Guide: Network Coding on bladeRF

1 GNU Radio Live Image and Automatic Installation (Recommend)

The simple way to implement a Network coding system on bladeRFs that is use the GNU Radio Live SDR Environment which is a bootable Ubuntu Linux DVD or USB drive image, with GNU Radio and third-party software pre-installed (common libraries, SDR's driver,...). The latest version of GNU Radio Live Image [1] is based on Ubuntu Linux `16.04.2 LTS`, 64-bit edition, and GNU Radio `3.7.11`.

Firstly, we have to download ISO image (because this project is deprecated, the link below is an our backup):

```
https://drive.google.com/file/d/1qtWKffJmvlreYkj4rgPaUUaj9estO_Np/view?usp=sharing
```

Then, we burn ISO image onto a USB flash drive using a utility such as the Rufus [2] (Windows) or Unetbootin [3] (Windows, MacOS, Linux). Finally, we shut down the computer, choose boot from this USB in BOIS and choose to Try Ubuntu without Installing when Ubuntu boot options appeared like Fig. 1.
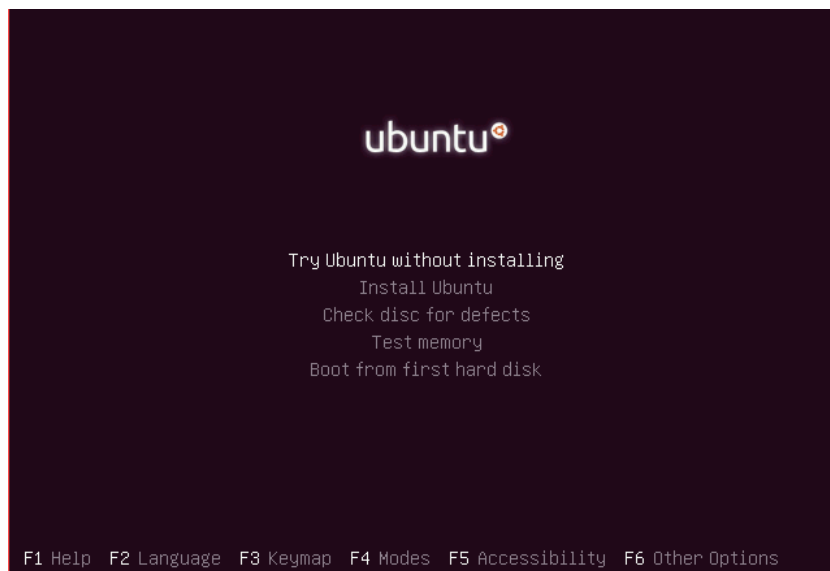

Figure 1: Boot menu of GNU Radio Live Image.

Download and extract file `SDR_NC.rar` onto path: `/home/ubuntu`, next, change the directory to the folder: `/home/ubuntu/SDR_NC/Software` in Files, after that, right click in this folder and choose `Open in Terminal`. Run following command on Terminal to install all Network coding libraries and new driver for bladeRF:

```
$ ./make
```

Just waiting until all softwares are installed, and then we can connect to the bladeRFs and load FPGA for them by this command:

```
$ bladeRF-cli -l /home/ubuntu/SDR_NC/Software/Software_for_BladeRF/hostedx115.rbf
```

At this step, we have all dependents software, now, opening GNU Radio by below command:

```
$ gnuradio-companion
```

Open any `.grc` file in SDR_NC folder, replace parameters by yourself and press "F6" or click to the icon "Execute the flow graph" on top of GNU Radio for running your system.

2 Manual Installation: GNU Radio, libraries, and bladeRF driver

2.1 GNU Radio Installation

GNU Radio is updated after each newer version and each version of GNU Radio is better compatible with some particular versions of Ubuntu OS. Therefore, choosing the version of GNU Radio should be taken carefully. Up to date, the newest version is GNU Radio `3.9.0.0`. This project is developed under Ubuntu 12.04 and GNU Radio 3.7.0.

The first to do is to open the Terminal window (press Ctrl + Alt + T), the environment for the installation. Before starting to install GNU Radio, the installed packages in Ubuntu should be updated:

```
$ sudo apt-get update
$ sudo apt-get -y dist-upgrade
```

Installation of GNU Radio requires the following steps:

```
$ sudo apt-get -y install git-core autoconf automake libtool g++ python-dev swig \
pkg-config libboost1.48-all-dev libfftw3-dev libcppunit-dev libgsl0-dev libusb-dev \
sdcc libsdl1.2-dev python-wxgtk2.8 python-numpy python-cheetah  python-lxmldoxygen \
python-qt4 python-qwt5-qt4 libxi-dev libqt4-opengl-dev libqwt5-qt4-dev \
libfontconfig1-dev libxrender-dev
```

To get GNU Radio v3.7.0, run following command:

```
$ git clone -b v3.7.0 https://github.com/gnuradio/gnuradio
```

Change the directory to the folder of your GNU Radio source code and run these commands:

```
$ cd gnuradio
$ mkdir build
$ cd build
$ cmake ../-DENABLE_BAD_BOOST=ON
$ make
```

Run the GNU Radio software self-check:

```
$ make test
```

Install GNU Radio for general use:

```
$ sudo make install
$ sudo ldconfig
```

Start and use GNU Radio semilar as section. 1.

2.2 Install bladeRF driver and Osmosdr

Osmosdr is needed for the use of GNU Radio with bladeRF, but the first we have to install latest version of bladeRF driver. The source code of bladeRF driver could be download: Install GNU Radio for general use:

```
$ git clone https://github.com/Nuand/bladeRF.git
```

Change the dictionary to the bladeRF path, and run these commands:

```
$ mkdir build
$ cd build
$ cmake ..
$ make
$ make test
$ sudo make install
$ sudo ldconfig
```

Download Osmosdr source code from:

```
$ git clone -b gr3.7 https://github.com/osmocom/gr-osmosdr
```

And simply, run following commands to install Osmosdr library support bladeRF:

```
$ cd gr-osmosdr
$ mkdir build
$ cd build
$ cmake ..
$ make
$ sudo make install
$ sudo ldconfig
```

2.3 Install Network Coding libraries

Network coding libraies includes gr-s4a, gr-CogNC, gr-display, gr-DNC, gr-DNC2, and gr-PNC4TWRN. Semilar as installation of Osmosdr block above, firstly, we go to dictionary path of each library and then run following commands:

```
$ mkdir build
$ cd build
$ cmake ..
$ make
$ sudo make install
$ sudo ldconfig
```

3 Extra Notes

We have some notes when you implement Network coding on bladeRF:

1. Connect more bladeRFs in a computer.

   Each bladeRF has a unique identify serial which could use to identify them, for example:

   ```
   $ bladeRF-cli -d libusb:serial=2a40 -i
   where 24a0 is the first four serial characters of a specific bladeRF.
   ```

2. Be careful with your data when using the Live Image method because all your data and setting will be lost after shut down the computer. It is great if you save data and flowgraph in internal storage or cloud.

3. Nuand provides a tool named "kalibrate-bladeRF" [4] that could help us calibrate bladeRF's VCTCXO using a nearby GSM Base Station. And you can find out a copy of this tool and backup calibrated file in SDR_NC/Software/Software_for_BladeRF. But be careful bladeRF has been pre-calibrated from the manufacturer with high quality, and if they still work fine or you have not read carefully about the note of "kalibrate-bladeRF", please do not try this tool, that can make your bladeRF inaccuracies.

References

[1] Corgan_Labs, *GNU Radio Live SDR Environment.* [Online]. Available: https://wiki.gnuradio.org/index.php/GNU_Radio_Live_SDR_Environment

[2] P. Batard, *Rufus - Create bootable USB drives the easy way.* [Online]. Available: https://rufus.ie/

[3] HabitLab, *UNetbootin.* [Online]. Available: https://unetbootin.github.io/

[4] *kalibrate-bladeRF*, Nuand. [Online]. Available: https://github.com/Nuand/kalibrate-bladeRF