



NXP Matter Zigbee Bridge User Guide

NXP Linux Matter solutions [matter v1.1]



Contents

1. Introduction	3
2. NXP Matter Zigbee Bridge Solution.....	3
2.1 Support Solutions.....	3
2.2 Feature List.....	4
2.2 Hardware Requirements.....	4
2.3 Software Requirements	4
3 Build Matter Zigbee Bridge App	5
3.1 Build Matter Zigbee Bridge App.....	5
4 Setup Matter Zigbee Bridge Demo.....	5
4.1 Matter over Ethernet Bridge	5
4.2 Matter over Wi-Fi Bridge	8
4.3 OTBR & Bridge on one device.....	10
Appendix.....	11
Revision History.....	11
Legal.....	11

1. Introduction

Matter (formerly Project Connected Home over IP, or Project CHIP) is a new Working Group within the Connectivity Standards Alliance (CSA, formerly Zigbee Alliance). This Working Group plans to develop and promote the adoption of a new, royalty-free connectivity standard to increase compatibility among smart home products, with security as a fundamental design tenet.

Traditional non-matter devices cannot directly join the matter network. But Matter Bridge solve the problem. Matter bridge can join a Matter network as a Matter device, and non-matter device need to be mapped to Matter network as a dynamic endpoint. In this way, other Matter devices can communicate with non-matter device through dynamic endpoint.

2. NXP Matter Zigbee Bridge Solution

NXP Matter Zigbee Bridge is based on i.MX MPU and K32W061. IW612 or other WiFi module as optional items. Matter Zigbee Bridge app can run on all i.MX MPU platform, K32W061 is a Zigbee Gateway and runs the Zigbee stack.

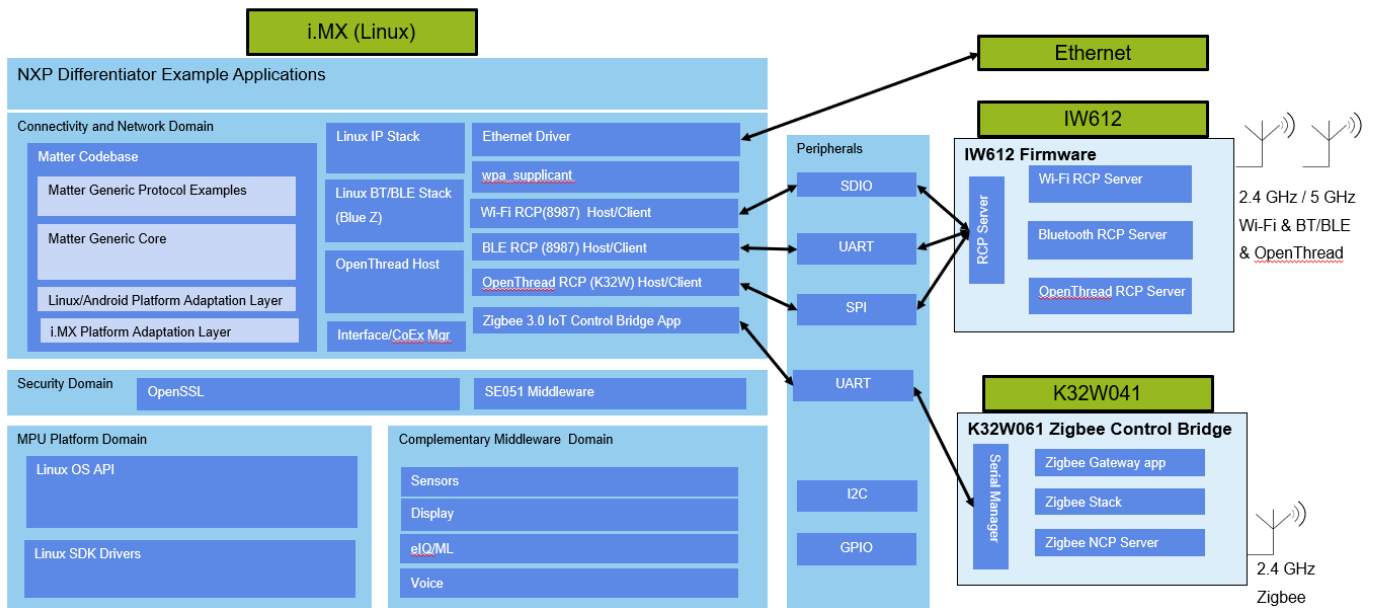


Figure 1-1 Matter Zigbee Bridge Framework

2.1 Support Solutions

2.1.1 Matter over ethernet

- i.MX6ULL + K32W0X(Zigbee NCP)
- i.MX8MM + K32W0X(Zigbee NCP)

- i.MX93 + K32W0X(Zigbee NCP)
- 2.1.2 Matter over wifi
- i.MX6ULL + K32W0X(Zigbee NCP) + IW416
 - i.MX8MM + K32W0X(Zigbee NCP) + 8987
 - i.MX93 + K32W0X(Zigbee NCP) + IW612

2.2 Feature List

- Matter over Ethernet
- Matter over Wi-Fi
- Register and Remove Zigbee Devices
- Connect Zigbee devices to Matter ecosystem seamlessly
- Zigbee Devices
 - OnOff cluster
 - Temperature Sensor Cluster
- Matter Actions
 - Start Zigbee Network
 - Zigbee Network Permit Join
 - Factory Reset
- No limitation if migrating to other i.MX MPU like i.MX6ULL, i.MX8MP
- OTBR and Zigbee bridge can be integrated into one single device

The guide is based on i.MX93 and other i.MX MPU can refer to the guide.

2.2 Hardware Requirements

- i.MX93EVK
- K32W USB Dongle
 - USB A to MINI USB
 - USB type C to A
- K32W061 DK006
- 2EL (IW612) M.2 WIFI Module (optional)

2.3 Software Requirements

- [Universal Update Utility \(UUU\)](#)
- A serial COM terminal for the host computer
 - For instructions on how to set up a serial COM terminal software, please refer the [i.MX 93 EVK Quick Start Guide](#).
- DK6Programmer.exe which Installer path is [K32W061DK6 SDK 2.6.11](#), SDK_2_6_11_K32W061DK6\tools\ JN-SW-4407 DK6 Production Flash Programmer v4564.exe
- **i.MX93 Demo Image.** [6.1.36 2.1.0 MX93 demo image](#)

- **Zigbee control bridge.** [ControlBridge_Full_GpProxy_1000000.bin](#)
- **Zigbee Light** , [JN-AN-1244](#)\Binaries\DimmableLight_GpProxy_OM15081\
DimmableLight_GpProxy_OM15081.bin
- **Zigbee Temperature Sensor** , [JN-AN-1246](#)\Binaries\LTOSensor_NtagIcode_Ota_OM15081R2\
LTOSensor_NtagIcode_Ota_OM15081R2_V1.bin

3 Build Matter Zigbee Bridge App

Currently, Matter Zigbee Bridge source code hasn't been officially released. Therefore, it's temporarily released through patch.

3.1 Install i.MX93 SDK on host PC

Install Yocto SDK. Refer to the README to generate SDK.[How to build OpenThread Border Router with the Yocto SDK](#)

```
# Generate yocto sdk
$ bitbake imx-image-multimedia -c populate_sdk

# install sdk for i.MX93
$ sudo tmp/deploy/sdk/fsl-imx-xwayland-glibc-x86_64-imx-image-multimedia-
armv8a-imx93evk-toolchain-6.1-langdale.sh
```

3.2 Build Matter Zigbee Bridge App

```
$ git clone https://github.com/NXP/matter.git -b v1.1.0.1-tag-nxp
$ cp 0001-Add-NXP-Linux-Matter-Zigbee-Bridge-demo.patch matter
$ cd matter
$ git apply 0001-Add-NXP-Linux-Matter-Zigbee-Bridge-demo.patch

# Build matter bridge app
$ export IMX_SDK_ROOT=/opt/fsl-imx-xwayland/6.1-langdale-imx93
$ source scripts/activate.sh
$ ./scripts/examples/imxlinux_example.sh examples/bridge-app/nxp/linux-imx
out/imx-bridge-app
```

The imx-chip-bridge-app will be generated under out/imx-bridge-app

4 Setup Matter Zigbee Bridge Demo

4.1 Matter over Ethernet Bridge

Running the Matter Zigbee Bridge app on the original Zigbee gateway to make it can join Matter network without increasing hardware costs.

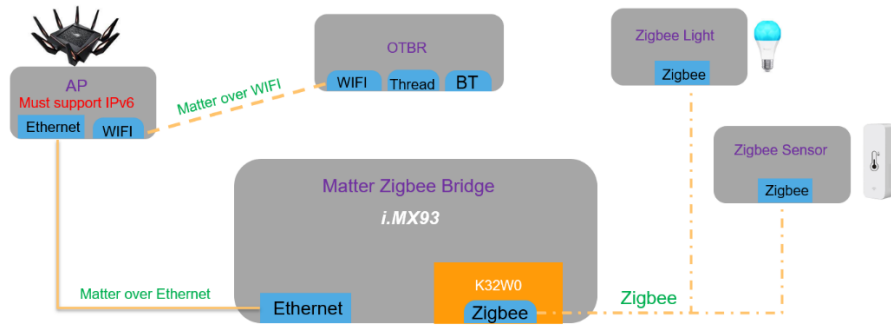


Figure 4-1 Matter Zigbee Bridge topology

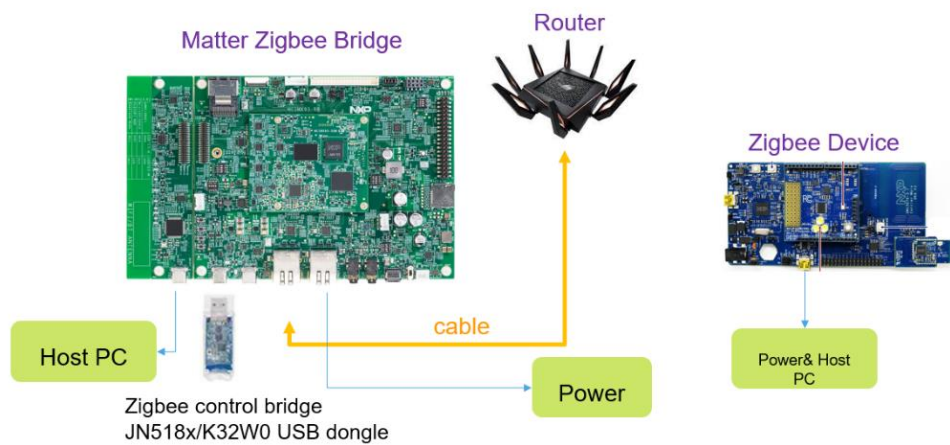


Figure 4-2 i.MX93 Matter Zigbee Bridge

1) Download i.MX93 image by UUU tool. Reference: https://www.nxp.com/docs/en/user-guide/IMX_LINUX_USERS_GUIDE.pdf Chapter 4.2

Demo Image: [6.1.36 2.1.0 MX93 demo image](#)

Run command on Host PC

```
$ uuu -b emmc_all .\imx-image-multimedia-imx93evk-matter.wic
```

2) Download Zigbee binary to K32W061

- Download [K32W061DK6 SDK 2.6.11](#)
- Install DK6Programmer.exe

DK6Programmer.exe Installer Path: SDK_2_6_11_K32W061DK6\tools\ JN-SW-4407 DK6 Production Flash Programmer v4564.exe

- Download **Zigbee Light** Binary. [JN-AN-1244](#)\Binaries\DimmableLight_GpProxy_OM15081\ DimmableLight_GpProxy_OM15081.bin

```
DK6Programmer.exe -s <COM_PORT> -e FLASH -p DimmableLight_GpProxy_OM15081.bin
```

- Download **Zigbee Temperature Sensor** Binary. [JN-AN-1246](#)\Binaries\LTOSensor_NtagIcode_Ota_OM15081R2\LTOSensor_NtagIcode_Ota_OM15081R2_V1.bin

```
DK6Programmer.exe -s <COM_PORT> -e FLASH -p LTOSensor_NtagIcode_Ota_OM15081R2_V1.bin
```

- Download **Zigbee control bridge** Binary to K32W USB dongle.
Must Download here [ControlBridge Full GpProxy 1000000.bin](#)

```
DK6Programmer.exe -s <COM_PORT> -e FLASH -p ControlBridge_Full_GpProxy_1000000.bin
```

1) Feature Test

- Setup OBTR
- Run imx-chip-bridge-app on i.MX93

```
$ ./imx-chip-bridge-app
```

- Test [run this command on controller, OTBR can as controller]

```
# 1. Onnetwork pairing
$ chip-tool pairing onnetwork 8888 20202021

# 2. Matter actions
# Action1: Start Network
# Action2: permit join
# Action3: Factory reset

Factory reset: $ chip-tool actions instant-action 0x1003 8888 1
Start Network: $ chip-tool actions instant-action 0x1001 8888 1
permit join: $ chip-tool actions instant-action 0x1002 8888 1

# 3. Add Zigbee Light, press the reset button or factory reset k32w061dk6 board
# 4. Add Zigbee Temperature sensor.
# Zigbee Device factory reset steps:
# 1st, Hold down the USER INTERFACE button and RESET button
# 2nd, release RESET button
# 3rd, release USER INTERFACE button

# 5. Read Matter device label. (last param is node id)
$ chip-tool bridgeddevicebasicinformation read node-label 8888 3
$ chip-tool bridgeddevicebasicinformation read node-label 8888 4

# 6. control Zigbee Light
$ chip-tool onoff toggle 8888 3

# 7. subscribe Zigbee Temperature Sensor data
```

```
$ chip-tool interactive start
>> temperaturemeasurement subscribe measured-value 10 20 8888 3
```

4.2 Matter-Zigbee Bridge over Wi-Fi

Users can enable WIFI/BT on i.MX93 matter bridge and join Apple Homekit.

Hardware Requirement:

- HomoPod Mini
- Murata 2EL Module (IW612)
- iPhone or iPad (OS version > 16.4.1, Recommended to upgrade to the latest version)

iPhone will discover devices through BT by default. So in this case, need to enable iPhone and Zigbee Bridge wifi&BT. BTW, If an OTBR wants to join the Homekit network, it needs to obtain permission from Apple. For details, please visit the Apple Community. Matter Bridge is a Matter end device, there is no such strict restriction.

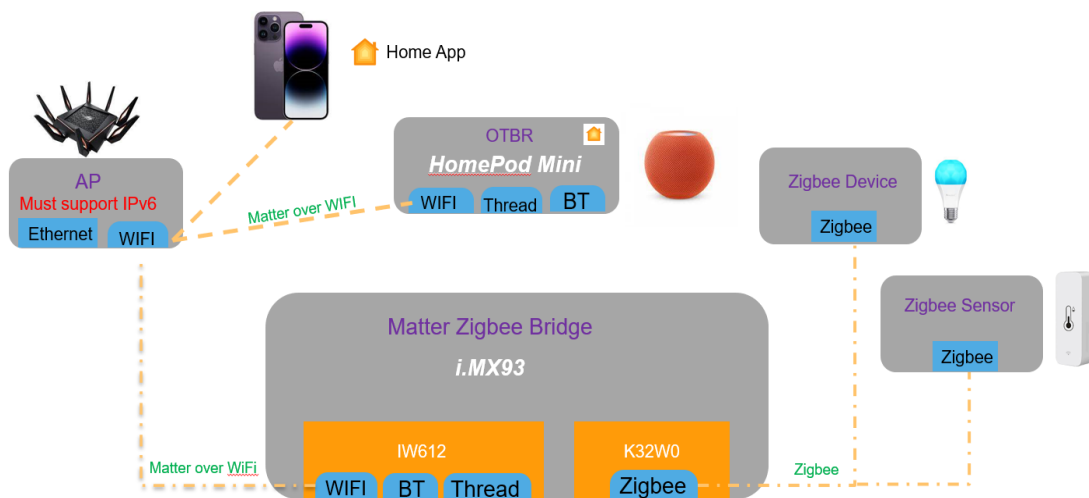


Figure 4-3 i.MX93 Matter Zigbee Bridge in HomeKit

- 1) Enable WIFI and BT, Reference: [IMX LINUX USERS GUIDE.pdf](#) Chapter11
i.MX93 EVK

```
$ modprobe moal mod_para=nxp/wifi_mod_para.conf
$ wpa_passphrase ${SSID} ${PASSWORD} > imxrouter.conf
$ wpa_supplicant -d -B -i wlan0 -c ./imxrouter.conf
$ udhcpc -i wlan0

$ modprobe btnxpuart
$ hciconfig hci0 up
```


2) Run Matter Bridge

```
$ ./imx-chip-bridge-app
```

3) Add Matter Bridge to HomeKit

- Open Apple "Home app" on iPhone
- Click "+" → "add accessory" → scan QR code
- QR code link show on imx93 log. <https://project-chip.github.io/connectedhomeip/qrcode.html?data=MT%3A-24J042C00KA0648G00>

4) Add Zigbee Device to Matter Zigbee Bridge

- Input "1" when imx-chip-bridge-app running. The log will be show.

```
ff fc ff 00      ....  
permit join done !
```

- Reset or Factory Reset Zigbee Devices.

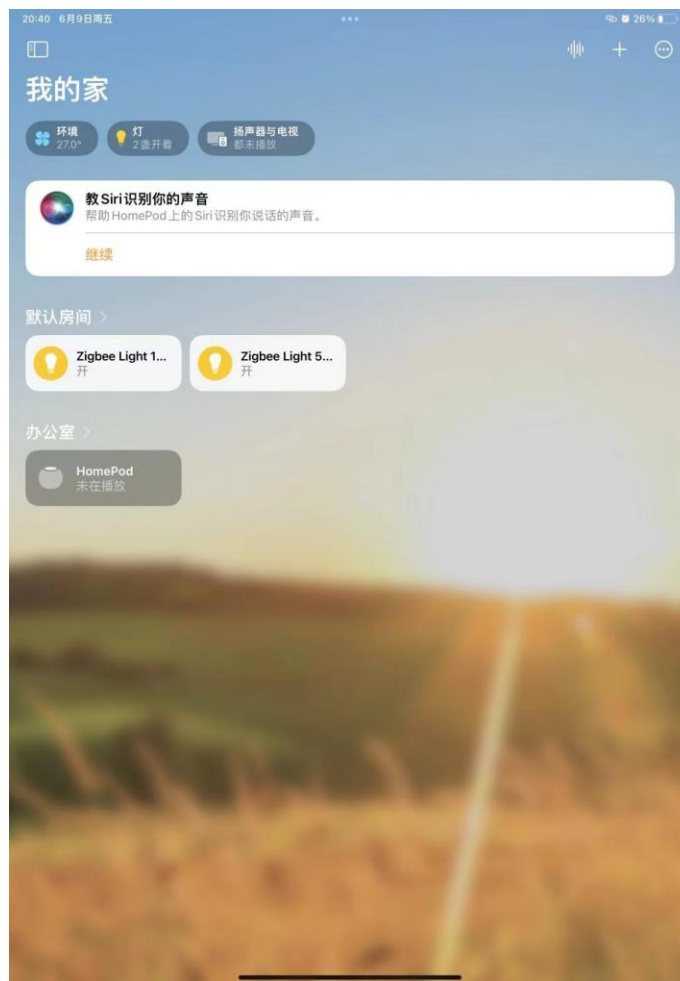


Figure 4-4 Home App with Zigbee Device

4.3 OTBR & Bridge on one device

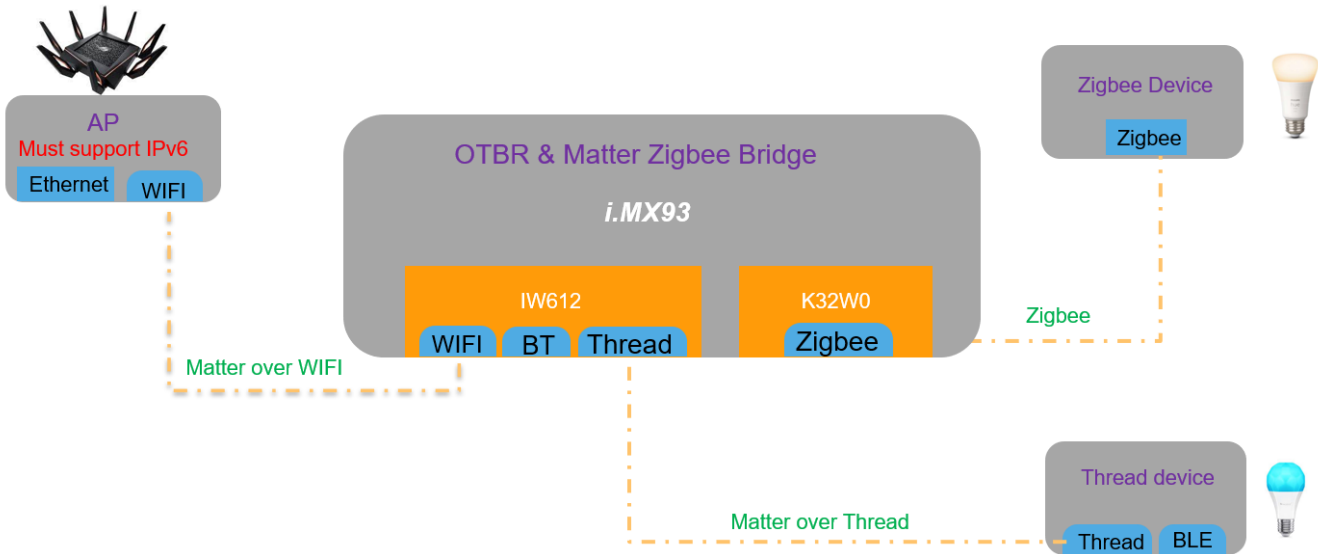


Figure 4-5 i.MX93 OTBR support Bridge solution

1) Download i.MX93 Image which support Matter.

Build guide: <https://github.com/nxp-imx/meta-matter>

```
$ uuu -b sd_all .\imx-image-multimedia-imx93evk-matter.wic
```

2) Setup OTBR, Reference. [IMX8MMINI-MATTER.pdf](#) or Run the script.

```
$ ./imx93_matter.sh init
```

3) Set up Bridge

```
$ ./imx-chip-bridge-app
```

Appendix

Revision History

Revision	Date	Description	Author
1.0	11/23/2023	Public Release	Gnar Fang/Forrest Sun

Legal

Limited warranty and liability – Information in this document is provided solely to enable system and software implementers to use NXP 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. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP 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 NXP 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. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

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.

Security – Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. 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, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, ICODE, JCOP, LIFE, VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. 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 or registered trademarks of Arm Limited (or its subsidiaries) 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. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

©NXP B.V. 2021. All rights reserved. For more information, please visit: <http://www.nxp.com>. For sales office addresses, please send an email to: salesaddresses@nxp.com