Creating Embedded Linux Solutions

Course Workbook

Table of Contents

About this Workbook

The contents of this workbook are created by Adiuvo Engineering & Training, Ltd.

If you have any questions about the contents, or need assistance, please contact Adam Taylor at adam@adiuvoengineering.com.

Pre-Lab
Creating Embedded Linux Solutions

Required Hardware

Arty Z7-20

SD Card

Micro USB Cable

Downloads and Installations

Step 1 – Download and install the following at least 1 day prior to the workshop. This may take a significant amount of time and drive space.

Watch the video available here to show how to configure the installation

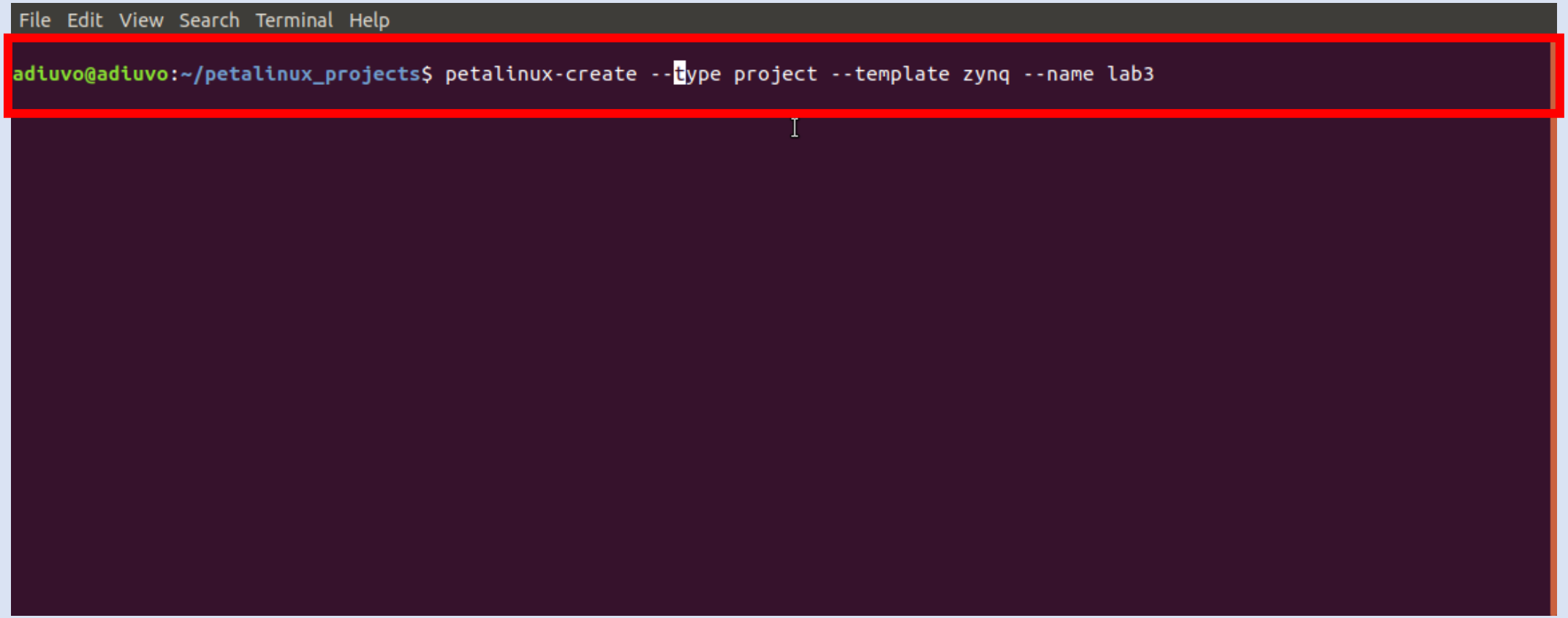
Vitis 2021.1	Download
PetaLinux 2021.1	Download

Lab 3

Creating Embedded Linux Solutions

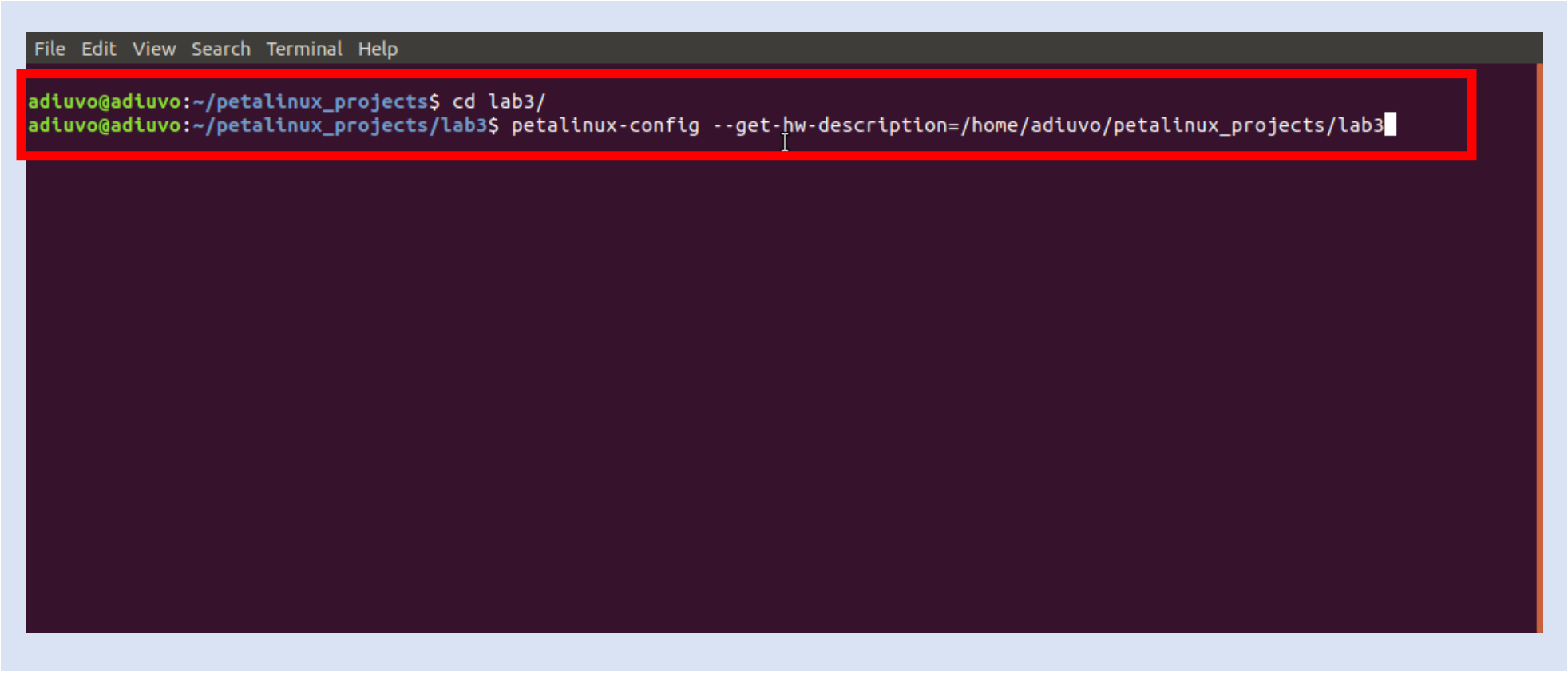
Lab 3: Creating Embedded Linux Solutions

Step 1 – Using a Linux Development machine with PetaLinux 2021.1 installed create a new project



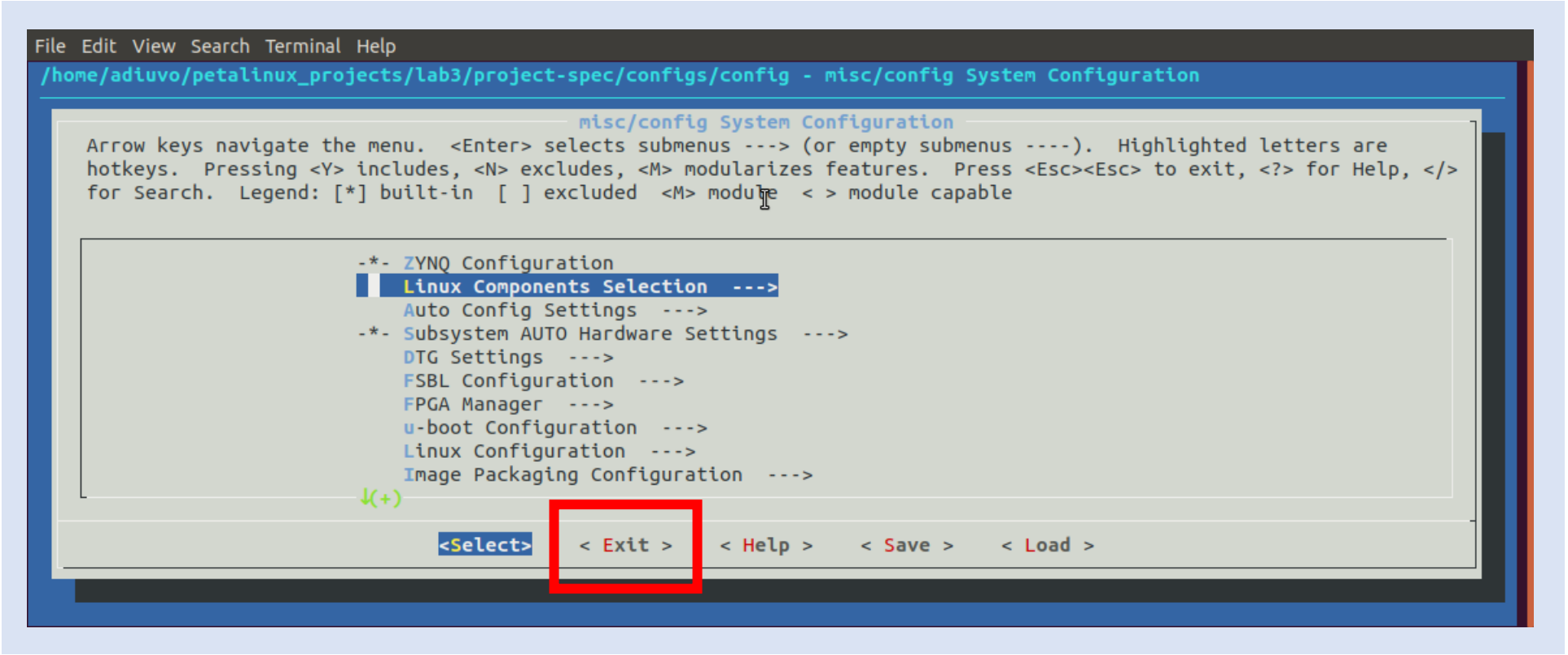
Lab 3: Creating Embedded Linux Solutions

Step 2 – Change directory into the project, import the XSA exported from Lab One



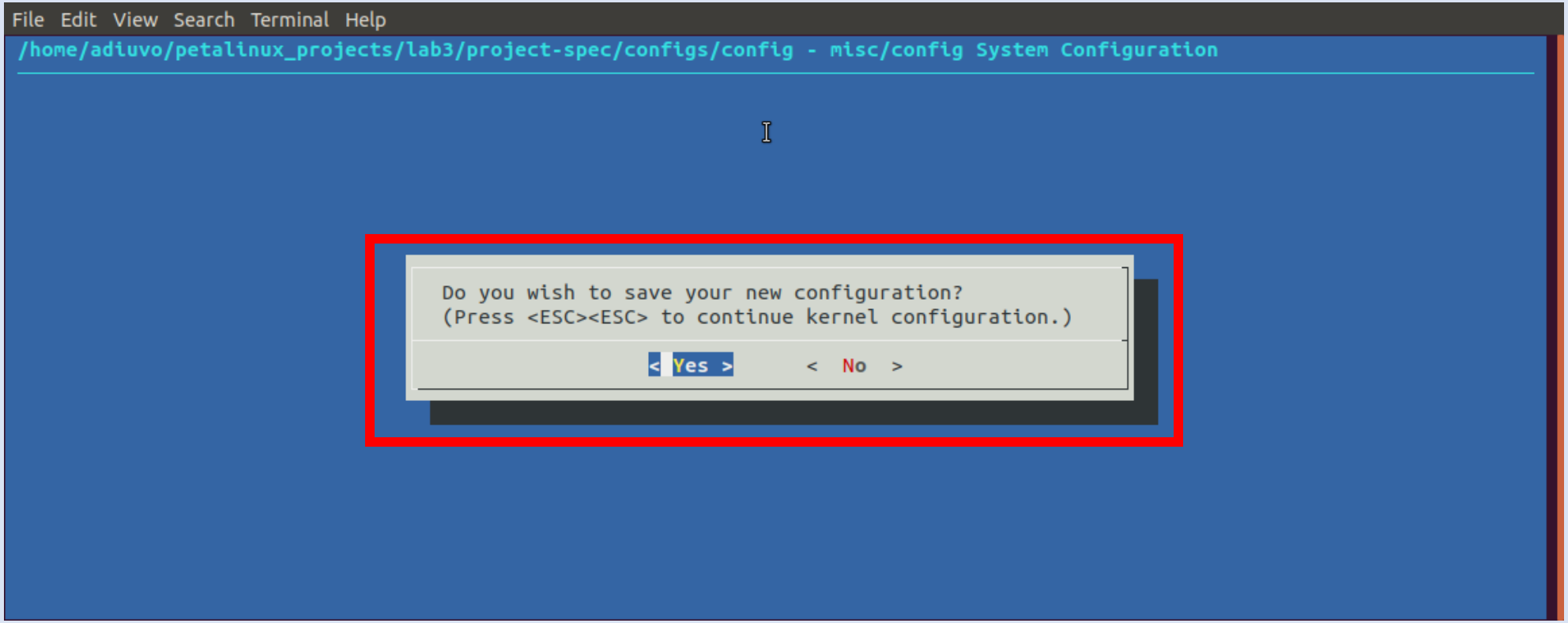
Lab 3: Creating Embedded Linux Solutions

Step 3 – Exit the dialog which is opened following the XSA Import.



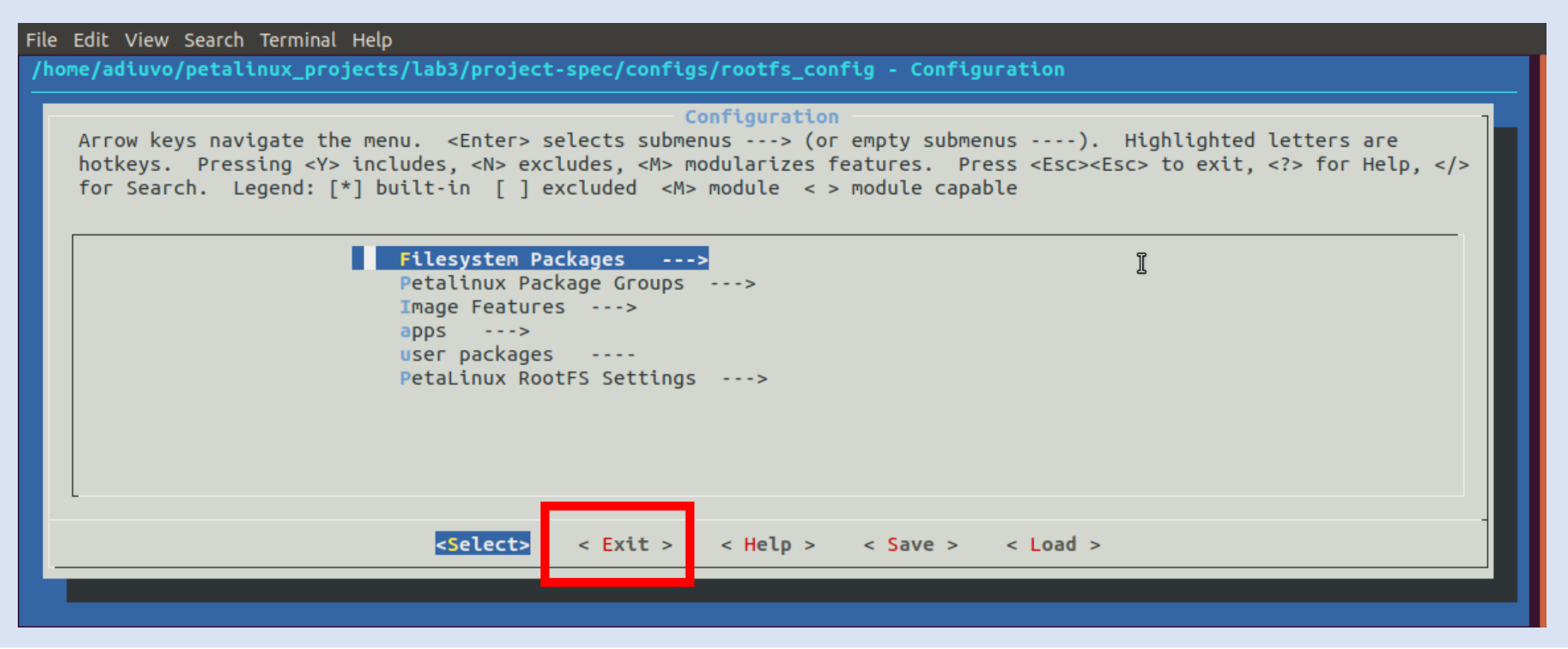
Lab 3: Creating Embedded Linux Solutions

Step 4 – When asked save any changes



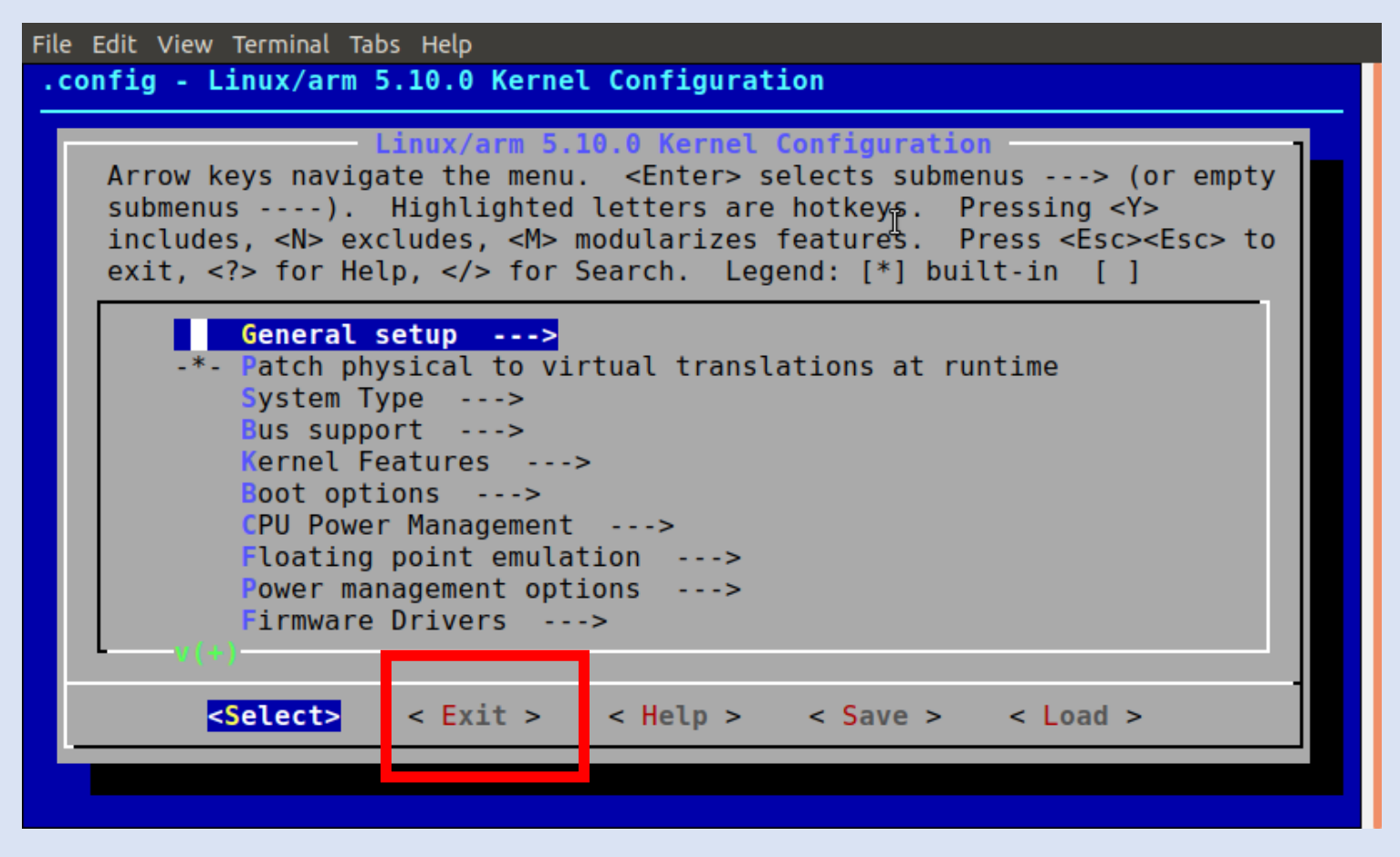
Lab 3: Creating Embedded Linux Solutions

Step 5 – Enter the command petalinux-config –c rootfs to examine the rootfs dialog. Close without saving changes after exploring the settings.



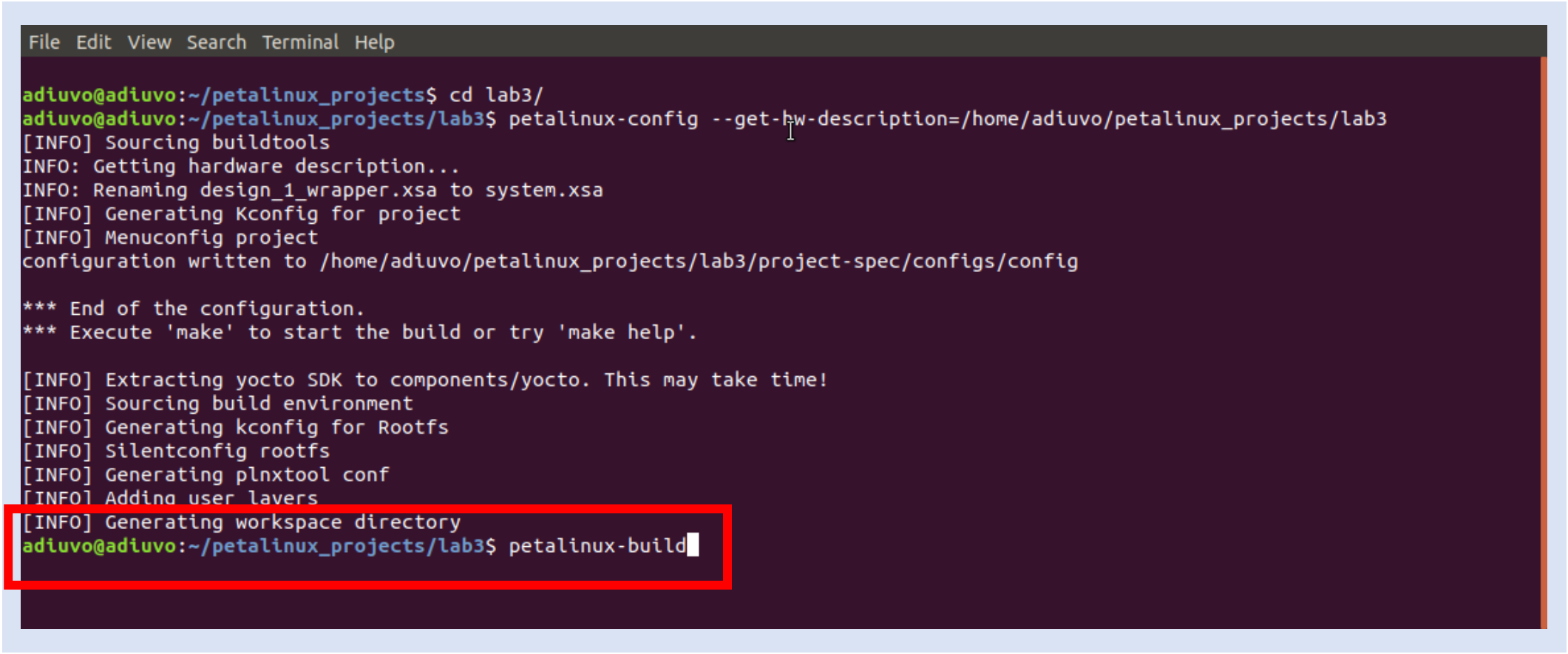
Lab 3: Creating Embedded Linux Solutions

Step 6 – Enter the command petalinux-config –c kernel to view the config settings, do not change anything and exist after exploring.



Lab 3: Creating Embedded Linux Solutions

Step 7 – Build the petalinux project with the command petalinux-build – this may take some time.



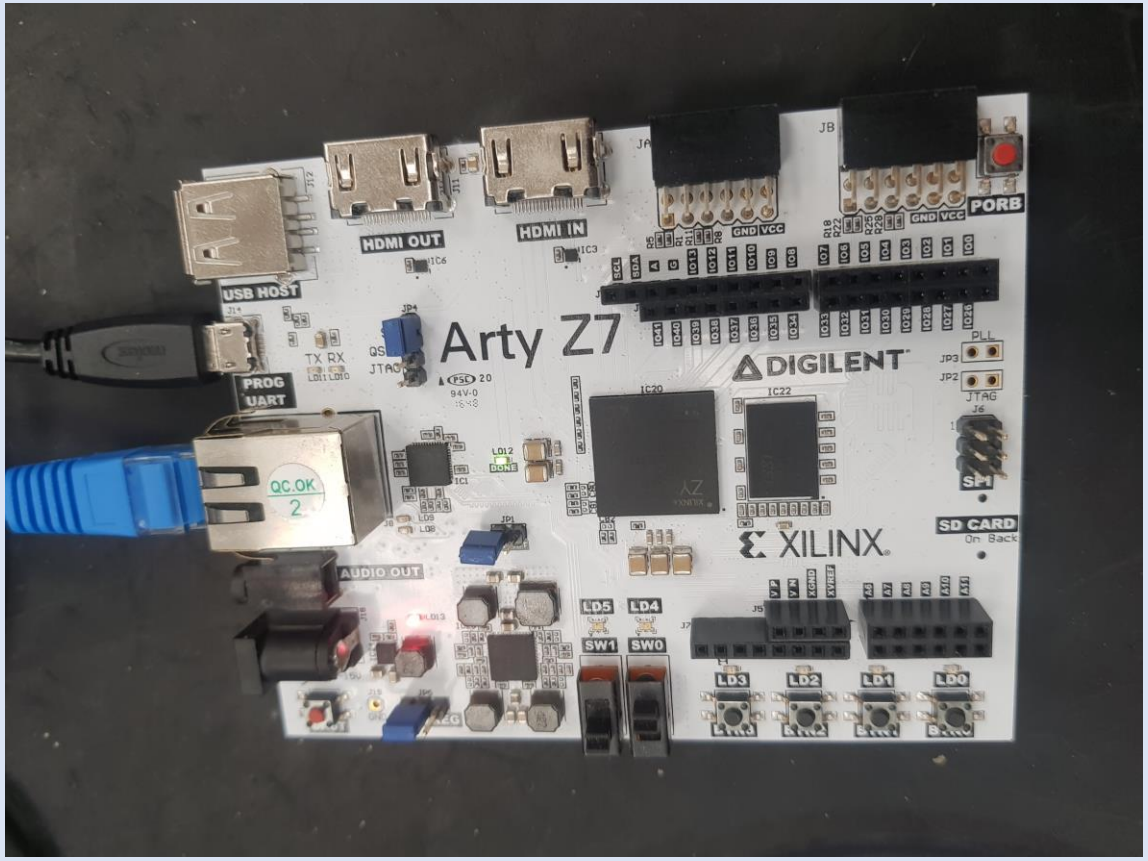
Lab 3: Creating Embedded Linux Solutions

Step 8 – Package the petalinux project to create the boot.bin necessary

```
File  Edit  View  Search  Terminal  Help

adiuvo@adiuvo:~/petalinux_projects/lab3$ cd images/linux/
adiuvo@adiuvo:~/petalinux_projects/lab3/images/linux$ petalinux-package --boot --fsbl zynq_fsbl.elf --u-boot u-boot.elf  --fpga
 system.bit --force
[INFO] Sourcing buildtools
INFO: Getting system flash information...
INFO: File in BOOT BIN: "/home/adiuvo/petalinux_projects/lab3/images/linux/zynq_fsbl.elf"
INFO: File in BOOT BIN: "/home/adiuvo/petalinux_projects/lab3/images/linux/system.bit"
INFO: File in BOOT BIN: "/home/adiuvo/petalinux_projects/lab3/images/linux/u-boot.elf"
INFO: File in BOOT BIN: "/home/adiuvo/petalinux_projects/lab3/images/linux/system.dtb"
INFO: Generating Zynq binary package BOOT.BIN...


****** Xilinx Bootgen v2021.1
  **** Build date : May 28 2021-21:36:22
    ** Copyright 1986-2021 Xilinx, Inc. All Rights Reserved.


[INFO]   : Bootimage generated successfully

INFO: Binary is ready.
WARNING: Unable to access the TFTPBOOT folder /tftpboot!!!
WARNING: Skip file copy to TFTPBOOT folder!!!
adiuvo@adiuvo:~/petalinux_projects/lab3/images/linux$
```

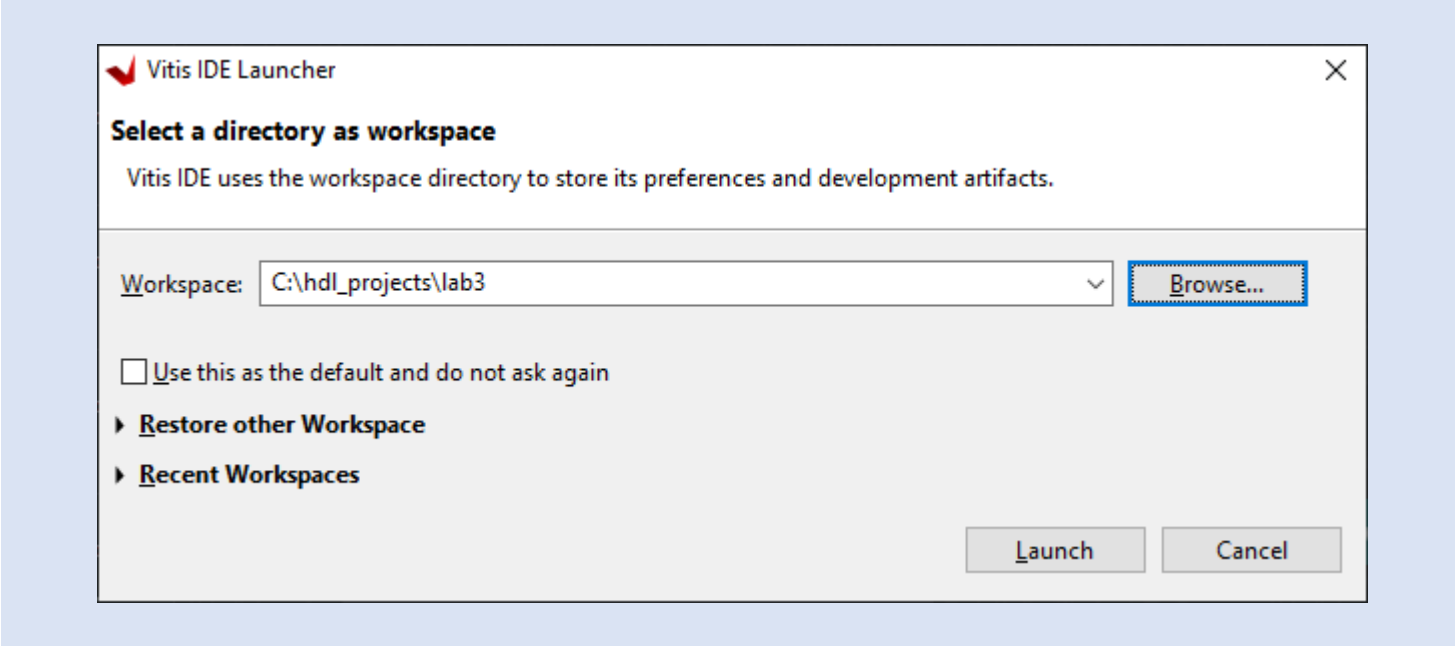
Lab 3: Creating Embedded Linux Solutions

Step 9 – Copy the image.ub boot.bin and boot.scr to the SD Card, insert in the Arty Z7 – ensure it is set for SD Card boot. Connect the power and Ethernet cable.



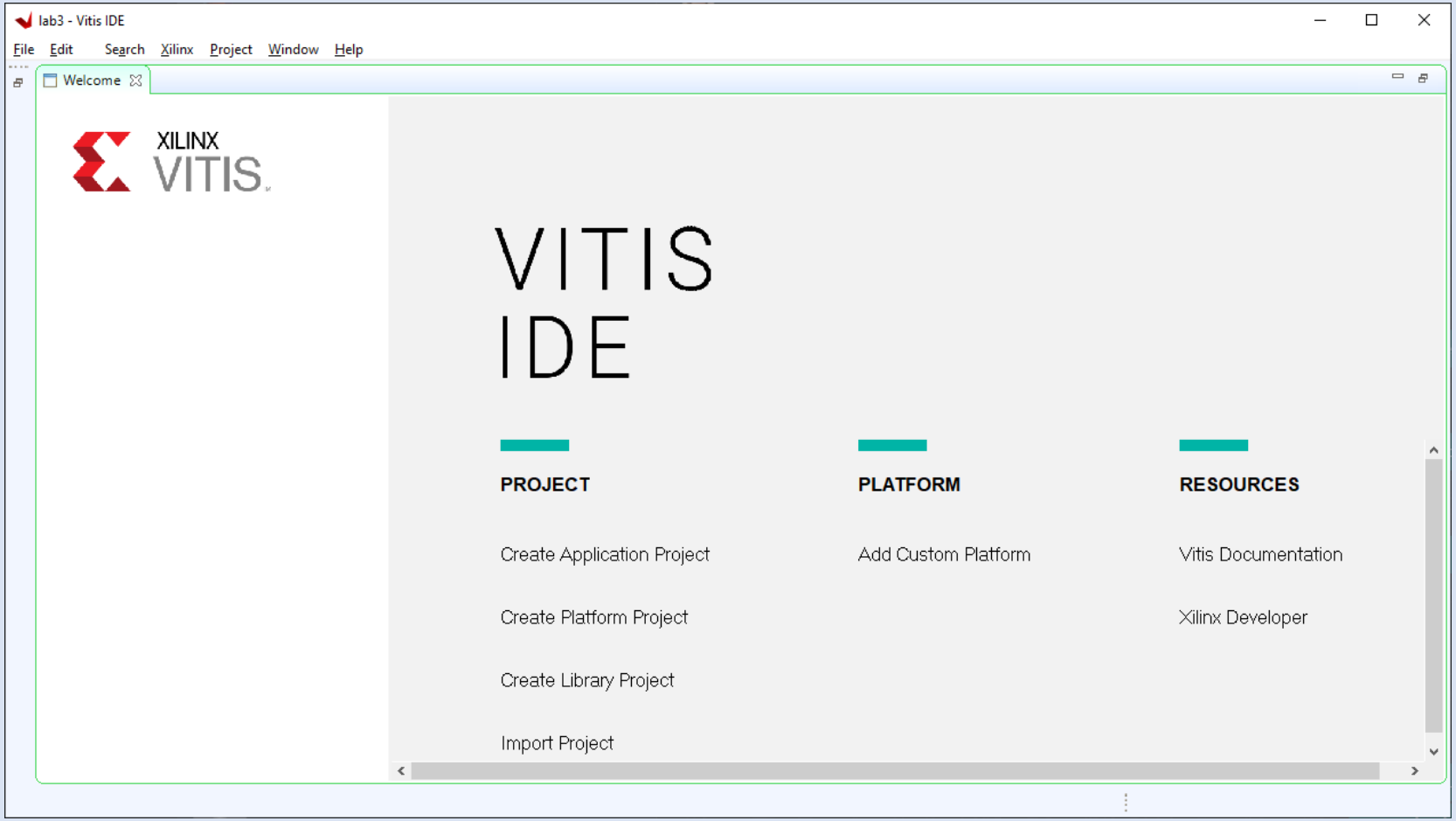
Lab 3: Creating Embedded Linux Solutions

Step 10 – Vitis will start and ask for the workspace, create a new directory using browse to store the projects.



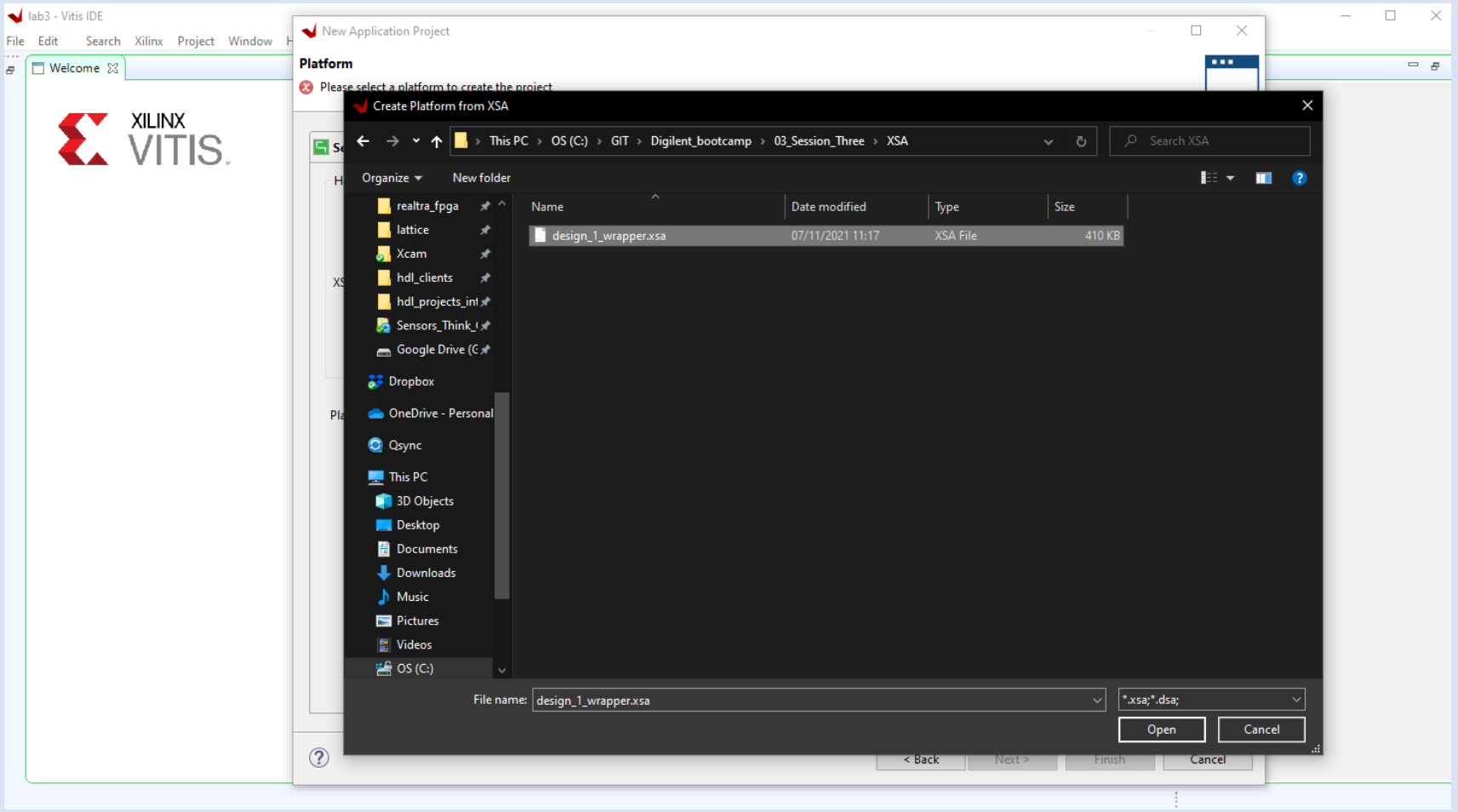
Lab 3: Creating Embedded Linux Solutions

Step 11 – Select Create Application Project



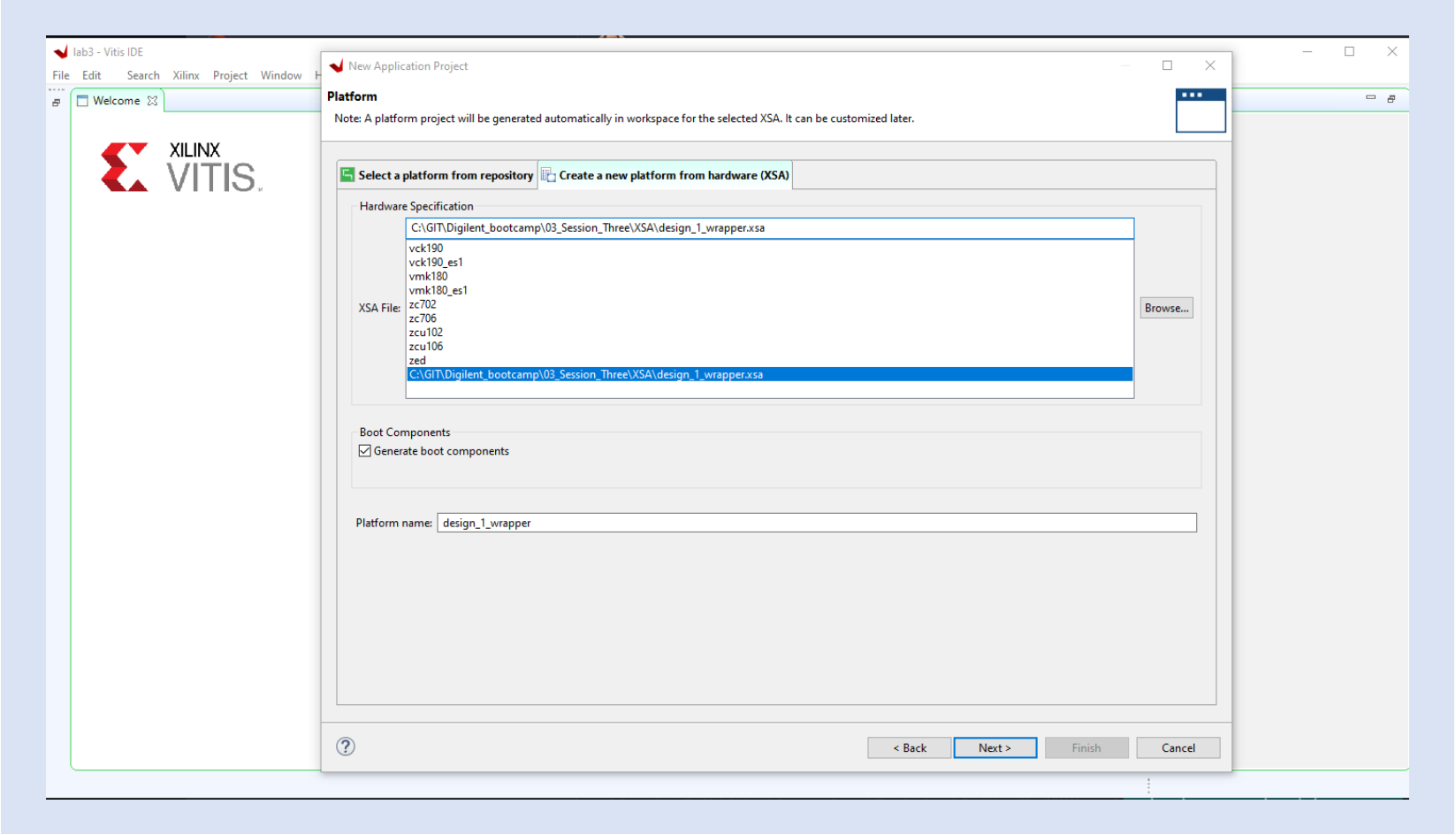
Lab 3: Creating Embedded Linux Solutions

Step 12 – Select the XSA downloaded with GitHub Repository under lab three



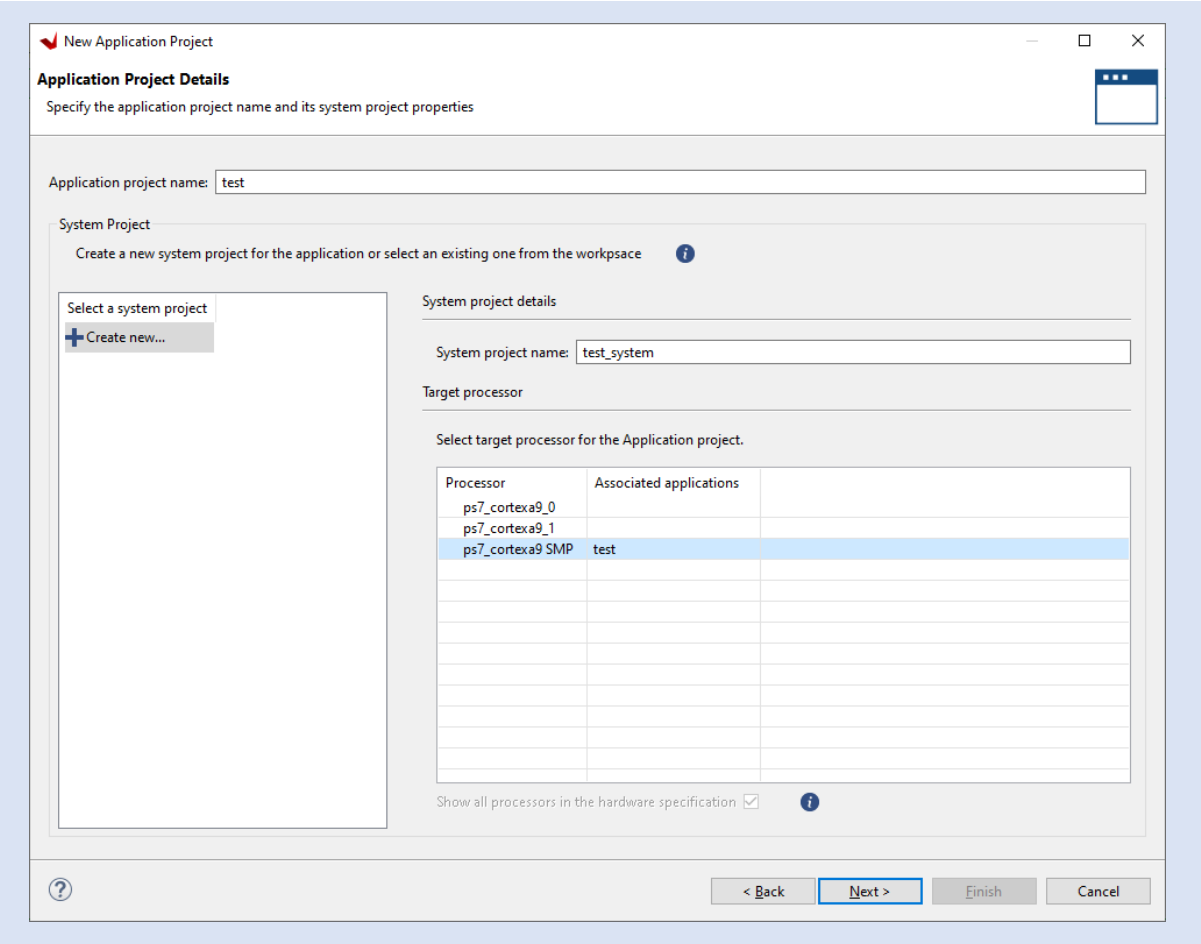
Lab 3: Creating Embedded Linux Solutions

Step 13 – Click next



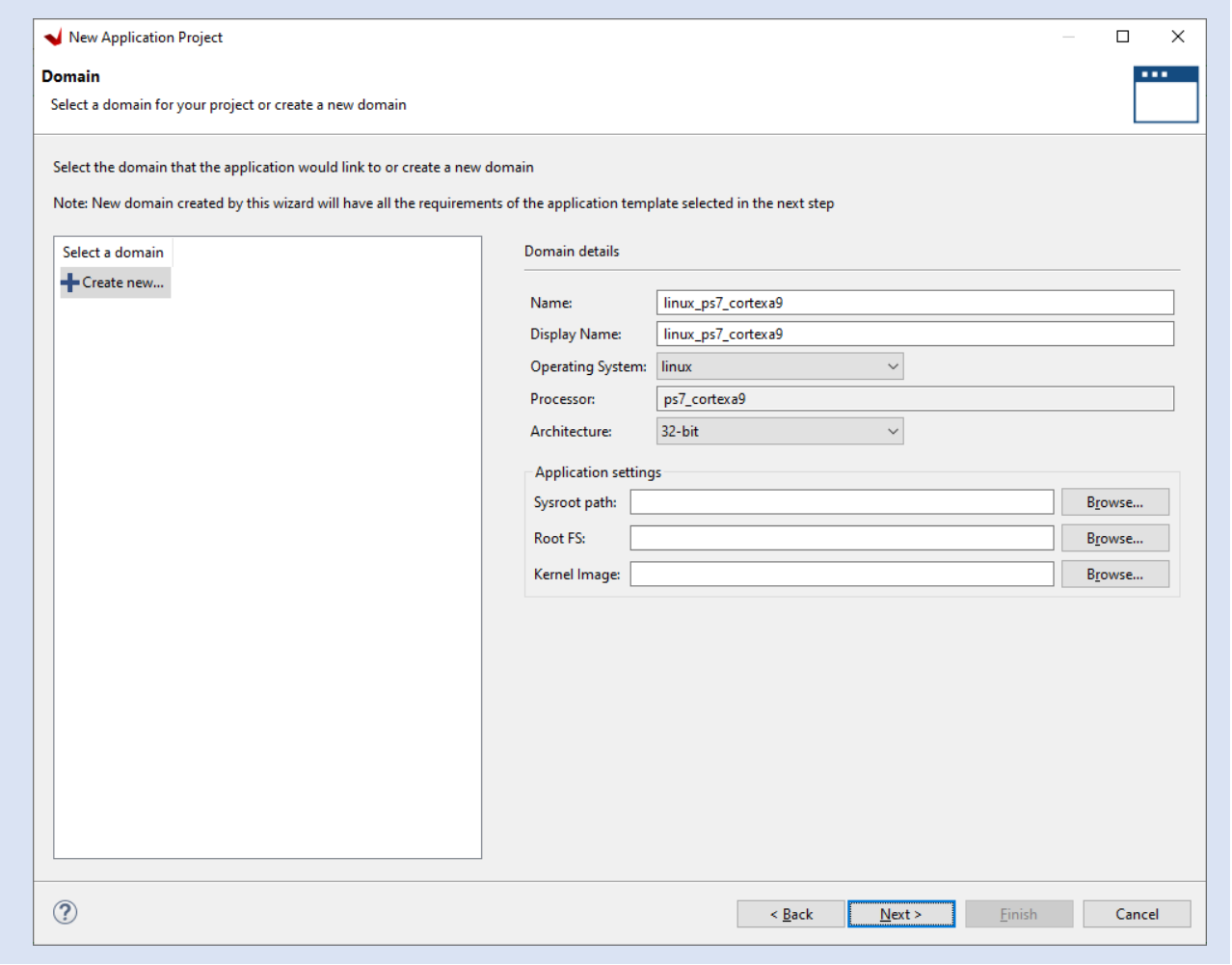
Lab 3: Creating Embedded Linux Solutions

Step 14 – Enter a name for the project and select the processors in SMP configuration



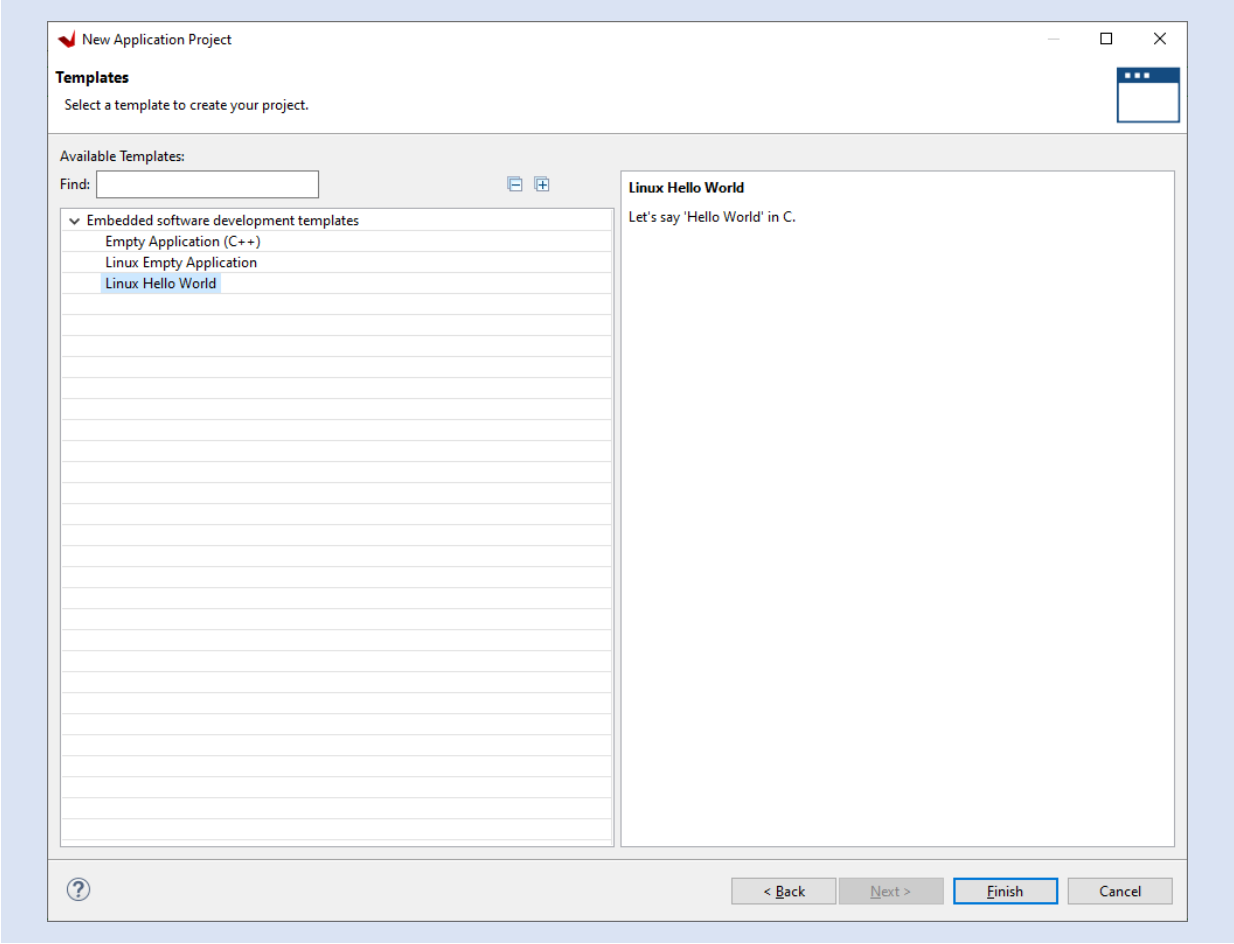
Lab 3: Creating Embedded Linux Solutions

Step 15 – Click Next on the Domain page



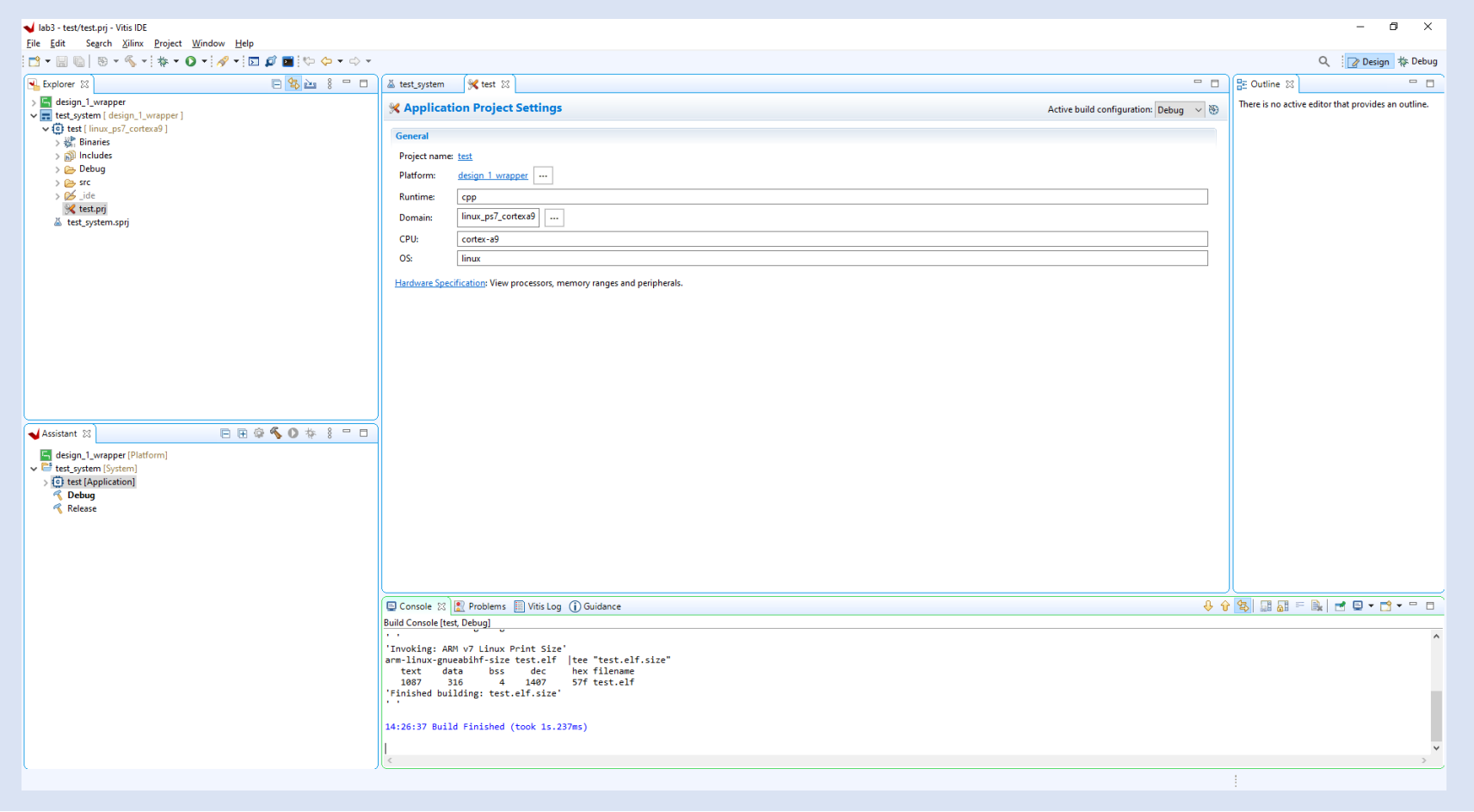
Lab 3: Creating Embedded Linux Solutions

Step 16 – Select the Linux Hello World application and click finish



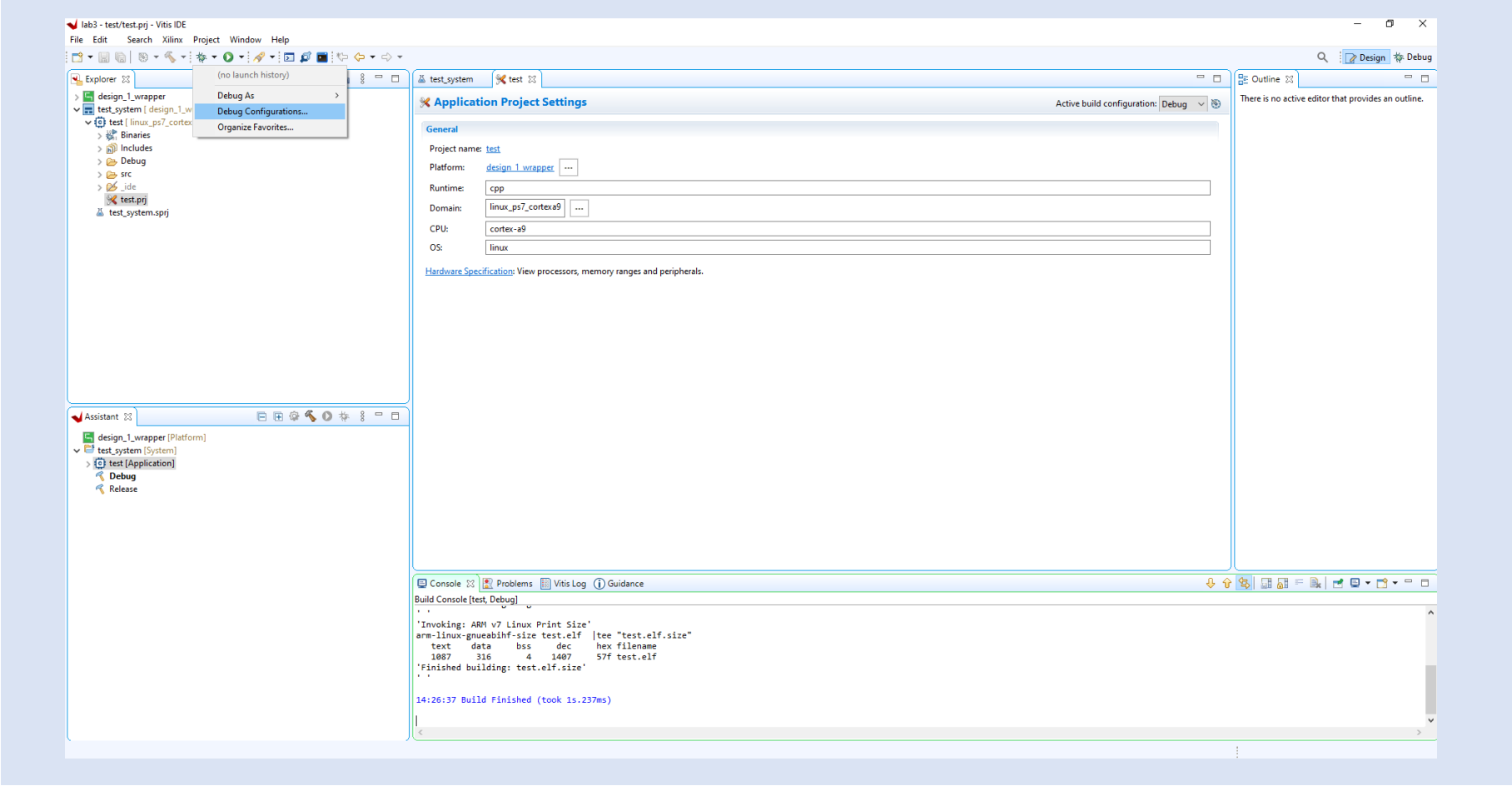
Lab 3: Creating Embedded Linux Solutions

Step 17 – Click on the Hammer to build the application



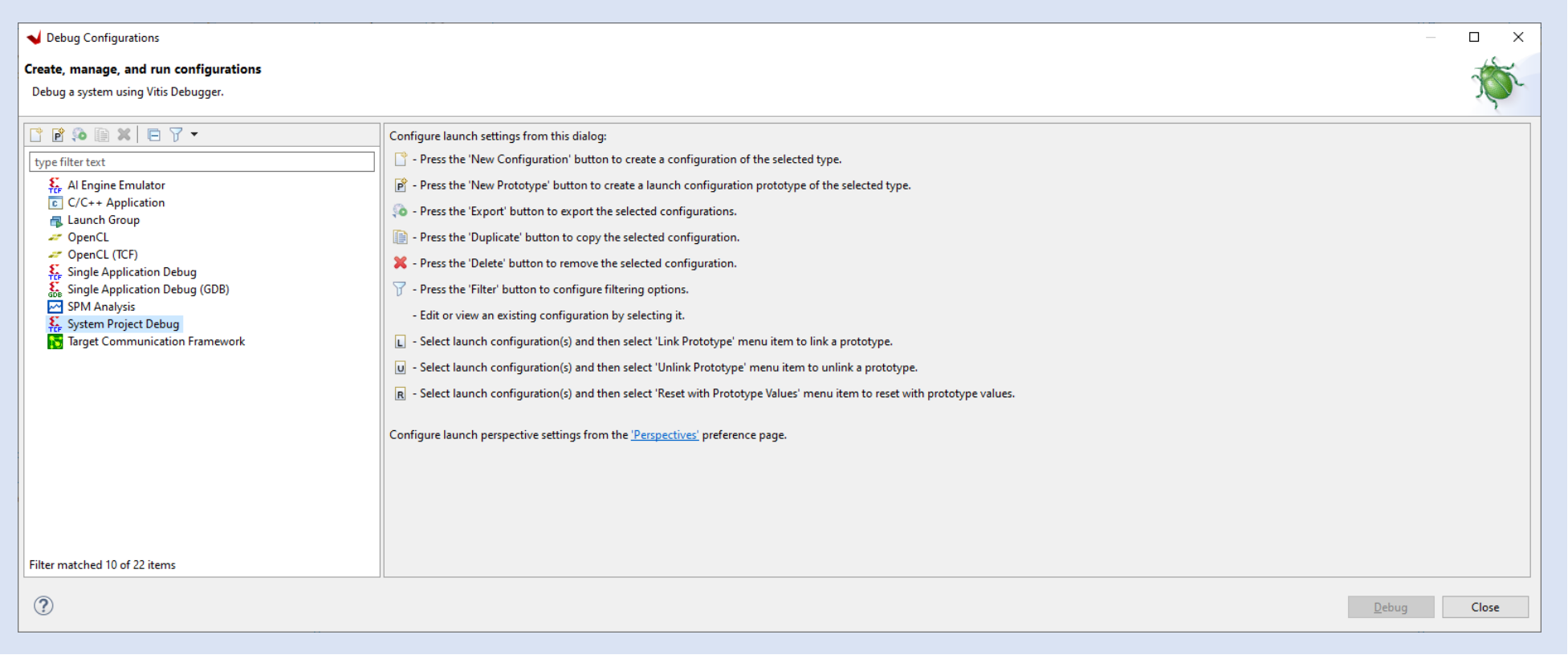
Lab 3: Creating Embedded Linux Solutions

Step 18 – Select the arrow by the green debug icon and select debug configuration



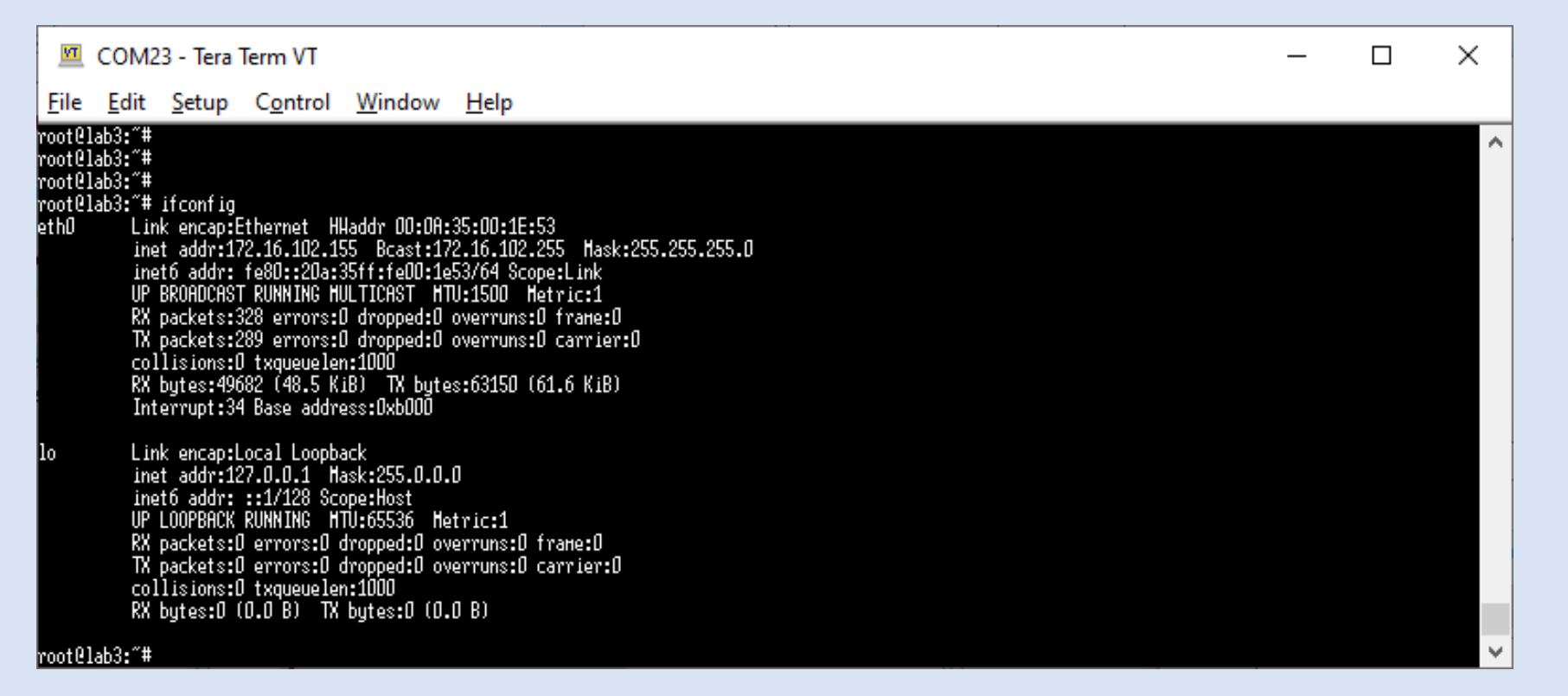
Lab 3: Creating Embedded Linux Solutions

Step 19 – Select System Project Debug and click new



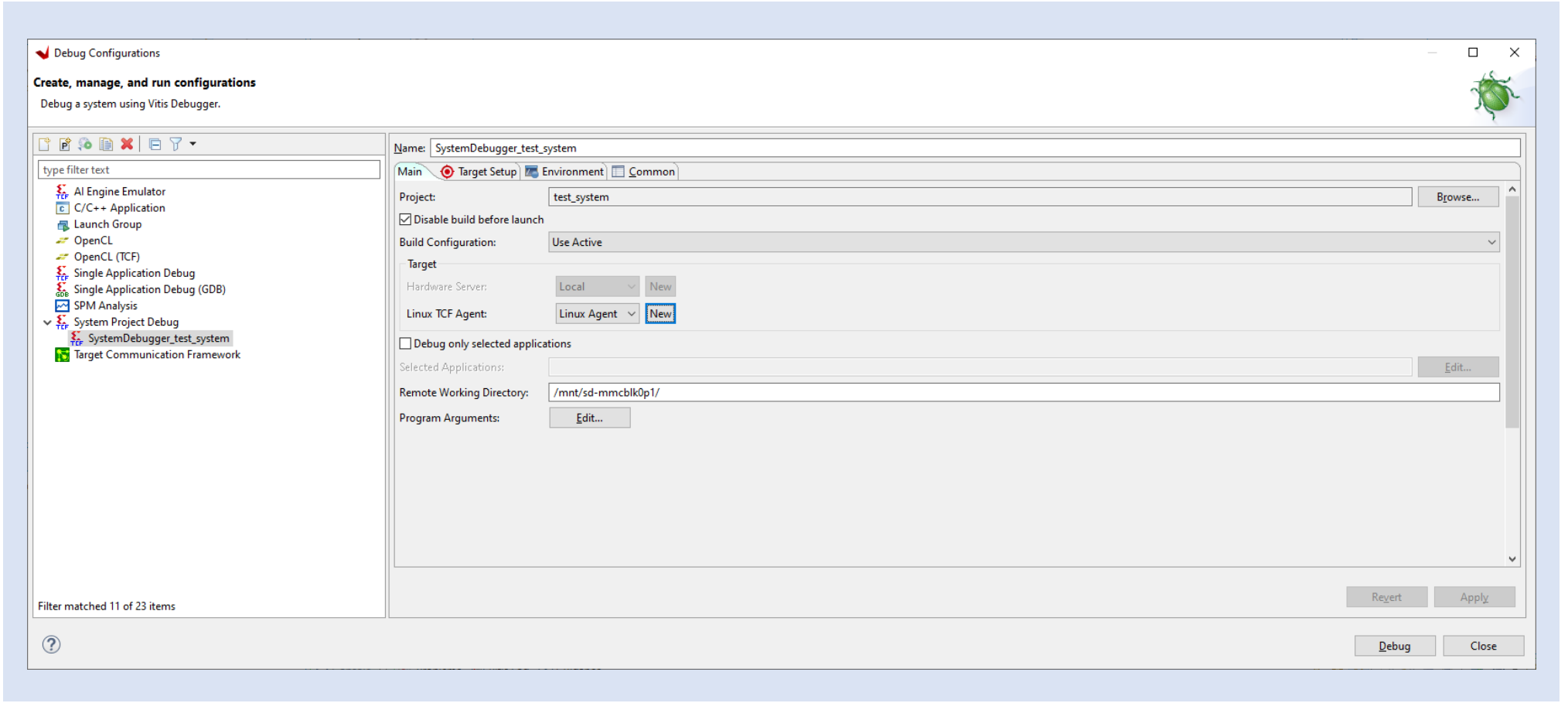
Lab 3: Creating Embedded Linux Solutions

Step 20 – Open a serial terminal 115200:n:1 and type the command ifconfig to list the IP Address of the board



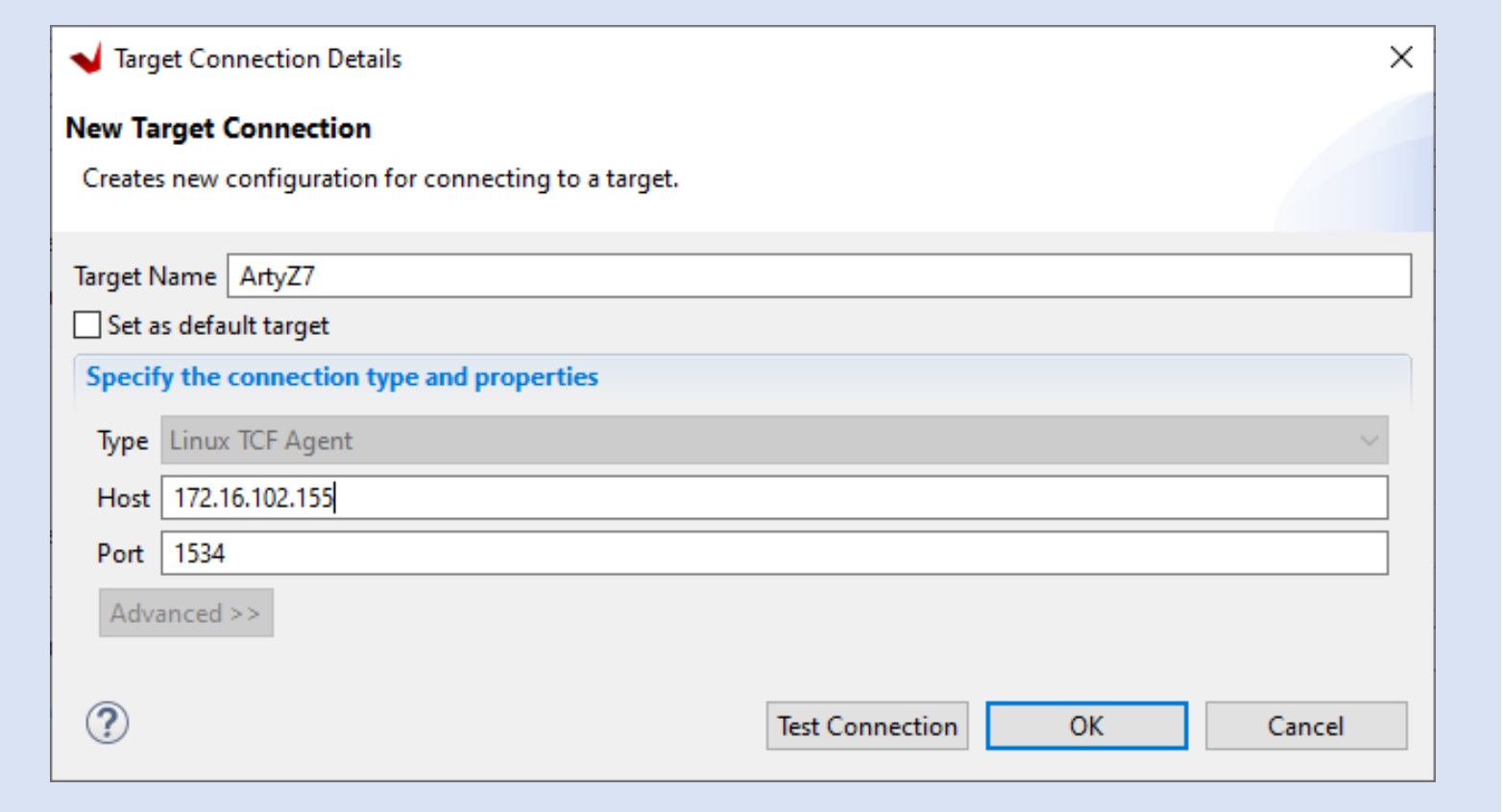
Lab 3: Creating Embedded Linux Solutions

Step 21 – Change the build configuration to Hardware and for Linux TCF Agent select new



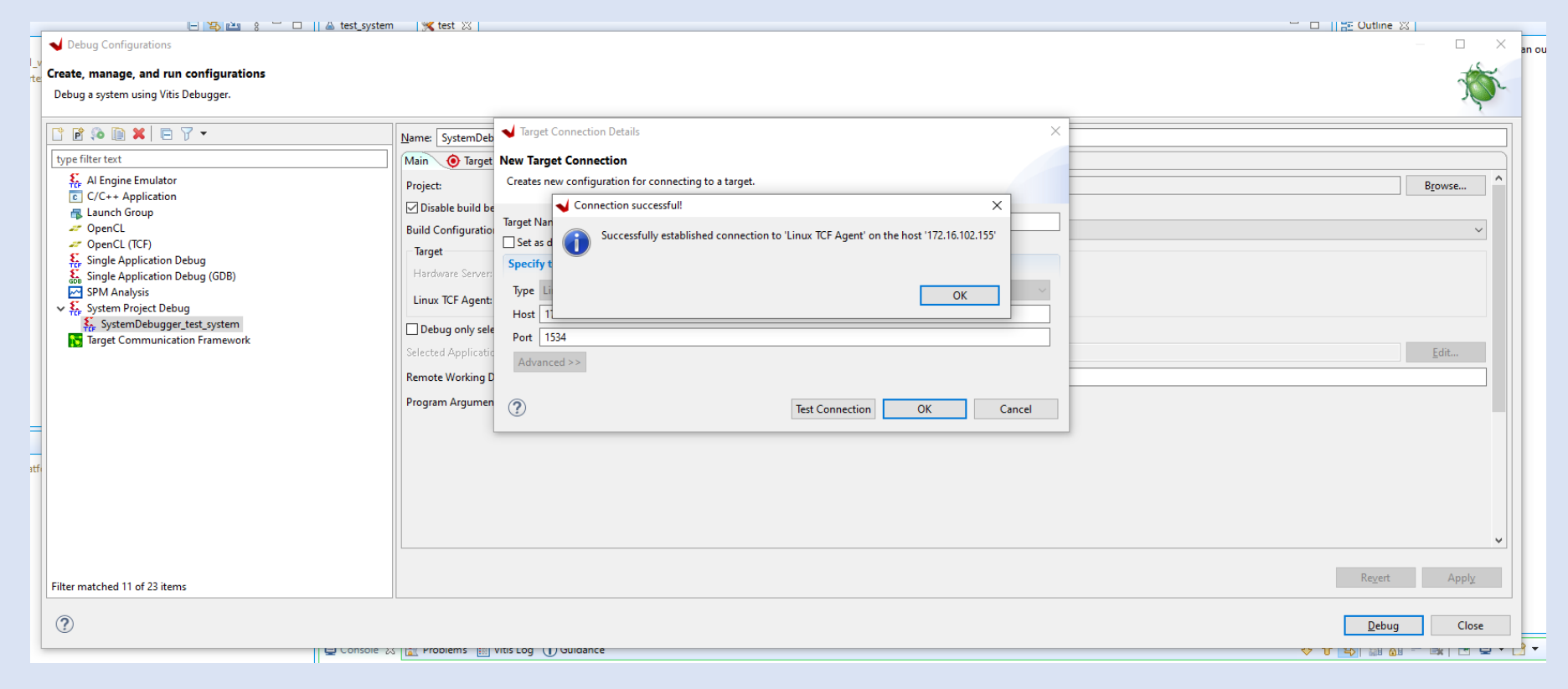
Lab 3: Creating Embedded Linux Solutions

Step 22 – Enter a name and enter the Ethernet IP address – Click test connection to verify the board can be seen.



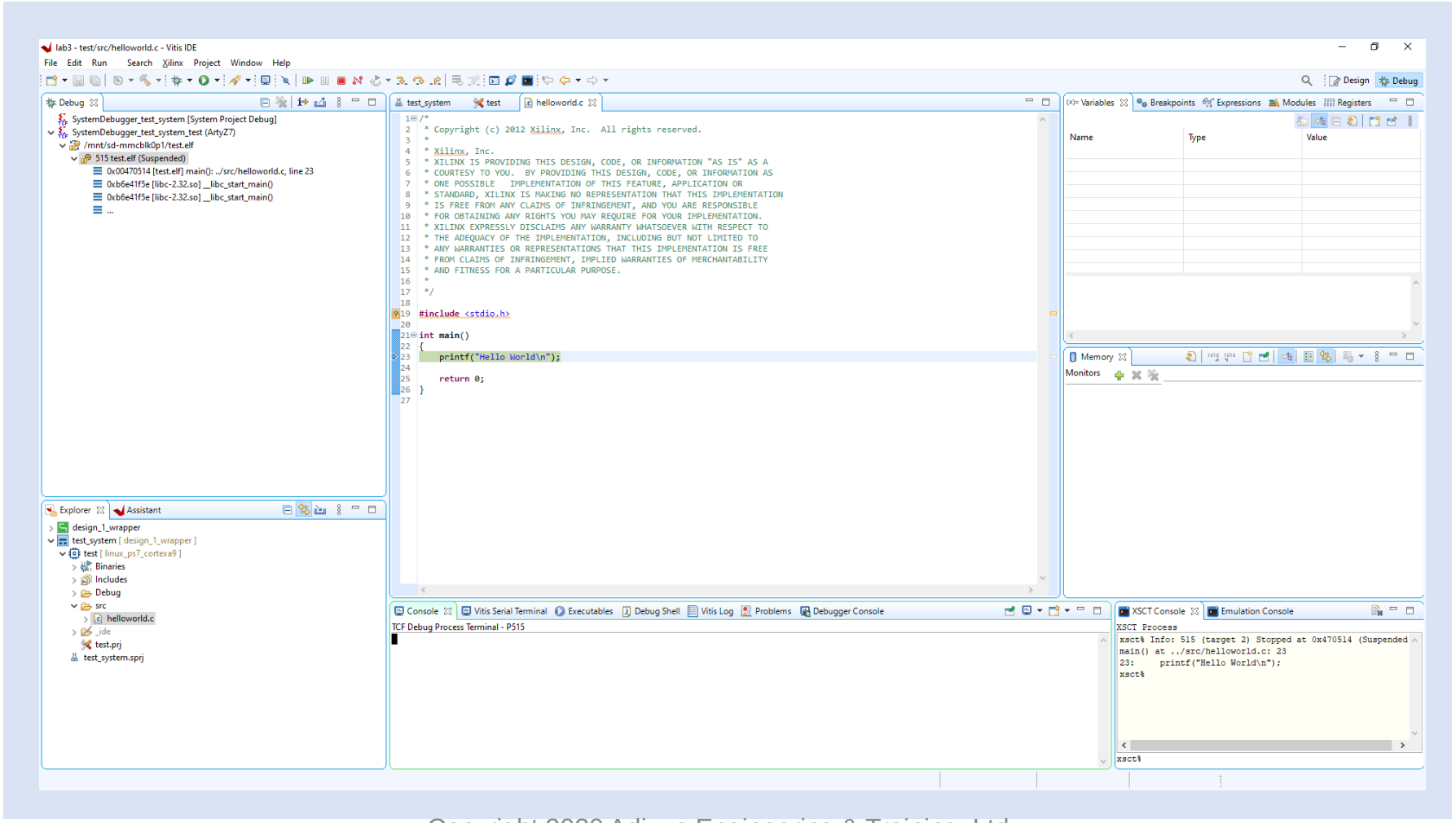
Lab 3: Creating Embedded Linux Solutions

Step 23 – Click OK when the connection is successful, click OK on the target connection and the click debug



Lab 3: Creating Embedded Linux Solutions

Step 24 – The application will be downloaded to the target and paused for execution, press the run button



Lab 3: Creating Embedded Linux Solutions

Step 25 – Click on the run button and you will see the hello world appear in the console.