AN14649

KW47-EVK In-System Programming Utility

Rev. 1.1 — 10 December 2025 Application note

Document information

Information	Content
Keywords	AN14649, KW47, ISP mode
Abstract	This document provides steps to boot the KW47 MCU into ISP mode and establish various serial connections to communicate with the MCU.

1 Introduction

The KW47 Microcontroller Unit (MCU) contains a Read-Only Memory (ROM) bootloader, which is a boot code in the ROM. The ROM bootloader starts to execute when the Cortex-M33 core is released from a reset. The bootloader follows different paths. One of them is the In-System Programming (ISP) path. To make the bootloader follow the ISP path, use the ISP utility. The ISP utility operates over a serial connection on the MCU. To upload/download the application code using the bootloader, use the host-side tools.

This document provides steps to boot the KW47 MCU into ISP mode and establish various serial connections to communicate with the MCU. To demonstrate the ISP functionality, use the KW47-EVK board as the KW47 MCU platform. The KW47 MCU is called the target MCU at some places in the document.

2 Entering ISP mode

The KW47 MCU has two different ways to enter the ISP mode. One uses the designated button on the KW47-EVK board, which asserts the appropriate signal for the ROM bootloader to boot in the ISP mode while powering the board. The KW47 MCU can follow the ISP path using the software method, which you can use while the MCU is running.

2.1 ISP entry via HW

To make the bootloader follow the ISP path (to boot the KW47 MCU into ISP mode), the BOOT_CFG (PTA4) pin of the MCU must be active. To activate this pin and boot the target MCU in the ISP mode, follow these steps:

1. Disconnect the KW47-EVK board from all power sources.
2. Press and hold the SW4 (BOOT_CONFIG) button while connecting the board to the host computer USB port using the J14 USB Type-C connector.
3. Release the SW4 (BOOT_CONFIG) button. The KW47 boots into ISP mode.
4. Connect any external power supply, if needed.

2.2 ISP entry via SW

To enter the ISP mode using software, assert the FORCECFG bit from the CMC[FM] register and perform an SW reset. When the device resets, it enters the ISP mode.

3 Software and tools

This document uses the following tools:

- Use the blhost utility of the Secure Provisioning Software Development Kit (SPSDK) when selecting the Inter-Integrated Circuit (I2C) or Serial Peripheral Interface (SPI) protocols as the communication interface.
- Use the standalone blhost application when selecting the Controller Area Network (CAN) interface as the communication interface, because the current version of SPSDK does not support CAN yet.

Table 1 shows the versions of the tools used in this document.

Table 1. Tool versions

Tool	Version
SPSDK	2.6.0
blhost	2.6.7

3.1 BusPal

BusPal is an embedded software tool available together with the blhost application. It acts as a bus translator between blhost and the target device. It connects to the blhost application over a Universal Asynchronous Receiver/Transmitter (UART) connection, and it connects to the target device over an I2C, SPI, or CAN connection. It helps the blhost application to perform commands and responses from the target device.

The source code for BusPal comes together with the Kinetis bootloader release. The source code is available only for selected platforms, but you can customize it to run on other platforms.

For more details on the BusPal software tool and the MCU bootloader for NXP MCUs, see MCU Bootloader for NXP Microcontrollers.

4 Establishing communication

This section describes how to establish the communication between the host computer and the target MCU using the UART, I2C, SPI, or CAN interface. Use a host-side command-line tool (for example, blhost) to communicate with the target MCU directly over a UART connection. To enable the I2C, SPI, or CAN communication between the host computer and the target MCU, create a bridge (for example, BusPal) using an external device, along with a host-side command-line tool (for example, blhost).

Use one of the following NXP MCUs to create a BusPal bridge:

- MKL25Z
- MKV46
- MK65F
- MCX W71
- MCX A156

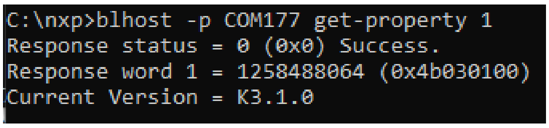
For more information about the BusPal devices and how to program the BusPal firmware, see NXP Application Code Hub.

Note: *The current document includes only examples with the MKL25Z, MCX W71, and MCX A156 MCUs, used as BusPal bridge devices.*

4.1 Using UART interface

To communicate with the target MCU using the UART interface, you do not need any external hardware or modifications. To establish a UART connection between the host computer and the KW47 MCU (KW47-EVK board), boot the MCU in the ISP mode, as explained in Section 2. The target MCU starts to receive ISP commands using the host-side tool.

Figure 1 shows the response from the target MCU after running the ISP command using the UART interface.



```
C:\nxp>blhost -p COM177 get-property 1
Response status = 0 (0x0) Success.
Response word 1 = 1258488064 (0x4b030100)
Current Version = K3.1.0
```

Figure 1. Running ISP command using UART interface

4.2 Using I2C interface

To communicate with the target MCU using the Inter-Integrated Circuit (I2C) interface, create a BusPal bridge (using an external device) between the host computer and the target MCU. You can use devices, such as MCX W71 or MCX A156, as BusPal bridge devices for the I2C communication.

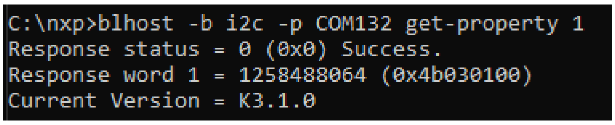
To establish an I2C connection between the host computer and the target MCU using MKL25Z (FRDM-KL25Z board), MCX W71 (FRDM-MCXW71 board), or MCX A156 (FRDM-MCXA156 board) as a bridge device, perform the following steps:

1. Boot the target MCU into ISP, as explained in Section 2.
2. Set up an I2C connection between the KW47-EVK board and the FRDM-KL25Z/FRDM-MCXW71/FRDM-MCXA156 board, as shown in Table 2.

Table 2. I2C connection setup

Signal	Target MCU		MKL25Z BusPal	MCX W71 BusPal	MCX A156 BusPal
	MCU pin	KW47-EVK connector	FRDM-KL25Z connector	FRDM-MCXW71 connector	FRDM-MCXA156 connector
LPI2C_SCL	PTB5	J2, pin 1 / J13, pin 5 / J19, pin 6	J1, pin 14	J2, pin 1	J5, pin 5
LPI2C_SDA	PTB4	J2, pin 3 / J13, pin 6 / J19, pin 8	J1, pin 16	J2, pin 2	J5, pin 6

After following the above steps, the target MCU starts to receive the ISP commands using the host-side tool. Figure 2 shows the response from the target MCU after running the ISP command using the I2C interface.



```
C:\nxp>blhost -b i2c -p COM132 get-property 1
Response status = 0 (0x0) Success.
Response word 1 = 1258488064 (0x4b030100)
Current Version = K3.1.0
```

Figure 2. Running ISP command using I2C interface

4.3 Using SPI

To communicate with the target MCU using the SPI, create a BusPal bridge (using an external device) between the host computer and the target MCU. You can use the MCX W71 or MCX A156 devices as a BusPal bridge for the SPI communication.

To establish an SPI connection between the host computer and the target MCU using MKL25Z (FRDM-KL25Z board), MCX W71 (FRDM-MCXW71 board), or MCX A156 (FRDM-MCXA156 board) as a bridge device, perform the following steps:

1. Boot the target MCU into the ISP mode, as explained in Section 2.
2. Set up an SPI connection between the KW47-EVK board and the FRDM-KL25Z/FRDM-MCXW71/FRDM-MCXA156 board, as shown in Table 3.

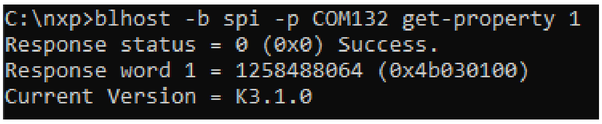
Table 3. SPI connection setup

Signal	Target MCU		MKL25Z BusPal	MCX W71 BusPal	MCX A156 BusPal
	MCU pin	KW47-EVK connector	FRDM-KL25Z connector	FRDM-MCXW71 connector	FRDM-MCXA156 connector
LPSPI_PCS0	PTB0	J2, pin 15 / J12, pin 3	J2, pin 6	J2, pin 8	J6, pin 3
LPSPI_SIN	PTB1	J2, pin 11 / J12, pin 5	J2, pin 8	J2, pin 7	J6, pin 6

Table 3. SPI connection setup*...continued*

Signal	Target MCU		MKL25Z BusPal	MCX W71 BusPal	MCX A156 BusPal
	MCU pin	KW47-EVK connector	FRDM-KL25Z connector	FRDM-MCXW71 connector	FRDM-MCXA156 connector
LPSPI_SCK	PTB2	J2, pin 9 / J12, pin 4	J2, pin 12	J2, pin 5	J6, pin 4
LPSPI_SOUT	PTB3	J2, pin 13 / J12, pin 6	J2, pin 10	J2, pin 6	J6, pin 5

After following the above steps, the target MCU starts receiving ISP commands using the host-side tool. Figure 3 shows the response from the target MCU after running the ISP command using the SPI interface.



```
C:\nxp>blhost -b spi -p COM132 get-property 1
Response status = 0 (0x0) Success.
Response word 1 = 1258488064 (0x4b030100)
Current Version = K3.1.0
```

Figure 3. Running ISP command using SPI interface

4.4 Using CAN interface

To communicate with the target MCU using the CAN, you must create a BusPal bridge (using an external device) between the host computer and the target MCU. For the CAN communication, only MCX W71 and MCX A156 MCUs can be used as BusPal bridge devices.

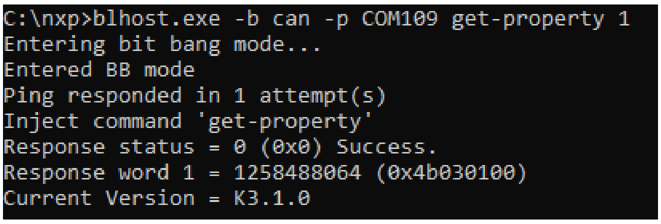
To establish a CAN connection between the host computer and the target MCU while using the MCX W71 (FRDM-MCXW71 board) or MCX A156 (FRDM-MCXA156 board) as a bridge device, follow these steps:

1. Boot the target MCU into ISP, as explained in Section 2. To power on the CAN transceiver in the KW47-EVK board, provide an external 12-V power supply at connector J9.
2. Set up a CAN connection between the KW47-EVK board and the FRDM-MCXW71/FRDM-MCXA156 board, as described in Table 4.

Table 4. CAN connection setup

Signal	Target MCU		MCX W71 BusPal	MCX A156 BusPal
	MCU pin	KW47-EVK connector	FRDM-MCXW71 connector	FRDM-MCXA156 connector
CAN0_TX	PTC4	J10, pin 1	J21, pin 1	J22, pin 2
CAN0_RX	PTC5	J10, pin 2	J21, pin 2	J22, pin 4

After following the above steps, the target MCU starts receiving the ISP commands using the host-side tool. Figure 4 shows the response from the target MCU after running the ISP command using the CAN interface.



```
C:\nxp>blhost.exe -b can -p COM109 get-property 1
Entering bit bang mode...
Entered BB mode
Ping responded in 1 attempt(s)
Inject command 'get-property'
Response status = 0 (0x0) Success.
Response word 1 = 1258488064 (0x4b030100)
Current Version = K3.1.0
```

Figure 4. Running ISP command using CAN interface

5 Revision history

Table 5. Revision history

Document ID	Release date	Description
AN14649 v.1.1	10 December 2025	• Initial public release
AN14649 v.1.0	11 April 2025	• Initial NDA 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.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

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.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Kinetis — is a trademark of NXP B.V.

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.