

Generative AI and AIoT (GenAIoT) Coding Skills Education

Lab Session 4.5

Object Inference with PYNQ-DPU on Ultra96-V2 Board

Overview:

In this lab, students will set up the Ultra96-v2 FPGA development and connect to it via serial communication with Putty (students may decide to use VSCode). We will then access the Jupyter server via a browser and Samba. Students will familiarize themselves with the board by running some Jupyter notebooks. Furthermore, we will install the Pynq-DPU to download the DPU overlay by installing pynq-dpu. The DPU notebooks will also be downloaded. Finally, students will run example AI applications such as Yolov3 and Resnet50 from the PYNQ-DPU provided example applications.

Required Hardware:

1. Avnet Ultra96-v2 FPGA board and accessories
2. SD card and SD card holder

Required Software:

1. Windows/Linux/MAC Host PC
2. Putty
3. Balena Etcher
4. A browser (Mozilla/Chrome etc)

Steps for the Lab Session:

1. Download and install Balena Etcher
2. Download PYNQ 3.0.1 image
3. Burn the PYNQ image into the SD card
4. Connect to the board via serial port
5. Connect to the board via Jupyter notebook
6. Try some example notebooks
7. Connect the board to the Wifi and Samba
8. Download pynq-dpu and notebooks
9. Run Yolov3 and Resnet50 inference

1.0 & 2.0 Download and Installed the Required Software

Note: We already downloaded and burnt the PYNQ image onto your SD card, no need to do that step.

1. Putty :

Download and install Putty: <https://www.putty.org/>

2. Balena Etcher :

Download and install Balena Etcher:

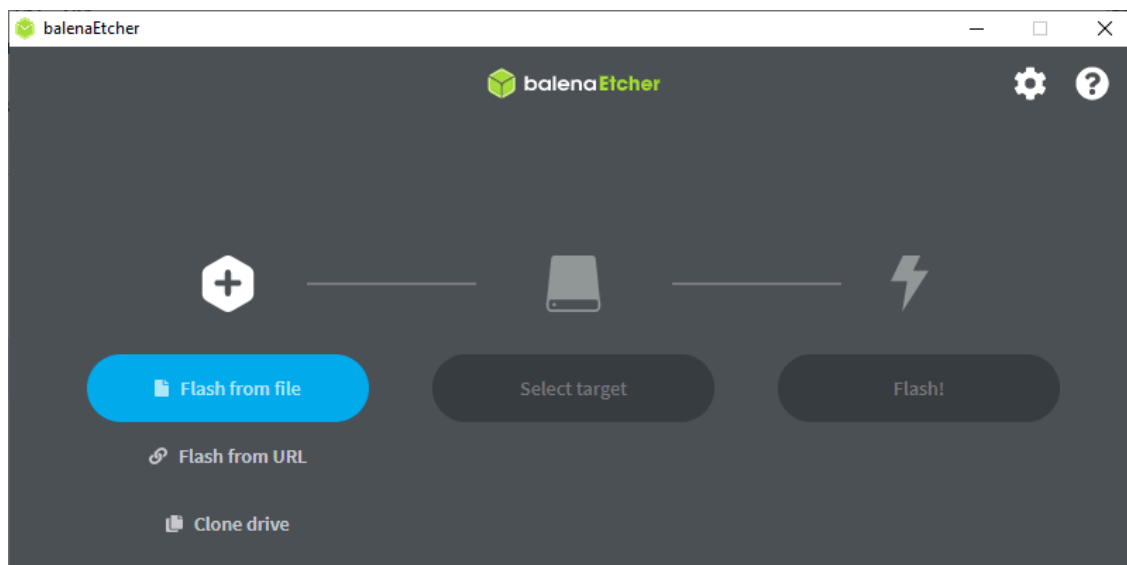
<https://github.com/balena-io/etcher/releases/download/v1.19.25/balenaEtcher-1.19.25.Setup.exe>

3. PYNQ 3.0.1 image:

Download the latest PYNQ image for Ultra96-v2: <http://www.pynq.io/boards.html>

3.0 Burn the PYNQ image onto the SD card

- Insert the SD card into the SD card holder.
- Open Balena Etcher.
- Click **Flash from File** and select the PYNQ image.
- Click **Select target** and select the SD card.
- Click **Flash !**, wait for it to flash and also validate.



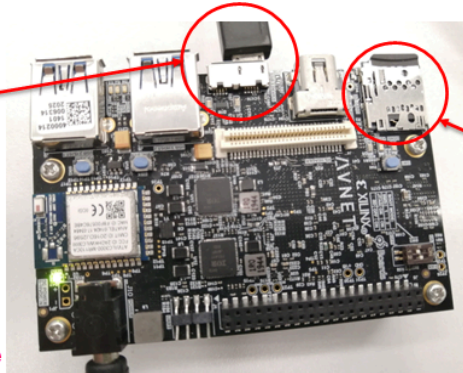
4.0 Connect to the board via serial port and Putty

- Insert the SD card into the SD card slot of the board
- Connect the power cable and the serial cable

Connect the board to the PC with a Micro-USB cable

Connect to the power here

Precaution: Make sure you turn OFF the electric plug before connecting the power cable!



Insert the Micro-SD card here

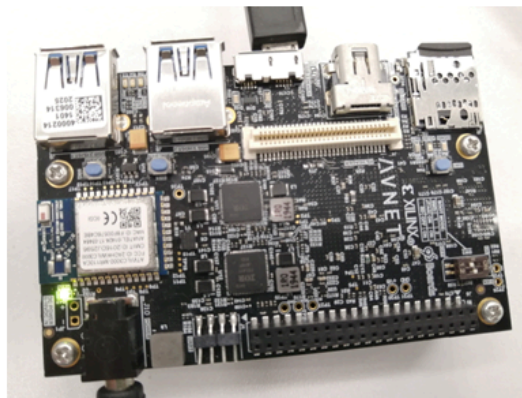
- Turn ON the board, wait until the blue DONE LED is ON

Turn ON:

After connecting the power cable to the board, turn on the board using the power button here

Turn OFF:

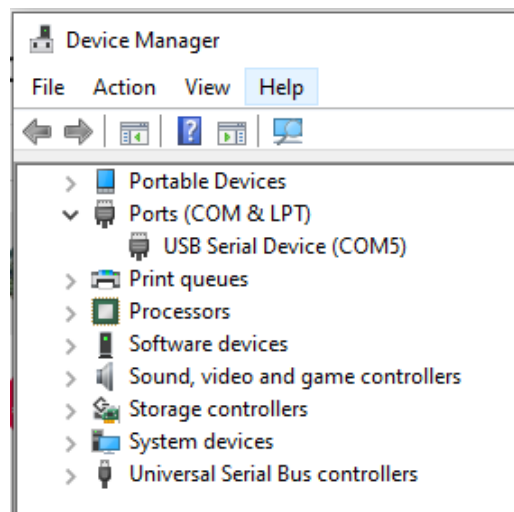
To turn off the board after using, press the power button for few seconds



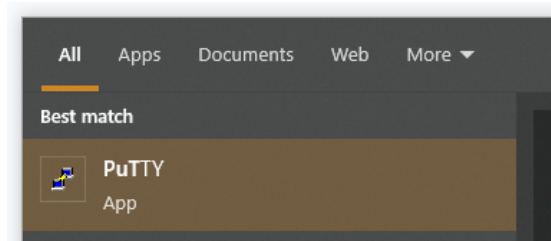
Note:

The green power LED (behind the power switch), and the red Init LED (near the SD Card socket) will turn on when the power button is pressed. After a few seconds, the red LED will switch off, and the green Done LED will switch on. The board will be ready after the blue and orange LEDs switch on.

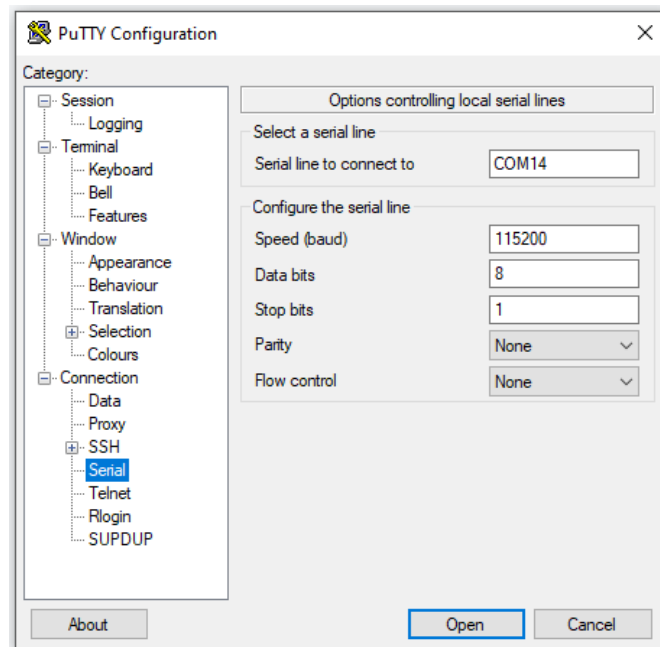
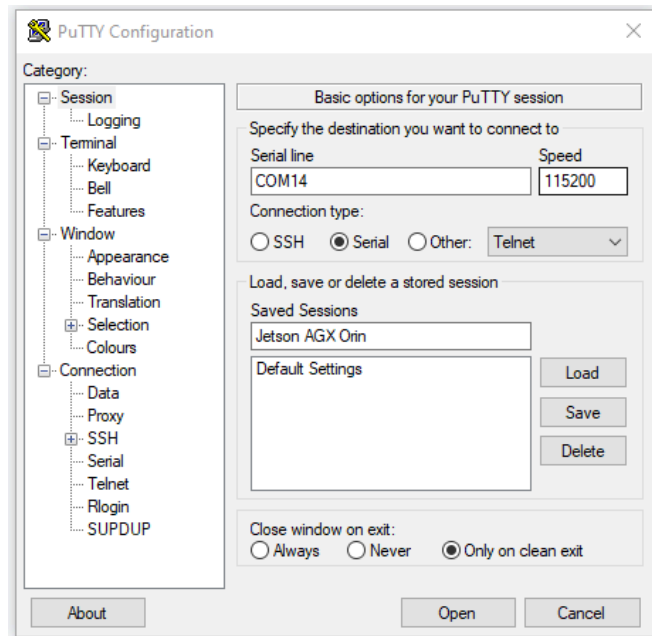
- Check device manager for the COM PORT number



- Open Putty and start the serial terminal



- Select the **Serial** option and enter the port number and the Baud rate (**115200**). Set **Flow Control** to **None** in the serial setting.



- Click Open. Press Enter if the board has already started.

```
COM5 - PuTTY
individual files in /usr/share/doc/*/copyright.

PYNQ Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the PYNQ Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

PYNQ Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Fri Oct 21 23:06:32 UTC 2022 on ttyGS0
No directory, logging in with HOME=/
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

(pynq-venv) xilinx@pynq:/$
```

- Check the ip address:

>> ifconfig

```
(pynq-venv) xilinx@pynq:/$ ifconfig
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 5183 bytes 376294 (376.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5183 bytes 376294 (376.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.3.1 netmask 255.255.255.0 broadcast 192.168.3.255
    inet6 fe80::2cfd:6bff:fe59:26cf prefixlen 64 scopeid 0x20<link>
    ether 2e:fd:6b:59:26:cf txqueuelen 1000 (Ethernet)
    RX packets 1094 bytes 127377 (127.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 75 bytes 14569 (14.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(pynq-venv) xilinx@pynq:/$
```

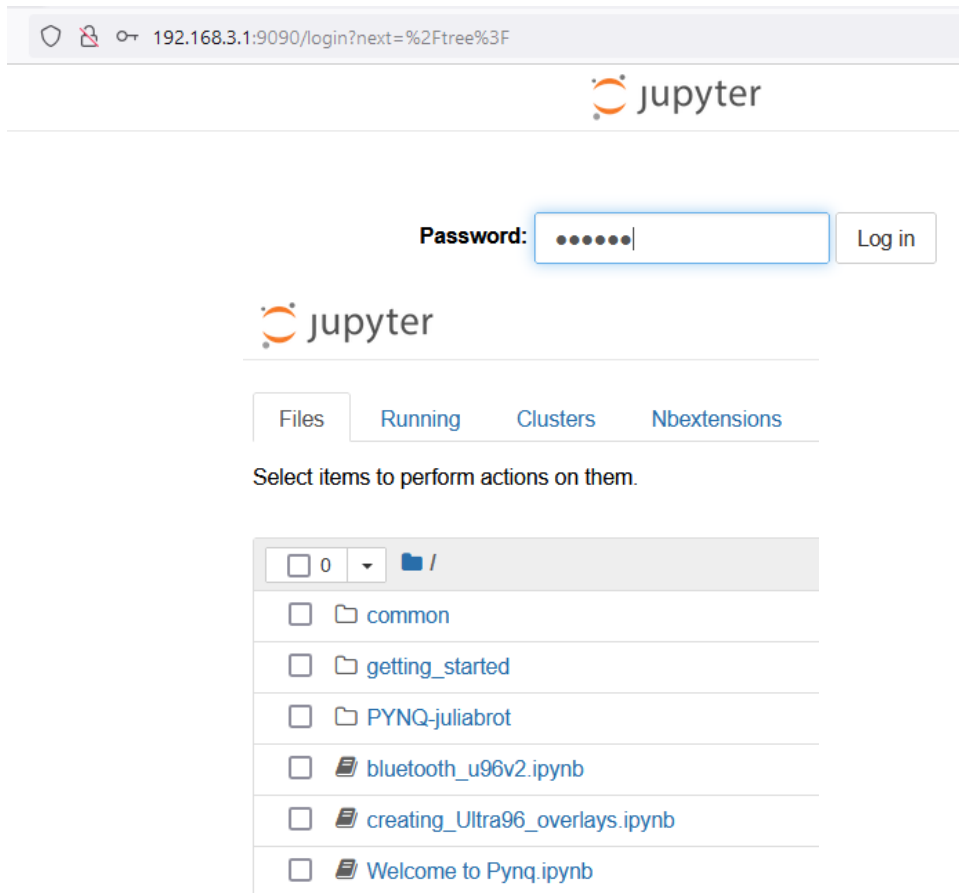
- Check the PYNQ version

>> pynq --version

```
(pynq-venv) xilinx@pynq:/$ pynq --version
PYNQ version: 3.0.1
Path: /usr/local/share/pynq-venv/lib/python3.10/site-packages/pynq
Git Id: 16022d5f2c61c7e5eld4aabcfc9b3e4c91b491b6
(pynq-venv) xilinx@pynq:/$
```

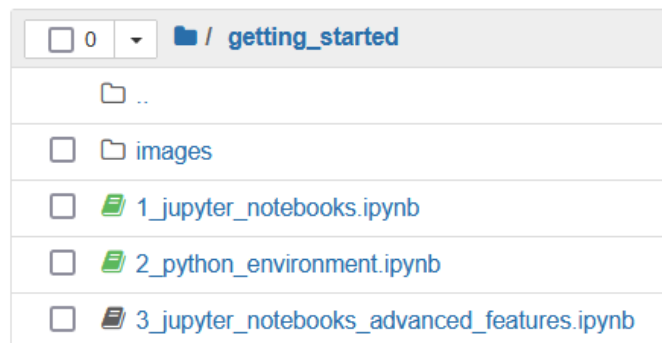
5.0 Connect to the board via Jupyter notebook

- Open a browser
- Enter the board ip **192.168.3.1** (192.168.3.1:9090)
- The Password is '**xilinx**'



6.0 Try running some example notebooks

- Open **getting_started** folder
- Explore and run the Jupyter notebooks

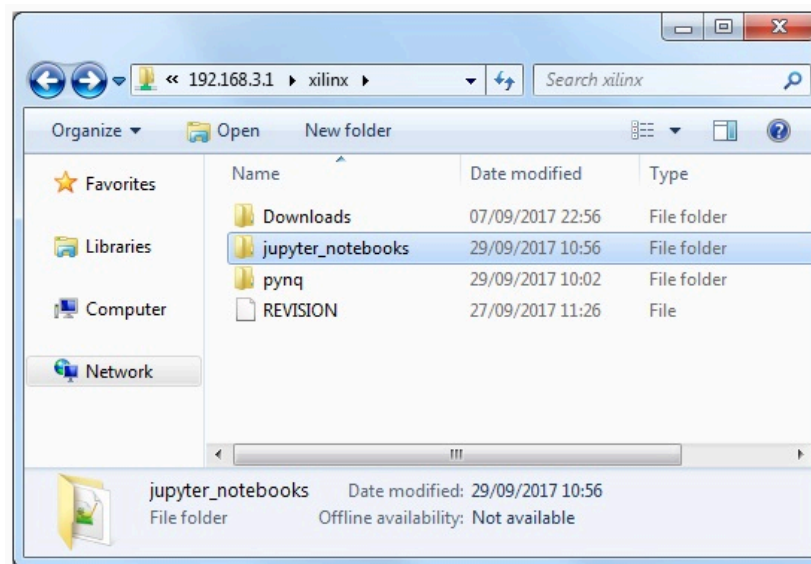


7.0 Connecting the board to Wifi and Samba

- Open the [wifi.ipynb](#) notebook in the **Common** folder
- Enter the SSID and Password
- Test the connection by pinging Google
 >> ping [www.google.com](#)

[Samba](#), a file sharing service, is running on the board. This allows you to access the PYNQ home area as a network drive. This can be a convenient way to transfer files, including bitstreams, to and from the board.

- To access the PYNQ home area in Windows Explorer type
 \\192.168.3.1\xilinx # If connected via USB Ethernet
- If prompted, the username is **xilinx** and the password is **xilinx**. The following screen should appear:

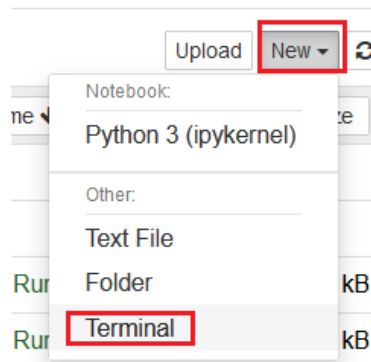


- To access the home area in Ubuntu, open a file browser, click Go -> Enter Location and type one of the following in the box:
 smb://192.168.3.1/xilinx # If connected via USB Ethernet
- When prompted, the username is **xilinx** and the password is **xilinx**

8.0 Download and install pynq-dpu and notebooks

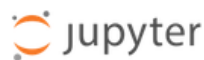
8.1 Download and install pynq-dpu

- To open a Jupyter Lab Terminal, Click New and Terminal



- Download the pynq-dpu by running the command

```
>> pip3 install pynq-dpu --no-build-isolation
```



```
root@pynq:/# pip3 install pynq-dpu --no-build-isolation
Collecting pynq-dpu
  Downloading pynq_dpu-2.5.1.tar.gz (1.4 MB)
  1.4/1.4 MB 344.8
  Preparing metadata (pyproject.toml) ... done
Requirement already satisfied: pybind11 in /usr/local/share/pyr
-dpu) (2.10.0)

Successfully built pynq-dpu
Installing collected packages: mnist, pynq-dpu
Successfully installed mnist-0.2.2 pynq-dpu-2.5.1
root@pynq:/#
```

Alternative pynq-dpu installation from serial or ssh:

If you are installing the package from an ssh or serial terminal instead of Jupyter Lab (e.g. using the usb network connection on the Ultra96 -- `ssh xilinx@192.168.3.1`).

Make sure you login as `root` (e.g., `sudo su`) and source the pynq profile scripts before installing the pynq_dpu package.


```
>> . /etc/profile.d/xrt_setup.sh
>> . /etc/profile.d/pynq_venv.sh
>> pip3 install pynq-dpu --no-build-isolation
```

8.2 Download pynq-dpu notebooks

Go to your jupyter notebook home folder and fetch the notebooks:

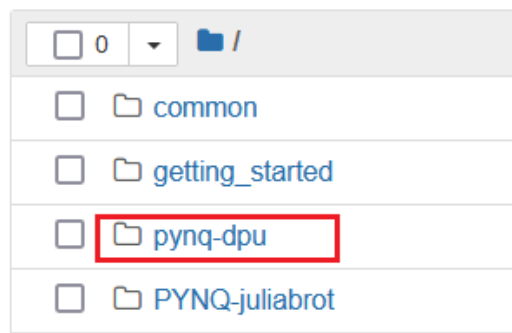
```
>> cd $PYNQ_JUPYTER_NOTEBOOKS
>> pynq get-notebooks pynq-dpu -p .
```

This will make sure the desired notebooks show up in your jupyter notebook folder.

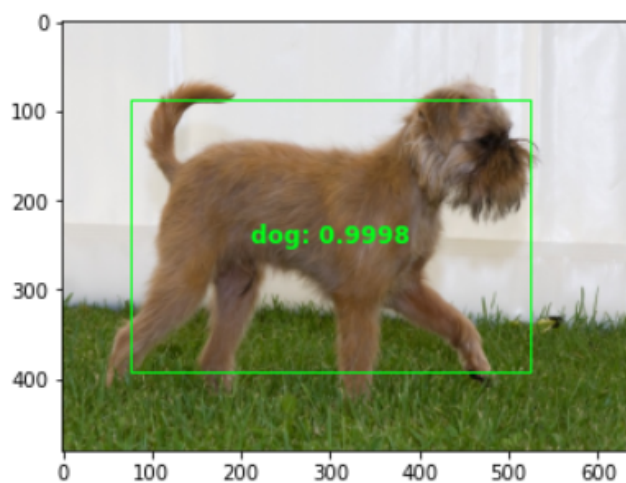
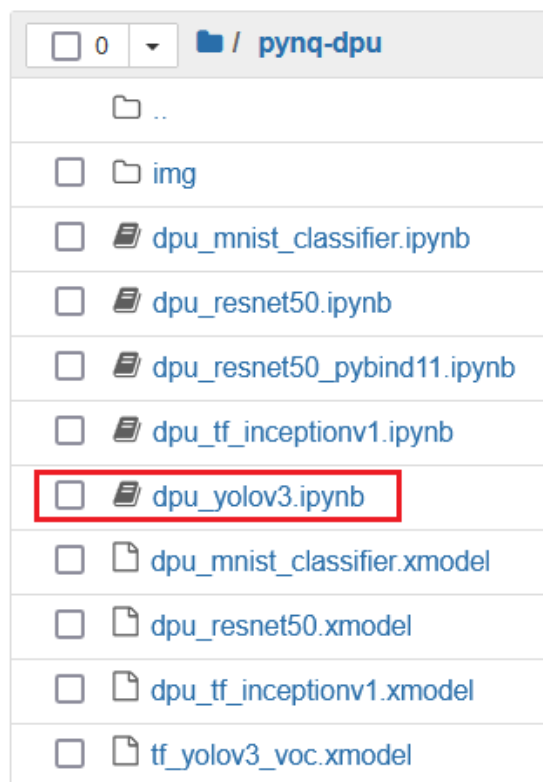
```
root@pynq:/# cd $PYNQ_JUPYTER_NOTEBOOKS
root@pynq:/home/xilinx/jupyter_notebooks# pynq get-notebooks pynq-dpu -p .
Delivering notebooks './pynq-dpu'...
root@pynq:/home/xilinx/jupyter_notebooks#
```

9.0 Run Yolov3 and Resnet

- From the Jupyter home, open **pynq-dpu** folder



- Open and run **dpu_yolov3.ipynb**



- Also Try Resnet50