

AN14748

How to Run Application on M7 Core of i.MX 95

Rev. 1.0 — 31 July 2025

Application note

Document information

Information	Content
Keywords	AN14748, i.MX 95, M7 Core, MCUXpresso SDK, Bare Metal Application, freeRTOS
Abstract	This document details the steps to get the toolchain, SDK, compile example source code, and create the image to run on the M7 core of i.MX 95.



1 Introduction

This document describes the steps to get the toolchain, SDK, compile example source code, and create the image to run on the M7 core of i.MX 95.

The steps are verified with i.MX Linux 6.12.20_2.0.0 release and SDK_25.06.00. The software is compiled on an Ubuntu host machine.

2 Toolchain setup

1. Link to the toolchain: https://developer.arm.com/-/media/Files/downloads/gnu/13.3.rel1/binrel/arm-gnu-toolchain-13.3.rel1-x86_64-arm-none-eabi.tar.xz
2. Set up the toolchain in the ~/project directory:

```
mkdir ~/project && cd ~/project
tar xvf arm-gnu-toolchain-13.3.rel1-x86_64-arm-none-eabi.tar.xz
export ARMGCC_DIR=~/project/arm-gnu-toolchain-13.3.rel1-x86_64-arm-none-eabi
```

3 Getting started

To download MCUXpresso-SDK, perform the following steps:

1. Go to the MCUXpresso-SDK webpage: [MCUXpresso-SDK](#).
2. Click **DOWNLOADS**. To download the software, choose the **MCUXpresso SDK - SDK Builder** option, which launches the MCUXpresso SDK Builder webpage.
3. On the MCUXpresso SDK Builder webpage, perform the following steps:
 - a. Click **Select Development Board**.
 - b. In the **Search for Hardware** field, type "i.MX 95".
 - c. From the search results, select a board—for example, IMX95LPD5EVK-19.
 - d. At the bottom of the page, locate the version number (for example, **25.06.00**) from the dropdown, and click **BUILD SDK**.
 - e. Keep the default settings:
 - **Host OS**: Linux
 - All packages selectedThen click **BUILD SDK** again.
 - f. The first item listed must be **SDK_25.06.00_IMX95LPD5EVK-19**.
 - g. Click **Download > Download SDK archive including documentation (18 MB) > AGREE**.
 - h. The file **SDK_25_06_00_IMX95LPD5EVK-19.tar.gz** downloads.

4 How to build imx-mkimage from BSP

The imx-mkimage tool is used to create a bootable image flash.bin. The imx-mkimage can be built and extracted from Linux BSP.

To build Linux BSP, install some packages to set up host. For more details, refer to *i.MX Yocto Project User's Guide* (document [UG10164](#)).

To install packages on Ubuntu 22.04, use the following command:

```
sudo apt install gawk wget diffstat unzip texinfo gcc build-essential chrpath
socat cpio python3 python3-pip python3-pexpect xz-utils debianutils iputils-
```

```
ping python3-git python3-jinja2 python3-subunit zstd liblz4-tool file locales  
libacl1 curl repo cma
```

To build the imx-mkimage tool, follow the steps below:

```
mkdir ~/bin (this step may not be needed if the bin folder already exists)  
curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo  
chmod a+x ~/bin/repo  
export PATH=~/bin:$PATH  
cd ~/project  
mkdir IMX_YOCTO_BSP && cd IMX_YOCTO_BSP  
repo init -u https://github.com/nxp-imx/imx-manifest -b imx-linux-walnascar \  
-m imx-6.12.20-2.0.0.xml  
repo sync  
MACHINE=imx95-19x19-lpddr5-evk DISTRO=fsl-imx-xwayland source \  
imx-setup-release.sh -b build-imx95-19x19-lpddr5-evk  
#Press SPACE continuously then answer y  
bitbake imx-boot
```

In Yocto, it is called imx-boot instead of imx-mkimage. After the compilation completes, the tool has all the necessary images and it resides in build-imx95-19x19-lpddr5-evk/tmp/work/imx95_19x19_lpddr5-evk-poky-linux/imx-boot/1.0/git/:

```
cp -rp tmp/work/imx95_19x19_lpddr5-evk-poky-linux/imx-boot/1.0/git/ ~/project/  
imx-boot
```

5 How to compile the SDK

To compile the SDK, perform the steps below:

1. To compile the Bare Metal hello_world, run the following commands:

```
cd ~/project  
mkdir SDK_25_06_00 && cd SDK_25_06_00  
tar xvf ../SDK_25_06_00_IMX95LPD5EVK-19.tar.gz  
cd ~/project/SDK_25_06_00/boards/imx95lpd5evk19/demo_apps/hello_world/cm7/  
armgcc  
./clean.sh  
./build_all.sh
```

Both images that run on ITCM and DDR are created. They are available in:

- ITCM: release/hello_world.bin
- DDR: ddr_release/hello_world.bin

2. To compile the freeRTOS hello_world, run the following commands:

```
cd ~/project/SDK_25_06_00/boards/imx95lpd5evk19/freertos_examples/  
freertos_hello/cm7/armgcc  
./clean.sh  
./build_all.sh
```

The created M7 image is available in:

- ITCM: release/freertos_hello.bin

Note: The freeRTOS only creates an image for running from ITCM.

To create an image for DDR:

- a. Use MIMX9596xxxxN_cm7_ddr_ram.ld in ~/project/SDK_25_06_00/boards/imx95lpd5evk19/demo_apps/hello_world/cm7/armgcc to replace MIMX9596xxxxN_cm7_ram.ld in ~/project/SDK_25_06_00/boards/imx95lpd5evk19/freertos_examples/freertos_hello/cm7/armgcc, using name MIMX9596xxxxN_cm7_ram.ld
- b. Then, run build_all.sh.
The created M7 image for DDR is in:
 - DDR: release/freertos_hello.bin

6 Generate a bootable image for running M7 from ITCM

To build the flash.bin to run from ITCM, use the following commands:

```
#from armgcc directory in the previous step
cp release/hello_world.bin ~/project/imx-boot/iMX95/m7_image.bin
or
cp release/freertos_hello.bin ~/project/imx-boot/iMX95/m7_image.bin
cd ~/project/imx-boot

make SOC=iMX95 REV=B0 flash_all LPDDR_TYPE=lpddr5 OEI=YES

iMX95/flash.bin is the bootable image that can be programmed to flash/sd/emmc.
```

7 Generate a bootable image for running M7 from DDR

To build the flash.bin for running from DDR, use the following commands:

```
#from armgcc directory in the compilation step
cp ddr_release/hello_world.bin ~/project/imx-boot/iMX95/m7_image.bin
or
cp release/freertos_hello.bin ~/project/imx-boot/iMX95/m7_image.bin
cd ~/project/imx-boot
make SOC=iMX95 REV=B0 flash_all_ddr LPDDR_TYPE=lpddr5 OEI=YES
```

8 Program flash.bin to i.MX 95

To program flash.bin to the board, perform the following steps:

1. Download and install the flash programming tool UUU:
The UUU tool can be downloaded from [here](#). On the right side of the page, click the latest release. Then download uuu.exe for your Windows machine.
2. Create a folder c:\UUU and move uuu.exe to c:\UUU.
3. Connect both USB-C ports to a Windows machine.
4. Consoles:
 - a. Start three TeraTerm consoles.
 - b. Connect 1st, 3rd, and 4th COM port.
 - c. Consider COM27, COM28, COM29, COM30, and use COM27/29/30.

```
- COM27: M7
- COM29: A55 U-boot/Linux
- COM30: M33 System Manager
```

5. Set the boot_mode to the serial download:

```
SW7[1:4] = 1001 (ON OFF OFF ON)
```

6. In a Windows machine, copy flash.bin to the c:\UUU directory.
7. In a Windows machine, open the DOS command in the UUU directory, and run the following command:
- For SD programming:

```
C:\UUU> uuu -b sd flash.bin
```

- For eMMC programming:

```
C:\UUU> uuu -b emmc flash.binc
```

Power cycle the board. UUU must start to program the flash.bin to SD card or eMMC.

8. Set the boot_mode:

- Boot from SD:

```
SW7[1:4] = 1011 (ON OFF ON ON)
```

- Boot from eMMC:

```
SW7[1:4] = 1010 (ON OFF ON OFF)
```

9. Power cycle the board. COM27 console must display the message from the hello_world program.

9 Note about the source code in the document

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

Copyright 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.

10 Revision history

[Table 1](#) summarizes the revisions to this document.

Table 1. Revision history

Document ID	Release date	Description
AN14748 v.1.0	31 July 2025	Initial public release

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.

Amazon Web Services, AWS, the Powered by AWS logo, and FreeRTOS
— are trademarks of Amazon.com, Inc. or its affiliates.

Contents

1 Introduction 2

2 Toolchain setup 2

3 Getting started 2

4 How to build imx-mkimage from BSP 2

5 How to compile the SDK 3

6 Generate a bootable image for running
M7 from ITCM 4

7 Generate a bootable image for running
M7 from DDR 4

8 Program flash.bin to i.MX 95 4

9 Note about the source code in the
document 5

10 Revision history 5

Legal information 7

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