

# AN13898

## OpenSDA on PN76 development boards

Rev. 1.0 — 13 March 2023

Application note

### Document information

Information	Content
Keywords	OpenSDA, PN76, development boards, SEGGER, debugging
Abstract	This document explains how to enable the OpenSDA interface for debugging on the PN76 development boards.



## Revision history

---

### Revision history

Rev	Date	Description
v.1.0	20230313	Initial version

## 1 Introduction

This document explains what OpenSDA is, what you can do with it and how to enable it on the PN76 family development boards. There are hardware changes involved and the OpenSDA microcontroller must be programmed.

- Due to the hardware changes, the JTAG/SWD connector "J17" is rendered useless. To use this connection again, you have to undo any changes.

The board used in this document is the PNEV7642A Rev.B without a preloaded K20.

**Note:** In future the K20 might come already preprogrammed with the OpenSDA application. This can be easily identified by plugging in a USB-C cable to J27 and check your device manager for an J-Link driver. If such devices appear, you do not have to undergo any programming steps to, and only need to prepare the hardware according to [Section 2](#).

### 1.1 OpenSDA

OpenSDA is an open-standard serial debug adapter offered by SEGGER [1]. It bridges serial and debug communications between a USB host and an embedded target processor. The hardware circuit is based on an NXP Kinetis K20 family microcontroller (MCU) with 128 kB of embedded flash and an integrated USB controller.

The host interface supports two application modes:

- **Mass storage device (MSD):** Allows easily programming the applications into the flash of the target processor by dragging the new binary onto the drive.
- **Debug Application:** Provides an interface to Integrated Development Environment (IDE) software, allowing the debug of an application directly from the OpenSDA interface.

Both application modes support the USB Communications Device Class (USB CDC Interface) that bridges serial communications between the USB host and a UART serial interface on the target processor.

In order to switch between these application modes, the OpenSDA interface processor must be started in bootloader mode and the corresponding OpenSDA application firmware has to be dragged and dropped into the drive.

## 2 Prepare hardware

Following hardware changes must be applied:

- Remove R70 and populate R69
  - This switches the SWD connection from the JTAG connector J17 to the OpenSDA block onboard.
- Populate R109 and R110
  - Connects the SWD from OpenSDA to the SWD connection of the board.
- Make sure that J31 and J32 are populated (default)

Following picture shows the location of the resistors which have to be resoldered:

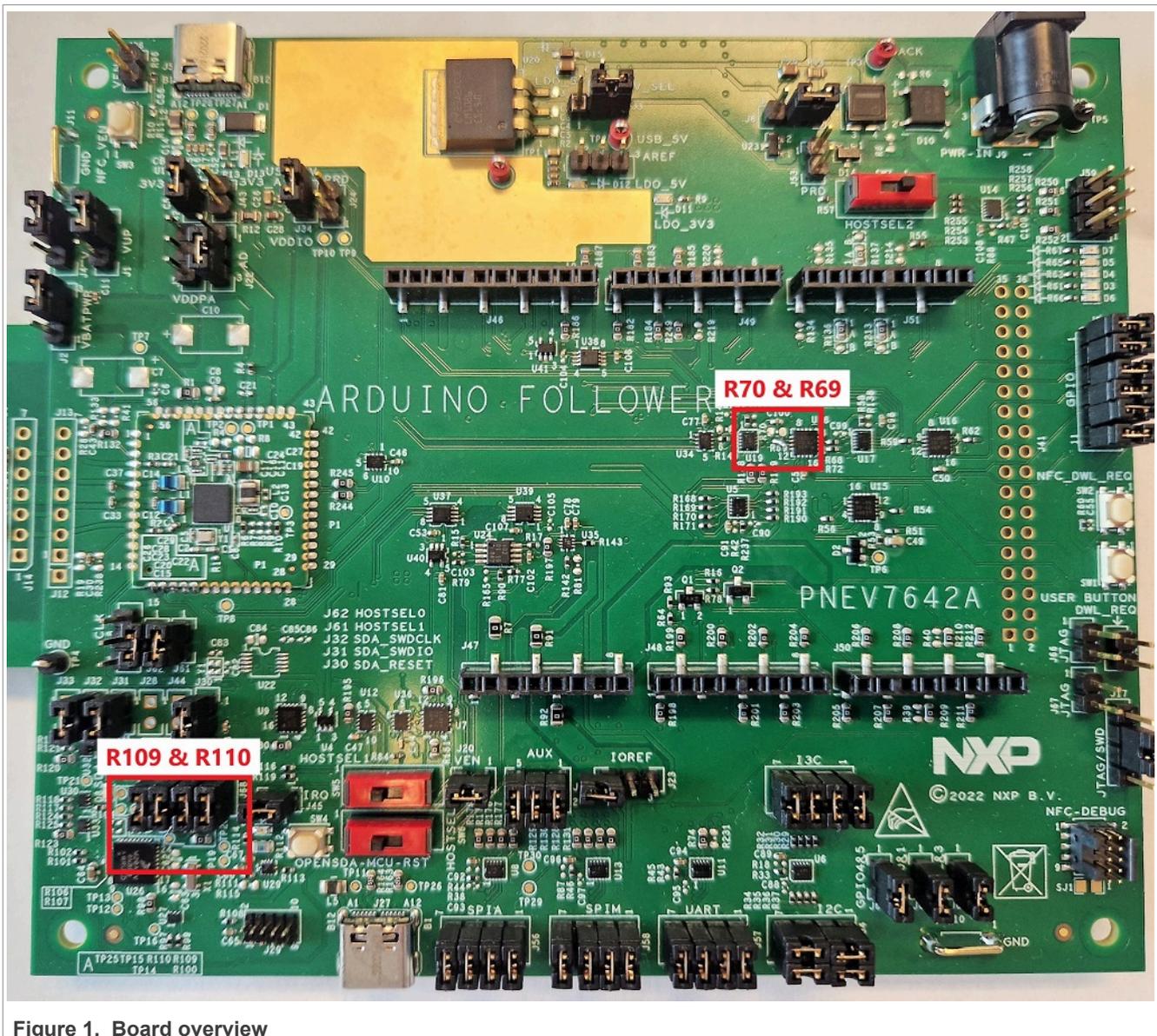


Figure 1. Board overview

### 2.1 R69 and R70

The resistor R70 must be removed and the resistor R69 must be populated with a 0 ohm resistor.

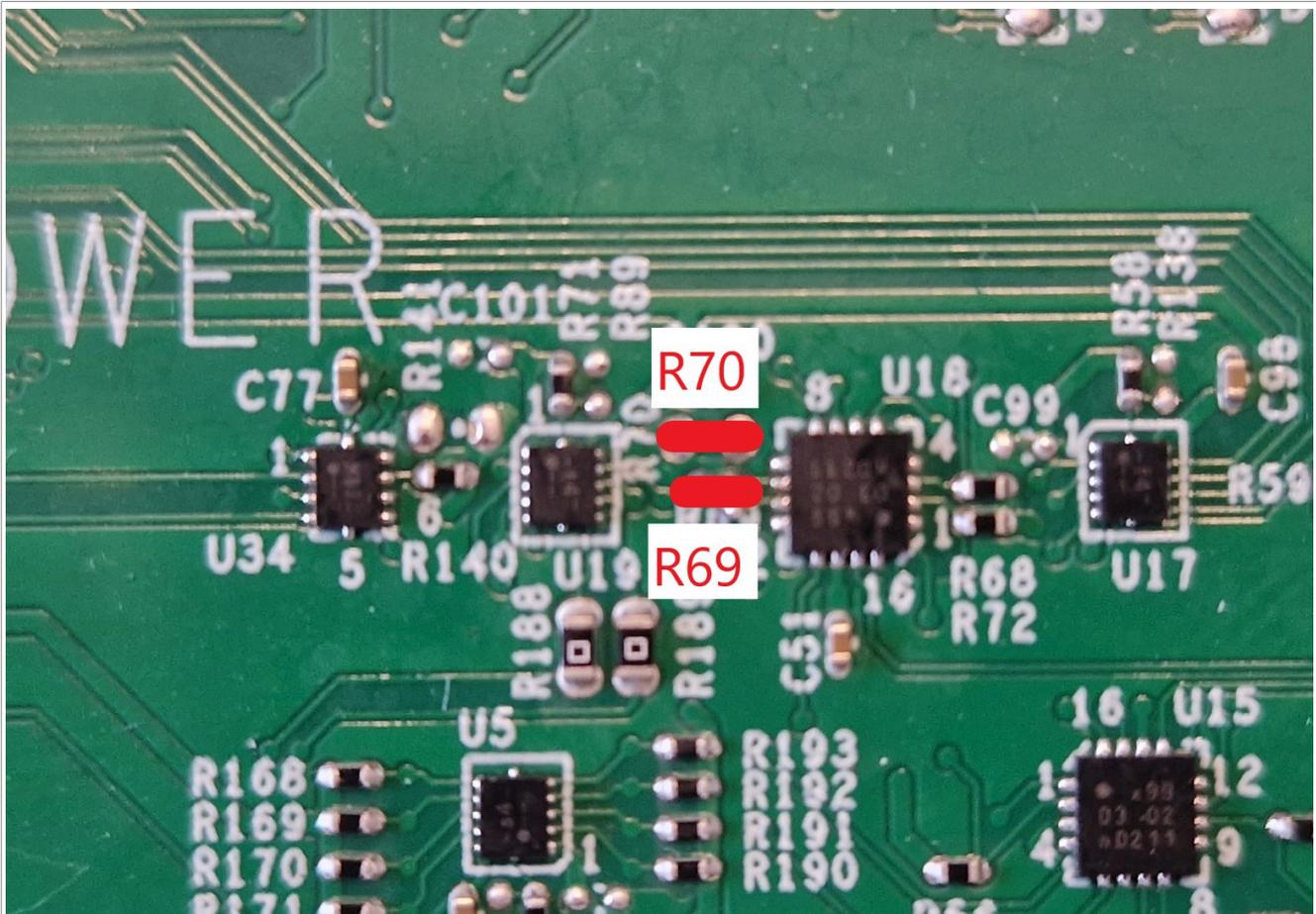


Figure 2. R69 and R70 location

### 2.2 R109 and R110

Both resistors are directly located below the jumper row and both are per default not populated. Solder a 0 ohm resistor to enable the SWD connection of the OpenSDA block.

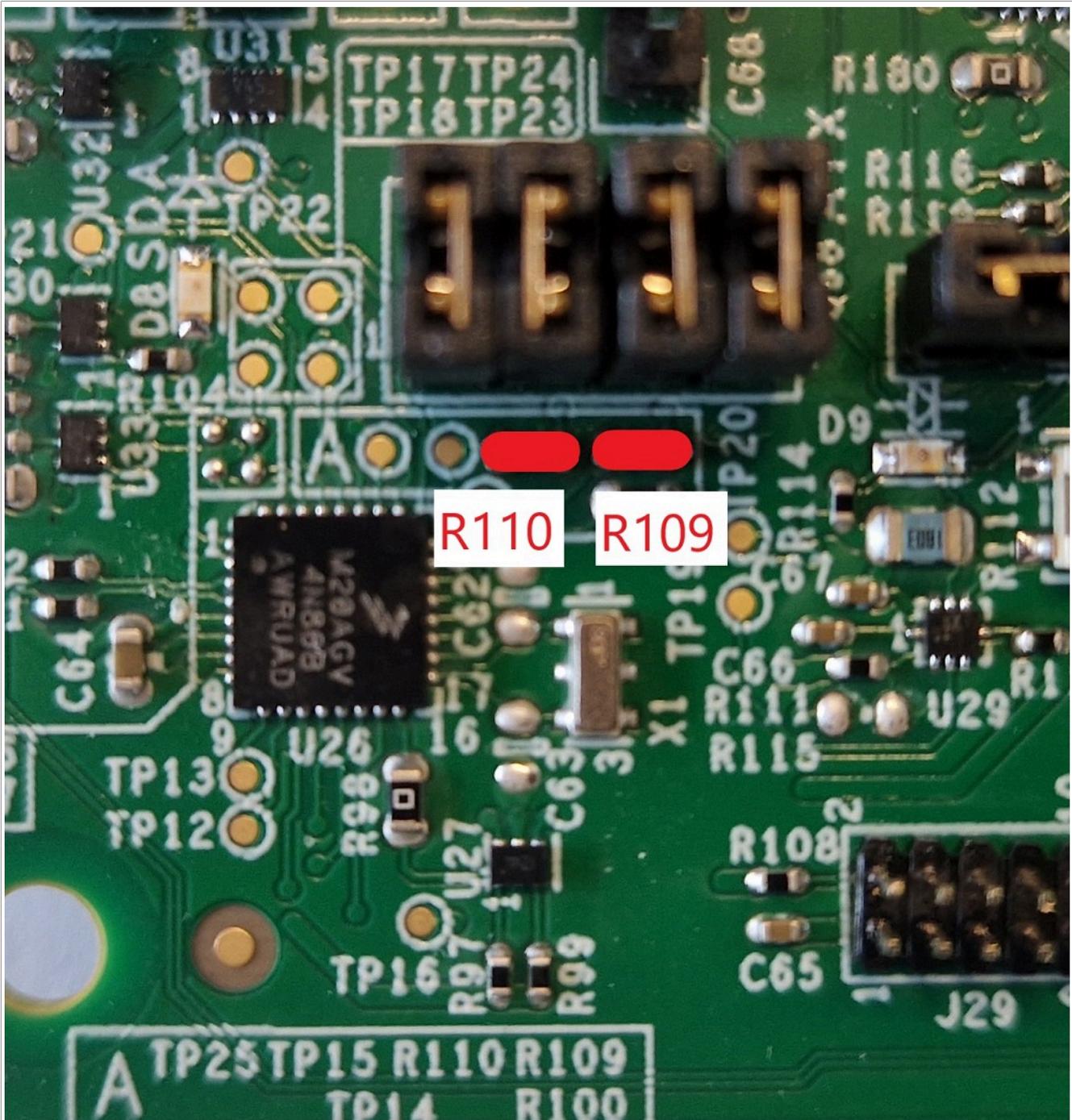


Figure 3. R109 and R110 location

### 3 Program OpenSDA bootloader

---

The onboard K20 is empty and the OpenSDA bootloader must be programmed from the user. This chapter explains how to program the bootloader and further on load the OpenSDA application on it.

For this document, a SEGGER J-Link and the tools coming with the J-Link SDK are used. Potentially any programmer which can program a Kinetis K20 can be used, but this is not explained within this document.

Prerequisites:

- Programmer capable to program Kinetis K20 (in this document used: SEGGER J-Link)
  - Connected to J29 (located in the OpenSDA section of the board)
- USB-C cable
  - Connected to J27 (located in the OpenSDA section of the board)
- K20 bootloader
  - Downloadable at the PN76 software section
- OpenSDA bootloader application (OpenSDA V2.1 bootloader)
  - Directly from SEGGER [\[1\]](#)

### 3.1 Program K20 bootloader

Connect the Programmer and USB-C cable to the PNEV7642A development board.



Figure 4. OpenSDA programming connection

Using J-Flash Lite to program the "K20\_Backup\_Bootloader.bin" to the K20:

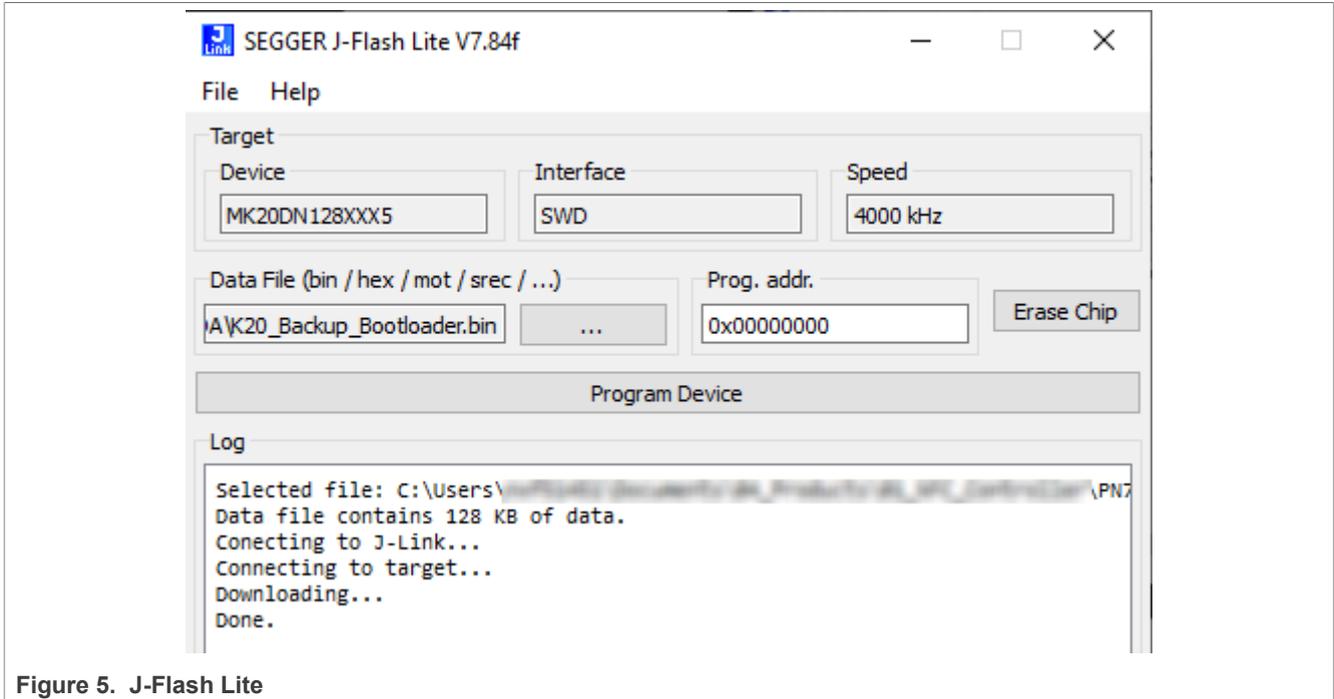


Figure 5. J-Flash Lite

After the download is successful, the programmer can be unplugged.

### 3.2 K20 bootloader

To bring the K20 into bootloader mode which populates a mass storage device on the host PC, follow these steps:

- Unplug the USB-C Cable on J27
- Press and hold SW4 (small white button located in the OpenSDA area)
- Plug in the USB-C cable on J27
- Release the reset button "SW4"

A mass storage device appears:

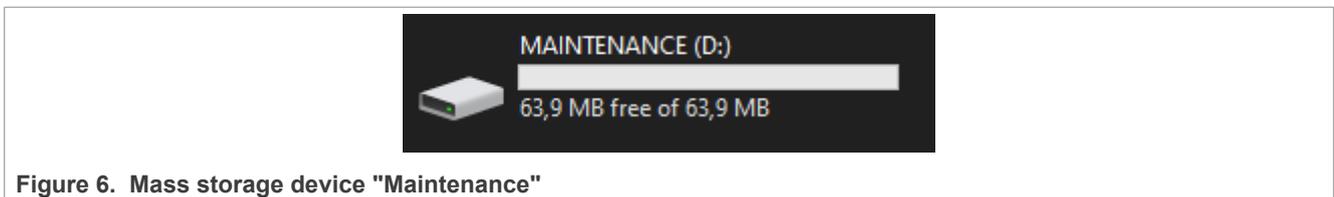


Figure 6. Mass storage device "Maintenance"

Onto this mass storage device, you can now load any OpenSDA application.

### 3.3 Load an OpenSDA application

After you have entered the bootloader mode, and a mass storage device appeared, you can load any OpenSDA application onto the K20. By simply drag&drop or copy&paste the application binary it is programmed.

We use the OpenSDA\_V2.1.bin which we obtained previously from the SEGGER homepage.

After successful operation you see a J-Link device driver and COM Port in your device manager:

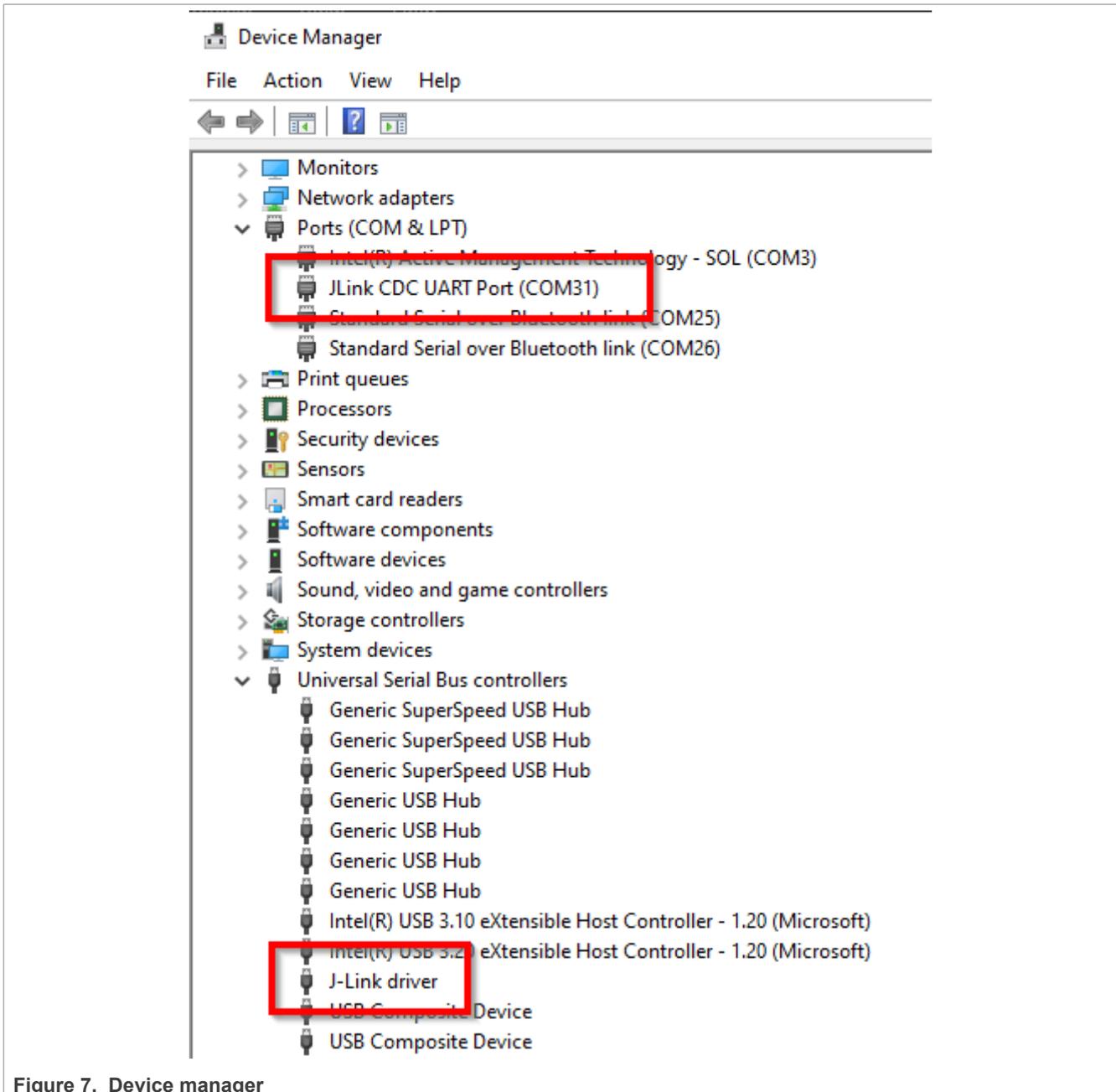
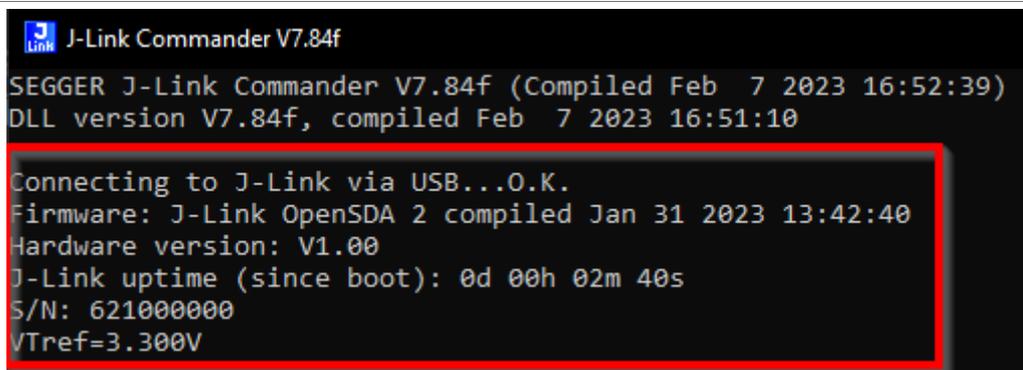


Figure 7. Device manager

To ensure the connection is working, you can open J-Link commander and see if it connects to J-Link OpenSDA.

A screenshot of the J-Link Commander V7.84f application window. The window title is "J-Link Commander V7.84f". The terminal output shows the following text: "SEGGER J-Link Commander V7.84f (Compiled Feb 7 2023 16:52:39)", "DLL version V7.84f, compiled Feb 7 2023 16:51:10", "Connecting to J-Link via USB...O.K.", "Firmware: J-Link OpenSDA 2 compiled Jan 31 2023 13:42:40", "Hardware version: V1.00", "J-Link uptime (since boot): 0d 00h 02m 40s", "S/N: 621000000", and "VTref=3.300V". A red rectangular box highlights the connection status and hardware information lines.

```
J-Link Commander V7.84f
SEGGER J-Link Commander V7.84f (Compiled Feb 7 2023 16:52:39)
DLL version V7.84f, compiled Feb 7 2023 16:51:10
Connecting to J-Link via USB...O.K.
Firmware: J-Link OpenSDA 2 compiled Jan 31 2023 13:42:40
Hardware version: V1.00
J-Link uptime (since boot): 0d 00h 02m 40s
S/N: 621000000
VTref=3.300V
```

Figure 8. J-Link commander OpenSDA connection

## 4 Use OpenSDA in MCUXpresso

The OpenSDA can be used as any other debugger within MCUXpresso [2].

Make sure that it is plugged in. To work on the PNEV7642A development board, you need at least the OpenSDA connected to J27 and a powered PNEV7642 board (by either USB "J5" or external power jack).

In MCUXpresso, before downloading or debugging code, it searches for attached probes. A window pops up with all discovered probes. Here you can use the J-Link OpenSDA same as any other debugger with the full capability of debugging, programming, and erasing the PN7642.

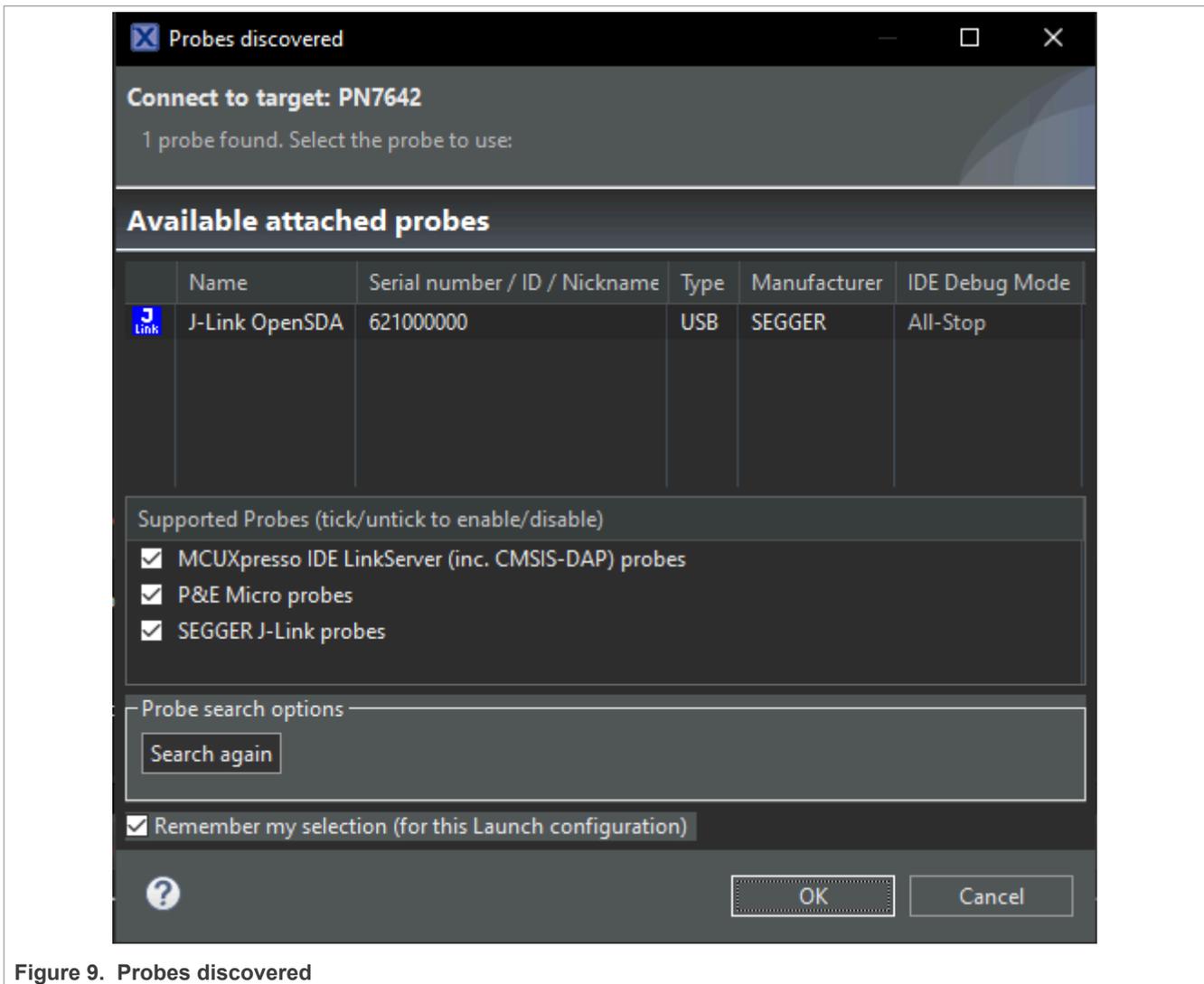


Figure 9. Probes discovered

After having chosen the J-Link OpenSDA, it downloads the chosen example to the PN7642 and start debugging.

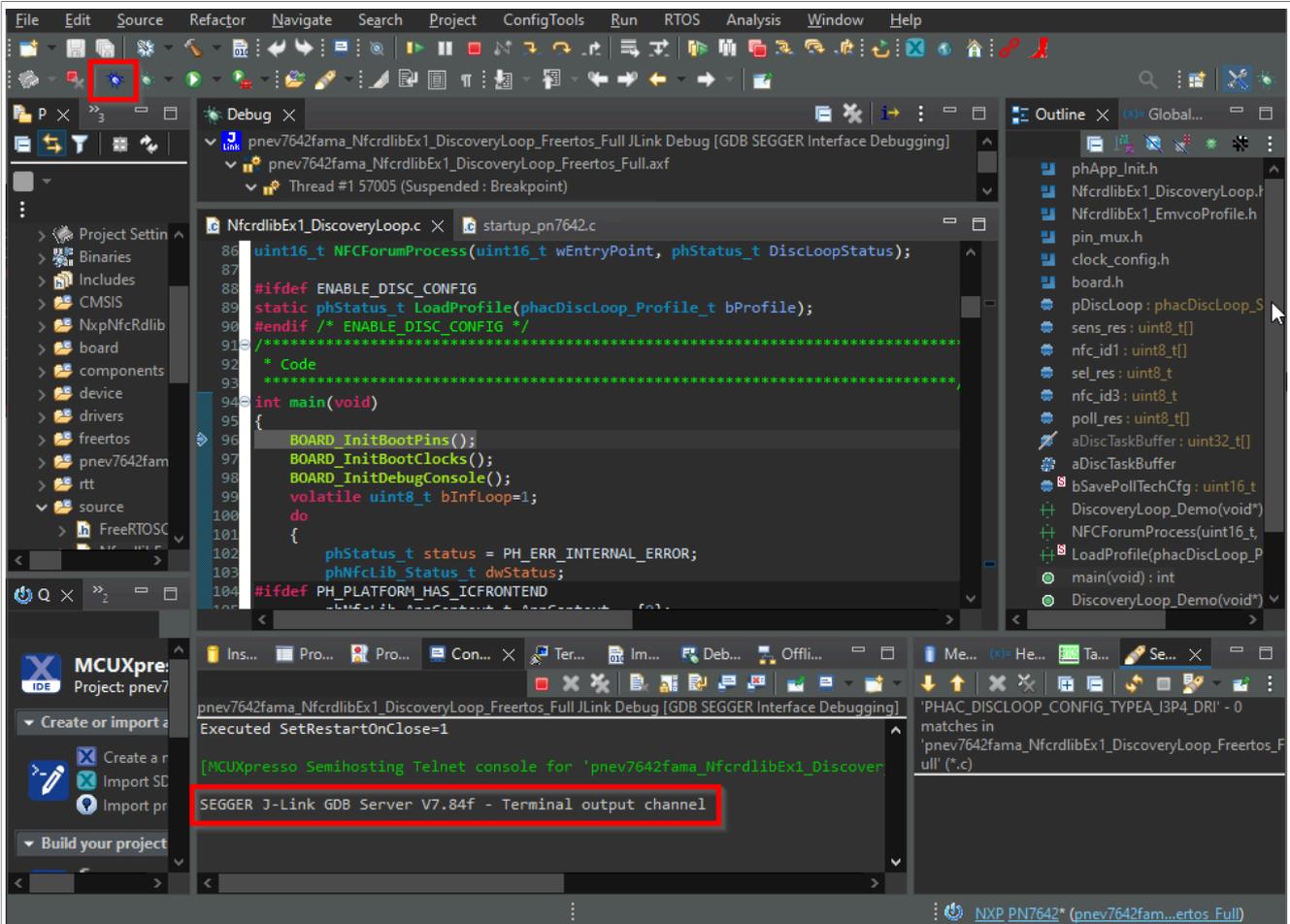


Figure 10. MCUXpresso debugging

## 5 References

---

- [1] SEGGER OpenSDA: <https://www.segger.com/products/debug-probes/j-link/models/other-j-links/opensda-sda-v2/>
- [2] MCUXpresso: <https://www.nxp.com/design:MCUXpresso-IDE>

## 6 Abbreviations

Table 1. Abbreviations

Acronym	Description
CDC	Communications Device Class
IDE	integrated development environment
JTAG	Joint Test Access Group
MCU	microcontroller
MSD	mass storage devic
OpenSDA	open-standard serial debug adapter
SDK	software development kit
SWD	serial wire debug
UART	universal asynchronous receiver transmitter

## 7 Legal information

### 7.1 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.

### 7.2 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 <http://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 data sheet 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](mailto:PSIRT@nxp.com)) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

### 7.3 Licenses

**Purchase of NXP ICs with NFC technology** — Purchase of an NXP Semiconductors IC that complies with one of the Near Field Communication (NFC) standards ISO/IEC 18092 and ISO/IEC 21481 does not convey an implied license under any patent right infringed by implementation of any of those standards. Purchase of NXP Semiconductors IC does not include a license to any NXP patent (or other IP right) covering combinations of those products with other products, whether hardware or software.

## 7.4 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.

**EdgeVerse** — is a trademark of NXP B.V.

**Kinetis** — is a trademark of NXP B.V.

**Tables**

Tab. 1. Abbreviations ..... 15

**Figures**

Fig. 1.	Board overview .....	4	Fig. 6.	Mass storage device "Maintenance" .....	9
Fig. 2.	R69 and R70 location .....	5	Fig. 7.	Device manager .....	10
Fig. 3.	R109 and R110 location .....	6	Fig. 8.	J-Link commander OpenSDA connection .....	11
Fig. 4.	OpenSDA programming connection .....	8	Fig. 9.	Probes discovered .....	12
Fig. 5.	J-Flash Lite .....	9	Fig. 10.	MCUXpresso debugging .....	13

---

## Contents

---

<b>1</b>	<b>Introduction .....</b>	<b>3</b>
1.1	OpenSDA .....	3
<b>2</b>	<b>Prepare hardware .....</b>	<b>4</b>
2.1	R69 and R70 .....	5
2.2	R109 and R110 .....	6
<b>3</b>	<b>Program OpenSDA bootloader .....</b>	<b>7</b>
3.1	Program K20 bootloader .....	8
3.2	K20 bootloader .....	9
3.3	Load an OpenSDA application .....	10
<b>4</b>	<b>Use OpenSDA in MCUXpresso .....</b>	<b>12</b>
<b>5</b>	<b>References .....</b>	<b>14</b>
<b>6</b>	<b>Abbreviations .....</b>	<b>15</b>
<b>7</b>	<b>Legal information .....</b>	<b>16</b>

---

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

---

© 2023 NXP B.V.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

Date of release: 13 March 2023  
Document identifier: AN13898