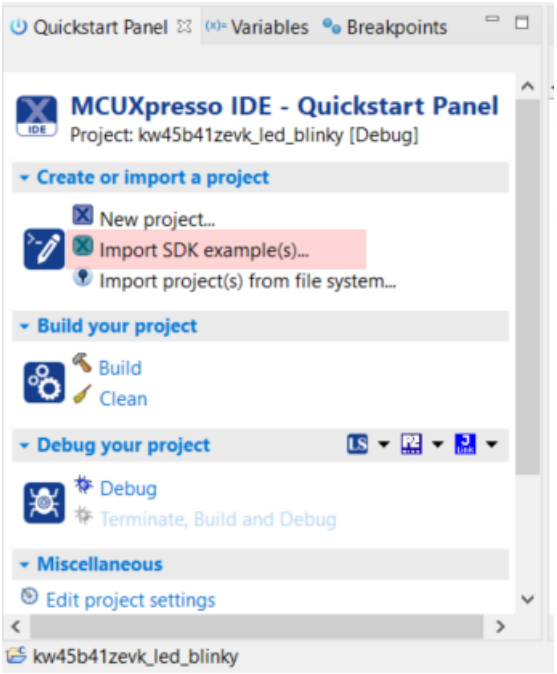

Figure 8. Select the "Import SDK example(s)" option

4. Search for the KW45B41Z-EVK board and click the Next button.

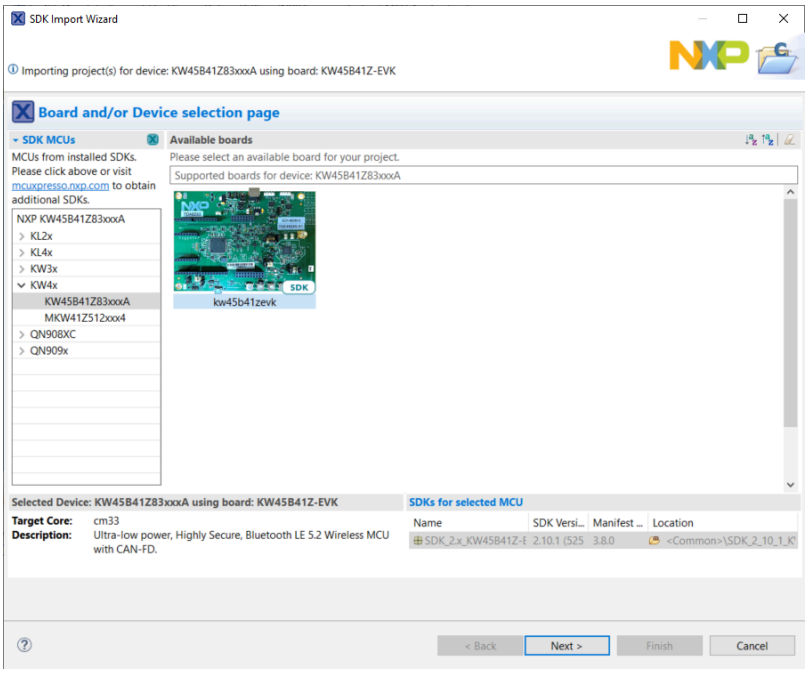

Figure 9. Search and select the KW45B41Z-EVK board

5. Select the example project. For this case, select the "hello_world" example. Select "UART" as the SDK Debug Console under Project Options.

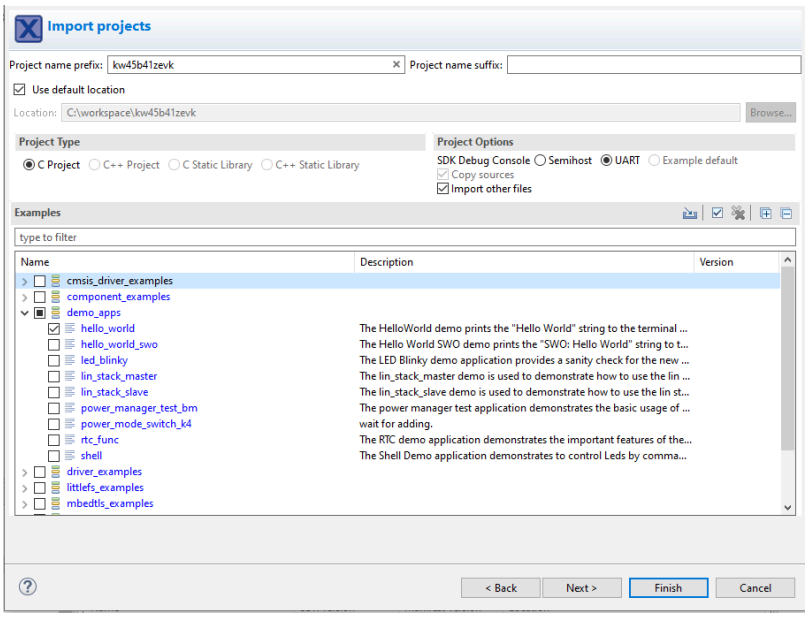

Figure 10. Select the example project

6. The `hello world.c` file is displayed.

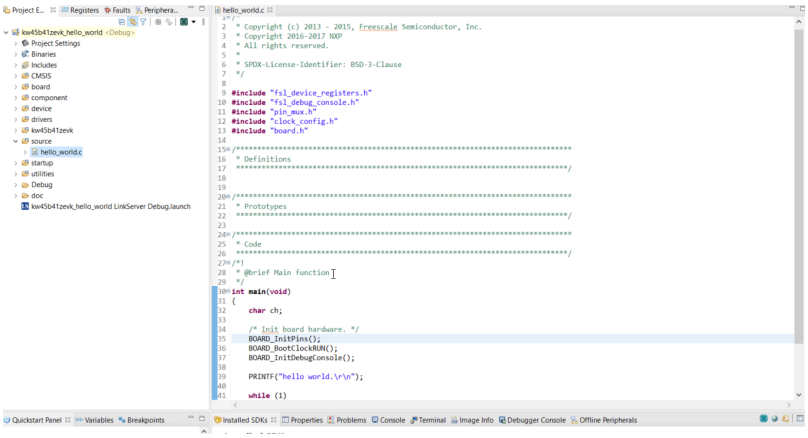

Figure 11. Project file is displayed

7. Compile the example by selecting the hammer icon and select the "Debug build" option.

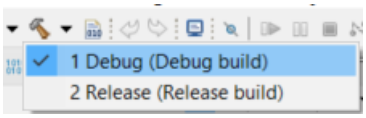

Figure 12. Select the "Debug build" option

8. After a successful compilation, select the "Debug" button. You can find it under the Quickstart Panel.

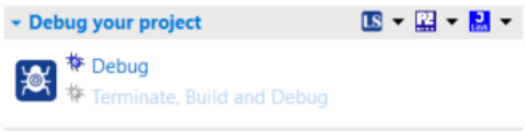

Figure 13. Select the Debug button

9. The MCUXpresso discovers the connected debuggers.

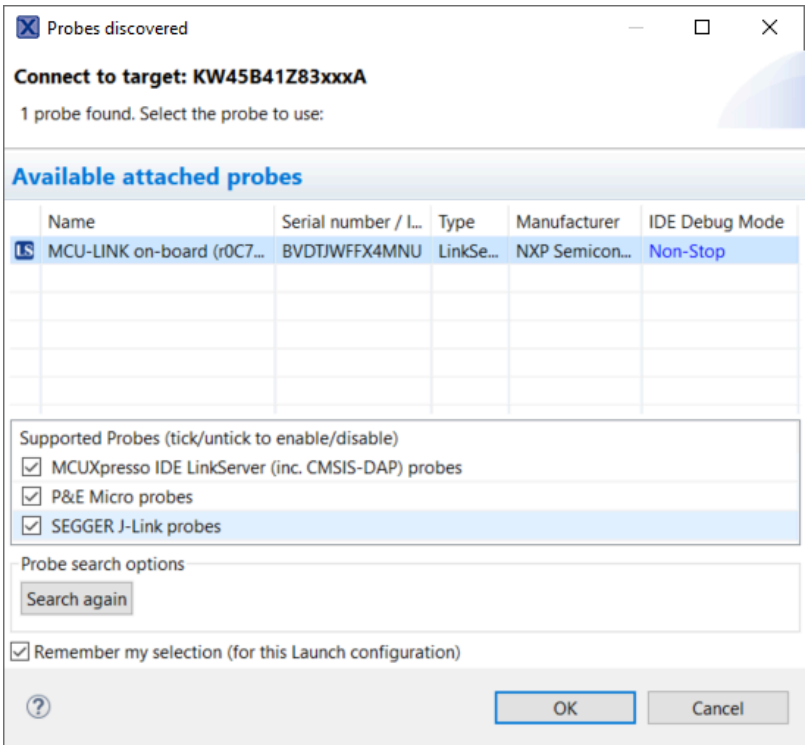

Figure 14. Connected debuggers

10. Enter into Debug mode.

Note: *In the example code, you have several step-by-step options for pause, stop, or run.*

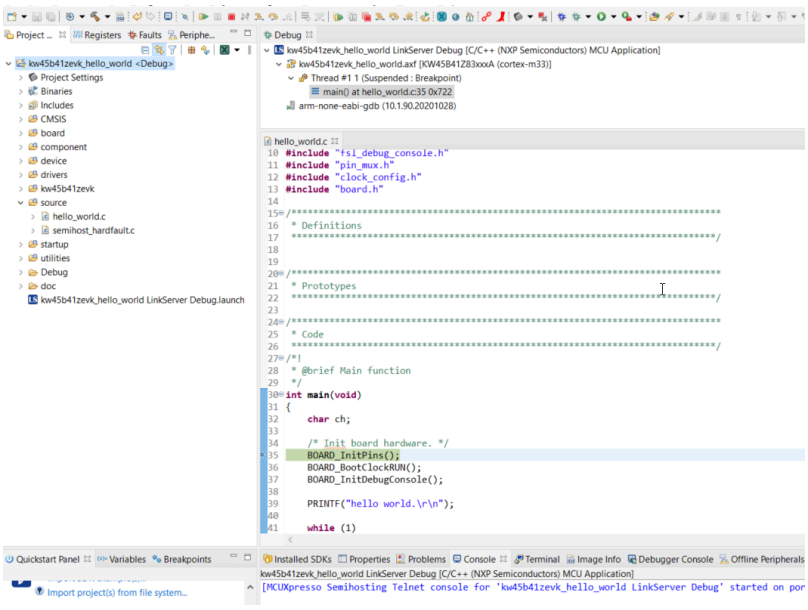

Figure 15. Debug mode

11. Open the terminal software and run the program. A print log appears.

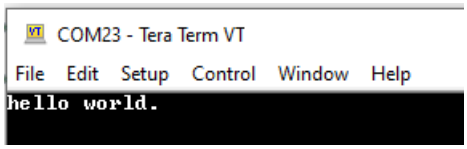

Figure 16. Log

2.4 IAR EW software setup

Depending on the KW45 flash, change the linker file; It can be 1 MB flash or 512 kB flash. This file is available at the following path:

"SDK_2_12_4_KW45B41Z-EVK\boards\kw45b41zevk\wireless_examples\bluetooth\w_uart\bm\iar"

Note: *The folder w-uart is chosen according to the demo that you use.*

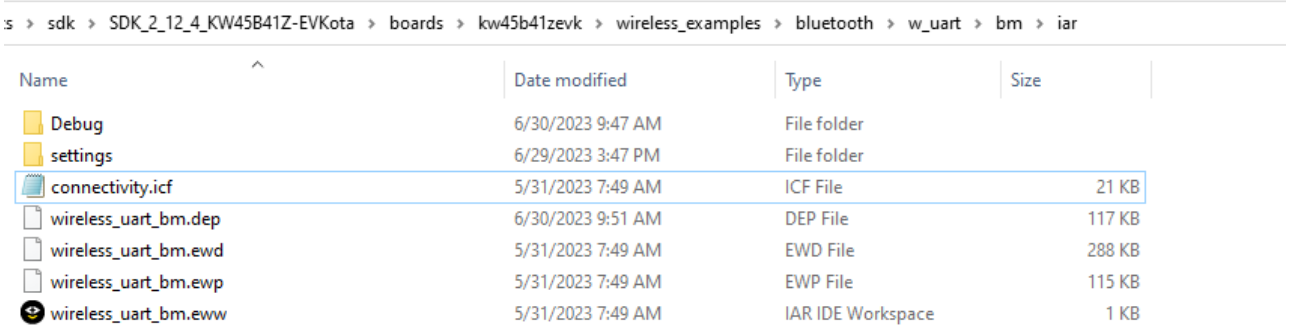

Figure 17. Location for the linker file

Once the linker file is located, ensure to change "m_flash_size".

```
/*********************************************** FLASH ***********************************************/

/* Flash defines */
define symbol m_flash_start                          = 0x00000000;
define symbol m_flash_size                           = 0x100000;
define symbol m_flash_end                            = m_flash_start + m_flash_size - 1;
define symbol m_sector_size                          = 0x2000;
define symbol m_vector_table_size                    = 0x400;   /* define is used for warmboot stack size too */
```

Figure 18. KW45 linker file 1 MB flash

```
/*********************************************** FLASH ***********************************************/

/* Flash defines */
define symbol m_flash_start                          = 0x00000000;
define symbol m_flash_size                           = 0x80000;
define symbol m_flash_end                            = m_flash_start + m_flash_size - 1;
define symbol m_sector_size                          = 0x2000;
define symbol m_vector_table_size                    = 0x400;   /* define is used for warmboot stack size too */
```

Figure 19. KW45 linker file 512 kB flash

To use the linker file, click "Project options > Linker".

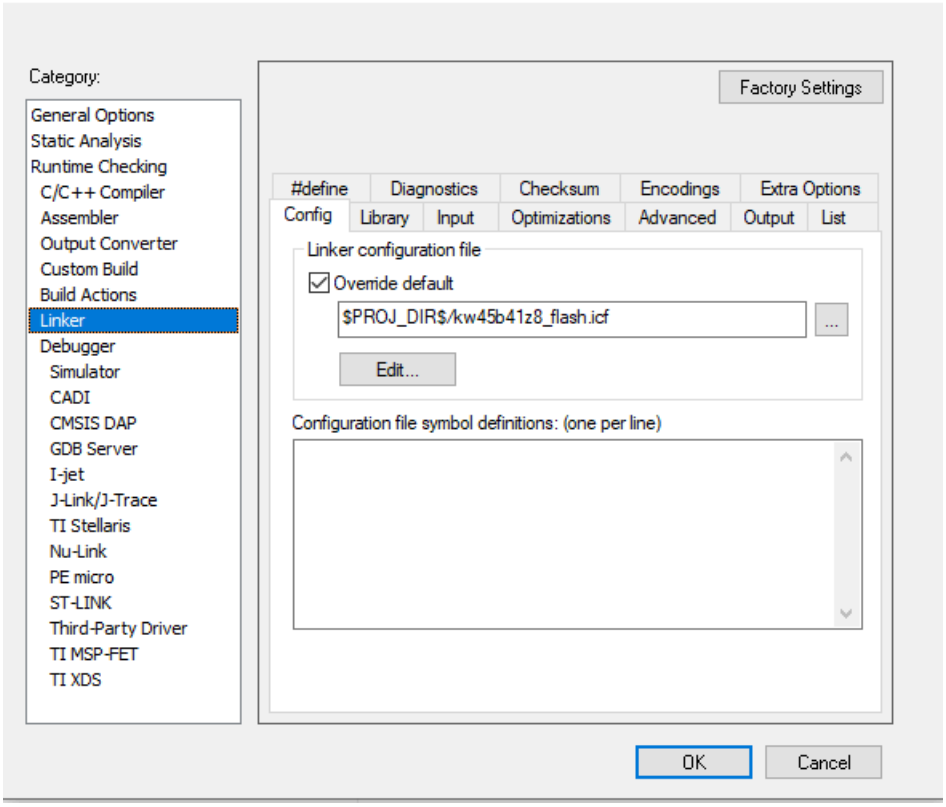

Figure 20. Project options

2.5 Running the wireless UART example

To run the wireless UART example from the KW45 SDK using the IAR IDE, perform the following steps:

1. Open the Wireless UART example at the following path: "`SDK_2_12_4_KW45B41Z-EVK\boards\kw45b41zevk\wireless_examples\bluetooth\w_uart\bm\iar\wireless_uart_bm.eww`"
2. Right-click on the project name and select "Options".

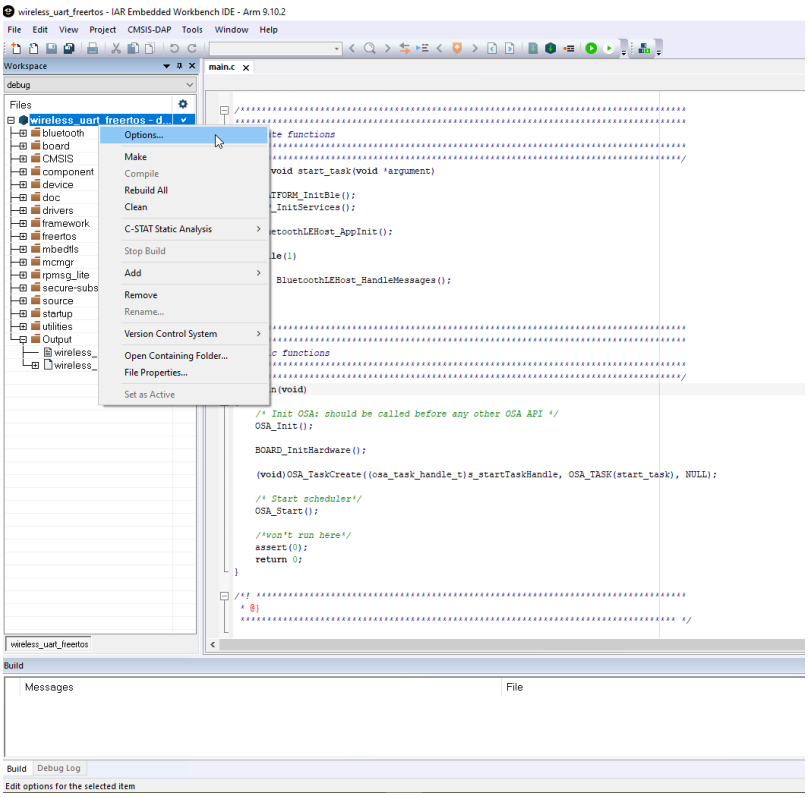

Figure 21. Project options

3. To ensure that the driver is correctly selected according to the probe configuration (CMSIS DAP or J-Link), select the options under "Debugger" category.

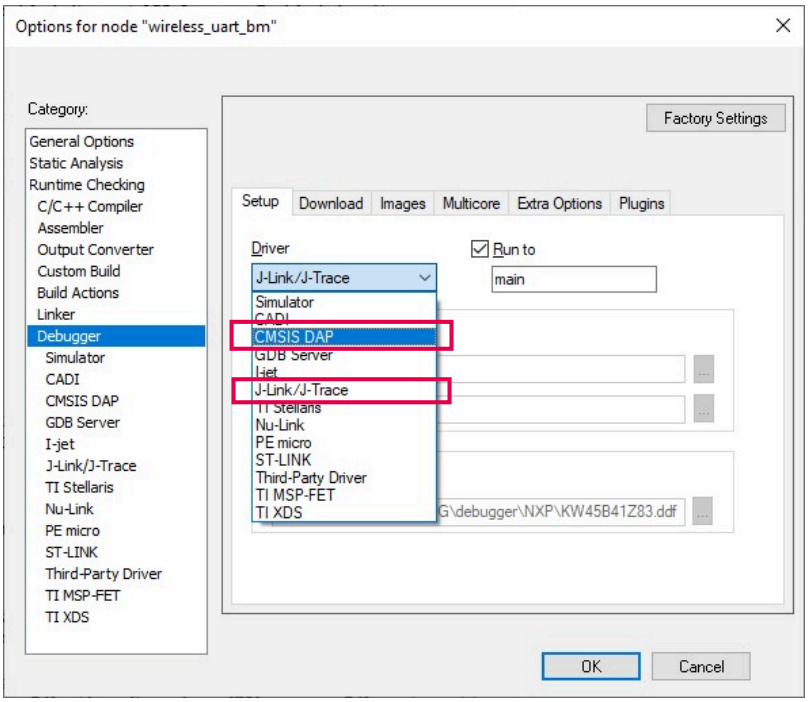

Figure 22. Debugger options

4. Click the start debug button, wait for the compilation process, and enter the Debug mode.

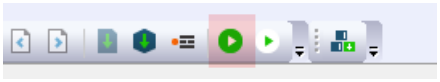

Figure 23. Start debug button

5. Click the run button or press F5 to execute the code.

Figure 24. Run button

6. Open a Serial terminal on your PC and select the correct VCOM when the board has been connected.

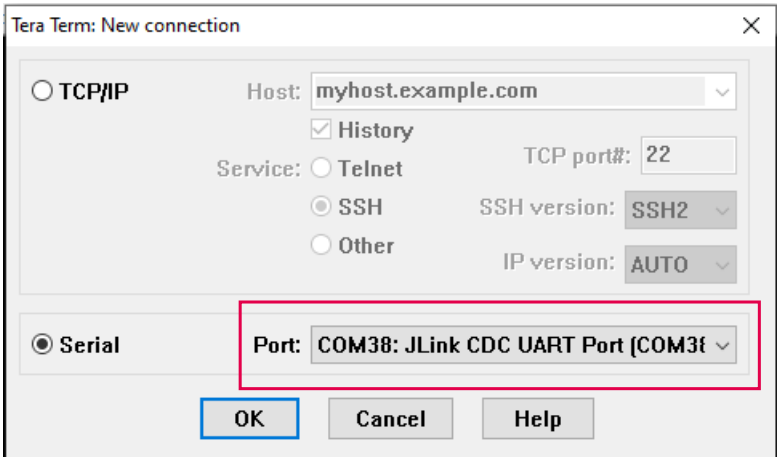

Figure 25. Select the terminal port

7. The board starts as a GAP central. To use this demo with a smartphone, press "SW2" to change it to the GAP peripheral and start advertising.

```
Wireless UART starting as GAP Central, press the role switch to change it.
Switched role to GAP Peripheral.
Advertising...
```

Figure 26. Wireless UART example

8. Open the IoT Toolbox on your smartphone. The toolbox is available at the Google Play Store and Apple App Store.

Figure 27. IoT toolbox

9. Open the Wireless UART application.
10. A device shows up at the scanning window. Tap on it to connect.

Figure 28. Scanning window

11. After establishing the connection, a new line appears on the Serial terminal. Type any message and press enter to send the message to your smartphone.

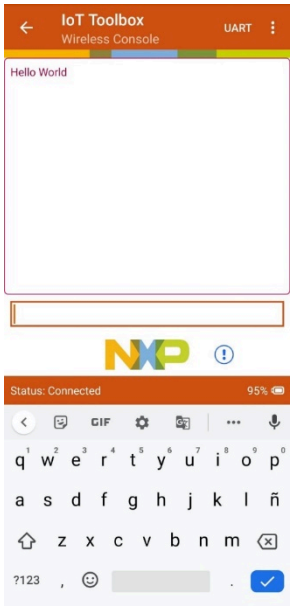

Figure 29. Wireless UART functionality

3 References

Table 1 lists the resources that can be referred for more information.

Table 1. References

Resource	Link/how to access
MCUXpresso	MCUXpresso SDK Builder
KW45B41Z Evaluation Kit	Getting Started with the KW45B41Z Evaluation Kit

4 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2023 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN

ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE

5 Revision history

Table 2 summarizes the revisions to this document.

Table 2. Revision history

Revision number	Release date	Description
2	27 October 2023	• Updated Select the example project in Section 2.3 • Added the Note/disclaimer for the code snippet
1	6 October 2023	Initial public release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

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.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

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.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Apple — is a registered trademark of Apple Inc.

IAR — is a trademark of IAR Systems AB.

J-Link — is a trademark of SEGGER Microcontroller GmbH.

Contents

For more information, please visit: https://www.nxp.com

Date of release: 27 October 2023
Document identifier: AN14077