Infineon high-reliability Rad-Hard NOR flash for AMD Versal configuration memory (v2025.1 and v2025.2)

About this document

Scope and purpose

This guide provides information and instructions to enable Infineon high-reliability radiation-hardened (Rad-Hard/RH) NOR flash as a configuration memory device for AMD Versal ACAP in a local user environment.

***Note**:* *This guide supports system integrators in enabling and using flash memory devices with the AMD toolchain releases 2025.1 and 2025.2 for the products described in* Table 1*.*

Intended audience

This document is intended for design engineers, technicians, developers, and system integrators of electronic systems who have a basic understanding of the Versal ACAP specification, the AMD Versal evaluation kit, and AMD development tools. It describes how to program a boot image onto Infineon NOR flash and focuses on enabling Infineon NOR flash for use with AMD Versal ACAP, which is not currently supported by AMD tools.

Important notice

“Evaluation Boards and Reference Boards” shall mean products embedded on a printed circuit board (PCB) for demonstration and/or evaluation purposes, which include, without limitation, demonstration, reference and evaluation boards, kits and design (collectively referred to as “Reference Board”).

Environmental conditions have been considered in the design of the Evaluation Boards and Reference Boards provided by Infineon Technologies. The design of the Evaluation Boards and Reference Boards has been tested by Infineon Technologies only as described in this document. The design is not qualified in terms of safety requirements, manufacturing and operation over the entire operating temperature range or lifetime.

The Evaluation Boards and Reference Boards provided by Infineon Technologies are subject to functional testing only under typical load conditions. Evaluation Boards and Reference Boards are not subject to the same procedures as regular products regarding returned material analysis (RMA), process change notification (PCN) and product discontinuation (PD).

Evaluation Boards and Reference Boards are not commercialized products, and are solely intended for evaluation and testing purposes. In particular, they shall not be used for reliability testing or production. The Evaluation Boards and Reference Boards may therefore not comply with CE or similar standards (including but not limited to the EMC Directive 2004/EC/108 and the EMC Act) and may not fulfill other requirements of the country in which they are operated by the customer. The customer shall ensure that all Evaluation Boards and Reference Boards will be handled in a way which is compliant with the relevant requirements and standards of the country in which they are operated.

The Evaluation Boards and Reference Boards as well as the information provided in this document are addressed only to qualified and skilled technical staff, for laboratory usage, and shall be used and managed according to the terms and conditions set forth in this document and in other related documentation supplied with the respective Evaluation Board or Reference Board.

It is the responsibility of the customer’s technical departments to evaluate the suitability of the Evaluation Boards and Reference Boards for the intended application, and to evaluate the completeness and correctness of the information provided in this document with respect to such application.

The customer is obliged to ensure that the use of the Evaluation Boards and Reference Boards does not cause any harm to persons or third party property.

The Evaluation Boards and Reference Boards and any information in this document is provided "as is" and Infineon Technologies disclaims any warranties, express or implied, including but not limited to warranties of non-infringement of third party rights and implied warranties of fitness for any purpose, or for merchantability.

Infineon Technologies shall not be responsible for any damages resulting from the use of the Evaluation Boards and Reference Boards and/or from any information provided in this document. The customer is obliged to defend, indemnify and hold Infineon Technologies harmless from and against any claims or damages arising out of or resulting from any use thereof.

Infineon Technologies reserves the right to modify this document and/or any information provided herein at any time without further notice.

Table of contents

About this document . . . 1
Important notice . . . 2
Table of contents . . . 3
1 Introduction . . . 4
2 Prerequisites . . . 6
2.1 Host machine . . . 6
2.2 Hardware . . . 6
2.3 AMD software tools . . . 7
2.4 Infineon software tools . . . 8
3 BOOT.BIN and u-boot.bin build . . . 10
3.1 Obtaining AMD PLM sources and Infineon patch integration . . . 10
3.2 Obtaining AMD U-Boot sources and Infineon patch integration . . . 10
3.3 Compiling U-Boot and PLM for BOOT.BIN using PetaLinux . . . 11
3.4 OSL flow for compiling U-Boot to generate u-boot.bin . . . 13
4 Program configuration memory using JTAG boot . . . 15
4.1 Set up and boot from JTAG on Versal EK . . . 15
4.2 Programming configuration memory in Vivado . . . 15
4.3 Programming configuration memory in Vitis . . . 20
5 Program configuration memory with SD boot . . . 24
5.1 Setting up the SD card on the Versal EK board . . . 24
5.2 Booting from the SD card for programming . . . 24
6 Boot from Configuration Memory . . . 27
References . . . 28
Revision history . . . 29
Disclaimer . . . 30

1 Introduction

This guide helps system integrators support the Infineon NOR flash memories listed in Table 1 by using the AMD toolchain releases 2025.1 and 2025.2.

***Note**: The instructions in this guide have been verified by Infineon as part of the AMD vendor-supported flash verification program.*

Table 1 Supported Infineon NOR flash products for the AMD toolchain

Density	Product	Interface	Voltage	Family
512 Mb	5962F2122202VYC	Quad SPI	3.3 V	Rad-Hard NOR flash
	5962F2122201PXE	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYRS17B512-133UZMB[1)]	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYRS17B512-133AZMB[1)]	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B512-133UZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B512-133AZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYEL17B512-133AZM	Quad SPI	3.3 V	Rad-Hard NOR flash
1 Gb	CYRS17B01G-133UZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYRS17B01G-133AZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B01G-133UZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B01G-133AZMB	Quad SPI	3.3 V	Rad-Hard NOR flash
	CYEL17B01G-133AZM	Quad SPI	3.3 V	Rad-Hard NOR flash
2 Gb	CYRS17B02G-133UZMB	Dual Quad SPI	3.3 V	Rad-Hard NOR flash
	CYRS17B02G-133NZMB	Dual Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B02G-133UZMB	Dual Quad SPI	3.3 V	Rad-Hard NOR flash
	CYPT17B02G-133NZMB	Dual Quad SPI	3.3 V	Rad-Hard NOR flash
	CYEL17B02G-133NZM	Dual Quad SPI	3.3 V	Rad-Hard NOR flash

1) The CYRS part number has been superseded by the 5962-DLAM part number.

Unless otherwise noted, this guide uses factory default settings for the Infineon NOR flash memories listed in Table 1.

***Notes**:*

1. *For dual devices in a parallel memory configuration, the `CFR1N[1] QUADIT` bit of both high-reliability radiation-hardened NOR flash devices must be set to 1 to enable 4x Quad mode*
2. *For dual-die package devices that are 1 Gb, the `CFR1N[0] AD34_NV` bit of the high-reliability radiation-hardened NOR flash devices must be set to 1 to enable 4-byte address mode*
3. *For the 2 Gb device with two Quad SPI dies (Quad-1 and Quad-2) in parallel, with two chip selects, and 8-bit I/O, only a dual parallel x8 memory configuration is supported.*

This guide describes two approaches that require specific boot capabilities of AMD Versal ACAP:

- **JTAG boot**: Applicable to Versal platforms that support JTAG boot capability. For additional details, see Program configuration memory using JTAG boot
- **SD boot**: Applicable to Versal platforms that support SD boot capability. For additional details, see Program configuration memory with SD boot

Depending on the boot capability of the target Versal platform, you can use either the SD boot or JTAG boot approach. For information related to the high-reliability radiation-hardened NOR flash family, contact TYO-RHSONOS-patches-versal@infineon.com.

2 Prerequisites

This section describes the required hardware and software prerequisites.

2.1 Host machine

- 8 GB RAM is recommended as the minimum requirement for AMD tools
- 2 GHz CPU clock or equivalent with a minimum of eight cores
- 100 GB of free HDD space

Supported operating systems

- CentOS and RHEL are not supported and have been removed to align with upstream Yocto
- Ubuntu Desktop or Server 22.04.2 LTS[1)], 22.04.3 LTS, 22.04.4 LTS, and 22.04.5 LTS
- openSUSE Leap 15.4
- AlmaLinux 8.10, 9.4, and 9.5

2.2 Hardware

This guide is based on the following hardware configuration, which has been validated by Infineon to enable the Infineon NOR flash devices listed in Table 1. The guide is also applicable to user-designed Versal platforms on which Infineon NOR flash is installed. As a result, the following hardware modules are not mandatory to use this guide:

- AMD Versal Evaluation Kit, including EK-VMK180 or EK-VCK190
- Infineon memory module

The Infineon memory module consists of two stacked printed circuit boards, as shown in Figure 1. The PISMO card hosts the Infineon NOR flash device, either soldered or socketed. The converter card connects the PISMO card, using the Infineon connector definition, to the fast-boot module mezzanine card connector on the Versal Evaluation Kit board as shown in Figure 2.

- **PISMO card**:
  - Radiation-hardened NOR flash PISMO `INFINEON-22-P-00306-REV1P`
  - Radiation-hardened NOR flash PISMO for 2Gb `INFINEON-22-P-00309-REV1P0`
- **Converter card**: Quad PISMO adapter for Versal Evaluation Kit `INFINEON-0324-257-REV0`

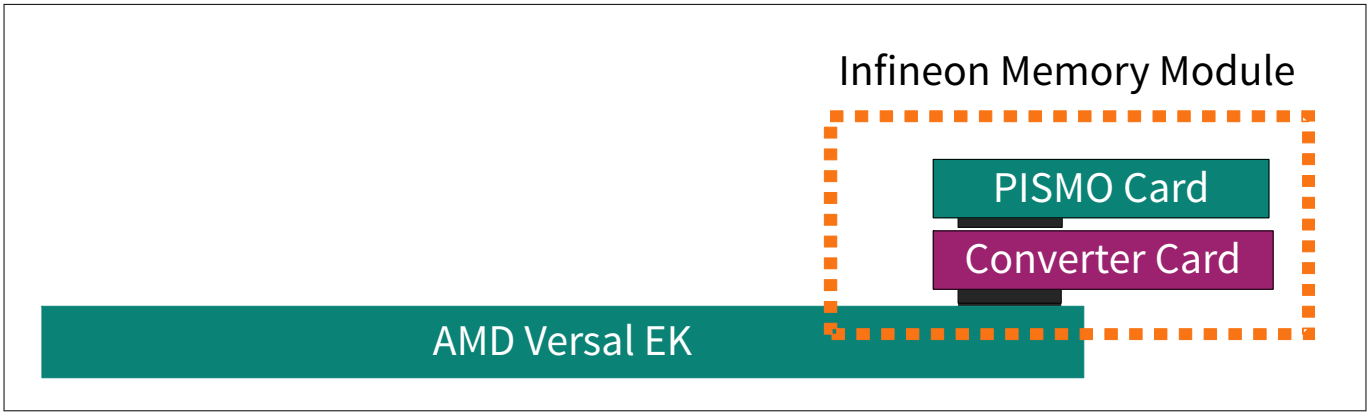


Figure 1 Infineon memory module on Versal EK

1 Applicable to version 2025.1 only.

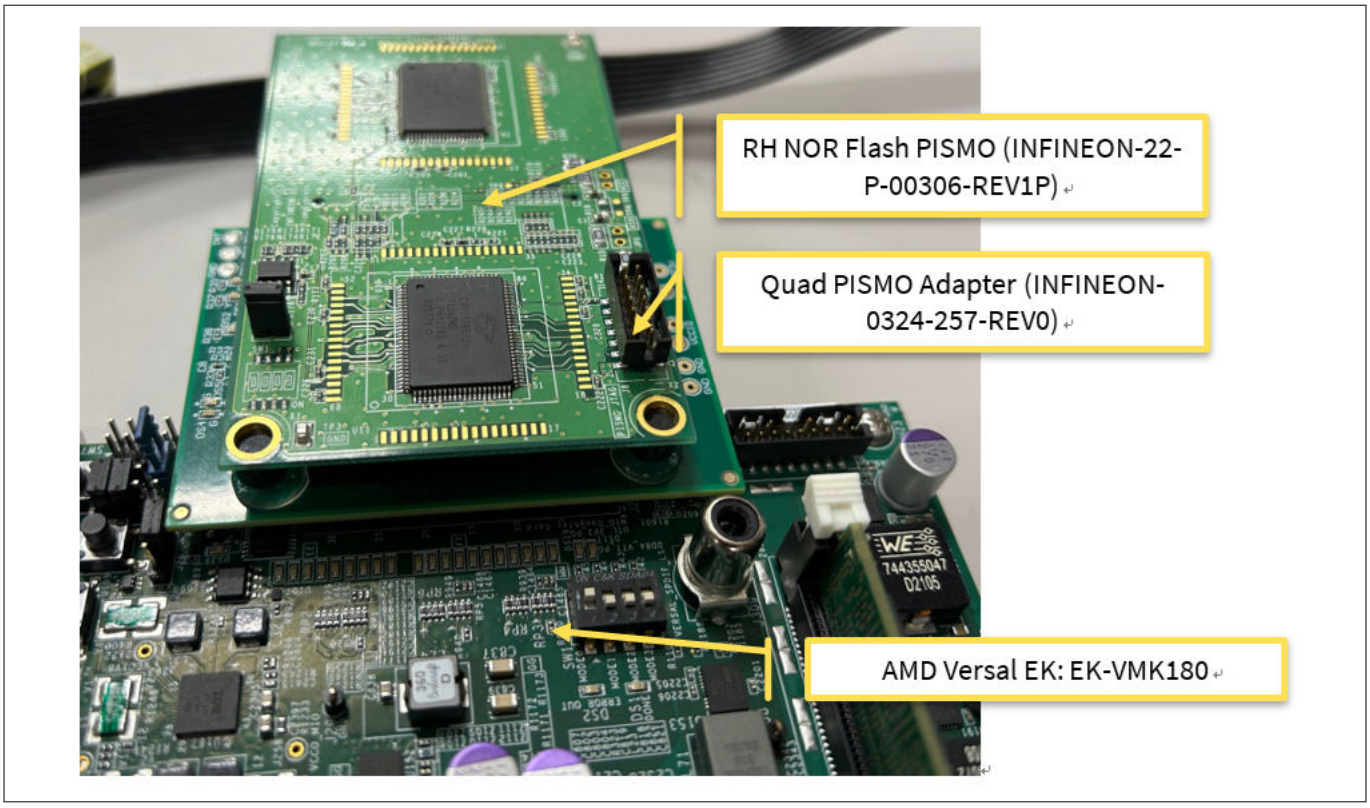


Figure 2 Infineon Rad-Hard NOR flash installed on Versal EK

2.3 AMD software tools

Download and install the following software tools on the host machine:

- **Version 2025.1**:
  - PetaLinux Tools 2025.1 (AMD DL site)
  - PetaLinux BSP for VMK180 or VCK190 (AMD DL site)
  - Vitis Core Development Kit 2025.1 (AMD DL site)
  - Vivado Edition 2025.1 (AMD DL site)
  - AMD Platform Loader and Manager (PLM) Git repository (xlnx_rel_v2025.1)
  - AMD U-Boot Git repository (xlnx_rebase_v2025.01_2025.1)
  - AMD Linux Git repository (xlnx_rebase_v6.12_2025.1)
- **Version 2025.2**:
  - PetaLinux Tools 2025.2 (AMD DL site)
  - PetaLinux BSP for VMK180 or VCK190 (AMD DL site)
  - Vitis Core Development Kit 2025.2 (AMD DL site)
  - Vivado Edition 2025.2 (AMD DL site)
  - AMD PLM Git repository (xlnx_rel_v2025.2)
  - AMD U-Boot Git repository (xlnx_rebase_v2025.01_2025.2)
  - AMD Linux Git repository (xlnx_rebase_v6.12_2025.2)

2.4 Infineon software tools

The following tools and patches are required to enable the Infineon NOR flash devices listed in Table 1. The configuration memory programming tool is required for the SD boot approach described in Program configuration memory with SD boot.

- High-reliability radiation-hardened NOR flash patch support for AMD Versal tools
  - Infineon configuration memory programming tool: `ProgConfigMem.scr`
  - Infineon patches for high-reliability radiation-hardened NOR flash

The high-reliability radiation-hardened NOR flash patches are listed in Table 2. The high-reliability radiation-hardened NOR flash patch support for AMD Versal tools and the configuration memory programming tool are available for download from the Infineon Developer Center. This document of Rev *A corresponds to the highreliabilityradhardflashpatchamdversaltool_2025.1_2025.2_Linux_arm_r2.zip on the Infineon Developer Center.

In the Patch number column of Table 2, the abbreviations UB, LX, and ES refer to U-Boot, Linux, and Embedded SW, respectively.

Table 2 Infineon patch for high-reliability RH NOR flash

Patch number	Patch subject	Patch file for AMD tools 2025.1/2025.2	Single x4	Dual parallel x8
UB-001	Enable SFDP support in defconfig (UB-001)	`u-boot-xlnx-qspi-v2025.2-UB001-cyrs17b.patch`	x	x
UB-002	Add support SPI-NOR ID table for CYRS17B(UB-002)	`u-boot-xlnx-qspi-v2025.2-UB002-cyrs17b.patch`	x	x
UB-003	Add support fixup APIs for CYRS17B(UB-003)	`u-boot-xlnx-qspi-v2025.2-UB003-cyrs17b.patch`	x	x
UB-004	Add support Device requiring Dummy Cycle for RDID (UB-004)	`u-boot-xlnx-qspi-v2025.2-UB004-cyrs17b.patch`		x
UB-005	Set data.nbytes of RDAR to 2 for parallel mode (UB-005)	`u-boot-xlnx-qspi-v2025.2-UB005-cyrs17b.patch`		x
UB-006	Disable stripe mode for WRAR in parallel mode (UB-006)	`u-boot-xlnx-qspi-v2025.2-UB006-cyrs17b.patch`		x
UB-007	Add support 4 byte address mode for CYRS17B (UB-007)	`u-boot-xlnx-qspi-v2025.2-UB007-cyrs17b.patch`	x	x
UB-008	Add support for CYRS17B2G (UB-008)	`u-boot-xlnx-qspi-v2025.2-UB008-cyrs17b.patch`		x
LX-001	Add support fixup APIs and SPI-NOR ID table for CYRS17B(LX-001)	`linux-xlnx-qspi-v2025.2-LX001-cyrs17b.patch`	x	x
LX-002	Add support Device requiring dummy cycle for RDID (LX-002)	`linux-xlnx-qspi-v2025.2-LX002-cyrs17b.patch`	x	x
LX-003	Set data.nbytes of RDAR to 2 for parallel mode (LX-003)	`linux-xlnx-qspi-v2025.2-LX003-cyrs17b.patch`		x

(table continues...)

Table 2 (continued) Infineon patch for high-reliability RH NOR flash

Patch number	Patch subject	Patch file for AMD tools 2025.1/2025.2	Single x4	Dual parallel x8
LX-004	Disable stripe mode for WRAR in parallel mode (LX-004)	`linux-xlnx-qspi-v2025.2-LX004-cyrs17b.patch`	x	x
LX-005	Add support for CYRS17B2G (LX-005)	`linux-xlnx-qspi-v2025.2-LX005-cyrs17b.patch`		x
ES-001	Add support for CYRS17B for QSPI boot (ES-001)	`plm-qspi-v2025.2-ES001-cyrs17b.patch`	x	x
ES-002	Add support for Device requiring mode bits for FAST_READ (ES-002)	`plm-qspi-v2025.2-ES002-cyrs17b.patch`	x	x
ES-003	Add support for CYRS17B2G for QSPI Boot (ES-003)	`plm-qspi-v2025.2-ES003-cyrs17b.patch`		x
ES-004	Add support for QUAD bit for MCP and stripe flag in parallel mode (ES-004)	`baremetal-qspi-v2025.2-ES004-cyrs17b.patch`	x	x
ES-005	Add support for CYRS17B (ES-005)	`baremetal-qspi-v2025.2-ES005-cyrs17b.patch`	x	x
ES-006	Add support for SR1 using RDAR for CYRS17B (ES-006)	`baremetal-qspi-v2025.2-ES006-cyrs17b.patch`	x	x

3 BOOT.BIN and u-boot.bin build

This section describes the method for building `BOOT.BIN` and `u-boot.bin` for version 2025.2. The same method is applicable to version 2025.1 by updating the AMD Git repository URLs defined in AMD software tools.

3.1 Obtaining AMD PLM sources and Infineon patch integration

Run the following command in the host machine terminal. After execution, the `amd-plm` directory contains the AMD PLM sources with support for the target Infineon NOR flash. The sources are ready to be compiled.

Commands for AMD PLM sources

```
mkdir amd-plm

cd amd-plm

git clone -b xlnx_rel_v2025.2 https://github.com/Xilinx/embeddedsw.git

cd embeddedsw

## Apply Infineon patch supporting your target device

## For HiRel RH NOR Flash
git apply plm-qspi-v2025.2-ES001-cyrs17b.patch
git apply plm-qspi-v2025.2-ES002-cyrs17b.patch
git apply plm-qspi-v2025.2-ES003-cyrs17b.patch
```

3.2 Obtaining AMD U-Boot sources and Infineon patch integration

Run the following command in the host machine terminal. After execution, the `amd-u-boot` directory contains the AMD U-Boot sources with support for the target Infineon NOR flash. The sources are ready to be compiled.

Commands for AMD U-Boot sources

```
mkdir amd-u-boot

cd amd-u-boot

git clone -b xlnx_rebase_v2025.01_2025.2 https://github.com/Xilinx/u-boot-xlnx.git

cd u-boot-xlnx

## Apply Infineon patch supporting your target device

## For High Reliability RH NOR flash
git apply u-boot-xlnx-qspi-v2025.2-UB001-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB002-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB003-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB004-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB005-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB006-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB007-cyrs17b.patch
git apply u-boot-xlnx-qspi-v2025.2-UB008-cyrs17b.patch
```

3.3 Compiling U-Boot and PLM for BOOT.BIN using PetaLinux

The PetaLinux tool is used to compile U-Boot and PLM from a local source directory that contains the cloned sources.

Using PetaLinux to compile U-Boot and PLM for BOOT.BIN

```
## Source petalinux tool
source petalinux_tool_installed_path/petalinux/2025.2/settings.sh

## For creating project from VMK180 bsp file (VMK180 bsp file is dual parallel x8 memory config
only)
petalinux-create -t project -s <path_to_bsps>/xilinx-vmk180-v2025.2-11160223.bsp

## For creating project from VMK180 xsa file (When configuring the H/W settings in Vivado, use
this method.)
petalinux-create -t project --template versal --force -n xilinx-vmk180-v2025.2 cd xilinx-vmk180-
v2025.2 petalinux-config --get-hw-description=<path to xsa directory> --silentconfig

## Change the directory to created project path
cd xilinx-vmk180-v2025.2/

## To compile u-boot and plm (v2025.1) with local source, open and edit config as shown below
lines
vim project-spec/configs/config

  CONFIG_SUBSYSTEM_COMPONENT_PLM_NAME_EXT__LOCAL__SRC=y
  # CONFIG_SUBSYSTEM_COMPONENT_PLM_NAME_PLM is not set
  CONFIG_SUBSYSTEM_COMPONENT_PLM_NAME_EXT_LOCAL_SRC_PATH="<plm_local_repo_full_path>"

  CONFIG_SUBSYSTEM_COMPONENT_U__BOOT_NAME_EXT__LOCAL__SRC=y
  # CONFIG_SUBSYSTEM_COMPONENT_U__BOOT_NAME_U__BOOT__XLNX is not set
  CONFIG_SUBSYSTEM_COMPONENT_U__BOOT_NAME_EXT_LOCAL_SRC_PATH="<u-boot_local_repo_full_path>"

## To compile plm (v2025.2) with local source, open and edit config as shown below lines
vim project-spec/meta-user/conf/petalinuxbsp.conf

  EXTERNALSRC:pn-embeddedsw-source-${ESW_VER} = "<plm_local_repo_full_path>"

## Build petalinux project
petalinux-config --silentconfig
petalinux-build

## To update DTB, convert DTB file to DTS file and edit DTS file as show below lines.
dtc -I dtb -O dts -o images/system.dts images/system.dtb
vim images/system.dts

  # For High Reliability RH NOR flash
  spi@f1030000 {
  flash@ {
    # Add the following parameters in flash@{} (e.g., cyrs17b01g)
    # For single x4 memory config
    compatible = “cyrs17b01g\0jedec,spi-nor”;
    reg = < 0x00>;

    # For dual parallel x8 memory config
    compatible = “cyrs17b01g\0jedec,spi-nor”;
    reg = < 0x00 0x01>;
```

```
        # Set the flash density size to match the flash density of the flash device
        parallel-memories = <0x00 0x8000000 0x00 0x8000000>; /* For example 128MB */

dtc -I dts -O dtb -o images/system.dtb images/system.dts
## To Generate BOOT.BIN for SD and QSPI Boot
petalinux-package --boot --force --u-boot --dtb images/linux/system.dtb
```

The generated `BOOT.BIN`, which includes support for the target Infineon NOR flash, is located in the `xilinx-vmk180-v2025.2/images/linux` directory. This `BOOT.BIN` enables the Versal platform to boot in SD and QSPI modes. The U-Boot image included in `BOOT.BIN` functions as the device driver for programming the Infineon NOR flash when using the SD boot approach.

3.4 OSL flow for compiling U-Boot to generate u-boot.bin

The Makefile-based Open Source Linux (OSL) flow enables compilation of U-Boot from a local source directory. The generated `u-boot.bin`, which includes support for the target Infineon NOR flash, is located in the `u-boot-xlnx` directory. The `u-boot.bin` image functions as a device driver for programming the Infineon NOR flash when using the JTAG boot approach.

The `u-boot.bin` image is generated for execution from the Versal on-chip memory (OCM). Therefore, initialization of external memory, such as DRAM, is not required for the JTAG boot approach.

Compiling U-Boot for `u-boot.bin` using the OSL flow

```
# Navigate to your local clone of AMD U-Boot with Infineon Patch
cd u-boot-xlnx

# Set the cross-compiler toolchain path and arch
export CROSS_COMPILE=aarch64-linux-gnu-
export ARCH=aarch64

## For High Reliability RH NOR flash , open and edit config as shown below lines, then make it
vim configs/xilinx_versal_mini_qspi_defconfig

  # For single x1 memory config
  CONFIG_DEFAULT_DEVICE_TREE="versal-mini-qspi-x1-single"

  # For single x4 memory config
  CONFIG_DEFAULT_DEVICE_TREE="versal-mini-qspi-single"

  # For dual parallel x8 memory config
  CONFIG_DEFAULT_DEVICE_TREE="versal-mini-qspi-parallel"

## For dual parallel x8 memory config, open and edit config as shown below lines, then make it
vim arch/arm/dts/versal-mini-qspi-parallel.dts

  # For flash density (Mbit) for dual parallel memory config.
  # Set the flash density size to match the flash density of the flash device
  parallel-memories = /bits/64 <0x8000000 0x8000000>; /* For example 128MB */

make xilinx_versal_mini_qspi_defconfig

# Build the images to generate u-boot.elf for JTAG Boot
make -j22

# Rename from u-boot.elf to u-boot.bin
mv u-boot.elf u-boot.bin
```

4 Program configuration memory using JTAG boot

This section describes a programming method using the JTAG boot mode. After connecting the JTAG interface, set the required environment variables for the AMD toolchain to use the generated `u-boot.bin` generated in OSL flow for compiling U-Boot to generate u-boot.bin instead of the default U-Boot binary bundled with the AMD tools. This configuration enables support for Infineon NOR flash in the AMD tools, as described in Set up and boot from JTAG on Versal EK.

After the environment is configured, you can use either Vivado or Vitis for programming, as described in the Programming configuration memory in Vivado and Programming configuration memory in Vitis sections of this guide. The programming process follows the standard procedures defined by the AMD tools and does not require any special handling for Infineon NOR flash.

4.1 Set up and boot from JTAG on Versal EK

1. Connect the USB Type-C cable between the host machine and the Versal EK board
2. Set the boot mode of the Versal EK board to JTAG mode
3. Power on the Versal EK board
4. Verify that the JTAG connection to the Versal device is established in the Vivado Hardware window as shown in Figure 3

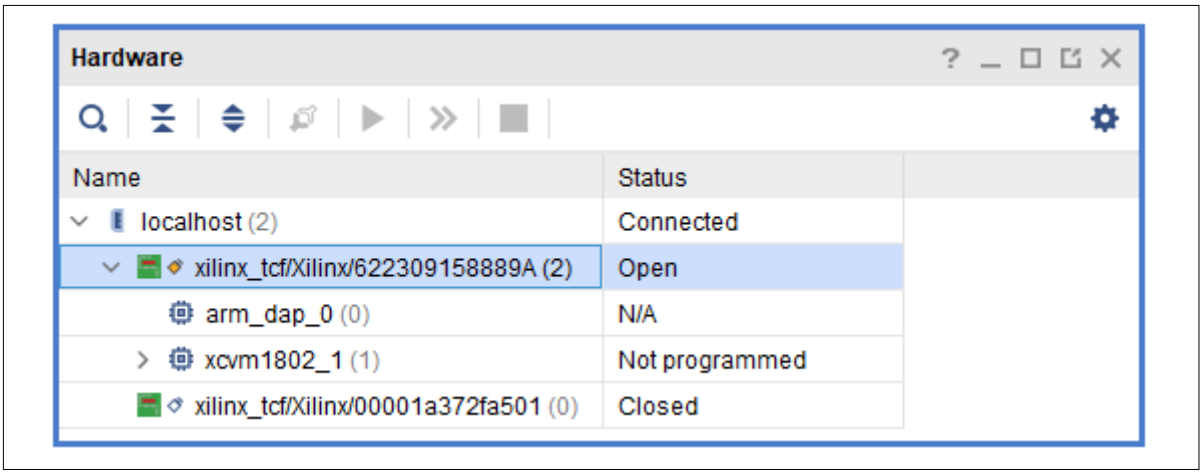


Figure 3 JTAG connection to the Versal device in the Vivado Hardware window

5. On the host machine terminal, run the following command to configure the AMD tools. This command enables Vivado and Vitis to use the generated `u-boot.bin` generated in OSL flow for compiling U-Boot to generate u-boot.bin as the device driver for programming the Infineon NOR flash

```
#Set environment variables on your host machine
export XIL_UBOOT_IMAGE_FILE=" <u-boot_local_repo_full_path>/u-boot.bin"
```

4.2 Programming configuration memory in Vivado

1. In the **Vivado Hardware** window, right-click the hardware target and select **Add Configuration Memory Device** as shown in Figure 4

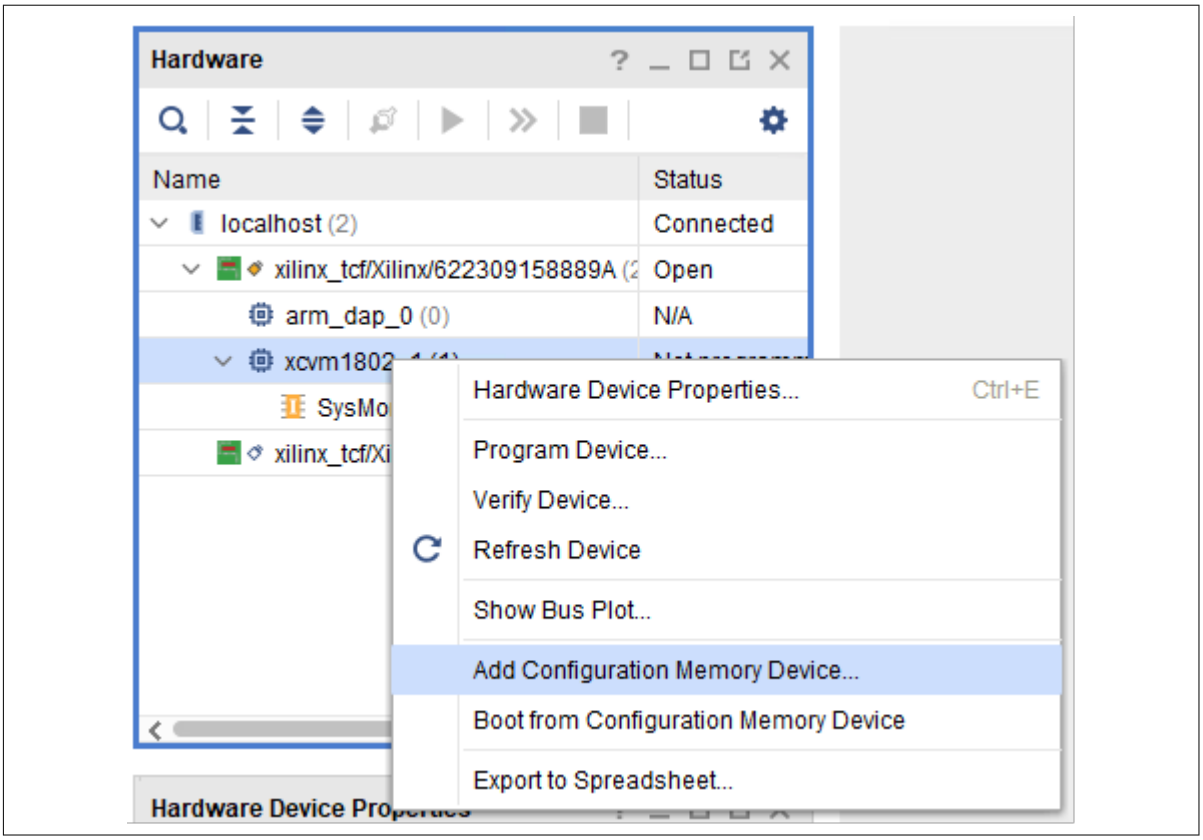


Figure 4 Adding a configuration memory device in the Vivado Hardware window

2. When the **Add Configuration Memory Device** dialog box appears, select the appropriate memory device and click **OK** as shown in Figure 5

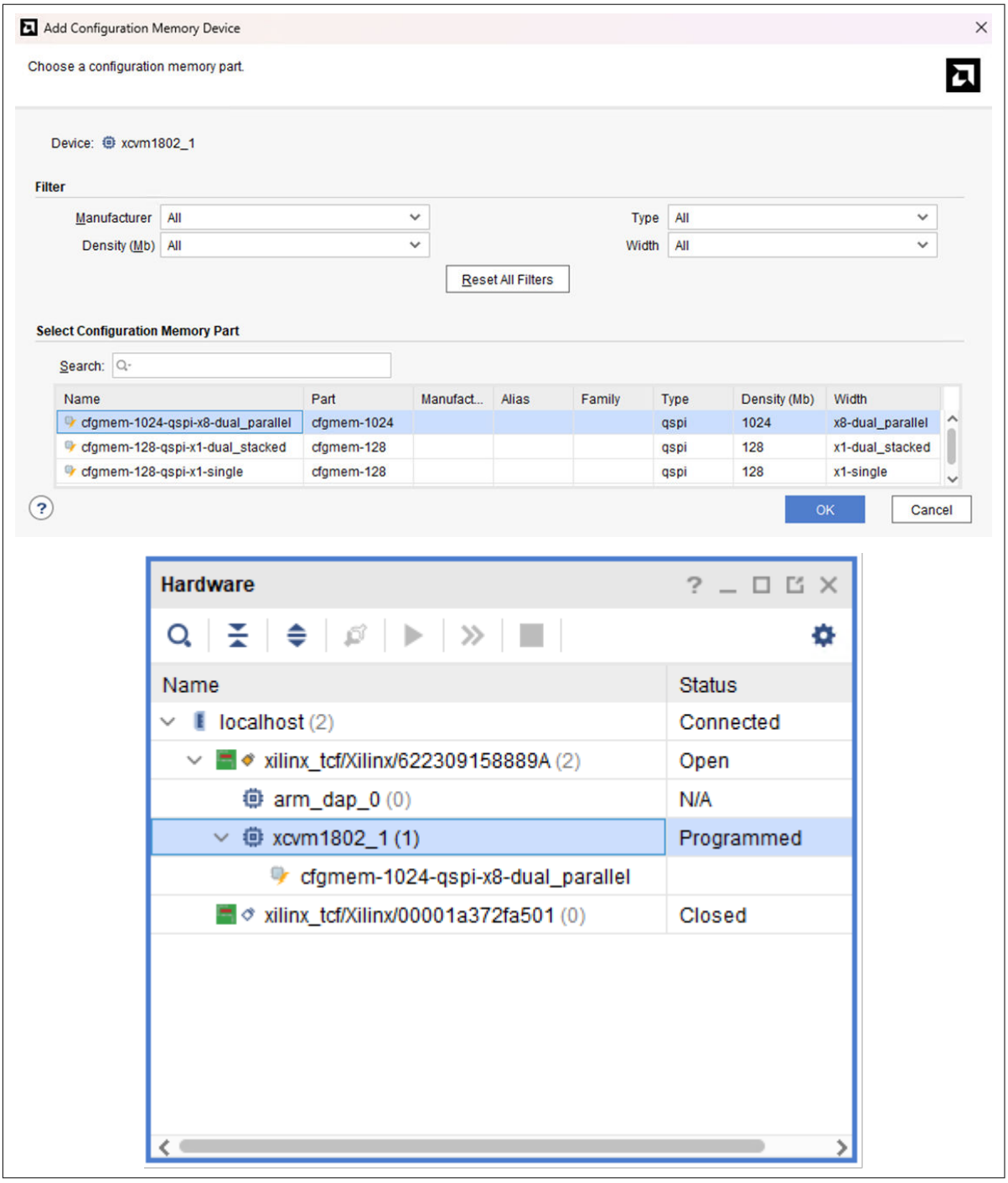


Figure 5 Add Configuration Memory Device dialog box

3. After the configuration memory device is created, the Vivado Device Programmer displays the prompt "Do you want to program the configuration memory device now" as shown in Figure 6

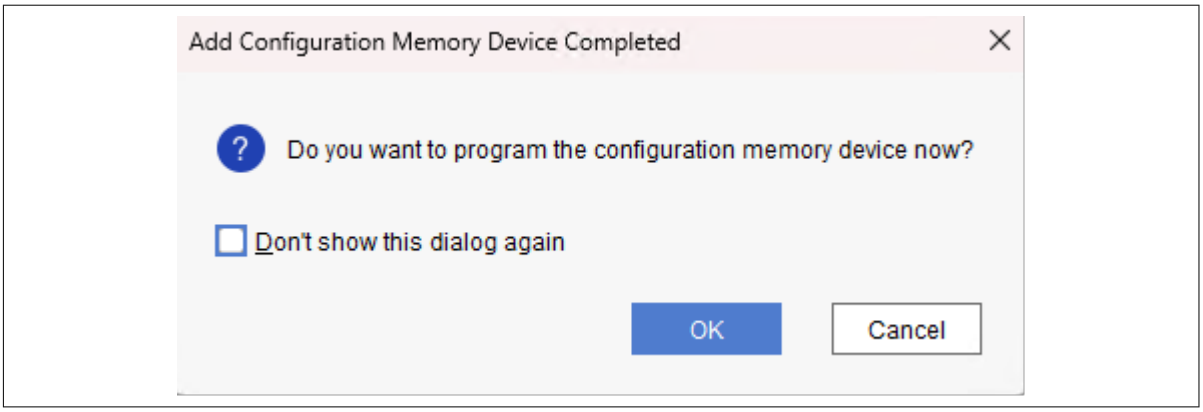


Figure 6 Configuration memory programming prompt

4. In the programming dialog box, configure all required fields appropriately:

- The **Configuration file** field specifies the user binary image that is programmed into the Infineon NOR flash. This file can be the `BOOT.BIN` described in Compiling U-Boot and PLM for BOOT.BIN using PetaLinux, or a custom PDI that includes an operating system image, application, and file system, depending on user requirements

  ***Note**: If a Linux operating system image is used, the Infineon Linux patch must be applied. Contact Infineon for additional support*

- The **Initialization PDI** field specifies the image used to initialize the Versal platform. For other available options, see Vivado Design Suite User Guide: Programming and Debugging (AMD UG908)

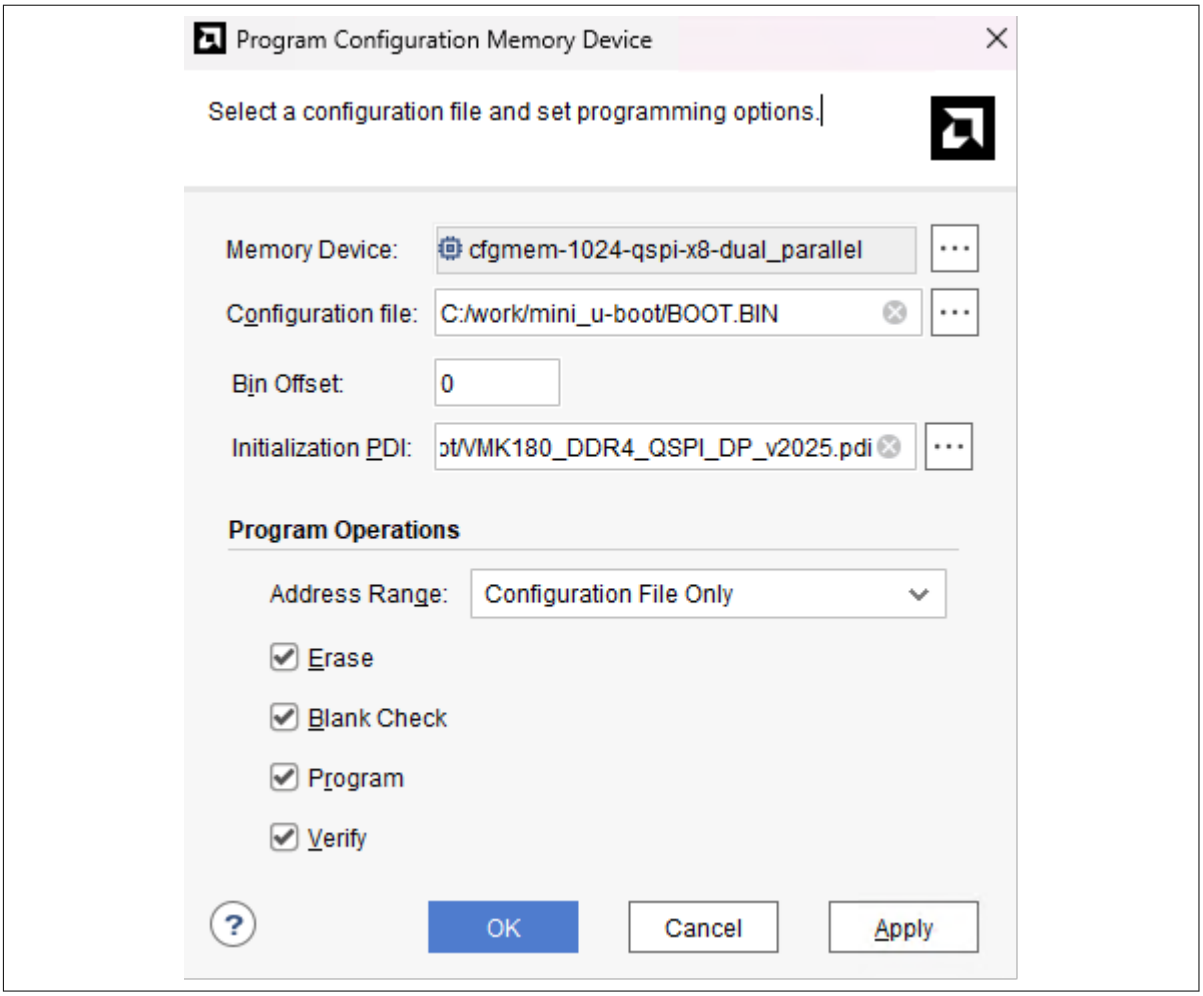


Figure 7 Selecting the configuration file for Infineon NOR flash

5. Click **OK** to start the **Erase**, **Blank Check**, **Program**, and **Verify** operations on the configuration memory device based on the selected options. Vivado notifies you as each operation completes

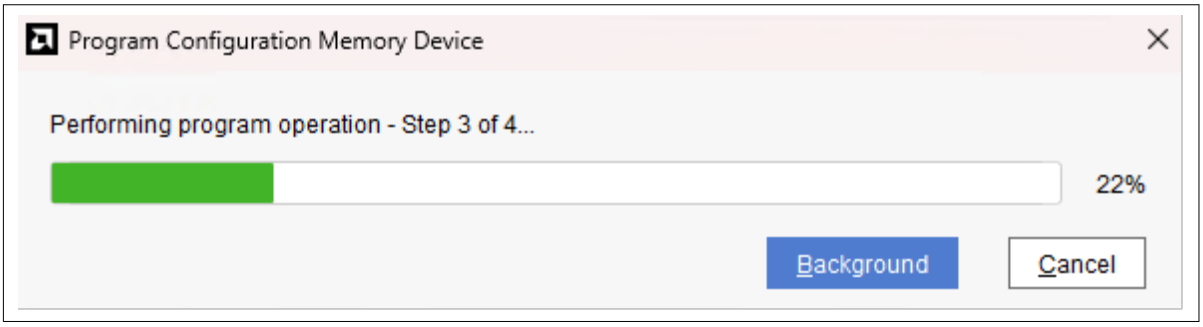


Figure 8 Programming configuration memory operations in progress

6. After all operations complete successfully, the Vivado Device Programmer displays the message “Flash programming completed successfully” as shown in Figure 9

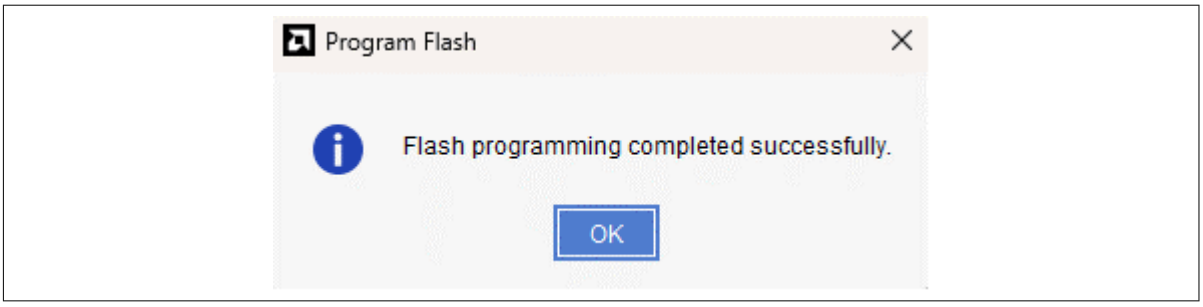


Figure 9 Flash programming completed message

7. Power off the Versal EK board

4.3 Programming configuration memory in Vitis

1. On the host machine terminal, run the following commands:

Commands for Program configuration memory in Vitis

```
cd <vitis installation path>/bin

# For High Reliability RH NOR flash - single x4 memory config
program_flash -f <Name of ConfigurationFile> -pdi <pdi file> -flash_type qspi-x4-single
-no_erase -verify -url TCP:localhost:3121

# For High Reliability RH NOR flash - dual parallel x8 memory config
program_flash -f <Name of ConfigurationFile> -pdi <pdi file> -flash_type qspi-x8-dual_parallel
–flash_density 1024 -no_erase
-verify -url TCP:localhost:3121

# For more information for program_flash
program_flash -help
```

`program_flash` is a utility bundled with the AMD Vitis toolchain. The `program_flash` utility runs on the host machine to load U-Boot onto the Versal device through JTAG and invoke U-Boot commands, such as the `sf` command, to program the target Infineon NOR flash.

The environment variable `XIL_UBOOT_IMAGE_FILE` allows `program_flash` to load a user-specified `u-boot.bin` generated in OSL flow for compiling U-Boot to generate u-boot.bin instead of the default U-Boot binary bundled with AMD Vitis.

2. Specify the configuration file to be programmed. The ConfigurationFile is the user binary image programmed into the Infineon NOR flash. This file can be the `BOOT.BIN` described in Compiling U-Boot and PLM for BOOT.BIN using PetaLinux, or a custom PDI that includes an operating system image, application, and file system, depending on user requirements

   ***Note**: If a Linux operating system image is used, the Infineon Linux patch must be applied. Contact Infineon for additional support*

3. Use the following key arguments with `program_flash`
   - **`-f <ConfigurationFile>`**: Specifies the image to be written to the flash memory

- **`-pdi <pdi file>`**: Specifies the image used to initialize the Versal device
- **`-flash_density <1024 | 2048>`**: Specifies the flash density in Mb for a dual parallel memory configuration

4. After `program_flash` execution completes, the terminal displays a message indicating successful programming of the **ConfigurationFile** into the Infineon NOR flash

Message for JTAG programming

```
program_flash -f BOOT.BIN -pdi VMK180_DDR4_QSPI_DP_v2025_2.pdi
-flash_type qspi-x8-dual_parallel -flash_density 1024 -no_erase -offset 0x0 -verify -url
TCP:localhost:312

****** Program Flash
****** Program Flash v2025.2 (64-bit)
  **** software Build 6140146 on Wed May 21 17:48:38 MDT 2025
  **** IP Build 6138677 on Thu May 22 03:10:11 MDT 2025
  **** SharedData Build 6139179 on Tue May 20 17:58:58 MDT 2025
  **** Start of session at: Mon Mar 23 17:40:57 2026
    ** Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.
    ** Copyright 2022-2025 Advanced Micro Devices, Inc. All Rights Reserved.


WARNING: Failed to connect to hw_server at TCP:localhost:3121
Attempting to launch hw_server at TCP:localhost:3121

Connected to hw_server @ TCP:localhost:3121

Target not specified. Selecting target_id 2 (arm_dap) by default.

Retrieving Flash info...

Initialization done
Overriding default mini u-boot image with file - C:\work\mini_u-boot\u-boot.bin
INFO: [Xicom 50-237] Overriding default mini u-boot image with file - C:\work\mini_u-boot\u-
boot.bin


U-Boot 2025.01-ga09d2660433d-dirty (Mar 23 2026 - 16:04:34 +0900)

Model: Xilinx Versal MINI QSPI PARALLEL
DRAM:  WARNING: Initializing TCM overwrites TCM content
256 KiB
EL Level:    EL3
Multiboot:    0
Core:  6 devices, 6 uclasses, devicetree: separate
Loading Environment from <NULL>... OK
Versal> sf probe 0 0 0

SF: Detected cyrs17b01g with page size 4 KiB, erase size 2 MiB, total 256 MiB
Versal> Sector size = 2048.
Total size = 268435456.
WARNING: [Xicom 50-353] WARNING: Flash Mismatched. If flash programming fails, select the
correct Flash and try again.
Selected Flash: qspi-x8-dual_parallel
Detected Flash: cyrs17b01g
f probe 0 0 0

~
```

```
device 0 offset 0x2b0000, size 0x8000
SF: 32768 bytes @ 0x2b0000 Read: OK
Versal> cmp.b FFFC0000 FFFC8000 8000

Total of 32768 byte(s) were the same
Versal> sf read FFFC0000 2B8000 8000

device 0 offset 0x2b8000, size 0x8000
SF: 32768 bytes @ 0x2b8000 Read: OK
Versal> cmp.b FFFC0000 FFFC8000 8000

Total of 32768 byte(s) were the same
Versal> sf read FFFC0000 2C0000 8000

device 0 offset 0x2c0000, size 0x8000
SF: 32768 bytes @ 0x2c0000 Read: OK
Versal> cmp.b FFFC0000 FFFC8000 8000

Total of 32768 byte(s) were the same
Versal> 100%
sf read FFFC0000 2C8000 69D0

device 0 offset 0x2c8000, size 0x69d0
SF: 27088 bytes @ 0x2c8000 Read: OK
Versal> cmp.b FFFC0000 FFFC8000 69D0

Total of 27088 byte(s) were the same
Versal> INFO: [Xicom 50-44] Elapsed time = 32 sec.
Verify Operation successful.

Flash Operation Successful
```

5. Power off the Versal EK board

5 Program configuration memory with SD boot

5.1 Setting up the SD card on the Versal EK board

1. Format the SD card and copy the following files to the root directory:
   - `BOOT.BIN`
   - `ProgConfigMem.scr`
   - `<ConfigurationFile>`

   The generated `BOOT.BIN` file as described in Compiling U-Boot and PLM for BOOT.BIN using PetaLinux is the programmable device image that enables the Versal platform to boot and load U-Boot for programming the target Infineon NOR flash.

   The `ProgConfigMem.scr` file is a script provided by Infineon to erase the NOR flash and program the `<ConfigurationFile>` image. The script programs the flash by invoking U-Boot commands, such as the `sf` command, and includes verification of the programmed image. This process emulates programming the device using Vivado

   The `<ConfigurationFile>` is the user binary image programmed into the Infineon NOR flash. This file can be the same `BOOT.BIN` described in Compiling U-Boot and PLM for BOOT.BIN using PetaLinux or a custom PDI, for example, one that includes an operating system image, application, and file system, depending on user requirements
2. Insert the SD card into the SD card slot on the top side of the Versal EK board

   ***Note**: The SD card slot on the back side of the board is reserved for the Zynq UltraScale+ system controller*

3. Set the boot mode of the Versal EK board to SD mode

5.2 Booting from the SD card for programming

1. Power on the Versal EK board
2. Verify that the Versal platform boots from `BOOT.BIN` on the SD card

3. On the **Versal U-Boot** terminal, run the following commands to execute the `ProgConfigMem.scr` script for programming the `<ConfigurationFile>` into the target Infineon NOR flash on the Infineon memory module

```
# byte size from offset 0 of configuration memory to be erased. (If setting 0x0, erase is
skipped.)
setenv erase_size 0x2000000

# byte offset of configuration memory to program ConfigurationFile
setenv bin_offset 0x0

# ConfigurationFile to be programmed
setenv config_file  <Name of ConfigurationFile>

# save environment
saveenv
# load script file (0x800000 is code exec. RAM for script file on DIMM DDR, don't change.)
load mmc 0:1 0x800000 ProgConfigMem.scr

# execute script file
source 0x800000
```

4. After the script execution completes, the terminal displays a message indicating successful programming of the <ConfigurationFile> into the Infineon NOR flash

```
Versal> setenv erase_size 0x2000000
Versal> setenv bin_offset 0x0
Versal> setenv config_file BOOT.BIN
Versal> saveenv
Saving Environment to FAT... OK
Versal> load mmc 0:1 0x800000 ProgConfigMem.scr
2086 bytes read in 21 ms (96.7 KiB/s)
Versal> source 0x800000
## Executing script at 00800000
=========== Program ConfigurationFile - Start ===========
-------- erase_size = 0x2000000 --------
-------- bin_offset = 0x0 --------
-------- config_file = BOOT.BIN --------
-------- Detecting flash ------
+sf probe 0 0 0
SF: Detected cyrs17b01g with page size 4 KiB, erase size 2 MiB, total 128 MiB
-------- Erasing flash --------
+sf erase 0x0 0x2000000
SF: 33554432 bytes @ 0x0 Erased: OK
-------- writing BOOT.BIN from SD to DDR -------
+load mmc 0 0x80000 BOOT.BIN
3141616 bytes read in 229 ms (13.1 MiB/s)
-------- writing BOOT.BIN from DDR to address of flash -------
+sf write 0x80000 0x0 2feff0
device 0 offset 0x0, size 0x2feff0
SF: 3141616 bytes @ 0x0 Written: OK
-------- Reading copied data from flash to another DDR address ------
+sf read 0x5000000 0x0 2feff0
device 0 offset 0x0, size 0x2feff0
SF: 3141616 bytes @ 0x0 Read: OK
-------- Comparing data at two DDR addresses ---------
+cmp.b 0x80000 0x5000000 2feff0
Total of 3141616 byte(s) were the same
######## PASS ########
=========== Program ConfigurationFile - End ===========
--------------------------------
##### RESULT SUMMARY ##############################################
TOTAL: 1
PASS: 1
FAIL: 0
####################################################################
```

5. Power off the Versal EK board

6 Boot from Configuration Memory

1. Set the boot mode of the Versal EK board to QSPI mode
2. Power on the Versal EK board
3. Verify that the Versal platform boots from the programmed `<ConfigurationFile>` stored in the Infineon NOR flash
4. If `BOOT.BIN` is used as the `<ConfigurationFile>`, as described earlier in AMD software tools, U-Boot starts successfully and displays the corresponding boot message on the terminal

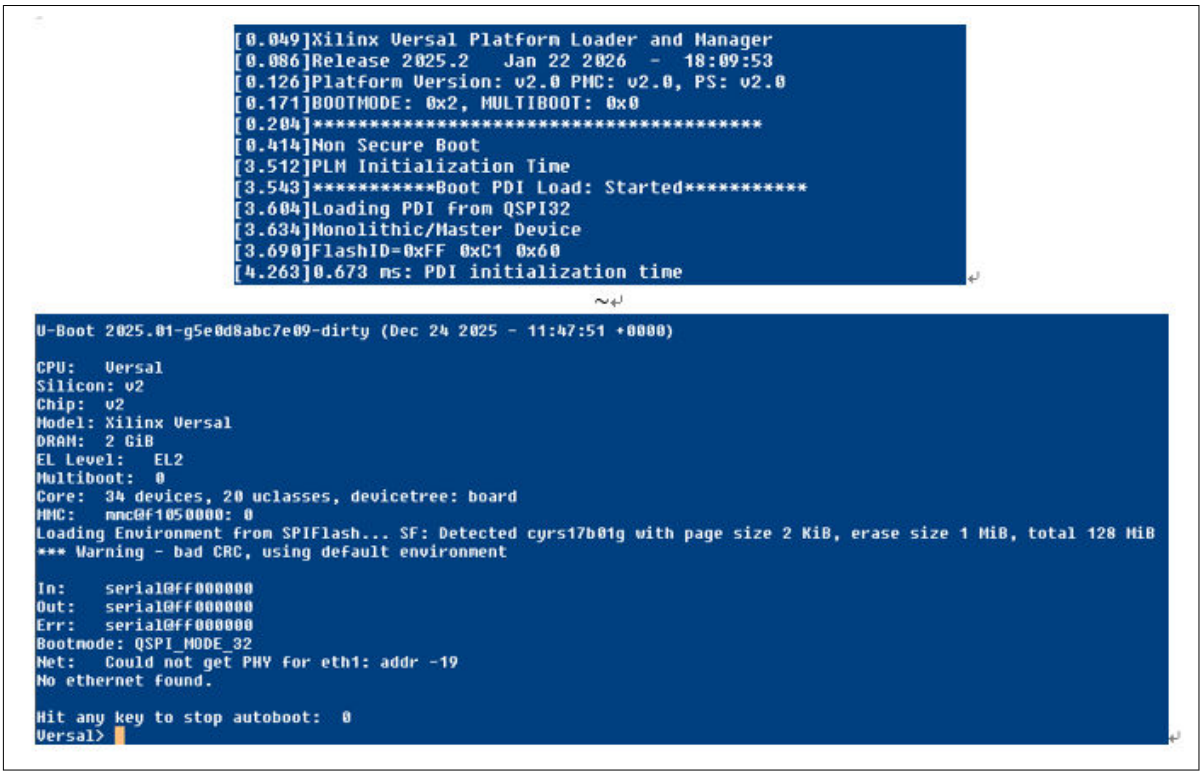


Figure 10 U-Boot message in the terminal

References

[1] Infineon Technologies AG: *Infineon Rad-Hard NOR flash*; Available online
[2] Advanced Micro Devices, Inc./AMD: *Versal Adaptive SoC Technical Reference Manual (AM011)*; Available online
[3] Advanced Micro Devices, Inc./AMD: *Vivado Design Suite User Guide: Programming and Debugging (UG908)*; Available online
[4] Advanced Micro Devices, Inc./AMD: *PetaLinux Tools Documentation: Reference Guide (UG1144)*; Available online
[5] Advanced Micro Devices, Inc./AMD: *VMK180 Evaluation Board User Guide (UG1411)*; Available online
[6] Advanced Micro Devices, Inc./AMD: *VCK190 Evaluation Board User Guide (UG1366)*; Available online

Revision history

Document version	Date of release	Description of changes
**	2026-06-10	Initial release
*A	2026-09-21	Updated Introduction Updated Table 1 Updated Hardware

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2026-09-21

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2026 Infineon Technologies AG

All Rights Reserved.

Do you have a question about any aspect of this document?

Email: erratum@infineon.com

Document reference

IFX-cux1764912347343

Important notice

This evaluation board, demonstration board, reference board or kit and documentation (“Evaluation Board”) is sold or provided and delivered by Infineon Technologies AG and its affiliates (“Infineon”). Evaluation Boards are part of Infineon’s non-serial products.

This Evaluation Board is delivered to customer "AS IS", without any warranty (whether express or implied) or liability of Infineon of any kind, including but without limitation any warranty of merchantability, fitness for a particular purpose or non-infringement. These limitations and exclusions shall not apply in case of Infineon's willful misconduct or in any other case where liability is mandatory at law.

This Evaluation Board shall only be used by customer for test purposes and not for series production. However, the Evaluation Board shall not to be used for reliability testing. The Evaluation Board is exclusively intended for technically qualified and skilled customer representatives. The Evaluation Board shall not be used in any way where a failure of the Evaluation Board, or any consequences of the use thereof, can reasonably be expected to result in personal injury.

Evaluation Boards may not comply with CE or similar standards (including without limitation the EMC Directive 2004/EC/108 and the EMC Act) and may not fulfill other requirements of the country in which they are operated by the customer. The customer shall ensure that each Evaluation Board will be handled in a way which is compliant with all relevant requirements and standards in the country in which it is operated. Software and associated documentation and materials included or referenced in the Evaluation Board (“Software”) is owned by Infineon and is protected by and subject to worldwide patent protection, worldwide copyright laws, and international treaty provisions. Therefore, you may use this Software only as provided in the license agreement accompanying the software package from which you obtained this Software. If no license agreement applies, then any use, reproduction, modification, translation, or compilation of this Software is prohibited without the express written permission of Infineon.

Unless otherwise expressly agreed with Infineon, this Software is provided to customer "AS IS", without any warranty (whether express or implied) or liability of Infineon of any kind, including but without limitation any warranty of merchantability, fitness for a particular purpose or non-infringement. These limitations and exclusions shall not apply in case of Infineon's willful misconduct or in any other case where liability is mandatory at law. Infineon reserves the right to make changes to the Software without notice. You are responsible for properly designing, programming, and testing the functionality and safety of your intended application of the Software, as well as complying with any legal requirements related to its use. Infineon does not guarantee that the Software will be free from intrusion, data theft or loss, or other breaches (“Security Breaches”), and Infineon shall have no liability arising out of any Security Breaches. The Software shall not be used in any way where a failure of the EvaluationBoard, or any consequences of the use thereof, can reasonably be expected to result in personal injury.

Safety & Operating Instructions:
Customer shall check the Evaluation Board for any physical damage which may have occurred during transport. If customer detects any damages or defects in the Evaluation Board, customer shall not connect the Evaluation Board to a power source. Customer shall contact Infineon for further support. If customer observes unusual operating behavior during the evaluation process, customer shall immediately shut off the power supply to the Evaluation Board and consult Infineon for support.
Customer shall not touch the Evaluation Board during operation and keep a safe distance.
Customer shall not touch the Evaluation Board after disconnecting the power supply, several components may still store electrical voltage and can discharge through physical contact. Several parts, like heat sinks and transformers, may still be very hot. Allow the components to cool before touching or servicing.
The electrical installation must be completed in accordance with the appropriate safety requirements.