AN14754

Using MCX E247 QuadSPI Module

Rev. 1.1 — 2 September 2025 Application note

Document information

Information	Content
Keywords	AN14754, MCX E247, QuadSPI
Abstract	This application note describes the QuadSPI module on the MCX E247 devices. It describes how the module is implemented on these devices, specifically focusing on setting up LUT sequences, using commands to interface with an external memory, and using the AHB interface.

1 Introduction

This application note describes the QuadSPI module on the MCX E247 devices. It describes how the module is implemented on these devices, specifically focusing on setting up LUT sequences, using commands to interface with an external memory, and using the AHB interface. For more details about the QuadSPI module, see the device respective reference manual.

The application note is supported by a bare-metal example and an SDK example. The bare-metal example is released on the NXP App Code Hub GitHub repository, while the SDK example is in the SDK release.

2 QuadSPI protocol

The Quad Serial Peripheral Interface (QuadSPI) is a communications protocol used for communications between an MCU and an external flash memory. It is based on the popular Serial Peripheral Interface (SPI). An SPI uses of up to four connections – Data In, Data Out, Clock, and Chip Select (used to signify that a transmit or receive is active). A QuadSPI uses Clock, up to six Chip Select channels, and up to four bidirectional data channels. This extra connectivity allows for data to be read from the flash in a prompt manner, making QuadSPI an excellent choice for using additional off-chip memory. Due to the smaller number of pins, requests for reads/writes/erases are carried out by sending commands across the bus. For example, to read data from a flash memory, the "Read Data (0xEB)" command is sent, followed by the 24-bit address to be read. The data is then sent to the MCU.

In the examples provided with this application note, we use the W26Q64JV serial flash memory manufactured by Winbond, which is present on the FRDM-MCXE247. The flash memory devices manufactured by other vendors work in a similar way.

3 MCX E247 QuadSPI implementation

This section describes several features supported by the QuadSPI module implemented on the MCX E247 series MCU.

3.1 Side A and side B

The QuadSPI module supports two sets of memory interfaces, side A and side B, respectively. Side A and side B have different feature sets:

- Side A: Maximum frequency of 80 MHz, four data lines, no DDR and HyperRAM support
- Side B: Maximum frequency of 20 MHz, eight data lines, DDR and HyperRAM support

Note: *Only one side of the QuadSPI module can be used at the same time.*

4 Look-Up Table (LUT)

This section describes the Look-Up Table (LUT).

4.1 LUT sequence

The Look-Up Table (LUT) is the mechanism used by the QuadSPI module to communicate with the external memory. Using a set of instructions, The QuadSPI can execute multiple types of memory commands such as read, write, erase, or waiting, according to user-programmable LUT sequences. The QuadSPI contains a special memory region for the LUT sequence storage and this LUT memory region can hold up to 16 LUT sequences.

Each LUT instruction is 16 bits in length, each LUT sequence contains up to eight instructions, and memory commands launched either by an AHB access or an IP command trigger the execution of the corresponding LUT sequence configured by the user.

4.2 Programming LUT sequence

This section provides an example to implement a flash read command using LUT sequences.

4.2.1 LUT lock mechanism

As a safety mechanism, unlock the LUT before programming new sequences.

To unlock the LUT region, perform the following operations in a sequence:

- Write 0x5AF05AF0 to the LUTKEY register.
- Write 0x2 to the LCKCR register.
- Poll the UNLOCK bit of the LCKCR register until this bit is set by the hardware.

After all the LUT operations are completed, you can lock the LUT again to prevent accidental changes using the following sequence:

- Write 0x5AF05AF0 to the LUTKEY register.
- Write 0x1 to the LCKCR register.
- Poll the LOCK bit of the LCKCR register until this bit is set by the hardware.

4.2.2 Constructing LUT sequence

An LUT instruction consists of three parts:

- Instruction (Opcode): Indicates the command type.
- Pads: The number of data lines used to carry out the instruction.
- Operand: Command parameters. Refer to user manual for details.

The organization of the LUT instructions is shown in Figure 1.

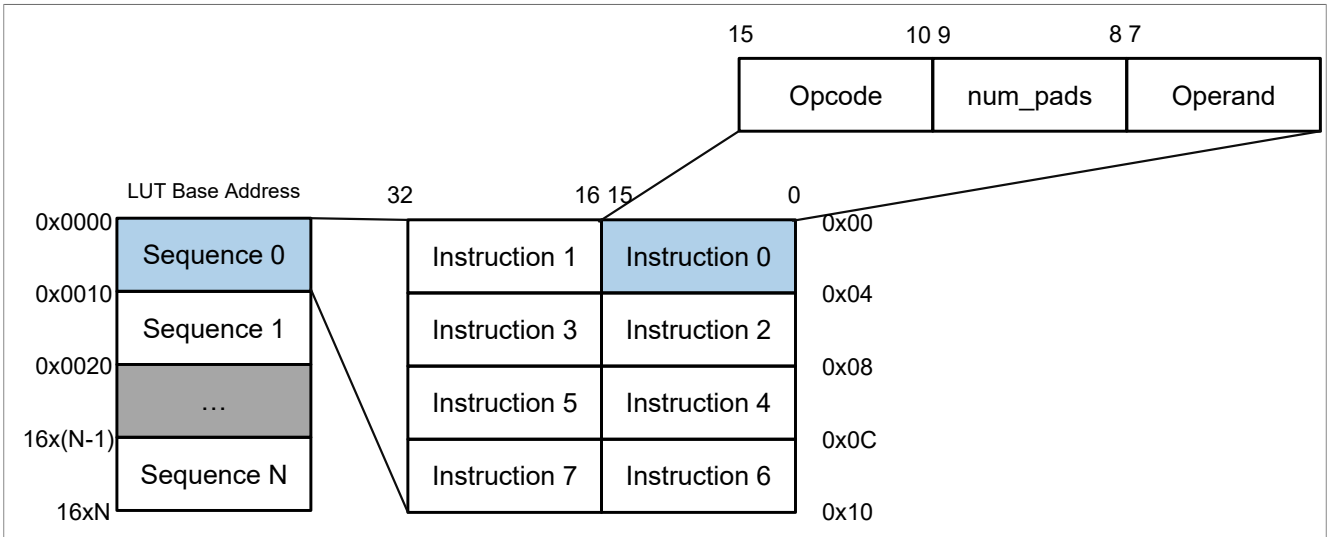


Figure 1. LUT sequence and instructions

4.2.3 Flash read example sequence

In this section, we use the "Fast Read Quad I/O" instruction according to the data sheet to access an external flash device. This instruction contains the following LUT instructions:

- Issue a "Fast Read Quad I/O" (0xEB) instruction using one data line.
- Write a 24-bit address using four data lines.
- Write an alternate byte (0xF0) using four data lines.
- Wait for four dummy cycles.
- Read the data using four data lines.

This constructs the following sequence (using an SDK macro):

```
LUT[0] = QSPI_LUT_SEQ(QSPI_CMD, QSPI_PAD_1, 0xEB, QSPI_ADDR, QSPI_PAD_4, 0x18)
LUT[1] = QSPI_LUT_SEQ(QSPI_MODE, QSPI_PAD_4, 0xF0, QSPI_DUMMY, QSPI_PAD_4, 0x04)
LUT[2] = QSPI_LUT_SEQ(QSPI_READ, QSPI_PAD_4, 0x80, QSPI_STOP, QSPI_PAD_1, 0x00)
```

Note: *The last register contains a special STOP command, which is required to fill the unused spaces and instruct the QuadSPI sequence engine to stop fetching the next instruction.*

5 AHB command operation

The QuadSPI peripheral can map the contents of an external memory device to the system AHB bus, so the processor or other bus masters can fetch data directly from the AHB bus, similar to the internal memory space.

To achieve this, point the corresponding LUT sequence in the BFGENCR[SEQID] register field and the QuadSPI automatically translates the AHB access requests to the LUT sequence.

6 IP command operation

You can use the IP command to execute indirect operations, such as flash programming, erasing, or configuring commands. These commands are triggered by IPCR registers, while TX and RX FIFOs are used for writing and reading of the external memory devices.

Program the IP command address to the SFAR register, program the data length to IPCR[IDATSZ], and launch the IP command by programming the IPCR[SEQID].

Note: *Be careful on the FIFO overrun/underrun condition to ensure correct operation of the external memory device. For example, prefill the TX FIFO before launching a program page command by writing to IPCR[SEQID]. After the command launches, fill the TX FIFO constantly until all the data specified by IPCR[IDATSZ] is written to the TX FIFO.*

7 Setting up and running demo on FRDM-MCXE247

This section describes how to set up the FRDM-MCXE247 board to run the SDK QuadSPI example and demo from the NXP Application Code Hub.

7.1 Hardware preparation

The FRDM-MCXE247 is configured for the onboard Ethernet by default. To use the board for the QuadSPI module, perform the following hardware modifications:

On the FRDM-MCXE247 board:

- Remove 0-Ω resistors R164 and R165.

- Install 0-Ω resistors R120, R122, R124, R126, R129, and R130.

Note: *There is no onboard level shifter for QuadSPI signals. After a hardware modification, the voltage selection jumper J10* ***must be installed on the 3.3-V side*** *to prevent damage to the serial flash device.*

8 Running SDK example on FRDM-MCXE247

This section describes how to run the SDK example on FRDM-MCXE247.

8.1 Software requirements

The following software is required:

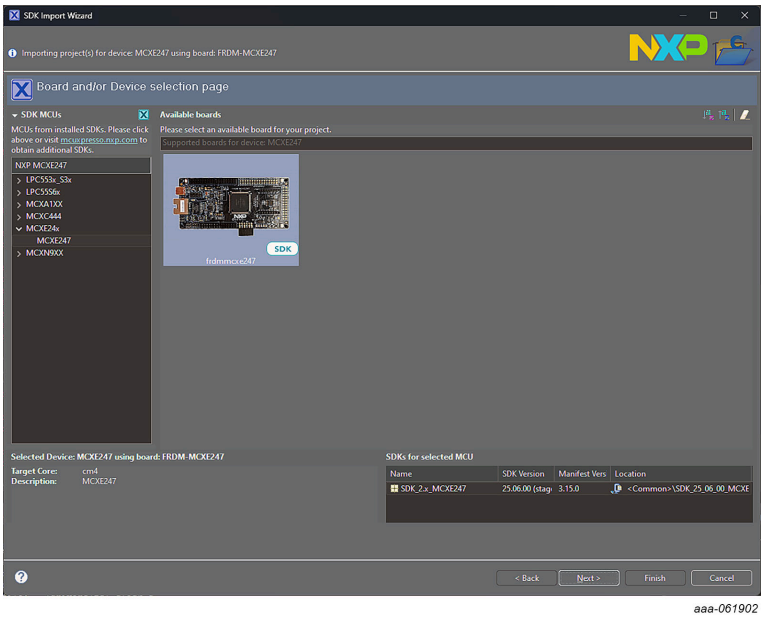
- MCUXpresso IDE
- MCUXpresso SDK

Visit https://mcuxpresso.nxp.com to get the software dependencies and SDK packages.

8.2 Creating SDK example project

After you install the MCUXpresso IDE and import the SDK, perform the following steps:

1. Use the "File-New-Import SDK example(s)…" menu bar item to launch the SDK importer.
2. In the left-hand-side navigation pane, select "FRDM-MCXE247" and click "Next".
3. Navigate to the "driver_examples/qspi" option in the "Examples" list, check the "qspi_polling_transfer" checkbox, and click "Finish".



aaa-061902

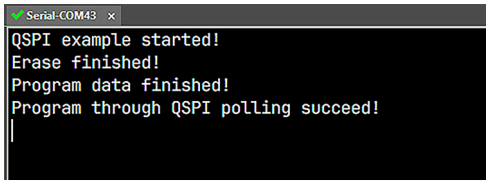
Figure 2. Creating example project for FRDM-MCXE247

8.3 Building and running SDK example project

After successfully creating the SDK example project, perform the following steps:

1. Click the "Build" button in the toolbar menu.
2. Connect the board to the PC using a USB-C cable.

3. Click the "GUI Flash Tool" button in the toolbar menu to download the project to the board.
4. Open a serial monitor tool and connect to the board serial interface using the following parameters:
   a. Baud rate: 115200, eight data bits, one stop bit, no parity bit, no flow control.
5. Reset the board by pressing the SW1 button.
6. Find the message printed through the UART interface.



aaa-061903

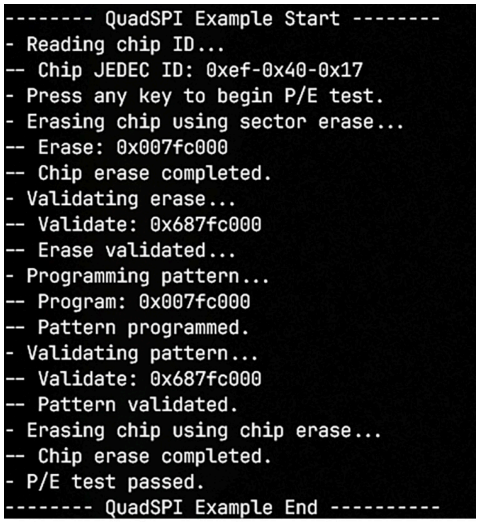
Figure 3. UART output for SDK example

8.4 Creating Application Code Hub demo project

1. Select "Import from Application Code Hub" from "Quickstart Panel" and search for the "MCX E247 QuadSPI" demo in the search box.
2. Select the project card and click "Copy GitHub Link" to import the demo project.

8.5 Building and running Application Code Hub demo project

1. Follow the same procedure as for the SDK example project and find the message printed through the UART interface.
2. Press any key to start the Program/Erase (P/E) test.



aaa-061904

Figure 4. UART output for SDK example

9 Revision history

Table 1. Revision history

Document ID	Release date	Description
AN14754 v.1.1	2 September 2025	• Updated figures

Table 1. Revision history*...continued*

Document ID	Release date	Description
AN14754 v.1.0	22 July 2025	• Initial version

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.

Contents