A Cognitive Radio Multimedia Network Testbed for Multimedia Communication

Concept Design Review

Eric Pires

Jeffrey Kobza

Kevin Francois

Tobias Young

University of Massachusetts Dartmouth

16 November 2017

Sponsored by: Dr. Honggang Wang

Advised by Dr. Honggang Wang, Dr. Dayalan Kasilingam and Dr. Liudong Xing

Table of Contents

Updated System Description and Overview

The purpose of this project is to test and develop a working network testbed utilizing Software Defined Radio (SDR) technologies as the primary means of data communications. The testbed will be the underlying system for further research within the scope of radio communications. The primary goal specified by the customer is to achieve reliable data communications between two systems connected to this network. To do so, the testbed must contain a programmed transmitter (TX) and receiver (RX) that is capable of sending and receiving raw data reliably from one point to another. The range of this communications network must work within the confines of a single enclosed area.

The **Co**gnitive Radio **M**ultimedia N**e**twork **T**estbed for multimedia communication (COMET) comprises of a network of Universal Software Radio Peripherals (USRP) configured to transmit data within the testbed. This is done through the USRP hardware Driver that allows users on a Windows or Linux System to upload firmware onto the USRP’s RISC softcore microprocessor. The device accommodates many solutions for application development, including open-source platforms such as GNU Radio. This software works with the UHD drivers to provide a high-level design interface for Radio Frequency (RF) applications. This software is easily available on Unix based systems, which provide many needed dependencies missing in the Windows installation.

The USRP N210 is one of the few Network-based USRP devices that utilizes an Ethernet interface. Though this is limited to 1Gbps speed, the Ethernet interface can be utilized within a network switch for use within a central network testbed. The device also contains a Multiple Input Multiple Output (MIMO) port that can be utilized for increased spectral efficiency and linkage reliability [1]. Figure 1 demonstrates the proposed system diagram for COMET:

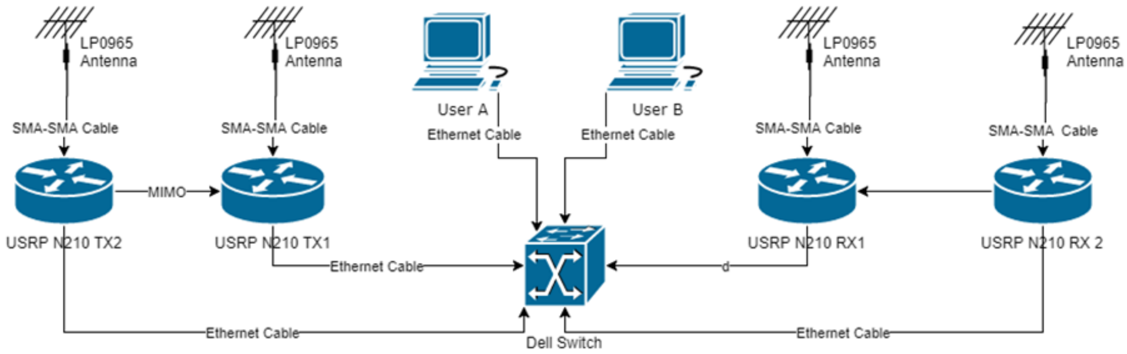


Figure 1: Network Map Version 2.0

Research on the TX/RX composition was done after the system requirements review. During this stage, GNU Radio source code was analyzed and executed to understand the system functionality of a transmitter/receiver. In a general scope, a transmitter is a system that takes in a data source and modulates the data into a radio signal capable of being broadcast over the RF spectrum. Before this modulation, the signal must first be encoded to a transmittable format. Encoding is dependent on the source data which can be anything from a simple text file to an audio file. The scope of this project does not include any particular encoding schemes. Therefore, complex encoding methods like MPEG or H.264 are not required by the customer. Encoding in this case is ensuring sample data types match to their high-level design blocks on the GNU Radio companion application. Following modulation, the signal will most likely need some amplification to strengthen it prior to being broadcasted. This leads to the final step, the actual transmission through the connected antenna. Figure 2 demonstrates this system flow in order along with a sample GNU Radio diagram.

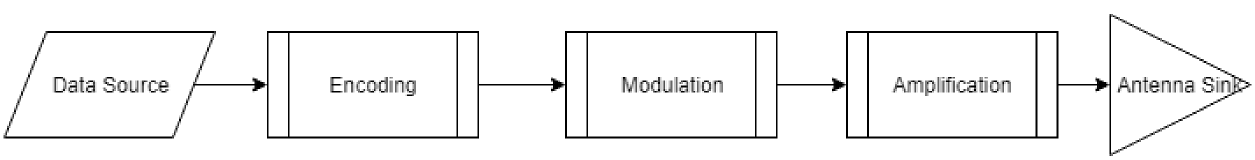


Figure 2a: TX System Flowchart

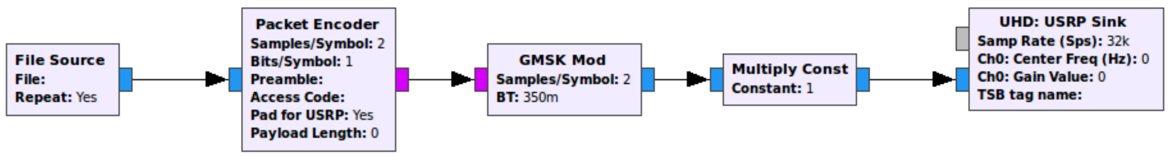


Figure 2b: Sample GNU Radio TX Flowchart

Likewise, the receiver operates similarly with reversed roles. However, this time a new component is included. A filter is required to parse any unwanted received data within a particular range. This signal is then decoded and demodulated to revert back to its original raw form from the transmission stage. Figure 3 demonstrates this system flow in order along with a sample GNU Radio diagram.

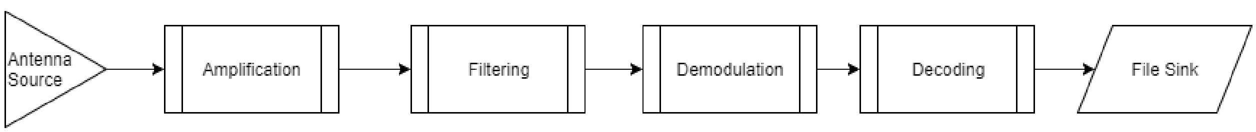


Figure 3a: RX System Flowchart

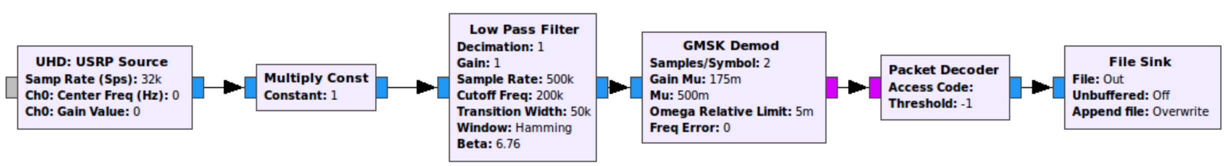


Figure 3b: Sample GNU Radio RX Flowchart

Updated Requirements

Customer Requirements

As discussed in the last report, there are steps and procedures that must be followed for the implementation of this system. Upon further research and communication with the customer, these requirements were elaborated. Requirement number two was added to further detail the purpose of the testbed. This purpose is to utilize the testbed for raw data transmission (TX) and reception (RX). These customer requirements are detailed below.

No.	Customer Requirement	Requirement Description
1	Setup network testbed for multimedia communication	Using the USRP N210 SDR device, design and implement a reliable duplex communication network capable of transmitting and receiving data from one device to another over a common but reliable frequency.
2	Use testbed to transmit and receive a raw data sample	Testbed must be able to send and receive raw data from one device (Client) into another (Server)
3	Expand the network testbed by configuring multiple devices to work synchronously	Improve on the base testbed by adding multiple transmission and receiving components that can communicate synchronously. Use a network switch to connect devices into a central network hub

Table 1: Tabular Representation of Customer Requirements

Engineering Requirements

Likewise, with further research came more explicit engineering requirements. The majority of these requirements involved the analysis and construction of a transmitter and receiver through GNU Radio. Below is Table 2, which outlines the procedures necessary for the testbed implementation.

Customer Requirement No.	Engineering Requirement No.	Engineering Requirement Description	Justification/ Comments	Test Method
1	1.a	Map out a network in which one device transmits over a frequency within the S-band ranges of 2-2.4Ghz and another receives over the same frequency	Transmission errors expected over common frequency channel, reliability aspect covered in second semester.	Inspection

	1.b	Install the USRP Hardware Driver on a host PC to detect and update firmware within the USRP devices	The USRP N210 communicates to a host environment via ethernet, and must utilize the hardware drivers distributed by Ettus Research for proper PC communications on either a Windows or Linux environment.	Test
	1.c	Using GNU Radio, implement a Transmission Protocol that can encode a data input and broadcast a radio signal within the given frequency point through the antenna port of the USRP	In order for data to be wirelessly transmitted, a data source must be encoded and modulated to a carrier waveform to be broadcast over the frequency spectrum	Demo/ Analysis
	1.d	Using GNU Radio, implement a Reception Protocol that can demodulate a radio signal received from the antenna port if the USRP and decode a raw data output.	In order to receive wireless data transmissions, a USRP data source must be filtered and demodulated to recover the original data stream, then decode it to its original format.	Demo/ Analysis
2	2.a	Run Full-Duplex benchmark to test working specifications for sample rate and frequency communication	Benchmark will provide information that will confirm the hardware’s functionality given a sample rate for Tx and Rx along with a supported frequency. Test included in UHD installation	Test
	2.b	Run GUI receiver plot and waveform generator to further test and confirm radio communications.	GUI programs provided in GNU Radio installation will provide a testable input transmission and GUI spectrum plot to view the received input.	Test/ Analysis
	2.c	Develop a custom transmitting and receiving protocol for data transmission	Following the previously mentioned system diagram, utilize GNU Radio to map out a transmitter and receiver protocol	Test/ Demo

3	3.a	Add additional transmitters and receivers through MIMO configuration.	MIMO (Multiple Input Multiple Output) allows for optimization of network speed by synchronizing device computation between multiple units rather than depending on one.	Test/ Inspection
	3.b	Attach main transmitter and receiver to Ethernet switch to connect all devices within network testbed. Benchmark sampling results of transmitted and received packets	Number of added devices will increase throughout the year; therefore a network switch will be utilized to accommodate all device traffic within the network.	Demo/ Analysis

Table 2: Tabular Representation of Engineering Requirements

Constraints and Applicable Standards

The requirements review discussed the frequency range restrictions imposed by the Federal Communications Commission (FCC). As further research and testing was done, a few more constraints were observed. The LP0965 antenna frequency range varies from 850 – 6500 MHz. This broad range falls within both the Ultra High Frequency (UHF) band and Super High Frequency (UHF) band. This must also consider the specifications of the SBX Daughterboard, which has a range from 400 to 4400 MHz Therefore, the system will be able to cover any frequency ranging from 850 to 4.4GHz. Although it limits the full capabilities of the antenna, this range is more than enough for suitable modern data communications.

Another noted constraint was the support of the development software. GNU Radio works best with a Fedora or Debian Linux distribution as discovered during through trial and error. The official installation documentation offers a simple build script that installs all necessary dependencies and packages for GNU Radio as well as the latest compatible UHD drivers [2]. The binary installers provided for Windows systems is not sufficient compared to its

Linux counterpart. Therefore, the Ubuntu Operating System is currently being used to configure and program the USRP devices.

Overview of Alternative Solutions

Modulation

There are several modulation schemes that could potentially work in conjunction with the transmitter or receiver schematics for the USRP N210. Each has its advantages and disadvantages. A few of these include frequency-shift keying and phase-shift keying.

Frequency-Shift Keying (FSK) is a frequency modulation scheme in which digital information is transmitted through discrete frequency shifts or changes of a carrier signal, usually an analog carrier sine wave. The simplest method of FSK is Binary Frequency-Shift Keying (BFSK). This uses two binary logic states (“1” and “0”) represented by two different waves at specified frequencies. A modem can then convert this binary data to be then transmitted through cables or wireless media [3].

Another type of FSK is Minimum Shift Keying (MSK). In this case, the sharp transitions between logics “1” and “0” can potentially create signals with sidebands extending beyond the allotted bandwidths, which can cause interference to adjacent frequency channels [4]. To reduce these problems, MSK encodes each bit as a half sinusoid.

Phase-Shift Keying (PSK) is a digital modulation process which conveys data by changing the phase of the carrier wave. Like frequency shift keying, PSK has several types to modulate a message signal, one of which is Quadrature Phase-Shift Keying (QPSK). This modulation scheme can be used either to double the data rate compared with a BPSK system while maintaining the same bandwidth of the signal, or to maintain the data-rate of BPSK but halving the bandwidth needed [5].

Another type of PSK is Differential Phase Shift Keying (DPSK). This scheme modifies the transmissions so that each transmission also depends on the previous. Using this technique for demodulating the current signal in a signaling time period, the received signal in the previous time period can be used as a local carrier at the received end for demodulation [6]

As expected, there are benefits and issues to using each modulation scheme. Using a frequency-shift keying method would entail a high immunity to noise interference. An FSK scheme would also be easy to implement, although it is highly error prone. It's also not very optimal for large data transmission. Because of these issues, this modulation scheme is not really used in modern data communication systems.

For phase-shift keying, the modulation scheme is more difficult to implement, as it requires a larger bandwidth. It is the most power efficient however, and the PSK scheme, unlike FSK, is widely used in modern data communication systems. It's particularly suited to the growing area of data communications. PSK can allow for higher data rates to be carried within a given bandwidth, although with higher data rates, there comes a need for a better signal-to-noise ratio before errors start counteracting this improvement.

Another useful modulation scheme is Quadrature Amplitude Modulation (QAM), which is a form of PSK combined with amplitude keying [7]. Similarly, QAM doubles the data rate, but by taking in two amplitude signals. QAM is used in 4G and Wi-Fi communication systems and is the hardest to implement, as it's highly susceptible to noise.

Testbed

The construction of the testbed had to be looked at from multiple perspectives, as the device used has various capabilities. The USRP has both a transmitter/receiver circuit, as well as a circuit dedicated to receiving. This means that any one device could be used to transmit, receive, or do both.

The first alternative the team looked at was the possibility of using each of the devices as purely a transmitter or receiver. With this configuration, no one device would be performing more than one task, which minimizes the stress each device will handle. This also means simpler programs, as each program needs to tell the USRP to receive or transmit, never both.

This configuration scheme will also mean that each device is not using its maximum potential. The USRP is built to and is capable of running both its TX/RX circuit as well as its RX circuit simultaneously. This is also the minimum required functionality that the customer requested.

The team also looked at using each USRP device as a transceiver, meaning it would utilize both of its communication circuits, allowing a back and forth "dialogue" between any two given devices. This would use each of the devices to the fullest potential. The bidirectional data transmission would allow for faster data communication.

Unfortunately, there are a few drawbacks to the transceiver system. The bidirectional data path can cause more errors in the transmission system. This system will also be more of challenge to code. While this system would exceed the customer's requirements, the system itself is also outside of the main scope of the project, making it not necessary to complete the project.

Analysis and Selection of Alternatives

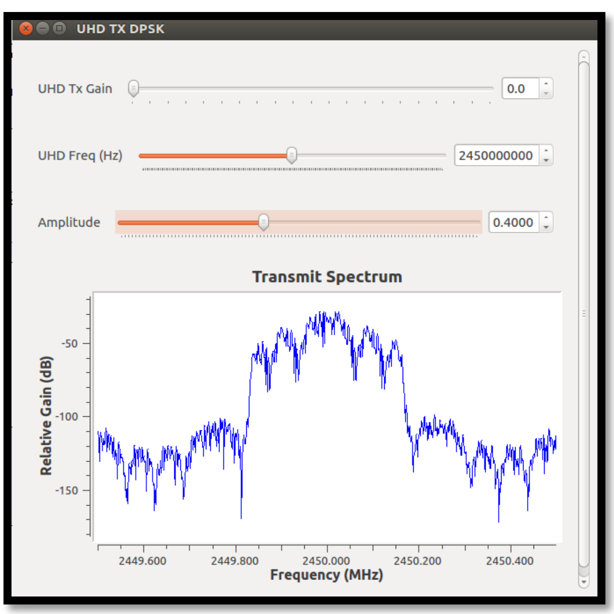
Modulation Analysis

After careful research and testing, the Phase Shift Keying modulation was decided upon as the selected implementation. The usage of Phase Modulation is much more complicated than

Amplitude and Frequency modulation, however the method is much more suitable for the role of the system in design. Amplitude modulation would be an inefficient method due to its sensitivity to noise and constant frequency allocation. By utilizing a PSK implementation, neither amplitude nor frequency is kept as a constant. This can lead for more efficient spectral allocation despite the complexity of the modulation scheme. Fortunately, GNU Radio offers many PSK implementation designs that can be modified to work for UHD compatible devices. One such implementation was selected for a transmission and reception test. In this test, Differential Phase-Shift Keying (DPSK) was utilized to send a random signal from one USRP device and receive it on another. This quick test was done in Lab using the following equipment and software:

- 2x USRP N210
- 2x LP0965 Antenna
    - Connected to Antenna ports on USRPs via SMA cable connector
- 2x Laptop PCs
    - Connected to Ethernet ports on USRP via Category 5e Ethernet cable
- GNU Radio Companion

Figure 4: DPSK TX GUI



One laptop was designated as the transmitter and utilized the TX/RX antenna port for data transmission (User A). The other was designated as a receiver and utilized the Rx2 antenna port (User B). The test began with executing the DPSK_RX module on User B's end system. The flowchart generated the python script that was uploaded to the USRP's firmware and ran a GUI interface for the user to view the selected spectral frequency over its relative gain. When executed, the waveform displayed the noise interception within the 2.45Ghz frequency in the lab environment. This captured frequency was uneventful prior to the transmission execution. This was then done on User A's end system, similarly generating and executing code onto the other USRP device. This also prompted a GUI with a similar plot and sliders for changing the Transmission gain, frequency, and amplitude. Figure 4 demonstrates the GUI interface of the transmission module.

Upon transmitting the signal seen above, the receiving end plotted a change in frequency activity. To ensure that it was indeed the transmitted signal causing this change, the Transmission was stopped momentarily while observing the receiving GUI. This forced the plot back to its original state of spectral activity. Launching the transmission once more displayed a clear change in spectral activity, matching the waveform seen in the TX GUI. The received signal can be seen in Figure 5.

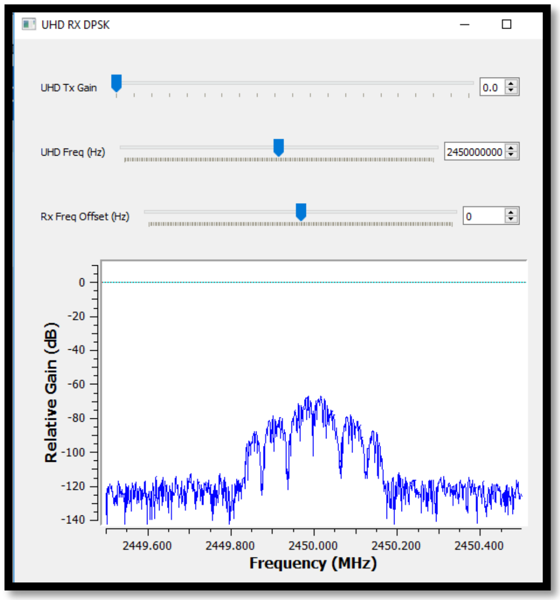


Figure 5: DPSK RX GUI

It is clear based on the Relative Gain spike and signal shape that the tranmission is being captured by User B's end. For further obervation, the amplitude slider was manipulated by User A to observe the spike in spectral activity on User B's side. The captured data in this example is a randomly generated waveform that can be changed into a new data source through the GNU Radio software. Overal, this test concluded that both USRP devices were properly configured and able to communicate reliably.

Testbed Analysis

To simplify the system in development, the TX/RX tesbed configuration was chosen as the base implementation. Despite the clear advantages of a bidirectional data communication testbed, the TX/RX testbed is easily distinguishable and configured. The XCVR alternative could lead to potential issues in data tranmission and falls outside the scope of the given customer requirements. This testbed was prototyped in the previously mentioned testing. Figure 6 shows the pototype of the system in development, utilizing a single tranmitter and receiver.

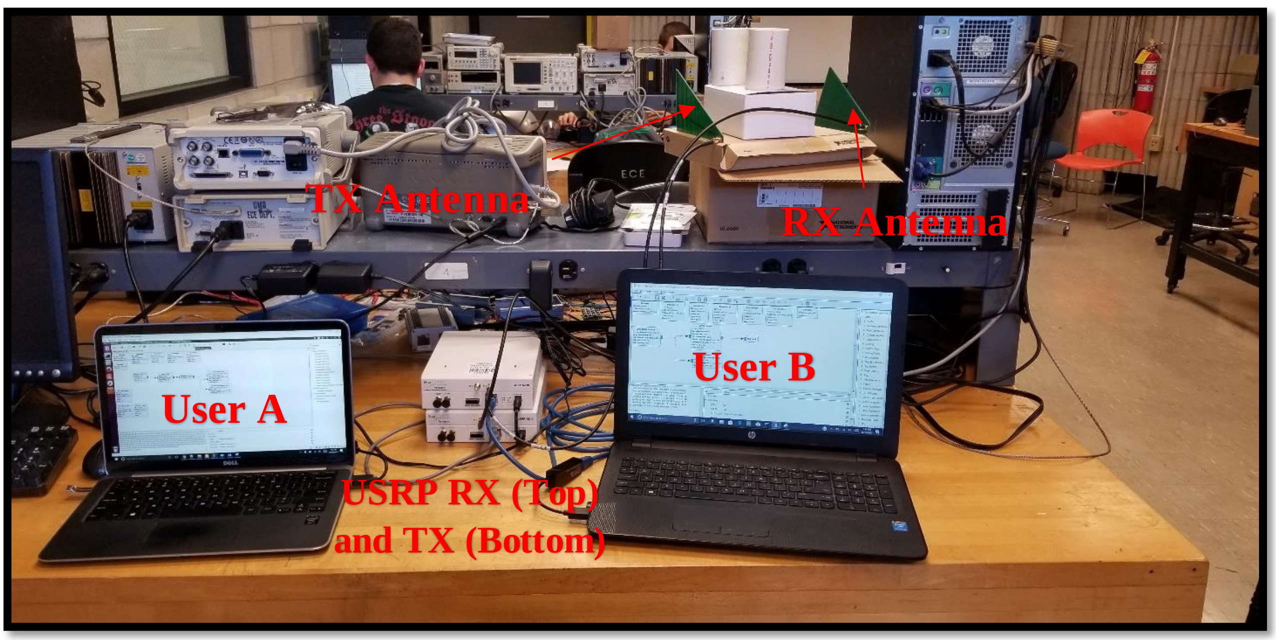


Figure 6: Prototype Testbed

Updated Plan and Schedule

The progress timeline originally conceived in the previous report has matched well with our actual progress. At the moment, the USRP devices have been configured and tested. Despite this success, the estimations were very off for different steps of the project. The biggest discrepancy was the time needed for configuration. The previous schedule had this step completed within an hour. Unfortunately, this took much more time to accomplish. The issue lies within GNU Radio and its Windows compatibility. The binary installer does not include all required dependencies, requiring more individual installations. The build scrip provided in the GNU Radio installation documentation provides all required dependencies for Linux systems. Once the installation was completed, the software design took less time than expected. This is due to the vast open-source library offered by GNU Radio to provide a better starting point for RF system programming. These changes are reflected in the Gantt Chart found in Table 3 below.

LEGEND	Task	Team Member(s)	Time	13-Oct	20-Oct	27-Oct	3-Nov	10-Nov	17-Nov	24-Nov	1-Dec	8-Dec
Completed	Draft Final Report for System Requirements	Eric	2 -4hrs									
In-Progress	Configure USRP drivers on host PC	Eric, Toby	10 hr									
	Configuration Documentation	Eric	2 hr									
	Write-up (Tx Research)	Kevin	2 hr									
	Write-up (Rx Research)	Jeff	2 hr									
	Write-up (MIMO hardware testbed Research)	Toby	2 hr									
	Research and Design a Transmission schematic on GNU Radio	Kevin, Eric	15 hrs									
	Research and Design a Receiver schematic on GNU Radio	Jeff, Toby	15 hrs									
	Concept Design Review	Eric	5 hrs									
	Integrate Transmitter/Receiver Phase 1 Testbed	Eric, Toby	5-10 hrs									
	Test and Document Transmitter/Receiver Phase 1 Testbed	Kevin, Jeff	5-10 hrs									
	Preliminary Design Review	Eric	10-20 hrs									

Table 3: Updated COMET Project Gantt Chart for Phase 1

References

[1] I. Poole, "MIMO Technology", *Radio-Electronics*, 2017. [Online]. Available: http://www.radio-electronics.com/info/antennas/mimo/multiple-input-multiple-output-technology-tutorial.php. [Accessed: 15- Nov- 2017].

[2] InstallingGRFromSource", *GNU Radio Wiki*, 2017. [Online]. Available: https://wiki.GNU Radio.org/index.php/InstallingGRFromSource. [Accessed: 16- Nov- 2017].

[3] "What is Frequency-Shift Keying (FSK)? - Definition from Techopedia", *Techopedia*, 2017. [Online]. Available: https://www.techopedia.com/definition/3004/frequency-shift-keying-fsk. [Accessed: 16- Nov- 2017].

[4] I. Poole, "What is MSK, Minimum Shift Keying Modulation", *Radio-Electronics*, 2017. [Online]. Available: http://www.radio-electronics.com/info/rf-technology-design/pm-phase-modulation/what-is-msk-minimum-shift-keying-tutorial.php. [Accessed: 16- Nov- 2017].

[5] R. Keim, "Understanding Quadrature Phase Shift Keying (QPSK) Modulation", *AllAboutCircuits*, 2017. [Online]. Available: https://www.allaboutcircuits.com/technical-articles/quadrature-phase-shift-keying-qpsk-modulation/. [Accessed: 16- Nov- 2017].

[6] P. Krishnan, "What are the advantages of differential phase shift keying (DPSK) over phase shift keying (PSK)?", *Quora*, 2017. [Online]. Available: https://www.quora.com/What-are-the-advantages-of-differential-phase-shift-keying-DPSK-over-phase-shift-keying-PSK. [Accessed: 16- Nov- 2017].

[7] I. Poole, "What is PSK, Phase Shift Keying", *Radio-Electronics*, 2017. [Online]. Available: http://www.radio-electronics.com/info/rf-technology-design/pm-phase-modulation/what-is-psk-phase-shift-keying-tutorial.php [Accessed: 16- Nov- 2017].