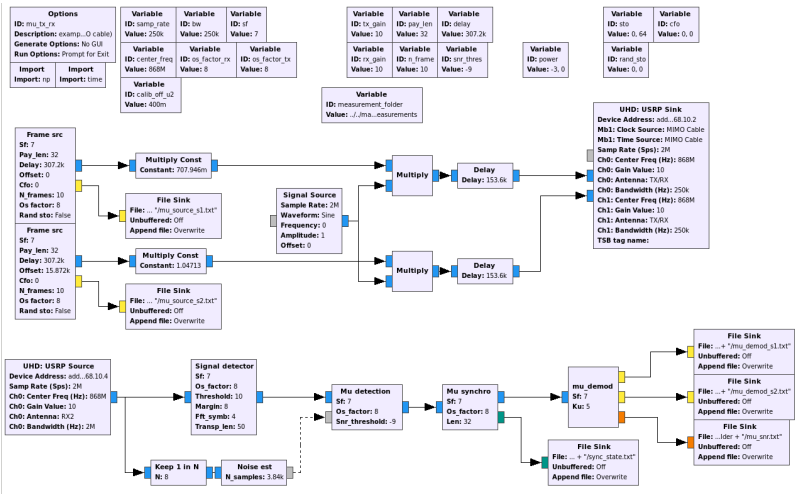


# A LoRa Multi-User Receiver based on GNU Radio



Download: GNU Radio LoRa Implementation

This is the GNU Radio software-defined radio (SDR) implementation of a LoRa receiver that can jointly decode the signal from two interfering LoRa users with the same spreading factor (SF). The implementation contains all the necessary receiver components to demodulate the colliding LoRa packets correctly even at very low SNRs.

In the GNU Radio implementation of the LoRa Tx and Rx chains the user can choose the transmission conditions of each user, such as sampling time offset (STO), carrier frequency offset (CFO), and power.

- In the Tx chain, the implementation contains all the necessary blocks to transmit colliding LoRa packets. Each packet consists of a preamble and a payload. Each user is subject to an individual CFO, time offset and transmit power.
- The receiver side is composed of:
  - A signal detection block that is able to detect when a LoRa packet is present.
  - A user detection block responsible of detecting and estimating the parameters of the incoming user, such as the STO, CFO, and power.
  - A synchronization block which realigns the samples to the strongest user for STO correction and applies the required CFO correction.
  - A demodulation block which recovers the symbol values of both users using a maximum-likelihood-based demodulation method.
- The implementation can be used for experimental performance evaluation of a two-user LoRa receiver at low SNRs.

### Reference

<https://arxiv.org/abs/2012.08173>

If you find this implementation useful for your project, please consider citing the aforementioned paper.

### How to use

- Installation:
  - Download the zip archive and extract it
  - The installation path can be set in CMakeLists.txt under #set destination.(default: home/lora\_sdr)
  - Similarly to any GNU Radio OOT module, it can be build with:

```
cd [path to gr-lora_sdr]/build
cmake ../
make
(sudo) make install
```
  - The new blocks can be loaded in gnuradio-companion(GRC) by adding the following lines in home/.gnuradio/config.conf (If this file doesn't exist you need to create it):

```
[grc]
local_blocks_path=path_to_the_downloaded_folder/gr-lora_sdr/grc
```
  - The custom blocks necessary for the multi-user transceiver can be found under the LoRa\_mu section.
- Usage:
  - The script /gr-lora\_sdr/apps/setpaths.sh add the pythonpaths required to run the generated python files **for the current shell**

**process.** It has to be adapted accordingly to the installation folder, and should be sourced with: `source setpaths.sh`. It can be sourced automatically on shell start by appending: `source [path to the script]/setpaths.sh` to `~/bashrc.sh`.

- An example of a two interfering user flowgraph can be found in the `gr-lora_sdr/apps/two_users/`.
- Set the `measurement_folder` variable in GRC to a desired location. (Default: `gr-lora_sdr/matlab/measurement`).
- Generate the python script corresponding to the flowgraph in GRC.
- In a terminal navigate to `gr-lora_sdr/apps/two_users/`
- Execute the generated script using:  
`./mu_tx_rx_(simulation).py`
- A matlab script interpreting the results can be found in: `gr-lora_sdr/matlab`. This script compare the demodulated symbols with the reference ones, based on a frame ID present in the beginning of each payload.

- Frequent errors:
  - “ImportError: No module named lora\_sdr” under other distribution than Redhat7:
    - This issue comes probably from erroneous PYTHONPATH and LD\_LIBRARY\_PATH set in the script `setpaths.sh`.
    - The paths might need to be changed to:  
`~/lora_sdr/lib/python2.7/dist-packages`  
`~/lora_sdr/lib`
- Requirements:
  - Gnuradio 3.7
  - python 2
  - cmake
  - swig
  - libvolk
  - UHD

In the same section

Embedded DRAMs Emulation Platform

GEMTOO: A Gain-Cell Embedded DRAM Modeling Tool

Path Metric Sorters for SCL Decoding of Polar Codes

A LoRa Multi-User Receiver based on GNU Radio

LoRa PHY based on GNU Radio

A Multi-User LoRa Receiver Using Soft SIC Implemented in MATLAB

About	Education	Research	Innovation	Campus
Who we are	Bachelor	Research Domains	Innovation Initiatives	Services & Resources
Associated Campuses	Master	Faculty Members	Industry Collaboration	Library
Facts	Doctorate	Awards & Prizes	Startup Launchpad	Restaurants, Shops & Hotels
Presidency	Continuing Education	Collaborate With Our Researchers		Security, Prevention & Health
Vice Presidencies	International	EPFL Research Facilities		Sports
Working at EPFL	Teaching	Access Our Technologies		Community & Support
Recruiting EPFL Talents	Admission	Research Ethics		Chaplaincy
News & Media	Study Management	Research With Animals		Events
Sustainability	Educational Initiatives	Meet the Support Units For Research		Arts & Culture
Equality & Diversity	Education & Science Outreach	Services to EPFL Laboratories		Associations
Respect		Open Science		Visit EPFL
Philanthropy				Mobility & Travel
EPFL Alumni				
Schools & Colleges				
■ School of Architecture, Civil & Environmental Engineering ENAC	■ School of Basic Sciences SB	■ School of Engineering STI	■ School of Computer & Communication Sciences IC	■ School of Life Sciences SV
■ College of Management of Technology CDM	■ College of Humanities CDH			

