

DESCRIPTION

To create an outdoor cat detection system to work as doorbell.

STARTING ALGORITHM

The inspiration for the algorithm comes from:

<https://pyimagesearch.com/2020/06/22/turning-any-cnn-image-classifier-into-an-object-detector-with-keras-tensorflow-and-opencv/>

And the CNN classifier has been trained as per:

https://keras.io/examples/vision/image_classification_from_scratch/

A good tutorial can be found also at:

<https://python.plainenglish.io/cat-dog-classification-with-cnn-84af3ae98c44>

JTAG DRIVERS INSTALLATION

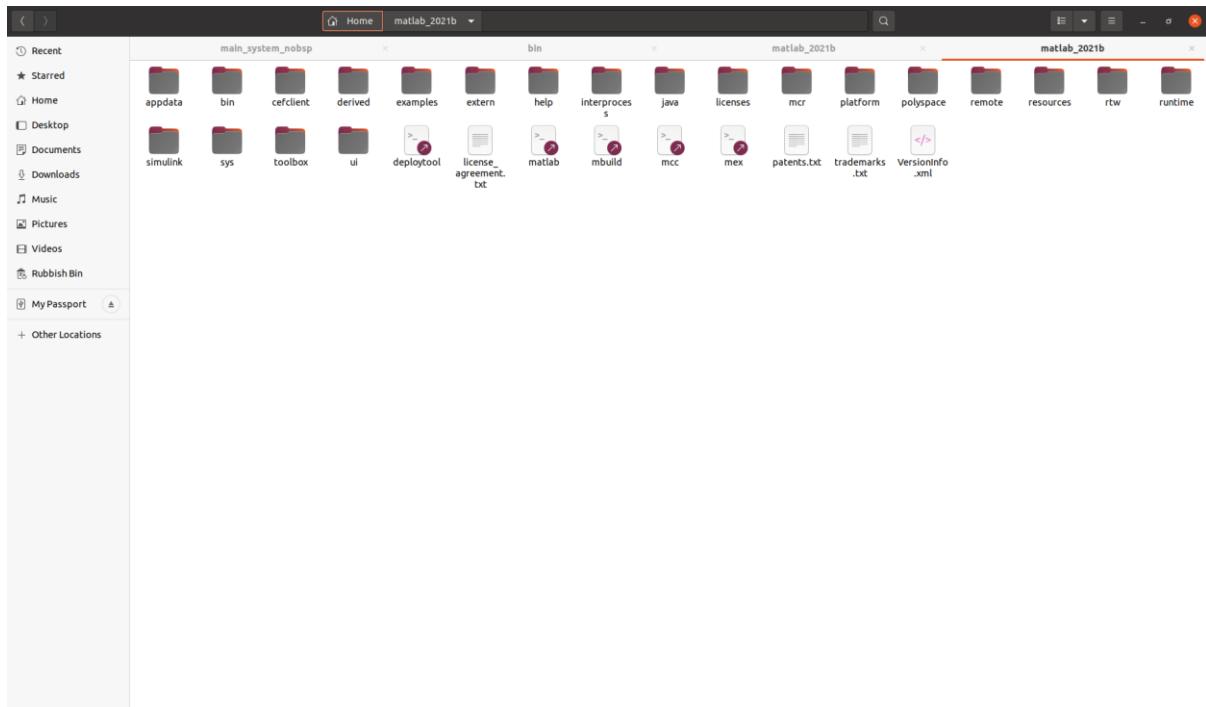
In order to allow Xilinx tools to recognize the board, cable drivers must be installed first.

To do so, go in the folder below and once there, launch the “install_drivers” script:

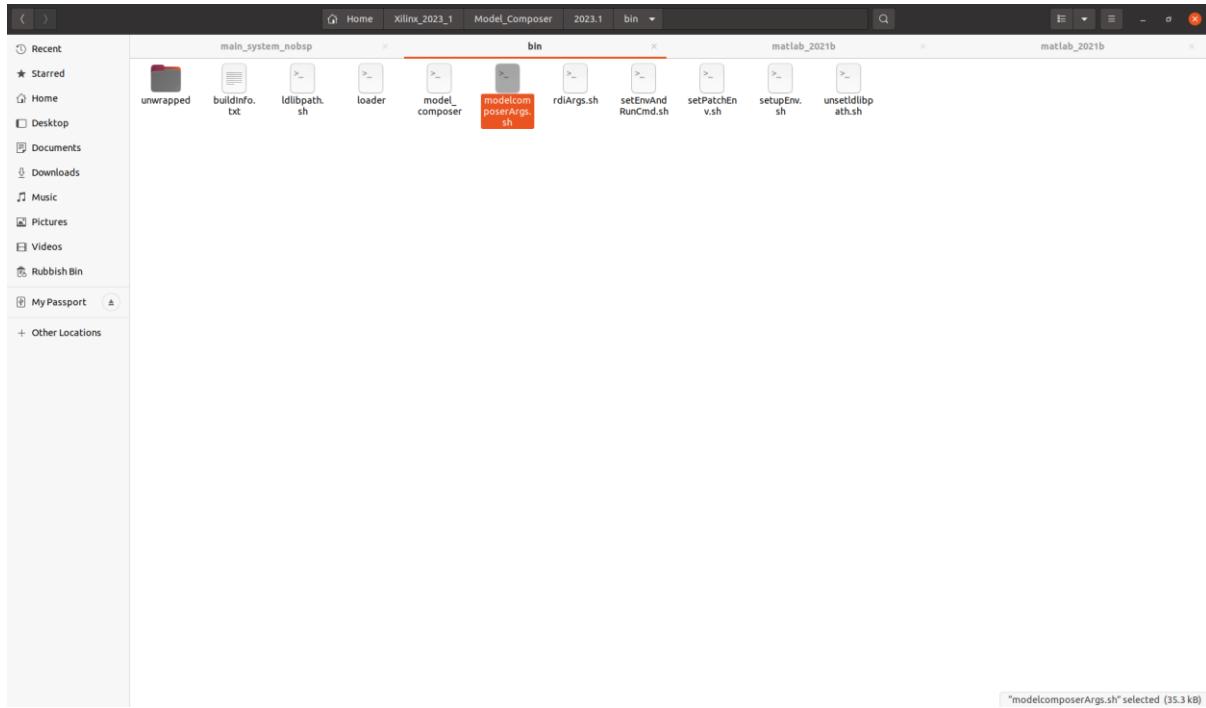


MODEL COMPOSER MATLAB/SIMULINK INSTALLATION

MATLAB 2021b has been installed in the position below:



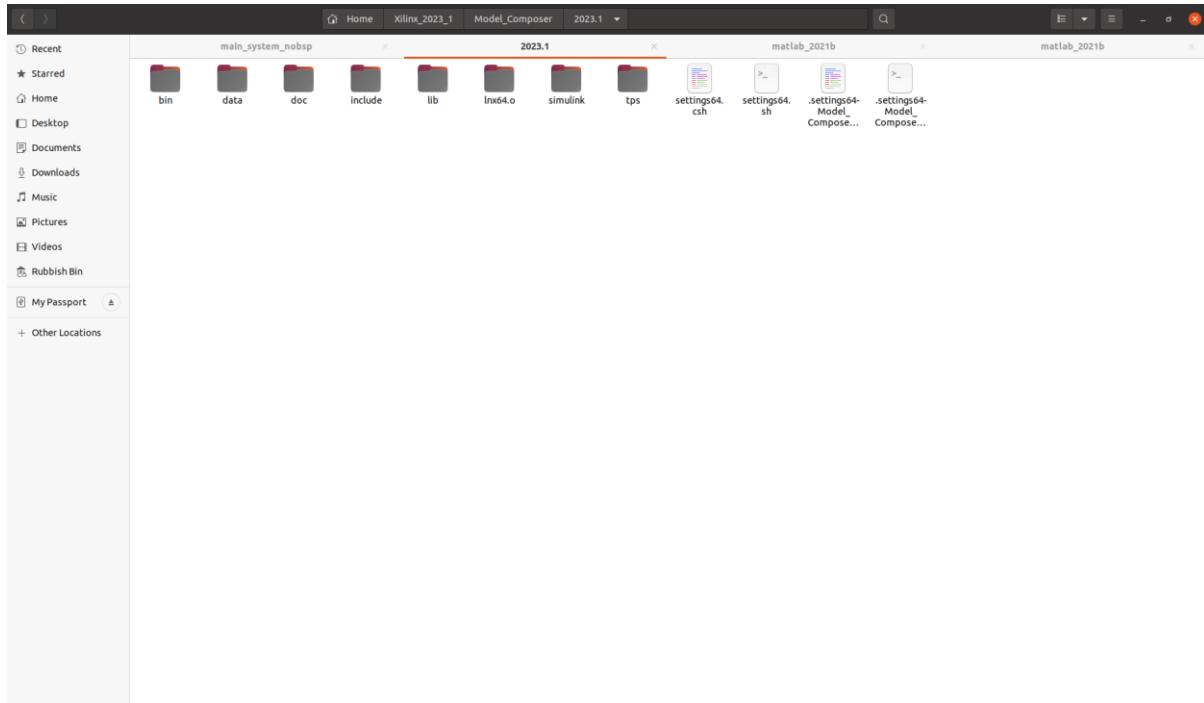
And model composer in the following one:



The matlab path has been added to the .bashrc file in the home folder:

```
Open Save .bashrc
65 unset color_prompt force_color_prompt
66 # If this is an xterm set the title to user@host:dir
67 case $TERM in
68 xterm*|rxvt*)
69   PS1="\[\e]0;${debian_chroot:+($debian_chroot)}\u@\h: \w\[\a]\$PS1"
70 ;;
71 *) ;;
72 esac
73 esac
74
75 # enable color support of ls and also add handy aliases
76 if [ -x /usr/bin/dircolors ]; then
77   test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"
78   alias ls='ls --color=auto'
79   alias dir='dir --color=auto'
80   alias vdir='vdir --color=auto'
81
82   alias grep='grep --color=auto'
83   alias fgrep='fgrep --color=auto'
84   alias egrep='egrep --color=auto'
85 fi
86
87 # colored GCC warnings and errors
88 #export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'
89
90 # come more ls aliases
91 alias ll='ls -l'
92 alias la='ls -A'
93 alias l='ls -C'
94
95 # Add an "alert" alias for long running commands. Use like so:
96 # sleep 10; alert
97 alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "%s(history|tail -n1|sed -e '\''${s/^/\$/g;s/[0-9]/\$/g}'')%s%alert%$'\''"
98
99 # Alias definitions.
100 # You may want to put all your additions into a separate file like
101 # .bash_aliases, instead of adding them here directly.
102 # See /usr/share/doc/bash-doc/examples in the bash-doc package.
103
104 if [ -f ~/.bash_aliases ]; then
105   . ~/.bash_aliases
106 fi
107
108 # enable programmable completion features (you don't need to enable
109 # this, it's already enabled in /etc/bash.bashrc and /etc/profile
110 # sources /etc/bash.bashrc).
111 if ! shopt -q posix; then
112   if [ -f /etc/bash_completion/bash_completion ]; then
113     . /usr/share/bash-completion/bash_completion
114   elif [ -f /etc/bash_completion ]; then
115     . /etc/bash_completion
116   fi
117 fi
118
119 source /home/caccolillo/petalinux/settings.sh
120 export MAILTO=/home/caccolillo/mailto_2021b
```

To start model composer, run the script “settings64.sh”:



and on the command line give “model_composer”:

```

caccoollo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Xilinx_2023_1/Model_Composer/2023.1$ model_composer
Launching Vitis Model Composer with System Generator functionality!
=====
Model Composer v2023.1
Build 3865809 on Sun May 7 15:11:57 MDT 2023
Copyright (C) 1986-2022 Xilinx, Inc. All Rights Reserved.
Copyright (C) 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
=====

***** ENVIRONMENT INFO:
Model Composer Bin Dir : /home/caccoollo/Xilinx_2023_1/Model_Composer/2023.1/bin
Vitis Bin Dir : /home/caccoollo/Xilinx_2023_1/Vitis/2023.1/bin
System Generator Bin Dir : /home/caccoollo/Xilinx_2023_1/Vlaldo/2023.1/bin
*****
Gtk-Message: 08:18:35.065: Failed to load module "canberra-gtk-module"
Inconsistency detected by ld.so: /lib/libcanberra-gtk-module.so: /lib/libcanberra-gtk-module.so: undefined symbol: _talloc
caccoollo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Xilinx_2023_1/Model_Composer/2023.1$ model_composer
Launching Vitis Model Composer with System Generator functionality!
=====
Model Composer v2023.1
Build 3865809 on Sun May 7 15:11:57 MDT 2023
Copyright (C) 1986-2022 Xilinx, Inc. All Rights Reserved.
Copyright (C) 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
=====

***** ENVIRONMENT INFO:
Model Composer Bin Dir : /home/caccoollo/Xilinx_2023_1/Model_Composer/2023.1/bin
Vitis Bin Dir : /home/caccoollo/Xilinx_2023_1/Vitis/2023.1/bin
System Generator Bin Dir : /home/caccoollo/Xilinx_2023_1/Vlaldo/2023.1/bin
*****
Gtk-Message: 08:18:35.949: Failed to load module "canberra-gtk-module"
Inconsistency detected by ld.so: /lib/libcanberra-gtk-module.so: /lib/libcanberra-gtk-module.so: undefined symbol: _talloc
caccoollo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Xilinx_2023_1/Model_Composer/2023.1$ model_composer
Launching Vitis Model Composer with System Generator functionality!
=====
Model Composer v2023.1
Build 3865809 on Sun May 7 15:11:57 MDT 2023
Copyright (C) 1986-2022 Xilinx, Inc. All Rights Reserved.
Copyright (C) 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
=====
```

To make Simulink work, the workaround below must be followed:

https://uk.mathworks.com/matlabcentral/answers/1454674-why-does-matlab-crash-on-linux-with-inconsistency-detected-by-ld-so-elf-dl-tls-c-597- dl_allo

Solution:

1. Install the gtk-2.0+ version of canberra by running the following command in the terminal:

sudo apt-get install libcanberra-gtk-module -y

2. Add a soft connection, run the following command in the terminal (many people are missing this step):

**sudo ln -s /usr/lib/x86_64-linux-gnu/gtk-2.0/modules/libcanberra-gtk-module.so
/usr/lib/libcanberra-gtk-module.so**

SSH and SCP to transfer files to the board

To transfer files from the host PC to the target, configure the static IP addresses of the UBUNTU PC and the Ultra 96 V2:

GTKTerm - /dev/ttyUSB1 115200-8-N-1

File Edit Log Configuration Controlsignals View Help

```
#1) Respect the privacy of others.  
#2) Think before you type.  
#3) With great power comes great responsibility.
```

Password:

```
ultra96:~$ ifconfig  
eth0      Link encap:Ethernet HWaddr 7C:C2:C6:41:8D:D0  
          inet addr:192.168.1.11 Bcast:192.168.1.255 Mask:255.255.255.0  
          inet6 addr: fe80::7ec2:c6ff:fe41:8dd0/64 Scope:Link  
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
            RX packets:15 errors:15 dropped:0 overruns:0 frame:0  
            TX packets:26 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1000  
            RX bytes:2139 (2.0 KiB) TX bytes:3694 (3.6 KiB)  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
            UP LOOPBACK RUNNING MTU:65536 Metric:1  
            RX packets:2 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:2 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1000  
            RX bytes:140 (140.0 B) TX bytes:140 (140.0 B)  
  
ultra96:~$ asdass
```

/dev/ttyUSB1 115200-8-N-1

DTR RTS CTS CD DSR RI

```

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~      caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~
TX packets 9262 bytes 1372613 (1.3 MB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ sudo ifconfig enp5s0 192.168.1.10
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ifconfig
enp5s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.10 netmask 255.255.255.0 broadcast 192.168.1.255
        ether 6c:02:e0:45:16:fb txqueuelen 1000 (Ethernet)
        RX packets 51 bytes 9653 (9.6 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 153 bytes 21816 (21.8 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 2165 bytes 237621 (237.6 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 2165 bytes 237621 (237.6 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp4s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.149.17 netmask 255.255.255.0 broadcast 192.168.149.255
    inet6 fe80::2c71:d382:e43b:7316 prefixlen 64 scopeid 0x20<link>
        ether a8:93:4a:5f:50:dd txqueuelen 1000 (Ethernet)
        RX packets 10285 bytes 9407409 (9.4 MB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 9298 bytes 1377289 (1.3 MB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ssh petalinux@192.168.1.11
petalinux@192.168.1.11's password:
ultra96:~$ ls
ultra96:~$ ls
ultra96:~$ scp
usage: scp [-1246BCpqrv] [-c cipher] [-F ssh_config] [-i identity_file]
           [-l limit] [-P port] [-S program]
           [[user@]host1:]file1 [...] [[user@]host2:]file2
ultra96:~$ 

```

On the linux pc connect to the Ultra96 v2 with “[ssh petalinux@192.168.1.11](ssh://petalinux@192.168.1.11)”.

To transfer a file use the following syntax:

```

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~      caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~      caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~
PetaLinux environment set to '/home/caccoollo/petalinux'
INFO: Checking free disk space
INFO: Checking installed tools
INFO: Checking installed development libraries
INFO: Checking network and other services
WARNING: No tftp server found - please refer to "UG1144 2022.2 PetaLinux Tools Documentation Reference Guide"
         for its impact and solution
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ls
Desktop          petalinux_prj    vivado_120092.backup.jou  vivado_3475.backup.log  Xilinx_2023_1
Documents        petalinux_shared  vivado_120092.backup.log  vivado.jou       xilinx2023installer
Downloads        Pictures       vivado_217118.backup.jou vivado.log      xrc.log
github_token.txt Public        vivado_217118.backup.log vivado_pid120316.str
matlab_2021b     snap          vivado_3434.backup.jou  vivado_pid3434.str
Music            Templates      vivado_3434.backup.log   vivado_pid3475.str
petalinux        Videos        vivado_3475.backup.jou  Xilinx_2022_2
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ scp ./vivado.jou petalinux@192.168.1.11:/home/petalinux
petalinux@192.168.1.11's password:
vivado.jou                                         100%  886    825.7KB/s  00:00
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ 

```

When it complains about the SSH key, use the trick below:

```
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/python$ scp ./get4frames.py petalinux@192.168.1.12:/home/petalinux/
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/python$ scp ./get4frames.py petalinux@192.168.1.12:/home/petalinux/
@ WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!
@ IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!

It is also possible that a host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
SHA256:+v9vD5jibGa7SiPeno1kUHGDBH8APKbvniCCx+0M.
Please contact your system administrator.
Add correct host key in /home/accolillo/.ssh/known_hosts to get rid of this message.
Offending RSA key in /home/accolillo/.ssh/known_hosts:
remove with:
ssh-keygen -f "/home/accolillo/.ssh/known_hosts" -R "192.168.1.12"
RSA host key for 192.168.1.12 has changed and you have requested strict checking.
Host key verification failed.
lost connection
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/python$ ssh-keygen -R 192.168.1.12
# Host 192.168.1.12 found: line 2
/home/accolillo/.ssh/known_hosts updated.
Original contents retained as /home/accolillo/.ssh/known_hosts.old
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/python$ scp ./get4frames.py petalinux@192.168.1.12:/home/petalinux/
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/python$
```

To copy everything from the remote to a local directory, use:

```
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/tmp$ scp petalinux@192.168.1.12:/home/petalinux/*.* ./
petalinux@192.168.1.12's password:
driver_lmg_avg_prj.elf
frame1.png
frame1.txt
frame2.png
frame2.txt
frame3.png
frame3.txt
frame4.png
frame4.txt
get4frames.py
outputframe.txt
100%   80KB  16.2MB/s  00:00
100%  839KB  21.5MB/s  00:00
100%  12MB  22.8MB/s  00:00
100%  867KB  21.9MB/s  00:00
100%  12MB  22.8MB/s  00:00
100%  874KB  22.1MB/s  00:00
100%  12MB  22.8MB/s  00:00
100%  103KB  22.3MB/s  00:00
100%  13MB  22.8MB/s  00:00
100%  2013   2.8MB/s  00:00
100%   10MB  22.8MB/s  00:00
caccoollo@accolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/tmp$
```

THE BOARD

The algorithm has to be ported on the ULTRA96 board, using a COTS webcam as image acquisition device:

<https://community.element14.com/products/devtools/avnetboardscommunity/w/boards/23079/ultra96-v2>

The training material is at:

<https://community.element14.com/products/devtools/avnetboardscommunity/avnetboard-training/technical-training-courses/w/documents/27905/ultra96-v2-hardware-technical-training-2021-2-lesson-1?ICID=ultra96-2021-hardwaretraining>

THE OUTLINE

Use:

- Linux
- Image acquisition
- PYNQ
- OpenAMP
- Python on Zynq
- HLS for image processing acceleration
- Adding HLS devices to the device tree
- System Verilog testbench
- Verification plan
- Vivado
- Vitis
- TCL
- Bash

- C++/C

BOARD VIVADO INSTALLATION

To install the board under Vivado, follow the directions at:

<https://github.com/Avnet/bdf>

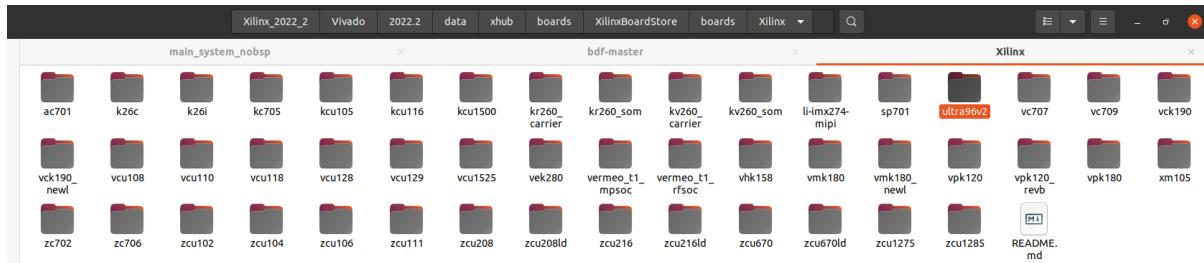
Download the repo as a zip file, then follow:

https://support.xilinx.com/s/article/The-board-file-location-with-the-latest-Vivado-tools?language=en_US

And:

[Xilinx Vivado Installation and Ultra96 V2 BDF setup \(Windows OS\)](#)

In this way, the board will be supported by Vivado:



CREATING A BOOTABLE IMAGE ON A MICRO-SD

Go in the petalinux image folder and create the BOOT.BIN boot file as per instructions below:

-
10. Change directory into **images/linux** within the lab1 project.

```
cd images/linux
```

11. Once in the project enter:

```
$ petalinux-package --boot --fsbl zynqmp_fsbl.elf --u-boot u-boot.elf --pmufw pmufw.elf --fpga system.bit
[INFO] Sourcing buildtools
[INFO] Getting system flash information...
[INFO] File In BOOT BIN: "/home/training/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux/zynqmp_fsbl.elf"
[INFO] File In BOOT BIN: "/home/training/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux/pmufw.elf"
[INFO] File In BOOT BIN: "/home/training/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux/system.bit"
[INFO] File In BOOT BIN: "/home/training/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux/bl31.elf"
[INFO] File In BOOT BIN: "/home/training/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux/system.elf"
[INFO] Generating zynqmp binary package BOOT.BIN...
***** Xilinx Bootgen v2021.2
**** Build date : Sep 30 2021-06:08:18
** 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!!!
[INFO] Creating/Training-VirtualBox:/AvnetTTC-u96/MPSoC_PetalLinux/2021.2/Labi/Images/Linux$
```

Figure 6 Generate **BOOT.BIN**

Create a bootable micro-SD card as per instructions in the following:

14. Now that the SD card is in the Virtual machine, we can use fdisk and disks to create and format our partitions. Open disks:



Figure 8 Open Disk Application

15. In disks we can see our SD card:

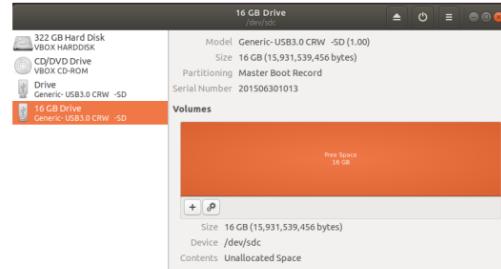


Figure 9 SD card in Disk Application

Note if you already have a partition on the drive, we recommend you delete the partitions before continuing. Simply press the “minus” button (only shows if you have partitions) on the selected partition to delete them. Continue if you have “Free Space”

16. We are going to create the first Partition as FAT32. Simply press the plus button. We recommend that you have at least 1 GB for the boot files and leave the rest for the Linux OS.

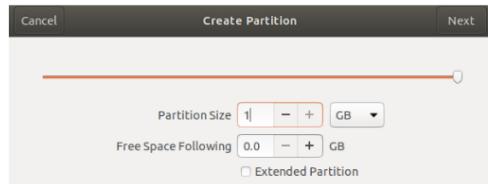


Figure 10 Create new partition 1GB

17. Next add a name for the boot partition. Leave the settings as default and press Create when done.

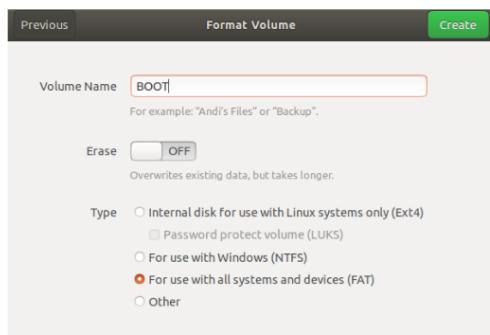


Figure 11 Name the new partition and FAT32 type

18. Now we create the second partition in the same way. Click on the free space and then the plus button.



Figure 12 Create second partition with remaining free space

19. Name the Linux File System for the partition and **select the file system type as Ext4** click create once done.

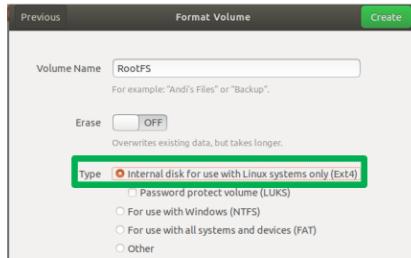


Figure 13 Name the Linux File System and EXT4

20. Open a terminal and enter:

```
sudo fdisk -l
```

This command will show all the drives on the virtual machine. Look for the SD card as we need the device name.

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sdc1		2048	1955839	1953792	954M	c	W95 FAT32 (LBA)
/dev/sdc2		1955840	31115263	29159424	13.9G	83	Linux

Figure 14 Check in fdisk

21. Navigate into the images/Linux folder of our PetaLinux project. In our case

```
cd ~/AvnetTTC/home/training/AvnetTTC-u96/MPSoC_PetaLinux/2021.2/Lab1
```

21. Navigate into the images/Linux folder of our PetaLinux project. In our case

```
cd ~/AvnetTTC/home/training/AvnetTTC-u96/MPSoC_PetaLinux/2021.2/Lab1
```

22. Next is to enter the command below. Make sure the correct partition is selected. Our ext4 partition is mounted to /dev/sdc2. This took just under an hour to complete

```
sudo dd if=images/linux/rootfs.ext4 of=/dev/sdc2 status=progress
```

```
transferring-VirtualBox:~/AvnetTTC-u96/MPSoC_PetaLinux/2021.2/Lab1$ sudo dd if=images/linux/rootfs.ext4 of=/dev/sdc2 status=progress
[sudo] password for training:
Sorry, try again.
[sudo] password for training:
4577792+0 records in
4577792+0 records out
2343449088 bytes (2.3 GB, 2.2 GiB) copied, 3534 s, 663 kB/s
```

Figure 15 Flash Completed

While we wait for the flashing process to complete, we are going to copy the boot files into the boot partition that we created earlier. The boot files that we need to copy over is the boot.scr, BOOT.BIN and image.ub.

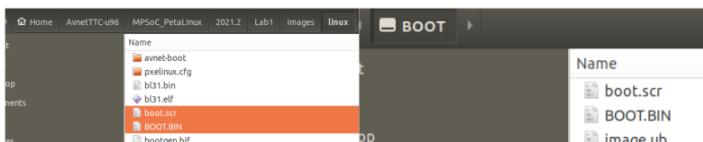


Figure 16 Copy the boot files in the BOOT partition

Insert the micro-SD card and boot as follows:

1. Power down the Ultra96 board by pressing and holding the power down, then change the Boot Mode switch on Ultra96 to 01 for SD Card as shown below.

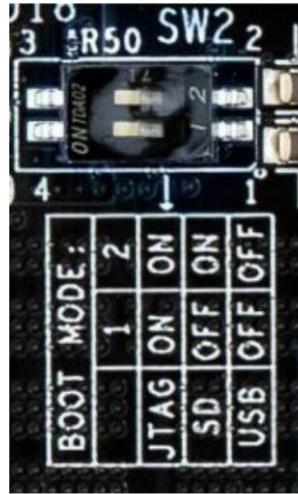


Figure 17 Set switch to SD

2. Plug-in the USB Cable to the USB to JTAG/UART pod
3. Open Terra Term and connect to the serial port at 115200, No Parity and 1 Stop
4. Power on the Ultra96 by pressing the power on button. You should see the u-boot and PetaLinux boot messages scroll in the terminal window.

The port to use for the serial terminal is /dev/ttyUSB1.

PETALINUX

The petalinux distribution gets built as per the tutorial:

<https://www.hackster.io/LogicTronix/petalinux-2020-2-bsp-for-ultra96v2-with-wifi-vnc-server-4764ea>

The board support package file, can be gathered from Avnet website:

<https://www.avnet.com/wps/portal/us/products/avnet-boards/avnet-board-families/ultra96-v2/>

Also the following guide can be followed:

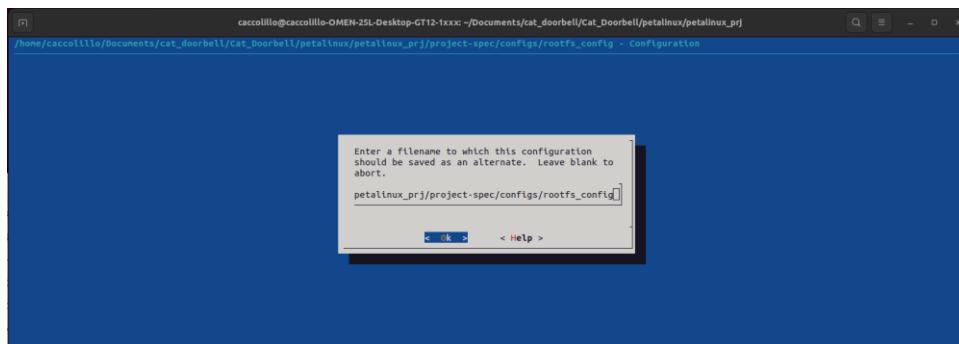
<https://www.96boards.org/documentation/consumer/ultra96/ultra96-v2/build/petalinux.md.html#install-petalinux-sdk>

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18842385/How+to+format+SD+card+for+SD+boot>

Some guidance about how to enable Python and deep learning packages, is available at:

https://support.xilinx.com/s/article/1171977?language=en_US

To know where the configuration files for the kernel and the root filesystem are, it will suffice to have a look at the save menu:



An Ubuntu distribution is available at:

<https://github.com/ikwzm/ZynqMP-FPGA-Ubuntu20.04>

To generate the device tree when PL is involved, follow the guide:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/341082130/Quick+guide+to+Debugging+Device+Tree+Generator+Issues>

Petalinux user manual is available at:

<https://docs.xilinx.com/r/2022.1-English/ug1144-petalinux-tools-reference-guide/Installation-Requirements>

And specifies which is the supported OS and Vivado release as well.

Petalinux 2022.2 supports Ubuntu 20.04 LTS and Vivado 2022.2.

To get it working, has been necessary install Ubuntu 20.04 LTS in virtual box.

In order to allow the board to get connected to the wifi network, the following file has been modified to include the network name and the wifi password:

<https://logictronix.com/machine-learning-with-fpga/vitisai-dpu-dnndk/deephi-dnndk-tutorial-for-ultra96/>

In order to get the Wifi and all the rest working, login as:

USER = root

PASSWORD = root

```

fonts          lsb-release      pulse        skel        wpa_supplicant.conf.old
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
update_config=1

network={
    ssid="VodafoneMobileWiFi-6B1862"
    psk="3122407191"
}

-
-
-
-
-
" ./wpa_supplicant.conf" [dos] 19L, 187B written
u96v2-sbc-base-2022-2:~# cat ./wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
update_config=1

network={
    ssid="VodafoneMobileWiFi-6B1862"
    psk="3122407191"
}

u96v2-sbc-base-2022-2:~# ls
bt.sh pmic-prog usb-gadget ethernet.sh wifi.sh wpa_supplicant.conf wpa_supplicant.conf~
/dev/ttyUSB1 115200-8-N-1                                         DTR RTS CTS CD DSR RI

```

And running the script wifi.sh in the root folder, the internet connection is achieved:

```

u96v2-sbc-base-2022-2:~# ls
bt.sh pmic-prog usb_gadget_ethernet.sh wifi.sh wpa_supplicant.conf wpa_supplicant.conf~
u96v2-sbc-base-2022-2:~# ./wifi.sh
Stopping Connection Manager
ifconfig: SIOCGIFFLAGS: No such device
Successfully initialized wpa_supplicant
nl80211: kernel reports: Authentication algorithm number required[ 1089.924856] power up request for already
powered up source Wifi

udhcpc: started, v1.34.1
udhcpc: broadcasting discover
udhcpc: broadcasting discover
udhcpc: broadcasting select for 192.168.217.175, server 192.168.217.154
udhcpc: lease of 192.168.217.175 obtained from 192.168.217.154, lease time 3599
/etc/udhcpc.d/50default: Adding DNS 192.168.217.154
u96v2-sbc-base-2022-2:~# ping www.google.com
PING www.google.com (142.250.178.4): 56 data bytes
64 bytes from 142.250.178.4: seq=0 ttl=114 time=44.521 ms
64 bytes from 142.250.178.4: seq=1 ttl=114 time=43.148 ms
64 bytes from 142.250.178.4: seq=2 ttl=114 time=48.147 ms
64 bytes from 142.250.178.4: seq=3 ttl=114 time=47.646 ms
64 bytes from 142.250.178.4: seq=4 ttl=114 time=43.286 ms
64 bytes from 142.250.178.4: seq=5 ttl=114 time=44.862 ms
^C
--- www.google.com ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 43.148/45.268/48.147 ms
u96v2-sbc-base-2022-2:~#     dnf install python3-pip
[[T]AOE Remote Repo: sswareleases sswboa      [==          ] --- B/s | 0   B   --:--
OE Remote Repo: sswareleases sswboardfeeds zsoc all           45 KB/s | 92  kB   00:02
OE Remote Repo: sswareleases sswboardfeeds zsoc aarch64       753 KB/s | 3.7 MB   00:05
OE Remote Repo: sswareleases sswboardfeeds zsoc plnx_aarch64   19 KB/s | 31  kB   00:01
Package python3-pip-21.2.4-r0.0.cortexa72_cortexa53 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
/dev/ttyUSB1 115200-8-N-1                                         DTR RTS CTS CD DSR RI

```

Now packages can be installed from the net with the dnf command:

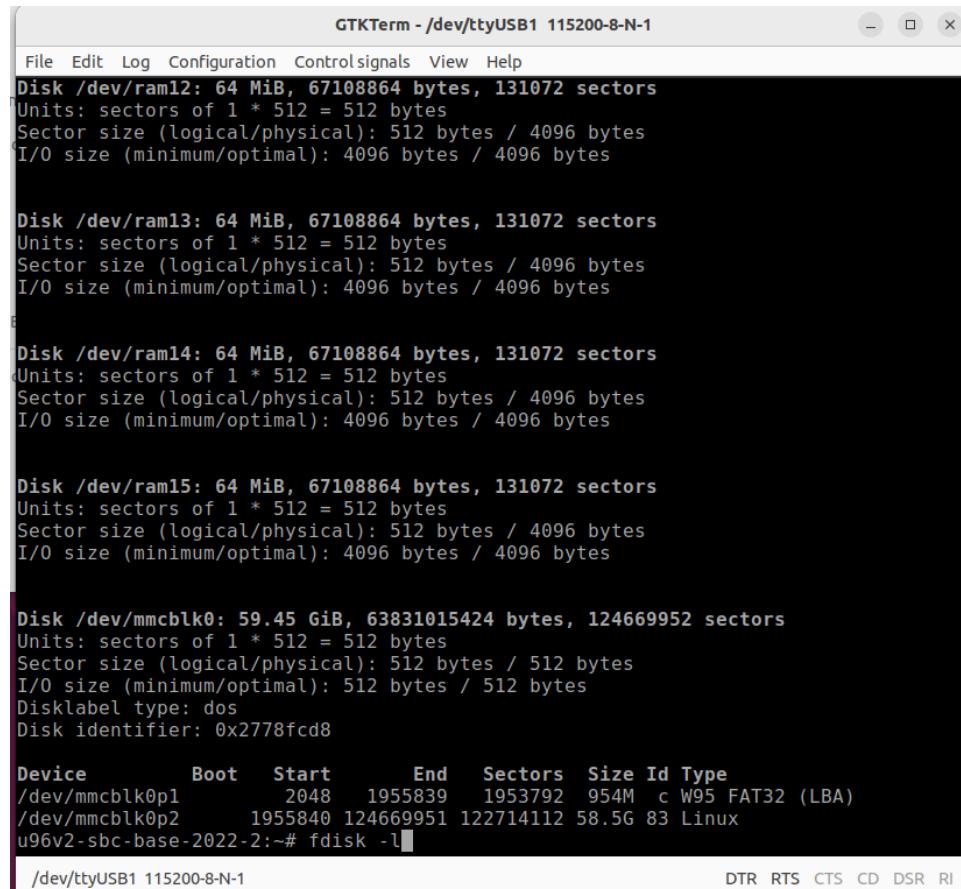
<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18841870/Install+and+run+applications+through+DNF+on+target?showComments=true&showCommentArea=true>

Because all gets installed in the ramdisk, and this is not big enough, issues about insufficient space gets raised.

To circumvent this, when using pip3 install, it must be instructed to use another location, such as an USB memory drive.

So the USB memory stick gets mounted, as per the following instructions:

<https://linuxconfig.org/howto-mount-usb-drive-in-linux>



The screenshot shows a terminal window titled "GTKTerm - /dev/ttyUSB1 115200-8-N-1". The terminal displays the following text:

```
Disk /dev/ram12: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram13: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram14: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram15: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/mmcblk0: 59.45 GiB, 63831015424 bytes, 124669952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x2778fcdb

Device      Boot   Start     End   Sectors  Size Id Type
/dev/mmcblk0p1        2048 1955839  1953792  954M  c W95 FAT32 (LBA)
/dev/mmcblk0p2    1955840 124669951 122714112 58.5G  83 Linux
u96v2-sbc-base-2022-2:~# fdisk -l

/dev/ttyUSB1 115200-8-N-1
```

Disk /dev/sda: 29.88 GiB, 32078036992 bytes, 62652416 sectors

Disk model: Flash Drive

Units: sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: dos

Disk identifier: 0x00000000

u96v2-sbc-base-2022-2:~# mount /dev/sda /media/usb-drive/

u96v2-sbc-base-2022-2:~#

/dev/ttyUSB1 115200-8-N-1

And by following these links:

<https://stackoverflow.com/questions/40755610/ioerror-errno-28-no-space-left-on-device-while-installing-tensorflow>

<https://askubuntu.com/questions/1326304/cannot-install-pip-module-because-there-is-no-space-left-on-device>

<https://superuser.com/questions/1652496/pip-install-gives-no-free-space>

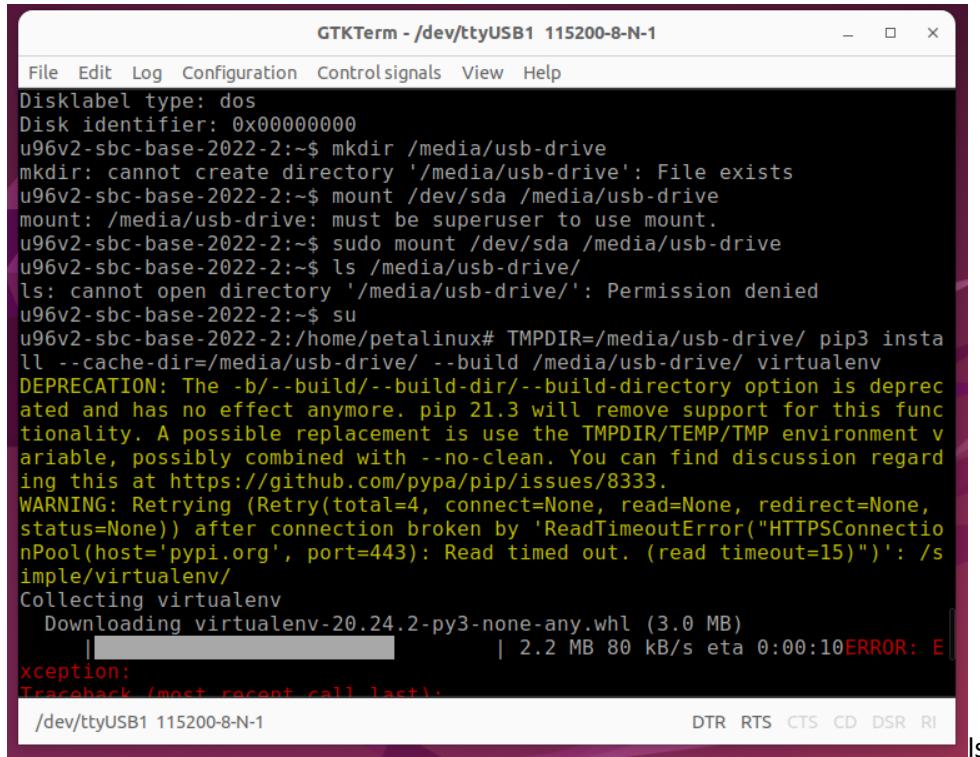
The mount point gets used as temporary workspace for pip3, to circumvent issues about limited space.

Even in this case, the space is not enough.

The virtual environment solution specified below is going to be used:

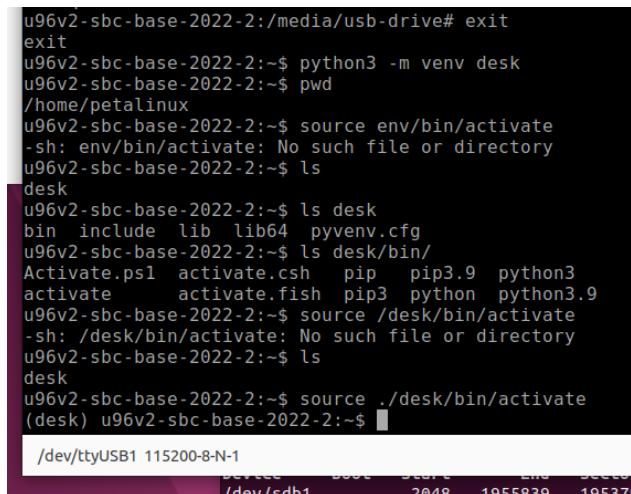
<https://wiki.dave.eu/index.php/DESK-XZ7-AN-0001: Using Python with BORA>

To install the virtual environment:



```
GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Control signals View Help
Disklabel type: dos
Disk identifier: 0x00000000
u96v2-sbc-base-2022-2:~$ mkdir /media/usb-drive
mkdir: cannot create directory '/media/usb-drive': File exists
u96v2-sbc-base-2022-2:~$ mount /dev/sda /media/usb-drive
mount: /media/usb-drive: must be superuser to use mount.
u96v2-sbc-base-2022-2:~$ sudo mount /dev/sda /media/usb-drive
u96v2-sbc-base-2022-2:~$ ls /media/usb-drive/
ls: cannot open directory '/media/usb-drive/': Permission denied
u96v2-sbc-base-2022-2:~$ su
u96v2-sbc-base-2022-2:/home/petalinux# TMPDIR=/media/usb-drive/ pip3 install --cache-dir=/media/usb-drive/ --build /media/usb-drive/ virtualenv
DEPRECATION: The -b/--build/--build-dir/--build-directory option is deprecated and has no effect anymore. pip 21.3 will remove support for this functionality. A possible replacement is use the TMPDIR/TEMP/TMP environment variable, possibly combined with --no-clean. You can find discussion regarding this at https://github.com/pypa/pip/issues/8333.
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ReadTimeoutError("HTTPSConnectionPool(host='pypi.org', port=443): Read timed out. (read timeout=15)": /simple/virtualenv/
Collecting virtualenv
  Downloading virtualenv-20.24.2-py3-none-any.whl (3.0 MB)
    |██████████| 2.2 MB 80 kB/s eta 0:00:10ERROR: E
      exception:
      Traceback (most recent call last):
      /dev/ttyUSB1 115200-8-N-1
      DTR RTS CTS CD DSR RI
ls
```

Exit as su and create the virtual environment in the home folder:



```
u96v2-sbc-base-2022-2:/media/usb-drive# exit
exit
u96v2-sbc-base-2022-2:~$ python3 -m venv desk
u96v2-sbc-base-2022-2:~$ pwd
/home/petalinux
u96v2-sbc-base-2022-2:~$ source env/bin/activate
-sh: env/bin/activate: No such file or directory
u96v2-sbc-base-2022-2:~$ ls
desk
u96v2-sbc-base-2022-2:~$ ls desk
bin include lib lib64 pyvenv.cfg
u96v2-sbc-base-2022-2:~$ ls desk/bin/
Activate.ps1 activate.csh pip pip3.9 python3
activate activate.fish pip3 python python3.9
u96v2-sbc-base-2022-2:~$ source /desk/bin/activate
-sh: /desk/bin/activate: No such file or directory
u96v2-sbc-base-2022-2:~$ ls
desk
u96v2-sbc-base-2022-2:~$ source ./desk/bin/activate
(desk) u96v2-sbc-base-2022-2:~$ 
/dev/ttyUSB1 115200-8-N-1
      device   busid   size   LBA   sectors
      /dev/sdb1       2048 1955839 195376
```

```
[File Edit Log Configuration ControlLog View Help]
:sh /desk/bin/activate; No such file or directory
[u96v2-sbc-base-2022-2:~]$ ls
desk
[u96v2-sbc-base-2022-2:~]$ source ./desk/bin/activate
[desk] u96v2-sbc-base-2022-2:~$ pip3 install --upgrade pip
Requirement already satisfied: pip in ./desk/lib/python3.9/site-packages (21.2.4)
Collecting pip
  Downloading pip-23.2.1-py3-none-any.whl (2.1 MB)
    100% |██████████| 2.1 MB 446 kB/s
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 21.2.4
    Uninstalling pip-21.2.4...
      Successfully uninstalled pip-21.2.4
WARNING: The found .pyc, .pyo, .pyd, .pyd files do not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /home/petalinux/desk/lib/python3.9/site-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: The found .pyc, .pyo, .pyd, .pyd files do not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /home/petalinux/desk/lib/python3.9/site-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Additional context:
User: False
home = None
root = None
prefix = None
Successfully Installed pip-23.2.1
[desk] u96v2-sbc-base-2022-2:~$ pip3 install --upgrade setuptools
Requirement already satisfied: setuptools in ./desk/lib/python3.9/site-packages (58.1.6)
[desk] u96v2-sbc-base-2022-2:~$ pip3 install wheel
  Obtaining dependency information for setuptools from https://files.pythonhosted.org/packages/c7/42/belc7bdd83e1bf160c94b9cafde2e5efc7400346cf7ccdbdb452c467fa/setuptools-68.0.0.dist-info
    Obtaining dependency information for wheel from https://files.pythonhosted.org/packages/17/11/f1390e25018ea2218aedbedcf85cd8dd8bebed29n38ac1fd7f5a8889382/wheel-0.41.0-py3-none-any.whl
      Downloading wheel-0.41.0-py3-none-any.whl (6.4 kB)
        100% |██████████| 6.4 kB 119.1 kB/s eta 0:00:00
Installing collected packages: setuptools
  Attempting uninstall: setuptools
    Found existing installation: setuptools 58.1.0
    Uninstalling setuptools-58.1.0...
      Successfully uninstalled setuptools-58.1.0
Successfully Installed setuptools-68.0.0
[desk] u96v2-sbc-base-2022-2:~$ pip3 install wheel
Collecting wheel
  Obtaining dependency information for wheel from https://files.pythonhosted.org/packages/17/11/f1390e25018ea2218aedbedcf85cd8dd8bebed29n38ac1fd7f5a8889382/wheel-0.41.0-py3-none-any.whl
    Downloading wheel-0.41.0-py3-none-any.whl (6.4 kB)
      100% |██████████| 6.4 kB 119.1 kB/s eta 0:00:00
Installing collected packages: wheel
  Successfully Installed wheel-0.41.0
[desk] u96v2-sbc-base-2022-2:~$
```

It keeps on complaining about not having enough space.

An attempt to move the home folder to the USB drive can be done:

<https://ostechnix.com/move-home-directory-to-new-partition-in-linux/>

The issue turned out to be that the USB pendrive had to be formatted in ext4 filesystem.

The pendrive is mounted in the home folder, and all starts fine:

```

GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Controlsignals View Help
[ 64.292065] wilc_sdio mmc1:0001:1 wlan0: handle_rcvd_gnrl_async_info:
conn_result is NULL
[ 66.539821] wilc_sdio mmc1:0001:1 wlan0: handle_rcvd_gnrl_async_info:
conn_result is NULL

u96v2-sbc-base-2022-2 login:
u96v2-sbc-base-2022-2 login: [ 68.551598] wilc_sdio mmc1:0001:1 wlan0:
handle_rcvd_gnrl_async_info: conn_result is NULL
[ 70.787796] wilc_sdio mmc1:0001:1 wlan0: handle_rcvd_gnrl_async_info:
conn_result is NULL

u96v2-sbc-base-2022-2 login: petalinux
Password:

^[[46;73Ru96v2-sbc-base-2022-2:~$ u96v2-sbc-base-2022-2:~$ ;73R
-u96v2-sbc-base-2022-2:~$ -sh: syntax error near unexpected token `;'
u96v2-sbc-base-2022-2:~$ mount /dev/sda /home/petalinux/
mount: /home/petalinux: must be superuser to use mount.
u96v2-sbc-base-2022-2:~$ sudo mount /dev/sda /home/petalinux/
Password:
u96v2-sbc-base-2022-2:~$ cd /home/petalinux/
u96v2-sbc-base-2022-2:~$ ls
desk lost+found
u96v2-sbc-base-2022-2:~$ cd desk
u96v2-sbc-base-2022-2:~/desk$ python3 -m venv desk
u96v2-sbc-base-2022-2:~/desk$ ls
bin lib
capture.py lib64
capture.py- model_final.h5
cat_doorbell.py pippo.jpg
cat_doorbell_mytrainednetwork.py pyenv.cfg
desk README.txt
detect_with_classifier.py share
include train_cat_dog_crash.py
u96v2-sbc-base-2022-2:~/desk$
```

/dev/ttyUSB1 115200-8-N-1 DTR RTS CTS CD DSR RI

To make everything work, is necessary using a monitor because of a failing QT issue.

By launching the script from a command shell on the video of the Ultra96 Linux distribution, all works as on the PC.

PYTHON APPLICATION PROFILING

The next step to follow, after having in place the Python script, is to find how to measure the time of execution of the main pieces of code, by using timers.

<https://realpython.com/python-profiling/>

<https://www.geeksforgeeks.org/profiling-in-python/>

It can be done via a profiler:

<https://stackoverflow.com/questions/582336/how-do-i-profile-a-python-script>

By using:

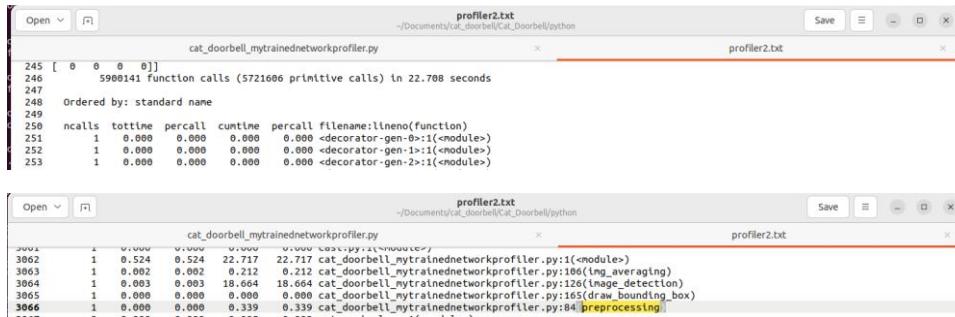
```

cat@cat:~/Documents/cat_doorbell$ python3 -m cProfile cat_doorbell_mytrainednetworkprofiler.py > profiler2.txt
2023-08-17 18:49:31.422154: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following
CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-08-17 18:49:31.212970: W tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library libnvinfer.so.7; derror: libnvinfer.so.7: cannot
open shared object file: No such file or directory; LD_LIBRARY_PATH : /home/cacollo/.local/lib/python3.10/site-packages/nvidia/cuda_runtime/lib
2023-08-17 18:49:32.212970: W tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library libnvinfer_plugin.so.7; derror: libnvinfer_plugin.so.7: cannot
open shared object file: No such file or directory; LD_LIBRARY_PATH : /home/cacollo/.local/lib/python3.10/site-packages/nvidia/cuda_runtime/lib
2023-08-17 18:49:32.212970: W tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library libcudnn.so.11; derror: libcudnn.so.11: cannot
open shared object file: No such file or directory; LD_LIBRARY_PATH : /home/cacollo/.local/lib/python3.10/site-packages/cv2/.:/lib64:/home/cacollo/.local/lib/python3.10/site-pac
kages/nvidia/cuda_runtime/lib
2023-08-17 18:49:33.922370: W tensorflow/compiler/xla/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library libcublaslt.so.11; derror: libcublaslt.so.11: ca
not open shared object file: No such file or directory; LD_LIBRARY_PATH : /home/cacollo/.local/lib/python3.10/site-packages/cv2/.:/lib64:/home/cacollo/.local/lib/python3.10/site-

```

We get a text file with the profiling outcome.

From there, we get data about the key functions:

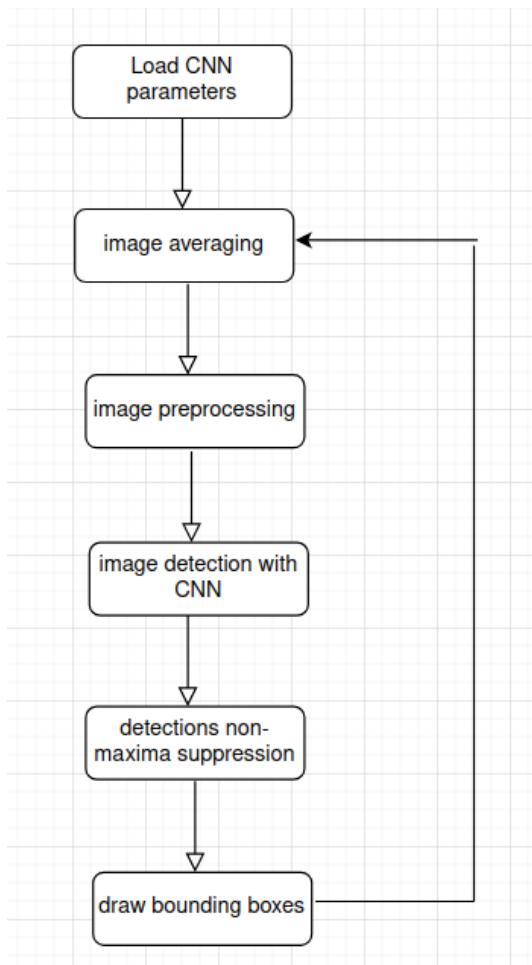


It comes out that the most computing intensive tasks are:

Function	Execution time	Description
Image_detection	18.664 s	CNN cat detector
Preprocessing	0.339 s	Illumination correction with CLAHE, and denoising (via OPENCV)
Img_averaging	0.212 s	Averages 4 consecutive frames

TASK PARTITIONING

A high-level flow chart of the python script application follows:



TASK ALLOCATION

On the Zynq US+, there are several options to implement the key tasks, in the following we will name a few.

Vitis Vision Library:

<https://www.xilinx.com/products/design-tools/vitis/vitis-libraries/vitis-vision.html>

Pynq:

<http://www.pynq.io/>

XRT for C/C++/Python:

<https://www.xilinx.com/products/design-tools/vitis/xrt.html#overview>

<https://xilinx.github.io/XRT/2021.2/html/pyxrt.html>

Linux CMA for HW accelerators:

https://support.xilinx.com/s/article/000034737?language=en_US

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18842412/Accessing+BRAM+In+Linux>

Kernels in Vitis

<https://docs.xilinx.com/r/en-US/ug1393-vitis-application-acceleration>

<https://docs.xilinx.com/r/en-US/Vitis-Tutorials-Hardware-Acceleration>

The Python code of “Image_detection” is:

```
def image_detection(image,min_conf,stepSize):
    #stepSize = 100 #stride (10)
    locs = np.array([0,0,0,0])
    entry = np.array([0,0,0,0])
    first_det = True
    (w_width, w_height) = (150, 200) # search window size
    for x in range(0, image.shape[1] - w_width , stepSize):
        for y in range(0, image.shape[0] - w_height , stepSize):
            window = image[x:x + w_width, y:y + w_height, :]
            cv2.imshow('roi' , np.array(window, dtype = np.uint8 ) )
            cv2.waitKey(1)
            time.sleep(0.12)
            tmp = image.copy()
            #cv2.rectangle(tmp, (x, y), (x + w_width, y + w_height), (255, 0, 0), 2) # draw rectangle on image
            #cv2.imshow('scan window' , np.array(tmp, dtype = np.uint8 ) )
            #cv2.waitKey(1)
            time.sleep(0.12)
            #classify with resnet50
            image_resnet50 = cv2.resize(window, (180, 180)) #resize in the format expected by resnet50
            image_resnet50 = img_to_array(image_resnet50) #convert to numpy array
            #cv2.imshow('scan window resnet' , np.array(image_resnet50, dtype = np.uint8 ) )
            #cv2.waitKey(1)
            #time.sleep(0.12)

            image_resnet50 = np.expand_dims(image_resnet50, axis=0) #adds a dimension to the image
            preds = model.predict(image_resnet50)[0] #predicts the ROI by using resnet50
            print(preds)
            score = float(preds[0])
            #print(f"This image is {100 * (1 - score):.2f}% cat and {100 * score:.2f}% dog.")
            prob_cat = 1 - score
            prob_dog = score

            # filter out weak detections by ensuring the predicted probability
            # is greater than the minimum probability
            if prob_cat >= min_conf:
                entry = [x, y, x + w_width, y + w_height, prob_cat]
                locs = np.vstack((locs,entry))
    return locs
```

And it relies upon a CNN, it is natural to accelerate it with Vitis AI.

The Python code of “preprocessing” is:

```

#image pre-processing function
def preprocessing(frame):
    tmp = frame.copy()
    #illumination correction using CLAHE
    img = cv2.cvtColor(tmp, cv2.COLOR_RGB2Lab)
    #configure CLAHE
    clahe = cv2.createCLAHE(clipLimit=10, tileGridSize=(8,8))
    #0 to 'L' channel, 1 to 'a' channel, and 2 to 'b' channel
    img[:, :, 0] = clahe.apply(img[:, :, 0])
    img = cv2.cvtColor(img, cv2.COLOR_Lab2RGB)
    cv2.imshow("CLAHE", img)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()
    #image denoising
    tmp2 = frame.copy()
    noiseless_image_colored = cv2.fastNlMeansDenoisingColored(tmp2, None, 20, 20, 7, 21)
    cv2.imshow("Denoised", img)
    #cv2.waitKey(0)
    #cv2.destroyAllWindows()

    return noiseless_image_colored

```

And as it relies upon OpenCV, it is natural to use the Vitis Vision libraries and an HW Kernel accelerator.

The Python code of “Img_averaging” is:

```

)frame averaging
def img_averaging(vid):
    # Capture the video frame
    # by frame
    ret, frame = vid.read()
    np_frame = np.array(frame).astype(np.float32)

    ret, frame1 = vid.read()
    np_frame1 = np.array(frame1).astype(np.float32)

    ret, frame2 = vid.read()
    np_frame2 = np.array(frame2).astype(np.float32)

    ret, frame3 = vid.read()
    np_frame3 = np.array(frame3).astype(np.float32)

    average_frame = (np_frame+np_frame1+np_frame2+np_frame3)/4
    average_frameint = average_frame .astype(np.uint8)
    return average_frameint

```

And it can be done via OpenAMP on the R5 real time cores.

Then summarizing, we will have the following HW acceleration methods in use:

Function	Hw acceleration method
Image_detection	Vitis AI
preprocessing	Vitis Vision Libraries
Img_averaging	Open AMP on the R5 core

OPEN AMP: accelerating “Img_averaging”

The home page for the Open AMP project is at:

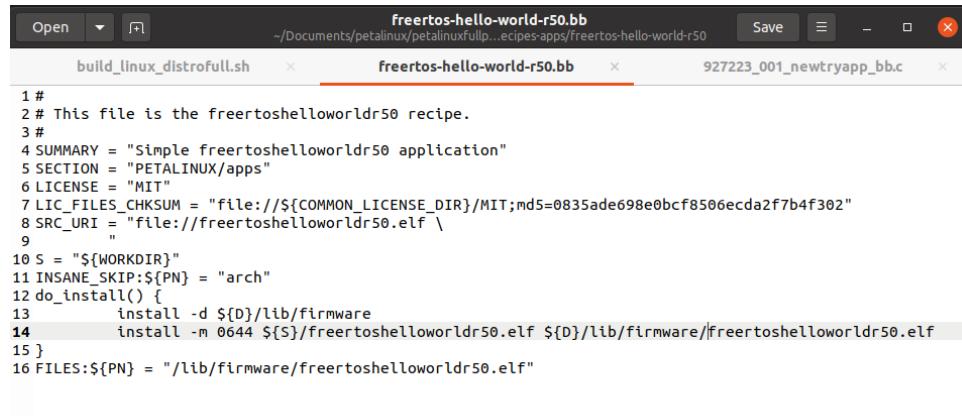
<https://www.openampproject.org/>

And Xilinx is among the key members.

A good tutorial about its usage on the board is at:

<https://www.hackster.io/LogicTronix/openamp-in-xilinx-mpsoc-fpga-running-petalinux-baremetal-e025ca>

A sample elf file for the r5 has been generated in vitis, and the instructions have been followed to create the petalinux application recipe:



```
freertos-hello-world-r50.bb
...
1 # This file is the freertos-helloworldr50 recipe.
2 #
3 #
4 SUMMARY = "Simple freertoshelloworldr50 application"
5 SECTION = "PETALINUX/apps"
6 LICENSE = "MIT"
7 LIC_FILES_CHKSUM = "file://${COMMON_LICENSE_DIR}/MIT;md5=0835ade698e0bcf8506ecda2f7b4f302"
8 SRC_URI = "file://freertos-helloworldr50.elf "
9
10 S = "${WORKDIR}"
11 INSANE_SKIP:${PN} = "arch"
12 do_install() {
13     install -d ${D}/lib/firmware
14     install -m 0644 ${S}/freertos-helloworldr50.elf ${D}/lib/firmware/freertos-helloworldr50.elf
15 }
16 FILES:${PN} = "/lib/firmware/freertos-helloworldr50.elf"
```

The tutorial:

<https://www.hackster.io/timothy-vales/openamp-on-the-snickerdoodle-black-691912>

Has been followed so to configure the kernel and the root filesystem.

The bitbake recipe has been changed as per:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/2423488521/OpenAMP+Project+Build+Process>

By replacing “INSANE_SKIP:\${PN}” to “INSANE_SKIP_\${PN}”.

The dts file has been found at:



And the Xilinx wiki is at:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18841718/OpenAMP>

Other useful links follow:

https://support.xilinx.com/s/article/826083?language=en_US

<https://www.hackster.io/sasha-falkovich/openamp-on-kria-kv260-ccb30>

https://support.xilinx.com/s/article/826083?language=en_US

The first step to check, is seeing if OpenAMP is enabled in the Petalinux distribution generated by the Avnet BSP:

A screenshot of a terminal window titled "GTKTerm - /dev/ttyUSB1 115200-8-N-1". The window has a menu bar with "File", "Edit", "Log", "Configuration", "Controls", "signals", "View", and "Help". Below the menu is a command prompt: "u96v2-sbc-base-2022-2:~\$ ls /sys/class/remoteproc/". The terminal window shows the output of the command: "u96v2-sbc-base-2022-2:~\$ [REDACTED]".

The folder "remoteproc" is empty.

Petalinux must be rebuilt by enabling the missing bits.

If the build fail, check this thread:

https://support.xilinx.com/s/question/0D54U00006sZafqSAC/error-petalinuximageminimal10r0-doinagecpio-executionerrorerror-logfile-of-failure-stored-in?language=en_US

To get rid of a cpio error message.

Vitis Vision Libraries: accelerating “preprocessing”

A quickstart video introducing the library, is available at:

<https://www.xilinx.com/video/software/vitis-hls-l1-library-wizard.html>

Follow it to clone from git the library of interest, and to use it.

A decent video with a tutorial about how to use the libraries, is at:

<https://www.youtube.com/watch?v=hyhISFa74bs>

And at:

<https://www.youtube.com/watch?v=a9ozwIT98rc>

A tutorial using petalinux, is at:

<https://www.xilinx.com/video/software/vitis-embedded-acceleration-platform-creation-demystified.html>

Vitis AI: accelerating “Image_detection”

End toolsuite selection

After having had many issues with Petalinux, Ubuntu 22.04 LTS has been chosen, for it the following guide has been followed:

<https://www.hackster.io/whitney-knitter/vivado-vitis-petalinux-2023-1-install-on-ubuntu-22-04-ab28da>

How to debug R5 applications on Linux via VITIS

To debug R5 applications on a running Petalinux distribution, setup the TCF agent as per the tutorial below:

<https://www.hackster.io/whitney-knitter/debugging-your-custom-linux-applications-using-vitis-67c022>

In order to make it work, follow the kernel build instructions below, needed to not keep in idle the R5:

https://support.xilinx.com/s/article/69143?language=en_US

For a step by step procedure to set up the debug for bare metal r5 debug on a running linux system, look at:

https://support.xilinx.com/s/article/1225679?language=en_US

How to debug applications on Linux via VITIS

To debug Linux applications from within Vitis, follow the following tutorial:

<https://www.xilinx.com/video/software/building-linux-application-vitis.html>

<https://www.hackster.io/whitney-knitter/debugging-your-custom-linux-applications-using-vitis-67c022>

OPENMP in Vitis

https://xilinx.github.io/Vitis-Tutorials/2021-2/build/html/docs/Hardware_Acceleration/Introduction/06-meet-the-other-shoe.html

Vitis tutorials and HW accelerators

According to AVnet:

<https://community.element14.com/technologies/fpga-group/b/blog/posts/avnet-hdl-git-howto-vivado-2020-2-and-later>

Reference designs are available on their servers.

BSPs are provided:

<https://community.element14.com/technologies/fpga-group/b/blog/posts/petalinux-2021-1-bsps-now-available-for-avnet-soms-and-sbcs-244855059>

And they support the “vector addition example”.

This example is described at the home page about HW acceleration on Vitis:

https://xilinx.github.io/Vitis_Accel_Examples/2021.1/html/simple_vadd.html

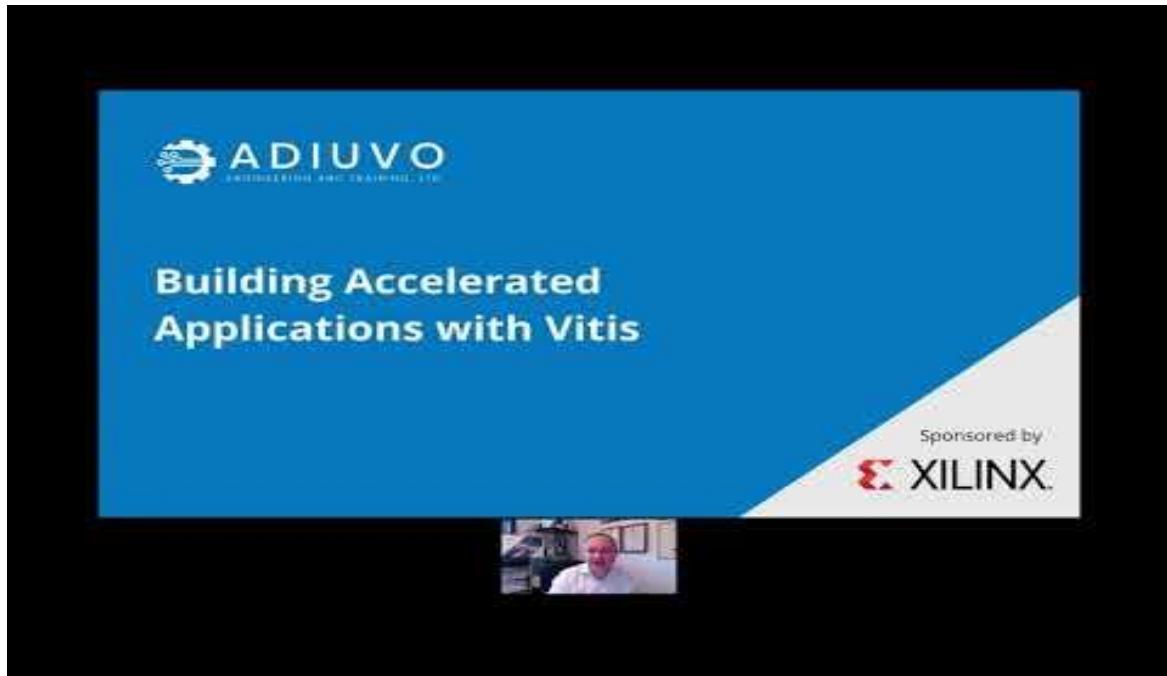
<https://docs.xilinx.com/r/en-US/Vitis-Tutorials-Vitis-Platform-Creation/Test-2-Run-Vector-Addition-Application?tocId=8w7OXVMcu2r75uTH6UGdhA>

The landing page about these tutorials is at:

https://github.com/Xilinx/Vitis-Tutorials/tree/2023.2/Getting_Started/Vitis

And a webinar can be found at:

<https://www.youtube.com/watch?v=a9ozwIT98rc>



The official tutorial for the HW accelerator is at:

<https://docs.xilinx.com/r/en-US/Vitis-Tutorials-Hardware-Acceleration/Design-Tutorials?tocId=ySpgAKBHTPrnnilZGgczRA>

And the manual for HW accelerators under vitis:

<https://docs.xilinx.com/r/en-US/ug1393-vitis-application-acceleration/Getting-Started-with-Vitis>

It is possible to develop the HW accelerator in Vitis HLS, and then import the generated xo file in Vitis as per the tutorial:

<https://xilinx.github.io/Vitis-Tutorials/2020-1/docs/build/html/docs/Getting Started/Vitis HLS/using the kernel.html>

And an in-depth explanation of the whole flow is at:

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-software-in-petalinux-b2456c>

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-vitis-platform-c3537e>

Official Xilinx Vitis in depth tutorial are at:

<https://github.com/Xilinx/Vitis-Tutorials>

And the video about it:

<https://www.xilinx.com/video/software/vitis-embedded-acceleration-platform-creation-demystified.html>

And an in depth tutorial is at:

<https://docs.xilinx.com/r/en-US/Vitis-Tutorials-Hardware-Acceleration/Open-the-IP-Packager>

The page for the 2022.2 Vitis release is at:

<https://docs.xilinx.com/r/2022.2-English/Vitis-Tutorials-Hardware-Acceleration/Design-Tutorials?tocId=ySpgAKBHTPrnnilZGgczRA>

In order to use XRT, it has to be downloaded from the GIT psge:

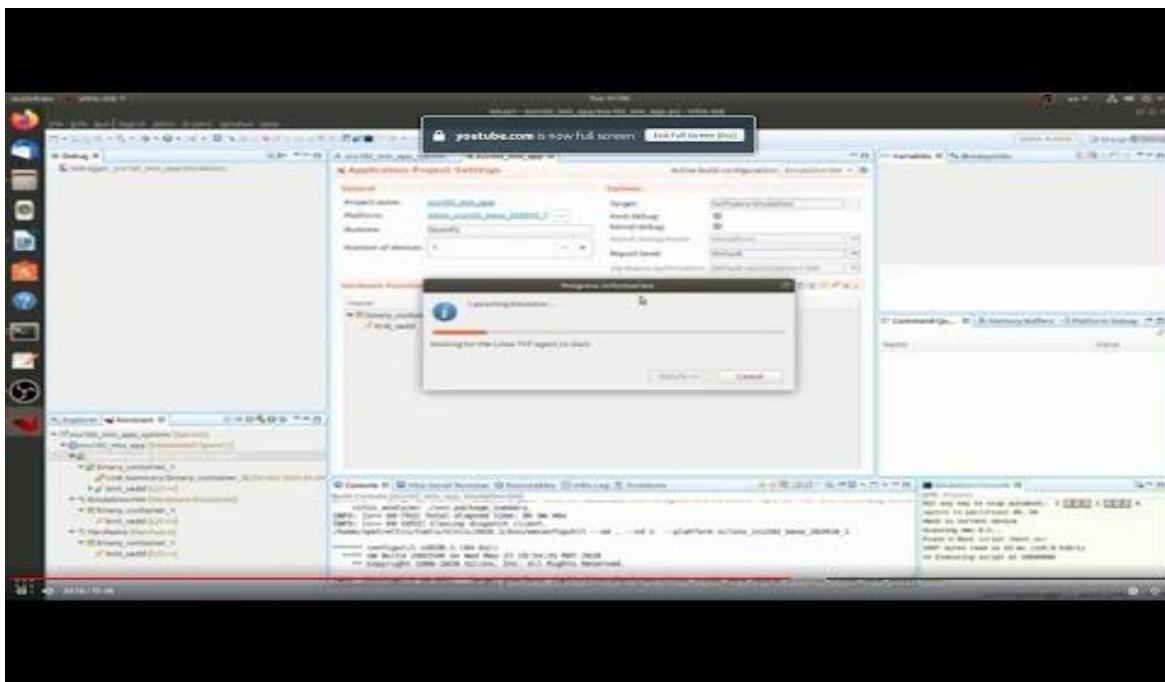
<https://github.com/Xilinx/XRT>

And installed as per the instructions at the wiki page:

<https://xilinx.github.io/XRT/master/html/build.html>

A good set of tutorials about vitis accelerated flow are on youtube at:

<https://www.youtube.com/watch?v=6-Yox8fcUxc&list=PLXUuQj2gQ4s8HjchuPAXtkTg37cYvIKKp>



How to create and debug Linux applications in VITIS

To create and debug Linux applications in Vitis follow the tutorial below:

https://support.xilinx.com/s/article/1141772?language=en_US

And the video:

<https://www.xilinx.com/video/software/building-linux-application-vitis.html>

In order to develop, libraries are needed (e.g. Vitis will complain about missing /usr/lib folder).

So the following tutorial can be followed:

https://support.xilinx.com/s/article/1175461?language=en_US

In order to get something meaningful, libraries must be added to the Petalinux image, to accomplish this, follow the tutorial below:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18842346/Linux+Applications+with+Petalinux+Libraries>

Also worth considering building an SDK after having generated a Linux image:

<https://docs.xilinx.com/r/en-US/ug1144-petalinux-tools-reference-guide/SDK-Generation-Target-Sysroot-Generation>

Vish-list image processing and enhancement python

<https://www.toptal.com/opencv/python-image-processing-in-computational-photography>

Vitis and GIT

Vitis has an integrated GIT plugin.

Instructions about how to use it are at:

https://support.xilinx.com/s/article/1173362?language=en_US

<https://www.youtube.com/watch?v=2VYeujilJk>



And on the UG1553, at the git integration chapter:

<https://docs.amd.com/r/en-US/ug1553-vitis-ide/Git-Integration>

HOWTO to remove issues with the interrupt controller

Describes the issues of the interrupt controller hanging up the kernel during booting:

<https://community.element14.com/products/devtools/avnetboardscommunity/avnetboard-forums/f/software-application-development/51813/ultra96-v2-petalinux-boot-error-with-vivado-custom-design>

[How to solve /amba/axi-interrupt-ctrl issue on ultra96v2 \(xilinx.com\)](#)

HOWTO build the vivado design and the project used by the BSP

It contains the instructions to get the Vivado design used in the BSP:

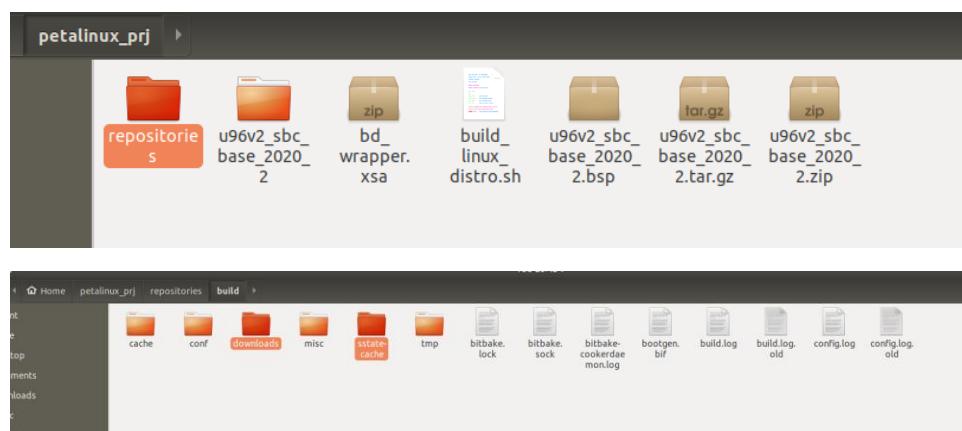
[Ultra96-V2 - Building the foundational designs - Hackster.io](#)

HOWTO accelerate the build by using cached source files

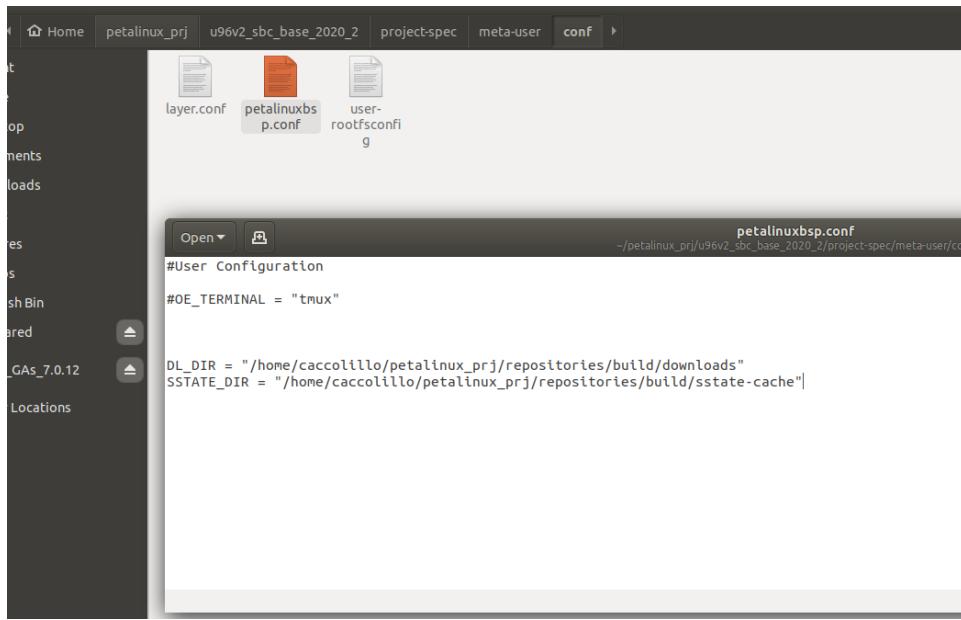
Describes how to speed up the build by using cached source files:

[Accelerating PetaLinux BSP Build Time - element14 Community](#)

In my case, I've saved and renamed an old build of petalinux:



And in the new projects, once configured and imported the xsa file, I edit the file "petalinuxbsp.conf" as follows:



HOWTO add to the device tree a custom HLS peripheral

By using the BSP, custom HW does not get added to the device tree.

To get the device tree compiler and use it, follow the guide below:

[Build Device Tree Compiler \(dtc\) - Xilinx Wiki - Confluence \(atlassian.net\)](#)

To see an example about how to integrate an HLS block in Petalinux, have a look at:

[Zedboard: Linux UIO Application using Vitis IDE - Hackster.io](#)

Another example of usage of HLS in Petalinux, is at:

[Integrating an HLS accelerator into Petalinux – FPGAWorld \(wordpress.com\)](#)

Yet another in-depth guide about how to get the device tree for HLS and use it, is at:

[KV260 DPU-TRD Petalinux 2022.1 Vivado Flow - Hackster.io](#)

The following guide has been followed to install the device tree compiler:

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-software-in-petalinux-b2456c#toc-install-device-tree-generator-4>

And to create the dts file.

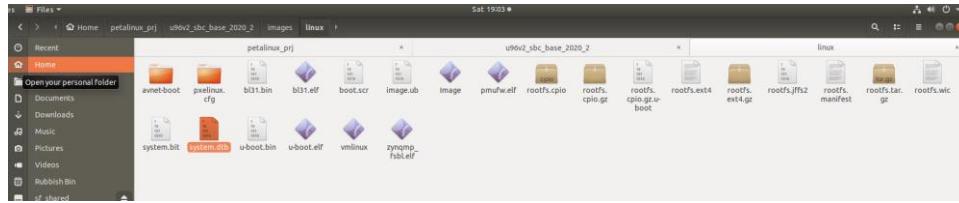
Instructions about how to add a node in the device tree, are at:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18842482/Device+Tree+Tips?showComments=true&showCommentArea=true#DeviceTreeTips-9Howtoaddordelete nodesandpropertiesinPetaLinux>

At the following link, instructions about how to debug device tree:

[PetaLinux Image Debug Series: Debugging the Device Tree Generator \(xilinx.com\)](https://www.xilinx.com/support/documentation/sw_documentation/ug1102-petalinux-image-debug-series/index.html)

At the end of the build, the dtb (device tree blob) file is in the folder below:



And it can be decompiled as per the following guide:

[Decompiling the Device Tree \(DTB\) for Embedded Linux – Silicon Blade Consultants](https://siliconbladeconsultants.com/decompiling-the-device-tree-dtb-for-embedded-linux/)

```
caccolillo@caccolillo-VirtualBox:~/petalinux_prj/u96v2_sbc_base_2020_2$ pwd
/home/caccolillo/petalinux_prj/u96v2_sbc_base_2020_2
caccolillo@caccolillo-VirtualBox:~/petalinux_prj/u96v2_sbc_base_2020_2$ dtc -I dtb system.dtb -o pipo.dts
pipo.dts: Warning (ranges_format): /zyqnmp_ipi1 has empty "ranges" property but its #address-cells (1) differs from / (2)
pipo.dts: Warning (ranges_format): /zyqnmp_ipi1 has empty "ranges" property but its #size-cells (1) differs from / (2)
pipo.dts: Warning (unit_address_vs_reg): Node /amba/axi-interrupt-ctrl has a reg or ranges property, but no unit name
pipo.dts: Warning (unit_address_vs_reg): Node /amba_pl@0 has a unit name, but no reg property
pipo.dts: Warning (unit_address_vs_reg): Node /memory has a reg or ranges property, but no unit name
pipo.dts: Warning (unit_address_vs_reg): Node /zyqnmp_rpu has a reg or ranges property, but no unit name
pipo.dts: Warning (unit_address_vs_reg): Node /zyqnmp_rpu/r5@0 has a unit name, but no reg property
pipo.dts: Warning (unit_address_format): Node /amba/spi@ff040000/spidev@0x00 unit name should not have leading "0x"
pipo.dts: Warning (unit_address_format): Node /amba/spi@ff040000/spidev@0x00 unit name should not have leading "0x"
pipo.dts: Warning (unit_address_format): Node /amba/spi@ff050000/spidev@0x00 unit name should not have leading "0x"
pipo.dts: Warning (unit_address_format): Node /amba/spi@ff050000/spidev@0x00 unit name should not have leading "0x"
pipo.dts: Warning (unit_address_format): Node /krln_img_averaging@0xa00a0000 unit name should not have leading "0x"
pipo.dts: Warning (pci_device_reg): Node /amba/pcie@fde0000/legacy-interconnect-controller missing PCI reg property
pipo.dts: Warning (simple_bus_reg): Node /amba/axi-interrupt-ctrl simple-bus unit address format error, expected "a0e90000"
pipo.dts: Warning (simple_bus_reg): Node /amba/zynqlmm_drm missing or empty reg/ranges property
pipo.dts: Warning (gpios_property): Could not get phandle node for /_symbols_:gpto(cell 0)
caccolillo@caccolillo-VirtualBox:~/petalinux_prj/u96v2_sbc_base_2020_2$ aver
```

The system-user.dtsi file used to create the dts file is:

```
/include/ "system-conf.dtsi"
{
    amba_pl: amba_pl@0 {
        krnl_img_averaging@0xa00a0000 {
            /* This is a place holder node for a custom IP, user may need to update the entries */
            clock-names = "s_axi_aclk";
            clocks = <&zynqmp_clk 71>;
            compatible = "xlnx,krnl-img-averaging-1.0";
            reg = <0x0 0xa00a0000 0x0 0x10000>;
            xlnx,s-axi-control-addr-width = <0x7>;
            xlnx,s-axi-control-data-width = <0x20>;
        };
    };
}
```

```

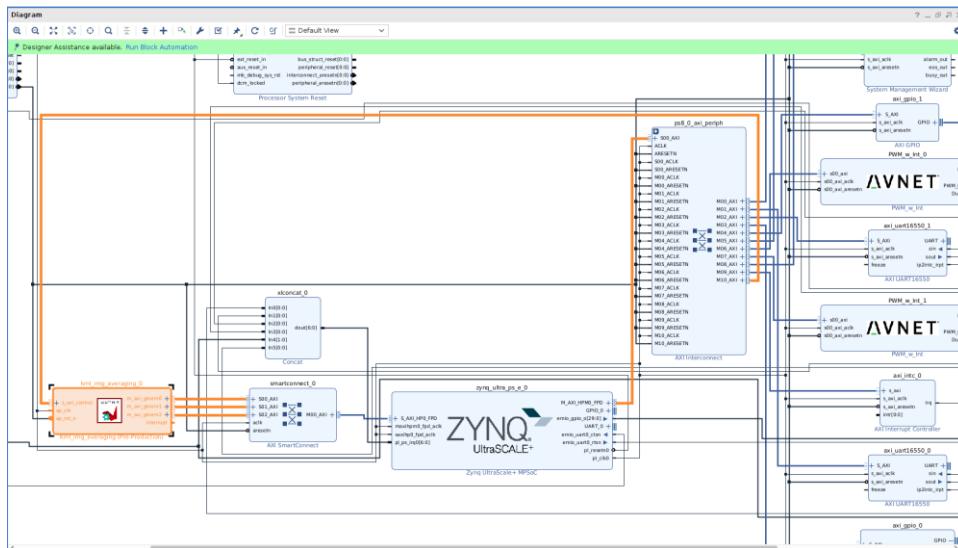
system-user.dtsi [Read-Only]
~/petalinux_prj/u96v2_sbc_base_2020.2/proj_ec/meta-user/recipes-bsp/device-tree/files

/include/ "system-conf.dtst"
/ {
    amba_pl: amba_pl@0 {
        krln_img_averaging@0xa00a0000 {
            /* This is a place holder node for a custom IP, user may need to update the entries */
            clock-names = "s_axi_aclk";
            clocks = <&cyqmp_clk_7>;
            compatible = "xlnx,krln-img-averaging-1.0";
            reg = <0x0 0xa00a0000 0x0 0x10000>;
            xlnx,s-axi-control-addr-width = <0x7>;
            xlnx,s-axi-control-data-width = <0x20>;
        };
    };
}

```

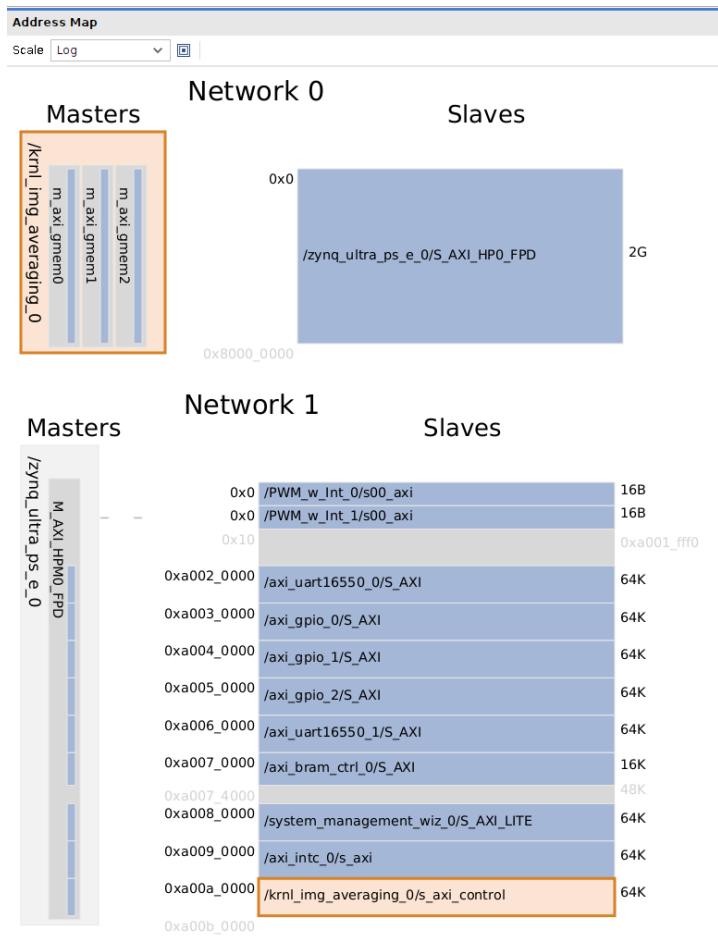
C development under LINUX for a custom HLS peripheral

The image averaging HLS IP core has been integrated and connected in Vivado:



It has three AXI master ports used to get the two source images and write back the result one to the DDR memory, and an AXI slave port for management purposes.

The address map for the peripheral follows:



The AXI registers for the peripherals are mapped starting at 0xa00a0000.

By looking at the automatically generated device drivers in HLS, we get the register set internal details:

```

Open ▾ xkrnl_img_averaging_hw.h
~/Documents/Cat_Doorbell/vitis_hls/m...sc/drivers/krn...img_averaging_v1_0/src

1 // =====
2 // Vitis HLS - High-Level Synthesis from C, C++ and OpenCL v2022.2 (64-bit)
3 // Tool Version Limit: 2019.12
4 // Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.
5 // =====
6 // control
7 // 0x00 : Control signals
8 //   bit 0 - ap_start (Read/Write/COH)
9 //   bit 1 - ap_done (Read)
10 //   bit 2 - ap_idle (Read)
11 //   bit 3 - ap_ready (Read/COR)
12 //   bit 4 - ap_continue (Read/Write/SC)
13 //   bit 7 - auto_restart (Read/Write)
14 //   bit 9 - interrupt (Read)
15 //   others - reserved
16 // 0x04 : Global Interrupt Enable Register
17 //   bit 0 - Global Interrupt Enable (Read/Write)
18 //   others - reserved
19 // 0x08 : IP Interrupt Enable Register (Read/Write)
20 //   bit 0 - enable ap_done interrupt (Read/Write)
21 //   bit 1 - enable ap_ready interrupt (Read/Write)
22 //   others - reserved
23 // 0x0c : IP Interrupt Status Register (Read/TOW)
24 //   bit 0 - ap_done (Read/TOW)
25 //   bit 1 - ap_ready (Read/TOW)
26 //   others - reserved
27 // 0x10 : Data signal of in1
28 //   bit 31-0 - in1[31:0] (Read/Write)
29 // 0x14 : Data signal of in1
30 //   bit 31-0 - in1[63:32] (Read/Write)
31 // 0x18 : reserved
32 // 0xic : Data signal of in2
33 //   bit 31-0 - in2[31:0] (Read/Write)
34 // 0x20 : Data signal of in2
35 //   bit 31-0 - in2[63:32] (Read/Write)
36 // 0x24 : reserved
37 // 0x28 : Data signal of out_r
38 //   bit 31-0 - out_r[31:0] (Read/Write)
39 // 0x2c : Data signal of out_r
40 //   bit 31-0 - out_r[63:32] (Read/Write)
41 // 0x30 : reserved
42 // 0x34 : Data signal of scale_by_4
43 //   bit 31-0 - scale_by_4[31:0] (Read/Write)
44 // 0x38 : reserved
45 // 0x3c : Data signal of size
46 //   bit 31-0 - size[31:0] (Read/Write)
47 // 0x40 : reserved
48 // (SC = Self Clear, COR = Clear on Read, TOW = Toggle on Write, COH = Clear on Handshake)
49

```

A petalinux distribution gets generated for this HW release integrating the image averaging IP, and a quick test with devmem is done, to ascertain that the internal registers can be accessed:

```

File Edit Log Configuration Control signals View Help
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a0000
0x00000081
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a0000 32 0x00000000
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a0000
0x0000000E
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a0000 32 0x00000004
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a0000
0x00000006
root@u96v2-sbc-base-2020-2:~# █

root@u96v2-sbc-base-2020-2:~# devmem 0xa000a003c
0x00000000
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a003c 32 0x000f0000
root@u96v2-sbc-base-2020-2:~# devmem 0xa000a003c
0x000F0000
root@u96v2-sbc-base-2020-2:~# █

```

It proves the device is where it's meant to be and can be accessed and managed by petalinux.

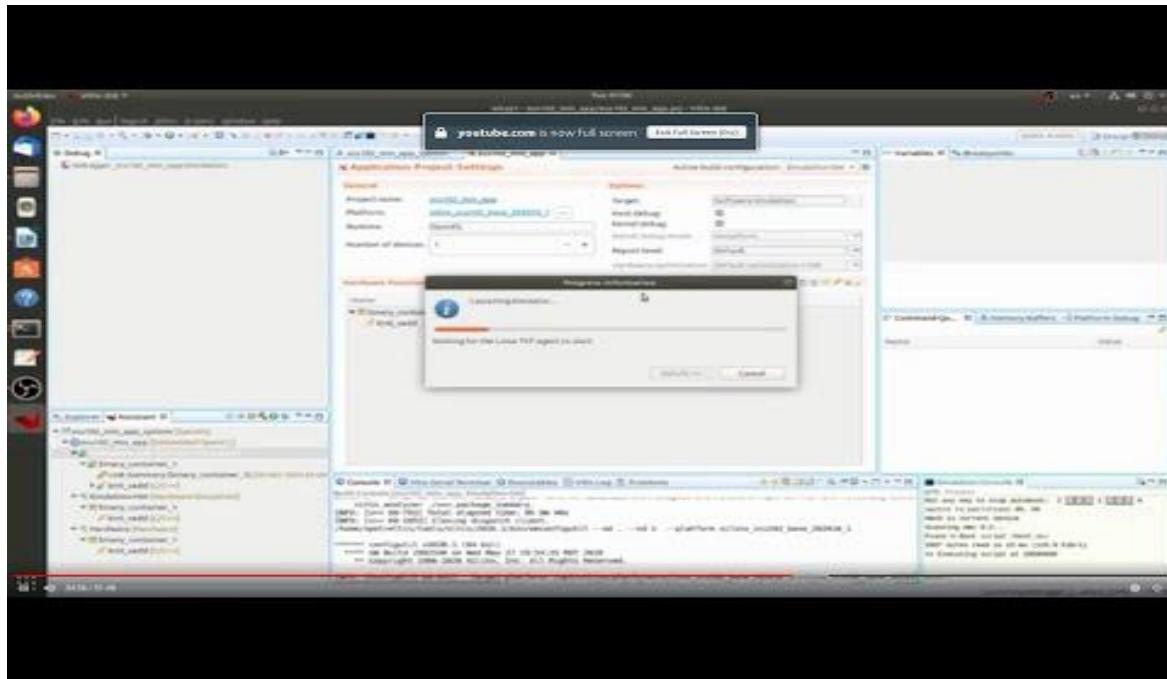
Now a C application under Linux can be developed to manage the IP core.

The guidance at:

<https://www.hackster.io/whitney-knitter/debugging-your-custom-linux-applications-using-vitis-67c022>

https://xilinx.github.io/Vitis-Tutorials/master/docs-ip/docs/Vitis_Platform_Creation/Design_Tutorials/03_Edge_VCK190/step4.html

<https://www.youtube.com/watch?v=6-Yox8fcUxc&list=PLXUUQj2gQ4s8HjchuPAXtkTg37cYvIKKp>



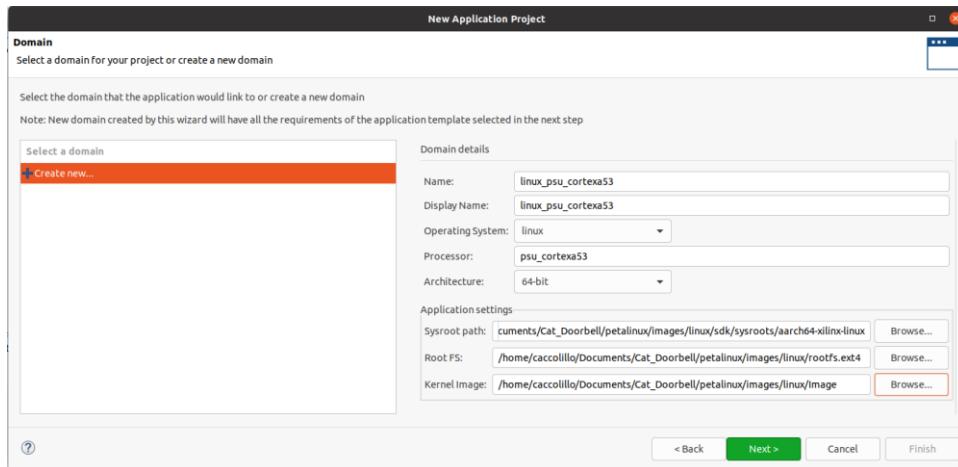
Shall be followed. It shows how to write and test on a live Linux system a C application.

Once confident about it, a petalinux recipe for permanent deployment can be found at:

<https://www.hackster.io/whitney-knitter/custom-application-creation-in-petalinux-on-the-zynqberry-15ff77>

A new application project gets created, and the xsa file generated in Vivado gets used.

The SMP option has to be used, and the sysroot, root fs and the image gets specified as follow:



Where sysroot is generated as sdk in petalinux.

A trivial helloworld application has been chosen as starting point, and when built, gives the following error:

```
Console Problems Vitis Log Guidance
Build Console [img_averaging_manager.Debug]
10:45:48 **** Build of configuration Debug for project img_averaging_manager ****
make: Entering directory `/home/caccolillo/Documents/Cat_Doorbell/petalinux/images/linux/sdk/sysroots/aarch64-xilinx-linux'
Building file: ./src/helloworld.c
Invoking: ARM v8 Linux gcc compiler
armv8-gcc -c -fmessage-length=0 -MT src/helloworld.o -MF .depend/src/helloworld.d
Finished building: ./src/helloworld.c

Building target: img averaging manager.elf
Invoking: ARM v8 Linux gcc linker
aarch64-linux-gnu-gcc -L/home/caccolillo/Documents/Cat_Doorbell/petalinux/images/linux/sdk/sysroots/aarch64-xilinx-linux/lib -L/home/caccolillo/Documents/Cat_Doorbell/petalinux/images/linux/sdk/sysroots/aarch64-xilinx-linux/lib -L/home/caccolillo/Documents/xilinx/Vitis/2022.2/gnu/aarch64/lib/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `ldscript.o': No such file or directory
/home/caccolillo/Documents/xilinx/Vitis/2022.2/gnu/aarch64/lib/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `lgcc'
collect2: error: ld returned 1 exit status
make: *** [makefile:38: img_averaging_manager.elf] Error 1
10:45:48 Build Finished (took 214ms)
```

By looking online:

https://support.xilinx.com/s/article/73705?language=en_US

It comes out that petalinux has to be restarted and petalinux-bsp.conf changed as follows:

73705 - Vitis 2020.1 - Linker error due to missing up-to-date gcc package in sysroot

DESCRIPTION
As a result of the gcc version in sysroot not being up-to-date, the user might see the below linker error:

```
Vitis/2020.1/gnu/aarch64/lin/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `ldscript.o': No such file or directory
Vitis/2020.1/gnu/aarch64/lin/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `lgcc'
Vitis/2020.1/gnu/aarch64/lin/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `lgcc_stis/2020.1/gnu/aarch64/lin/aarch64-linux/x86_64-petalinux-linux/usr/bin/aarch64-xilinx-linux/aarch64-xilinx-linux-id.real: cannot find `lgcc_s'
collect2: error: ld returned 1 exit status
make: *** [dpd-app-smp.elf] Error 1
```

SOLUTION
To resolve this issue, please add the below configuration into the `petalinux-bsp.conf` file in Petalinux.

```
IMAGE_INSTALL_append += " libgcc-dev"
```

This delivers GCC development files to sysroot.

Despite this modification, Vitis still complaints about the presence of the same error.

By looking online, the following posts have been looked at:

https://support.xilinx.com/s/question/0D52E00006hpLY6SAM/petalinux-build-in-libgcc?language=en_US

https://support.xilinx.com/s/question/0D52E00006hpl3YSAQ/petalinux-20183-missing-libgccss01-while-using-pthread?language=en_US

https://support.xilinx.com/s/question/0D52E00006rAMJCSA4/libgccso1-not-installed-even-though-libgcc-selected-in-rootfs-configuration?language=en_US

<https://www.hackster.io/fabioc9675/empowering-dune-creating-petalinux-2022-2-os-image-for-kria-2fbca1>

Trying again by following this Xilinx tutorial:

[Building a Linux Application in the Vitis IDE \(xilinx.com\)](#)

Here is shown how to develop natively on the Ultra96 v2:

[Twitch_SpringSales_UK \(youtube.com\)](#)



BSP PETALINUX issues and build without a BSP

After many attempts, builds have started failing, by using the AVNET BSP for Petalinux.

This was due mainly to issues with the Microchip WiFi driver.

In light of this, the choice to go BSP-less has been taken.

The method of using a template for the Zynq Ultrascale+ described in the blog below has been used:

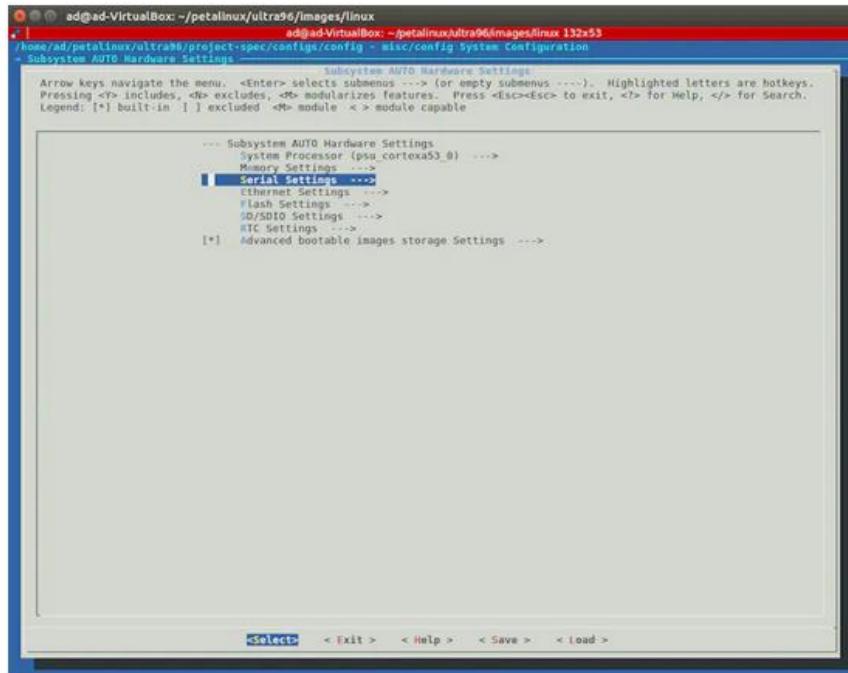
<https://www.hackster.io/adam-taylor/two-methods-of-building-petalinux-for-the-ultra96-77c8e0>

And the Vivado project has been reverted back, with a block design which is hosting only a Zynq ultrascale+ with default settings:

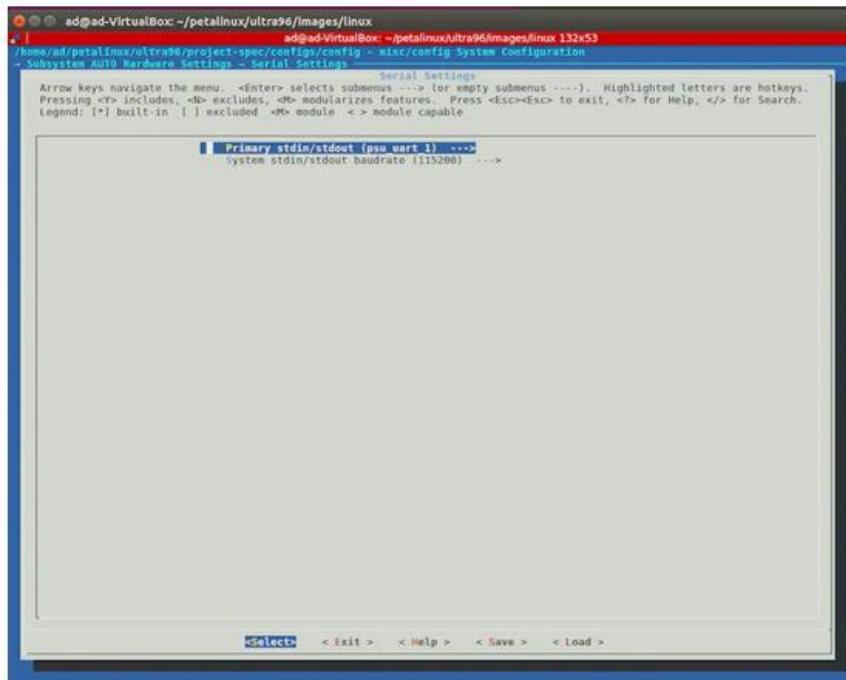


No fan control is in place: all is relying upon the usage of some sort of pull-up resistor on the PCB, as the fan is spinning at full speed anyway.

Once built the XSA, the zynq ultrascale + template has been used as is, with the exception of changing the stdio to the following way:



Selecting Serial interfaces



Selecting UART 1

Plugging the ethernet cable between the PC and the USB to ethernet dongle on the ULTRA96 v2, static IP addresses get set:

```

GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Controlsignals View Help
[ 37.875218] audit: type=1006 audit(1637342400.924:2): pid=596 uid=0 old-auid=4294967295 auid=1000 tty=(none) old-ses=4294967295 ses=1 res=
[ 37.887928] audit: type=1300 audit(1637342400.924:2): arch=c00000b7 syscall=64 success=yes exit=4 a0=8 a1=fffff569f940 a2=4 a3=ffff9a3d96b0 items=
0 pid=1 pid=596 auid=1000 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=1 comm="(systemd)" exe="/lib/systemd/systemd" key=(null)
[ 37.914545] audit: type=1327 audit(1637342400.924:2): proctitle="(systemd)"
ultra96:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 7C:C2:C6:41:8D:D0
          inet6 addr: fe80::7ec2:c6ff:fe41:8d0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:4 errors:4 dropped:0 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:184 (184.0 B) TX bytes:2611 (2.5 KiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 B) TX bytes:140 (140.0 B)

ultra96:~$ sudo ifconfig eth0 192.168.1.11
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:
#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

Password:
ultra96:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 7C:C2:C6:41:8D:D0
          inet addr:192.168.1.11 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::7ec2:c6ff:fe41:8d0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:8 errors:8 dropped:0 overruns:0 frame:0
          TX packets:27 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:368 (368.0 B) TX bytes:3956 (3.8 KiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 B) TX bytes:140 (140.0 B)

```

DTR RTS CTS CD DSR RI

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~
INFO: Checking installed development libraries
INFO: Checking network and other services
WARNING: No tftp server found - please refer to "UG1144 2022.2 PetaLinux Tools Documentation Reference Guide" for its impact and solution
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ifconfig
enp5s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 6c:02:e0:45:16:fb txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 180 bytes 14948 (14.9 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 180 bytes 14948 (14.9 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

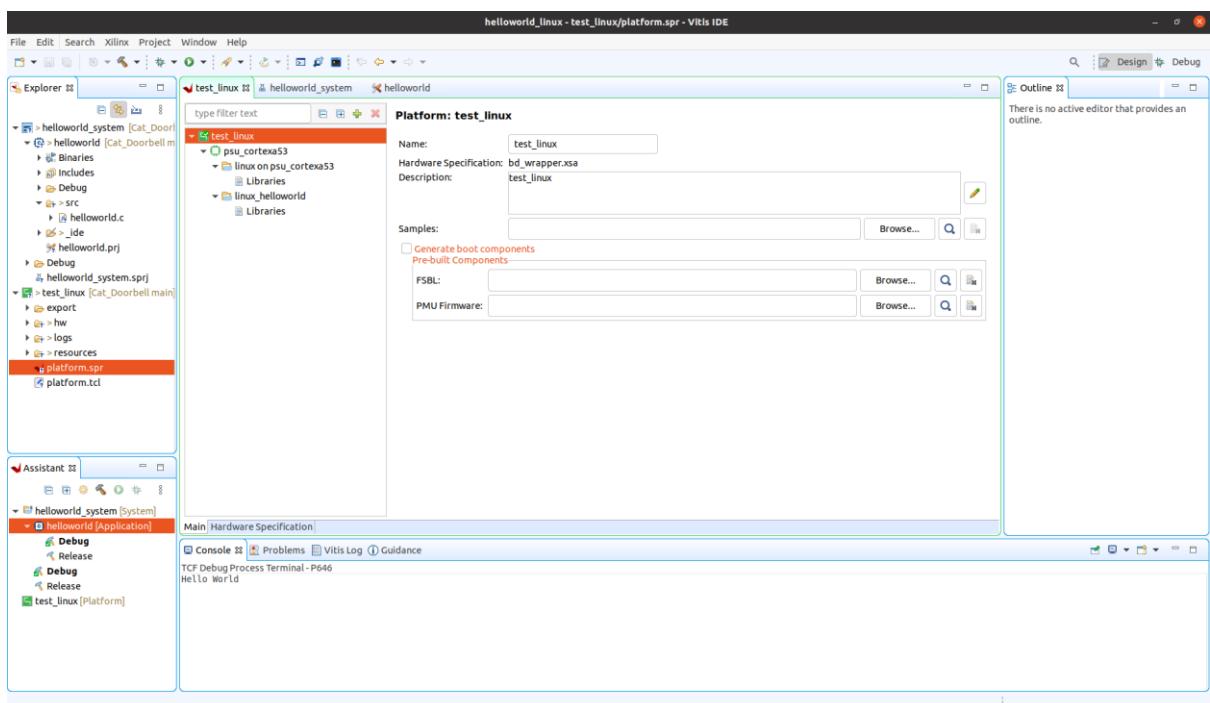
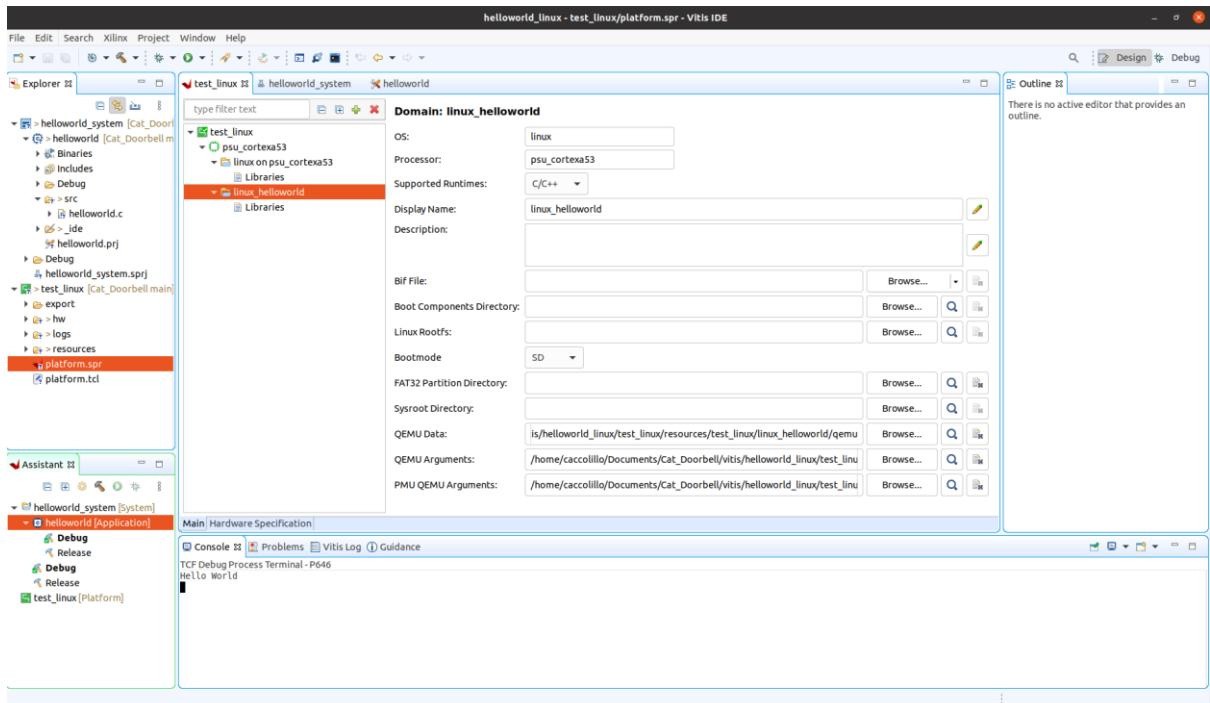
wlp4s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.149.17 netmask 255.255.255.0 broadcast 192.168.149.255
        inet6 fe80::2c71:d382:e43b:7316 prefixlen 64 scopeid 0x20<link>
            ether a8:93:4a:5f:50:dd txqueuelen 1000 (Ethernet)
            RX packets 376 bytes 365931 (365.9 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 350 bytes 40198 (40.1 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ifconfig enp5s0 192.168.1.10
SIOCSIFADDR: Operation not permitted
SIOCSIFFLAGS: Operation not permitted
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ sudo ifconfig enp5s0 192.168.1.10
[sudo] password for caccoollo:
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~$ ifconfig
enp5s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.1.10 netmask 255.255.255.0 broadcast 192.168.1.255
        ether 6c:02:e0:45:16:fb txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

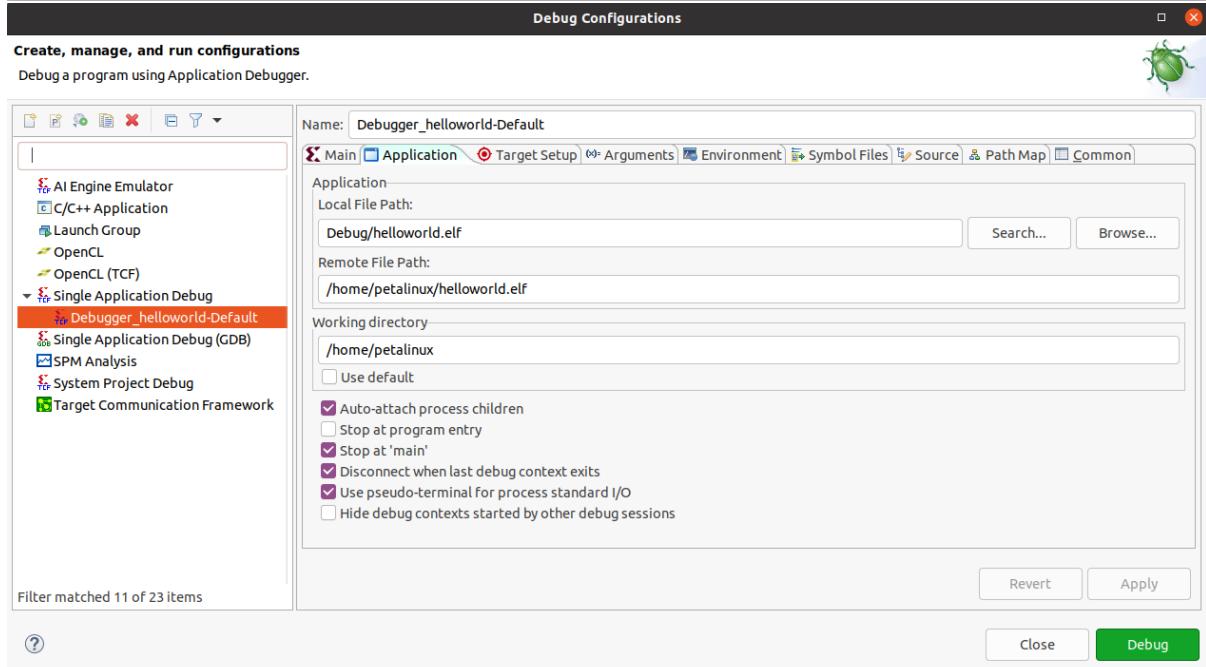
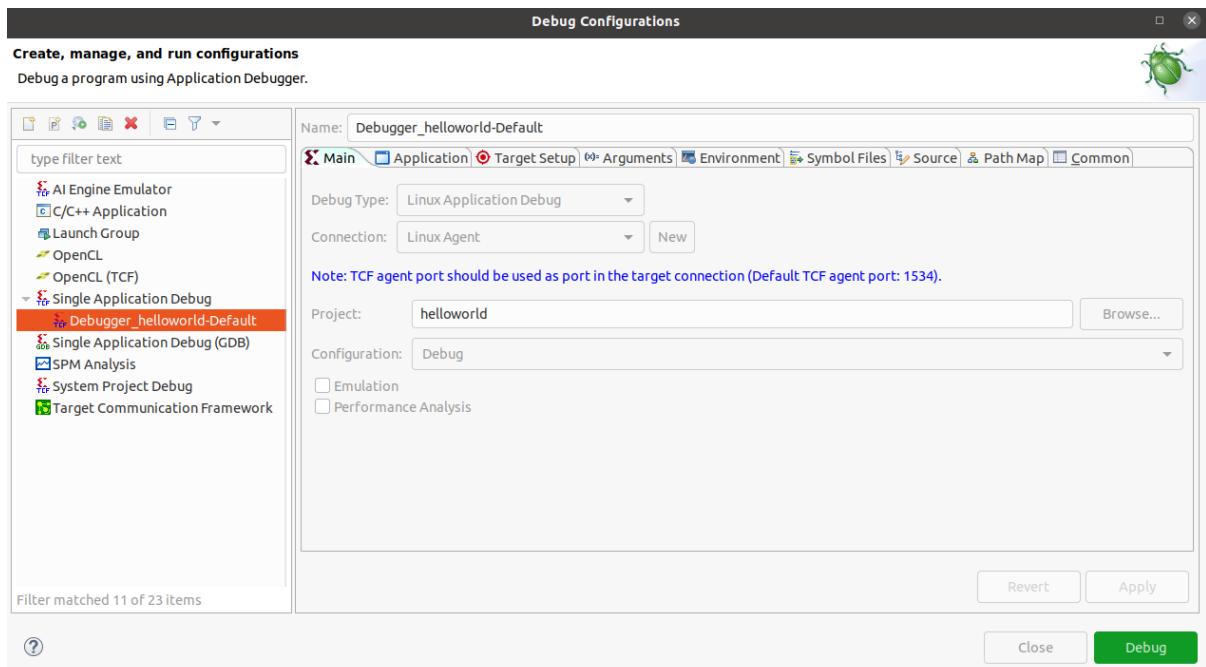
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
```

The vitis environment gets set as per the following tutorial to debug via a TCF connection on the ULTRA96 V2 board:

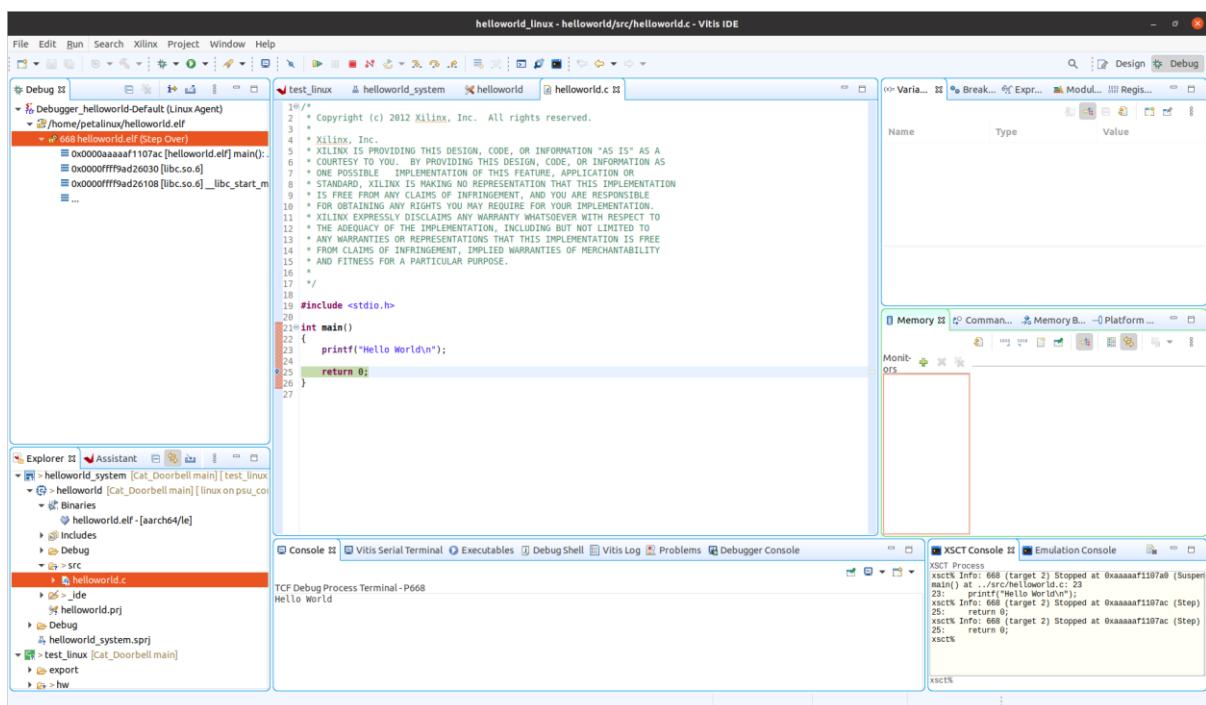
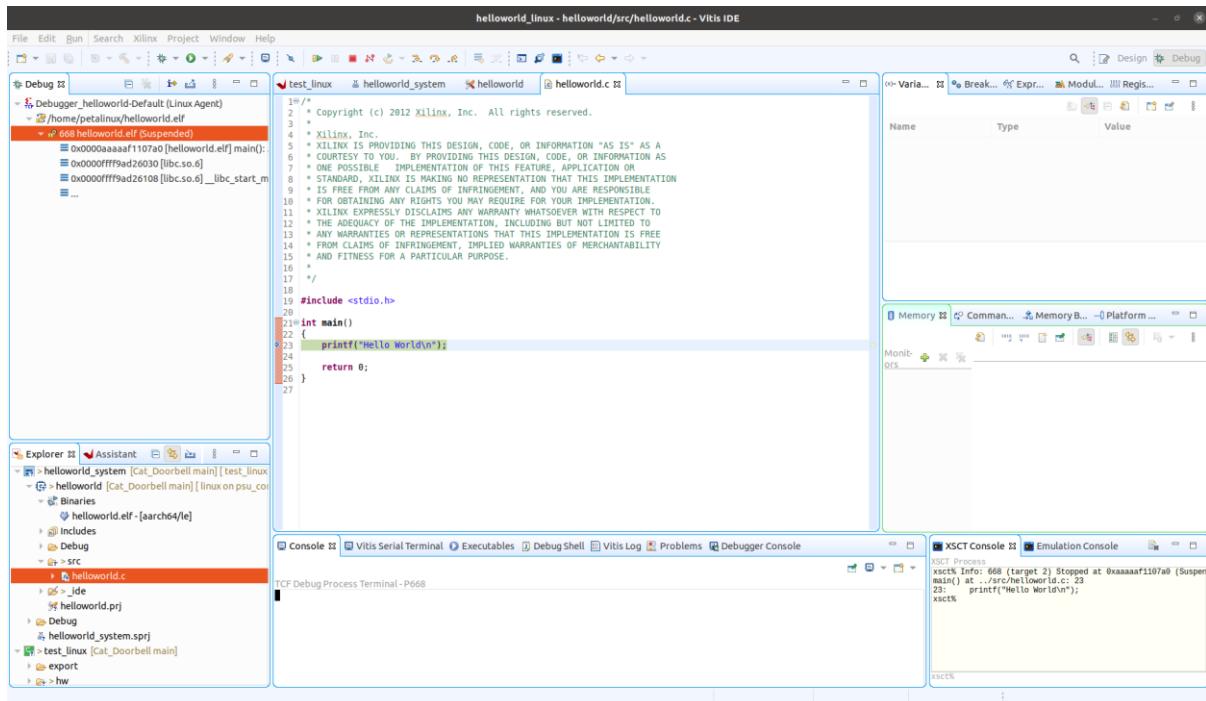
<https://xilinx.github.io/Embedded-Design-Tutorials/docs/2021.2/build/html/docs/introduction/Zynq7000-EDT/4-linux-for-zynq.html>



The debugger has been configured as follows:



once launched the debug process, the "hello world" gets written in the console window:



By looking at the Ultra96 v2 shell, the elf file gets downloaded in the folder of choice set up in the debug configuration, and can also be executed locally:

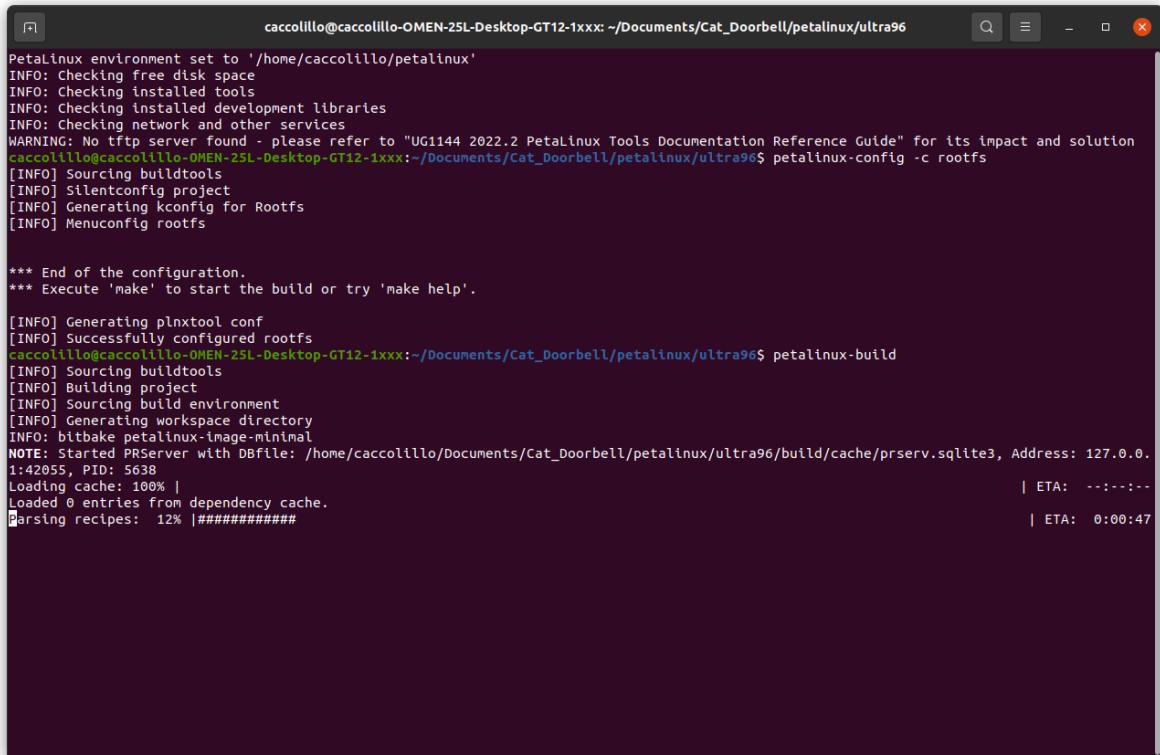
```
GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Controlsignals View Help
ultra96:~$ pwd
/home/petalinux
ultra96:~$ ls
helloworld.elf
ultra96:~$ ./helloworld.elf
Hello World
ultra96:~$ 
```

Python 3 interpreter is not installed as a default on the petalinux build from the template, so it has to be added.

To do so follow:

https://support.xilinx.com/s/question/0D52E00006ihQL7SAM/petalinux-and-python3?language=en_US

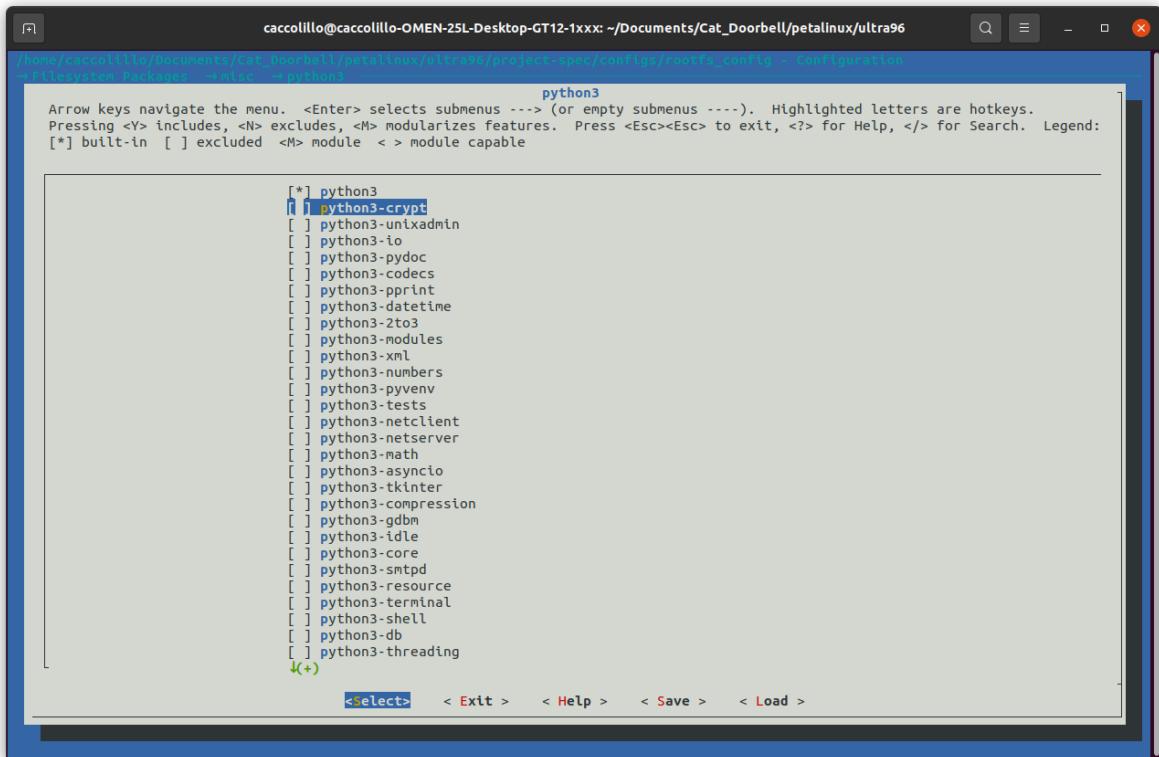
So it gets enabled as per pictures below:



```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/ultra96
PetaLinux environment set to '/home/caccolillo/petalinux'
INFO: Checking free disk space
INFO: Checking installed tools
INFO: Checking installed development libraries
INFO: Checking network and other services
WARNING: No tftp server found - please refer to "UG1144 2022.2 PetaLinux Tools Documentation Reference Guide" for its impact and solution
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/ultra96$ petalinux-config -c rootfs
[INFO] Sourcing buildtools
[INFO] Silentconfig project
[INFO] Generating kconfig for Rootfs
[INFO] Menuconfig rootfs

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

[INFO] Generating pnxtool conf
[INFO] Successfully configured rootfs
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/ultra96$ petalinux-build
[INFO] Sourcing buildtools
[INFO] Building project
[INFO] Sourcing build environment
[INFO] Generating workspace directory
INFO: bitbake petalinux-image-minimal
NOTE: Started PRServer with DBfile: /home/caccolillo/Documents/Cat_Doorbell/petalinux/ultra96/build/cache/prserv.sqlite3, Address: 127.0.0.1:42055, PID: 5638
Loading cache: 100% |
Loaded 0 entries from dependency cache.
| ETA:  ---:--:--
Parsing recipes: 12% |#####| ETA: 0:00:47
```



And on the ULTRA96 v2 console, the python interpreter can be started:

```
GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Controlsignals View Help
Starting Permit User Sessions...
Starting Target Communication Framework agent...
[ OK ] Started initedd.busybox.service.
[FAILED] Failed to start LSB: Kernel NFS server support.
See 'systemctl status nfsserver.service' for details.
[ OK ] Finished Permit User Sessions.
[ OK ] Started Target Communication Framework agent.
[ OK ] Started Getty on tty1.
[ OK ] Started Serial Getty on ttyPS0.
[ OK ] Reached target Login Prompts.
[ OK ] Reached target Multi-User System.
Starting Record Runlevel Change in UTMP...
[ OK ] Finished Record Runlevel Change in UTMP.

PetaLinux 2022.2_release_S10071807 ultra96 ttyPS0

ultra96 login: [ 12.135121] ax88179_178a 1-1.1:1.0 eth0: ax88179 - Link status is: 1
[ 12.146646] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready

ultra96 login: petalinux
You are required to change your password immediately (administrator enforced).
New password:
Retype new password:
[ 26.880491] audit: type=1006 audit(1637342389.124:2): pid=510 uid=0 old-auid=4294967295 auid=1000 tty=(none) old-ses=4294967295 ses=1 res=1
[ 26.893135] audit: type=1300 audit(1637342389.124:2): arch=c00000b7 syscall=64 success=yes exit=4 a0=8a1=fffffc99fffc30 a2=4 a3=fffffa98cd6b0 items=0 ppid=1 pid=510 auid=1000 uid=0 gid=0 euid=0 suid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=1 comm="(systemd)" exe="/lib/systemd/systemd" key=(null)
[ 26.919812] audit: type=1327 audit(1637342389.124:2): proctitle="(systemd)"

ultra96:~$ ;74R
-sh: syntax error near unexpected token `;'
ultra96:~$ p
-sh: p: command not found
ultra96:~$ python3
Python 3.9.9 (main, Nov 15 2021, 18:05:17)
[GCC 11.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

To enable the webcam, the tutorial below has been followed:

<https://www.hackster.io/whitney-knitter/using-a-usb-web-camera-with-the-minized-5783b1>

In doing so, the user-rootfsconfig has been edited as follows:

```
user-rootfsconfig
no_bsp.sh
1 #Note: Mention Each package in individual line
2 #These packages will get added into rootfs menu entry
3
4 CONFIG_gpio_demo
5 CONFIG_peekpoke
6
7 CONFIG_packagegroup-petalinux-opencv
8 CONFIG_packagegroup-petalinux-x11
9 CONFIG_packagegroup-petalinux-v4lutils|
```

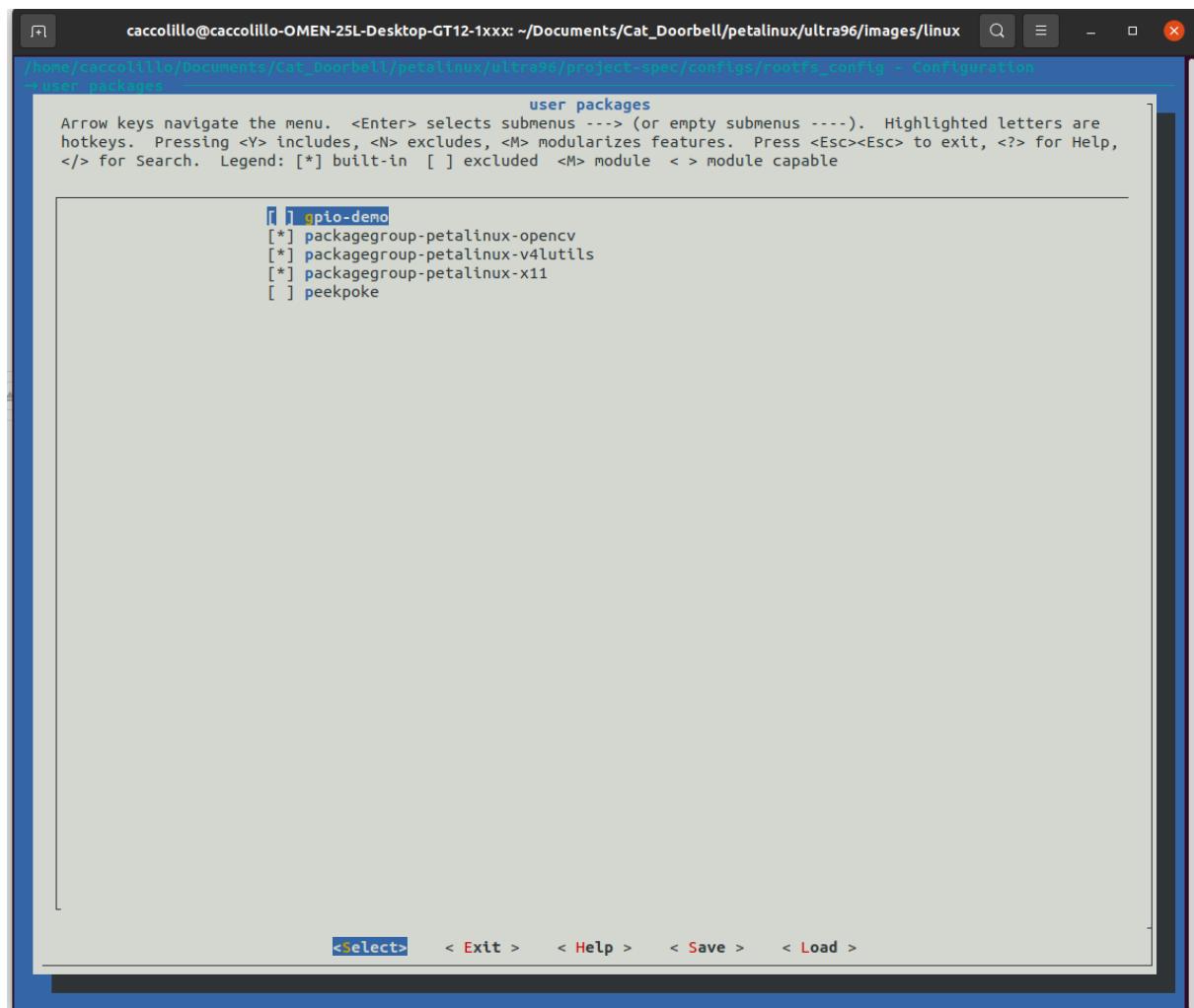
Then launch the root filesystem configuration editor GUI with “petalinux-config -c rootfs”, and then enable the:

- packagegroup-petalinux,
- packagegroup-petalinux-display-debug,
- packagegroup-petalinux-opencv,
- packagegroup-petalinux-self-hosted,
- packagegroup-petalinux-utils,
- packagegroup-petalinux-v4lutils,
- packagegroup-petalinux-x11.

Then navigate to the user packages tab to enable all of the custom options that were added to the *user-rootfsconfig* file.

Compiling the image and running the board, by issuing the following command, the USB webcam gets recognized correctly:

```
GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Control signals View Help
--verbose          turn on verbose ioctl status reporting
ultra96:~$ v4l2-ctl --list-devices
[ 232.532791] usb 1-1.2: reset high-speed USB device number 5 using xhci-hcd
C270 HD WEBCAM (usb-xhci-hcd.1.auto-1.2):
/dev/video0
/dev/video1
/dev/media0
```



Save and exit the root filesystem configuration editor then build the PetaLinux project:

petalinux-build

```

Petalinux Package Groups
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

[ packagegroup-petalinux --->
  packagegroup-petalinux-audio --->
  packagegroup-petalinux-benchmarks --->
  packagegroup-petalinux-display-debug --->
  packagegroup-petalinux-gststreamer --->
  packagegroup-petalinux-lmsensors --->
  packagegroup-petalinux-matchbox --->
  packagegroup-petalinux-mraa --->
  packagegroup-petalinux-multimedia --->
  packagegroup-petalinux-networking-debug --->
  packagegroup-petalinux-networking-stack --->
  packagegroup-petalinux-ocicontainers --->
  packagegroup-petalinux-openamp --->
  packagegroup-petalinux-opencv --->
  packagegroup-petalinux-python-modules --->
  packagegroup-petalinux-qt --->
  packagegroup-petalinux-qt-extended --->
  packagegroup-petalinux-self-hosted --->
  packagegroup-petalinux-utils --->
  packagegroup-petalinux-v4lutils --->
  packagegroup-petalinux-vitis-acceleration --->
  packagegroup-petalinux-weston --->
  packagegroup-petalinux-x11 --->
  packagegroup-petalinux-xen --->

```

<Select> < Exit > < Help > < Save > < Load >

The board gets programmed with the new image, the ETHERNET static addresses on the ULTRA96 and the HOST pc get set, via scp the python script gets transferred to the ULTRA96:

```

caccoillo@caccoillo-OMEN-2SL-Desktop-GT12-xxxx: ~/Documents/Cat_Doorbell/python
[ -S program] source ... target
caccoillo@caccoillo-OMEN-2SL-Desktop-GT12-xxxx:~/Documents/Cat_Doorbell/python$ scp ./get4frames.py petalinux@192.168.1.12:/home/petalinux
The authenticity of host '192.168.1.12' can't be established.
RSA key fingerprint is SHA256:qgFZIn/xuo0NG1WrwBirrlM8mtqP73RRjSEXTN7/B0Q.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.12' (RSA) to the list of known hosts.
petalinux@192.168.1.12's password:
get4frames.py
100% 678 598.0KB/s 00:00
caccoillo@caccoillo-OMEN-2SL-Desktop-GT12-xxxx:~/Documents/Cat_Doorbell/python$ scp petalinux@192.168.1.12:/home/petalinux/frame1.jpg

```

And then it gets executed on the board:

```

GTKTerm - /dev/ttyUSB1 115200-N-1
File Edit Log Configuration ControlSignals View Help
ultra96:~$ python3 ./get4frames.py
[ WARN:0] global /usr/src/debug/opencv/4.5.2-r0/git/modules/videoio/src/cap_gststreamer.cpp (597) isPipelinePlaying OpenCV | GStreamer warning: GStreamer: pipeline have not been created
[ 2221.268784] usb 1-1.2: reset high-speed USB device number 5 using xhci-hcd
ultra96:~$ ls
frame1.jpg frame2.jpg frame3.jpg frame4.jpg get4frames.py
ultra96:~$ 

```

After the execution, the four frames files are created, as expected, and they can be collected always via scp, and brought in the host pc, with the command:

```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/python
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/python$ scp petalinux@192.168.1.12:/home/petalinux/frame4.jpg ./
petalinux@192.168.1.12's password:
frame4.jpg
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/python$
```

Now it's time to create the Vitis "driver" C application to manage the image averaging kernel HLS IP-core.

A source of inspiration comes from:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18842018/Linux+User+Mode+Pseudo+Driver>

In order to get usable memory for the hardware accelerator, the technique of using the CMA memory shall be used:

<https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18841683/Linux+Reserved+Memory>

The system-user.dtsi has been hence changed as per below:

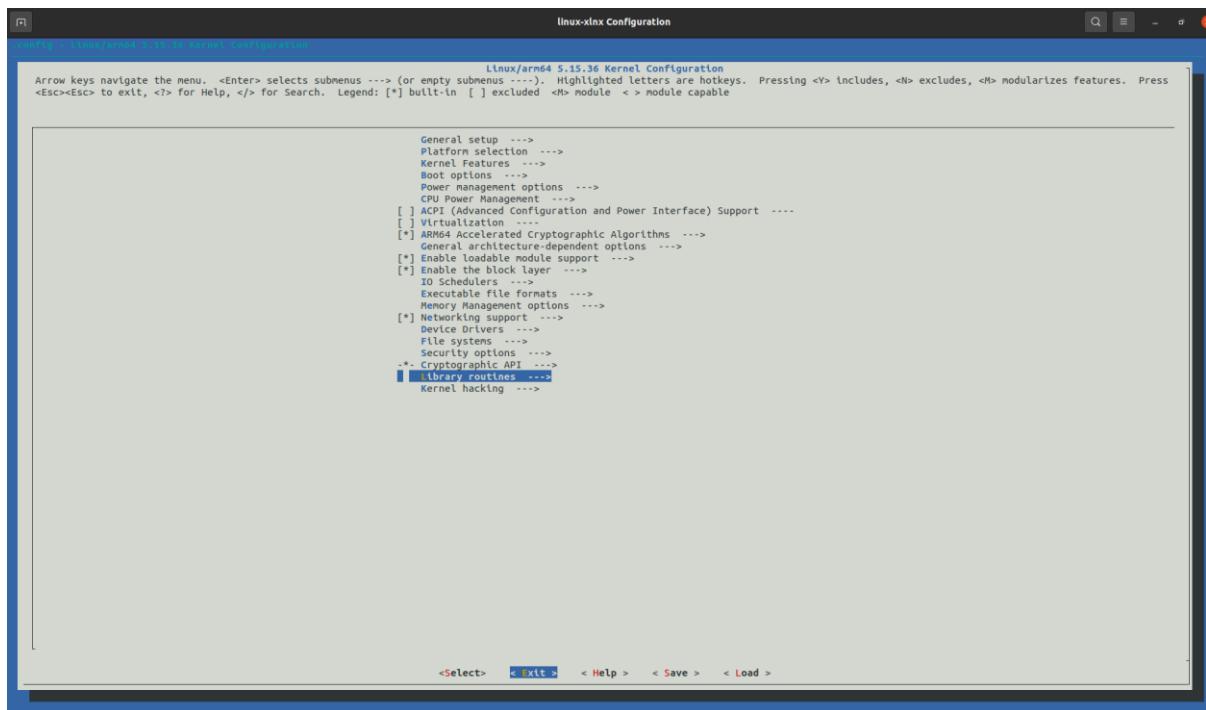
```
/include/ "system-conf.dtsi"
/ {
    reserved-memory {
        #address-cells = <2>;
        #size-cells = <2>;
        ranges;
        reserved: buffer@0 {
            compatible = "shared-dma-pool";
            reusable;
            reg = <0x0 0x70000000 0x0 0x10000000>;
            linux,cma-default;
        };
    };
};
```

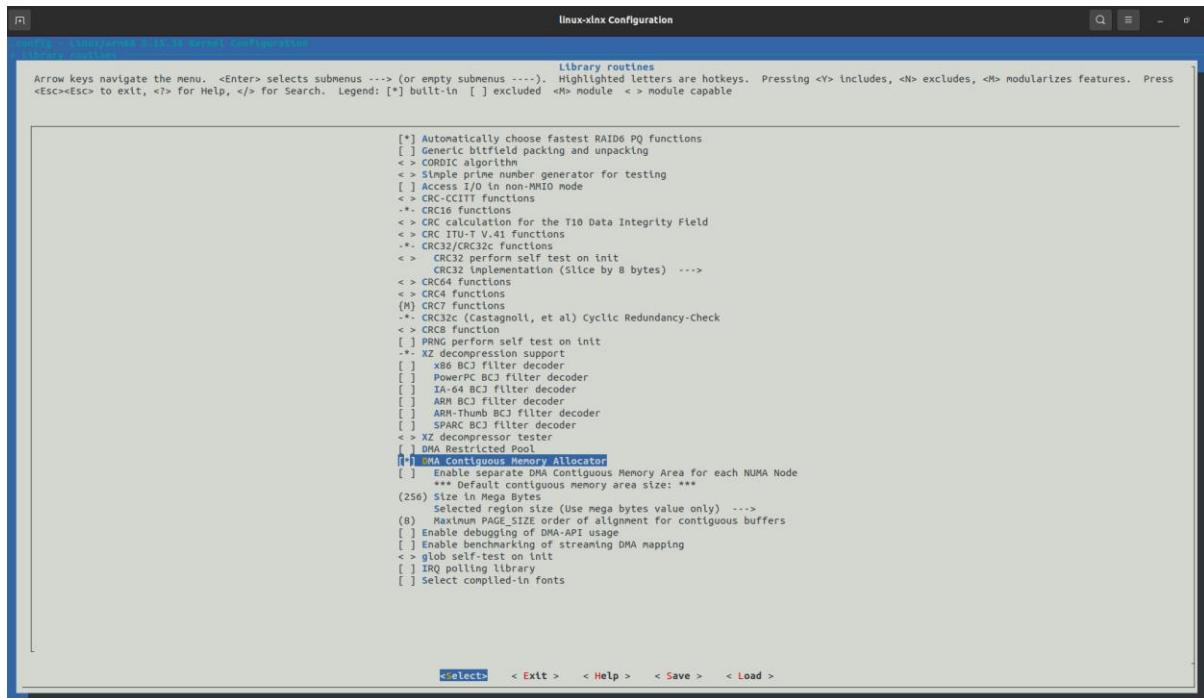
```
Open | Save | system-user.dtsi
~/Documents/Cat_Doorbell/petalinux/ultra9..ec/meta-user/recipes-bsp/device-tree/files

1 /include/ "system-conf.dtsi"
2 / {
3
4 reserved-memory {
5     #address-cells = <2>;
6     #size-cells = <2>;
7     ranges;
8
9     reserved: buffer@0 {
10        compatible = "shared-dma-pool";
11        reusable;
12        reg = <0x0 0x70000000 0x0 0x10000000>;
13        linux,cma-default;
14    };
15 };
16
17
18 };
```

And the CMA has been enabled in the kernel:

```
petalinux-config -c kernel
```





After building petalinux and flashing the image, the system boots up with a correct CMA region:

```
GTKTerm - /dev/ttyUSB1 115200-8-N-1

File Edit Log Configuration Controlsignals View Help
Type: RAMDisk Image
Compression: uncompressed
Data Start: 0x108e0f720
Data Size: 252026150 Bytes = 240.4 MiB
Architecture: AArch64
OS: Linux
Load Address: unavailable
Entry Point: unavailable
Hash algo: sha256
Hash value: ee41c42f7108ba7323af6614ccb06a76dcf0c1b3532a329872e4a6423ealc257
Verifying Hash Integrity ... sha256+ OK
## Loading fdt from FIT Image at 10000000 ...
Using 'conf-system-top.dtb' configuration
Trying 'fdt-system-top.dtb' subimage
Description: Flattened Device Tree blob
Created: 2022-10-03 7:56:07 UTC
Type: Flat Device Tree
Compression: uncompressed
Data Start: 0x108d70e4
Data Size: 38251 Bytes = 37.4 KiB
Architecture: AArch64
Hash algo: sha256
Hash value: 6006f155fec82d4fdcb4595d0fb494e4dd3a3994439de6efa2595bc252a78862
Verifying Hash Integrity ... sha256+ OK
Booting using the fdt blob at 0x108d70e4
Uncompressing Kernel Image
Loading Ramdisk to Scd01000, end 6bdffad26 ... OK
Loading Device Tree to 0000000005cd94000, end 0000000005cda056a ... OK

Starting kernel ...

[ 0.000000] Booting Linux on physical CPU 0x0000000000 [0x410fd034]
[ 0.000000] Linux version 5.15.36-xilinx-v2922.2 (oe-user@oe-host) (aarch64-xilinx-linux-gcc (GCC) 11.2.0, GNU ld (GNU Binutils) 2.37.20210721) #1 SMP Mon Oct 3 07:
50:07 UTC 2022
[ 0.000000] Machine model: xlnx_zynqmp
[ 0.000000] earlycon: cndns0 at MMIO 0x00000000ff010000 (options '115200n8')
[ 0.000000] printk: bootconsole [cndns0] enabled
[ 0.000000] efi: UEFI not found.
[ 0.000000] Reserved memory: created CMA memory pool at 0x000000000700000000, size 256 MiB
[ 0.000000] OF: reserved mem: initialized node buffer@0, compatible in shared-dma-pool
```

VITIS VISION LIBRARIES L1

Instructions about how to make work OpenCV under Vitis to be used with Vision libraries are at:

https://support.xilinx.com/s/article/Vitis-Libraries-Compiling-and-Installing-OpenCV?language=en_US

An example about how to get a working Vitis vision example is at:

<https://support.xilinx.com/s/article/000035905?language=en> US

A tutorial about how to use the L1 in Vivado, Vitis HLS and Vitis is at:

https://support.xilinx.com/s/article/000035309?language=en_US

Several interesting tutorials are at:

<https://community.element14.com/technologies/fpga-group/b/blog/posts/learning-xilinx-zynq-try-to-make-my-own-accelerated-opencv-function---1-vitis-hls>

<https://community.element14.com/technologies/fpga-group/b/blog/posts/learning-xilinx-zynq-hardware-accelerated-software>

<https://www.linkedin.com/pulse/vitis-20201-beauty-acceleration-martin-kellermann>

<https://www.xilinx.com/video/software/vitis-hls-l1-library-wizard.html>

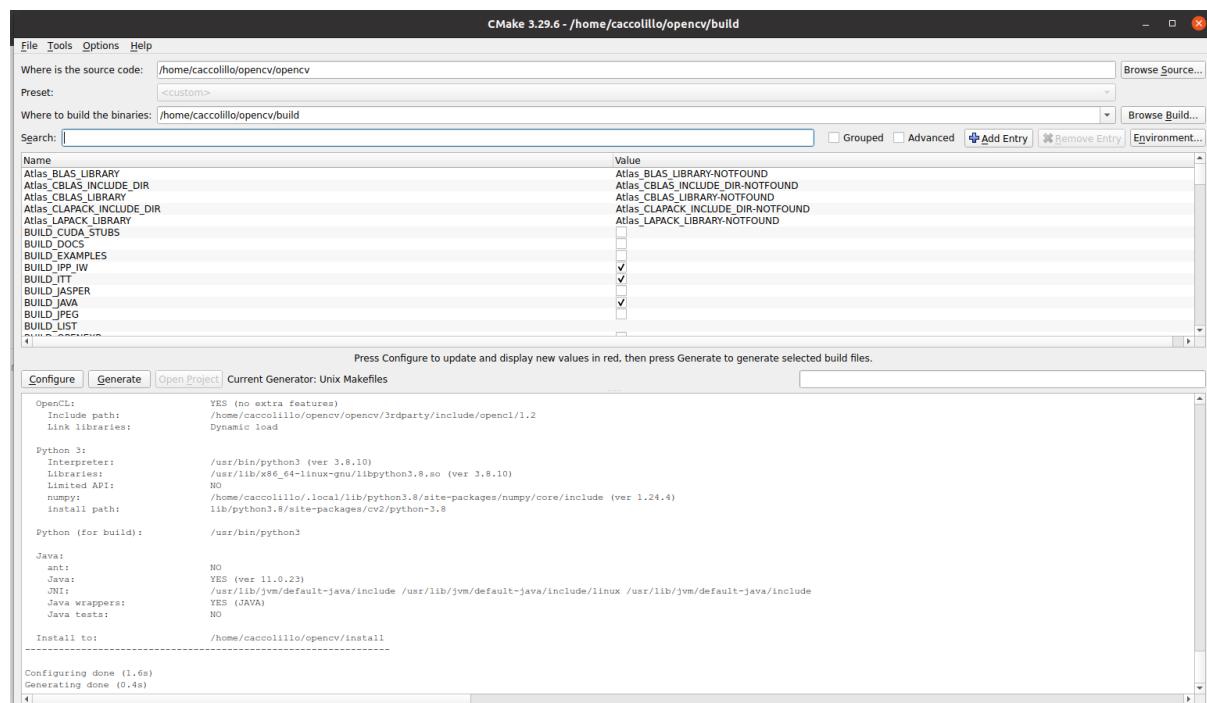
<https://docs.amd.com/r/2022.2-English/Vitis-Tutorials-Getting-Started/Step-1-Create-Vivado-Design-and-Generate-XSA>

https://github.com/Xilinx/Vitis_Libraries

To compile the OpenCV libraries, the easier way has been to follow the windows like in Linux:

https://support.xilinx.com/s/article/000035890?language=en_US

By using the GUI.

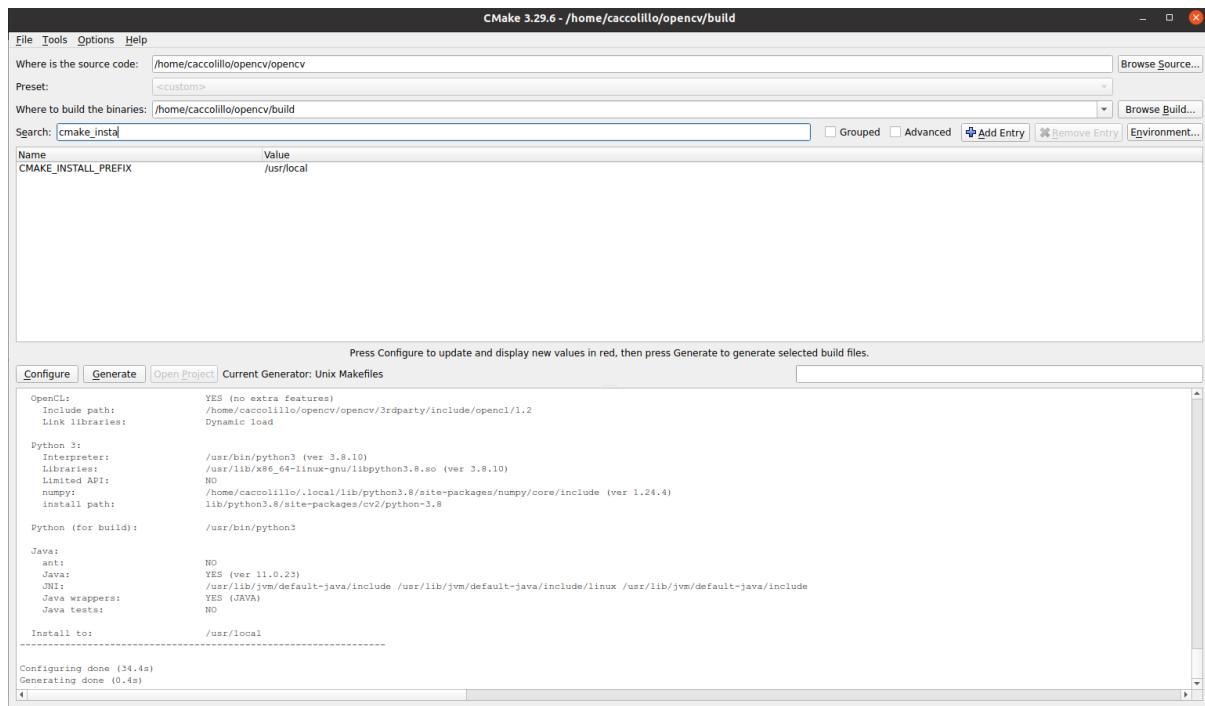


```

[100%] Built target opencv_visualisation
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibPipeline.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/frameProcessor.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/main.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/parametersController.cpp.o
[100%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/rotationConverters.cpp.o
[100%] Linking CXX executable ../../bin/opencv_interactive-calibration
[100%] Built target opencv_interactive-calibration
[100%] Building CXX object apps/version/CMakeFiles/opencv_version.dir/opencv_version.cpp.o
[100%] Linking CXX executable ../../bin/opencv_version
[100%] Built target opencv_version
[100%] Building CXX object apps/model-diagnostics/CMakeFiles/opencv_model_diagnostics.dir/model_diagnostics.cpp.o
[100%] Linking CXX executable ../../bin/opencv_model_diagnostics
[100%] Built target opencv_model_diagnostics
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx:~/opencv/build$ 

```

The libraries are installed in /usr/local as per picture below:



Once done export in ".bashrc" as per:

https://support.xilinx.com/s/article/Vitis-Libraries-Compiling-and-Installing-OpenCV?language=en_US

```

.bashrc
-
72 ;;
73 esac
74
75 # enable color support of ls and also add handy aliases
76 if [ -x /usr/bin/dircolors ]; then
77   test -r ./dircolors && eval "$(dircolors -b ~./dircolors)" || eval "$(dircolors -b)"
78   alias ls='ls --color=auto'
79   alias dir='dir --color=auto'
80   alias vdir='vdir --color=auto'
81
82   alias grep='grep --color=auto'
83   alias fgrep='fgrep --color=auto'
84   alias egrep='egrep --color=auto'
85 fi
86
87 # colored GCC warnings and errors
88 export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'
89
90 # some more ls aliases
91 alias ll='ls -alF'
92 alias la='ls -A'
93 alias l='ls -CF'
94
95 # Add an "alert" alias for long running commands.  Use like so:
96 #   sleep 10; alert
97 alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]+\s*/;s/[;&]\s*alert$/\'')"'"
98
99 # Alias definitions.
100 # You may want to put all your additions into a separate file like
101 # ~/.bash_aliases, instead of adding them here directly.
102 # See /usr/share/doc/bash-doc/examples in the bash-doc package.
103
104 if [ -f ~/.bash_aliases ]; then
105   . ~/.bash_aliases
106 fi
107
108 # enable programmable completion features (you don't need to enable
109 # this, if it's already enabled in /etc/bash.bashrc and /etc/profile
110 # sources /etc/bash.bashrc).
111 if ! shopt -q posix; then
112   if [ -f /usr/share/bash-completion/bash_completion ]; then
113     . /usr/share/bash-completion/bash_completion
114   elif [ -f /etc/bash_completion ]; then
115     . /etc/bash_completion
116   fi
117 fi
118
119
120 source /home/caccolillo/petalinux/settings.sh
121 source /home/caccolillo/Xilinx_2022_2/Vitis/2022.2/settings64.sh
122 source /home/caccolillo/Xilinx_2022_2/Vitis_HLS/2022.2/settings64.sh
123 source /home/caccolillo/Xilinx_2022_2/Vivado/2022.2/settings64.sh
124
125 export MATLAB=/home/caccolillo/matlab_2021b"
126 ifconfig enp5s0 192.168.1.10
127 PATH=$PATH:/home/caccolillo/temp/cmake-3.28.1/bin

```

The tutorial below has been followed to create the “clahe”:

<https://xilinx.github.io/Vitis-Tutorials/2022-1/build/html/docs/Getting Started/Vitis Libraries/README.html>

```

caccolillo@caccolillo-OMEN-25L-Desktop-GT12-xxxx: ~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clahe
INFO: [HLS 200-112] Total CPU user time: 12.44 seconds. Total CPU system time: 0.84 seconds. Total elapsed time: 24.62 seconds; peak allocated memory: 754.879 MB.
INFO: [Common 17-206] Exiting vitis_hls at Tue Jun 25 21:23:17 2024...
make: *** [Makefile:250: runhls] Error 1
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-xxxx:~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clahe$ make run CSIM=0 CSYNTH=1 COSIM=0 XPART=xczu3eg-svva484-1-e
XPARt is directly set to xczu3eg-svva484-1-e
Configured: settings.tcl
-----
set XPART xczu3eg-svva484-1-e
set CSIM 0
set CSYNTH 1
set COSIM 0
set VIVADO_SYN 0
set VIVADO_IMPL 0
set XF_PROJ_ROOT "/home/caccolillo/vitis_libraries/Vitis_Libraries-2022.2/vision"
set OPENCV_INCLUDE "/home/caccolillo/opencv/install/include/opencv4"
set OPENCV_LIB "/home/caccolillo/opencv/install/lib"
set CUR_DIR "/home/caccolillo/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clahe"
-----
vitis_hls -f run_hls.tcl;
***** Vitis HLS - High-Level Synthesis from C, C++ and OpenCL v2022.2 (64-bit)
**** SW Build 3670227 on Oct 13 2022
**** IP Build 3669848 on Fri Oct 14 08:30:02 MDT 2022

```

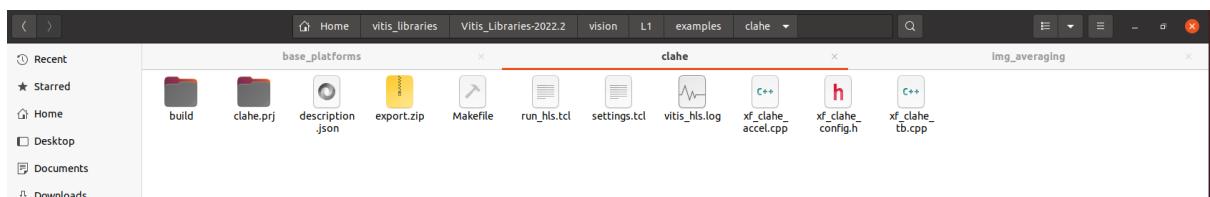
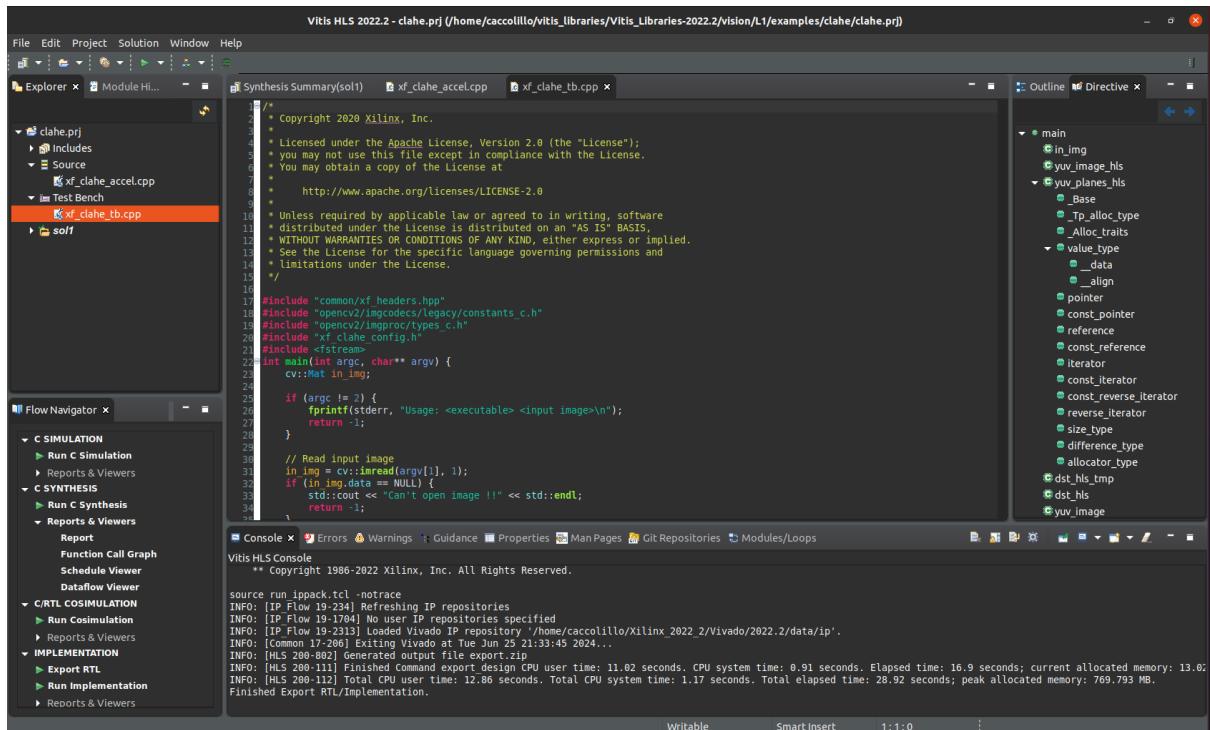
Opening the vitis_hls gui end exporting the IP, a zip file gets created:

```

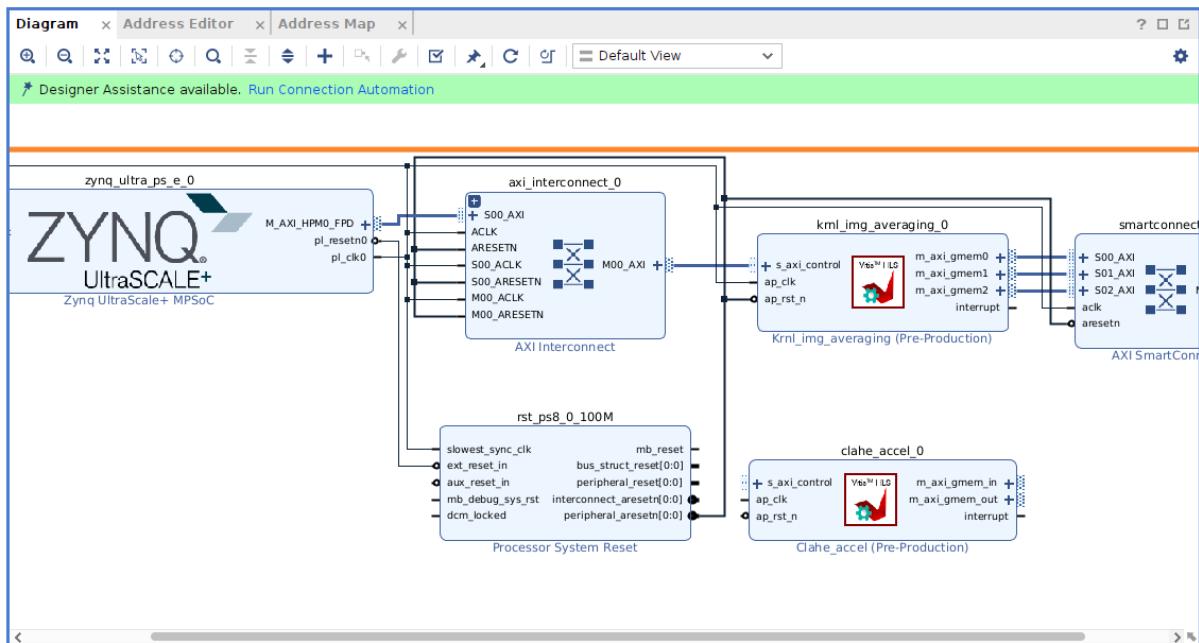
caccollillo@caccollillo-OMEN-25L-Desktop-GT12-1xxx: ~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clah
INFO: [HLS 200-111] Finished Updating report files: CPU user time: 2.48 seconds. CPU system time: 0.02 seconds. Elapsed time: 2.5 seconds; current allocated memory: 1.442 GB.
INFO: [VHDL 208-304] Generating VHDL RTL for clah_accel.
INFO: [VLOG 209-307] Generating Verilog RTL for clah_accel.
INFO: [HLS 200-789] **** Estimated Fmax: 218.62 MHz
INFO: [HLS 200-111] Finished Command csynth_design CPU user time: 41.01 seconds. CPU system time: 1.78 seconds. Elapsed time: 43.7 seconds; current allocated memory: 722.504 MB
.
INFO: [HLS 200-112] Total CPU user time: 42.69 seconds. Total CPU system time: 2.01 seconds. Total elapsed time: 55.54 seconds; peak allocated memory: 1.443 GB.
INFO: [Common 19-206] Exiting vitis_hls at Tue Jun 25 21:25:19 2024...
caccollillo@caccollillo-OMEN-25L-Desktop-GT12-1xxx: ~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clah-
caccollillo@caccollillo-OMEN-25L-Desktop-GT12-1xxx: ~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clah_vitis_hls -p clah.eprj &
[1] 5261
caccollillo@caccollillo-OMEN-25L-Desktop-GT12-1xxx: ~/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clah-
***** Vitis HLS - High-Level Synthesis from C, C++ and OpenCL v2022.2 (64-bit)
**** SW Build 3670227 on Oct 13 2022
**** IP Build 3669848 on Fri Oct 14 08:30:02 MDT 2022
** Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.

source /home/caccollillo/Xlllnx_2022_2/Vitis_HLS/2022.2/scripts/vitis_hls/hls.tcl -notrace
INFO: [HLS 200-10] Running '/home/caccollillo/Xlllnx_2022_2/Vitis_HLS/2022.2/bin/unwrapped/lnx64.o/vitis_hls'
INFO: [HLS 200-10] For user 'caccollillo' on host 'caccollillo-OMEN-25L-Desktop-GT12-1xxx' (linux_x86_64 version 5.11.0-27-generic) on Tue Jun 25 21:32:18 BST 2024
INFO: [HLS 200-10] On os Ubuntu 20.04.3 LTS
INFO: [HLS 200-10] In directory '/home/caccollillo/vitis_libraries/Vitis_Libraries-2022.2/vision/L1/examples/clah'
INFO: [HLS 200-10] Bringing up Vitis HLS GUI ...

```



By adding the repo in the main vivado project, the IP can be imported and added to the block design:



By trying to run the testbench in Vitis HLS, the tool complains about OpenCV, to sort this out, the following ar# must be followed:

https://support.xilinx.com/s/article/75727?language=en_US

VITIS CLAHE SW implementation

The CLAHE algorithm pure sw implementation has been attempted in Vitis.

Tutorials about how to use OpenCV in vitis are available at:

<https://beetlebox.org/getting-started-with-computer-vision-for-vitis-embedded-systems-part-3-using-opencv-and-file-transfers/>

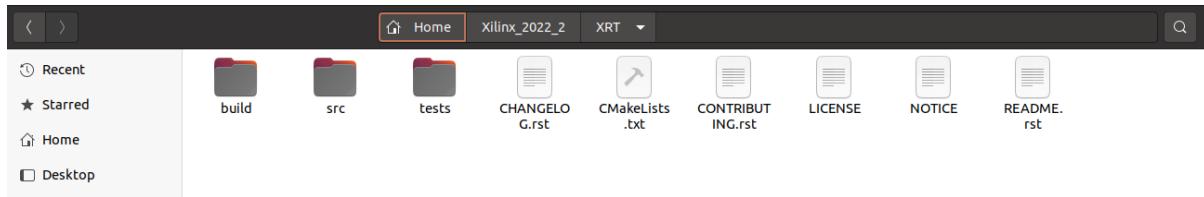
<https://beetlebox.org/getting-started-with-computer-vision-for-vitis-embedded-systems-part-4-vitis-vision-library/>

XRT has been downloaded and installed as per:

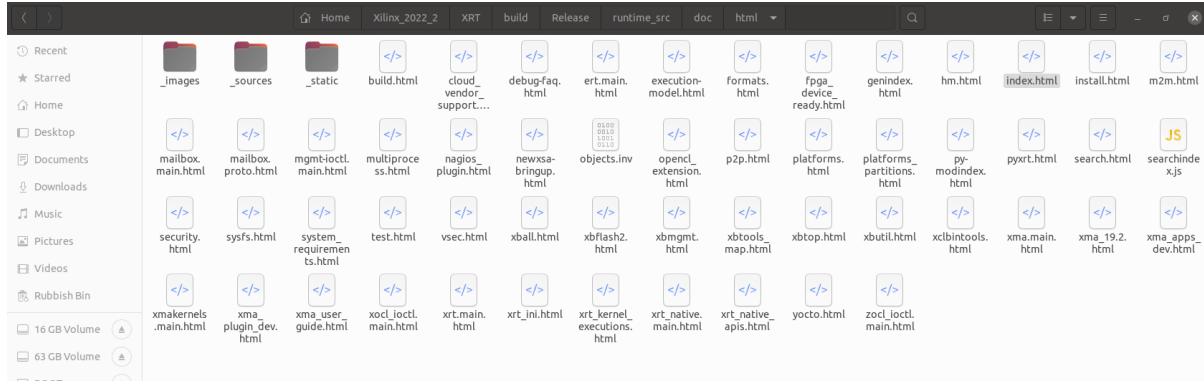
<https://xilinx.github.io/XRT/master/html/test.html>

<https://github.com/Xilinx/XRT/tree/2022.2?tab=readme-ov-file>

In the folder:



At the end, the documentation is in:



PETALINUX do_image_cpio: Function Failed

If during petalinux build this error appears, the solution is at:

https://docs.amd.com/r/2021.2-English/ug1144-petalinux-tools-reference-guide/do_image_cpio-Function-Failed

PetaLinux Tools Documentation: Reference Guide UG1144 | 2021-10-27 | 2021.2 English

Search in document

Keywords

do_image_cpio: Function Failed

CPIO format does not support sizes greater than 2 GB. Therefore, you cannot use INITRAMFS for larger sizes. The following steps describes the process for larger image sizes (greater than 2 GB).

1. Change the root file system type to EXT4 (SD/eMMC/SATA/USB).


```
$ petalinux-config
```

 Select **Image Packaging Configuration > Root filesystem type > EXT4 (SD/eMMC /SATA/USB)**.
2. Add the following lines in the <plnx-proj-root>/project-spec/meta-user/conf/petalinuxbsp.conf.


```
IMAGE_FSTYPES_remove = "cpio cpio.gz cpio.bz2 cpio.xz cpio.lzma cpio.lz4
cpio.gz.u-boot"
IMAGE_FSTYPES_DEBUGFS_remove = "cpio cpio.gz cpio.bz2 cpio.xz cpio.lzma
cpio.lz4
cpio.gz.u-boot"
```
3. Build the project.


```
$ petalinux-build
```

Note: Unlike earlier, currently PetaLinux does not generate the global DTS file. Use the following command to generate the global DTS file:

```
dts -I dtb -O dts -o system.dts system.dtb
```

CAUTION! Do not use the symlinked path to the project directories for any build operations, including simply "cd"ing into the directory.

ADD OPENCV AND OTHER LIBRARIES TO VITIS

In order to use openCV, XRT and other ROOTFS components, the following tutorial has been followed:

https://support.xilinx.com/s/article/1175461?language=en_US

https://support.xilinx.com/s/article/69166?language=en_US

https://support.xilinx.com/s/article/1141772?language=en_US

https://support.xilinx.com/s/article/1138750?language=en_US

https://support.xilinx.com/s/article/1138667?language=en_US

After having created SDK, we get a confirmation about openCV:

```

find: `opencv' : No such file or directory
caccoollo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/accl/images/linux$ find . -name "*opencv*"
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xfeatures2d.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_stereo.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xphoto.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_face.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_bioinspired.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_hfs.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_objdetect.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ccalib.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_line_descriptor.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ts.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_rgbd.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_imgproc.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_superres.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_rapid.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_core.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_dnn_objdetect.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_wechat_grcode.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_dnn_superres.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_photo.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_alphaamat.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_reg.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_videoio.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ccalib.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_imgcodecs.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ml.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xphoto.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_video.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xobjdetect.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_objdetect.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_surface_matching.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_structured_light.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_optflow.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_highgui.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ximgproc.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_datasets.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_fuzzy.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_img_hash.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_dnn.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_img_hash.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_intensity_transform.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_plot.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_features2d.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_surface_matching.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_plot.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_aruco.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xfeatures2d.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_xobjdetect.so.4.5.2
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_line_descriptor.so
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_flann.so.4.5
./sdk/sysroots/cortexa72-cortexa53-xilinx-linux/usr/lib/libopencv_ts.so

```

END PETALINUX CONFIGURATION SUPPORTING OPENCV, XRT AND ROOT FILESYSTEM ON MICROSOD

petalinux-config:

-set stdin stdout uart to psu_uart_1

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q E - □ ×  
/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/co  
n  
          misc/config System Configuration  
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty  
submenus ----). Highlighted letters are hotkeys. Pressing <Y>  
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to  
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]  
-*- ZYNQMP Configuration  
    Linux Components Selection --->  
    Auto Config Settings --->  
-*- Subsystem AUTO Hardware Settings --->  
    DTG Settings --->  
    PMUFW Configuration --->  
    FSBL Configuration --->  
    ARM Trusted Firmware Configuration --->  
    FPGA Manager --->  
    u-boot Configuration --->  
↓(+)  
<Select>   < Exit >   < Help >   < Save >   < Load >
```

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q E - □ ×  
/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/co  
n→ Subsystem AUTO Hardware Settings  
          Subsystem AUTO Hardware Settings  
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty  
submenus ----). Highlighted letters are hotkeys. Pressing <Y>  
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to  
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]  
--- Subsystem AUTO Hardware Settings  
    System Processor (psu_cortexa53_0) --->  
    Memory Settings --->  
    Serial Settings --->  
    Ethernet Settings --->  
    Flash Settings --->  
    SD/SDIO Settings --->  
    RTC Settings --->  
  
<Select>   < Exit >   < Help >   < Save >   < Load >
```

caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... /home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/cn0-> Subsystem AUTO Hardware Settings → Serial Settings →

Serial Settings

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in []

| **PMUFW Serial stdin/stdout (psu_uart_1) --->**
| FSBL Serial stdin/stdout (psu_uart_1) --->
| ATF Serial stdin/stdout (psu_uart_1) --->
| DTG Serial stdin/stdout (psu_uart_1) --->
| System stdin/stdout baudrate for psu_uart_1 (115200) --->
| System stdin/stdout baudrate for psu_uart_0 (115200) --->

<Select> < Exit > < Help > < Save > < Load >

petalinux-config -c rootfs

Enable python

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q - o
```

/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/ro

Configuration

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in []

| **Filesystem Packages** --->

- Petalinux Package Groups --->
- Image Features ---->
- user packages ---->
- PetaLinux RootFS Settings --->

<**Select**> < Exit > < Help > < Save > < Load >

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q - o
```

/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/ro

o->Filesystem Packages → misc

misc

Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in []

| **alsa-state** --->

- alsa-utils-scripts --->
- apache2 --->
- at-spi2-atk --->
- at-spi2-core --->
- babeltrace --->
- blktool --->
- blktrace --->
- ca-certificates --->
- chrpath --->

↓(+)

<**Select**> < Exit > < Help > < Save > < Load >

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q E M - □ X
```

```
/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/ro  
o->Filesystem Packages → misc
```

```
      misc
```

```
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty  
submenus ----). Highlighted letters are hotkeys. Pressing <Y>  
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to  
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]
```

```
↑(-) packagegroup-core-x11-base    --->  
     packagegroup-core-x11-xserver   --->  
     packagegroup-self-hosted     --->  
     perf    --->  
     pixman   --->  
     powertop   --->  
     ptest-runner   --->  
     python3   --->  
     python3-async   ---->  
     python3-git    --->  
↓(+)
```

```
<Select> < Exit > < Help > < Save > < Load >
```

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Docu... Q E M - □ X
```

```
/home/caccoollo/Documents/Cat_Doorbell/petalinux/accl/project-spec/configs/ro  
o->Filesystem Packages → misc → python3
```

```
      python3
```

```
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty  
submenus ----). Highlighted letters are hotkeys. Pressing <Y>  
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to  
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]
```

```
[*] python3  
[ ] python3-crypt  
[ ] python3-unixadmin  
[ ] python3-io  
[ ] python3-pydoc  
[ ] python3-codecs  
[ ] python3-pprint  
[ ] python3-datetime  
[ ] python3-2to3  
[ ] python3-modules  
↓(+)
```

```
<Select> < Exit > < Help > < Save > < Load >
```

Enable webcam and OpenCV

```
caccoillo@caccoillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/acel
/home/caccoillo/Desktop/Cat_Doorbell/petalinux/acel/project/conf/configs/configs.conf -- Configuration
metaLinux_Package_Groups->packagegroup-petalinux

    packagegroup-petalinux
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

    [+] packagegroup-petalinux
    [ ] packagegroup-petalinux-dbg
    [ ] packagegroup-petalinux-dev

<Select> < Exit > < Help > < Save > < Load >
```

```
caccoillo@caccoillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/acel
/home/caccoillo/Desktop/Cat_Doorbell/petalinux/acel/project/conf/configs/configs.conf -- Configuration
metaLinux_Package_Groups->packagegroup-petalinux-display-debug

    packagegroup-petalinux-display-debug
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

    [+] packagegroup-petalinux-display-debug
    [ ] packagegroup-petalinux-display-debug-dbg
    [ ] packagegroup-petalinux-display-debug-dev

<Select> < Exit > < Help > < Save > < Load >
```

```
caccoillo@caccoillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/acel
/home/caccoillo/Desktop/Cat_Doorbell/petalinux/acel/project/conf/configs/configs.conf -- Configuration
metaLinux_Package_Groups->packagegroup-petalinux-opencv

    packagegroup-petalinux-opencv
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

    [+] packagegroup-petalinux-opencv
    [ ] packagegroup-petalinux-opencv-dev
    [ ] packagegroup-petalinux-opencv-dbg

<Select> < Exit > < Help > < Save > < Load >
```

```
caccoillo@caccoillo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/acel
/home/caccoillo/Desktop/Cat_Doorbell/petalinux/acel/project/conf/configs/configs.conf -- Configuration
metaLinux_Package_Groups->packagegroup-petalinux-self-hosted

    packagegroup-petalinux-self-hosted
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

    [+] packagegroup-petalinux-self-hosted
    [ ] packagegroup-petalinux-self-hosted-dbg
    [ ] packagegroup-petalinux-self-hosted-dev

<Select> < Exit > < Help > < Save > < Load >
```

```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/accl
```

```
http://www.silabs.com/petalinux/configure?&file=~/Documents/Cat_Doorbell/petalinux/accl
```

```
packagegroup-petalinux-utils
```

```
Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
```

```
[*] packagegroup-petalinux-utils
[ ] packagegroup-petalinux-utils-dbg
[ ] packagegroup-petalinux-utils-dev
```

```
<Select> < Exit > < Help > < Save > < Load >
```

```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/accl
```

```
http://www.silabs.com/petalinux/configure?&file=~/Documents/Cat_Doorbell/petalinux/accl
```

```
packagegroup-petalinux-v4lutils
```

```
Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
```

```
[*] packagegroup-petalinux-v4lutils
[ ] packagegroup-petalinux-v4lutils-dbg
[ ] packagegroup-petalinux-v4lutils-dev
```

```
<Select> < Exit > < Help > < Save > < Load >
```

```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/accl
```

```
http://www.silabs.com/petalinux/configure?&file=~/Documents/Cat_Doorbell/petalinux/accl
```

```
packagegroup-petalinux-x11
```

```
Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
```

```
[*] packagegroup-petalinux-x11
[ ] packagegroup-petalinux-x11-dev
[ ] packagegroup-petalinux-x11-dbg
```

```
<Select> < Exit > < Help > < Save > < Load >
```

```
caccolillo@caccolillo-OMEN-25L-Desktop-GT12-1xxx:~/Documents/Cat_Doorbell/petalinux/accl
```

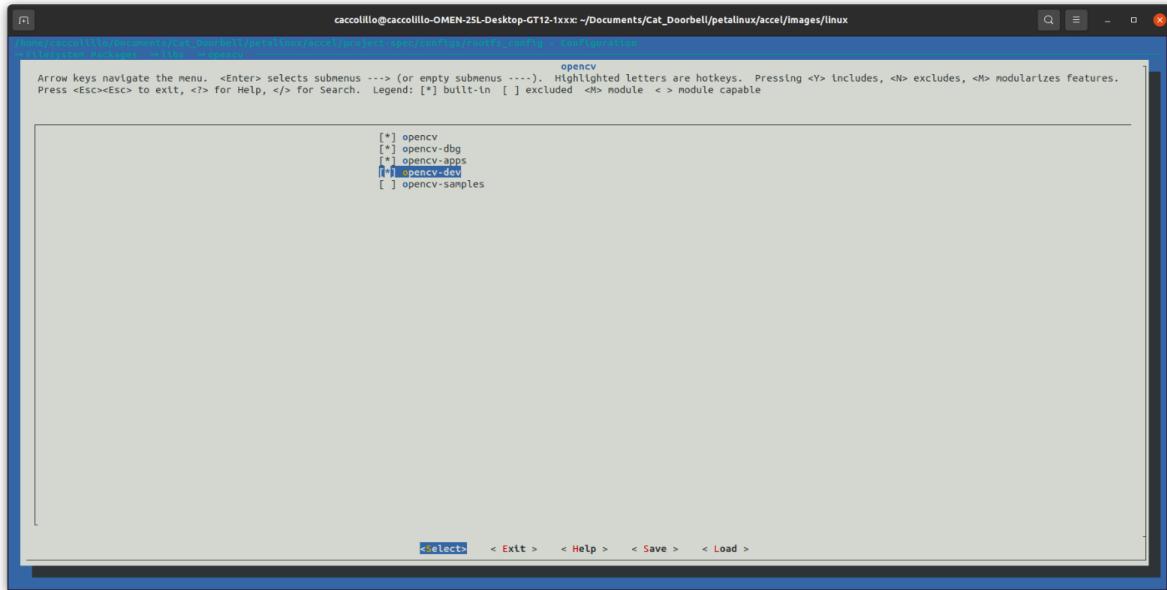
```
http://www.silabs.com/petalinux/configure?&file=~/Documents/Cat_Doorbell/petalinux/accl
```

```
Packagedgroup-petalinux-opencv
```

```
Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
```

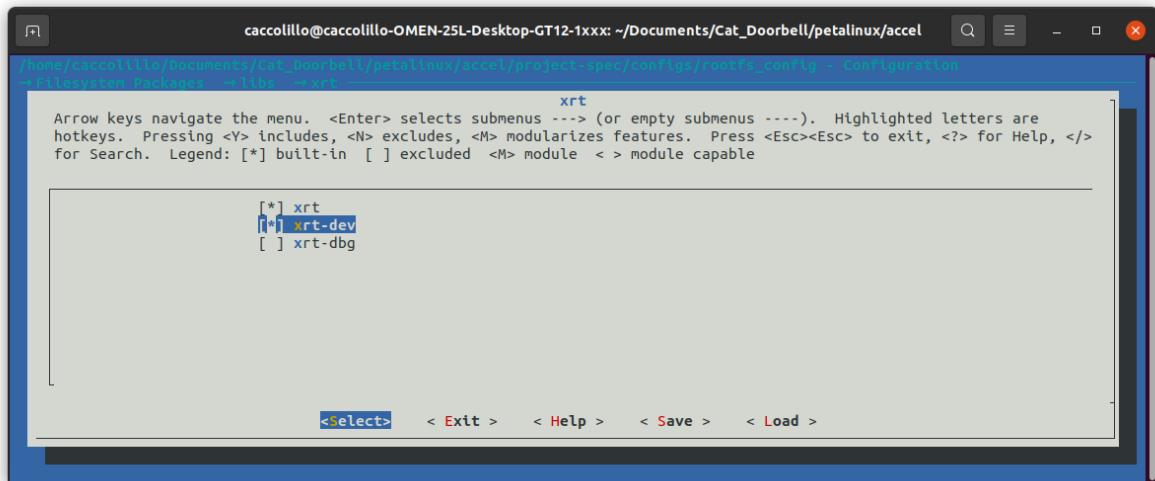
```
[*] packagegroup-petalinux-opencv
[*] packagegroup-petalinux-opencv-dev
[ ] packagegroup-petalinux-opencv-dbg
```

```
<Select> < Exit > < Help > < Save > < Load >
```



Enable XRT and OpenCL

petalinux-config -c rootfs



```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/accel
/home/caccoollo/Documents/cat_doorbell/petalinux/accel/project-spec/configs/rootfs_config -- Configuration
→ filesystem Packages → libs → zocl

zocl
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

[+] zocl
[ ] zocl-dev
[ ] zocl-dbg

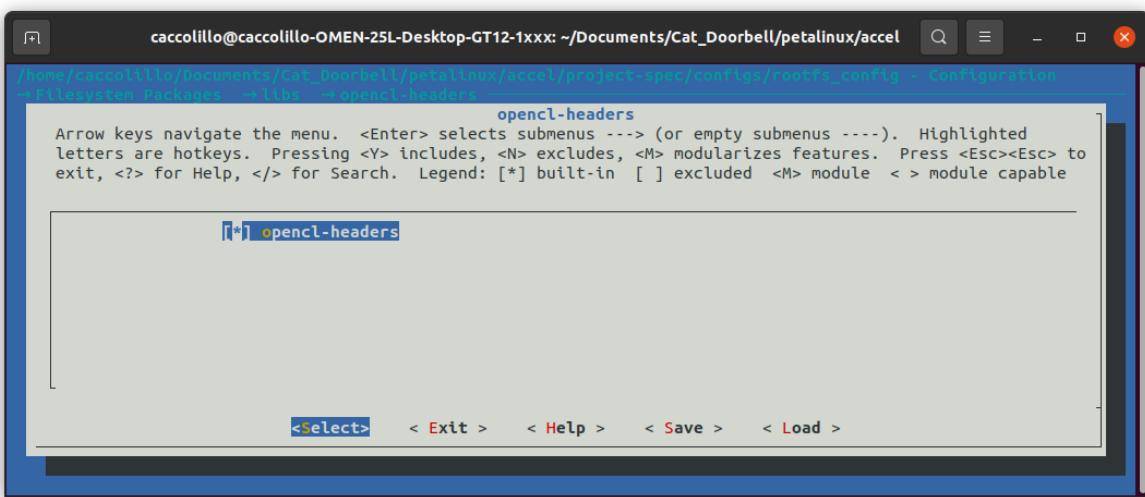
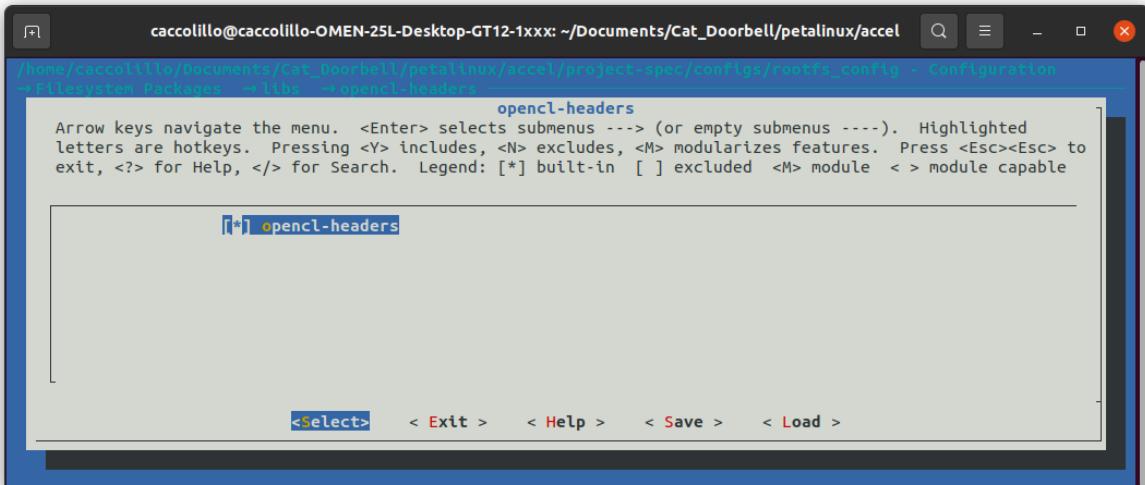
<Select> < Exit > < Help > < Save > < Load >
```

```
caccoollo@caccoollo-OMEN-25L-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/petalinux/accel
/home/caccoollo/Documents/cat_doorbell/petalinux/accel/project-spec/configs/rootfs_config -- Configuration
→ filesystem Packages → libs → opencl-clhpp

opencl-clhpp
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

[+] opencl-clhpp-dev

<Select> < Exit > < Help > < Save > < Load >
```



Change the *system-user.dtsi* as below:

```
/include/ "system-conf.dtsi"
{
chosen {
    bootargs = "console=ttyPS0,115200 root=/dev/mmcblk0p2 rw earlyprintk
rootfstype=ext4 rootwait uio_pdrv_genirq.of_id=generic-uio devtmpfs.mount=1
earlycon";
};

&axi_intc_0 {
    xlnx,kind-of-intr = <0x0>;
    xlnx,num-intr-inputs = <0x20>;
};

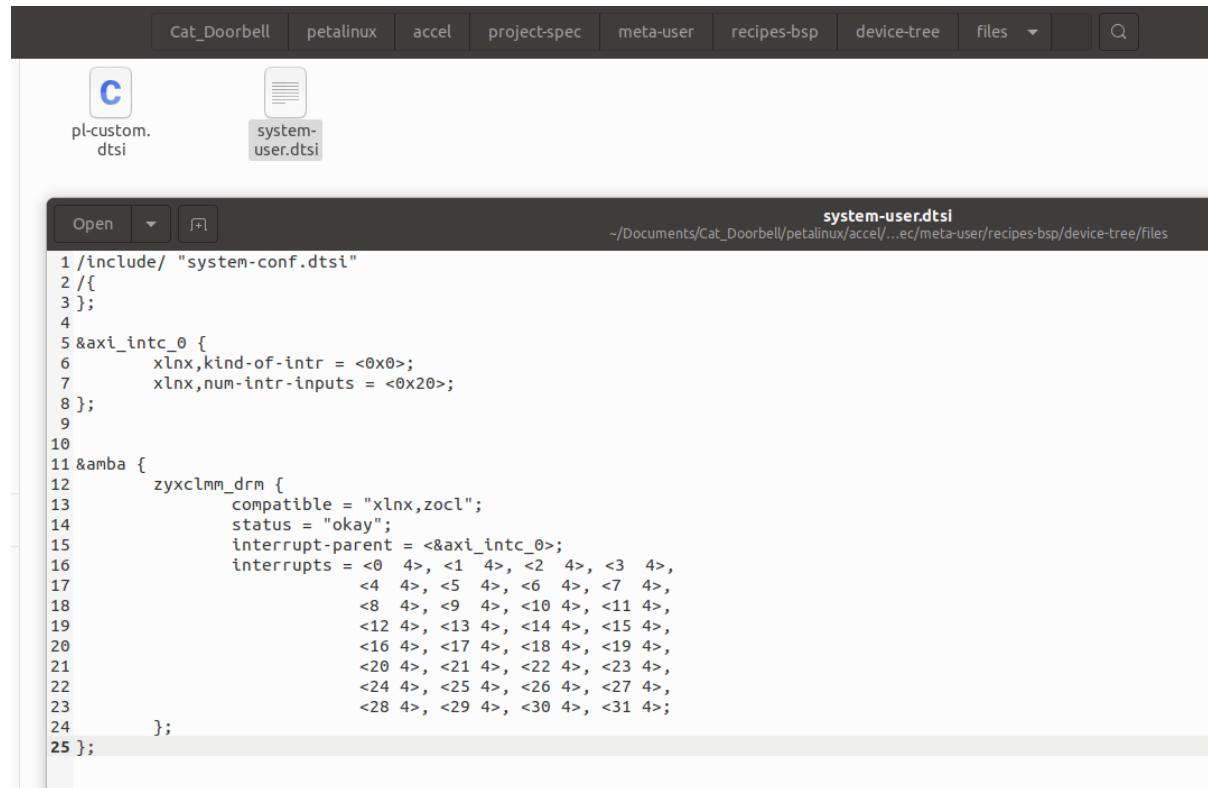
&amba {
    zyxclmm_drm {
        compatible = "xlnx,zocl";
```

```

status = "okay";
interrupt-parent = <&axi_intc_0>;
interrupts = <0 4>, <1 4>, <2 4>, <3 4>,
            <4 4>, <5 4>, <6 4>, <7 4>,
            <8 4>, <9 4>, <10 4>, <11 4>,
            <12 4>, <13 4>, <14 4>, <15 4>,
            <16 4>, <17 4>, <18 4>, <19 4>,
            <20 4>, <21 4>, <22 4>, <23 4>,
            <24 4>, <25 4>, <26 4>, <27 4>,
            <28 4>, <29 4>, <30 4>, <31 4>;
};

};


```



The screenshot shows a file manager interface with a dark header bar containing navigation links: Cat_Doorbell, petalinux, accel, project-spec, meta-user, recipes-bsp, device-tree, files, and a search icon. Below the header, there are two icons: 'pl-custom.dtsi' (with a blue 'C' icon) and 'system-user.dtsi' (with a document icon). The 'system-user.dtsi' file is highlighted. A code editor window is open, showing the contents of the 'system-user.dtsi' file. The file path is indicated as ~/Documents/Cat_Doorbell/petalinux/accel/...ec/meta-user/recipes-bsp/device-tree/files. The code in the editor is:

```

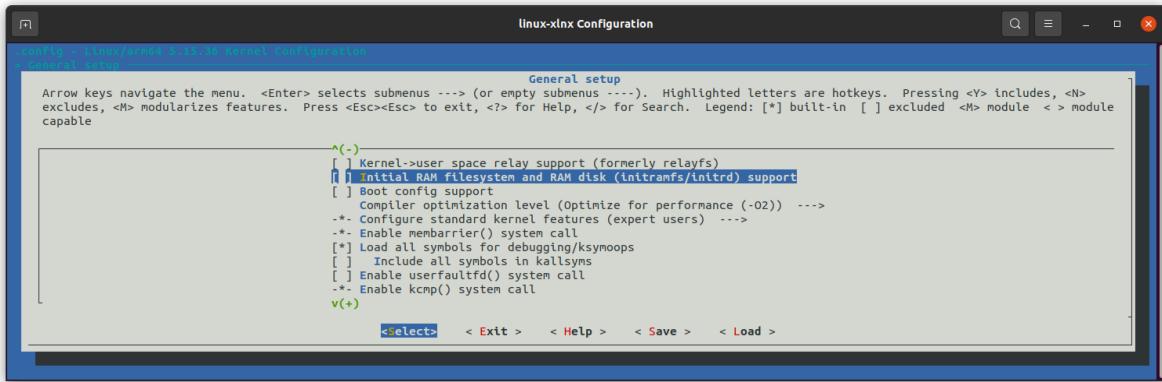
system-user.dtsi
~/Documents/Cat_Doorbell/petalinux/accel/...ec/meta-user/recipes-bsp/device-tree/files

1 /include/ "system-conf.dtsi"
2 /{
3 };
4
5 &axi_intc_0 {
6     xlnx,kind-of-intr = <0x0>;
7     xlnx,num-intr-inputs = <0x20>;
8 };
9
10
11 &amba {
12     zyxclmm_drm {
13         compatible = "xlnx,zocl";
14         status = "okay";
15         interrupt-parent = <&axi_intc_0>;
16         interrupts = <0 4>, <1 4>, <2 4>, <3 4>,
17                     <4 4>, <5 4>, <6 4>, <7 4>,
18                     <8 4>, <9 4>, <10 4>, <11 4>,
19                     <12 4>, <13 4>, <14 4>, <15 4>,
20                     <16 4>, <17 4>, <18 4>, <19 4>,
21                     <20 4>, <21 4>, <22 4>, <23 4>,
22                     <24 4>, <25 4>, <26 4>, <27 4>,
23                     <28 4>, <29 4>, <30 4>, <31 4>;
24     };
25 };


```

Disable ramdisk

```
petalinux-config -c kernel
```



Modify bootargs in the devicetree as per below:

```

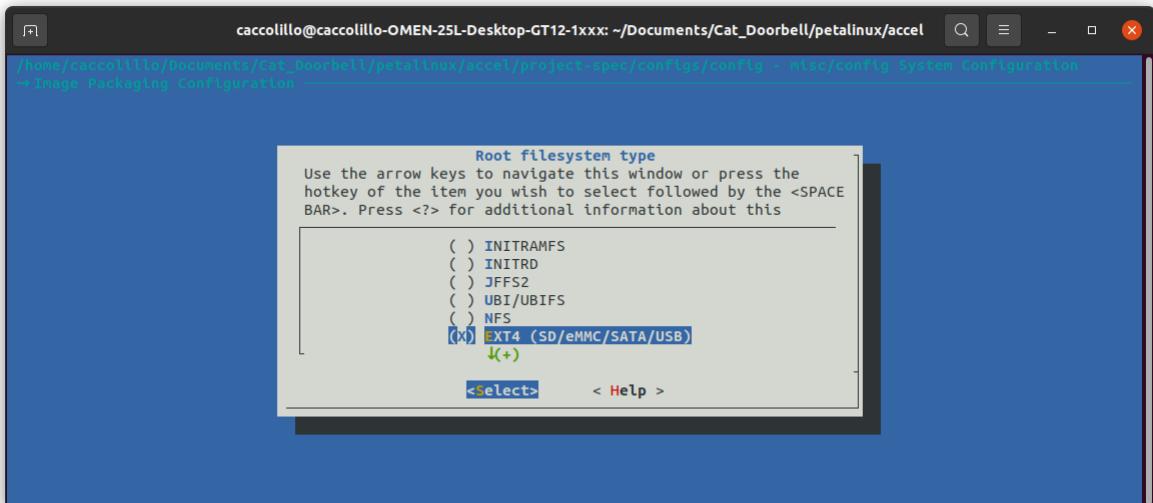
1 /include/ "system-conf.dtsi"
2 {
3     chosen {
4         bootargs = "console=ttyPS0,115200 root=/dev/mmcblk0p2 rw earlyprintk rootfstype=ext4 rootwait uio_pdrv_genirq.of_id=generic-uio devtmpfs.mount=1 earlycon";
5     };
6 };
7
8 &xlnx_intc_0 {
9     xlnx,kind-of-intr = <0x0>;
10    xlnx,num-intr-inputs = <0x20>;
11 };
12
13
14 &namba {
15     zynqmm_drm {
16         compatible = "xlnx,zocl";
17         status = "okay";
18         interrupt-parent = <xlnx_intc_0>;
19         interrupts = <0 4>, <1 4>, <2 4>, <3 4>,
20             <4 4>, <5 4>, <6 4>, <7 4>,
21             <8 4>, <9 4>, <10 4>, <11 4>,
22             <12 4>, <13 4>, <14 4>, <15 4>,
23             <16 4>, <17 4>, <18 4>, <19 4>,
24             <20 4>, <21 4>, <22 4>, <23 4>,
25             <24 4>, <25 4>, <26 4>, <27 4>,
26             <28 4>, <29 4>, <30 4>, <31 4>;
27     };
28 };

```

1) copy boot.scr, BOOT.BIN and image.ub to BOOT partition

2) run tar -xvf ./images/linux/rootfs.tar.gz -C /media/camelo/ROOT

petalinux-config -c kernel



```
/include/ "system-conf.dtsi"
{
chosen {
    bootargs = "console=ttyPS0,115200 root=/dev/mmcblk0p2 rw earlyprintk
rootfstype=ext4 rw rootwait uio_pdrv_genirq.of_id=generic-uio devtmpfs.mount=1
earlycon";
};

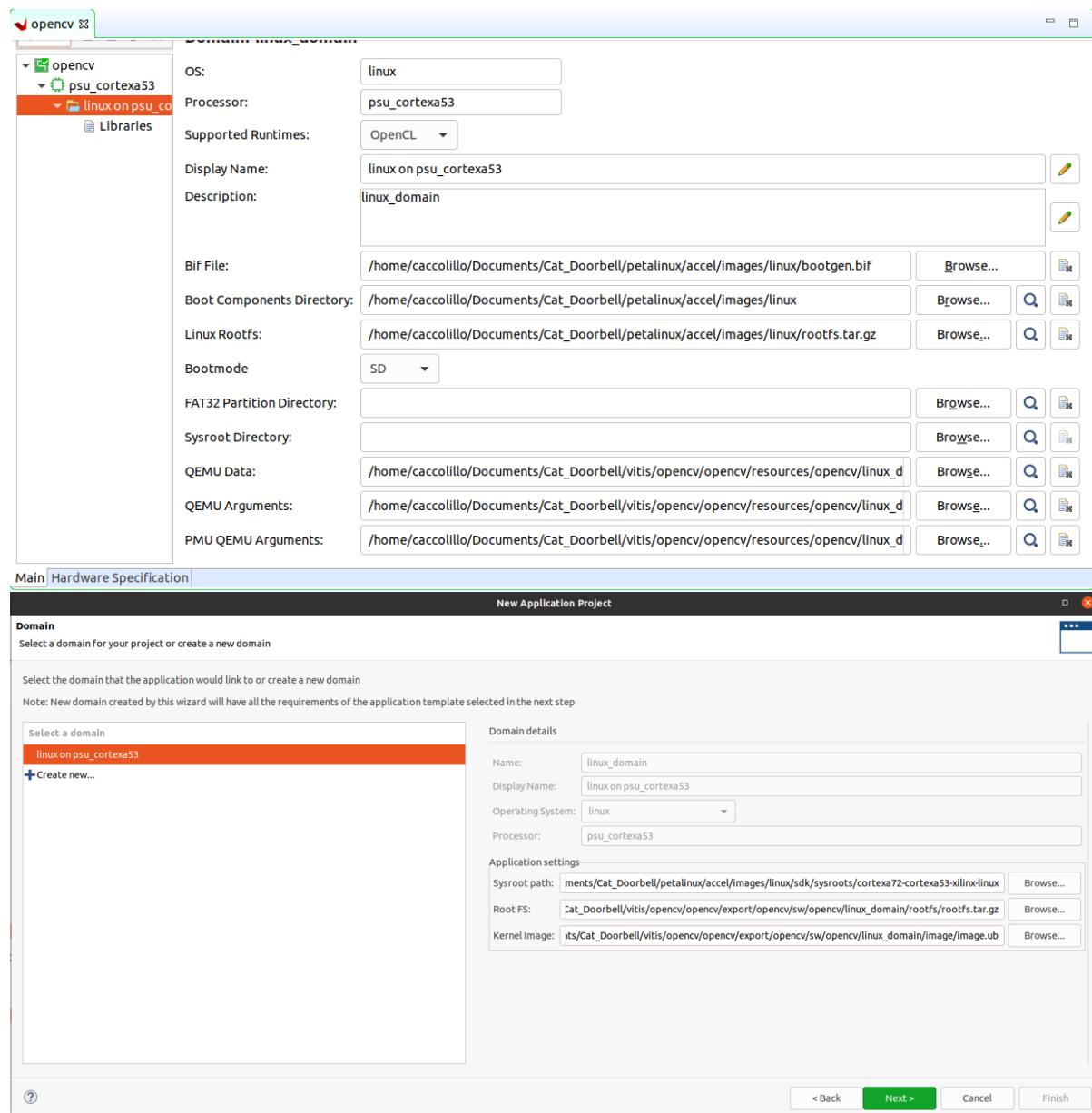
&sdhci0 {
no-1-8-v;
disable-wp;
};

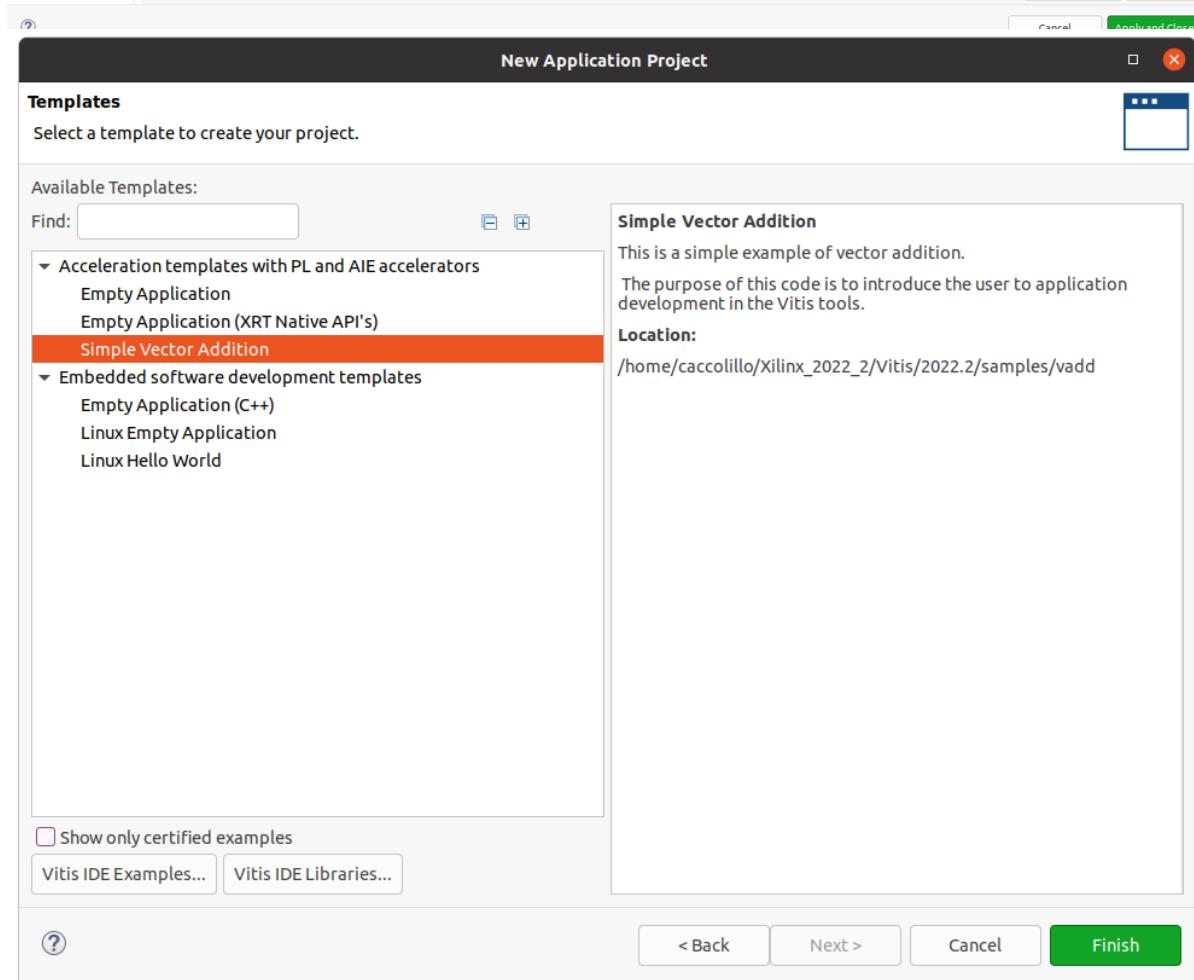
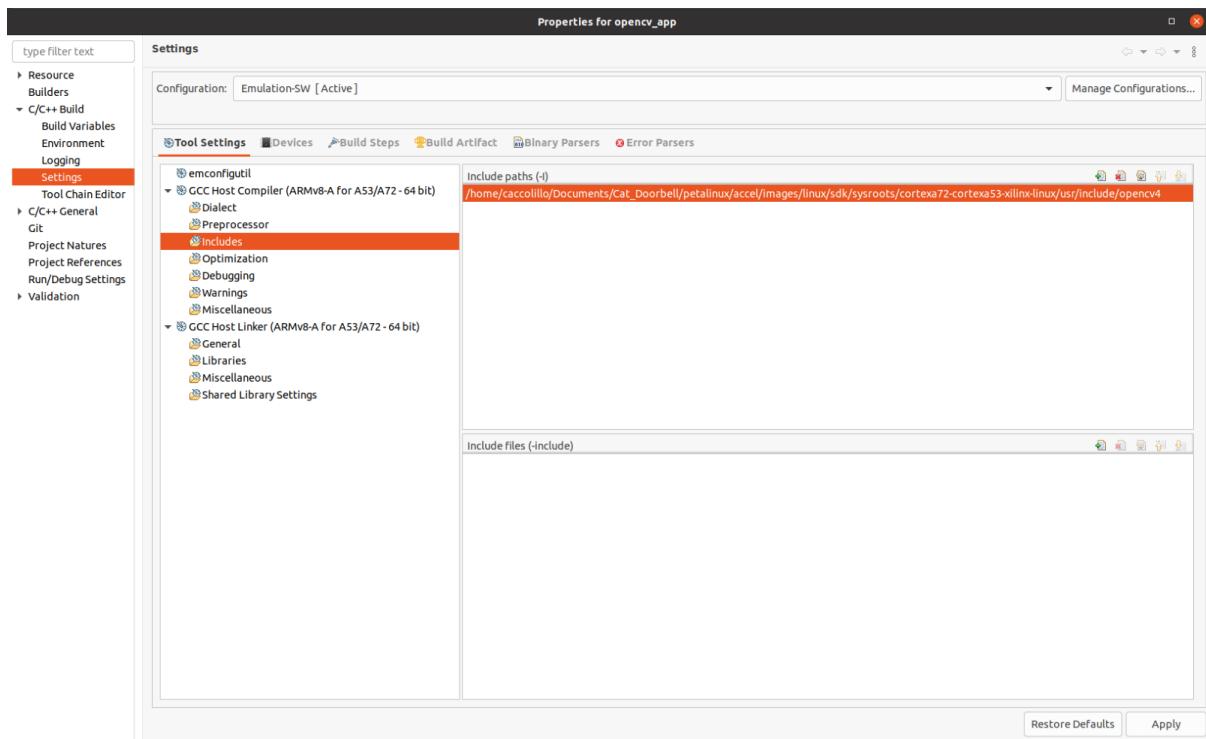
&axi_intc_0 {
    xlnx,kind-of-intr = <0x0>;
    xlnx,num-intr-inputs = <0x20>;
};

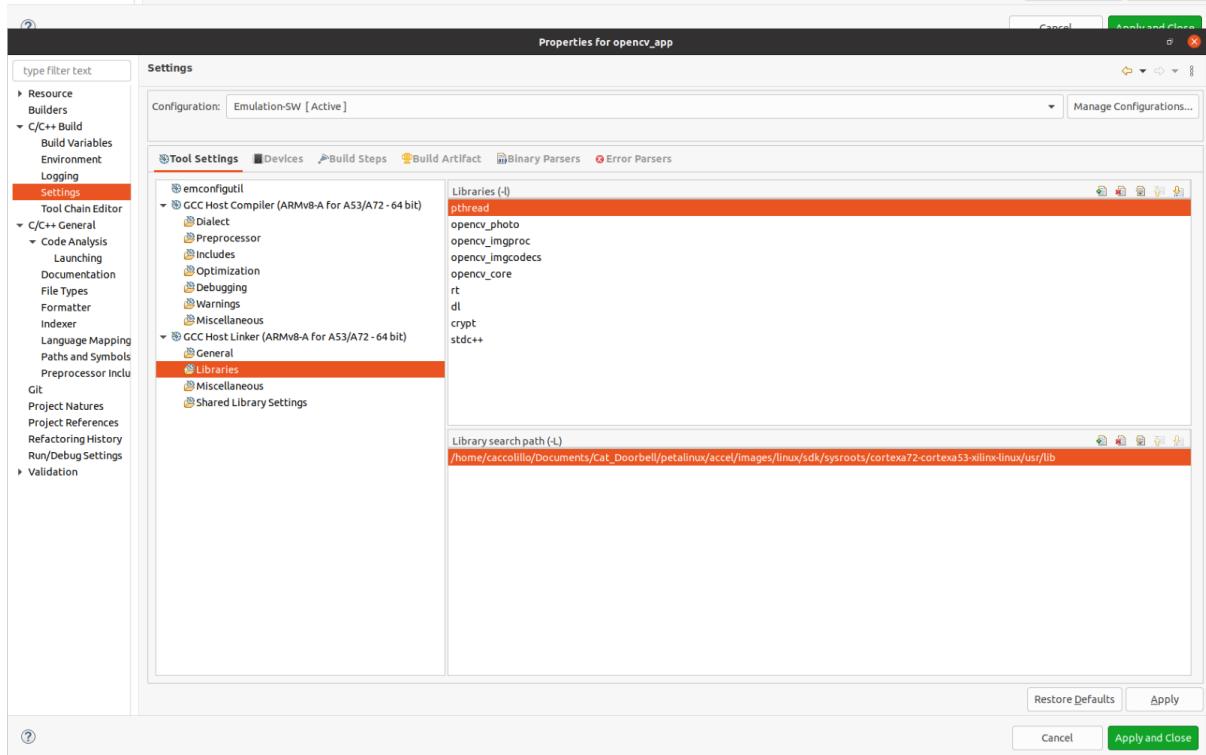
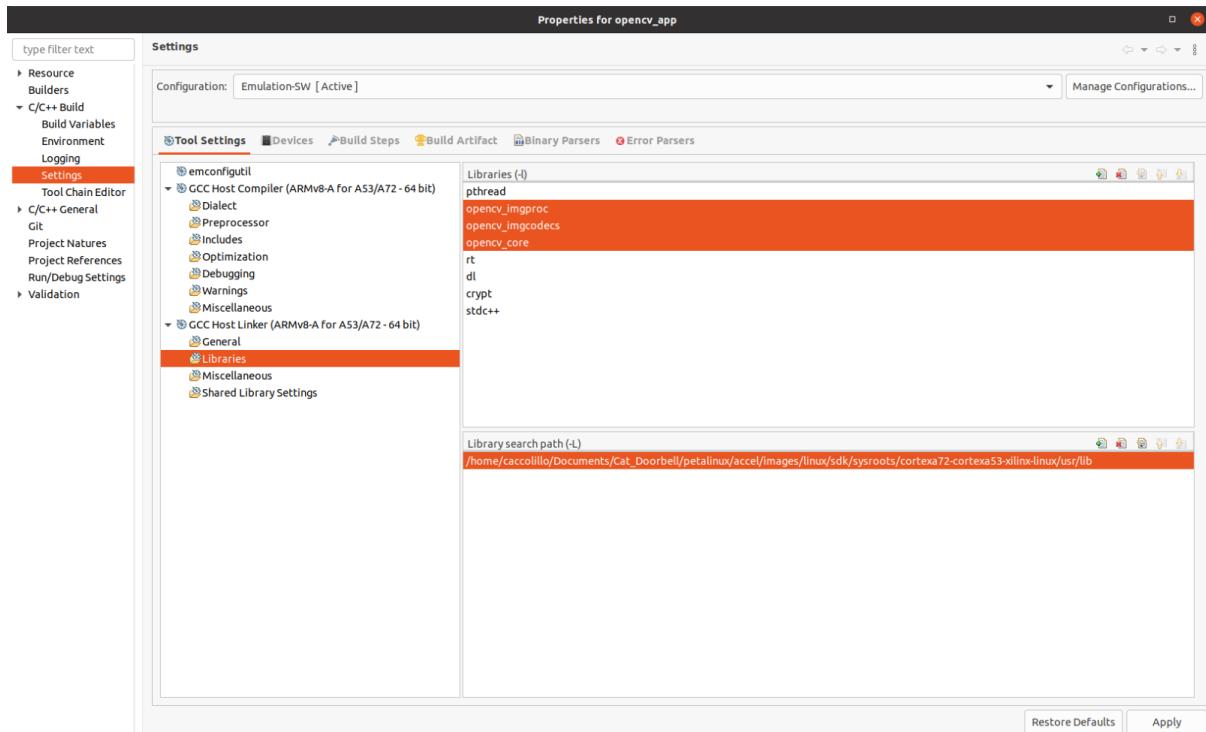
&amba {
zyxclmm_drm {
    compatible = "xlnx,zocl";
    status = "okay";
    interrupt-parent = <&axi_intc_0>;
    interrupts = <0 4>, <1 4>, <2 4>, <3 4>,
                 <4 4>, <5 4>, <6 4>, <7 4>,
                 <8 4>, <9 4>, <10 4>, <11 4>,
                 <12 4>, <13 4>, <14 4>, <15 4>,
                 <16 4>, <17 4>, <18 4>, <19 4>,
                 <20 4>, <21 4>, <22 4>, <23 4>,
                 <24 4>, <25 4>, <26 4>, <27 4>,
                 <28 4>, <29 4>, <30 4>, <31 4>;
```

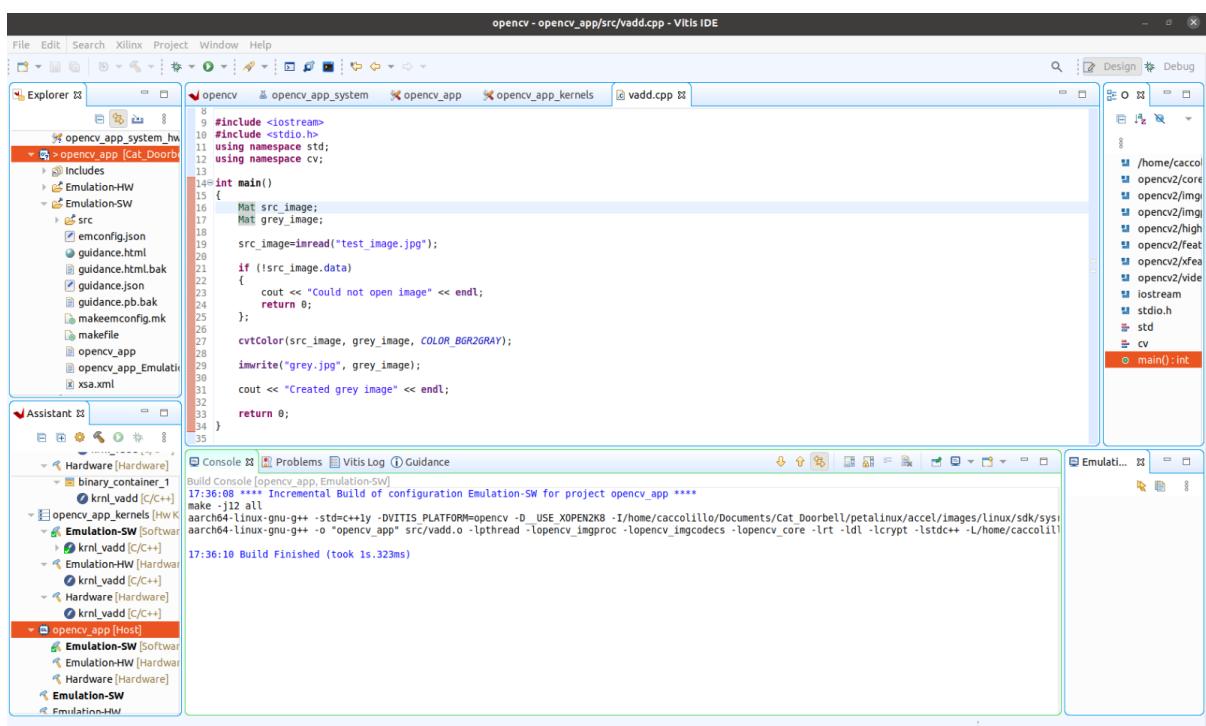
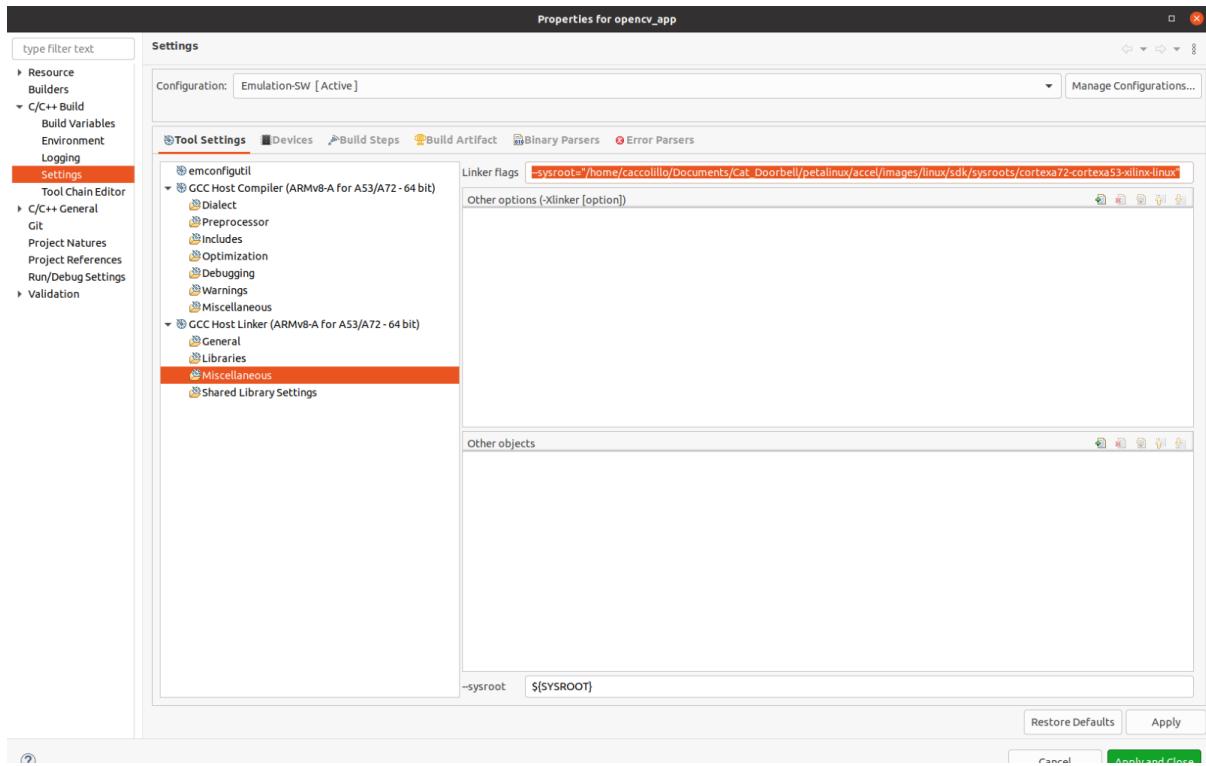
```
};  
};
```

How to compile opencv apps in vitis and where to find the executable





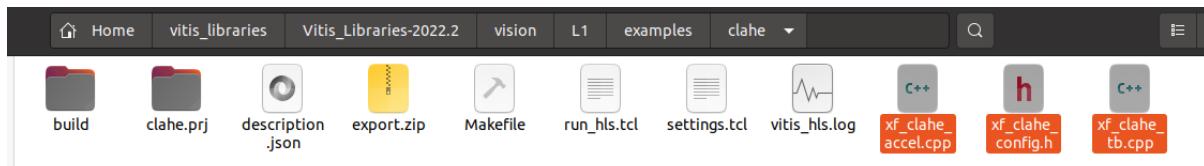




```
caccolillo@caccolillo-OMEN-2SL-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/vitis/opencv/opencv_app/Emulation-SW$ file opencv_app
opencv_app: ELF 64-bit LSB shared object, ARM aarch64, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux-aarch64.so.1, BuildID[sha1]=2c314779fd13f2ec47cb32826b45086dc4c37119, for GNU/Linux 3.14.0, with debug_info, not stripped
caccolillo@caccolillo-OMEN-2SL-Desktop-GT12-1xxx: ~/Documents/Cat_Doorbell/vitis/opencv/opencv_app/Emulation-SW$
```

CLAHE FROM VITIS VISION LIBRARIES L1 USAGE UNDER OPENCL

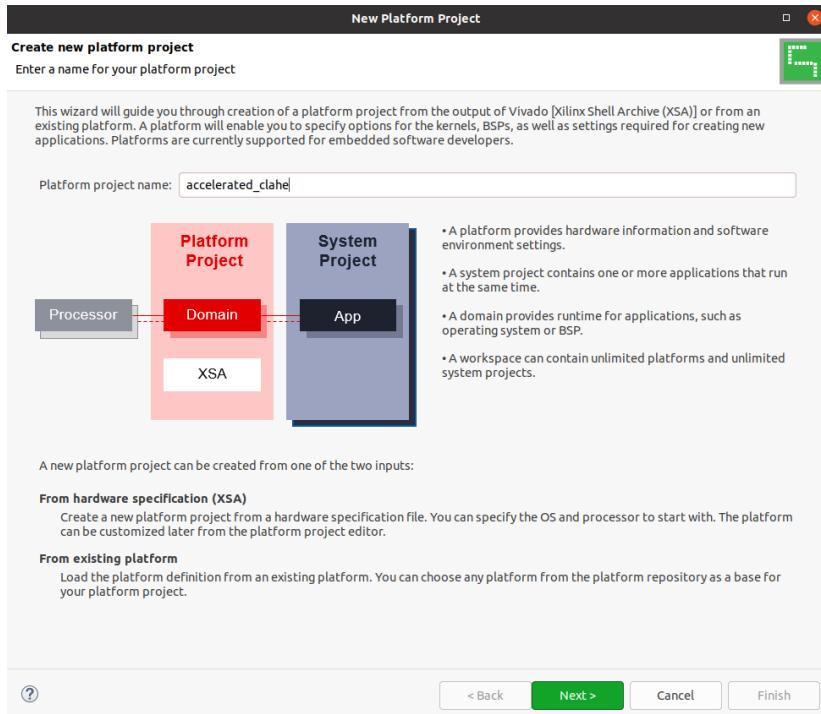
After having cloned the vitis vision libaries, in the example folder there is the CLAHE example:

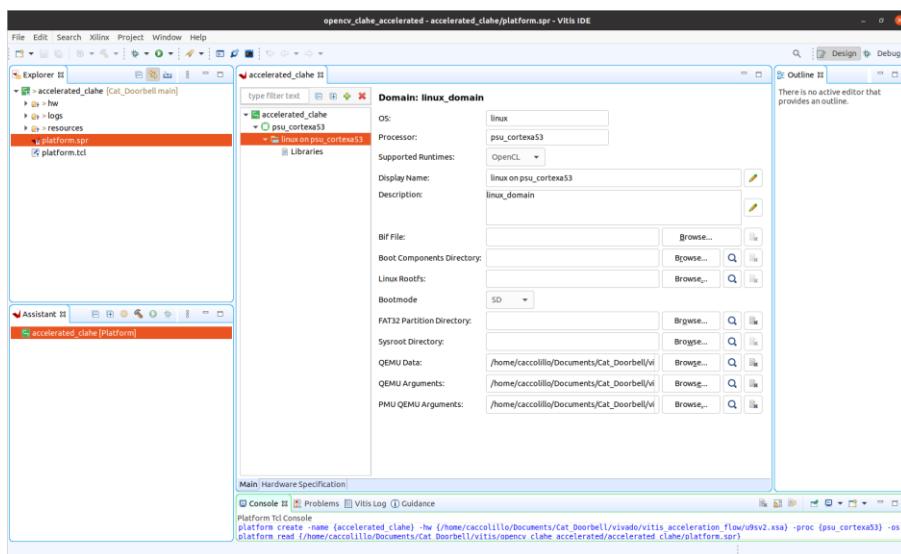
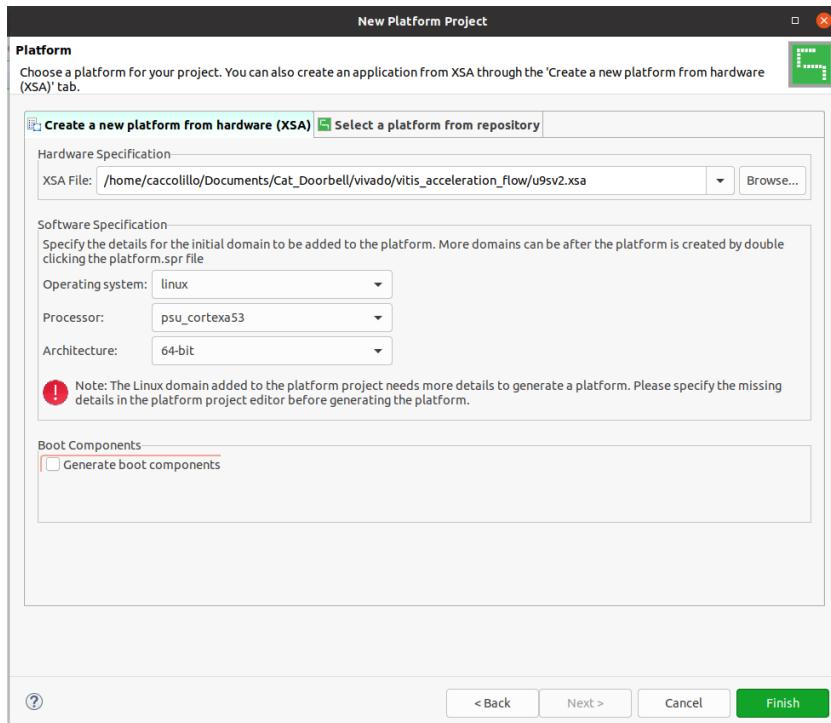


The testbench can be used in vivado to exercise the accelerated kernel in vitis HLS.

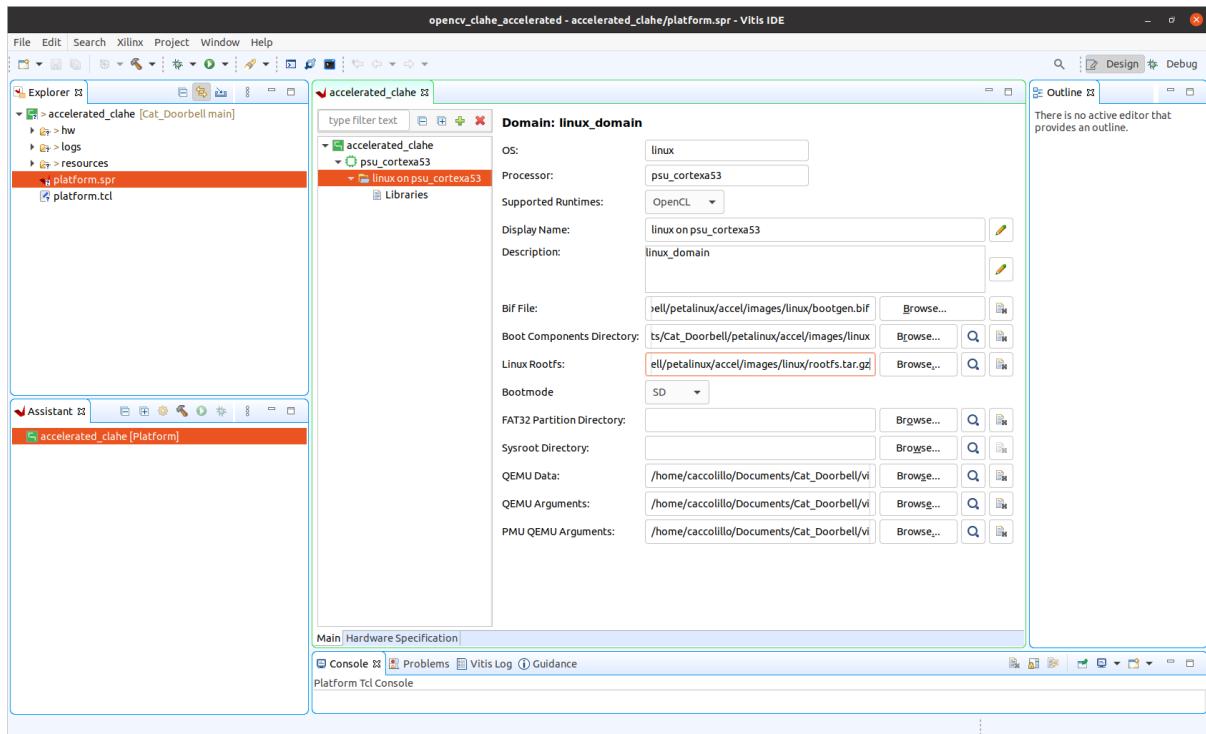
It would be nice using the code from within Vitis, so a Vitis project based upon the Petalinux distribution described insofar has been created, using the vector addition as a starting point.

Create a new platform project, specifying the xsa file:

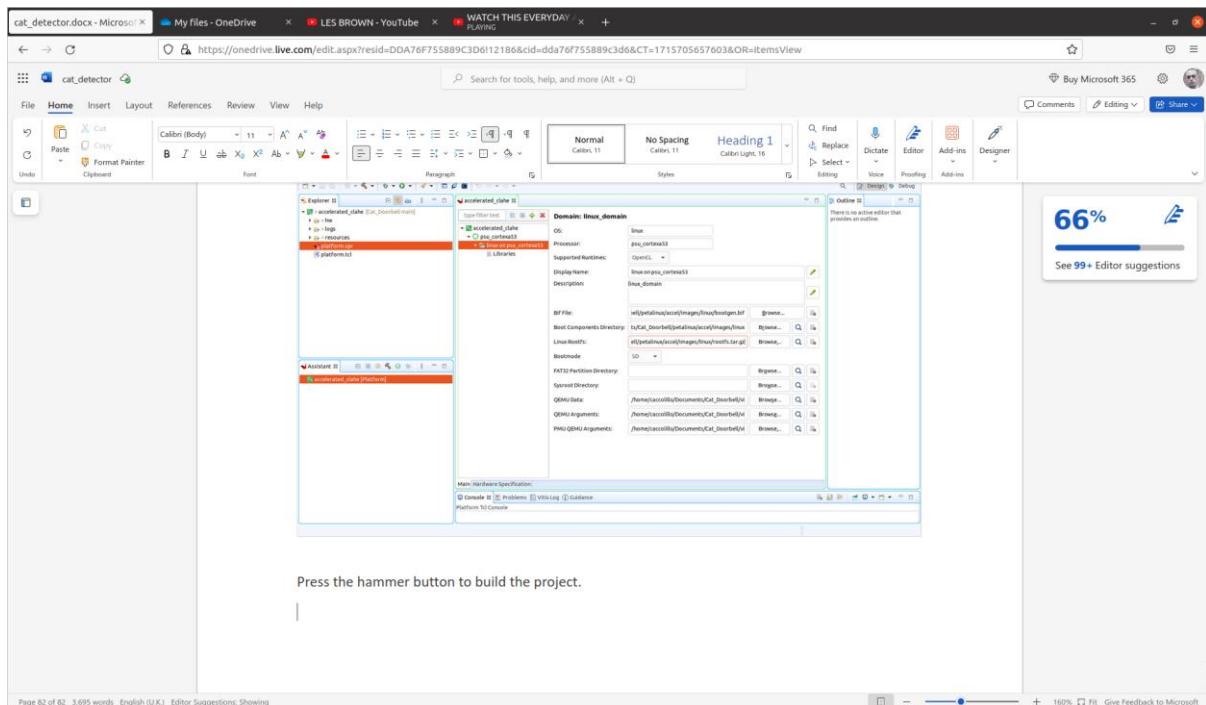




Point the bif file, boot components directory, and linux rootfs as specified below:



Press the hammer button to build the project.



Then create a new application project, and select as platform repository the one just created:

New Application Project

Platform

Choose a platform for your project. You can also create an application from XSA through the 'Create a new platform from hardware (XSA)' tab.

Select a platform from repository [Create a new platform from hardware \(XSA\)](#)

Find:

Add **Manage**

Name	Board	Flow	Vendor	Path
accelerated_clahe [custom]	ultra96v2	Embedded Accel	avnet	/home/caccolillo/Documents/Cat_Doorbell/vitis/op
xilinx_vck190_base_202220_1	xd	Embedded Accel	xilinx.com	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_
xilinx_vck190_base_dfx_202220	xd	Embedded Accel	xilinx.com	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_
xilinx_vmk180_base_202220_1	xd	Embedded Accel	xilinx.com	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_
xilinx_zcu102_base_202220_1	xd	Embedded Accel	xilinx.com	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_
xilinx_zcu102_base_dfx_202220	xd	Embedded Accel	xilinx	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_
xilinx_zcu104_base_202220_1	xd	Embedded Accel	xilinx.com	/home/caccolillo/Xilinx_2022_2/Vitis/2022.2/base_

Platform Info

General Info

Name: accelerated_clahe
Part: xczu3eg-sbva484-1-e
Family: zynqplus
Description: accelerated_clahe

Acceleration Resources

Clock Frequencies

Clock	Frequency (MHz)
PL 0	300.000000
PL 1	100.000000
PL 2	200.000000
PL 4	400.000000
PL 5	600.000000

Domain Details

Domains

Domain name	Details
linux on psu_cortexa53	CPU: cortex-a53 OS: linux

[?](#) [< Back](#) [Next >](#) [Cancel](#) [Finish](#)

New Application Project

Application Project Details

Specify the application project name and its system project properties

Application project name: clahe_kernel

System Project

Create a new system project for the application or select an existing one from the workspace i

Select a system project

+ Create new...

System project details

System project name: clahe_kernel_system

Target processor

Select target processor for the Application project.

Processor	Associated applications
psu_cortexa53 SMP	clahe_kernel

Show all processors in the hardware specification i

?
Next >
Cancel
Finish

New Application Project

Domain

Select a domain for your project or create a new domain

Select the domain that the application would link to or create a new domain
Note: New domain created by this wizard will have all the requirements of the application template selected in the next step

Select a domain

linux on psu_cortexa53

+ Create new...

Domain details

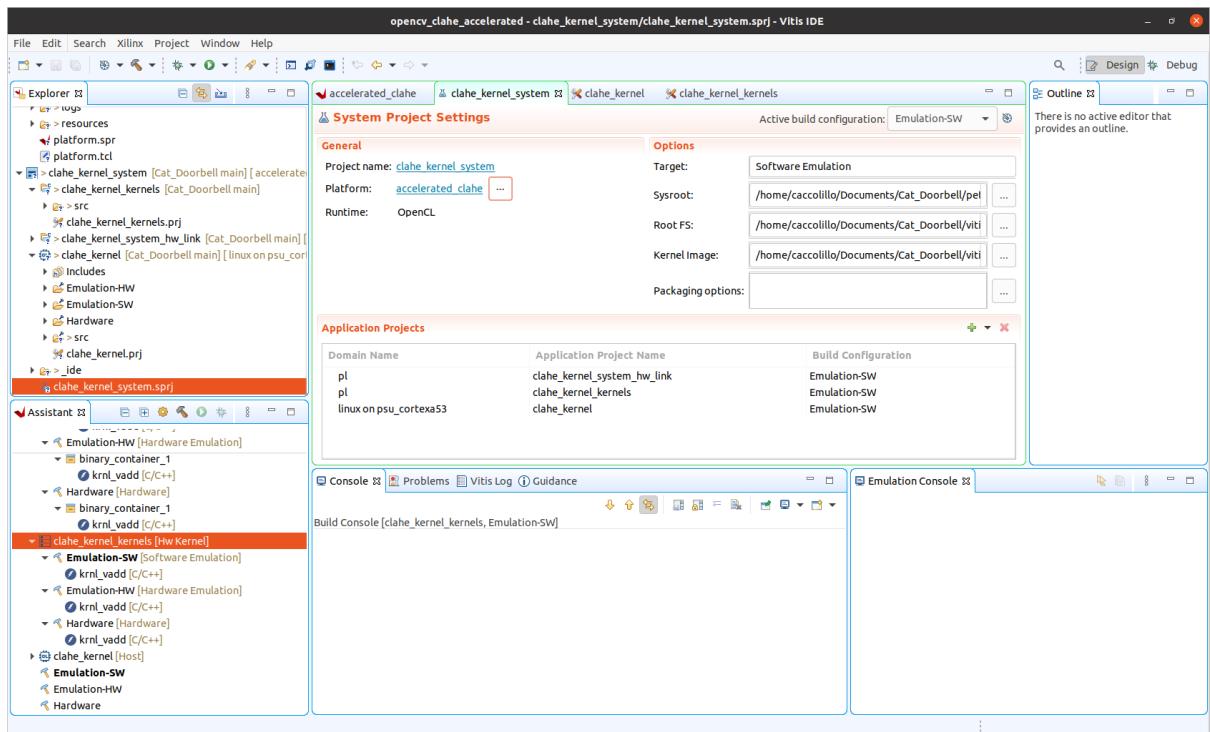
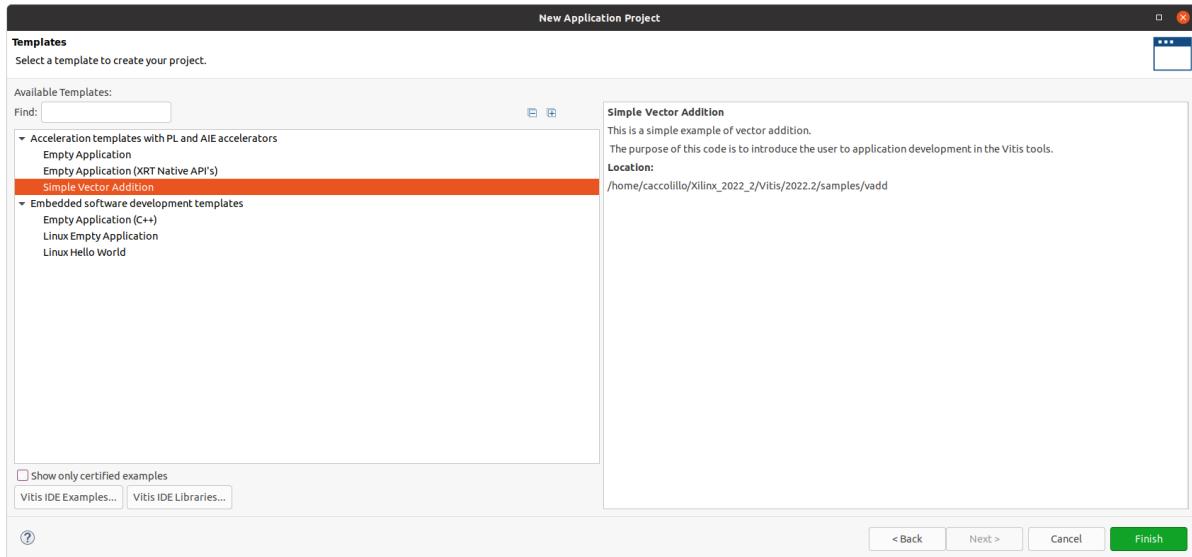
Name:	<input type="text" value="linux_domain"/>
Display Name:	<input type="text" value="linux on psu_cortexa53"/>
Operating System:	<input style="width: 100px;" type="text" value="linux"/>
Processor:	<input type="text" value="psu_cortexa53"/>

Application settings

Sysroot path:	<input type="text" value="documents/Cat_Doorbell/petalinux/accl/images/linux/sdk/sysroots/cortexa72-cortexa53-xilinx-linux"/>	Browse...
Root FS:	<input type="text" value="accelerated_clahe/export/accelerated_clahe/sw/accelerated_clahe/linux_domain/rootfs.tar.gz"/>	Browse...
Kernel Image:	<input type="text" value="d/accelerated_clahe/export/accelerated_clahe/sw/accelerated_clahe/linux_domain/image/image.ub"/>	Browse...

?
< Back
Next >
Cancel
Finish

Select the simple vector addition template:



As a safety check press the hammer to build the project: if everything has been set right, it should build successfully.

Bear in mind that good tutorials on the subject are available at:

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-extensible-vitis-platform-c3f947>

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-software-in-petalinux-b2456c>

<https://www.hackster.io/whitney-knitter/vitis-acceleration-flow-on-kv260-vitis-platform-c3537e>

Other excellent tutorials from Xilinx are available at:

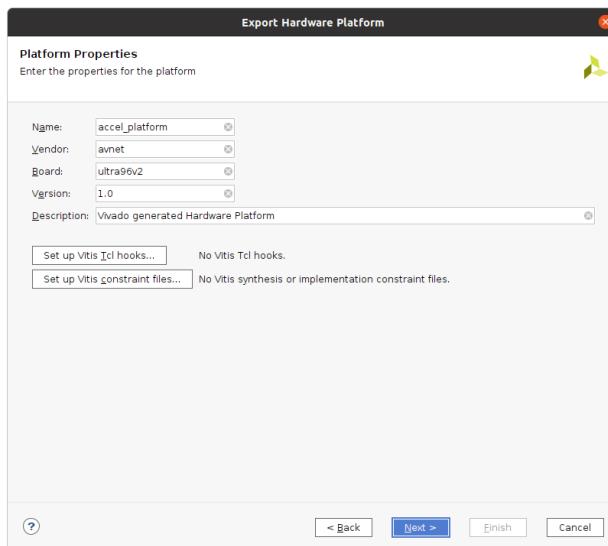
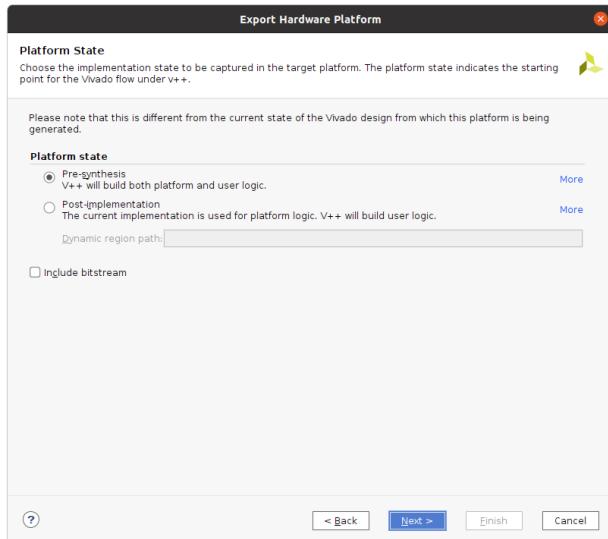
https://support.xilinx.com/s/article/1138208?language=en_US

https://support.xilinx.com/s/question/0D52E00007IPgTmSAL/creating-an-acceleration-platform-for-vitis-part-two-creating-the-software-project-for-the-acceleration-platform-in-petalinux?language=en_US

https://support.xilinx.com/s/article/1138750?language=en_US

https://support.xilinx.com/s/article/1138865?language=en_US

Be careful when exporting the hardware (in “export platform” -> “export hardware platform”-> “hardware”-> “pre-synthesis” with “include bitstream deselected”) of adding a valid name in the gui:



Otherwise the Vitis IDE will error out.

A tutorial about how to use the debugger is at:

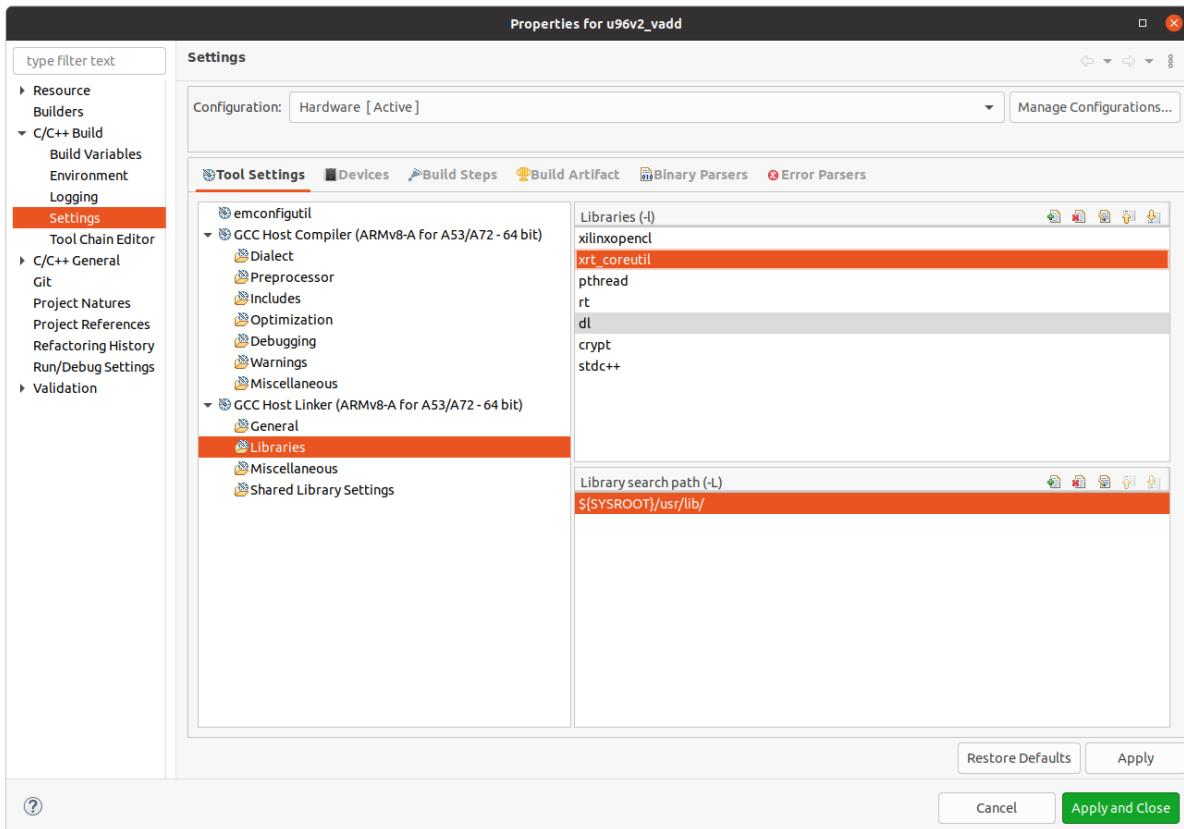
<https://www.hackster.io/whitney-knitter/debugging-accelerated-apps-using-emulation-in-vitis-2021-2-749cd4>

In the end, the tutorial to follow to get all sorted out in Vitis is:

<https://www.hackster.io/whitney-knitter/accelerated-design-development-on-kria-kr260-in-vitis-2022-1-883799>

At least for the Vitis 2022.2 version.

In order to use xrt, add xrt_coreutil library in vitis for the host code:



In the end this **DOES NOT WORK.**

To get it working, the tutorial design below has been followed:

<https://www.hackster.io/AlbertaBeef/ultra96-v2-building-the-foundational-designs-e4315f>

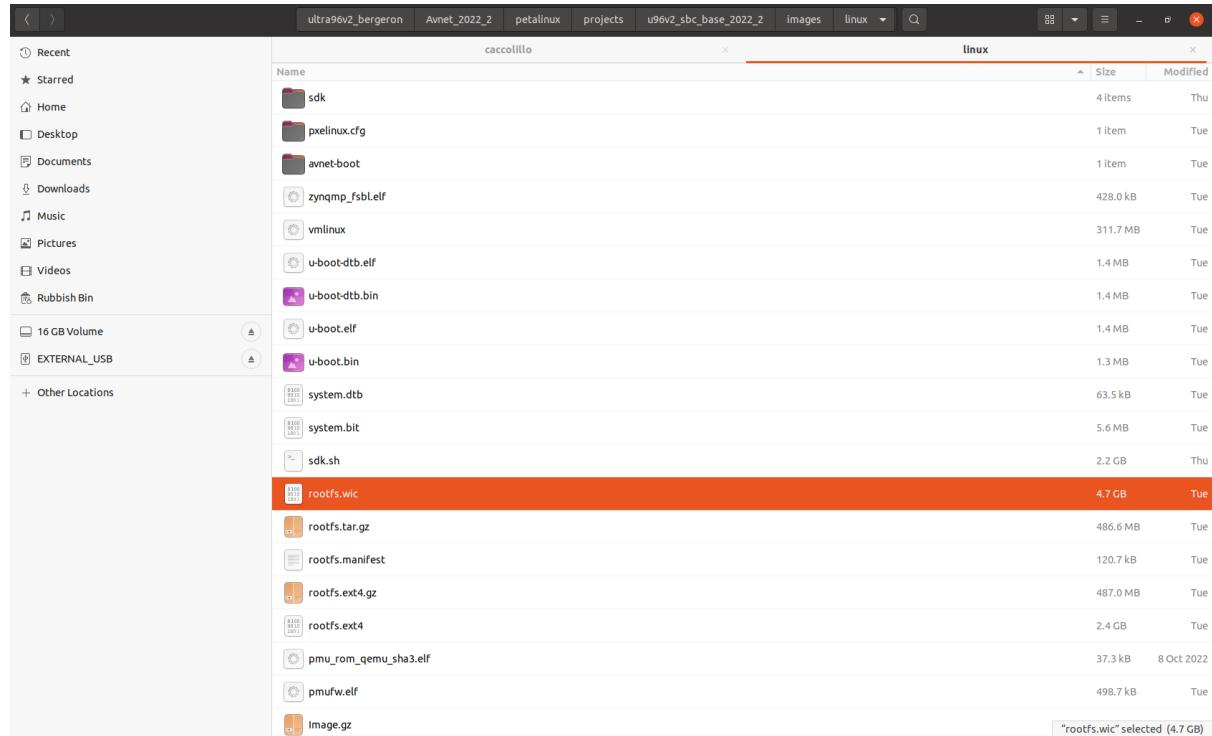
Creating the u96v2_sbc_base design.

Then the following tutorial:

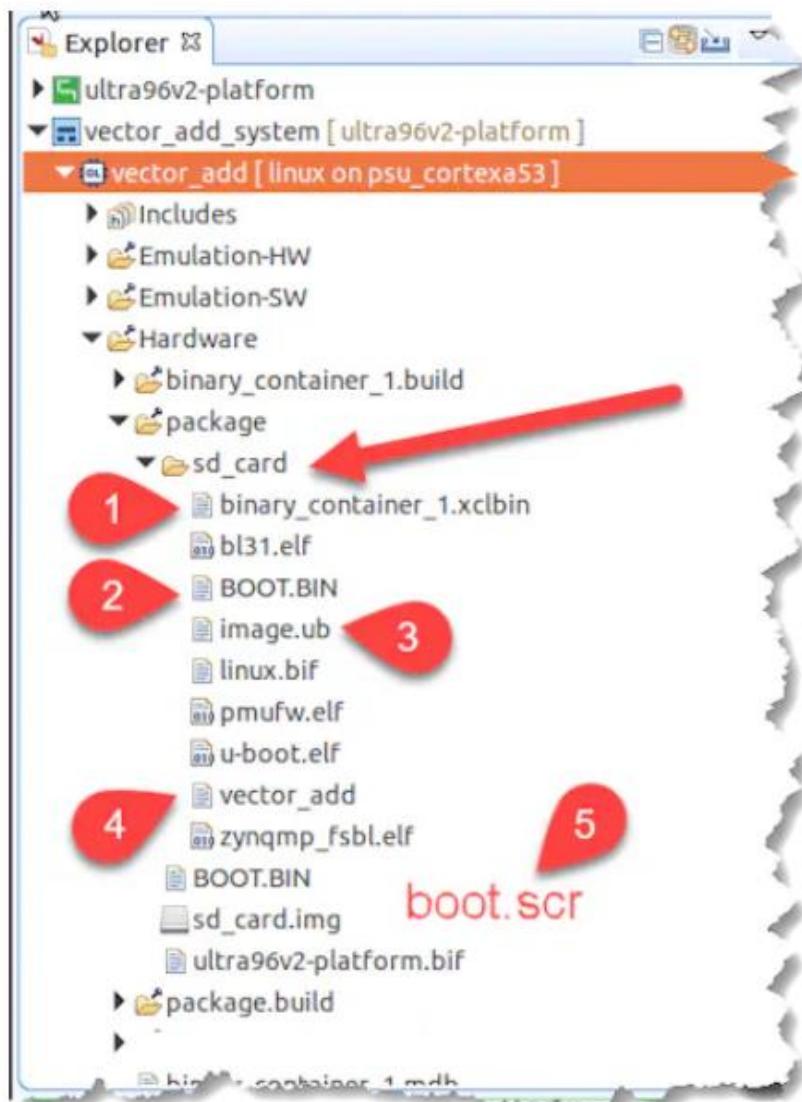
<https://highlevel-synthesis.com/2021/08/14/how-to-create-ultra96v2-linux-based-platform-in-xilinx-vitis-2020-2/>

Has been used to get the Vitis project, but the procedure for the bif was wrong, so the option in Vitis to let the tool to create the bif has been used.

The micro SD has been created from the WIC file generated by Bergeron tutorial and burned in Windows with Rufus:



Then the files specified in the tutorial have been copied in the BOOT partition ():



Boot.scr was the one already in the SD-card, as it hasn't been generated.

Once the board has booted, giving sudo su, then fdisk -l:

```

GTKTerm - /dev/ttyUSB1 115200-8-N-1
File Edit Log Configuration Controls/Signals View Help
Disk /dev/ram13: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram14: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ram15: 64 MiB, 67108864 bytes, 131072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/mmcblk0: 59.45 GiB, 63831015424 bytes, 124669952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xd004f1a2

Device      Boot   Start   End Sectors  Size Id Type
/dev/mmcblk0p1 *     8 2097159 2097152   1G c W95 FAT32 (LBA)
/dev/mmcblk0p2 2097160 9202023 7104864 3.4G 8 Linux
u96v2-sbc-base-2022-2:/home/petalinux# ls /dev/mmcblk0p1
/dev/mmcblk0p1
u96v2-sbc-base-2022-2:/home/petalinux# mount /dev/mmcblk0p1 /mnt
mount: /mnt: /dev/mmcblk0 already mounted or mount point busy.
u96v2-sbc-base-2022-2:/home/petalinux# ls /mnt
u96v2-sbc-base-2022-2:/home/petalinux# mount /dev/mmcblk0p1 /mnt
u96v2-sbc-base-2022-2:/home/petalinux# ls /mnt
System Volume Information boot.bin image.ub
binary container 1.xclbin boot.scr vector.add
u96v2-sbc-base-2022-2:/home/petalinux# cd /mnt/
u96v2-sbc-base-2022-2:/mnt# ls
System Volume Information boot.bin image.ub
binary container 1.xclbin boot.scr vector.add
u96v2-sbc-base-2022-2:/mnt# ./vector.add binary_container_1.xclbin
INFO: Reading binary_container_1.xclbin
Loading "binary_container_1.xclbin"
Trying to program device[0]: u96v2_custom
Device[0]: program successful!
TEST PASSED
u96v2-sbc-base-2022-2:/mnt# 

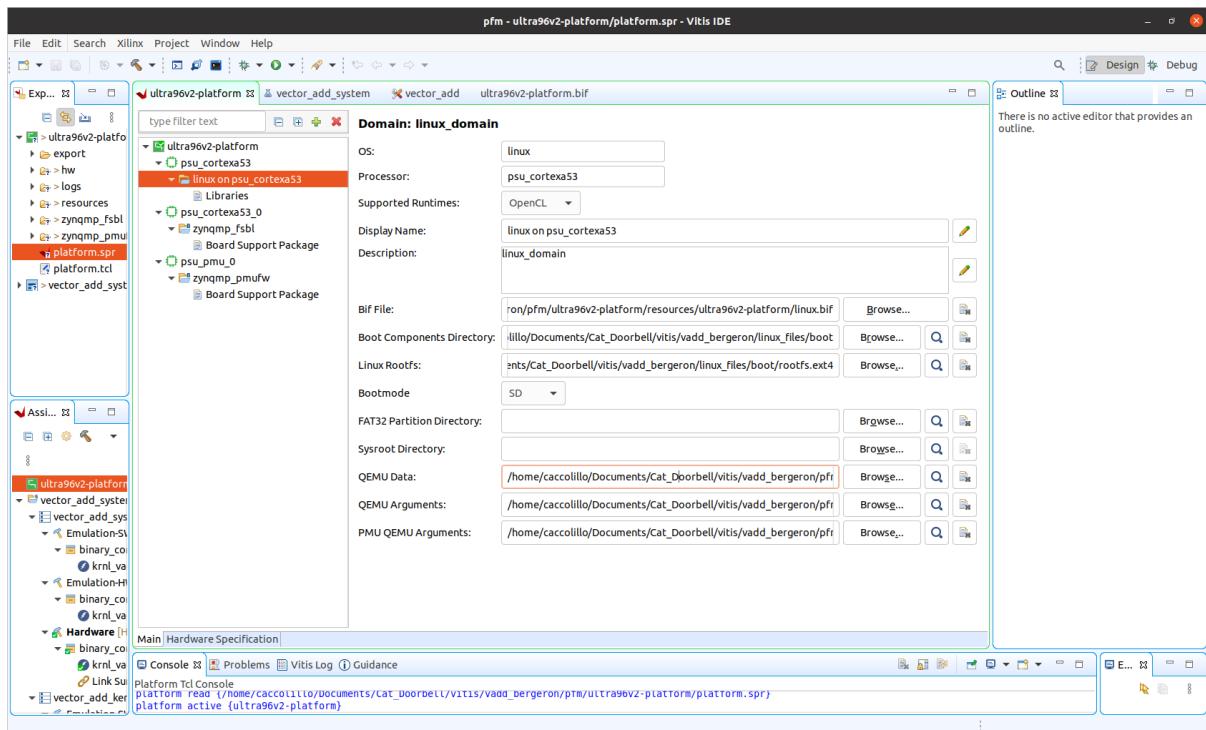
```

Run the following commands to define the Xilinx XRT library

```
export XILINX_XRT=/usr
```

And the program works!

In the following screenshots about the configuration in Vitis:



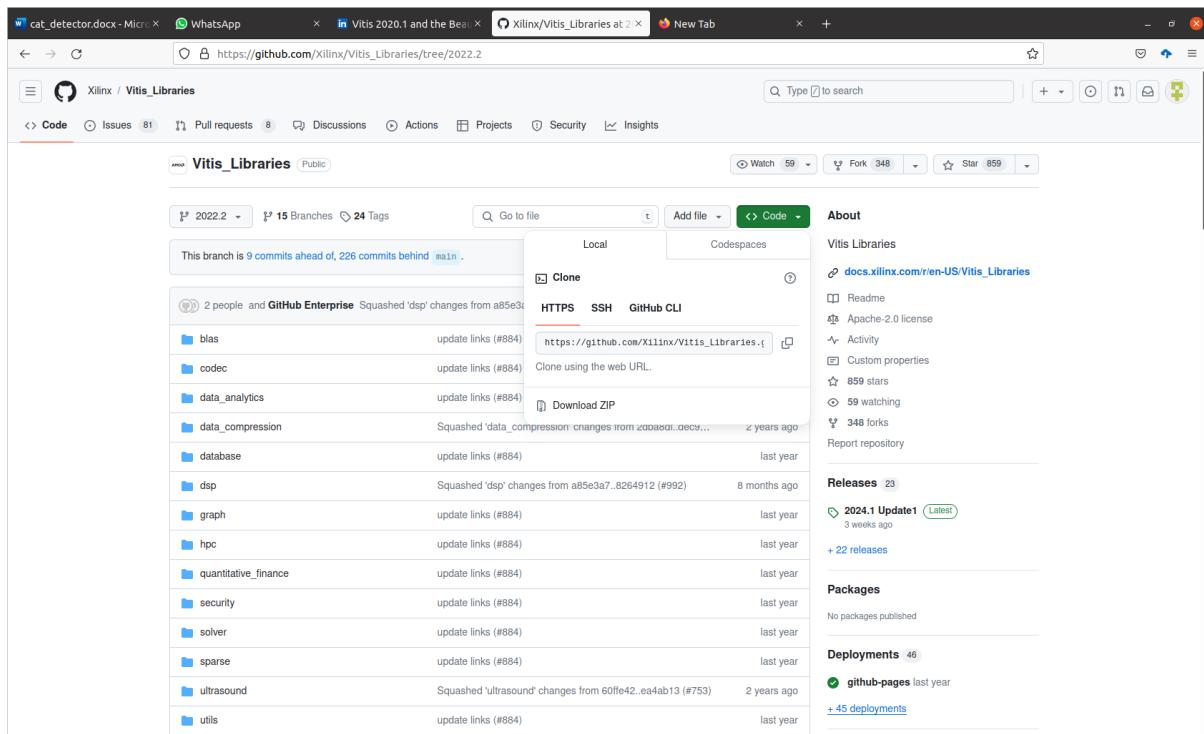
AUTOWHITE BALANCE FROM VITIS VISION LIBRARIES L2 USAGE UNDER OPENCL

As per tutorial below, an attempt to install Vitis Vision libraries in Vitis has been done:

<https://www.linkedin.com/pulse/vitis-20201-beauty-acceleration-martin-kellermann/>

But due to internet connections, the git cloning wasn't working.

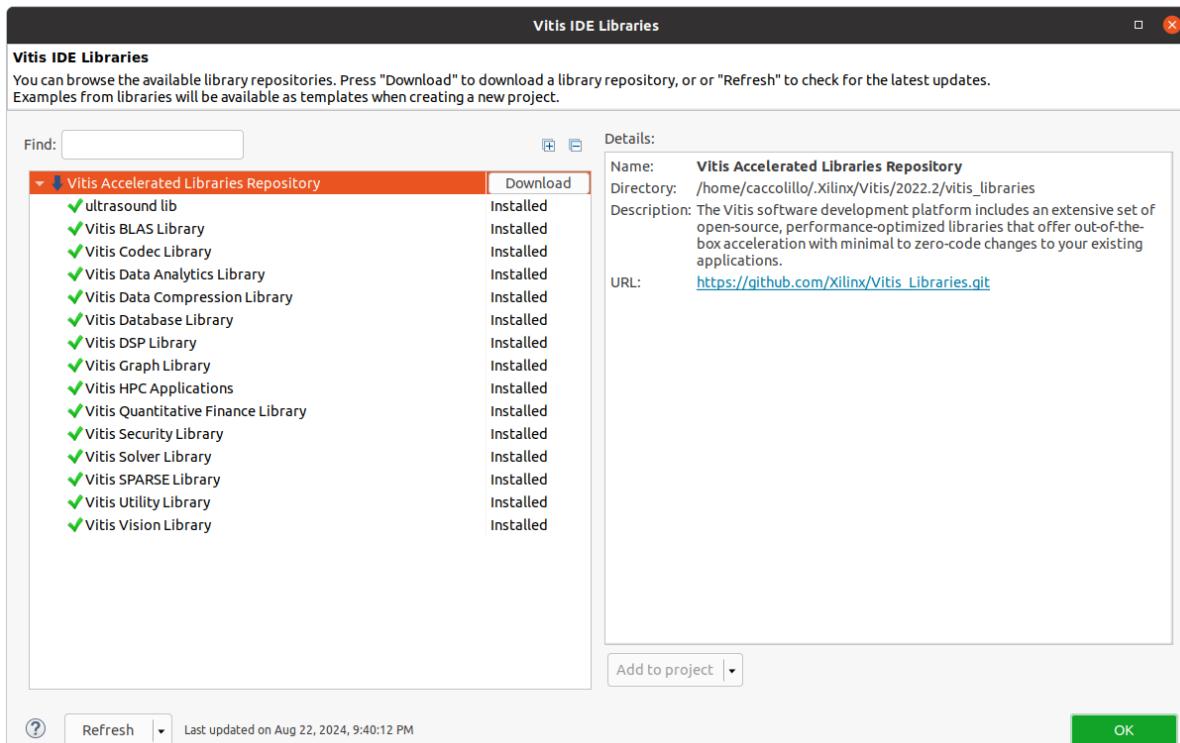
So an archive of the libraries has been downloaded from git:



The archive gets extracted in the folder below:



Once done, they appear in Vitis:



The CLAHE example has been attempted but it wasn't working.

The auto white balance has worked and has been implemented and tested on hardware.

VITIS AI

The compatibility list of Vitis AI releases follows:

https://xilinx.github.io/Vitis-AI/3.0/html/docs/reference/version_compatibility.html

Current Release

Vitis AI v3.0 and the DPU IP released with the v3.0 branch of this repository are verified as compatible with Vitis, Vivado™, and PetaLinux version 2022.2. If you are using a previous release Vitis AI, please refer to the table below release.

Previous Releases

Software Tools Version	DPUCZDX8G IP Version	Vitis AI Release Version
Vivado / Vitis / PetaLinux 2022.2	4.1	v3.0
Vivado / Vitis / PetaLinux 2022.1	4.0	v2.5
Vivado / Vitis / PetaLinux 2021.2	3.4	v2.0
Vivado / Vitis / PetaLinux 2021.1	3.3	v1.4
Vivado / Vitis / PetaLinux 2020.2	3.3	v1.3
Vivado / Vitis / PetaLinux 2020.1	3.2	v1.2
Vivado / Vitis / PetaLinux 2019.2	3.2	v1.1
Vivado / Vitis / PetaLinux 2019.1	3.1	v1.0
Vivado / PetaLinux 2019.1	3.0	N/A
Vivado / PetaLinux 2018.2	2.0	N/A
Vivado / PetaLinux 2018.1	1.0	First Release

Software Tools Version	DPUCVDX8G IP Version	Vitis AI Release Version
Vitis / PetaLinux 2022.2	1.3	v3.0
Vitis / PetaLinux 2022.1	1.2	v2.5
Vitis / PetaLinux 2021.2	1.1	v2.0
Vitis / PetaLinux 2021.1	1.0	v1.4
Vitis / PetaLinux 2020.2	Early Access	v1.3

Software Tools Version	DPUCV2DX8G IP Version	Vitis AI Release Version
Vitis / PetaLinux 2022.2	Early Access	v3.0

Docker has to be installed as per:

<https://docs.docker.com/engine/install/ubuntu/>

Instructions to use a readymade micro SD from Avnet are at:

<https://www.hackster.io/AlbertaBeef/ultra96-v2-adding-support-for-vitis-ai-3-0-704799>

Instead step by step tutorials are at:

https://octavosystems.com/app_notes/vitis-ai-tutorial-part-1/?highlight=vitis%20ai

https://octavosystems.com/app_notes/vitis-ai-tutorial-part-2/?highlight=vitis%20ai

https://octavosystems.com/app_notes/vitis-ai-tutorial-part-3/?highlight=vitis%20ai

https://octavosystems.com/app_notes/vitis-ai-tutorial-part-4/?highlight=vitis%20ai

The tutorials from Octavo systems look promising as they use pre-built libraries downloaded from Xilinx repositories.

Another source of inspiration is the webinar from ADIUVO:

<https://www.adiuvoengineering.com/vitis-training>

Yet another tutorial comes from Beetlebox:

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-1/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-2/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-3/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-4/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-5/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-6/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-7/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-8/>

<https://beetlebox.org/vitis-ai-using-tensorflow-and-keras-tutorial-part-9/>

<https://beetlebox.org/improving-convolutional-neural-networks-the-weaknesses-of-the-mnist-based-datasets-and-tips-for-improving-poor-datasets/>

<https://beetlebox.org/sign-language-recognition-hand-object-detection-using-r-cnn-and-yolo/>

Another interesting tutorial is:

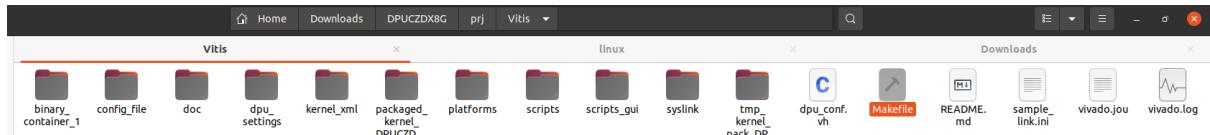
<https://www.hackster.io/LogicTronix/zcu102-vitis-dpu-trd-vitis-ai-3-0-c51609>

The product guide below has been used as a reference:

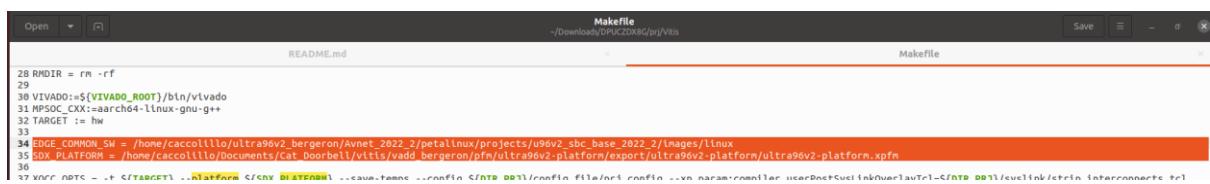
<https://docs.amd.com/r/4.0-English/pg338-dpu/Vitis-DPU-TRD-Flow>

And it refers to the following DPU archive, for the Vitis DPU TRD flow:

The archive gets downloaded and extracted, and the makefile gets modified:



By adding the location of the linux image and of the xfm file:



In the archive, there's also a readme file in MD. It can be opened as HTML by using "mdview README.md" on the command line.

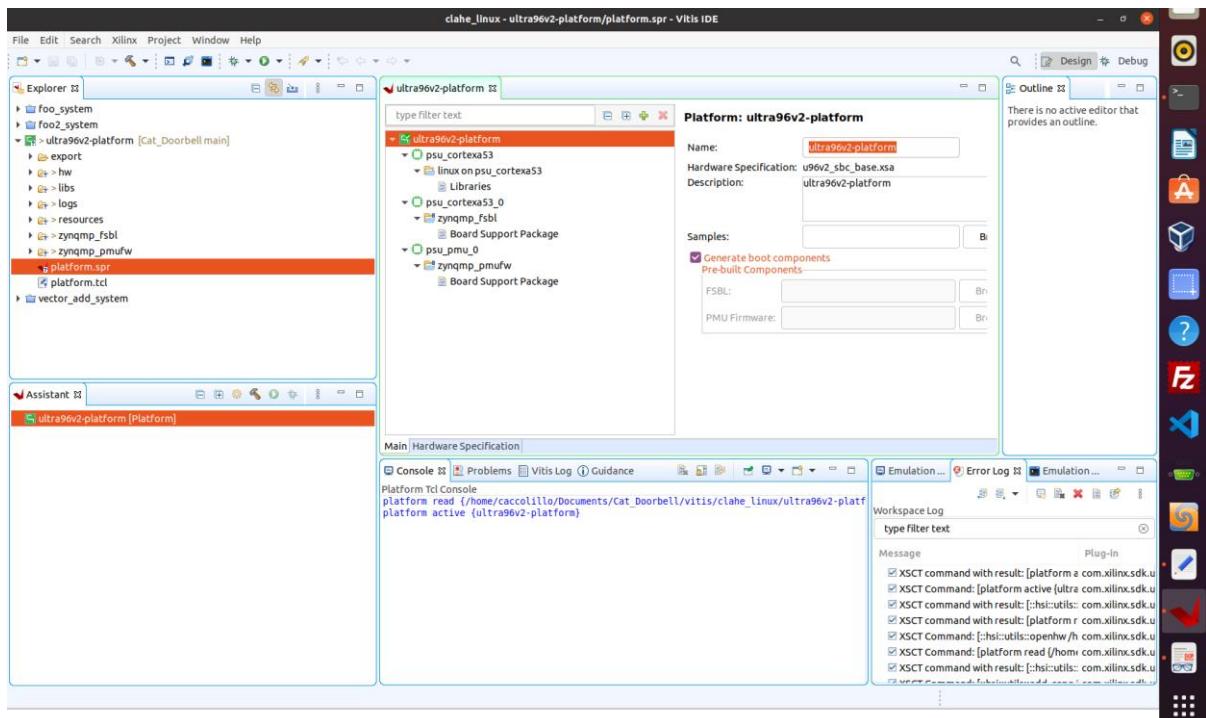
To generate the xo file, use "make binary_container_1/dpu.xo DEVICE=ultra96v2-platform":

```
cacollo@cacollo-OMEN-2SL-Desktop-GT12-1xx:/Downloads/DPUCZDX8G/prj/Vitis
[...]
/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis$ make binary_container_1/dpu.xo DEVICE=ultra96v2-platform
/home/cacollo/Xilinx_2022_2/Vivado/2022.2/bin/vivado -mode batch -source /home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/scripts/gen_dpu.xo.tcl -notrace -tclargs binary_container_1/dpu.xo DPUCZDX8G ultra96v2-platform

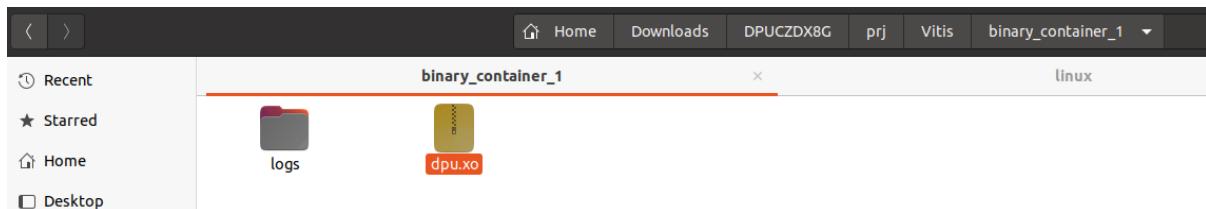
**** Vivado 2022.2 (64-bit)
*** 2022-10-26-14:54:54.000
*** IP Build 3699840 on Fri Oct 14 08:30:02 MOT 2022
** Copyright 1986-2022 Xilinx, Inc. All Rights Reserved.

source /home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/scripts/gen_dpu.xo.tcl -notrace
binary_container_1/dpu.xo
INFO: [IP_FLOW-19-5642] Module 'DPUCZDX8G' uses SystemVerilog sources with a Verilog top file. These SystemVerilog files will not be analysed by the packager.
INFO: [IP_FLOW-19-1842] HDL Parser: Found include file 'src/arch_def.vh' from the top-level HDL file.
INFO: [IP_FLOW-19-1842] HDL Parser: Found include file '/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh' from the top-level HDL file.
INFO: [IP_FLOW-19-1841] HDL Parser: Add include file '/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh' to file group xilinx_anylanguagesynthesis.
INFO: [IP_FLOW-19-1841] HDL Parser: Add include file '/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh' to file group xilinx_anylanguagebehaviorsimulation.
INFO: [IP_FLOW-19-1842] HDL Parser: Found include file 'src/arch_para.vh' from the top-level HDL file.
INFO: [IP_FLOW-19-1704] No IP Repositories specified
INFO: [IP_FLOW-19-2113] Located Vivado IP repository '/home/cacollo/Xilinx_2022_2/Vivado/2022.2/data/ip'.
INFO: [IP_FLOW-19-5107] Inferred bus interface 'ackl' of definition 'xilinx.com:signal:clock:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'ackl' of definition 'xilinx.com:signal:clock:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'ap_clk_2' of definition 'xilinx.com:signal:clock:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'ap_clk_2' of definition 'xilinx.com:signal:clock:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'ap_rst_n_2' of definition 'xilinx.com:signal:reset:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'aresetn' of definition 'xilinx.com:signal:reset:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'aresetn' of definition 'xilinx.com:signal:reset:1.0' (from X_INTERFACE_INFO parameter from HDL file).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'M_AXI_GPO' of definition 'xilinx.com:interface:aximm:1.0' (from Xilinx Repository).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'M_AXI_HPI' of definition 'xilinx.com:interface:aximm:1.0' (from Xilinx Repository).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'M_AXI_RP' of definition 'xilinx.com:interface:aximm:1.0' (from Xilinx Repository).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'S_AXI_CONTROL' of definition 'xilinx.com:interface:aximm:1.0' (from Xilinx Repository).
INFO: [IP_FLOW-19-5107] Inferred bus interface 'interrupt' of definition 'xilinx.com:signal:interrupt:1.0' (from Xilinx Repository).
INFO: [IP_FLOW-19-4728] Bus Interface 'interrupt': Added Interface parameter 'SENSITIVITY' with value 'LEVEL_HIGH'.
INFO: [IP_FLOW-19-4728] Bus Interface 'ackl': Added Interface parameter 'ASSOCIATED_BUSIF' with value 'M_AXI_GPO'.
INFO: [IP_FLOW-19-4728] Bus Interface 'ackl': Added Interface parameter 'ASSOCIATED_RESET' with value 'aresetn'.
INFO: [IP_FLOW-19-4728] Bus Interface 'ackl': Added Interface parameter 'ASSOCIATED_RESET' with value 'ap_rst_n_2'.
INFO: [IP_FLOW-19-4728] Bus Interface 'ap_rst_n_2': Added Interface parameter 'POLARITY' with value 'ACTIVE_LOW'.
INFO: [IP_FLOW-19-4728] Bus Interface 'aresetn': Added Interface parameter 'POLARITY' with value 'ACTIVE_LOW'.
WARNING: [IP_FLOW-19-5661] Bus Interface 'ap_clk_2' does not have any bus interfaces associated with it.
WARNING: [IP_FLOW-19-3157] Bus Interface 'ap_rst_n_2': Bus parameter POLARITY is ACTIVE_LOW but port 'ap_rst_n_2' is not *resetn - please double check the POLARITY setting.
WARNING: [IP_FLOW-19-9911] File Group 'xilinx_anylanguagesynthesis (Synthesis)': "/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh" file path is not relative to the IP root directory.
WARNING: [IP_FLOW-19-4816] The Synthesis file group has two include files that have the same base name. It is not guaranteed which of these two files will be picked up during synthesis/simulation:  src/dpu.conf.vh
/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh
WARNING: [IP_FLOW-19-9911] Unrecognized or unsupported file 'src/fingerprint_json.ttcl' found in file group 'Synthesis'.
Resolution: Remove the file from the specified file group.
WARNING: [IP_FLOW-19-731] File Group 'xilinx_anylanguagebehaviorsimulation (Simulation)': "/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh" file path is not relative to the IP root directory.
WARNING: [IP_FLOW-19-4816] The Simulation file group has two include files that have the same base name. It is not guaranteed which of these two files will be picked up during synthesis/simulation:  src/dpu.conf.vh
/home/cacollo/Downloads/DPUCZDX8G/prj/Vitis/dpu.conf.vh
WARNING: [IP_FLOW-19-9911] Unrecognized or unsupported file 'src/fingerprint_json.ttcl' found in file group 'Simulation'.
Resolution: Remove the file from the specified file group.
INFO: [IP_FLOW-19-2181] Payment Required is not set for this core.
INFO: [IP_FLOW-19-2107] The Product Guide file is missing.
INFO: [IP_FLOW-19-793] Syncing license key meta-data
```

Where "ultra96v2-platform" is the platform name in Vitis:



In the end, the xo file gets generated:



To install the DPU kernel template, follow:

https://adaptivesupport.amd.com/s/question/0D52E00007H6KQsSAN/why-application-template-of-dpu-kernel-rtl-kernel-is-not-exist-on-vitis-20221?language=en_US

It allows to create a project in Vitis easily.

A tutorial about how to use it in Vivado is at:

<https://github.com/Xilinx/Vitis-AI-Tutorials/tree/2.0/Tutorials/Vitis-AI-Vivado-TRD>

<https://xilinx.github.io/Vitis-AI/3.5/html/docs/workflow-system-integration.html>

<https://github.com/Xilinx/Vitis-AI-Tutorials/blob/1.4/Introduction/README.md>

https://github.com/Xilinx/Vitis-AI/blob/v3.0/dpu/ref_design_docs/README_DPU_Vitis.md

A tutorial about how to use the DPU in Vitis, is available at:

https://octavosystems.com/app_notes/vitis-ai-tutorial-part-4/