

UM11675

How to Download and Build NXP Wi-Fi Drivers

Rev. 4.0 — 30 June 2024

User manual

Document information

Information	Content
Keywords	NXP Wi-Fi driver source, Git tool, Windows OS, Linux OS, firmware binaries
Abstract	Guidelines to install Git on Linux/Windows systems, and download NXP Wi-Fi driver source files and firmware binaries from Git tool



1 Overview

This document describes how to download NXP Wi-Fi driver source and firmware binaries from GitHub. The user must first install the Git tool running on Linux and Windows PCs.

Note: *The URL to download the Wi-Fi driver source has changed from CodeAurora to GitHub in September 2022.*

2 Installing Git

This section describes the steps for Git installation. Before you start using Git, it should be installed on the system. If Git is already installed, make sure to use the latest version.

2.1 Installing Git on Linux

Use following command to install Git on the Debian-based distribution.

```
# sudo apt-get install git
```

2.2 Installing Git on Windows

- Download Git application using [ref.\[4\]](#).
- Double-click the executable file (.exe).
- Keep the default options and click **Next** to initiate the installation.

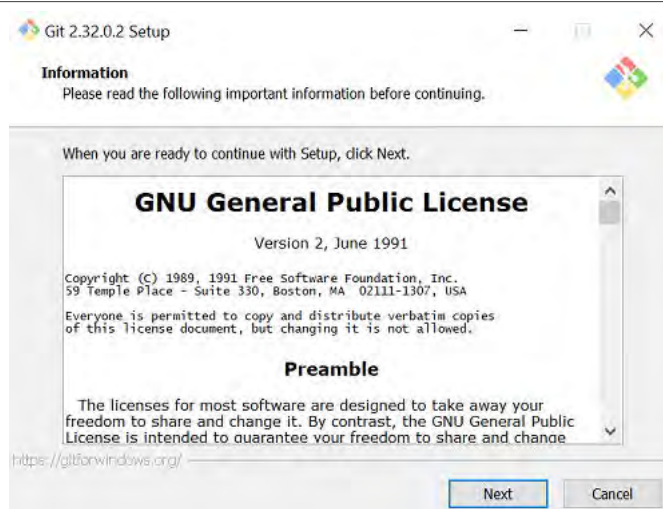


Figure 1. Git installation on Windows

When the installation is complete, Git CMD and Git GUI are available in Windows Explorer.

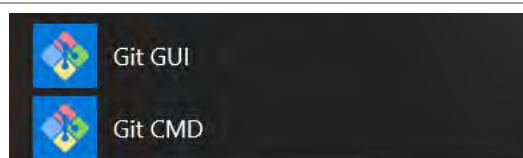


Figure 2. Git CMD and Git GUI in Windows Explorer

3 Download of NXP Wi-Fi driver source

This section shows how to use the Git tool to download NXP Wi-Fi driver source on your Windows or Linux Ubuntu system.

Note: HCI Bluetooth UART driver is open source and available in the Linux kernel source. The Bluetooth features are validated with the BlueZ version 5.56.

3.1 Using Git GUI on Windows

- Launch Git GUI.
- Click **Clone Existing Repository**.

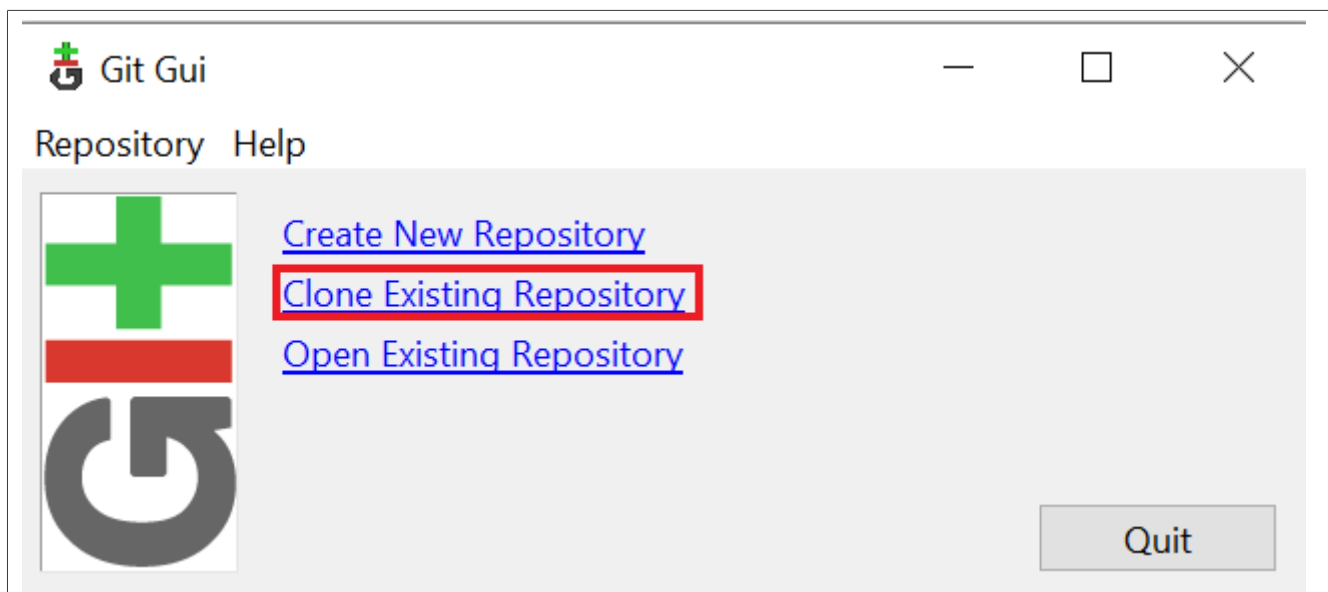


Figure 3. Clone existing repository

- Capture the address to the **source location** ([ref.\[3\]](#)).
- Define the **target directory** for NXP Wi-Fi driver source to download.

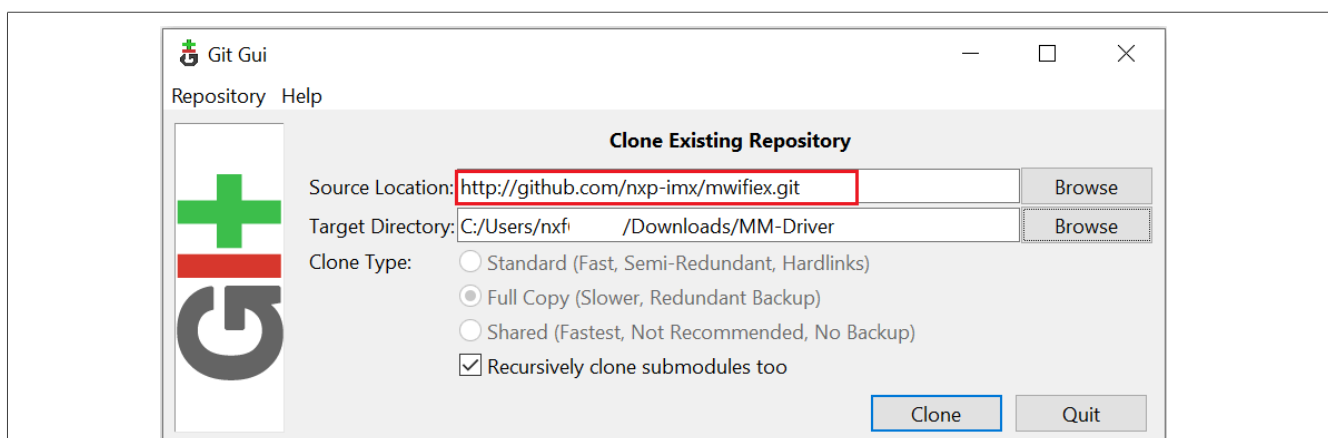


Figure 4. Capturing the source location and target directory

- To initiate the download of NXP Wi-Fi driver source, click the **Clone** button.

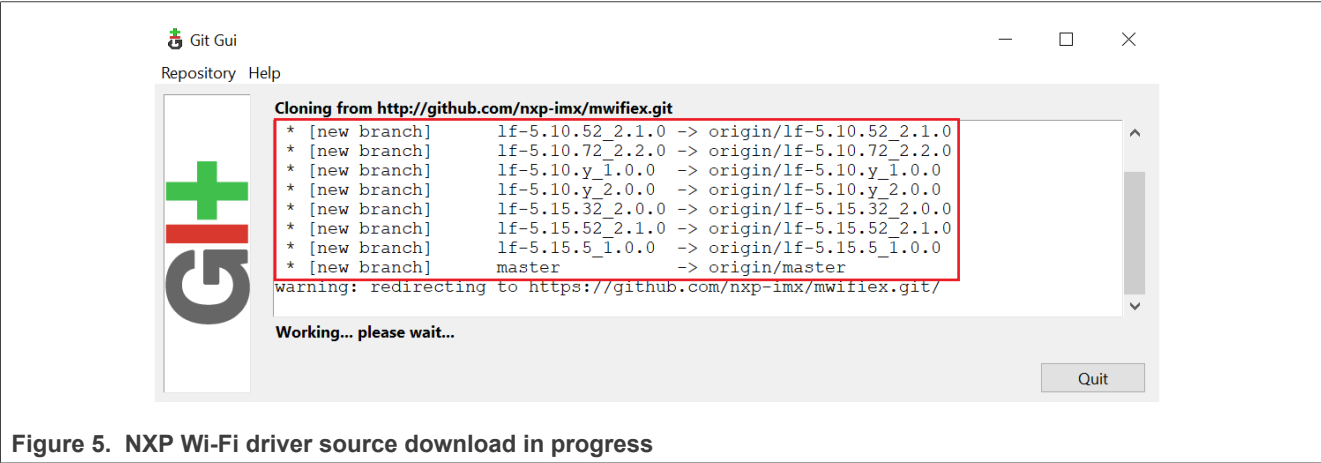


Figure 5. NXP Wi-Fi driver source download in progress

Once the download is complete, the driver source is available in the target directory.

3.2 Using Git CMD on Windows

- Launch Git CMD from Windows Explorer
- Go your Windows home directory
- Use `mkdir` command to create the *source* directory to store NXP Wi-Fi driver source
- Use `cd` command to go to *source* directory

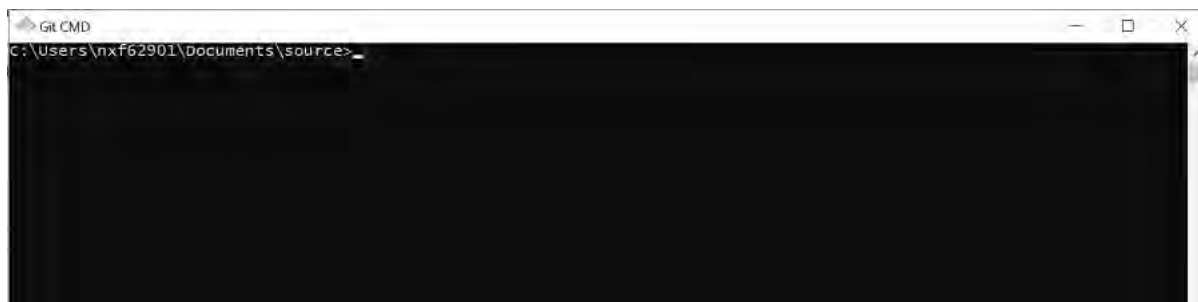


Figure 6. User home directory in Git CMD

- Use the `git clone` command to download NXP Wi-Fi driver source

```
git clone http://github.com/nxp-imx/mwifiex.git
```

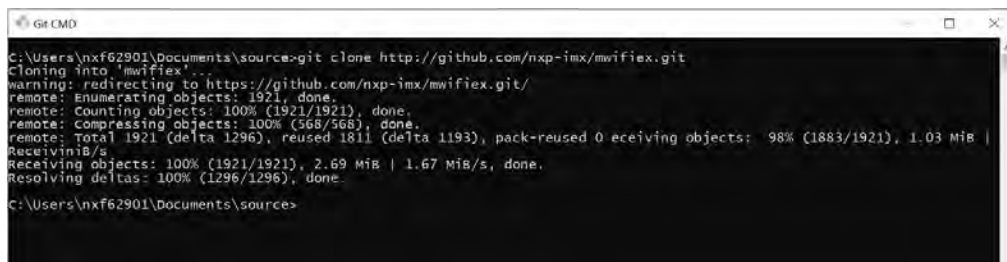


Figure 7. `git clone` command output

Once the download completed, NXP Wi-Fi driver source is available in the (*source*) directory.

3.3 Using Git CMD on Linux

- Launch the Linux terminal (CTRL+ALT+T)

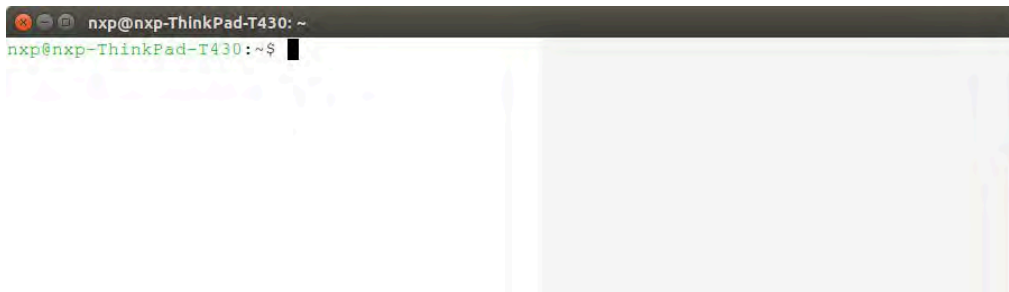


Figure 8. Git CMD prompt on Linux

- Go your Linux home directory
- Use `mkdir` command to create the *source* directory to store NXP Wi-Fi driver source
- Use `cd` command to go to *source* directory
- Use `git clone` command to download the Wi-Fi driver source

```
$ git clone http://github.com/nxp-imx/mwifiex.git
```

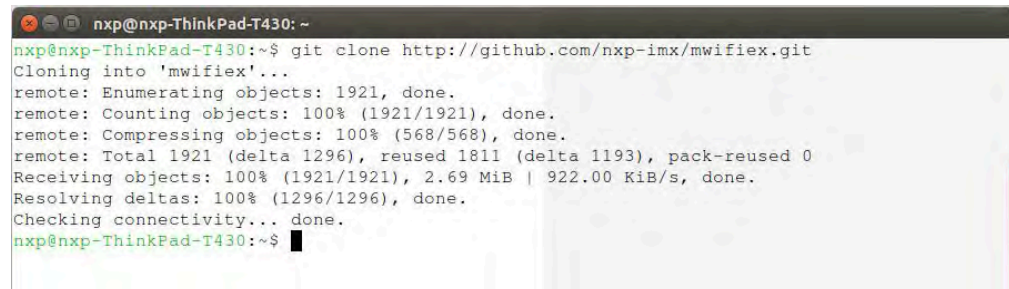


Figure 9. git-clone command output in Linux

Once the download completed, NXP Wi-Fi driver source is available in the *source* directory.

4 Download of firmware binaries

The firmware binary is available for download in GitHub. You can use either the command line or GitHub website to download the binaries.

4.1 Download using command line

- Follow the steps in [Section 3.1](#) or [Section 3.2](#) for Windows and [Section 3.3](#) for Linux to open the Git tool.
- Issue the `git clone` command to transfer the firmware binaries, and go to the directory with the new content:

```
# git clone https://github.com/NXP/imx-firmware.git
# cd imx-firmware/nxp/
```

4.2 Download using GitHub website

- Open a web browser and access *NXP/imx-firmware* page from [ref.\[2\]](#).

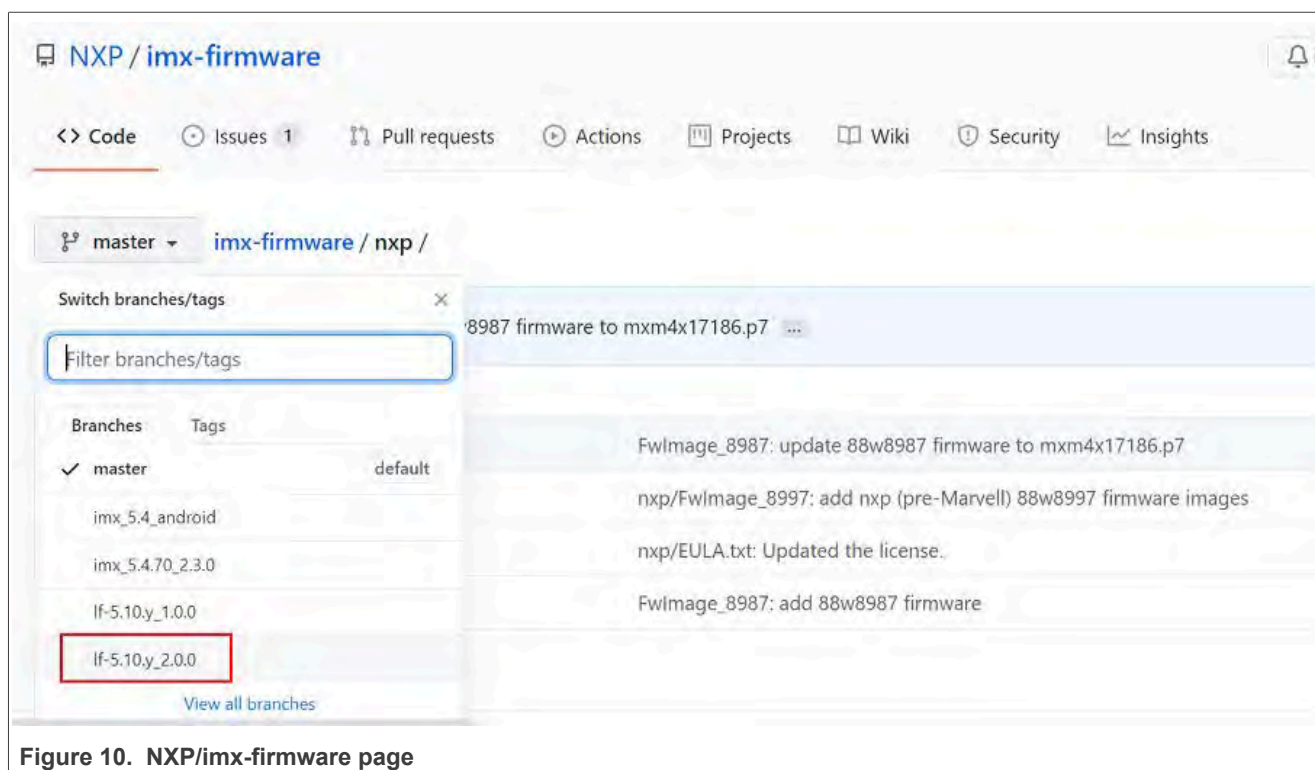


Figure 10. NXP/imx-firmware page

- Select the branch

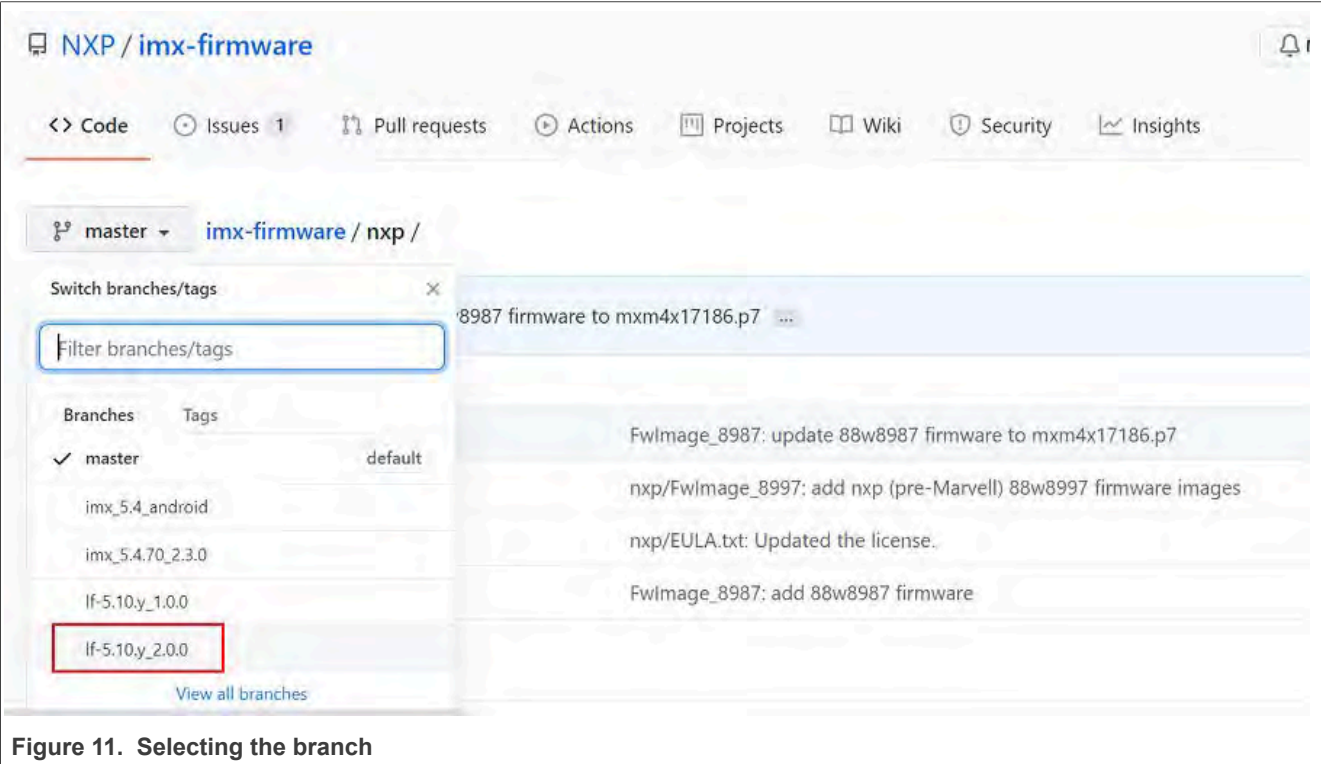


Figure 11. Selecting the branch

- Select the product for the firmware binary. [Figure 12](#) shows the example of 88W8997.

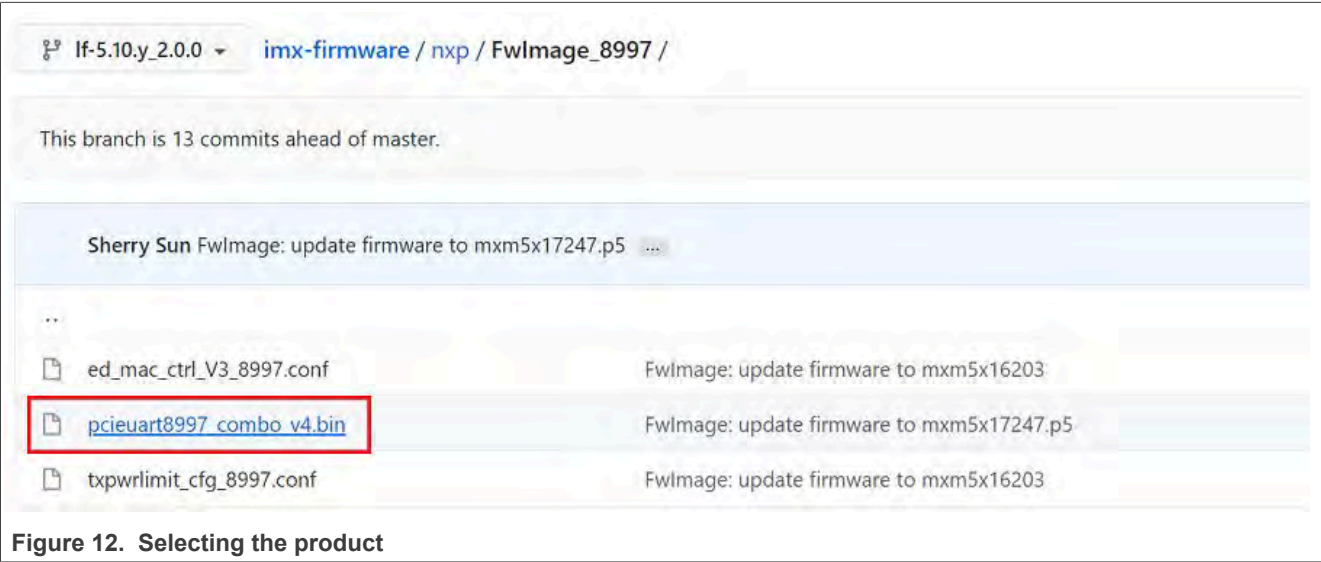


Figure 12. Selecting the product

- Click the **Download** button to initiate the firmware download

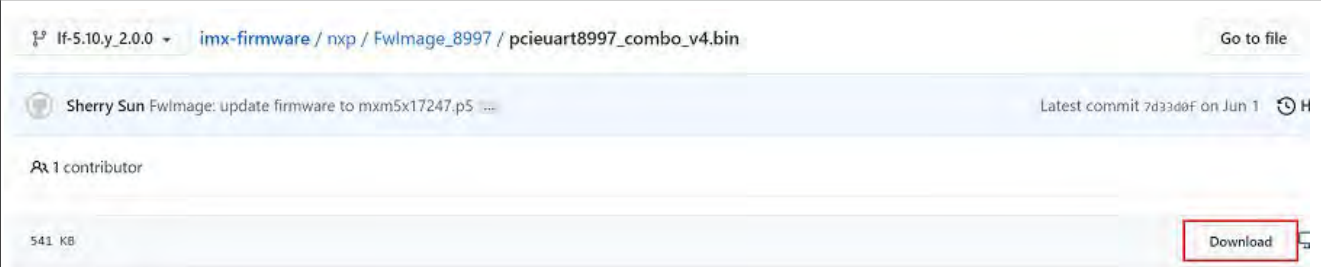


Figure 13. Initiating the firmware download

5 Build the driver for ARM64 platforms

This section describes the steps to build the driver package including the utility for ARM64 platforms.

The examples include building the driver for the following platforms:

- i.MX 8M evaluation platform (EVK) with Yocto build system
- Generic ARM64 platform

5.1 i.MX 8M evaluation platform with Yocto

This section shows how to generate a development SDK, build the toolchain, the driver, and utils binary.

Step 1 – Generate the development SDK

- Issue the command to generate an SDK from the Yocto Project build Yocto Project build environment

```
$ DISTRO=<distro name> MACHINE=<machine name> source  
imx-setup-release.sh -b <build dir>
```

- To set up the Yocto Project build environment, see [ref.\[1\]](#).
- For valid DISTRO options, see the section *Build configurations* in i.MX Yocto Project User's Guide (IMXLXOCTOUG).

Step 2 – Build the toolchain.

- Issue the command to build the toolchain from the *build* directory.

```
$ bitbake meta-toolchain-qt6
```

Step 3 – Install the toolchain and tools.

- Stay in the *build* directory.
- Execute the script located in *tmp/deploy/sdk/* directory.

```
$ sudo tmp/deploy/sdk/fsl-imx-wayland-glibc-x86_64-meta-toolchain-qt6-armv8a-imx8mqevk-  
toolchain-6.6-scarthgap.sh
```

Note: The name of the toolchain and/or script depends on your build machine and on the bitbake image.

- When prompted about the directory to install the toolchain, accept the default options.

The following message confirms the installation completion:

```
SDK has been successfully set up and is ready to be used
```

Step 4 – Set up the toolchain environment on the host terminal window.

- Issue the command to set up the toolchain environment on the host terminal window.

```
$ source <toolchain installed directory>/environment-setup-<toolchain script>
```

Example for i.MX 8M EVK:

```
$ source /opt/fsl-imx-wayland/6.6-scarthgap/environment-setup-armv8a-poky-linux
```

- Check that new environment variables are correctly set for the target i.MX 8 EVK.

```
$ echo $ARCH
arm64
$ echo $CROSS_COMPILE
aarch64-poky-linux
```

- Export *KERNELDIR* directory to build the compatible driver modules.
- Check that the Linux kernel version in *KERNELDIR* directory matches the Linux kernel version on the target i.MX 8M EVK.

```
$ export KERNELDIR=/<absolute directory path to yocto build>/tmp/work/imx8mqevk-poky-
linux/linux-imx/6.6.52+git/build/
```

Step 5 – Build the driver and utils.

- Issue the `cd` command to go to the directory with the driver source file.

```
$ cd <driver source wlan_src directory>
```

- Issue the `make` command to build the driver.

```
$ make clean
$ make build
```

The *bin_wlan* directory is created outside *wlan_src*. *bin_wlan* includes the driver and utils binary.

5.2 Build for ARM64 generic platforms

To build drivers for generic ARM64 platforms, first set up the toolchain for the generic ARM64 platform.

Refer to your platform BSP build environment setup guide for the steps and procedure.

Once the toolchain installation and setup are complete, follow the steps below to build the driver and utils.

Step 1 – Export *KERNELDIR* directory

- Export *KERNELDIR* directory to build the compatible driver modules.
- Check that the Linux kernel version in *KERNELDIR* directory path matches the Linux kernel version on your ARM64 platform

```
$ export KERNELDIR=/absolute path to kernel build directory/
```

Step 2 – Build the driver and utils binary

- Issue the `cd` command to go to the directory with the driver source file

```
$ cd <driver source wlan_src directory>
```

- Issue the `make` command to build the driver

```
$ make clean  
$ make build
```

The *bin_wlan* directory is created outside *wlan_src* directory. *bin_wlan* directory includes the driver and utils binary.

6 References

- [1] User guide – IMXLXYOCTOUG: i.MX Yocto Project User's Guide ([link](#))
- [2] NXP GitHub – *nxp* directory with firmware images ([link](#))
- [3] NXP GitHub – MXM Wi-Fi Driver ([link](#))
- [4] Webpage – Git – Download for Windows ([link](#))

7 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 2021-2022, 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.

8 Revision history

Revision history

Document ID	Date	Description
UM11675 v.4.0	30 June 2025	<ul style="list-style-type: none">• Section 5.1 "i.MX 8M evaluation platform with Yocto":<ul style="list-style-type: none">– Updated the command to build the toolchain (step 2).– Updated the command to install the toolchain (step 3).– Updated the example of command for i.MX 8M EVK in step 4.– Updated the command to export <i>KERNELDIR</i> directory in step 4.• Section 6 "References": added.
UM11675 v.3.0	12 October 2022	<ul style="list-style-type: none">• Section 1 "Overview": updated• Section 3.1 "Using Git GUI on Windows": replaced the URL of the source location, Figure 4, and Figure 5• Section 3.2 "Using Git CMD on Windows": updated the <code>git clone</code> command to download NXP Wi-Fi driver source, and replaced Figure 7• Section 3.3 "Using Git CMD on Linux": replaced the URL in the <code>git-clone</code> command to download the Wi-Fi driver source and replaced the two figures
UM11675 v.2.0	29 March 2022	<ul style="list-style-type: none">• Updated the document title• Section 5 "Build the driver for ARM64 platforms": added
UM11675 v.1.0	21 September 2021	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.

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.

Figures

Fig. 1.	Git installation on Windows	3	Fig. 7.	git clone command output	6
Fig. 2.	Git CMD and Git GUI in Windows Explorer	3	Fig. 8.	Git CMD prompt on Linux	7
Fig. 3.	Clone existing repository	4	Fig. 9.	git-clone command output in Linux	7
Fig. 4.	Capturing the source location and target directory	4	Fig. 10.	NXP/imx-firmware page	8
Fig. 5.	NXP Wi-Fi driver source download in progress	5	Fig. 11.	Selecting the branch	9
Fig. 6.	User home directory in Git CMD	6	Fig. 12.	Selecting the product	9
			Fig. 13.	Initiating the firmware download	10

Contents

1 Overview2

2 Installing Git3

2.1 Installing Git on Linux3

2.2 Installing Git on Windows3

3 Download of NXP Wi-Fi driver source4

3.1 Using Git GUI on Windows4

3.2 Using Git CMD on Windows6

3.3 Using Git CMD on Linux7

4 Download of firmware binaries8

4.1 Download using command line8

4.2 Download using GitHub website8

5 Build the driver for ARM64 platforms11

5.1 i.MX 8M evaluation platform with Yocto11

5.2 Build for ARM64 generic platforms13

6 References14

7 Note about the source code in the document15

8 Revision history16

Legal information17

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