

Build kernel, BOOT.bin and image.ub by Petalinux

Editor: Zhen Bian

Get the .hdf/.xsa file of your development board

If you have the same version board as mine, you can directly use my uploaded file `design_1_wrapper.hdf`.

If not, you can use Vivado to generate this file, there are many detailed guidance online that you can refer to.

Use Petalinux to create a project

Put the .hdf file in `./ax_peta/linux_base/` to make sure Petalinux could get the hardware description of you board.

`source /opt/pkg/petalinux/settings.sh` Run this command to set the environment for Petalinux before using it.

```
petalinux-create --type project --template zynq --name ax_peta
cd ax_peta
petalinux-config --get-hw-description ../linux_base.sdk
```

Run `petalinux-config` to re-configure this project if you need.

`Image Packaging Configuration` → `Root filesystem type` Choose `SD card`, and write the correct path of the rootfs, which is `mmcblk0p2` in this work.

Then choose `Save`, and `Exit` this window, wait for the ending of the generation.

Use Petalinux to configure the kernel

Run `petalinux-config -c kernel` to begin configure the kernel.

Configure the driver of PHY

`Device Drivers` → `Network device support` → `PHY Device support and infrastructure` then press `Y` to choose `Micrel PHYs`

Then choose `Save` and `Exit` to finish this configuration.

Configure the rootfs

Run `petalinux-config -c rootfs`, choose `Save` and `Exit`, nothing need to be changed in this work.

Compile and generate U-boot, kernel, dtb

Run `petalinux-build` to compile and generate U-boot, kernel, dtb. This may take about 30min.

Build BOOT.BIN

Run `petalinux-package --boot --fsbl ./images/linux/zynq_fsbl.elf --fpga ./images/linux/system.bit --u-boot --force` to generate the BOOT.bin file. Make sure you have the needed files in correct path.

At last, the BOOT.bin and image.ub are under the path `/images/linux/`.