AN14310

NXP Bluetooth UART Driver Integration

Rev. 3.0 — 4 March 2025   Application note

Document information

Information	Content
Keywords	NXP Bluetooth UART driver, i.MX, Linux kernel version
Abstract	Explains how to compile and integrate the NXP Bluetooth UART driver.

1 Introduction

This document explains how to compile and integrate NXP Bluetooth UART driver. For i.MX kernel versions 6.1.22 and lower, this update is required.

Note: This document does not apply to Linux kernel version 6.1.22 or lower.

For third party ARM-based Linux BSP, a Linux kernel version 6.1.x and above is required. Older Linux kernel versions 5.x are known to have some compilation issues.

If you are not using i.MX Linux BSP host systems, you must have kernel version 6.1.x or later to use NXP Bluetooth UART driver. Kernel versions 5.x and older have compatibility issues due to some missing APIs.

1.1 Supported products

The application note applies to the following products:

- 88W8987 [7]
- 88W8997 [8]
- 88Q9098 [9]
- 88W9098 [10]
- AW611 [11]
- AW692 [12]
- AW692 [13]
- IW416 [14]
- IW610 [15]
- IW611 [16]
- IW612 [17]

1.2 Advantages of using the NXP Bluetooth UART driver

Advantages of using the NXP Bluetooth UART driver include:

- Bring-up of the HCI interface without `hciattach` command.
  When the NXP Bluetooth UART driver is loaded, the driver detects whether Bluetooth firmware is downloaded or not. If the Bluetooth firmware is not downloaded, the driver downloads the firmware automatically.

- Bluetooth deep sleep mode
  The NXP Bluetooth driver reduces power consumption by waking up the Bluetooth controller, or by setting the Bluetooth controller to sleep.

Read more about the features of NXP Bluetooth UART driver in [1].

2 Compile and integrate NXP Bluetooth UART driver

This section describes the compilation and integration of NXP Bluetooth UART driver on i.MX Linux BSP and other non-i.MX-ARM-architecture-based Linux host platforms.

2.1 i.MX Linux BSP

Steps to compile and integrate a new NXP Bluetooth driver on an i.MX 8 platform:

Note: *Skip this section if you use the BSP release with kernel version v6.1.22 and later.*

Step 1 – Clone NXP Linux kernel source code from [2] using Git clone command on the Linux system.

Step 2 – Download NXP Bluetooth UART driver patches from [3] and apply to a kernel source code.

For example:

```
git apply patch.diff
```

Step 3 – Edit the kernel configuration file available in the kernel source code.

• Add the following line to the *.config* file:

```
CONFIG_BT_NXPUART=m
```

Step 4 – Update the DTS file.

The *.dts* file is kept in the *arch/arm64/boot/dts/freescale/* directory of the kernel build environment.

• Add a `bluetooth` node with a device compatibility string to the attached UART node.

```
&uart1 {
        bluetooth {
                compatibility = "nxp,88w8987-bt";
                fw-init-baudrate = <3000000>; # Optional. Default is considered 115200
                if this parameter not defined.
        };
};
```

Note: *The `fw-init-baudrate` parameter depends on the product OTP configuration. Contact your NXP representative for more information.*

Table 1 lists the compatibility option values for the supported products.

Table 1. Compatibility option values

Compatibility option	Description	Supported products
`nxp,88w8987-bt`	The binary file *helper_xxx.bin* is not required to download the Bluetooth firmware. Bluetooth UART driver checks the availability of Bluetooth firmware binary and downloads the firmware.	88W8987, 88Q9098, 88W9098, IW416, AW611, IW611, IW612, IW610, AW692, AW693
`nxp,88w8997-bt`	The binary file *helper_xxx.bin* is required to download the Bluetooth firmware. Bluetooth UART driver checks the availability of *helper_xxx.bin* and Bluetooth firmware binary, and downloads the *helper_xx.bin* file followed by the firmware binary.	88W8997

AN14310

All information provided in this document is subject to legal disclaimers.

© 2025 NXP B.V. All rights reserved.

Application note

Rev. 3.0 — 4 March 2025

Document feedback

3 / 14

Step 5 – Compile the kernel and the *.dts* file.

```
make -j32 ARCH=arm64 CROSS_COMPILE=aarch64-poky-linux-
```

Step 6 – Locate the generated files in the compiled kernel source.

- Path to the kernel image: *arch/arm64/boot/Image*
- Path to NXP Bluetooth UART driver module (*btnxpuart.ko*): *drivers/bluetooth/btnxpuart.ko*
- Path to *xxx.dtb* file: *arch/arm64/boot/dts/freescale/xxx.dtb*

Step 7 – Copy the compiled kernel Image, *xxx.dtb* and *btnxpuart.ko* files to the i.MX 8M platform and reboot the system.

Example for i.MX 8M Quad platform:

```
cp xxx.dtb /run/media/boot-mmcblk0p1/imx8mq-evk.dtb
cp Image /run/media/boot-mmcblk0p1/Image
cp btnxpuart.ko /home/root
sync
reboot
```

Step 8 – Follow the steps in [Section 3](#) to load the NXP UART driver module.

2.2 Third party ARM-based Linux BSP

Note: NXP Bluetooth UART driver is supported on ARM-based architectures. Other architectures such as x86 will not work.

Steps to compile and integrate a new NXP Bluetooth driver on a third party ARM-based Linux kernel:

Step 1 – Apply BT NXP UART driver patches on kernel source code.

- Download NXP Bluetooth UART driver patches from [4] and apply to your specific kernel source code.
- Then download other NXP Bluetooth UART driver patches from [5] and apply to a kernel source code. Command example:

```
git apply patch.diff
```

Note: Skip this step if Linux kernel version 6.4 or higher is in use as these patches are already included.

Step 2 – Edit the kernel configuration file available in the kernel source code.

- Add the following line to the *.config* file:

```
CONFIG_BT_NXPUART=m
```

Step 3 – Update the DTS file.

The *.dts* file is kept in the *arch/arm64/boot/dts/freescale/* directory of the kernel build environment.

- Add a `bluetooth` node with a device compatibility string to the attached UART node.

```
&uart1 {
        bluetooth {
                compatibility = "nxp,88w8987-bt";
                fw-init-baudrate = <3000000>; # Optional. Default is considered 115200
                if this parameter not defined.
        };
};
```

Note: The `fw-init-baudrate` parameter depends on the product OTP configuration. Contact your NXP representative for more information.

Table 2 lists the compatibility option values for the supported products.

Table 2. Compatibility option values

Compatibility option	Description	Supported products
`nxp,88w8987-bt`	The binary file *helper_xxx.bin* is not required to download the Bluetooth firmware. Bluetooth UART driver checks the availability of Bluetooth firmware binary and downloads the firmware.	88W8987, 88Q9098, 88W9098, IW416, AW611, IW611, IW612, IW610, AW692, AW693
`nxp,88w8997-bt`	The binary file *helper_xxx.bin* is required to download the Bluetooth firmware. Bluetooth UART driver checks the availability of *helper_xxx.bin* and Bluetooth firmware binary, and downloads the *helper_xx.bin* file followed by the firmware binary.	88W8997

Step 4 – Compile the kernel and the *.dts* file.

```
make
```

Step 5 – Locate the generated files in the compiled kernel source.

- Path to the kernel image: *arch/arm64/boot/Image*
- Path to NXP Bluetooth UART driver module (*btnxpuart.ko*): *drivers/bluetooth/btnxpuart.ko*
- Path to *xxx.dtb* file: *arch/arm64/boot/dts/freescale/xxx.dtb*

Step 6 – Copy the compiled kernel Image, *xxx.dtb, and btnxpuart.ko* files to the platform and reboot the system.

Command example for i.MX 8M Quad platform:

```
cp xxx.dtb /run/media/boot-mmcblk0p1/imx8mq-evk.dtb
cp Image /run/media/boot-mmcblk0p1/Image
cp btnxpuart.ko /home/root
sync
reboot
```

Step 7 – Follow the steps in Section 3 to load the NXP UART driver module.

3 Load NXP UART driver module

This section shows how to load NXP Bluetooth UART driver module (*btnxpuart.ko*).

By default, the NXP Bluetooth UART driver (*btnxpuart.ko*) is loaded. To disable the automatic load option for NXP Bluetooth UART driver, edit the */etc/modprobe.d/blacklist.conf* file to add one line as shown below, and reboot the system.

```
blacklist btnxpuart
```

Note: If you don't blacklist the btnxpuart driver, the NXP Bluetooth UART driver is loaded at every reboot, and the Bluetooth-only firmware is downloaded. To use the Wi-Fi and Bluetooth combo firmware, you must blacklist NXP Bluetooth UART driver.

Step 1 – Load the Wi-Fi driver firmware.

Note: Skip this step if Wi-Fi is not used.

- Use the `insmod/modprobe` command to load either the Wi-Fi and Bluetooth combo firmware or the Wi-Fi only firmware.
  Example of command:

```
modprobe mlan
modprobe moal mod_para=nxp/wifi_mod_para.conf
```

- Refer to [6] for detailed instructions if i.MX Linux BSP is used.
- Verify the kernel debug messages in the command output.

Note: To load the firmware (Wi-Fi only or Wi-Fi and Bluetooth combo), the type of firmware defined in wifi_mod_para.conf determines which driver is used. For Bluetooth only firmware, btnxpuart.ko driver loads the firmware within 5 seconds. For Wi-Fi and Bluetooth combo firmware, btnxpuart.ko driver does not load any firmware. The moal.ko or another driver loads the combo firmware.

Step 2 – Load NXP UART Bluetooth driver.

- Download the Bluetooth only firmware binary from [2].
- For Linux kernel v6.1.22 and lower, issue the command:

```
insmod btnxpuart.ko
```

- For Linux kernel v6.1.22 and higher, issue the command:

```
modprobe btnxpuart
```

- Verify that the Bluetooth interface is up and running.

```
hciconfig -a
```

Note: The command `hciattach` is not required to bring up Bluetooth.

4 Load the open-source UART driver

Linux open source UART driver is a standard UART driver available in the Linux open source directory. The driver does not support the Bluetooth deep sleep mode feature.

This section shows how to revert the changes for NXP Bluetooth UART driver when i.MX BSP source code has kernel version 6.1.22 and later.

Step 1 – Update the *.dtb* file

The *.dts* file is located in the *arch/arm64/boot/dts/freescale/* directory of the kernel source build environment.

- Edit the *.dts* file to remove the Bluetooth node for UART1:

```
&uart1 {
      bluetooth {
              compatibility = "nxp,88w8987-bt";
              fw-init-baudrate = <3000000>; # Optional. Default is considered 115200
              if this parameter is not defined.
      };
};
```

Step 2 – Compile the *.dts* file in the build environment.

```
dtc -O dtb -o imx8xx-evk-xxx.dtb imx8xx-evk-xxx.dts
```

Step 3 – Copy the newly generated *.dtb* file on Linux platform.

Example for i.MX 8M Quartz platform:

```
cp imx8xx-evk-xxx.dtb /run/media/boot-mmcblk0p1/imx8mq-evk.dtb
sync
reboot
```

Step 4 – Bring-up Bluetooth.

```
hciattach /dev/ttymxc2 any 115200 flow
hciconfig hci0 up
```

Note: *Refer to the section Bring-up of Bluetooth interfaces in [6].*

AN14310

All information provided in this document is subject to legal disclaimers.

© 2025 NXP B.V. All rights reserved.

Application note

Rev. 3.0 — 4 March 2025

Document feedback

8 / 14

5   References

[1]   Application note – AN13920: Enabling Bluetooth Deep-sleep with NXP Bluetooth UART Driver (link)

[2]   Resources – NXP Online Git Repository for NXP Kernel Source code (link)

[3]   Resources – NXP Bluetooth UART driver patches for kernel version lower than 6.1.22 (link)

[4]   Resources – NXP Bluetooth UART driver patches for non-i.MX Linux kernel (link)

[5]   Resources – NXP Bluetooth UART driver additional improvement patches with some bug fixes (link)

[6]   User manual – UM11483: Getting Started with NXP-based Wireless Modules on i.MX 8M Quad EVK Running Linux OS (link)

[7]   Webpage – 88W8987: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 5 (802.11ac) + Bluetooth® Solution (link)

[8]   Webpage – 88W8997: 2.4/5 GHz Dual-Band 2x2 Wi-Fi® 5 (802.11ac) + Bluetooth® Solution (link)

[9]   Webpage – 88Q9098/88Q9098S: 2.4/5 GHz Dual-Band 2x2 Wi-Fi® 6 (802.11ax) + Bluetooth® Automotive Solution (link)

[10] Webpage – 88W9098: 2.4/5 GHz Dual-Band 2x2 Wi-Fi® 6 (802.11ax) + Bluetooth® (link)

[11] Webpage – AW611: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 6 (802.11ax) + Bluetooth® Automotive Solution (link)

[12] Webpage – AW692: 2x2 Single-band (5 GHz) Concurrent Dual Wi-Fi® 6, 1x1 (2.4 GHz) Wi-Fi 6, and Bluetooth® Combo Solution (link)

[13] Webpage – AW693: 2x2 Dual-Band (5-7 GHz), 1x1 (2.4 GHz) Concurrent Dual Wi-Fi 6/6E and Bluetooth Combo Solution (link)

[14] Webpage – IW416: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 4 (802.11n) + Bluetooth® Solution (link)

[15] Webpage – IW610: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 6 + Bluetooth Low Energy + 802.15.4 Tri-Radio Solution (link)

[16] Webpage – IW611: 2.4/5 GHz Dual-band 1x1 Wi-Fi® 6 (802.11ax) + Bluetooth® Solution (link)

[17] Webpage – IW612: 2.4/5 GHz Dual-Band 1x1 Wi-Fi® 6 (802.11ax) + Bluetooth® + 802.15.4 Tri-Radio Solution (link)

AN14310

All information provided in this document is subject to legal disclaimers.

© 2025 NXP B.V. All rights reserved.

Application note

Rev. 3.0 — 4 March 2025

Document feedback

9 / 14

6 Note about the source code in the document

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

Copyright 2024-2025 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.

7 Revision history

Table 3. Revision history

Document ID	Release date	Description
AN14310 v.3.0	4 March 2025	• Section 1.1 "Supported products": updated. • Section 2.1 "i.MX Linux BSP": updated. • Section 2.2 "Third party ARM-based Linux BSP": updated. • Section 5 "References": updated.
AN14310 v.2.0	27 November 2024	Security status changed to public.
AN14310 v.1.0	16 September 2024	• Initial version

AN14310

All information provided in this document is subject to legal disclaimers.

© 2025 NXP B.V. All rights reserved.

Application note

Rev. 3.0 — 4 March 2025

Document feedback

11 / 14

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.

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.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

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.

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.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

AN14310

All information provided in this document is subject to legal disclaimers.

© 2025 NXP B.V. All rights reserved.

Application note

Rev. 3.0 — 4 March 2025

Document feedback

12 / 14

Tables

Contents

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