

DaisyPlus Petalinux Porting Guide

Revision 1.1
2021. 07. 21

[CRZ Technology](#)

Document History

Revision	Date	Change note
1.0	2021.04.19	Initial Version
1.1	2021.07.21	Updated due to new LPDDR4

Index

1. Overview	4
2. Installation of Petalinux 2019.1	5
3. Creation and build of Petalinux project.....	6
4. SD Boot.....	7
5. QSPI Boot	8
6. U-Boot & Kernel.....	9

1. Overview

This document explains how to build Petalinux for DaisyPlus board which is based on Xilinx ZYNQ UltraScale+ XCZU17EG-FFVC1760-2-E on Ubuntu 16.04.4 LTS PC and how to operate DaisyPlus board.

2. Installation of Petalinux 2019.1

Download Petalinux installer, version 2019.1, through the link below.

```
https://www.xilinx.com/member/forms/download/xef.html?filename=petalinux-v2019.1-final-installer.run
```

Install Petalinux Tool.

```
$ sudo -u <user_name> mkdir -p /opt/pkg/petalinux-v2019.1  
$ ./petalinux-v2019.1-final-installer.run /opt/pkg/petalinux-v2019.1
```

Configure environment variables by running below command to use Petalinux Tool.

```
$ source /opt/pkg/petalinux-v2019.1/settings.sh
```

3. Creation and build of Petalinux project

There are 2 HW versions per LPDDR4(U12). Check Micron marker code on U12.

In case marker code is "80C47 D9TFW", [daisyplus_20191_20210419.bsp](#) must be used.

In case marker code is "IDD77 D9ZZL", [daisyplus_20191_20210721.bsp](#) must be used.

Copy petalinux BSP for DaisyPlus board.

```
daisyplus\_20191\_20210419.bsp / daisyplus\_20191\_20210721.bsp
```

Create new project based on DaisyPlus BSP.

```
$ petalinux-create -t project -s daisyplus_20191_20210419.bsp -n daisyplus  
$ petalinux-create -t project -s daisyplus_20191_20210721.bsp -n daisyplus
```

Move to the project folder.

```
$ cd daisyplus
```

Configure the path of hardware description file(HDF).

```
$ petalinux-config --get-hw-description=../hardware/DaisyPlus_M.2_PCl_e_MIG_201901_20210414/ps.s0f  
$ petalinux-config --get-hw-description=../hardware/DaisyPlus_M.2_PCl_e_MIG_NEW_LPDDR4_201901_20210719/ps.sdk
```

Build bootable images for microSD and QSPI flash.

```
$ cd project-spec/build  
$ ./build.sh
```

Pass "sd" as a parameter if microSD bootable image is needed.

```
$ ./build.sh sd
```

To build QSPI bootable image only, pass "qspi" as a parameter.

```
$ ./build.sh qspi
```

Build artifacts are created under the folders below respectively.

```
$ project-spec/image/SD_Image  
$ project-spec/image/QSPI_Image
```

4. SD Boot

Connect microSD card(8GB) to Ubuntu host through microSD card adaptor.

Run "sudo ./create-sdcard.sh" under "project-spec/image" folder to make bootable microSD card.

```
$ cd project-spec/images  
$ sudo ./create-sdcard.sh
```

Set SW1 to OFF-OFF-OFF-ON(4-3-2-1) for SD boot mode on DaisyPlus board.

Insert microSD card to microSD card slot on DaisyPlus board and power on the board.

Check if the logs are displayed on serial terminal.

5. QSPI Boot

Check if SD boot is working with microSD card created in 4.

Power off the board.

Connect microSD card to Ubuntu host through microSD card adaptor again.

Make microSD card for QSPI Boot by running the command below.

```
$ sudo ./make_auto_QSPI_fuse_SD.sh
```

Set SW1 to OFF-OFF-OFF-ON(4-3-2-1).

Insert microSD card to microSD card slot of DaisyPlus board.

Power on the board and then QSPI flash will be configured.

Power off the board after configuring QSPI flash and remove microSD card from the slot.

Set SW1 to ON-ON-OFF-ON(4-3-2-1).

Power on the board and check if the board boots from QSPI flash.

6. U-Boot & Kernel

With the support of NFS on Ubuntu host, u-boot and kernel image can be replaced on the console easily.

Run the commands below to replace u-boot and kernel in case of SD boot.

```
$. /tmp_uboot_update_SD.sh  
$. /tmp_kernel_update_SD.sh
```

Run the commands below to replace u-boot and kernel in case of QSPI boot.

```
$. /tmp_uboot_update_QSPI.sh  
$. /tmp_kernel_update_QSPI.sh
```