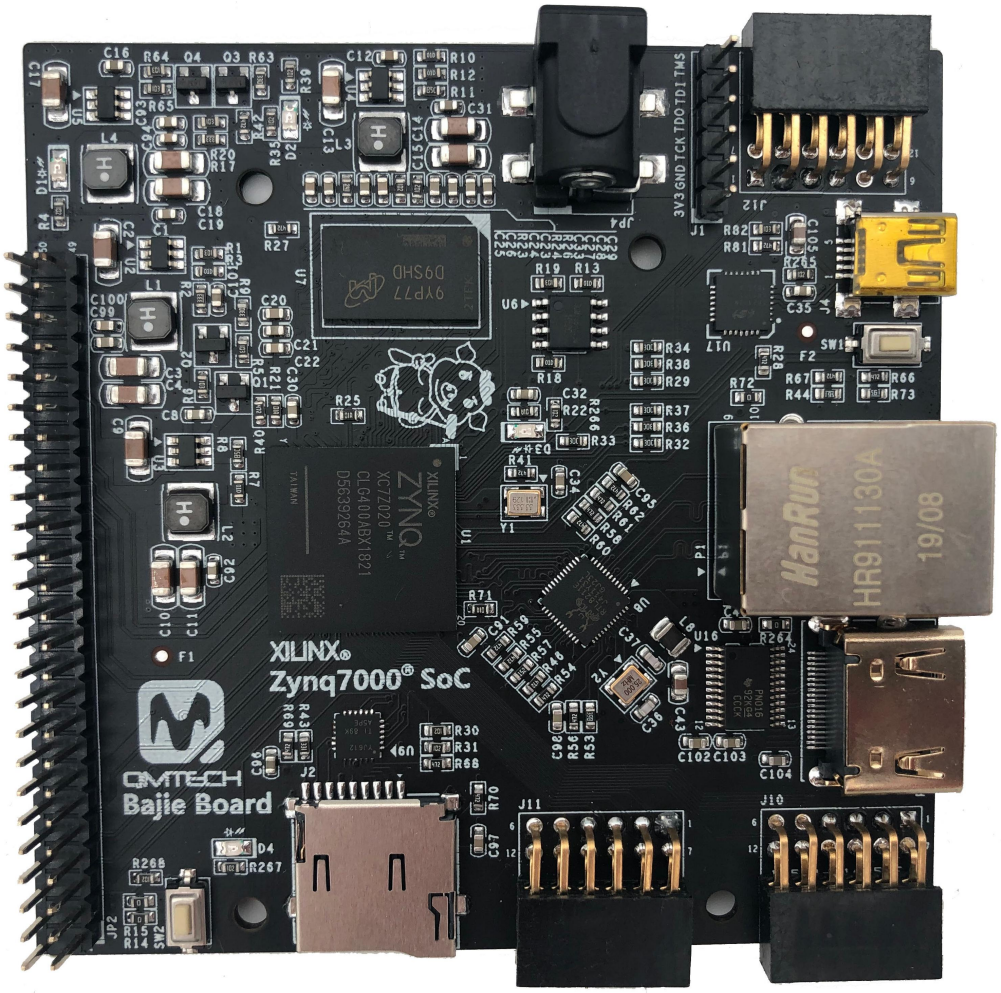
QMTECH ZYNQ7000 BAJIE BOARD

USER MANUAL



Preface

The QMTech® ZYNQ7000 Bajie Board uses Xilinx Zynq®-7000 device which integrates the software programmability of an ARM®-based processor with the hardware programmability of an FPGA, enabling key analytics and hardware acceleration while integrating CPU, DSP, ASSP, and mixed signal functionality on a single device. Consisting of single-core Zynq-7000S and dual-core Zynq-7000 devices, the Zynq-7000 family is the best price to performance-per-watt, fully scalable SoC platform for your unique application requirements.

For more information, updates and useful links, please visit QMTECH Official Website:

http://www.chinaqmtech.com

Table of Contents

1. Introduction

1.1 Document Scope

This user manual introduces the procedure to make the PetaLinux environment running on the QMTECH ZYNQ7000 Bajie Board. The PetaLinux environment mainly covers three parts: u-boot, Linux OS and file system. All of those parts are developed with PetaLinux 2019.2 under Ubuntu 18.04.1 (64bit) environment. The prerequisites before working with the PetaLinux are shown as below:

1. Preferred and verified Ubuntu version is Ubuntu 18.04.1 (64bit).

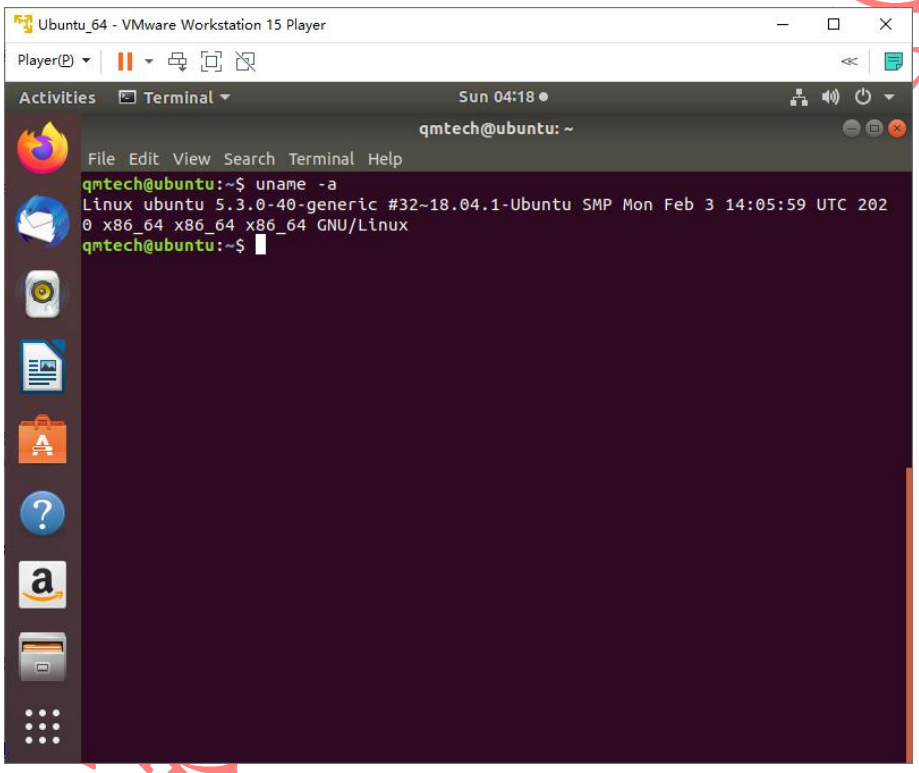


Figure 1-1. Ubuntu Version

2. Users shall install the required packages before install the PetaLinux 2019.2 in Ubuntu. The detailed required packages are mentioned in **UG1144**: ***Table 2: Packages and Linux Workstation Environments.***
   Notification: Users need to have root access to install the required packages mentioned in that table. The PetaLinux tools need to be installed as a non-root user.
3. Users shall have the basic knowledge about the usage of the Linux environment. Know how to use the cross-compile toolchain including arm-gcc-linux-, makefile, etc.

2. Getting Started

This chapter describes the detailed steps to create a customized PetaLinux. Comparing to the existing ZYNQ development board e.g. Xilinx ZC702, there are many differences in the QMTECH ZYNQ7000 Bajie Board. For example, there's only one 16bit width DDR3 memory chip connected to PS ARM core. And HDMI display interface is implemented at the PL side. Hence, the u-boot/ Linux device tree for the Bajie Board needs to be updated here.

2.1 Steps to Customize the PetaLinux without HDMI Display

The first step is to customize the hardware info required by PetaLinux. The hardware info could be retrieved from below Vivado project: $Bajie_Board\zynq_linux\PetaLinux_No_HDMI\Project06_PetaLinux_20200412_V02_No_HDMI.zip. Below image shows the customized ZYNQ system in Vivado 2018.3. Make sure the project could successfully pass the three steps: Synthesis, implementation and Generate Bitstream.

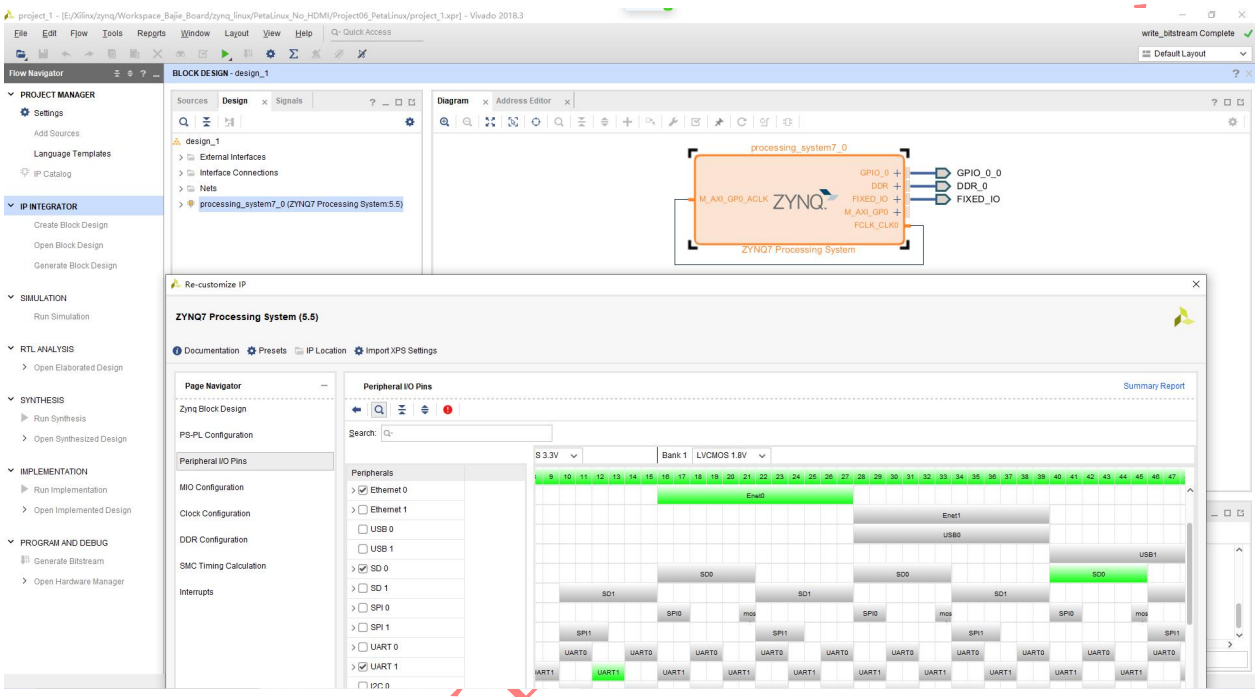


Figure 2-1. Modifications

Then users could retrieve the customized hardware info by clicking the [File] -> [Export] -> [Export Hardware]. And remember check the [Include Bitstream] before click the [OK] button.

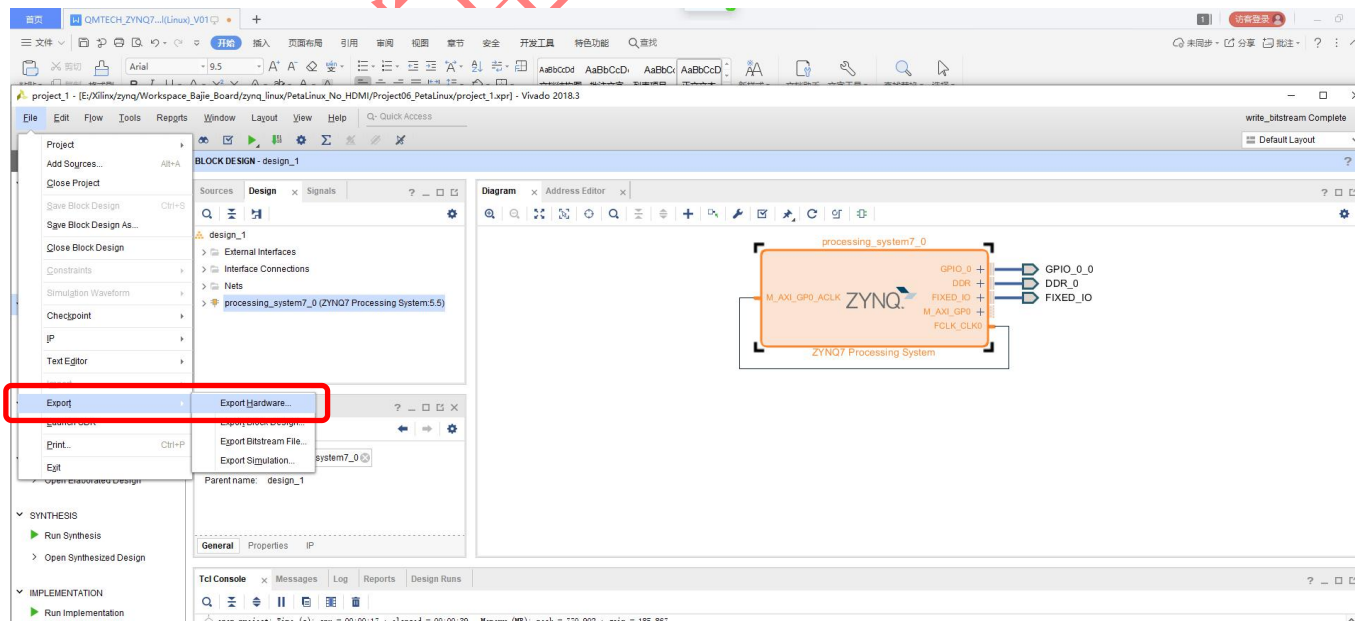


Figure 2-2. Export Hardware Info

The generated hardware info file could be found here:
$Bajie_Board\zynq_linux\PetaLinux_No_HDMI\Project06_PetaLinux\project_1.sdk\design_1_wrapper.hdf

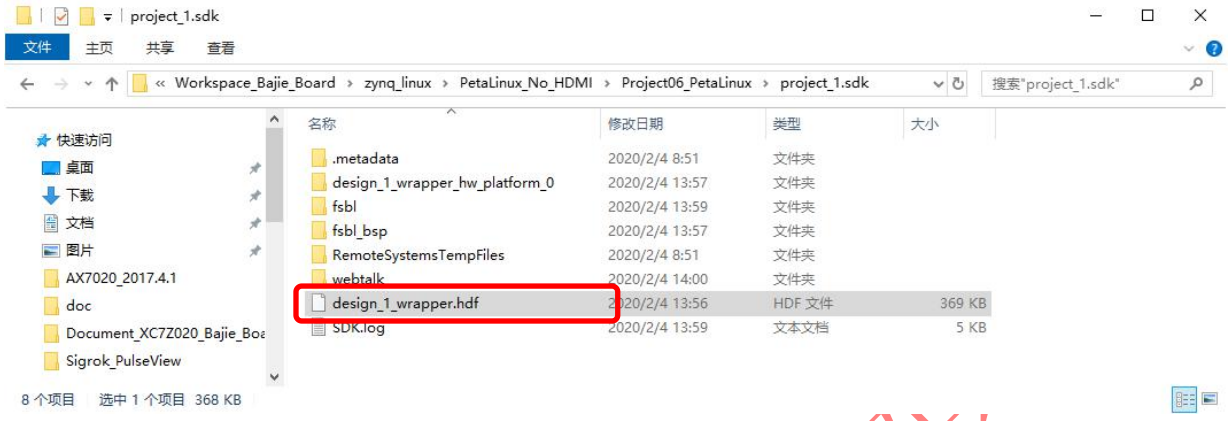


Figure 2-3. Target Hardware Info File

Copy the design_1_wrapper.hdf file into Ubuntu environment. E.g. put the file in folder $workspace/PetaLinux_workspace/linux_base.sdk.

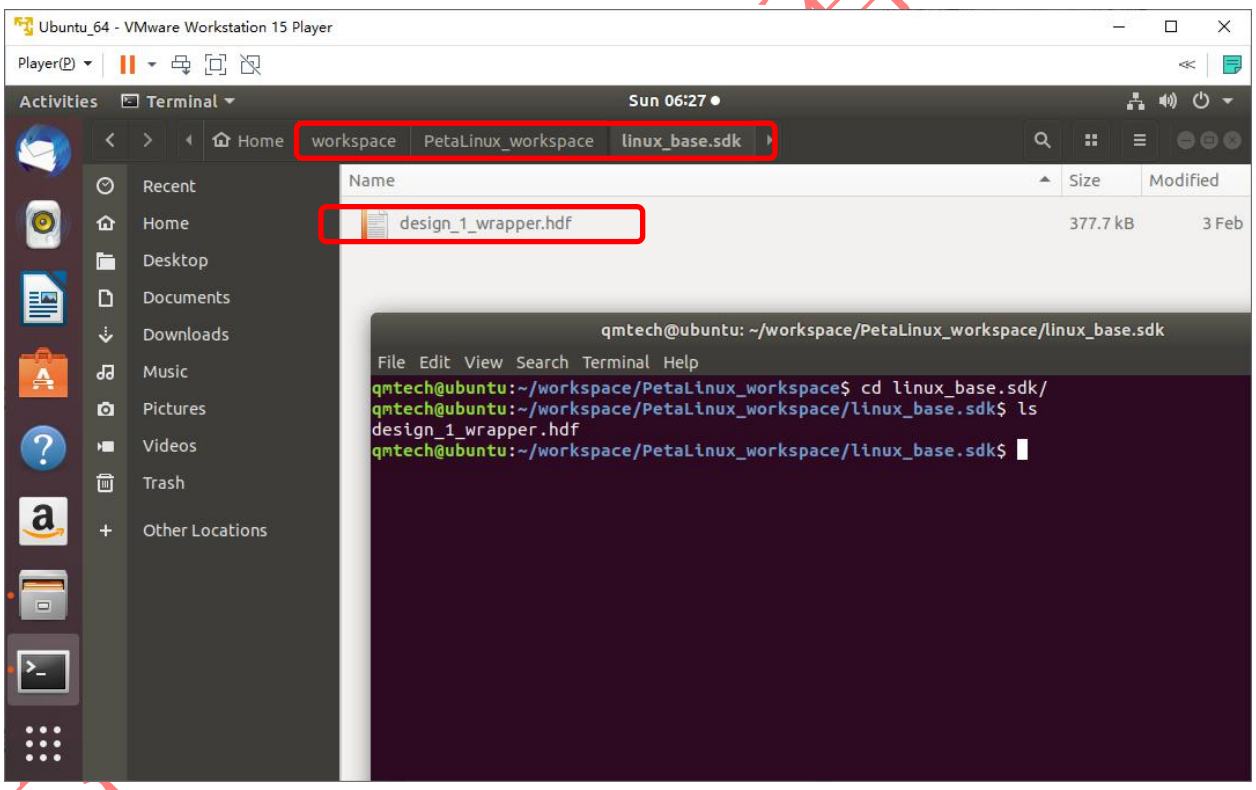


Figure 2-4. Import Hardware Info File

In our example, the PetaLinux 2019.2 package is installed in folder /opt/pkg/petalinux. Users need change the below commands according to the detailed directory that contains the PetaLinux.
Type command in the terminal to source the PetaLinux: **/opt/pkg/petalinux/settings.sh**. Create a new folder named as **qmtech_peta_01** by typing command: **petalinux-create --type project --template zynq --name qmtech_peta_01**. This new folder is in the same directory as the linux_base.sdk and used as the PetaLinux workspace.

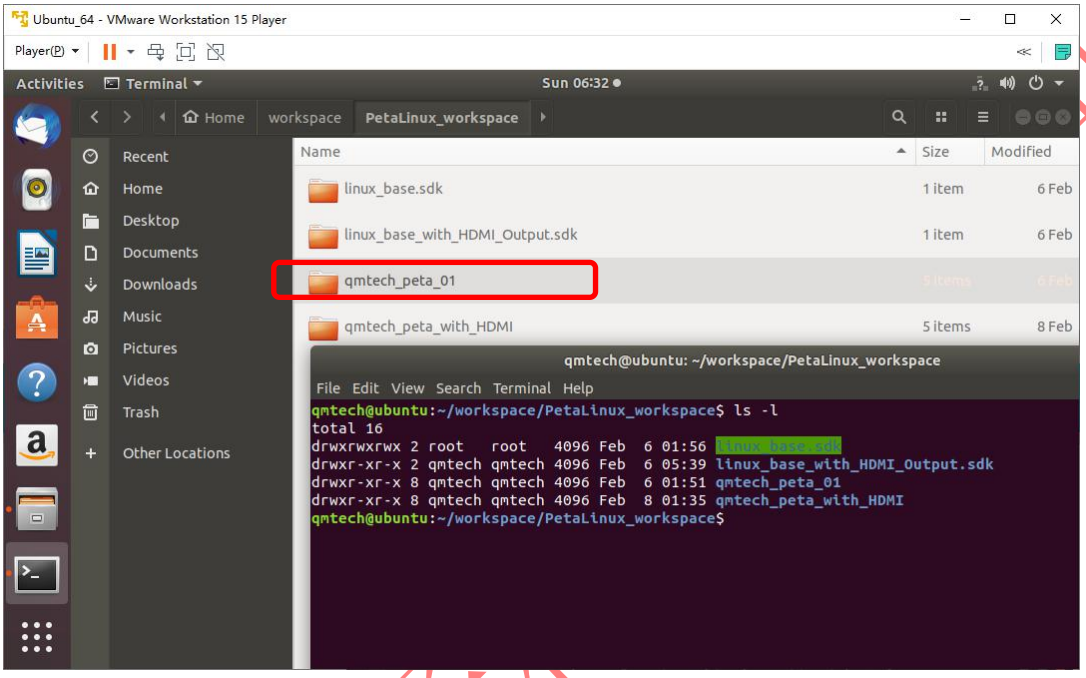


Figure 2-5. Create PetaLinux Working Folder

Then enter into the folder **qmtech_peta_01**.

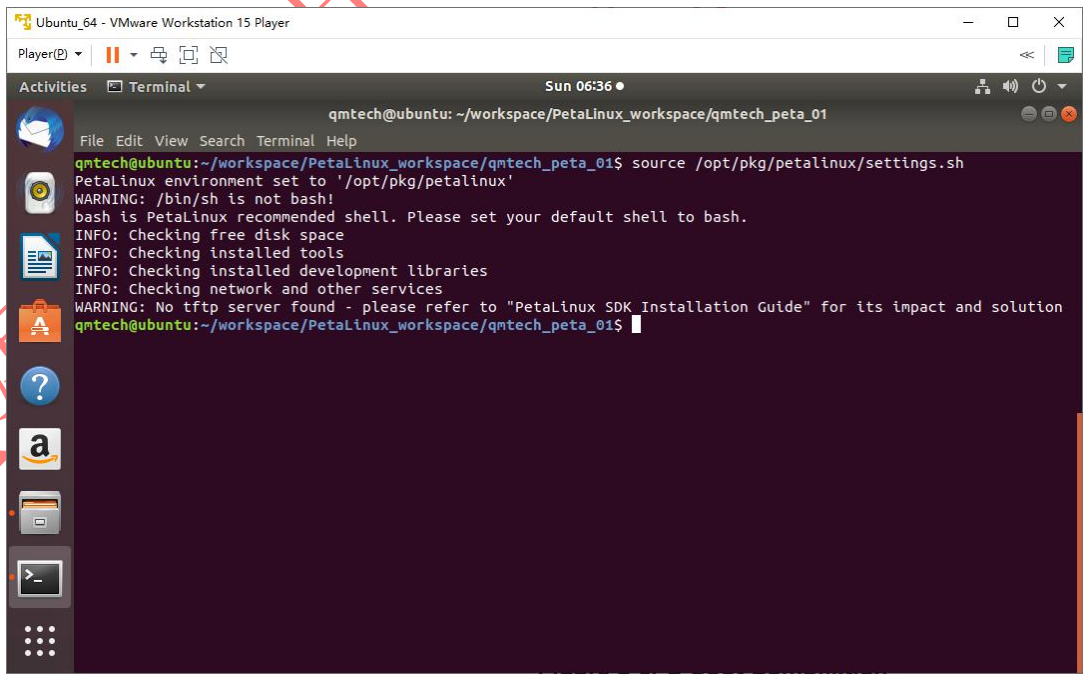


Figure 2-6. Enter Workspace

Import the hardware info into PetaLinux by command: **petalinux-config --get-hw-description ../linux_base.sdk**. Below configuration image will display and nothing needs to be changed here.

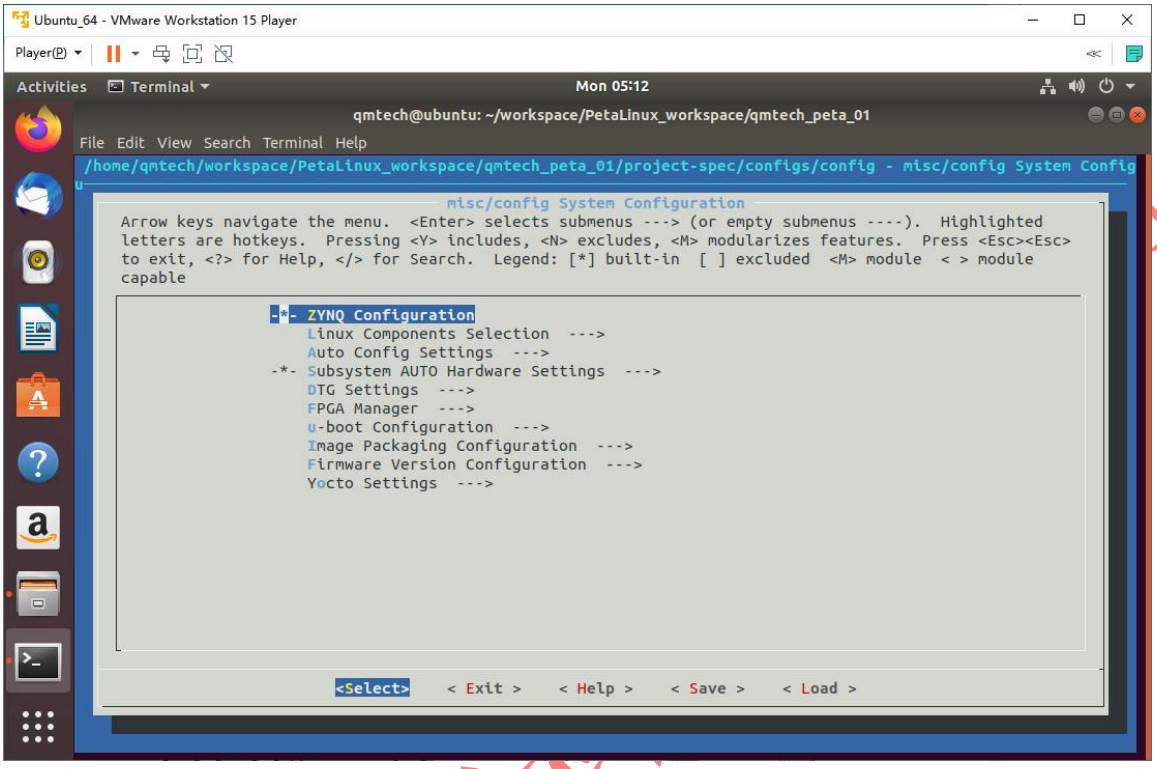


Figure 2-7. ZYNQ Configuration

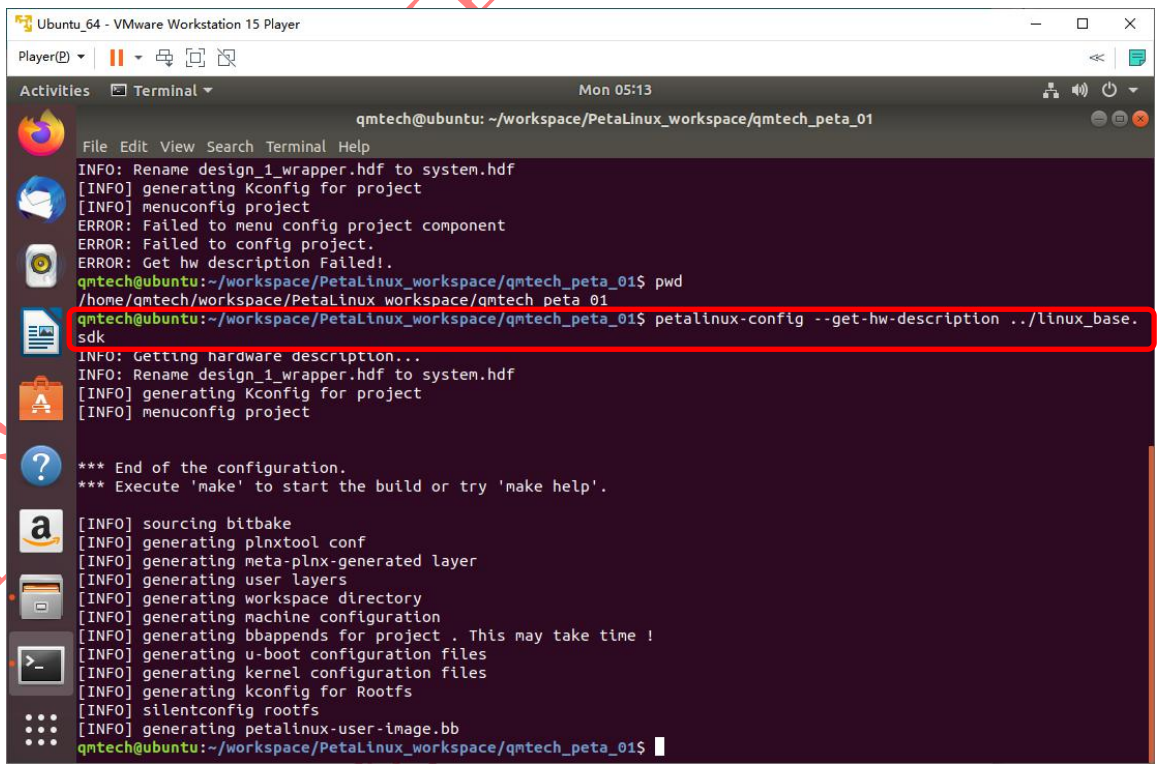


Figure 2-8. Import Hardware Info

Once the hardware info is successfully imported, users may start to configure the PetaLinux kernel by command: **petalinux-config -c kernel**
This step may take a really long time. Be patient here and make sure the VM's network status is perfect.

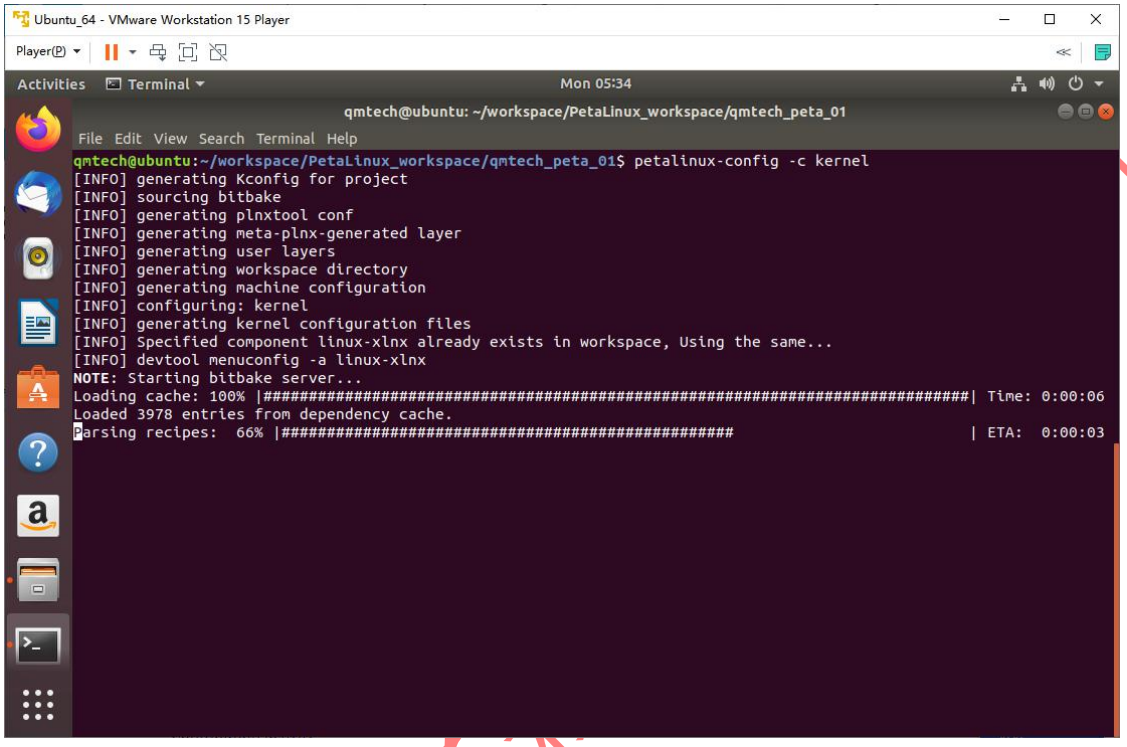


Figure 2-9. Configure PetaLinux Kernel

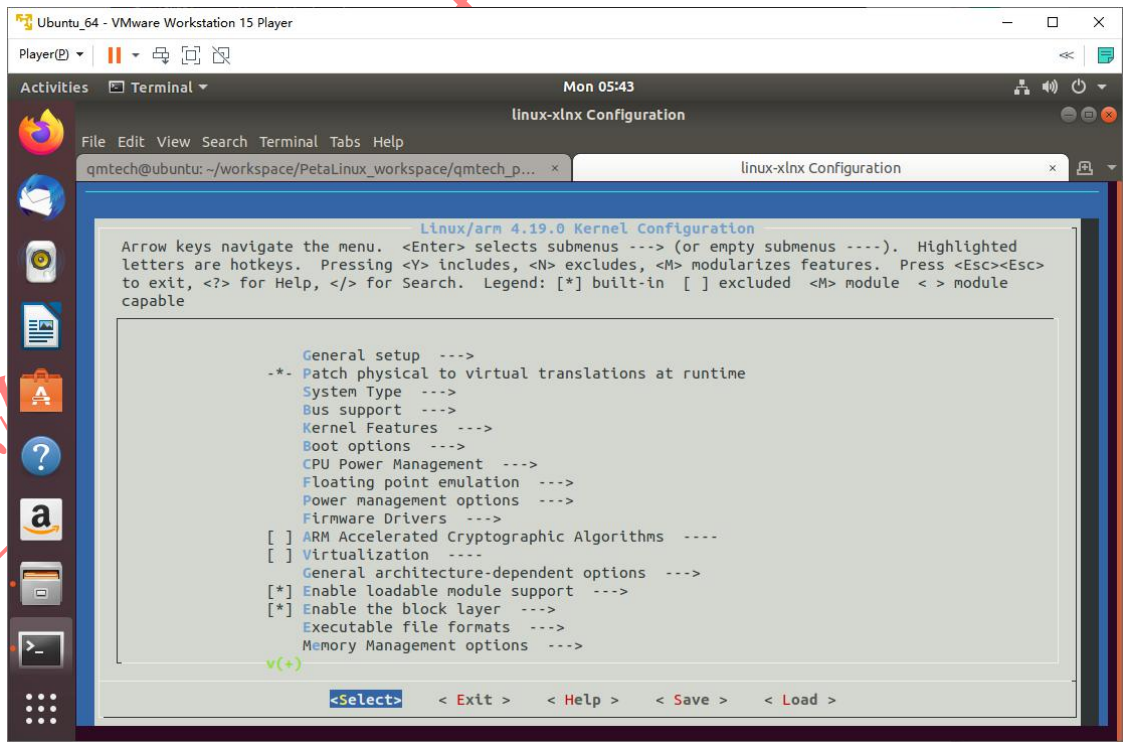


Figure 2-10. No Need to change the Kernel Configurations

Once the PetaLinux Kernel info is successfully configured, users may start to configure the PetaLinux file system by command: **petalinux-config -c rootfs**
Below image will display and nothing needs to be changed here.

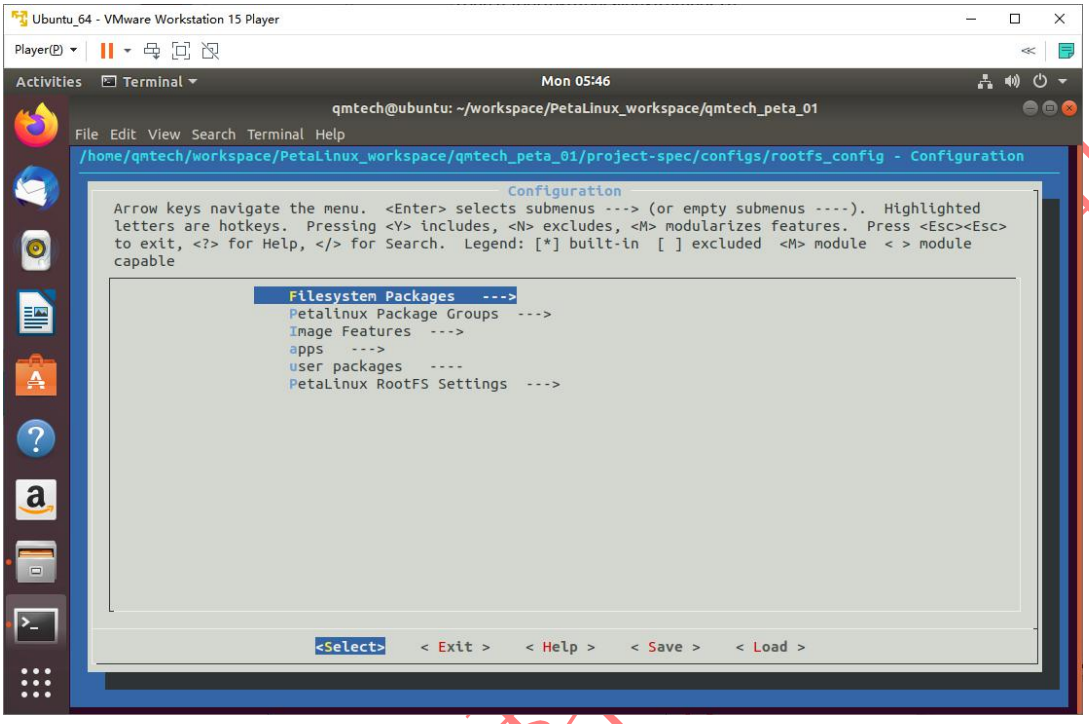


Figure 2-11. No Need to change the rootfs

Start to build the PetaLinux by command: **petalinux-build**

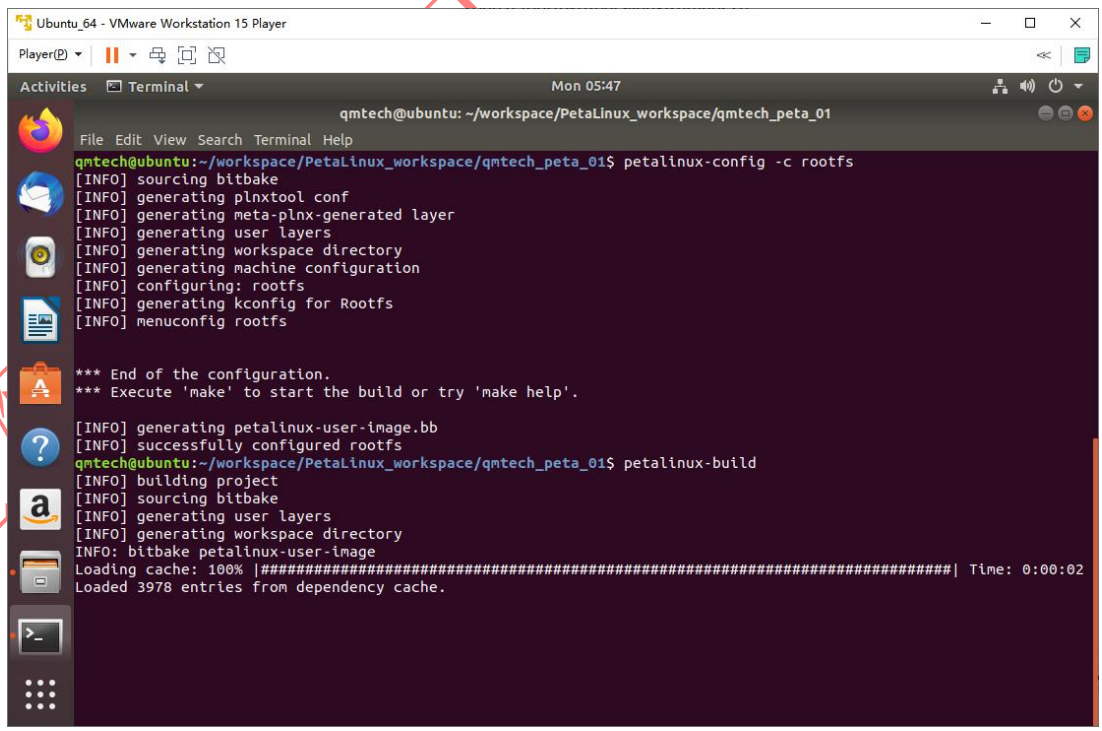
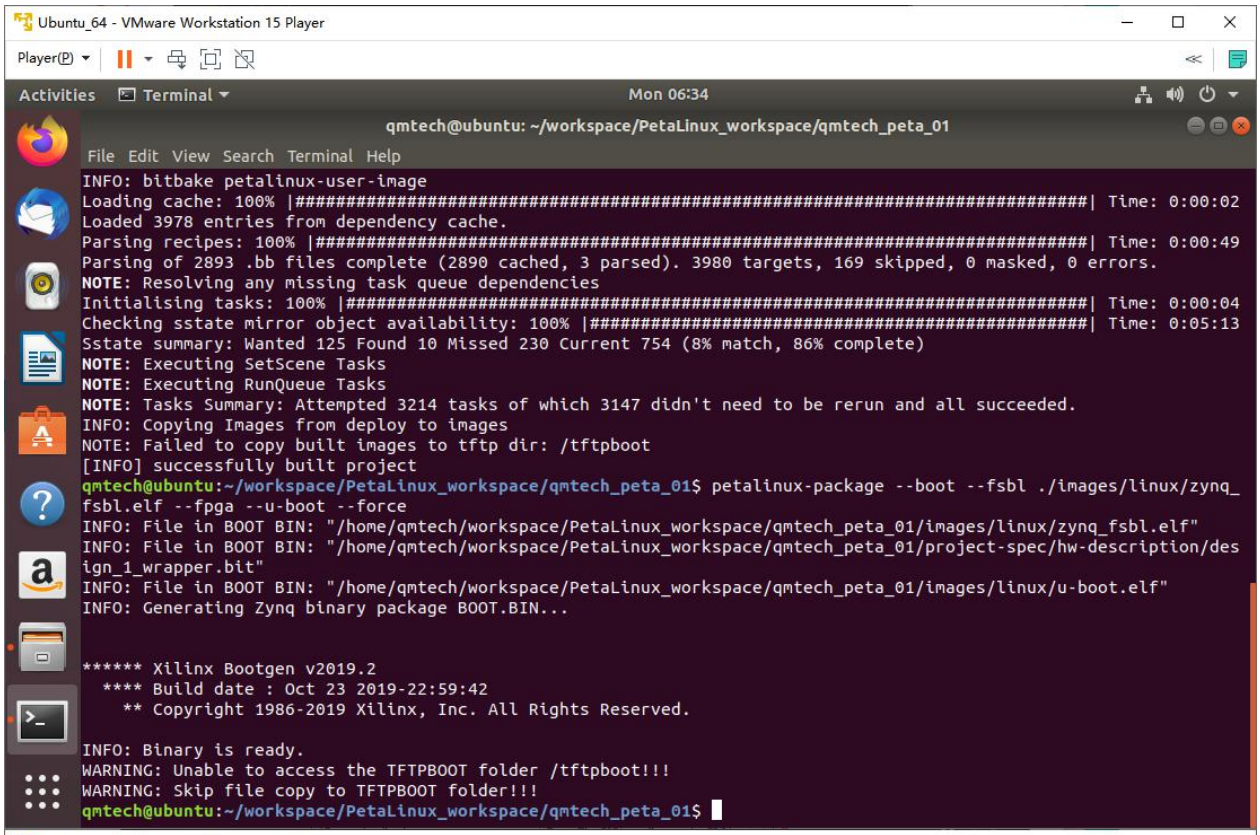
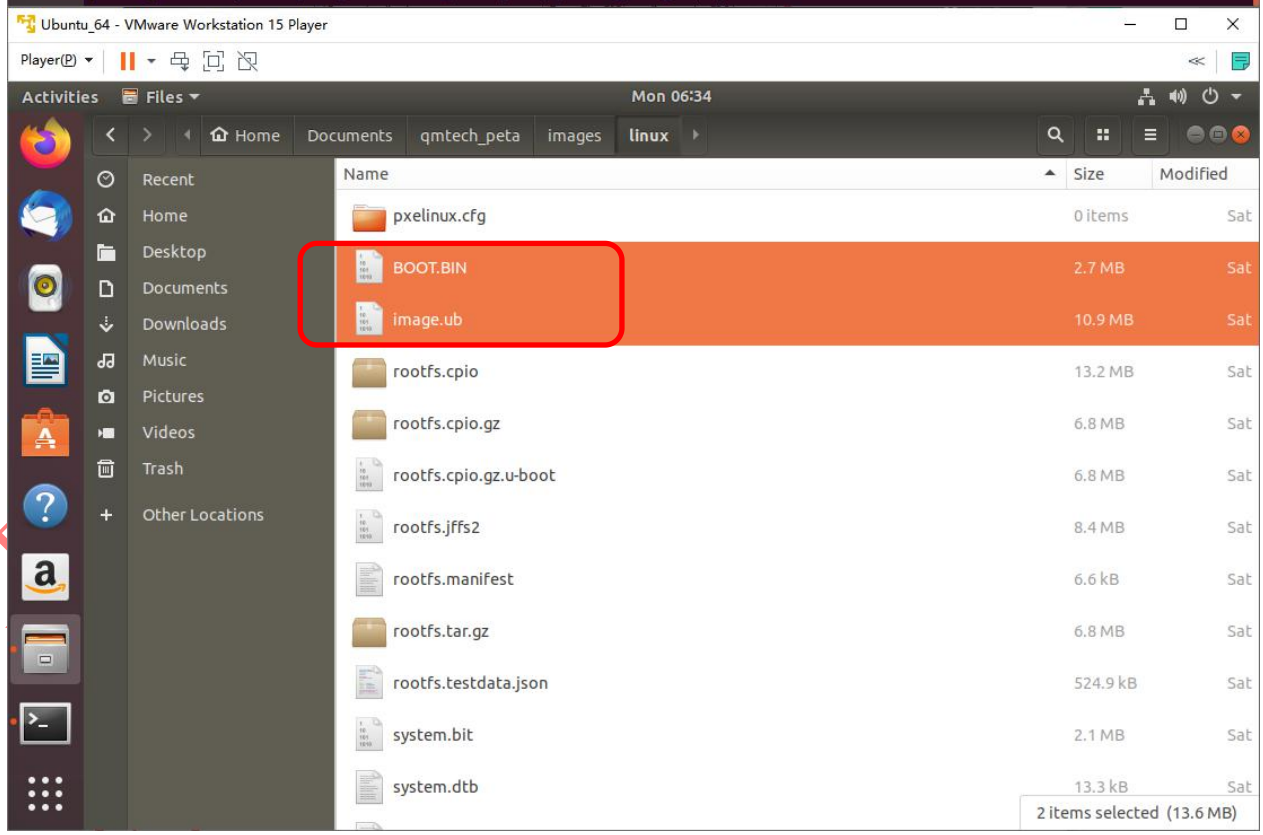


Figure 2-12. Build PetaLinux

After the PetaLinux is successfully built, users could generate the target BOOT.bin and image.ub by command: **petalinux-package --boot --fsbl ./images/linux/zynq_fsbl.elf --fpga --u-boot --force**





Copy the BOOT.bin and image.ub into MicroSD card and then insert the MicroSD card into ZYNQ7000 Bajie Board. Plug the MiniUSB cable into the Bajie Board and then power on the board with 5V DC power source. Below image shows the serial output from the board.

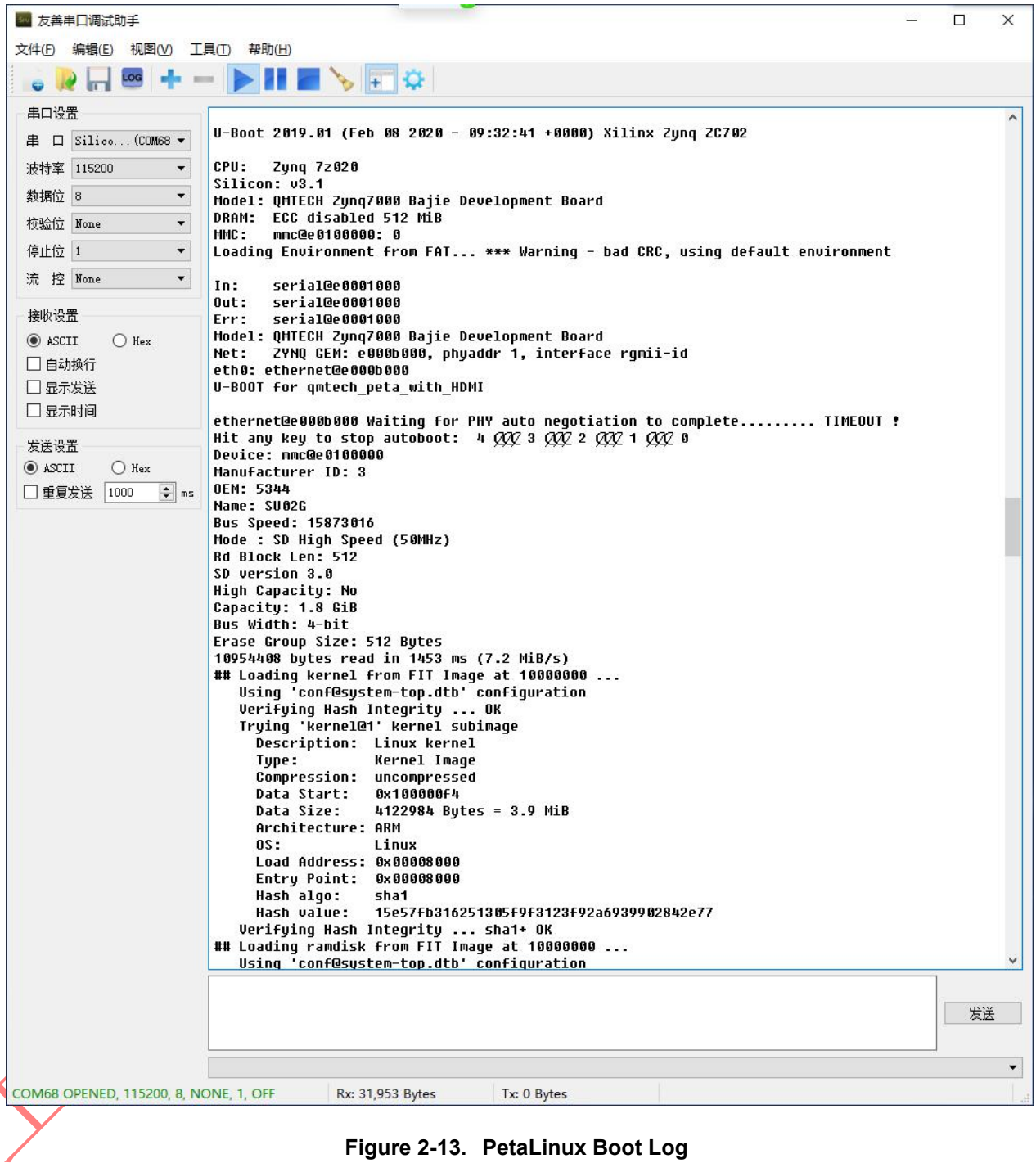


Figure 2-13. PetaLinux Boot Log

2.2 Test the Ethernet Under Linux Environment

Plug the ethernet cable into the Bajie Board and power on it. Below log will be displayed on the terminal tool when the ethernet link is ready:

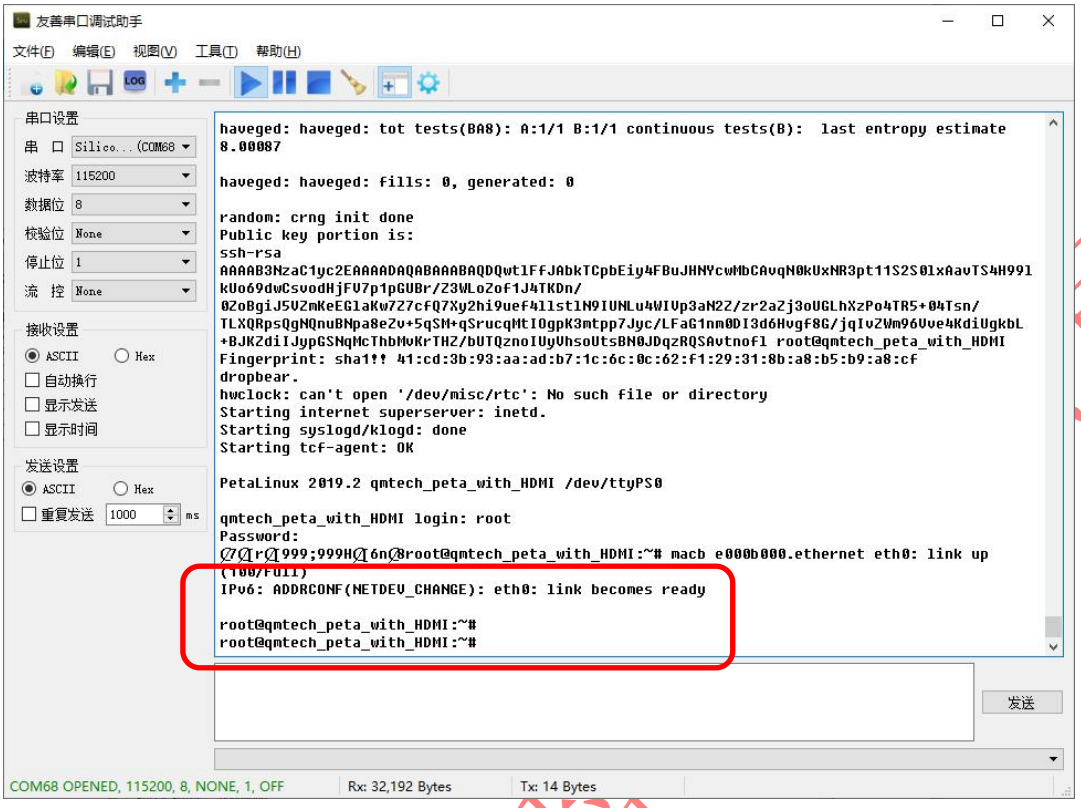


Figure 2-14. Log Info

Users may type below ethernet test related commands to check the status of ethernet interface.

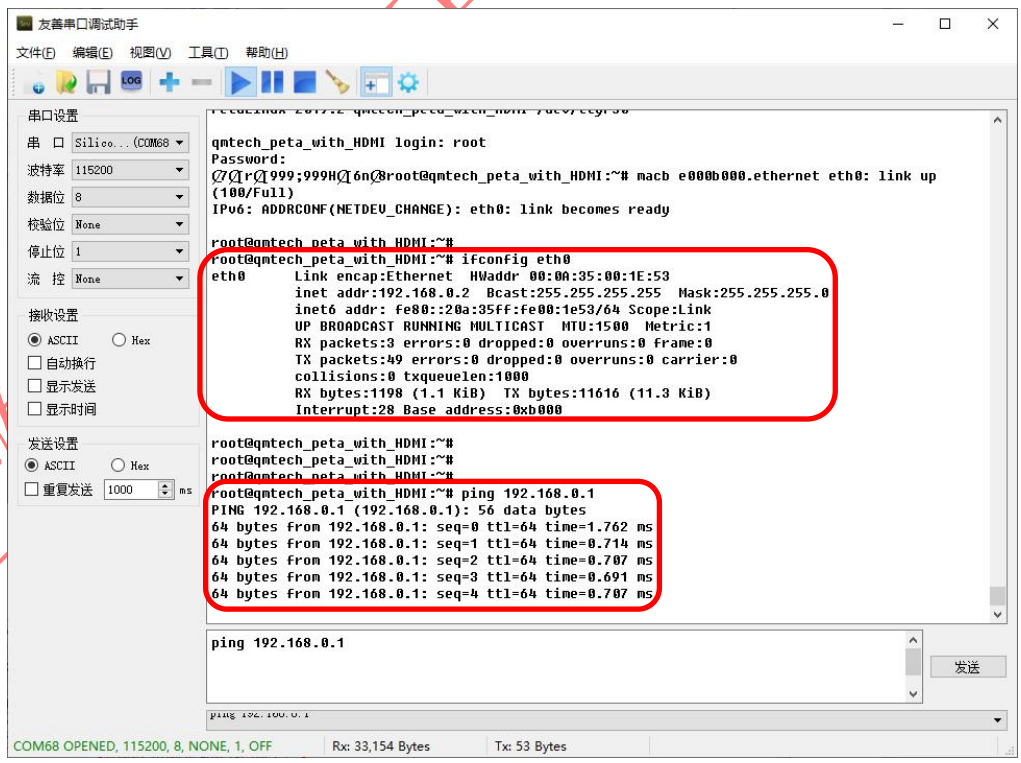


Figure 2-15. Echo Test under Linux

2.3 Steps to Customize the PetaLinux with HDMI Display

This chapter describes the steps to customize a new PetaLinux with HDMI display interface enabled. With this feature, users could develop some GUI based applications. Here, only lists the keys steps that enable the PetaLinux HDMI interface.

The first step is to customize the hardware info required by PetaLinux. The hardware info could be retrieved from below Vivado project: $Bajie_Board\zynq_linux\PetaLinux_With_HDMI\Project06_PetaLinux_With_HDMI_20200412_V02.zip. Below image shows the customized ZYNQ system in Vivado 2018.3. Make sure the project could successfully pass the three steps: Synthesis, implementation and Generate Bitstream.

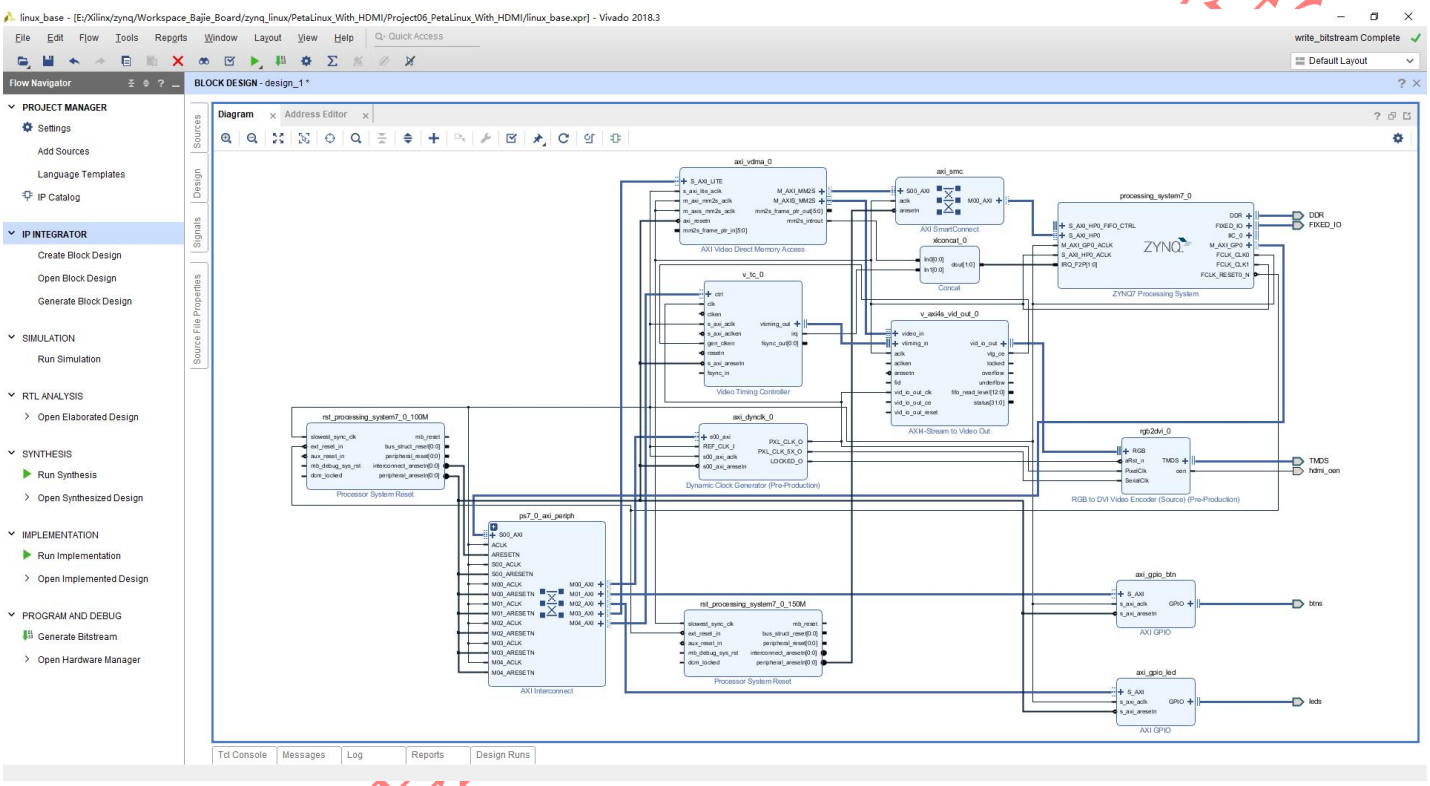
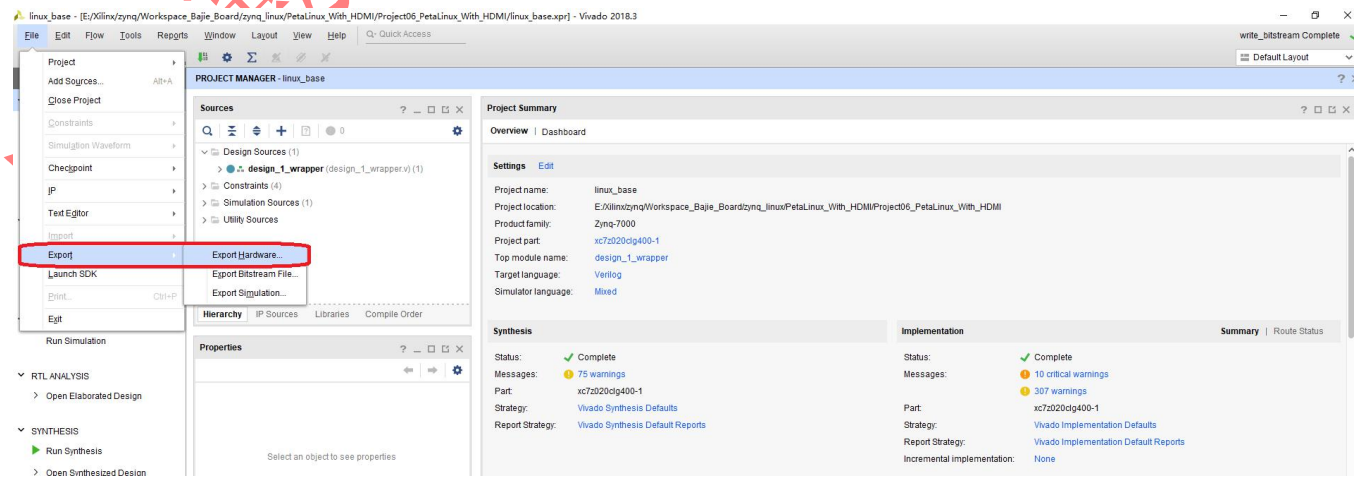


Figure 2-16. Hardware Architecture

Then users could retrieve the customized hardware info by clicking the [File] -> [Export] -> [Export Hardware]. And remember check the [Include Bitstream] before click the [OK] button.



Copy the design_1_wrapper.hdf file into Ubuntu environment. E.g. put the file in folder $workspace/PetaLinux_ workspace/linux_base_with_HDMI_Output.sdk.

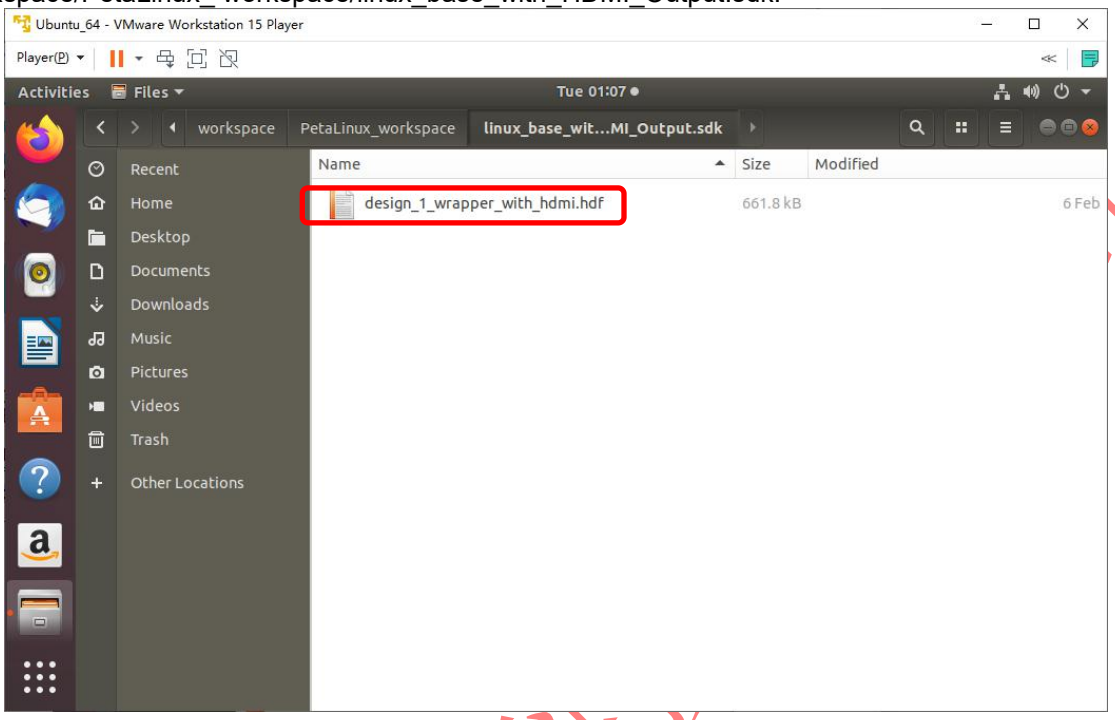


Figure 2-17. Create PetaLinux Working Folder

Type command in the terminal to source the PetaLinux: **/opt/pkg/petalinux/settings.sh**. Create a new folder named as **qmtech_peta_with_HDMI** by typing command: **petalinux-create --type project --template zynq --name qmtech_peta_with_HDMI**. This new folder is in the same directory as the linux_base_with_HDMI_Output.sdk and used as the PetaLinux workspace.

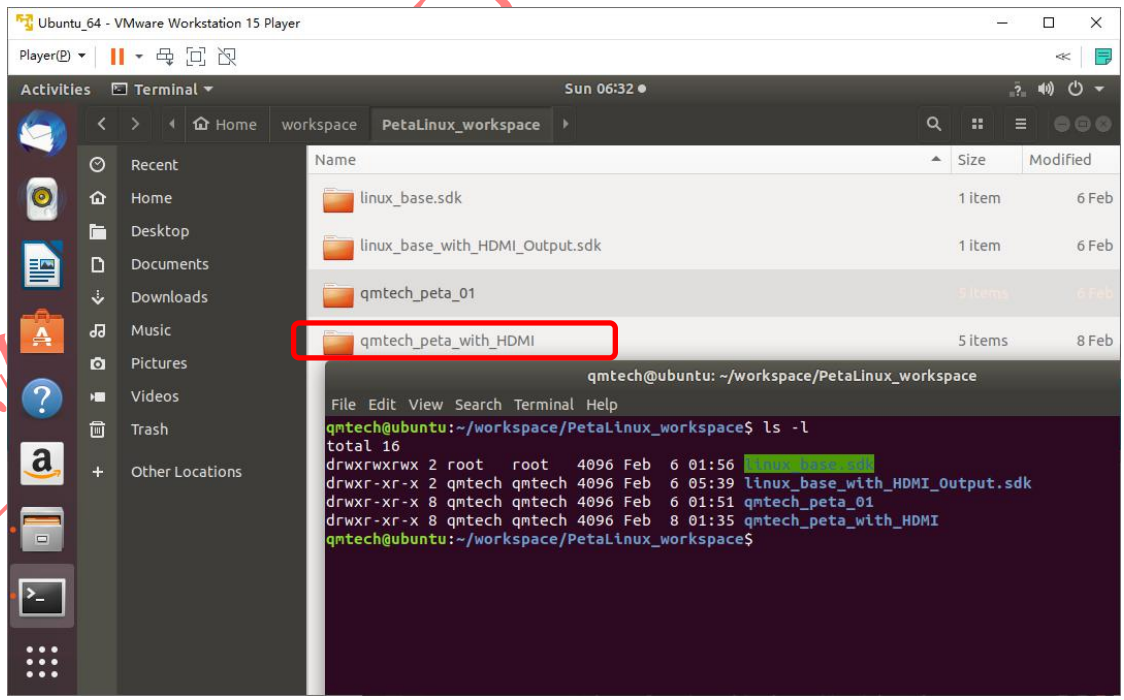


Figure 2-18. Create PetaLinux Working Folder

Import the hardware info into PetaLinux by command: **petalinux-config --get-hw-description ../linux_base_with_HDMI_Output.sdk.** Below configuration image will display and nothing needs to be changed.

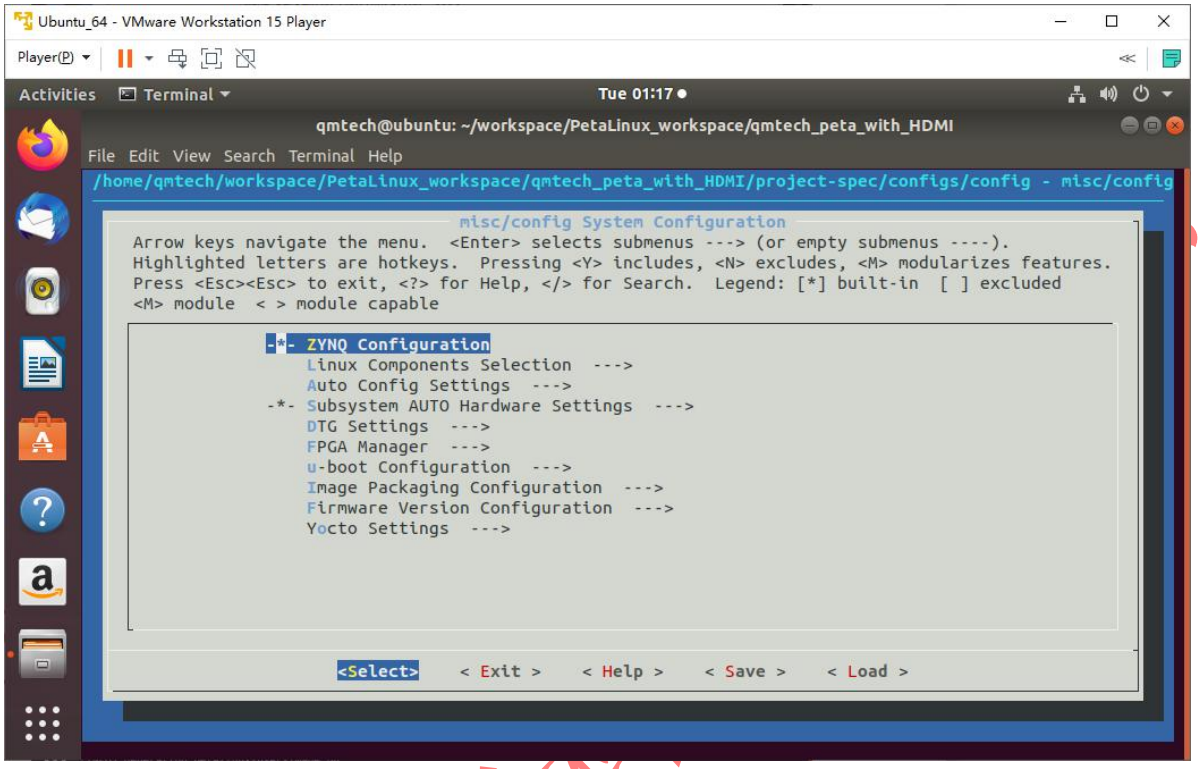


Figure 2-19. ZYNQ Configuration

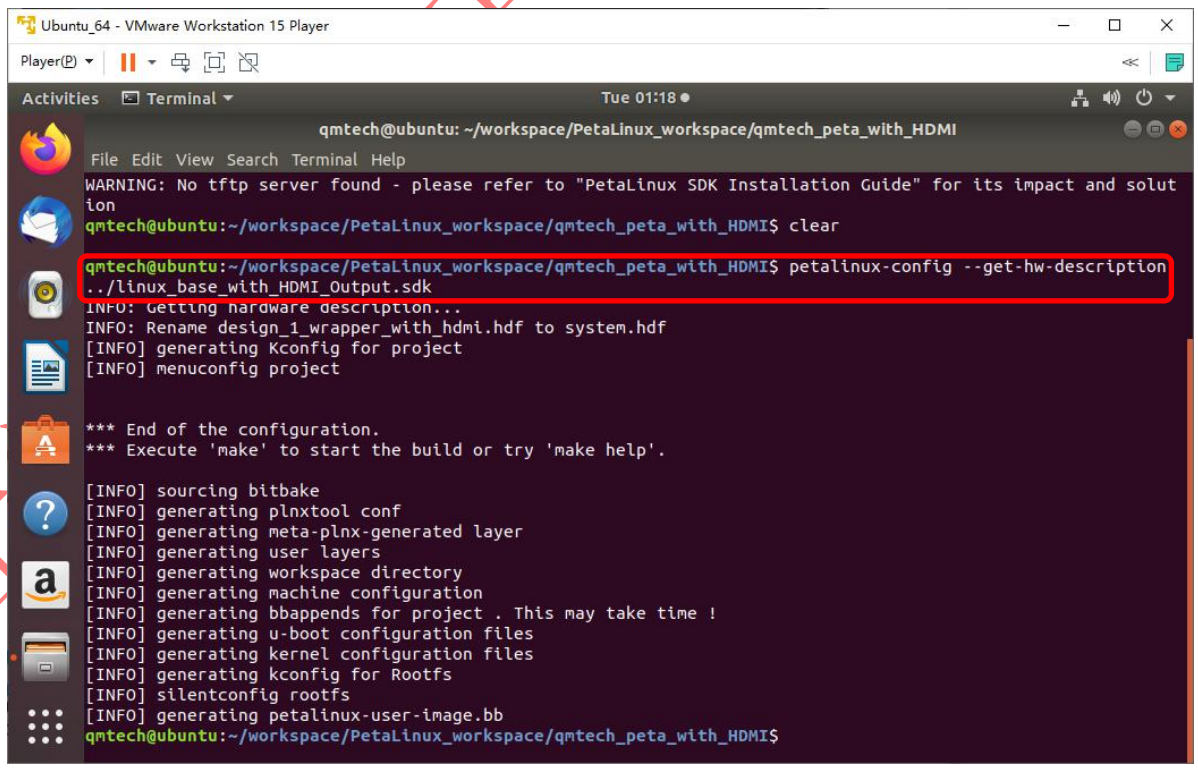


Figure 2-20. Import Hardware Info

Once the hardware info is successfully imported, users may start to modify the Linux kernel to enable the HDMI display port. Mainly two parts needs to be updated here.

- Dynamic Clock Generator
- RGB to DVI Video Encoder

The above two drivers are maintained by Digilent. Users may find the latest drivers in their official Github.
To enable the dynamic clock generator, users need to put the **clk-dglnt-dynclk.c** into below folder.

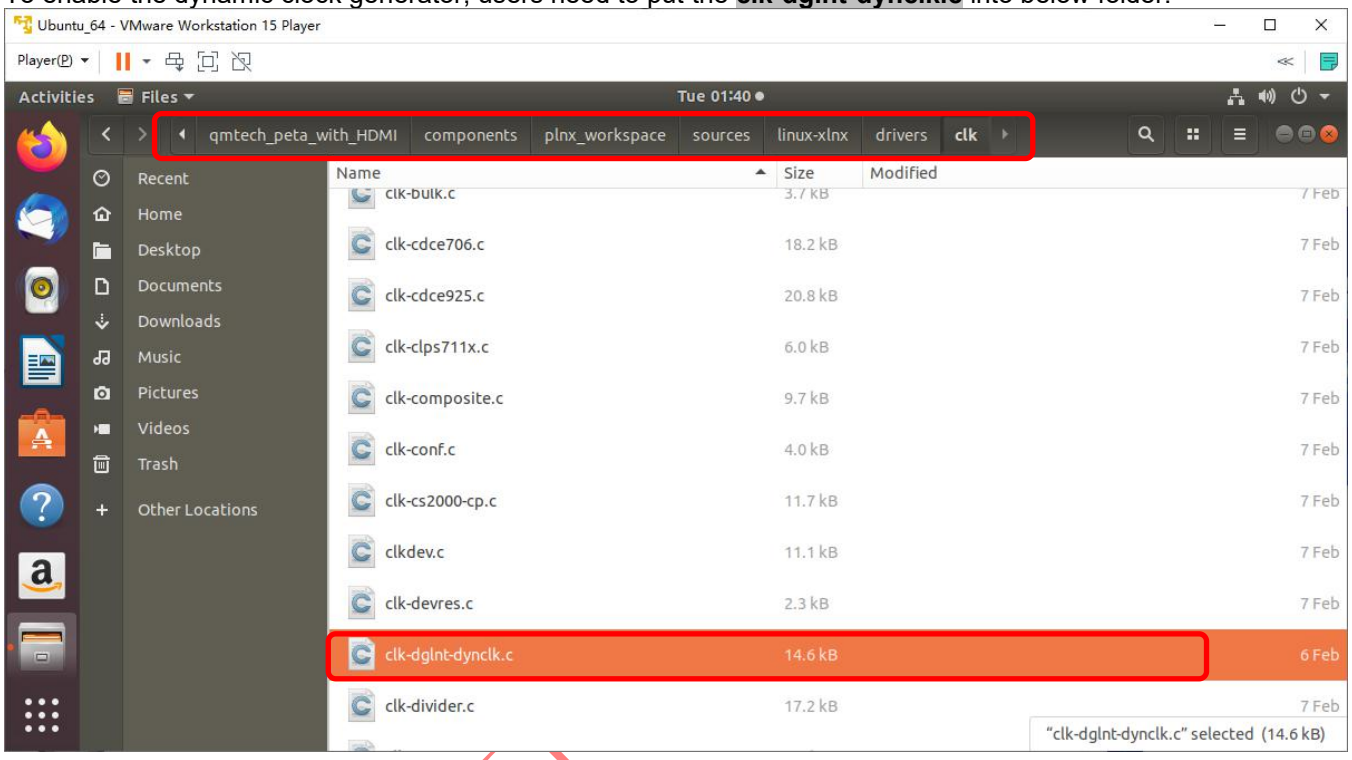
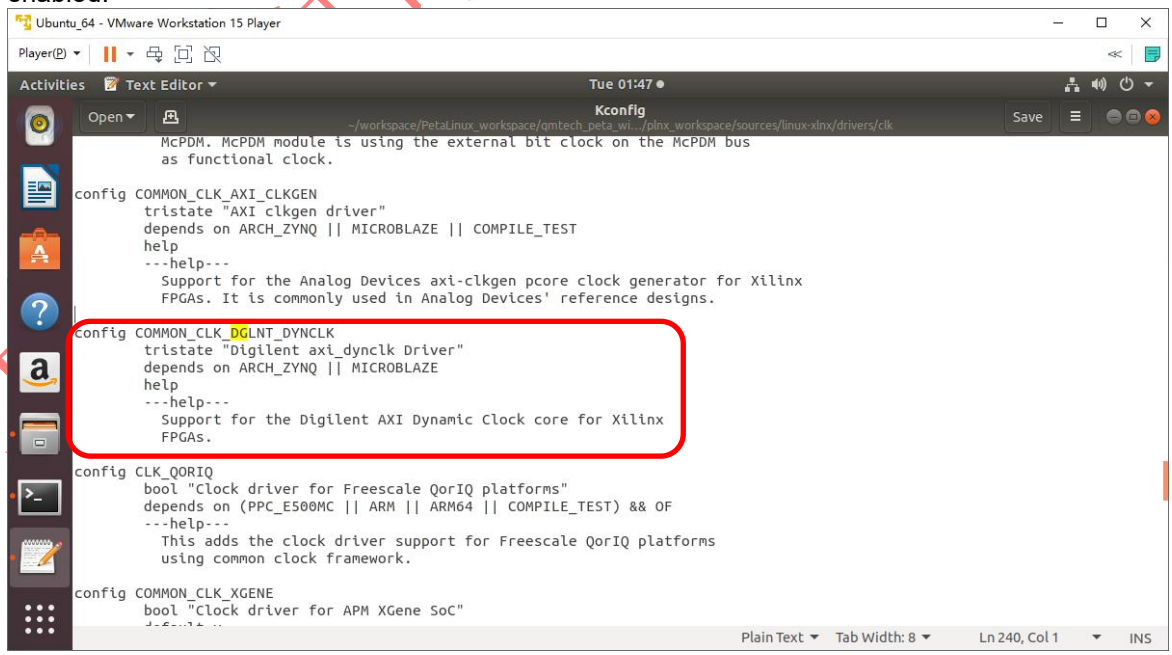


Figure 2-21. Add Clock Driver

Modify the KConfig file in the **clk** folder and add below code line to make the Digilent clock driver module enabled.



Modify the Makefile in **clk** folder accordingly.

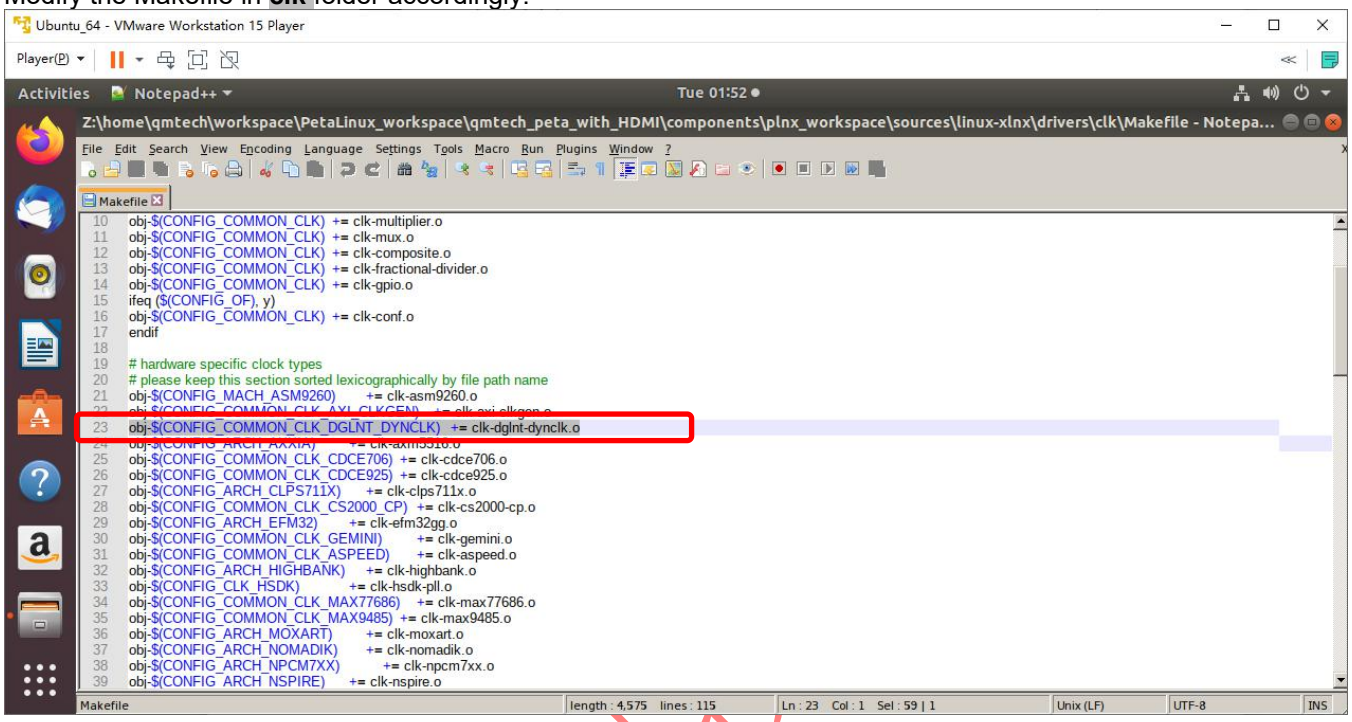


Figure 2-22. Modify the Makefile

To enable the RGB to DVI decoder, users need to put the **digilent_encoder.c** into below folder.

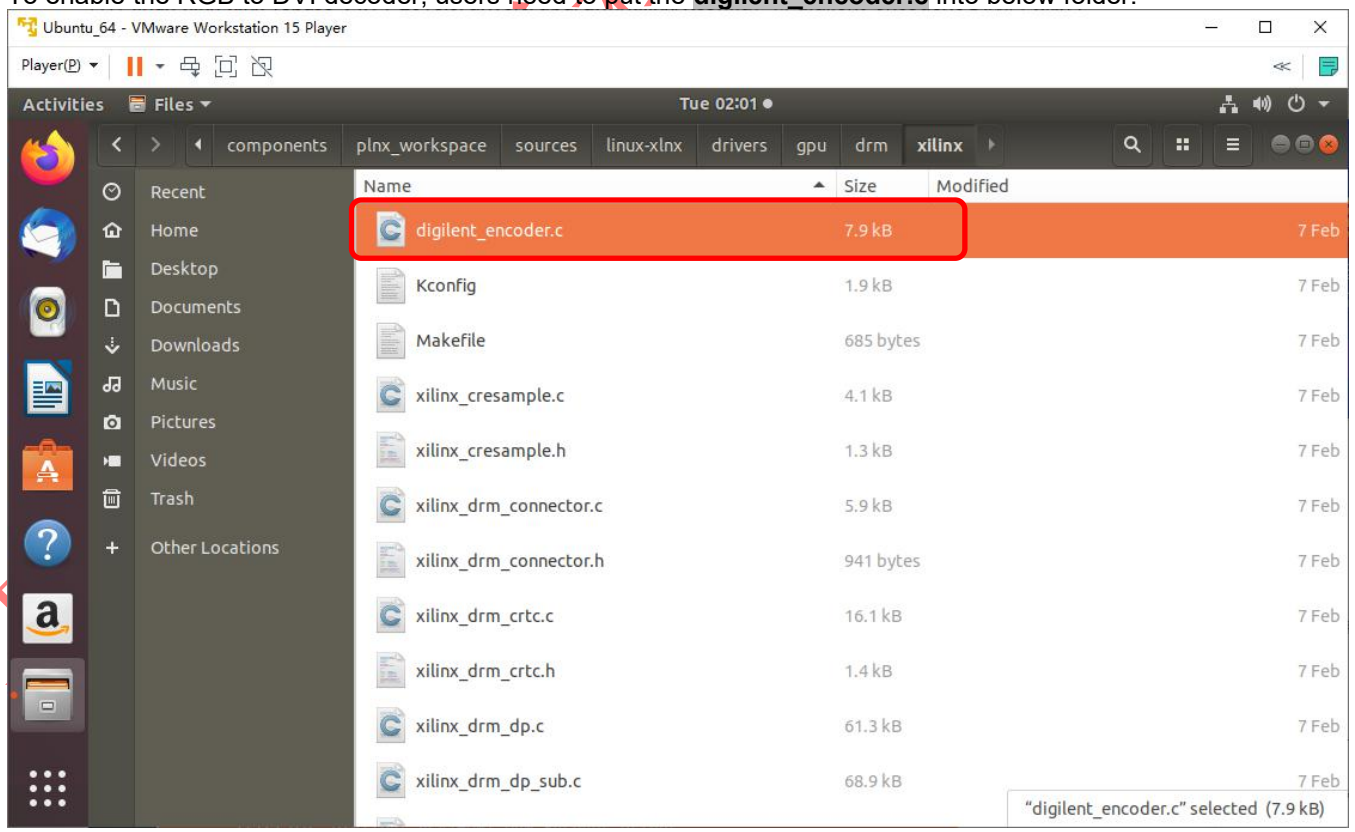


Figure 2-23. Add Driver File

Modify the KConfig file in the **xilinx** folder and add below code line to make the Digilent VGA/HDMI DRM Encoder Driver module enabled.

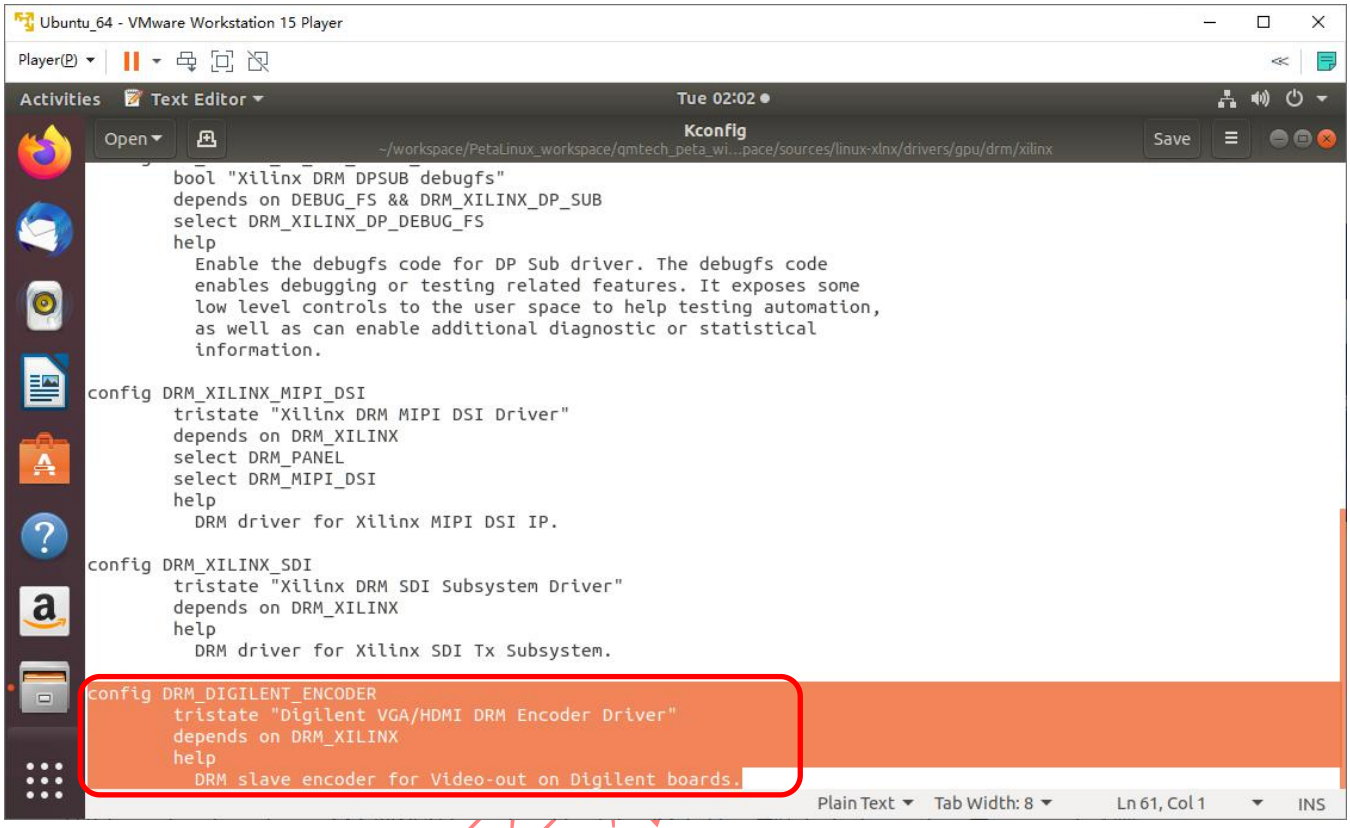


Figure 2-24. Modify the Kconfig File

Modify the Makefile in **xilinx** folder accordingly.

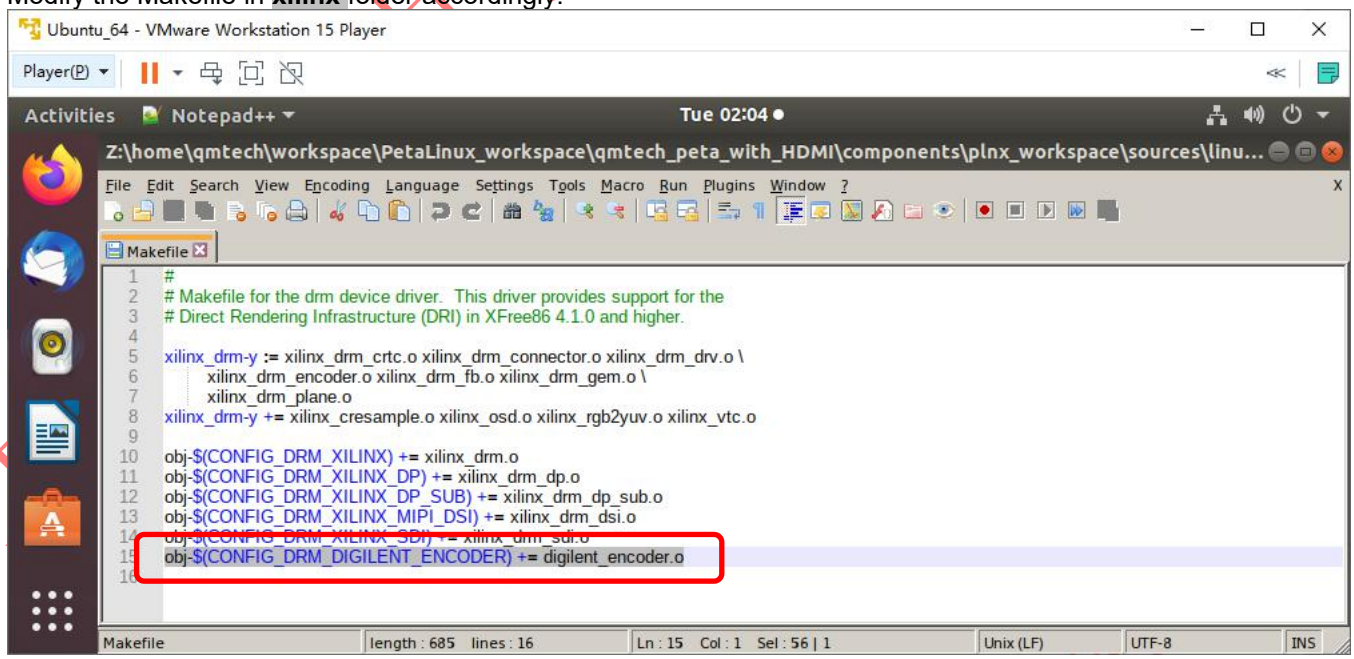


Figure 2-25. Modify the Makefile

Users then start to configure the PetaLinux kernel by command: **petalinux-config -c kernel**
This step may take a really long time. Be patient here and make sure the VM's network status is perfect.
Enable **Digilent axi_dynclk Driver** in **Device Drivers → Common Clock Framework.**

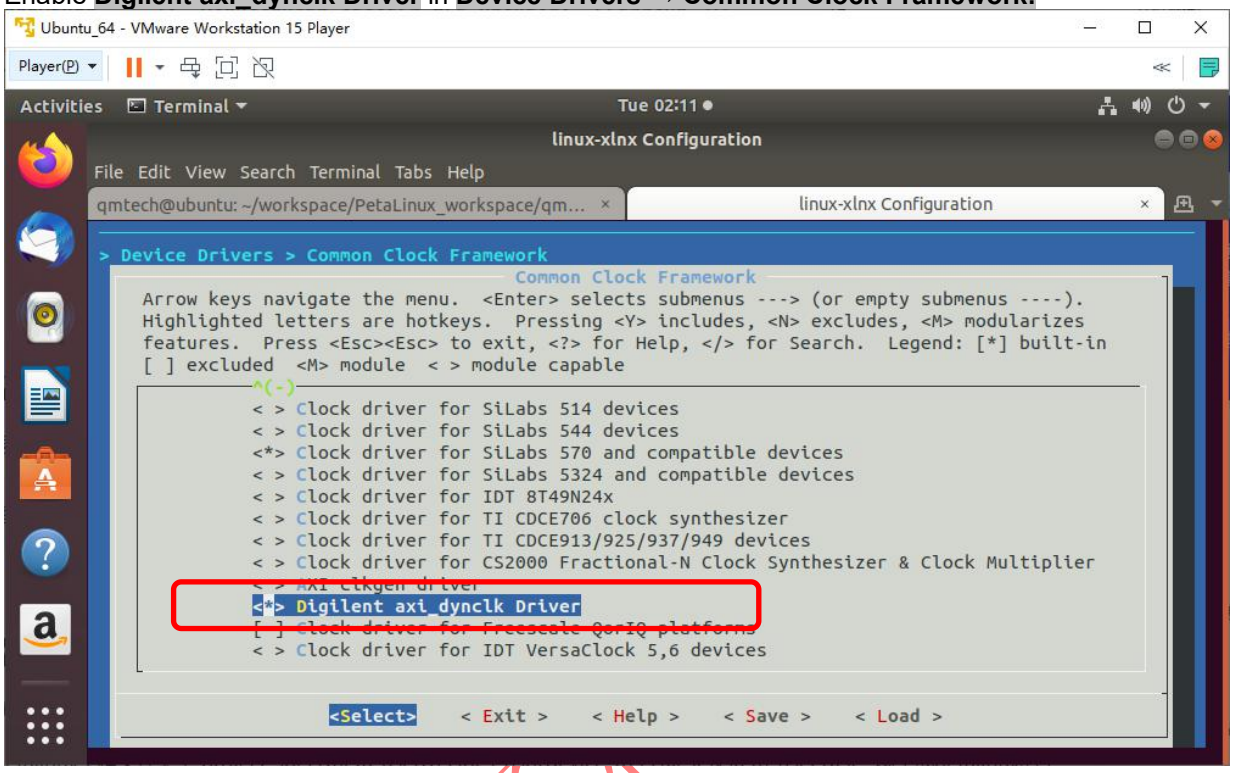
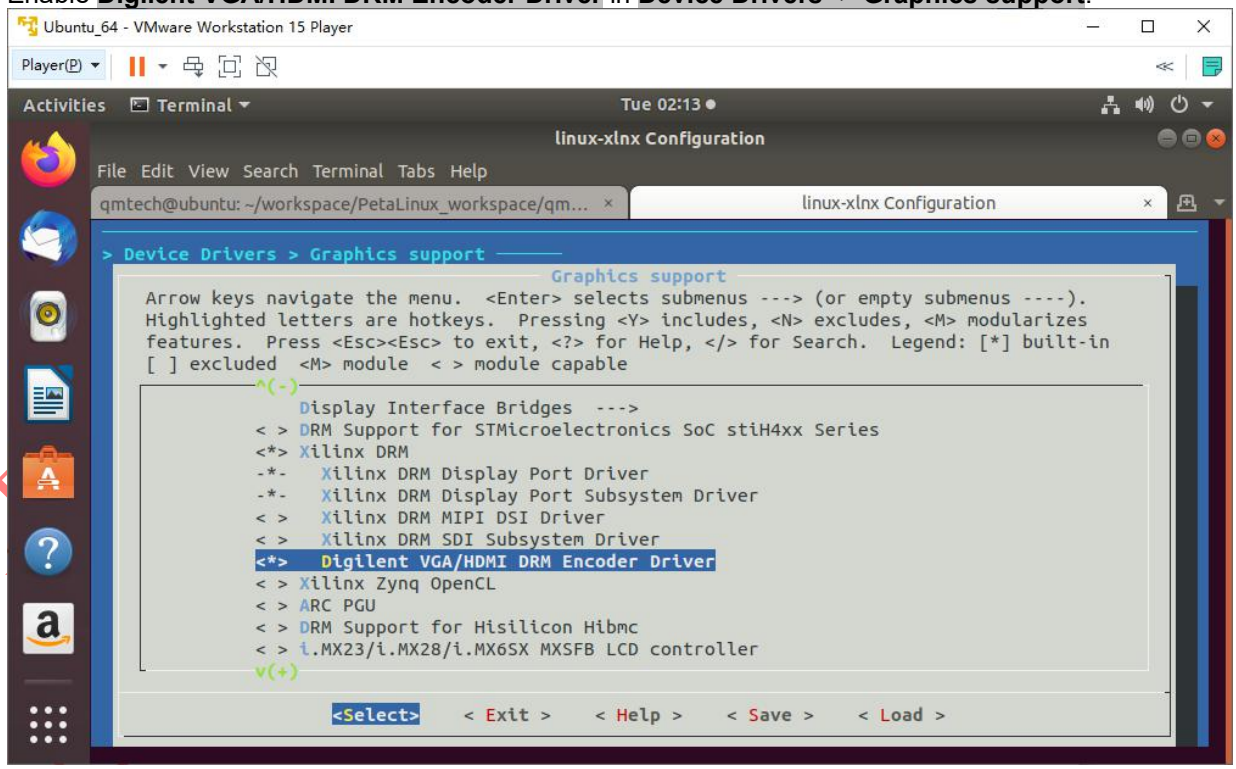


Figure 2-26. Enable Clock Driver

Enable **Digilent VGA/HDMI DRM Encoder Driver** in **Device Drivers -> Graphics support**.



Update the device tree in file **$PROJECT_DIR/project-spec/meta-user/recipes-bsp/device-tree/files/system-user.dtsi**:

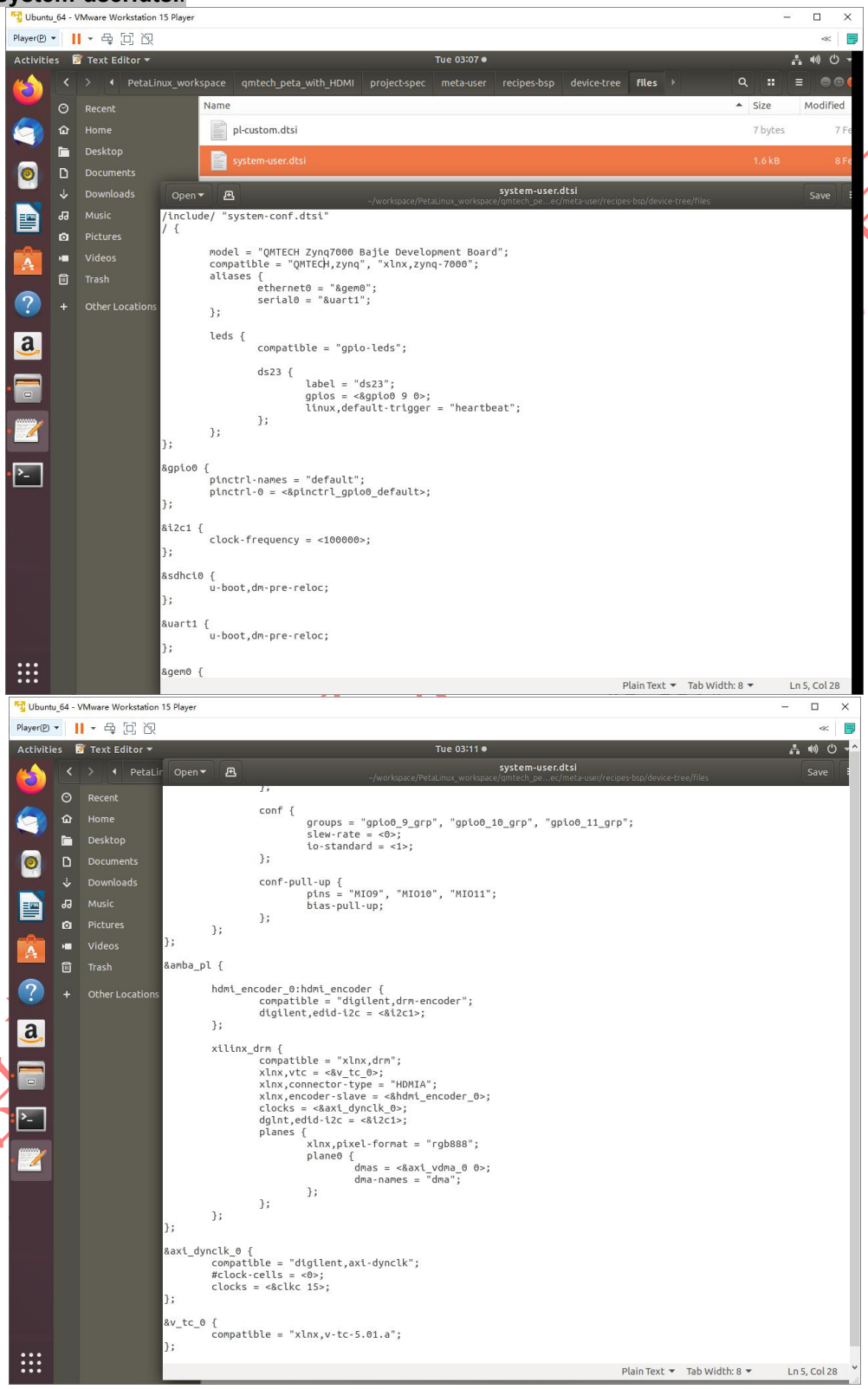


Figure 2-27. Update Device Tree

Once the PetaLinux Kernel info is successfully configured, users may start to configure the PetaLinux file system by command: **petalinux-config -c rootfs**
Below image will display and nothing needs to be changed here.

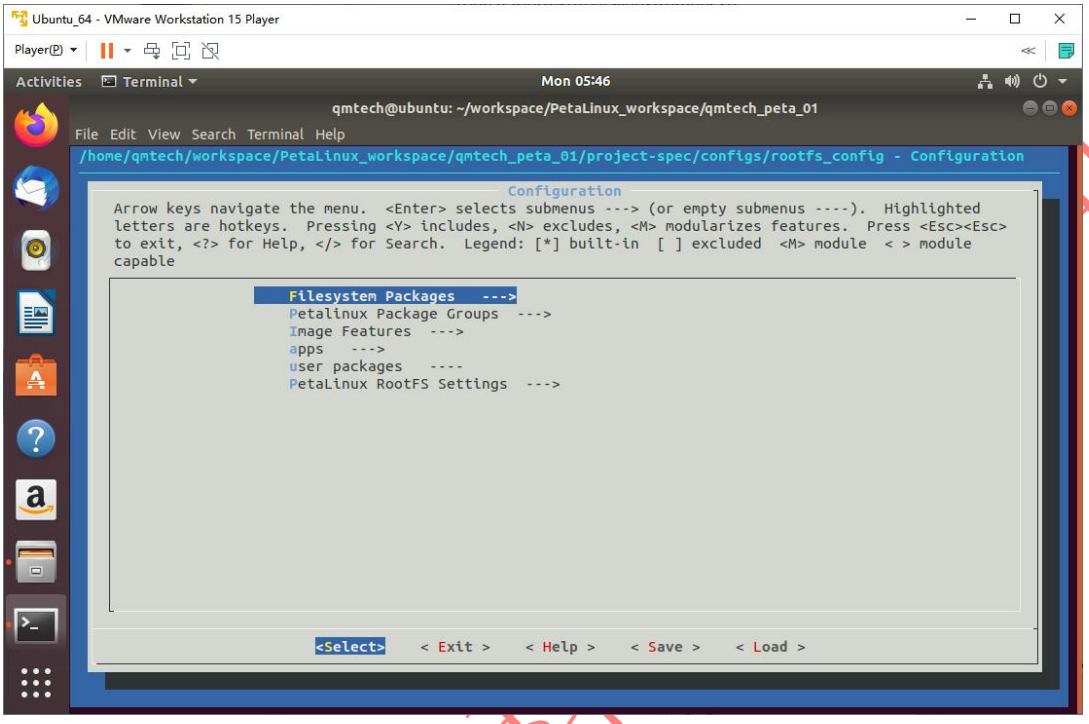


Figure 2-28. No Need to change the rootfs

Start to build the PetaLinux by command: **petalinux-build**

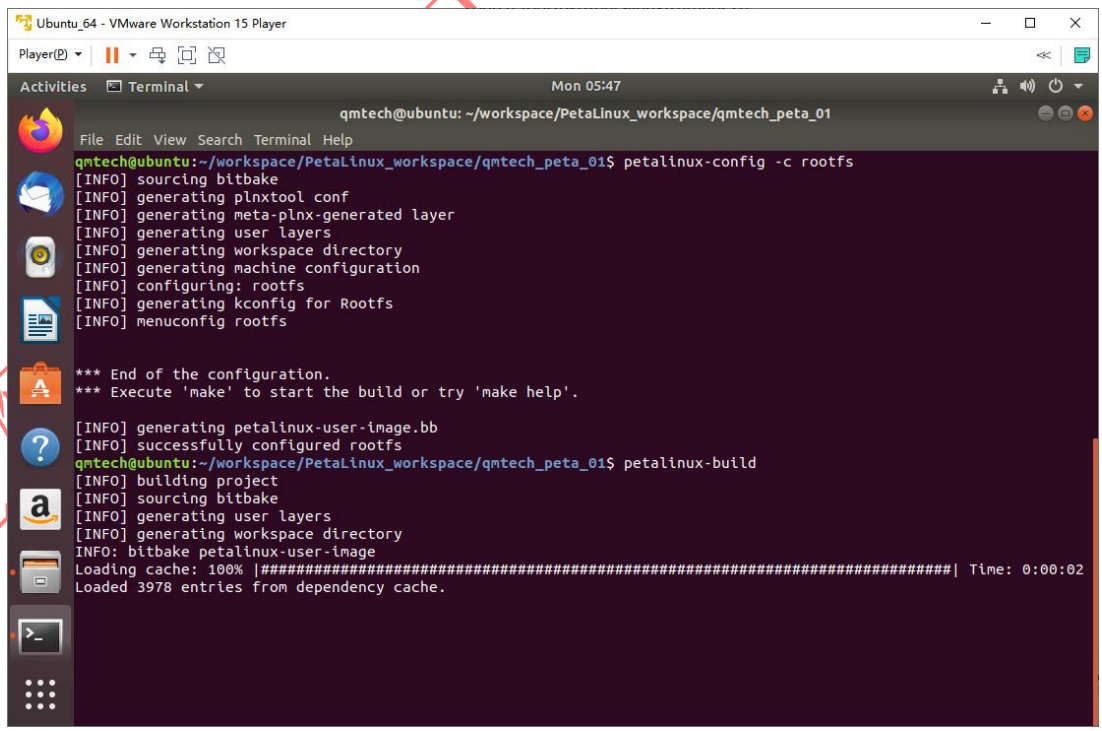
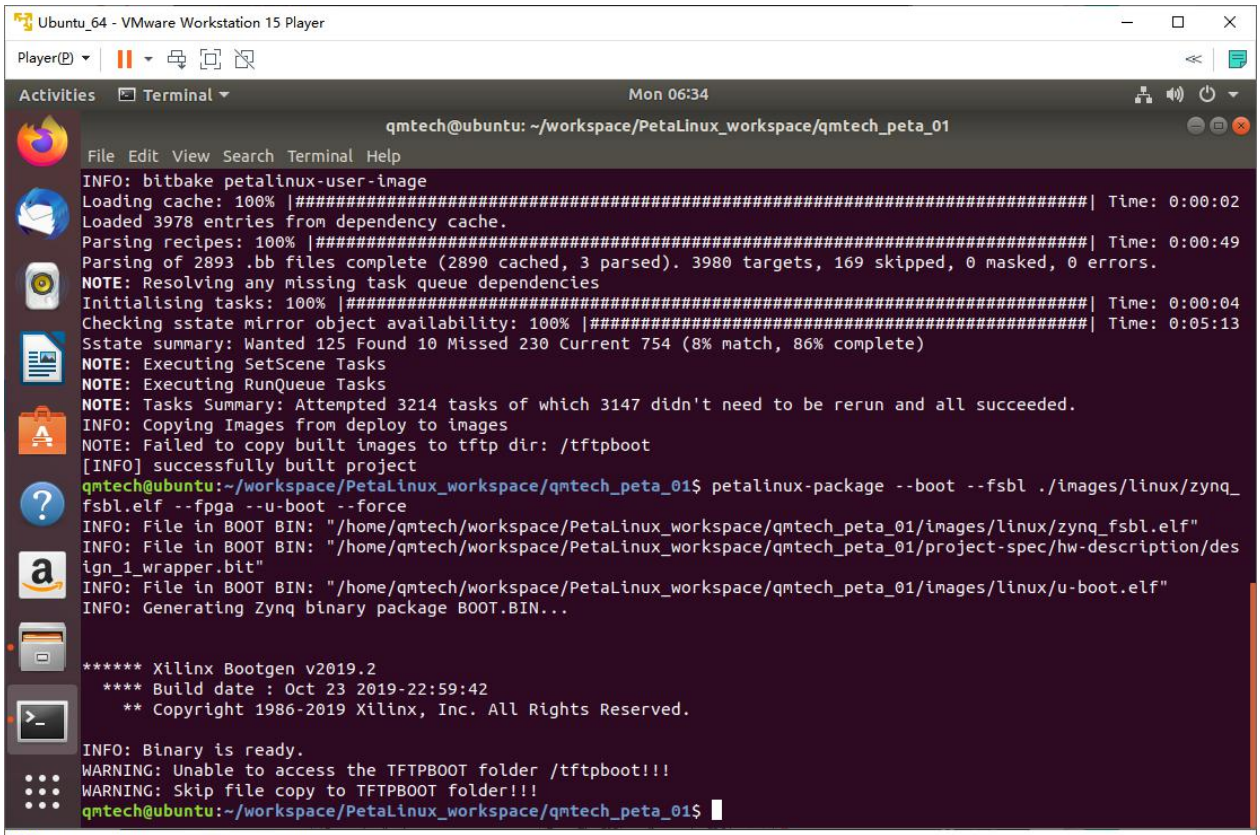
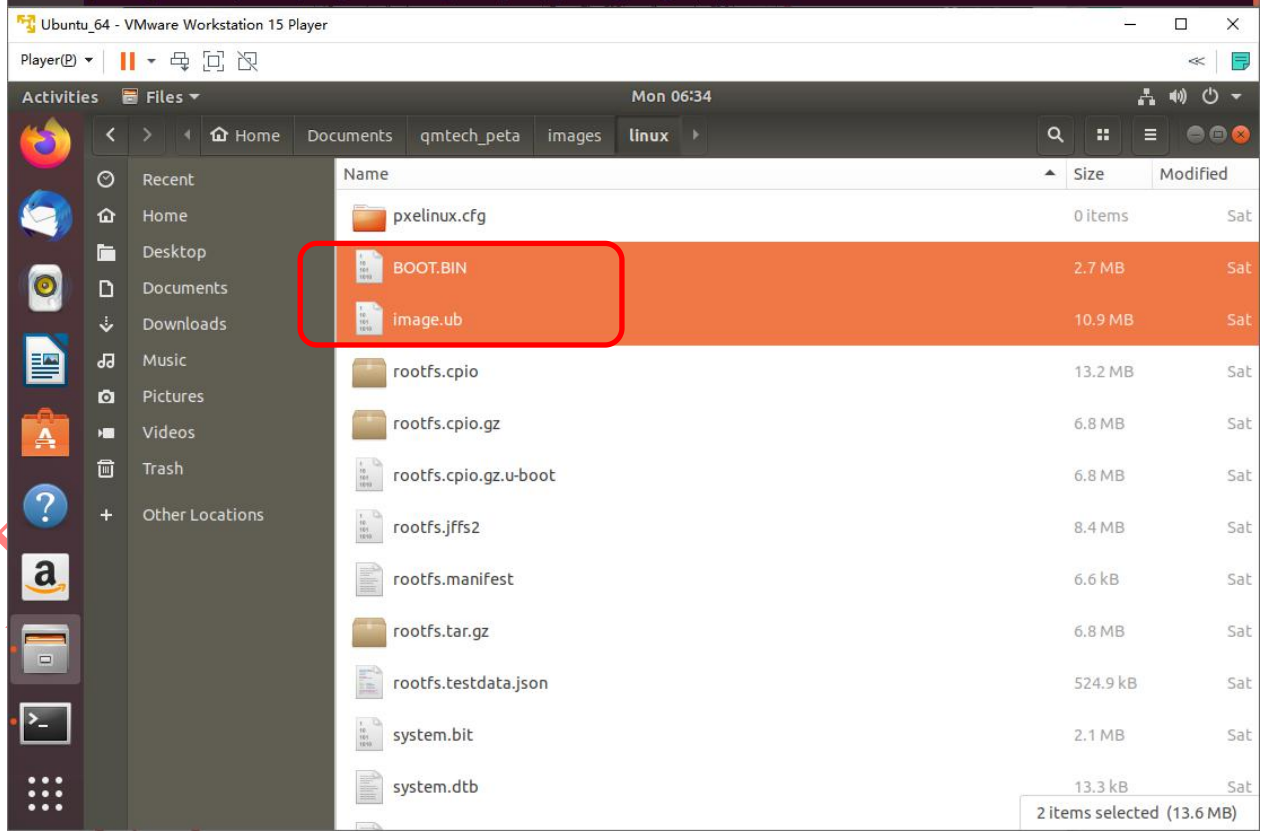


Figure 2-29. Build PetaLinux

After the PetaLinux is successfully built, users could generate the target BOOT.bin and image.ub by command: **petalinux-package --boot --fsbl ./images/linux/zynq_fsbl.elf --fpga --u-boot --force**





Copy the BOOT.bin and image.ub into MicroSD card and then insert the MicroSD card into ZYNQ7000 Bajie Board. Plug the MiniUSB cable into the Bajie Board and then power on the board with 5V DC power source. Below image shows the serial output from the board.

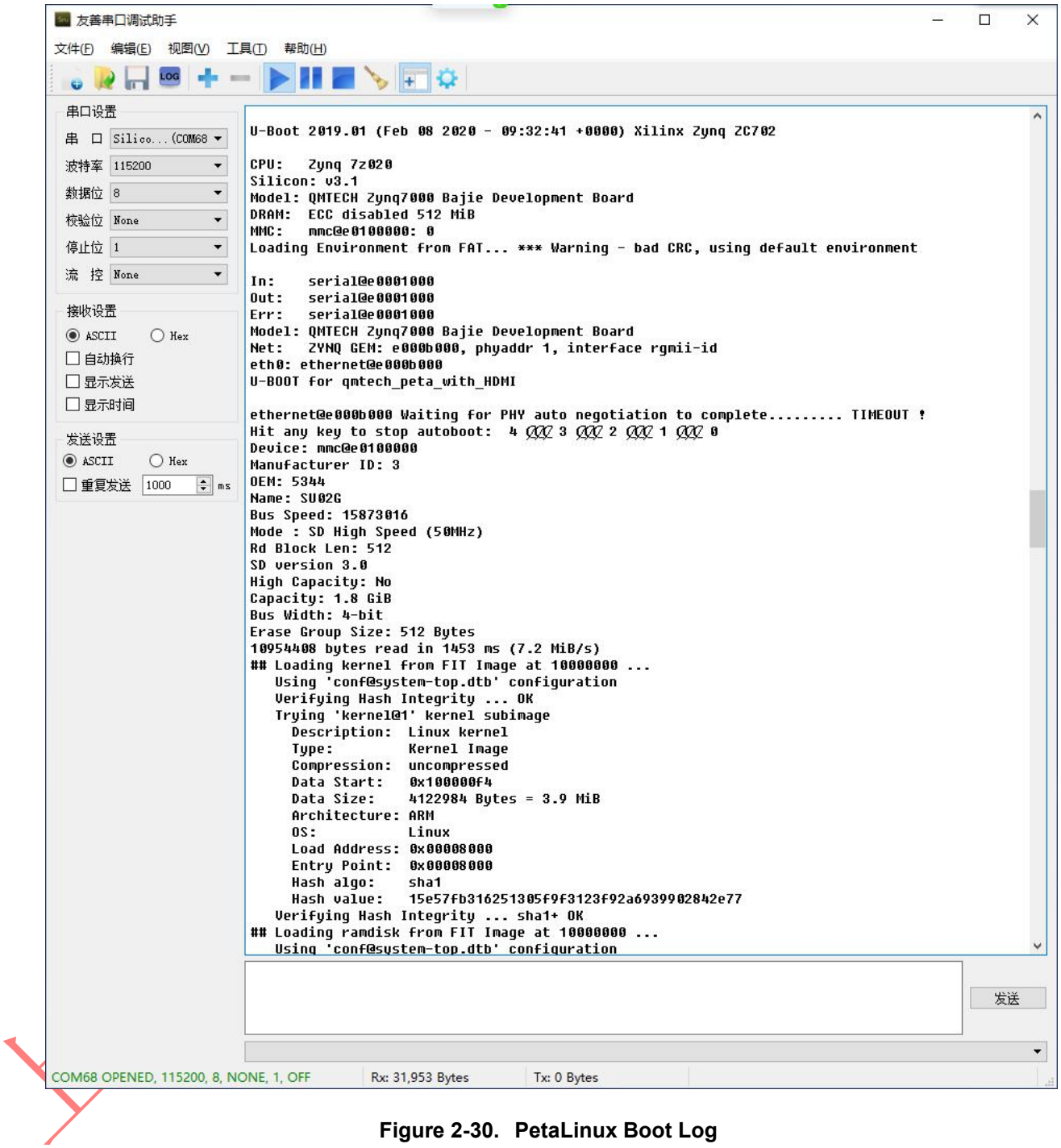


Figure 2-30. PetaLinux Boot Log

Below info will be displayed on the HDMI monitor that the Bajie Board connected.

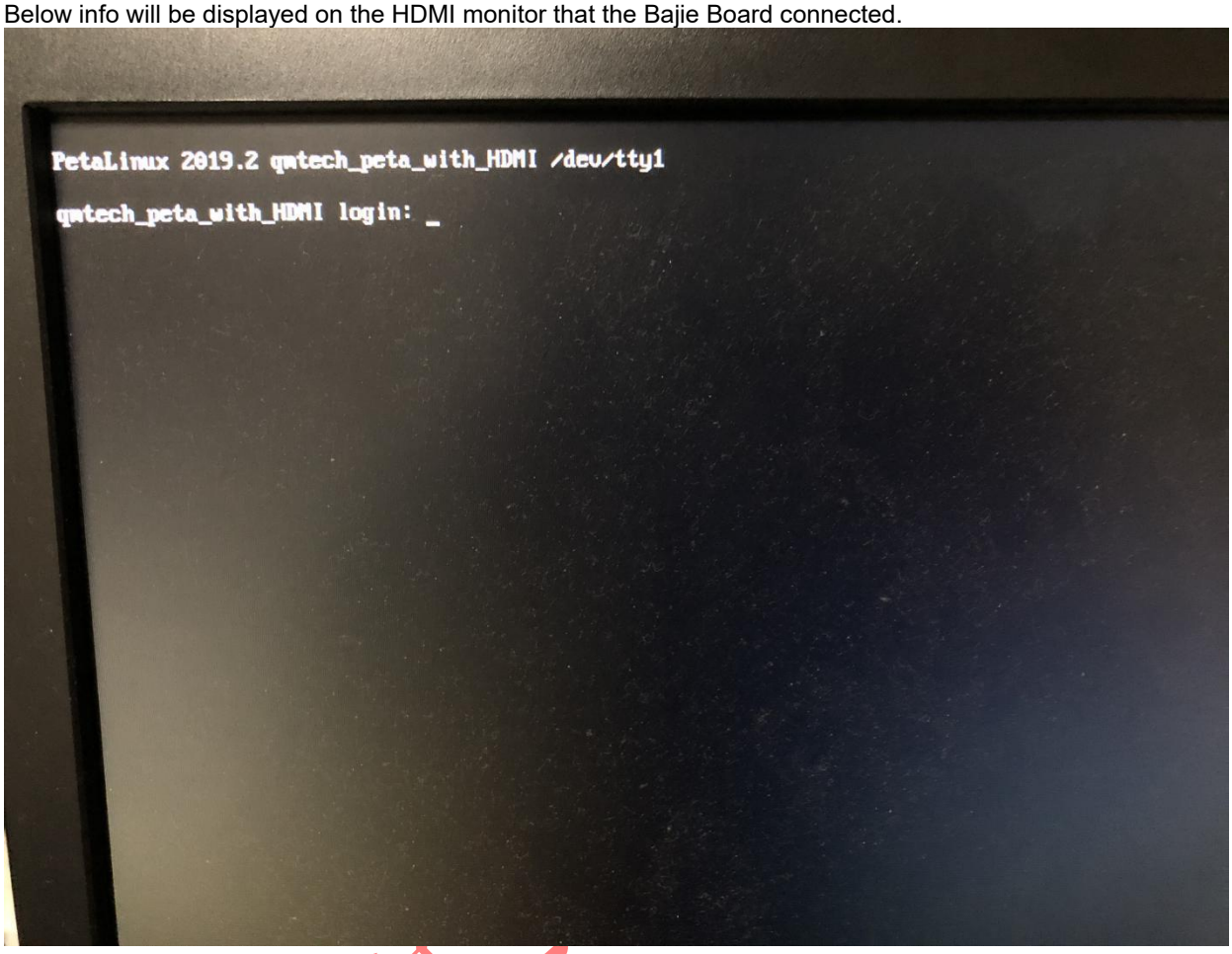


Figure 2-31. HDMI Display

3. Reference

[1] ug585-Zynq-7000-TRM.pdf
[2] ds187-XC7Z010-XC7Z020-Data-Sheet.pdf
[3] ug865-Zynq-7000-Pkg-Pinout.pdf
[4] MT41K256M16TW-107:P.pdf
[5] tps563201.pdf
[6] RTL8211E-VL.PDF

4. Revision

Doc. Rev.	Date	Comments
0.1	03/03/2020	Initial Version.
1.0	03/21/2020	V1.0 Formal Release.