DEGREE PROJECT IN ELECTRICAL ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2019

Implementation and performance analysis of software defined radio (SDR) based LTE platform for truck connectivity application.

BILAL MAQSOOD

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Abstract

In today's modern era of technology mobile communication is evolving with a higher pace than ever before. New features and applications are added in the existing networks each day. The faster development pace requires a faster way to prototype and test the mobile communication standards/ applications to offer faster delivery to the end user. In traditional practices hardware updates and new features growth take long time to market implementations. Technology tends to be obsolete by the time it is to be launched to the market. The reason being long time required for hardware production. However, in the recent days the trend is changing with the emergence of open source cellular stacks to be used with affordable software defined radio (SDR) hardware platforms.

Long term evolution (LTE) open source cellular stacks along with the SDR technology are widely used in research these days. However, the performance and limitations of these SDR based open source cellular stacks needs to be explored. In this project a thorough study is performed to access the performance of an open source SDR based LTE user equipment (UE) software stack. A prototype of Category 4 LTE modem is implemented using the srsLTE application suite. Performance analysis is done by looking into the datarate, SNR and radio frequency (RF) characteristics of the implemented solution for multiple system bandwidth settings. A performance comparison is presented between the high performance SDR platform Universal Software Radio Peripheral x310 and the LimeSDR. The results show that the SDR technology is capable of handling wideband signals like LTE. The choice of SDR hardware platform and open source cellular stack depends on the application. The chosen solution for this project i.e. srsLTE performed well for LTE bandwidths 10 MHz and above in terms of downlink data rate. However, the radio frequency characteristics of selected SDR platforms do not comply fully with the 3GPP standard requirements.

Keywords: SDR, LTE, srsLTE, srsUE, USRP, LimeSDR

sammanfattning

I dagens moderna era av teknik utvecklas mobilkommunikation med en högre takt än någonsin tidigare. Nya funktioner och applikationer läggs till i befintliga nätverk varje dag. Den snabbare utvecklingstakten kräver ett snabbare sätt att prototypa och testa mobil kommunikationsstandarder / applikationer för att erbjuda snabbare leverans till slutanvändaren. I traditionell praxis tar hårdvaruuppdateringar och nya funktioner tillväxt lång tid att implementera marknaden. Teknik tenderar att vara föråldrad när den ska lanseras på marknaden. Anledningen är att det krävs lång tid för hårdvaruproduktion. De senaste dagarna förändras emellertid trenden med uppkomsten av cellulära stackar med öppen källkod som ska användas med programvarudefinierad radio (SDR) till överkomliga programvara. par Långtidsutveckling (LTE) med öppna källor, cellulära staplar tillsammans med SDR-tekniken används i stor utsträckning i forskning idag. Prestandan och begränsningarna för dessa SDR-baserade öppna källkodsstapelar måste dock utforskas. I detta projekt utförs en grundlig studie för att fååtkomst till prestanda för en open source SDR-baserad LTE-användarutrustning (UE) -programvarubunke. En prototyp av kategori 4 LTE-modem implementeras med srsLTE-applikationssviten. Prestandeanalys görs genom att undersöka egenskaperna för datarate, SNR och radiofrekvens (RF) för den implementerade lösningen för flera systembandbreddinställningar. En prestandajämförelse presenteras mellan den högpresterande SDR-plattformen Universal Software Radio Peripheral x310 och LimeSDR. Resultaten visar att SDR-tekniken kan hantera bredbandssignaler som LTE. Valet av SDR-hårdvaruplattform och öppen källkods cellstapel beror påapplikationen. Den valda lösningen för detta projekt, dvs srsLTE, fungerade bra för LTE-bandbredd 10 MHz och högre i termer av nedlänks datahastighet. Radiofrekvensegenskaperna för utvalda SDR-plattformar uppfyller dock inte helt 3GPP-standardkraven. par

Nyckelord: SDR, LTE, srsLTE, srsUE, USRP, LimeSDR

Acknowledgement

I would like to thank my thesis supervisor Dr. Peng Wang and examiner Associate Professor Marina Petrova for their continuous support and guidance through out the project. Further, I want to especially thanks my industrial supervisor from Volvo trucks Mr. Roman Iustin for giving me this opportunity and for providing continuous assistance during my thesis work.

I would also like to thank my opponents Michal Tomaszuk and Oskar Näslund for their thorough constructive feedback on my work and report. Finally I would like to thank all my professors, friends and family for their continuous support, guidance and encouragement.

Contents

1 Introduction **13**
- 1.1 Background . . . 13
- 1.2 Problem . . . 14
- 1.3 Purpose . . . 14
- 1.4 Goal . . . 14
- 1.5 Research Methodology . . . 14
- 1.6 Delimitations . . . 15
- 1.7 Thesis Outline . . . 15

2 Background **16**
- 2.1 SDR . . . 16
  - 2.1.1 Architecture . . . 17
  - 2.1.2 Trade-offs and Challenges . . . 17
    - 2.1.2.1 Antenna Choice . . . 17
    - 2.1.2.2 Local-Oscillator Leakage . . . 18
    - 2.1.2.3 Interference Cancellation . . . 18
    - 2.1.2.4 Security Issue . . . 18
    - 2.1.2.5 Sampling . . . 19
    - 2.1.2.6 Timing Synchronization . . . 19
- 2.2 Comparison of Available SDR . . . 19
- 2.3 LTE . . . 20
  - 2.3.1 Mobile Communication Generations . . . 20
  - 2.3.2 Architecture of LTE . . . 21
    - 2.3.2.1 E-UTRAN . . . 21
    - 2.3.2.2 Evolved Packet Core (EPC) . . . 22
    - 2.3.2.3 User Equipment . . . 22
  - 2.3.3 LTE UE Protocol Stack . . . 23
  - 2.3.4 LTE Channels . . . 24
  - 2.3.5 LTE Handshake . . . 25
    - 2.3.5.1 UE and eNB RRC connection setup . . . 25
    - 2.3.5.2 Authentication and security setup . . . 26
- 2.4 SDR based LTE Solutions . . . 27
  - 2.4.1 OpenLTE . . . 27
  - 2.4.2 srsLTE . . . 28
  - 2.4.3 Open Air Interface . . . 28
  - 2.4.4 lte-sidelink . . . 28
  - 2.4.5 Amarisoft LTE 100 . . . 28
- 2.5 Related Work . . . 29

3 Methodology **30**
3.1 Research Process . . . 30
3.2 Experimental Design . . . 30
3.2.1 Test Environments and Methods . . . 31
3.2.1.1 Quantitative Performance Measurement Setup . . . 31
3.2.1.2 eNB RF Performance Analysis Setup . . . 31
3.2.1.3 UE RF Performance Analysis Setup . . . 32
3.2.1.4 Basestaion Simulator Test . . . 32
3.2.2 Hardware . . . 33
3.2.3 Software . . . 33
3.3 Data Collection . . . 33
3.4 Reliability and validity of the data collected . . . 34
3.5 Data Analysis . . . 34

4 Implementation **35**
4.1 srsLTE Components . . . 35
4.1.1 srsUE . . . 37
4.1.2 srseNB . . . 37
4.1.3 srsEPC . . . 37
4.2 Building, Installing, and Running srsLTE . . . 37
4.2.1 Supporting Softwares . . . 37
4.2.2 Building and Installing srsLTE . . . 40
4.3 Configurations . . . 46
4.3.1 UE Configurations . . . 46
4.3.2 eNB Configurations . . . 46
4.3.3 EPC Configurations . . . 47

5 Results and Analysis **48**
5.1 eNB . . . 48
5.1.1 RF Power Output . . . 48
5.2 UE (USRP X310) . . . 50
5.2.1 Downlink Data Rate . . . 51
5.2.2 SNR . . . 54
5.2.3 RF Power Output . . . 54
5.3 USRP x310 vs LimeSDR . . . 55
5.4 RF Conformance Test . . . 57
5.4.1 Interference Analysis . . . 58
5.5 Testing with Commercial LTE basestaion simulator . . . 59

6 Conclusion and Future Work **61**
6.1 Discussion . . . 62
6.2 Future Work . . . 62
6.2.1 Data rate tests in Different Frequency bands and Using MIMO configurations . . . 62
6.2.2 CAT 4 LTE modem test with USRP based eNB . . . 63
6.2.3 RF Conformance tests . . . 63

A Installation Steps **67**
A.1 UHD Installation 67
A.2 Updating USRP image 67
A.3 LimeSuite Installation 68
A.4 SoapySDR Installation 68
A.5 srsLTE Build and Install 68

B **70**

C Throughput calculation LTE **74**

List of Figures

2.1 Basic Schematic diagram of SDR . . . 17
2.2 block diagram of basic radio [19] . . . 18
2.3 LTE Architecture [9] . . . 21
2.4 EPC Architecture (User and Control Plane) [8] . . . 22
2.5 UE Protocol stack [28] . . . 23
2.6 Steps for RRC connection setup [28] . . . 26
2.7 NAS Authentication and Security [28] . . . 27

3.1 Test bed setup - Quantitative Measurements (datarate and SNR) . . . 31
3.2 Test bed setup - eNB RF Performance Analysis . . . 32
3.3 Test bed setup - UE RF Performance Analysis . . . 32
3.4 Test bed setup - srsUE Compatibility Testing with CMW500 . . . 32

4.1 SRSLTE Modules [15] . . . 36
4.2 UHD Version check . . . 38
4.3 LimeSuite Version check . . . 38
4.4 SoapySDR Version check . . . 38
4.5 limeUtil –find (Output) . . . 39
4.6 SoapySDRUtil –find (Output) . . . 39
4.7 PCSC version check . . . 39
4.8 PCSC Attached card reader detection . . . 40
4.9 EPC initialized . . . 41
4.10 eNB initialized . . . 41
4.11 EPC console output with eNB connected . . . 42
4.12 UE Initialized . . . 42
4.13 srseNB console (UE Connected) . . . 43
4.14 srsEPC console (UE Connected) . . . 44
4.15 srsUE console After connection to the Network . . . 45
4.16 UDP Traffic on Tun_srsue interface - UE . . . 45
4.17 UDP Traffic on srs_spgw_sgi interface - EPC . . . 46

5.1 A figure that contains three subfigures . . . 50
5.2 srseNB Output Power Test using USRP . . . 50
5.3 UE Downlink Data Rate (USRP) - 1/2 . . . 52
5.4 UE Downlink Data Rate (USRP) - 2/2 . . . 52
5.5 Error percentage plot . . . 53
5.6 Average error percentage plot against PRB values . . . 53
5.7 Downlink SNR at UE . . . 54
5.8 UE RF output Power . . . 55

5.9 UE - LimeSDR Peak D/L Data Rate Comparison 55
5.10 Peak data rate error percentage USRP vs LimeSDR 56
5.11 UE - LimeSDR SNR Comparison 56
5.12 UE - LimeSDR Output Power test 57
5.13 UE Transmit Off 58
5.14 USRP & LimeSDR interference Analysis 59
5.15 srsUE soft SIM configurations for testing with CMW 500 59
5.16 UE Attached To CMW 500 60
5.17 srsUE PC/SC Configurations 60
5.18 UE Authentication Reject CMW 5000 60

6.1 Cat 4 modem test setup 63

List of Tables

2.1 LTE Logical Channels [2] 24
2.2 LTE Transport Channels (Uplink/Downlink) [2] 24
2.3 LTE Physical Channels (Uplink/Downlink) [2] 25

5.1 LTE BW and PRB Mapping 51
5.2 UE USRP x310 vs limeSDR RF Parameters [5] 58

B.1 **SDR Supporting LTE Frequencies** 71

Acronyms

3GPP Third Generation Partnership Project.

4G Fourth Generation.

ADC Analog to Digital converter.

AGC automatic gain control.

API Application Program Interface.

c-RNTI cell Radio Network Temporary Identifier.

CQI Channel Quality Indicator.

DAC Digital to Analog converter.

DL Downlink.

E-UTRAN Evolved Universal Terrestrial Access Network.

eNB Evolved NodeB.

EPC Evolved Packet Core.

EPS Evolved Packet System.

FDD Frequency Division Duplex.

FPGA Field Programmable Gate Arrays.

GbE Gigabit Ethernet.

HSPA High Speed Packet Access.

HSS Home Subscriber Server.

IF intermediate frequency.

IMSI international mobile subscriber identity.

IP Internet Protocol.

LNA low-noise amplifier.

LO Local Oscillator.

LTE Long Term Evolution.

MAC Medium Access Layer.

MCS Modulation Coding Scheme.

MIMO Multiple Input-Multiple Output.

MME Mobile Management Entity.

NAS Non Access Stratum.

OAI Open Air Interface.

OFDMA Orthogonal Frequency Division Multiple Access.

OSI Open Systems Interconnection.

PC Personal computer.

PCI Peripheral Component Interconnect.

PDCP Packet Data Convergence Control.

PHY Physical Layer.

PLMN Public Land Mobile Network.

PRB Physical Resource Block.

RF Radio Frequency.

RFFE Radio Frequency Front-end.

RLC Radio Link Control.

RRC Radio Resource Control.

RX Receive.

SDR Software Defined Radio.

SISO single input, single output.

SNR Signal to Noise Ratio.

SR Scheduling Request.

TBS Transport Block Size.

TDD Time Division Duplex.

TX Transmit.

UDP User Datagram Protocol.

UE User Equipment.

UHD USRP Hardware Driver.

UL Uplink.

UMTS Universal Mobile Telecommunication System.

USRP Universal Software Radio Peripheral.

Chapter 1

Introduction

Long Term Evolution (LTE) is the major cellular technology used worldwide for mobile communication these days. Since the advent of the technology in 2008 [9] more and more features are added to the technology each year. In view of the above, importance of field trials, experimentation and faster market implementation cannot be overstated. Traditionally these steps take long time mainly due to hardware lackings and spectrum licensing issues. However, in recent days the scenario is changing with the development of open source cellular stacks based on affordable Software Defined Radio (SDR) platforms. Hence, there is a need to comprehensively understand the performance, limitations, and interoperability of these systems with existing networks for large scale market implementation.

In the context of modern wireless communication systems, SDR is one of the most important technologies. SDR is a radio system which can be tuned to a wide band of frequencies, can implement different modulation/ demodulation schemes and different wireless communication standards in a same device by using re-configurable hardware and compatible software [14]. Large amount of data is transmitted using wireless communication technology each day including the data communication, video communication, voice communication etc.To modify the radio devices easily and cost effectively is becoming more critical. SDR technology provides the flexibility and cost efficiency, with far-reaching benefits for service providers, product developers and end users. The concept of SDR allows users to design and implement various wireless communication protocols in software and use SDR hardware platform as the radio front-end or transceiver. This removes the requirement of different radio hardware for different wireless communication standards. SDR provides flexible, upgrade-able, multi-standard and longer lifetime radio equipment for wireless communication infrastructure.

1.1 Background

Mobile communication technology is developing at a faster pace. If we look at the latest in use technology i.e. Fourth Generation (4G) or LTE, more and more features are being added each year in form of Third Generation Partnership Project (3GPP) releases [9]. This fast development pace is a challenge for the service industry as relevant hardware takes long time to reach the market. This arises the need in the industry to develop systems where minimum hardware up-gradation is required with the change or upgrades in the existing communication standards.

In a conventional automobile industry the technology upgrades take a long time to market implementation. The reason is more dependency on hardware replacement. As describe above, SDR can provide flexible and cost-effective radio solution by doing software upgrades instead of replacing the hardware module. So SDR can be considered as a potential replacement to conventional LTE connectivity modules in trucks or vehicles for faster technology upgrades.

1.2 Problem

SDR technology is widely being used for prototyping of wireless communication standards. Several open source implementations of cellular protocols over SDR are available online for researchers or product developers. Among these SDR based solutions LTE is the most desired solution for future research. However, comprehensive understanding of the performance and limitations of these solutions needs to be explored in order to exploit their use in commercial products. In this project we will analyze the performance of one of the today's most popular SDR based open source LTE solution by performing some controlled experiments in the lab.

1.3 Purpose

The purpose of the project is to investigate the potential of using SDR to enable more cost-effective radio platform life cycles by providing updates and additional functionality without requiring hardware modifications.

Furthermore, The new features of SDR make it as a very promising tool for replacing the existing communication devices in the vehicles. Due to its flexibility, SDR can be used as an all-in-one solution which can integrate multiple radio access technologies and offer many services. Moreover, it is also expected to reduce the complexity and cost of upgrading to future communications systems.

1.4 Goal

The goal of the project is to perform a feasibility study by investigating the possibility of creating a SDR based LTE Category 4 modem for trucks connectivity based on generic Radio Frequency Front-end and SDR architecture. Further more, to draw a performance comparison of the implemented solution between the high performance Universal Software Radio Peripheral (USRP) x310 and some relatively cost effective option for large scale market implementation.

1.5 Research Methodology

In this project different approaches of research methodologies will be used, in the first phase literature based study and analytical research approach will be used to investigate the use of SDR technology for providing LTE connectivity in trucks using generic SDR platform. Next a comparison will be made among the available LTE standard solutions over SDR and a suitable solution as per requirements will be chosen for the implementation. In the second phase along with the first two research

approaches empirical and quantitative research approach will be used to analyze the performance of the overall system. The performance metric includes down-link data rate measurement, Signal to Noise Ratio (SNR) and Radio Frequency (RF) characteristics. A comparison will be drawn by implementing the solution on a high performance USRP and a relatively cost effective SDR platform.

1.6 Delimitations

The delimitations of this project are both hardware and software based. In the hardware side specialized equipment like LTE signal analyzer, an-echoic chamber, reverberation chamber and some other specialized hardware are required to do a detail RF conformance and performance testing [5]. However, in this project some basic RF performance/ characteristics will be observed using a spectrum analyzer in a lab. Secondly, as we know that 3GPP has included functionalities like NB-IoT and (intelligent transport system) ITS in their recent releases but all these functionalities are not included in this project.

1.7 Thesis Outline

The thesis is structured as follows. Chapter 2 provides the comprehensive background information along with hardware/ software selection and related work information. Detail methodology and test setups are explained in Chapter 3. Chapter 4 describes the detailed implementation steps. Detail results and analysis is presented in chapter 5.Finally, conclusion and future work details are explained in Chapter 6.

Chapter 2

Background

This chapter gives the necessary theoretical background about the related areas. Section 2.1 presents a brief introduction of SDR technology, followed by a comparison analysis of available SDR platforms that can be used in this project in section 2.2, next is the description of LTE technology in section 2.3 and lastly a comparison of existing LTE solutions over SDR in section 2.4.

2.1 SDR

The SDR is an emerging concept in the area of wireless communication systems. It defines a radio communication system in which signal processing tasks are performed using a software. Whereas, typically these tasks are performed by specialized hardware.

SDR technology has been an active research topic both in academia and industry for more than a decade now [30, 19]. If we go into the history of wireless communication, the concept of "Software Radio" was first presented by J. Mitola in 1991. During the early 1990's, the first large scale SDR, SPEAKEasy and Joint Tactical Radio System were exploited by US Military. Furthermore, with the development of powerful signal processing chips in year 2000, most of the existing SDR platforms were developed. Today one of the most commonly used SDR hardware platforms is the USRP by Ettus Research [30].

SDR is a radio that is more adaptable to different physical layer formats and wireless protocols. SDR is a multi-band, multi-mode radio with dynamic capability defined through software covering all layers of the Open Systems Interconnection (OSI) protocol stack [19].

Before the emergence of SDR technology radio system developers or researchers had to built separate hardware radio platform for each technology supporting a particular set of frequencies. However, with the emergence of SDR technology the issue has been addressed as it provides a generic platform to implement different radio technologies operating in wide range of frequency bands. In a typical scenario, the capabilities of a wireless communication device are fully dependent on the hardware support which is not a very efficient way as it cannot cope up with the faster developments in the technology. However, in SDR based system, wider range of wireless protocol functions or capabilities depend on components which are re-configurable through a software. Thus, the term software defined radio generally refers to a radio that derives its flexibility through software while using a static hardware platform.

Hence, some major advantages of SDR are flexibility, easy adaption and using the same hardware to implement different communication protocols [30].

2.1.1 Architecture

Figure 2.1 shows the basic schematic diagram of a typical SDR. In an SDR based communication system first there is a RF front-end to transmit and receive radio signals. At this analog stage the received signal is converted into an intermediate frequency (IF) frequency which has much lower bandwidth as compared to the original received signal. Than the second stage is analog to digital conversion (ADC) or vice versa as the signal processing is being performed using the general purpose processors in digital form. Next part is the Field Programmable Gate Arrays (FPGA), the main purpose is to perform the Digital Up Conversion (DUC) and Digital Down Conversion (DDC). Apart from this FPGA serves as the interface between the host and SDR by converting the Peripheral Component Interconnect (PCI) traces into a specific interface format used between host and the SDR. Lastly, all the base-band processing, modulations of the signal on the transmit path and demodulation of the signal on the receiving path is done on a host Personal computer (PC) having suitable processing power. [19].

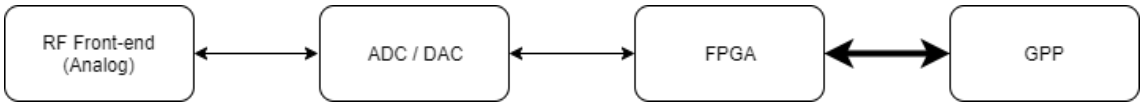


Figure 2.1: Basic Schematic diagram of SDR

2.1.2 Trade-offs and Challenges

As described in [19] and [30], SDR has a lot of benefits in terms of flexibility, software re-usability, easily Upgrade-able, multi-band/multi-mode operations, different standards (AM/FM/DAB/LTE) can coexist, enhances facilitates for experimentation, reduce life-cycle costs etc. However, there are some trade-offs/ challenges which one should keep in mind while choosing the right SDR platform for achieving the desired functionality.

2.1.2.1 Antenna Choice

First and foremost as described in [19] is the choice of antenna. Although SDR can support a range of frequencies supporting different communication protocol. However,there is a fundamental trade-off between the beam width and gain of the antenna.

Designing an antenna supporting wide range of frequencies could be quite challenging. The solution as mentioned in [19] to overcome this challenge is the use of Multiple Input-Multiple Output (MIMO) and tunable antenna implementation within an SDR. Furthermore, an electronic circuit like 'antenna tuner' can be used to connect the antenna to the rest of the circuit. They are optimized for different antennas and must be matched for optimal power performance. It improves power

delivery to the antenna under poor antenna matching condition. However, this requirement complicates the radio design and restricts the implementation of systems with many different frequency ranges on the same SDR platform.

As LTE supports different frequency band worldwide, so choosing a right kind of antenna for an SDR based application is critical.

2.1.2.2 Local-Oscillator Leakage

Local Oscillator (LO) leakage and spurious signals can interfere the desired signal thus distorting the desired information. Figure 2.2 shows the schematics of a basic RF part of an SDR. We can see that firstly a band-pass filter is used to eliminate all out-of-band signals, than the received signal is passed by a low-noise amplifier (LNA) and a mixer to convert the received signal frequency to the IF frequency. So during the RF/IF stage, challenge is to balance the noise, spurious signals and inter-modulation products.

These non-linearities can affect the overall performance of the SDR based communication system and may not conform to the requirements of 3GPP standards for LTE.

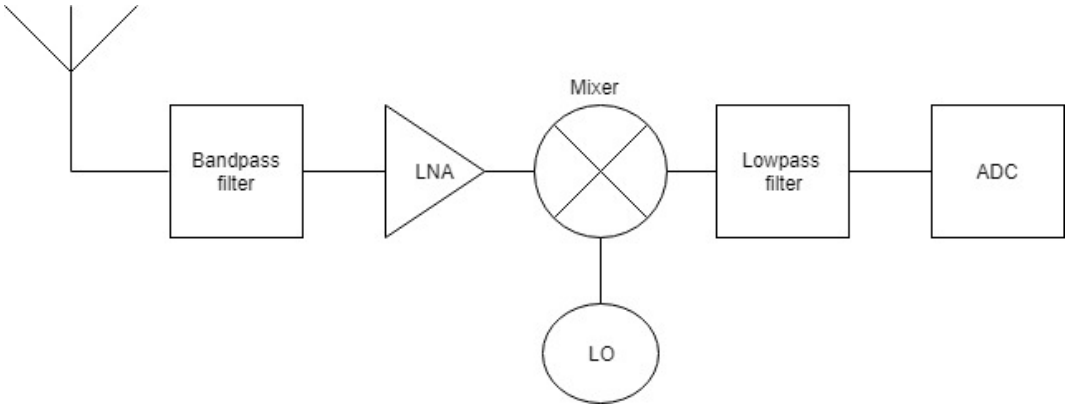


Figure 2.2: block diagram of basic radio [19]

2.1.2.3 Interference Cancellation

Another important consideration in SDR or any radio system is interference suppression. This can be achieved at the RF stage by antenna separation, frequency separation, by using programmable analog notch filters, and active cancellation. Active cancellation is the process of introducing a replica of the transmitted signal, so that it may be subtracted from the received signal [19].

2.1.2.4 Security Issue

The wireless communication is prone to many security threats. In SDR based system, security risk is high as different wireless standards and signal parameters (like frequency, power, and modulation types) are adjusted or changed through installing/downloading new software instead of removing or replicating hardware components. The successful deployment of SDR based communication system depends on the design and implementation of essential security mechanisms to ensure the

robustness of network and terminals against security threats. Some major security threats are mentioned and discussed in [30].

2.1.2.5 Sampling

As the idea of the SDR is based on the concept of processing signals in a digital domain. So the sampling capability of ADC and Digital to Analog converter (DAC) is fundamental to digitize the received analog signal. Poor sampling technique can cause performance and data loss. In the ideal case analog signal received at the antenna should be digitized using a high speed digitizers after the antenna. However, the issue is high power requirements and huge amount of data which is produced after digitization which again requires huge processing power [30].

So in practical SDR to avoid this issue received signal is passed through the band pass filter and is converted into an IF frequency at the RF front end stage. Now the remaining or desired signal is much lower in bandwidth as compared to what is initially received. This conversion greatly reduced the requirements of a high speed ADC and makes it possible to implement the concept of SDR in commercial products [30].

2.1.2.6 Timing Synchronization

Timing synchronization between the analog and digital domain is an important concern to address. The clock rates of the analog RF front-end hardware must match with the digital side in order to avoid sampling rate mismatch. In ADC case, techniques called channelization and sample rate conversion are often required to match the sampling rate to the clock rate of processing hardware and to interface the digital hardware that creates the modulated signals to the DAC [30].

2.2 Comparison of Available SDR

As with time SDRs are becoming more common, many companies and organizations have developed hardware front-ends and software packages to help in software radios development. The most prominent hardware front ends to date have been the USRP hardware boards, developed by Ettus Research [25]. Apart from USRP many other hardware boards are getting attention in terms of affordability and features they offer. A comparison list of available SDR platforms supporting the LTE frequency range is provided in appendix B.

Additionally, many software packages are introduced for SDR development, including GNU Radio, OSSIE (Open Source SCA Implementation- Embedded), limesuite, simulink and labView SDR packages. Using these development softwares, researchers have prototyped many of the most used radio standards. We can say that the major advantage of SDR is in the development. It allows us to change the modulation schema without necessarily modifying the hardware design which saves development cost [25].

In the current project we will use one of the most high end SDR USRP x310 and a relatively cost effective option i.e. limeSDR. The platform is chosen based on their ability to support LTE frequencies as well as other frequencies which can

be utilized later to implement additional communication technologies like AM, FM, wifi, DAB or DVB.

2.3 LTE

LTE commonly referred to as 4G, is a wireless broadband communication standard for mobile phones and data terminals providing higher data rates. It was designed by the Third Generation Partnership Project (3GPP). 3GPP is an alliance of telecommunication standards associations, known as the organizational partners. 3GPP alliance covers cellular telecommunication network technologies, like radio access network, services, systems aspects, core network and terminals by providing complete system specifications in these areas. [9]

LTE is evolved from previous generation of mobile network known as the Universal Mobile Telecommunication System (UMTS)/High Speed Packet Access (HSPA), which in turn evolved from the Global System for Mobile Communications (GSM) and Enhanced Data rates for GSM Evolution (EDGE). The specifications for LTE are specified by release 8 of 3GPP project. It was completed in December 2008 and this has been the basis for the starting point of LTE. LTE specifications are very stable, with the added benefit of enhancements having been introduced in all successive 3GPP releases [9, 25].

Mobile or cellular data usage demands of users have increased exponentially in the previous years. The demand has increased in terms of both bandwidth and quality of services. As a result of increasing demands second generation (2G) and third generation (3G) networks data traffic was increased drastically and started to become congested. To cater this increasing user data demands technologies like HSPA, evolution of UMTS were introduced, but could not satisfy the users needs. Keeping in view these increasing demands, high quality of service requirements and migration of broadband services to mobile devices were the prime motivation for the evolution of LTE [9, 25].

The main aim of LTE is to provide a high data rate, low latency and packet optimized radio access technology supporting flexible bandwidth deployments. Secondly, packet-switched network architecture has been designed with the goal to support seamless mobility and great quality of service. [9, 25]

2.3.1 Mobile Communication Generations

The first generation of mobile communication was introduced in the 1980s. It was based on the analog radio system technology and the users could only make phone calls. No SMS or data services were available in the first generation. However, with the introduction of 2G in the 1990s, mobile communication was digitized. So in the 2G SMS and data services were offered along with the voice. The major 2G technologies were: GSM/GPRS, EDGE, CDMAOne, PDC, iDEN, IS-136 and D-AMPS.

Furthermore, to cater the increasing user demands third 3G was introduced. 3G offered much faster data transfer and ability to transfer high amounts of data. So 3G supported services like video call, file sharing, internet surfing, watching TV online and playing online games.

Next the 4G or the LTE was introduced in 2008 when 3GPP finalized its release 8. The LTE is purely IP based. Both real time services and data communication services are carried by the IP protocol. An IP address is assigned to the user equipment as soon as it turns on and released when it is switched off. LTE is based on Orthogonal Frequency Division Multiple Access (OFDMA), supports higher order modulation (up to 256QAM), large bandwidths (up to 20 MHz) and spatial multiplexing in the downlink (up to 4x4) high data rates can be achieved. The highest theoretical peak data rate on the transport channel is 75 Mbps in the uplink, and in the downlink the rate can be as high as 300 Mbps using spatial multiplexing [9].

2.3.2 Architecture of LTE

The major components of LTE are User Equipment (UE), Evolved Universal Terrestrial Access Network (E-UTRAN), and Evolved Packet Core (EPC). In technical or 3GPP terms the whole system is termed as Evolved Packet System (EPS). The overall network architecture is shown in figure 2.3, where the core network EPC consists of many logical nodes and the access network E-UTRAN consists of Evolved NodeB (eNB) or the base station which connects to the UEs [9].

The LTE access network architecture is a flat architecture based on network of base stations (eNB). The eNBs are usually inter-connected via X2-interface. Whereas, S1-interface connection is used between the eNB and core network. In LTE there is no centralized control or controller. The intelligence or control is distributed among the base-stations to speed up the connection set-up and reduce the time required for handovers. The time for a handover is essential for real-time services where end-users tend to end calls if the handover takes too long [9].

Figure 2.3: LTE Architecture [9]

2.3.2.1 E-UTRAN

Access part of LTE is termed as E-UTRAN. It handles the radio communication between UE and the EPC both Uplink (UL) and Downlink (DL) communication. Apart from this, eNB also transmit signaling messages like handover commands. E-UTRAN consists of eNB base stations to control the mobile units/UE in different

cells. The figure above shows an overall E-UTRAN architecture. The eNBs are interconnected with each other and there is no centralized controller in E-UTRAN. So E-UTRAN architecture is said to be flat. So in short, the EUTRAN is responsible for all radio related functions such as radio resource management, header compression, security, positioning and connectivity to the EPC [9].

2.3.2.2 Evolved Packet Core (EPC)

EPC is the latest development in the core network architecture of the 3GPPÂ´s LTE wireless communication standards. In EPC based network Internet Protocol (IP) is uesd for all services i.e. voice, data, SMS etc. It is the evolution of the packet-switched architecture used in GPRS and UMTS. In EPC system user plane and the data plane have been separated, thus allow more flexibility and scale-ability.

Figure 2.4 shows how the user and data plane have been separated in a basic LTE system. The UE is connected to the EPC through E-UTRAN (LTE access network). As we can see in the figure that the EPC consists of four basic elements the serving gateway (Serving GW), the public data network gateway (PDN GW), the Mobile Management Entity (MME) and the Home Subscriber Server (HSS). The gateways (Serving GW and PDN GW) defines the user plane and are responsible for transporting the IP data traffic between the UE and the external networks. Whereas, MME and HSS deals with the control plane. MME handles the control signalling related to mobility and security of LTE network. It is responsible for tracking and paging of UE in idle-mode. The HSS is a database of subscribers apart from this it also provides mobility management functions like call and session setup, user authentication and access authorization.

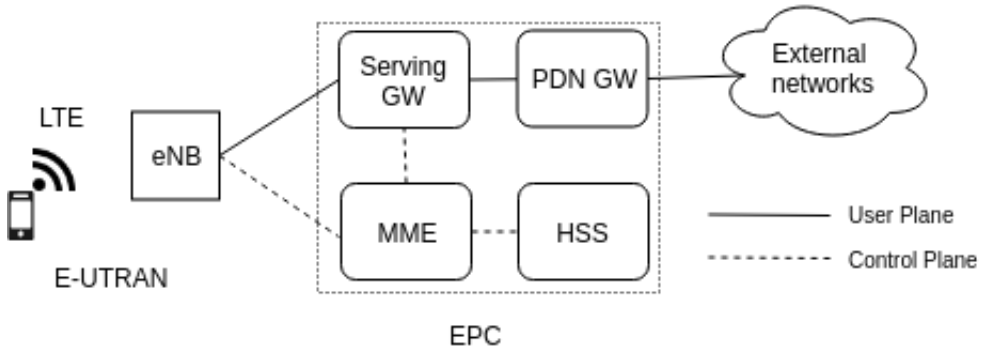


Figure 2.4: EPC Architecture (User and Control Plane) [8]

Overall EPC has a flat IP architecture which improves the network performance and few network nodes are required in the handling of the traffic because of flattened architecture [8].

2.3.2.3 User Equipment

Device used by end-user to communicate with mobile networks is termed as UE. The LTE UEs are divided into different categories based on their UL and DL capabilities. These categories allows the eNB to communicate effectively with all the connected UEs. UE radio access capabilities are listed in TS 36.306 [6].

In this project our focus will be UE Cat 4 with a maximum data rate of 50Mbps and 150Mbps in the UL and the DL respectively using MIMO configurations. And upto 75Mbps in DL using single input, single output (SISO) configurations.

2.3.3 LTE UE Protocol Stack

LTE protocol is divided into different layers namely Non Access Stratum (NAS), Radio Resource Control (RRC), Packet Data Convergence Control (PDCP), Radio Link Control (RLC), Medium Access Layer (MAC) and the Physical Layer (PHY) as shown in figure 2.5.

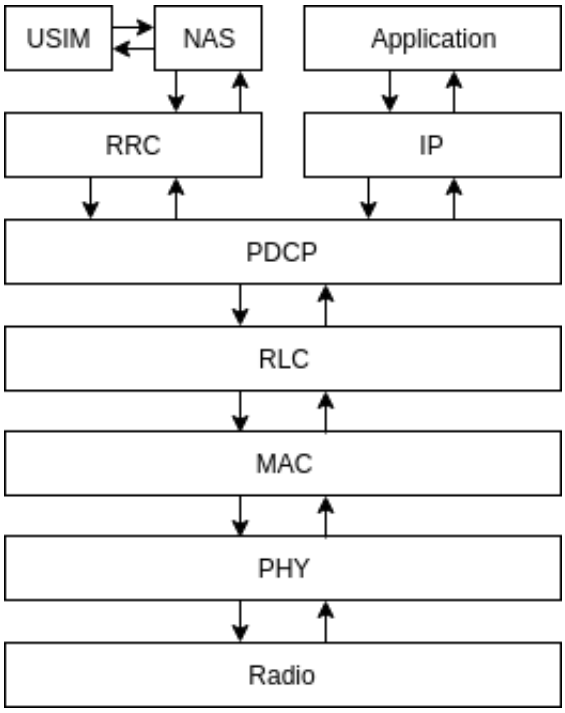


Figure 2.5: UE Protocol stack [28]

The highest layer of the control plane is the NAS layer. NAS layer manages the mobility, management and establishment of IP connectivity. It communicates between the UE and MME. During the initial connection setup,it transmits the USIM information to the MME. Next layer in the stack is the RRC defined in 3GPP TS 36.331 document [4]. The RRC layer is responsible for radio connection establishment and release functions, radio bearer establishment, mobility management, paging and broadcasting of the system information.

After RRC is the PDCP layer, as we can see in the figure 2.5, it handles packets from both the control plane (RRC) and data plane (IP). Header compression, ciphering and integrity protection tasks are performed at this layer. Further on, there is RLC layer, specifications are defined in 3GPP TS 36.322. The main responsibilities of this layer are segmentation, reassembly and re-transmission of lost packets. Next in the stack is the medium access (MAC) layer, basic purpose of this layer is to handle the channel access procedures or the random access procedure. It performs the multiplexing and demultiplexing of logical and transport channels. Tasks like scheduling requests, buffer status reporting, and hybrid automatic repeat request (HARQ) are also handled by the MAC layer.

The lowest layer in the stack is the PHY layer, it passes all the data from the MAC layer to the air interface. The main tasks performed are link adaptation, power control and cell search.

2.3.4 LTE Channels

LTE data channels are divided into three categories logical channels, transport channels and physical channels. The configurations are defined in 3GPP TS 36.331 section 9.1.1 [4]. Logical channels provide information about type of data being transmitted e.g. traffic, control, or broadcast. These channels exists between the RLC and MAC layer of the protocol. The list of logical channels in LTE protocol is given in table below:

Table 2.1: LTE Logical Channels [2]

Logical channel	Control channel	Traffic channel
Broadcast Control Channel (BCCH)	x	
Paging Control Channel (PCCH)	x	
Common Control Channel (CCCH)	x	
Dedicated Control Channel (DCCH)	x	
Multicast Control Channel (MCCH)	x	
Dedicated Traffic Channel (DTCH)		x
Multicast Traffic Channel (MTCH)		x

The Next in line are the transport channels which transfer the data between MAC layer and the PHY layer. The list of UL and DL transport channels is given in table 2.2.

Table 2.2: LTE Transport Channels (Uplink/Downlink) [2]

Transport channel	Downlink	Uplink
Broadcast Channel (BCH)	x	
Downlink Shared Channel (DL-SCH)	x	
Paging Channel (PCH)	x	
Multicast Channel (MCH)	x	
Uplink Shared Channel (UL-SCH)		x
Random Access Channel (RACH)		x

Lastly, the physical channels are the channels which carries the data over the air between the UE and the network. They are the lowest level of channels in LTE. List of UL and DL physical channels is listed in table below.

Table 2.3: LTE Physical Channels (Uplink/Downlink) [2]

Physical channel	Downlink	Uplink
Physical broadcast channel (PBCH)	x	
Physical downlink control channel (PDCCH)	x	
Physical downlink shared channel (PDSCH)	x	
Physical control format indicator channel (PCFICH)	x	
Physical HARQ indicator channel (PHICH))	x	
Physical uplink control channel (PUCCH)		x
Physical uplink shared channel (PUSCH)		x
Physical random access channel (PRACH)		x

2.3.5 LTE Handshake

In an LTE based system when a UE tries to connect to the available network, the process is generally termed as handshake. The whole handshake process can be divided into two major phases.

1. UE and eNB RRC connection setup
2. Authentication and security setup

These two major steps can be divided further into minor steps as explained below.

2.3.5.1 UE and eNB RRC connection setup

In this first step the UE detects the available eNB base-stations and try to establish a radio link connection with the base-station. Following major steps are performed in this phase i.e. synchronization, system information acquisition, random access procedure and RRC connection setup.

In the first step the UE gets the frequency and time synchronization by the decoding the primary and secondary synchronization signals (PSS and SSS), transmitted by the eNB after every 5ms. Sub frame level synchronization and physical layer cell identity is attained through this step [28].

In the next step the UE gets the network specification information which is stored in the master information block (MIB) and system information blocks (SIBs) messages. MIB is transmitted on PBCH which contains the system bandwidth and PHICH format information. Next the UE decodes the downlink control information (DCI) message transmitted on PDCCH. DCI contains the positions of SIB messages. The two SIB messages, SIB1 and SIB2 contains the cell access information, control and shared channel configuration and random access (RA) information necessary for the RA procedure [28].

Once above steps are performed, the UE know has the information of the available network. At the this time UE tries to get the access to the network by initiating the RA procedure and sends the RA preamble over the PRACH channel to the eNB. After sending RA preamble UE gets the response from the eNodeB called the random access response (RAR, or MSG2). In RAR UE is assigned with a temporary

identity e.g. temporary C-RNTI along with the information of UL timing adjustment and the slot information for the next message to be transmitted on PUSCH. Multiple UE can send the RA preamble at the same time in this case contention resolution will be performed as described in [1] section 10.1.5.

Next is the final step of the phase 1 where the UE sends the RRC connection request or the MSG3 which includes the temporary C-RNTI number and the subscriber details. In reply the eNB sends the RRC connection setup or MSG4 on PDSCH and assigns the available UL resources to the UE. On successful reception of MSG4 UE send the RRC connection setup complete message which contains the NAS service request message explained in the following section [28] [1]. Steps for RRC connection setup are shown in figure 2.6.

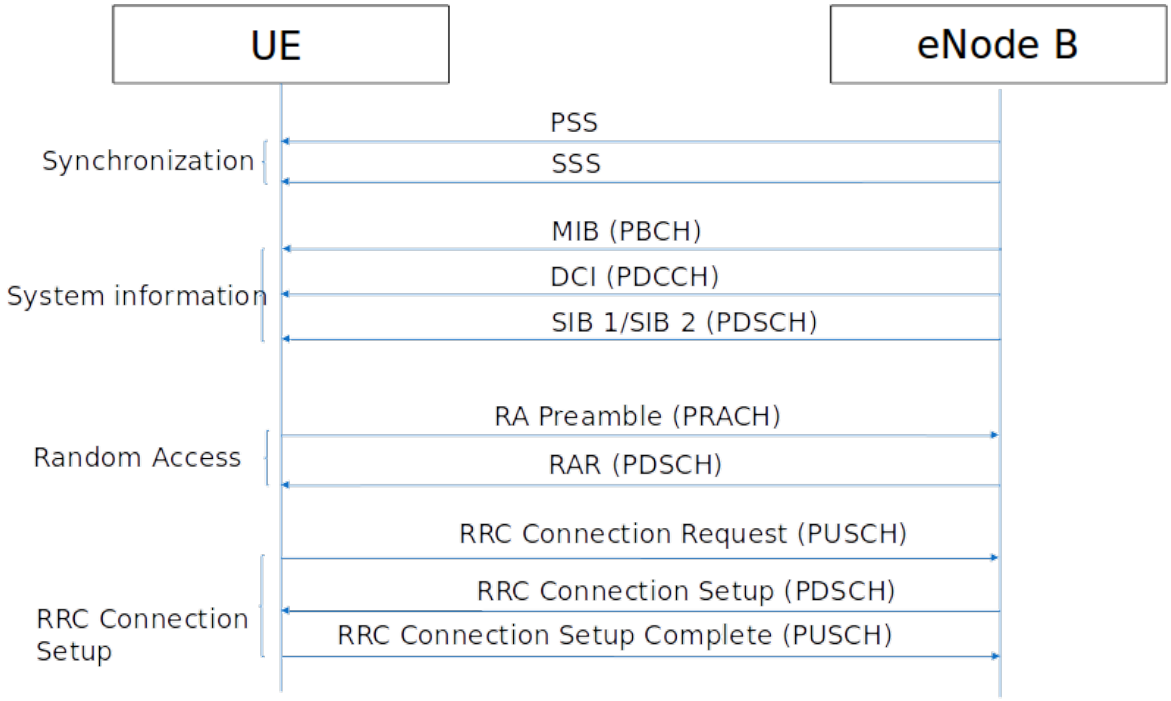


Figure 2.6: Steps for RRC connection setup [28]

2.3.5.2 Authentication and security setup

The second phase of the two major steps are performed; NAS authentication and security setup and RRC connection re-configurations. Once the RRC connection is setup between the eNB and the UE, the authentication step is performed. UE and eNB authenticate each other using the defined algorithm in 3GPP TS 35.206 V4.0.0 (2001-04). It is important to mention that in the authentication phase communication is done between the UE and the MME of the EPC via eNB as shown in figure 2.7.

After authentication, a security mode command is transmitted by the MME via eNB on PDSCH in order to activate integrity protection and ciphering. The UE after verifying the security mode command responds with security mode complete message by applying integrity protection and ciphering to this message [28]. Once

the security algorithms are implemented the eNodeB transmit a RRC connection reconfiguration message, to establish SRBs and the data radio bearer (DRB). The UE responds with a RRC connection reconfiguration complete message, allowing for user data transfer using a DRB [28]. Details about the security aspects are defined in the series 33 of 3GPP specifications.

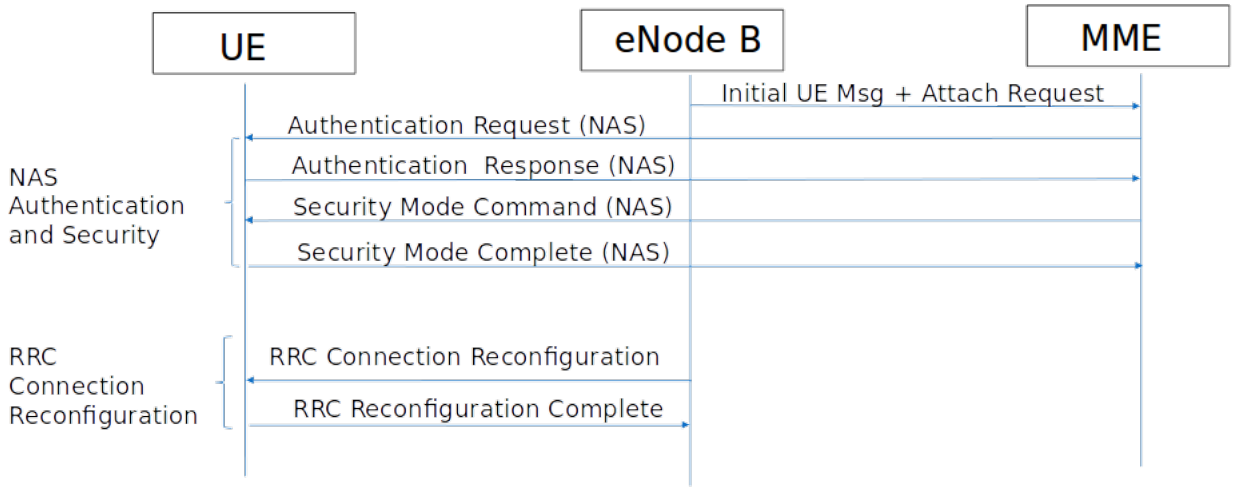


Figure 2.7: NAS Authentication and Security [28]

2.4 SDR based LTE Solutions

Prototyping of LTE wireless communication standard over the SDR technology is the area of interest especially for researchers to exploit new features. However, the applicability of these solutions for some practical applications is still to be explored.

There are quite a few open source implementations of LTE technology over SDR which we considered for our project. Below is the brief description of these available solutions.

2.4.1 OpenLTE

OpenLTE is an open source software based implementation of 3GPP LTE eNodeB. It includes a simple EPC part. It is GNU based application which allows scanning and recording LTE signals. The application provides functionalities of DL transmit, receive functionality and UL PRACH transmit and receive functionality. DL receive and LTE I/Q file recording can be done using rtl-sdr, HackRF, or USRP B2X0. A simple eNodeB (LTE-FDD-enodeb) can be implemented using USRP B210. [33, 25].

2.4.2 srsLTE

srsLTE is the open source LTE complaint linux based application developed by software radio systems (SRS), a private limited company in Ireland [18]. srsLTE application includes the functionalities of an LTE eNodeB, EPC and UE. The software application was initially designed according to release 8 of the LTE standard [15]. However, current open source version of the application is LTE release 15 complaint supporting both Frequency Division Duplex (FDD) and Time Division Duplex (TDD) modes [18]. The application has been tested with different SDR platforms like USRP, bladeRF and limeSDR. The implementation is highly modular and can be modified to specific needs. Some functions from the openLTE project has been used in this project [18]. For commercial purposes the srs sell the commercial licenses and also sell the proprietary products based on srsLTE code base[18]

2.4.3 Open Air Interface

OAI project is the open source implementation of LTE system covering the complete protocol stack of EPC and E-UTRAN as per the 3GPP standard [24]. The implementation is LTE release 10 complaint and can be used to fully built or customize a LTE base station, UE or a EPC. Its a Linux based application and it uses the software radio front end connected to a host computer to achieve the transceiver functionalities. As per the licensing terms and conditions, the code is freely available to be used for personal, research or academic use only [24]. The OAI software application is of particular interest for researchers for 5G research as described in [23].

2.4.4 lte-sidelink

lte-sidelink is another open source software implementation of 3GPP LTE sidelink features which enables device-to-device (D2D) communication. The LTE sidelink feature was first introduced in LTE 3GPP release 12 and further functions like public safety messages or vehicle-to-everything (V2X) were described in release 13 and 14. In a typical LTE system UL and DL signals always pass through the LTE eNB. Incase of sidelink two UEs, if in close proximity can communicate with each other directly without involving eNB. The service is called "Proximity Services (ProSe)" and the UEs having this feature are generally refered as "ProSe"-enabled UEs[13].

2.4.5 Amarisoft LTE 100

Amarisoft LTE 100 is the most advanced and well adopted software based implementation of LTE eNodeB basestation. It is developed by Amarisoft a France based software company related to telecom industry. The Amarisoft eNodeB application is LTE release 14 complaint and can work with various SDR platforms and high power remote radio head (RRH) [11].

2.5 Related Work

As mentioned above, there are quite a few implemented open source projects of LTE prototype over SDR. The one we will use in this project will be srsLTE. The choice is made keeping in view a few things. First of all we require a working LTE UE which is only available in the srsLTE or OAI project. However, the licensing condition of OAI doesnot allow it to be used for a commercial use whereas, srsLTE is distributed under commercial license if required. Another advantage of using srsLTE over OAI is that the coding is highly modular which makes it easy to understand as well as modify. A comparison study of OAI and srsLTE is well described in [14].

srsLTE project is widely used in different LTE and 5G related research projects. These studies include the performance assessment of SDR based eNB solutions for 5G prototyping [16]. Next in [27], the author has prototype the NB-IoT technology over SDR using the srsLTE library. Which proves the modularity of srsLTE project. Further in [21], researchers have used the SDR based LTE technology to study the coexistence of WiFi and LTE technologies in the unlicensed band. Lastly, in [12], the authors have implemented an independent LTE network for vehicular communication using the srsLTE and SDR technology. The application is named "tinyLTE". The application can be used to implement standalone LTE cells using off the shelf SDR hardware. Apart from these applications srsLTE library is also extensively used to locate the security vulnerabilities of an LTE network [34].

We can see that the srsLTE project is extensively used by different researchers to solve different LTE related problems. Similarly we will exploit the use of srsLTE library in our project to get the comprehensive understanding of the system in terms of performance, limitations, and interoperability with commercial network.

Chapter 3

Methodology

This chapter describes the research methodology and implementation steps. Section 3.1 will explain the research process/techniques used during the project. Section 3.2 describes the data collection techniques. Lab setup or experimental design setup is explained in section 3.3. Section 3.4 describes the evaluation process to access the reliability and validity of data collected. Data analysis tool and techniques are discussed in section 3.5 and last section 3.6 describes the framework selected to evaluate our work.

3.1 Research Process

This project has both a software and a hardware part. As the LTE application or the srsLTE is running on a linux based machine and for RF requirements an SDR hardware platform is used. First of all we begin with understanding the features and limitations of SDR technology as the SDR is the motivational technology behind this project. Next we get ourselves familiarize with the LTE technology and all the components in the LTE protocol stack.

After getting the understanding of SDR and LTE technology next step was implementation. The srsLTE application was installed in a linux based pc and was used with the selected SDR hardware platform, details are mentioned in section 3.3. Following this, performance analysis of the implemented solution was done by plotting the DL data rate plots, DL SNR plots and comparison between high end SDR device USRP X310 and the limeSDR. Apart from above tests a RF performance analysis has been done using a spectrum analyzer to look for LO leakage and interference signals generated by the SDRs. Finally, the conclusion is drawn based on the results obtained from above controlled experiments in lab, followed by the future recommendations.

3.2 Experimental Design

This section explains the test bed model or the lab setup for implementation and performance analysis of the system. It will give an insight to the hardware and software used in the project. A brief description on test environment implementation of SDR based LTE system is explained below.

3.2.1 Test Environments and Methods

In the lab the setup is done for two types of testing. First the performance measurements, which are done using the quantitative elements like SNR or data-rate so quantitative experimental design method was used in this case. The quantitative measurements are useful as they can be compared with the protocol standard and define the limitations clearly.

Second setup was done to analyze the RF performance of the system i.e. to observe the signal of interest and noise generated. The lab setups done to perform different tests are explained in sections below.

3.2.1.1 Quantitative Performance Measurement Setup

The test bed of our setup is shown in figure 3.1. The srsLTE software is running on a linux based PC. In one PC srsEPC and the srseNB are running and one is used for srsUE. Both the computers are connected to the SDR using a Gigabit Ethernet (GbE) cable in case of the USRP and USB3.0 cable is used for limeSDR. For testing one SDR is used as an eNB or base-station and one as UE. The transmit (TX) and receive (RX) ports of the SDRs were connected using one meter SMA cables to minimize the RF losses. 30dB attenuators were used between each connection to avoid any damage to the RF front-end due to high power. To test the DL data rate and SNR, we send a stream of User Datagram Protocol (UDP) packets using iperf3 making the eNB PC as client sending continuous packets to the UE which acts as the iperf3 server. The UDP packets are sent with a time interval of 0.1 sec, which is the minimum achievable interval in-case of iperf3. The SNR values and other parameters are stored in a CSV file by the srsLTE application.

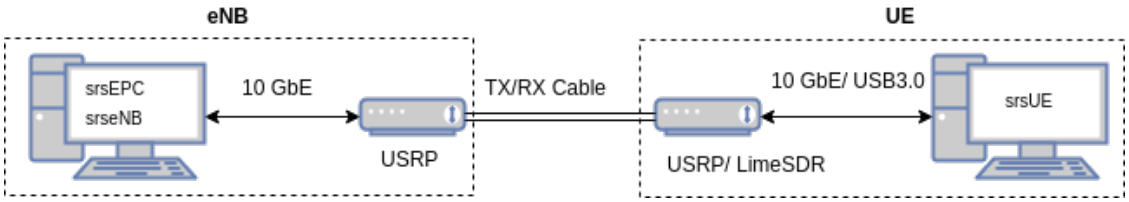


Figure 3.1: Test bed setup - Quantitative Measurements (datarate and SNR)

3.2.1.2 eNB RF Performance Analysis Setup

After setting up the system for testing, it is important to know the RF characteristics of the eNB and UE. To measure the maximum and minimum output power of the eNB the setup was done as shown in figure 3.2. The TX of the eNB USRP was attached to the spectrum analyzer and TX_gain was increased in fixed step sizes and the output was recorded.

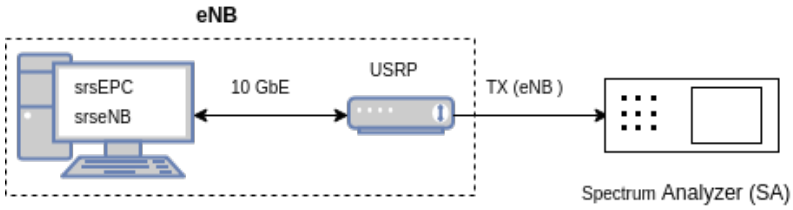


Figure 3.2: Test bed setup - eNB RF Performance Analysis

3.2.1.3 UE RF Performance Analysis Setup

As mentioned above, RF performance analysis of the srsUE was done to make out for the saturation point and minimum/maximum output power. So to analyze the performance the lab setup was done as shown in figure 3.3.

In the setup the TX port of the eNB is connected to the RX port of the UE side SDR platform and to observe the output RF signal of srsUE, the TX of the UE side SDR was attached to the spectrum analyzer. This is done because UE starts transmitting as soon as it detects the available network

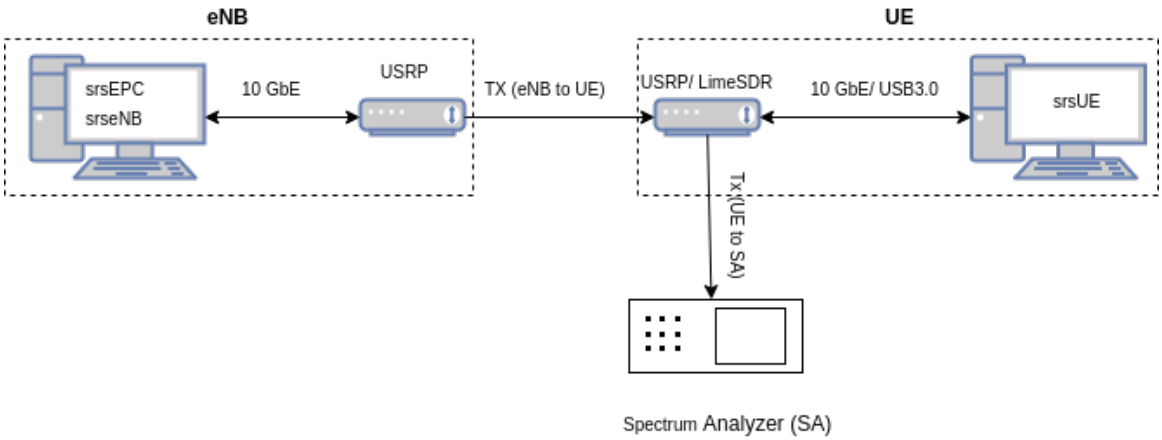


Figure 3.3: Test bed setup - UE RF Performance Analysis

3.2.1.4 Basestaion Simulator Test

As the goal of the project was to design a SDR based LTE UE modem which is compatible with commercial mobile networks. So to test the compatibility a commercial base station simulator by rohde & schwarz CMW500 was used. The test setup is shown in figure 3.4. To test the compatibility. The UEs TX and RX was connected with the CMW500 using coaxial cables.

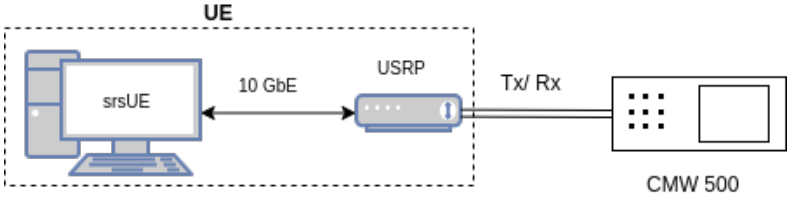


Figure 3.4: Test bed setup - srsUE Compatibility Testing with CMW500

3.2.2 Hardware

The host PCs used for the project are dell computers with intel i7-7th generation quadcore processor. Which is a requirement as mentioned in [18], for maximum data rate achievement. Further more the RF front-end used is USRP X310 for eNB. For UE side limeSDR and USRP X310 are used to draw a comparison between a high end device and a relatively cost effective option. Additionally ethernet cables are used to communicate between USRP and host PC. Incase of limeSDR USB3.0 cable is uesd. Anritsu MS 8604A spectrum analyzer was used to perform some RF characteristic analysis of the SDRs. Finally, to check the compatibility of srsUE with the commercial mobile networks a base station simulator CMW500 by rohde & schwarz was used. A PC/SC card reader OMNIKEY 6121 was used to test the pcsc mode of the srsLTE application, which allows the use of hard sim card instead of soft sim parameters.

3.2.3 Software

Number of software packages were used to implement and to draw the performance analysis of SDR based LTE modem. The software used are USRP Hardware Driver (UHD), limesuite, soapySDR, PCSClite, srsLTE, iperf3 and wireshark.

First of all UHD driver version 3.0.9 LTS is used. It is a free and open-source software driver and Application Program Interface (API) for the USRP developed by ettus research for their SDR platforms [29]. Next software used is limesuite for limeSDR configurations and to provide support for soapySDR which is an open source generalized API library for interfacing with different SDR devices [26] [22]. Further, the PCSClite library is used to allow hard SIM support in srsLTE. It is an API for accessing smart card readers [20].

All the above mentioned packages were installed to run and exploit the use of our main application i.e. srsLTE, which is a complete LTE solution consisting of an EPC, eNB and UE parts [18]. The iperf3 tool is used for active measurements of the maximum achievable DL data rate [17]. Lastly, wireshark was used to observe the packets sent and received on a virtual tunnel created between the eNB and UE.

3.3 Data Collection

To analyze the performance of the implemented LTE application, we measured the DL data rate, SNR and RF power output. Further more spurious signal generated by the SDR was observed over the span of 40MHz. The DL data rate of the srsUE was recorded for one hundred thousand iterations and average was recorded. The test was done using the USRP x310, for all bandwidth settings i.e. 1.4 MHz, 3 MHz, 5 MHz, 10 MHz, 15 MHz and 20 MHz and for all Modulation Coding Scheme (MCS) values from 0 to 28 for each bandwidth. Secondly, the signal SNR values were recorded every second in a CSV file and the average was calculated. It is pertinent to mention that maximum data rate can be achieved if their are minimum losses in the transmission medium, that is why all tests are conducted using a Tx/Rx SMA cable between the eNB and UE to minimize the external losses.

For RF power output test spectrum analyzer is used to visualize the maximum and minimum output power along with observing the spurious emissions in adjacent

bands.

3.4 Reliability and validity of the data collected

As mentioned above, in case of data rate and SNR evaluation each reading obtained for a particular setting is the average of one hundred thousand iterations. Based on this, the analysis was drawn about the maximum achievable data rates.

3.5 Data Analysis

The data analysis was performed by generating different plots. First we plotted DL data rate against each MCS value for each bandwidth setting and a comparison is shown between the theoretically achievable values as per 3GPP standard. For having a more detail performance idea the error plot is also plotted to show which Physical Resource Block (PRB) and MCS settings performed closest to the standard requirements.

Next the SNR values are plotted against each MCS value to get the overall picture. Further the maximum achievable out power was recorded and plotted against the Tx gain values to show the saturation point. Lastly the noise or interference generated by the UE was observed on the spectrum analyzer.

Chapter 4

Implementation

This chapter provides a brief introduction of srsLTE, its components, features and implementation. Installation and configuration steps are discussed along with the focus describing it as software based flexible solution to implement LTE services.

4.1 srsLTE Components

As mentioned in background chapter, srsLTE is an open source LTE software suite. It is developed and maintained by software radio systems, an Ireland based company [18]. The motivation behind the project was to provide a flexible platform for product development for LTE systems [15]. However, in recent days the implementation has emerged as an alternative option to provide LTE commercial products using the SDR technology [31].

The srsLTE offers an open source implementation of all LTE components i.e UE, eNB and the EPC covering the complete protocol stack [18, 15]. The main components of the project are: srsUE, srseNB and srsEPC.

The transceiver parts (UE and eNB) are realized using the SDR technology. Whereas, the EPC components are fully implemented on a PC. The specific features of each component are mentioned in subsequent sections. However, there are some common features which are listed below [18]:

- Open source implementation is LTE Release 10 complaint
- LTE Bandwidths covered: 1.4MHz, 3MHz, 5MHz, 10MHz, 15MHz and 20MHz
- Transmission modes supported:
  - single antenna mode
  - transmit diversity mode
  - Cyclic Delay Diversity (CCD) mode
  - closed-loop spatial multiplexing
- Frequency-based zero forcing (ZF) equalizer
- MMSE equalizer
- Evolved multimedia broadcast and multicast service (eMBMS)

- MAC, RLC, PDCP, RRC, NAS, S1AP and GW layers
- Per layer Log system
- Trace metrics to show BLER, datarate, FEC etc
- Input configuration files (UE.conf, eNB.config and EPC.config)

The coding of the srsLTE application is done in a modular way in order to allow researchers to modify the library as per requirements [15]. The hierarchical implementation of modules is shown in figure 4.1.

As we can see overall the library is divided into four modules. The lowest level or the core module contains the major building blocks like turbo and convolution coders/decoders, modulator and demodulator, synchronization, channel estimation and reference signal generation, OFDM and SC-FDMA processing etc [15]. Next module is the physical channels containing the data and control channel modules like PDSCH, PUSCH, PDCCH, PUCCH, etc. These channels are responsible for carrying the information between the UE and the eNB. Next, In the UE Processes module, the module implements the UE processes like sync, UL, DL, cell search etc. Last in the highest layer comes some example applications which are helpful to understand the srsLTE application functionality [15].

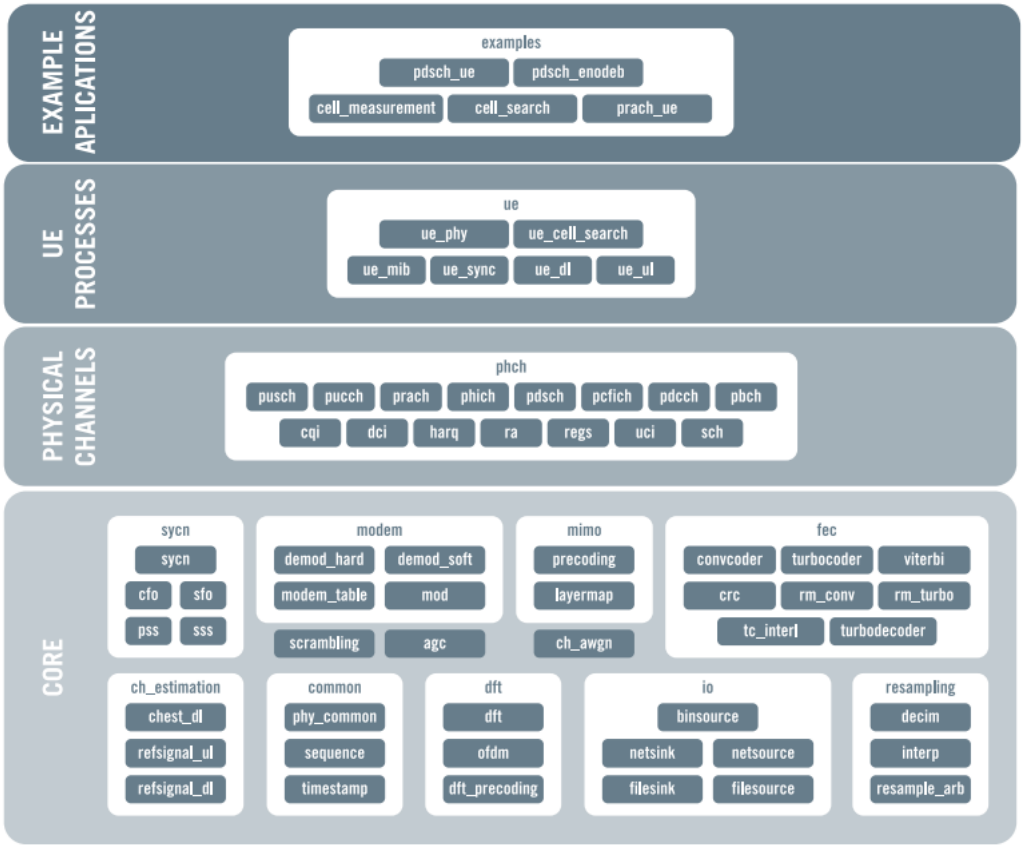


Figure 4.1: SRSLTE Modules [15]

4.1.1 srsUE

The srsUE is the UE part of the srsLTE suite. Its silent features include FDD and TDD modes, carrier aggregation, cell search and synchronization procedure. Furthermore, soft USIM is enabled which uses milenage or XOR authentication algorithm. Hard USIM support is provided using the PCSC framework. As soon as the UE connects to a network a virtual network interface 'tun_srsue' is created.

The maximum achievable data rate depends on the CPU power as all the signal processing is done using a CPU. As mentioned in [18] the maximum achieveable data rate is 150 Mbps in DL using 20 MHz MIMO (TM3/TM4) configuration and i7 Quad-Core CPU is required.

4.1.2 srseNB

The srseNB is the LTE basestation, the current open source code include the FDD configuration,Round Robin MAC scheduler, encryption, Channel Quality Indicator (CQI) feedback support and standard S1AP and GTP-U interfaces to the core network. The srseNB can support maximum of 150 Mbps DL in 20 MHz MIMO TM3/TM4 with commercial UEs [18].

4.1.3 srsEPC

The srsEPC is the very basic implementation of LTE core network. It includes MME, gateways and HSS containing the user database. The HSS directory is configurable and a user can be added or deleted from the data base. Further more paging support has been added [18].

4.2 Building, Installing, and Running srsLTE

This section discusses the installation steps of srsLTE on a Linux-based PC and supporting softwares required for interfacing with the SDR platform used.

The srsLTE application software has three main components srsepc, srsUE and srseNB. The srsUE and srseNB are the radio parts which require an interface to a SDR transceiver. Whereas, EPC part is software based having modules like MME, SGW, PGW and HSS which are the main components of the LTE core EPC architecture.

Before moving on with the srsLTE application installation, it is necessary to install all the device driver softwares as per the SDR platforms used. Below section give a brief overview of supporting softwares required for the project.

4.2.1 Supporting Softwares

In this project we have used USRP X310 and a limeSDR as the RF front-end device. The required supporting softwares were UHD for USRP interface, limeSuite and SoapySDR for limeSDR interface. Another application 'PCSClite' is also used to exploit the hard sim functionalities of srsLTE.

First we started with installing the UHD driver for the USRP. We used the UHD driver version 3.9 LTS as it is recommended by the srsLTE team. The installation steps are mentioned in Appendix A. To verify the successful installation "UHD_find_devices" command is used to verify if the driver detects the attached USRP as shown in figure 4.2.

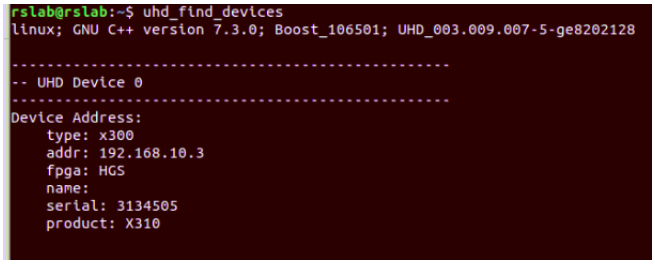


Figure 4.2: UHD Version check

If the USRP is not detected, check if the cable is properly connected and try updating the USRP FPGA image as per UHD version. Moving on with the installation of supporting softwares, limeSuite and soapySDR libraries were installed. Installation steps are mentioned in the appendix A.2. To verify the successful installation "limeUtil –info" and "SoapySDRUtil –info" commands were used to check the installed software version as shown in figure 4.3 and 4.4.

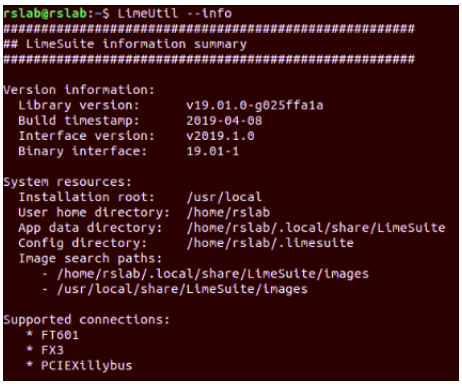


Figure 4.3: LimeSuite Version check

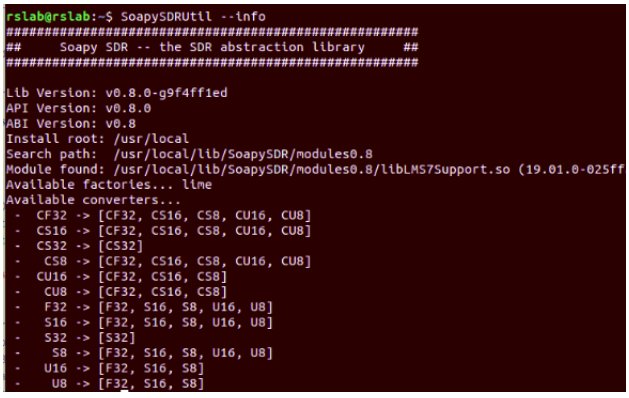


Figure 4.4: SoapySDR Version check

Further, it is recommended to check whether the attached SDR is detected by the installed software to confirm the successful installation. "limeUtil –find" and "SoapySDRUtil –find" commands were used to detect the attach limeSDR. If the SDR is not detected, its better to troubleshoot before moving ahead. Below figure 4.5 and 4.6 shows the expected output.

```
rslab@rslab:~$ SoapySDRUtil --find
######################################################
##     Soapy SDR -- the SDR abstraction library     ##
######################################################

Found device 0
  addr = 1d50:6108
  driver = lime
  label = LimeSDR-USB [USB 3.0] 90726074D2220
  media = USB 3.0
  module = FX3
  name = LimeSDR-USB
  serial = 00090726074D2220
```

Figure 4.5: limeUtil –find (Output)

```
rslab@rslab:~$ SoapySDRUtil --find
######################################################
##     Soapy SDR -- the SDR abstraction library     ##
######################################################

Found device 0
  addr = 1d50:6108
  driver = lime
  label = LimeSDR-USB [USB 3.0] 90726074D2220
  media = USB 3.0
  module = FX3
  name = LimeSDR-USB
  serial = 00090726074D2220
```

Figure 4.6: SoapySDRUtil –find (Output)

Another software library required to explore the hardsim support feature of the srsLTE application is PCSClite [20], it provides the interface to the sim card reader. The installation steps are mentioned in appendix. Once the software is successfully installed the installed version can be checked using the command "pcscd –version" and the expected output is shown below in figure 4.7.

```
rslab@rslab:~$ pcscd --version
pcsc-lite version 1.8.23.
Copyright (C) 1999-2002 by David Corcoran <corcoran@musclecard.com>.
Copyright (C) 2001-2015 by Ludovic Rousseau <ludovic.rousseau@free.fr>.
Copyright (C) 2003-2004 by Damien Sauveron <sauveron@labri.fr>.
Report bugs to <pcsclite-muscle@lists.alioth.debian.org>.
Enabled features: Linux x86_64-pc-linux-gnu libsystemd serial usb libudev usbdropdir=/usr/lib/pcsc/drivers ipcdi
r=/var/run/pcscd configdir=/etc/reader.conf.d
```

Figure 4.7: PCSC version check

To verify the successful installation, the "pcsc_scan" command is used to detect the attach card reader. The card reader name as mentioned "Reader 0" in figure 4.8 is required to be entered in ue configuration while using PCSC mode.

```
rslab@rslab:~$ pcsc_scan
PC/SC device scanner
V 1.5.2 (c) 2001-2017, Ludovic Rousseau <ludovic.rousseau@free.fr>
Using reader plug'n play mechanism
Scanning present readers...
0: HID Global OMNIKEY 6121 Smart Card Reader [OMNIKEY 6121 Smart Card Reader] 00 00

Mon Sep  2 18:24:20 2019
 Reader 0: HID Global OMNIKEY 6121 Smart Card Reader [OMNIKEY 6121 Smart Card Reader] 00 00
  Card state: Card inserted,
  ATR: 3B 9F 96 80 1F C7 80 31 E0 73 FE 21 13 67 98 07 02 04 03 02 01 41

ATR: 3B 9F 96 80 1F C7 80 31 E0 73 FE 21 13 67 98 07 02 04 03 02 01 41
+ TS = 3B --> Direct Convention
+ T0 = 9F, Y(1): 1001, K: 15 (historical bytes)
  TA(1) = 96 --> Fi=512, Di=32, 16 cycles/ETU
    250000 bits/s at 4 MHz, fMax for Fi = 5 MHz => 312500 bits/s
  TD(1) = 80 --> Y(i+1) = 1000, Protocol T = 0
-----
  TD(2) = 1F --> Y(i+1) = 0001, Protocol T = 15 - Global interface bytes following
-----
  TA(3) = C7 --> Clock stop: no preference - Class accepted by the card: (3G) A 5V B 3V C 1.8V
+ Historical bytes: 80 31 E0 73 FE 21 13 67 98 07 02 04 03 02 01
  Category indicator byte: 80 (compact TLV data object)
    Tag: 3, len: 1 (card service data byte)
      Card service data byte: E0
        - Application selection: by full DF name
        - Application selection: by partial DF name
        - BER-TLV data objects available in EF.DIR
        - EF.DIR and EF.ATR access services: by GET RECORD(s) command
        - Card with MF
    Tag: 7, len: 3 (card capabilities)
      Selection methods: FE
        - DF selection by full DF name
        - DF selection by partial DF name
        - DF selection by path
        - DF selection by file identifier
        - Implicit DF selection
        - Short EF identifier supported
        - Record number supported
      Data coding byte: 21
        - Behaviour of write functions: proprietary
        - Value 'FF' for the first byte of BER-TLV tag fields: invalid
        - Data unit in quartets: 2
      Command chaining, length fields and logical channels: 13
        - Logical channel number assignment: by the card
        - Maximum number of logical channels: 4
    Tag: 6, len: 7 (pre-issuing data)
      Data: 98 07 02 04 03 02 01
+ TCK = 41 (correct checksum)
```

Figure 4.8: PCSC Attached card reader detection

4.2.2 Building and Installing srsLTE

After installing the required supporting softwares, the srsLTE code was cloned and build using the steps mentioned in appendix A.4. It is pertinent to mention that it is always recommended to rebuild the srsLTE application if any new supporting software is installed.

The application was installed on two linux 18.04 based machines as one machine was running the srsUE. Whereas, EPC and the eNB were run on the other machine. Once the software installation is done the application is run by using the default configurations. First run the "sudo epc" on 1st machine to initialize the core network. The expected outcome is shown in figure 4.9.

```
rslab@rslab-1:~/srsLTE/srsepc$ sudo srsepc

Built in Release mode using commit 5343b33f on branch master.


---  Software Radio Systems EPC  ---

Reading configuration file /home/rslab/.config/srslte/epc.conf...
HSS Initialized.
MME S11 Initialized
MME GTP-C Initialized
MME Initialized. MCC: 0xf001, MNC: 0xff01
SPGW GTP-U Initialized.
SPGW S11 Initialized.
SP-GW Initialized.
```

Figure 4.9: EPC initialized

Once the epc is running, execute "sudo srsenb" command on the same machine but in another console to start the srseNB. The eNB will start and will connect to the srsepc. The expected srsenb console output is shown in figure 4.10.

```
rslab@rslab-1:~/srsLTE/srsenb$ sudo srsenb enb.conf
linux; GNU C++ version 7.4.0; Boost_106501; UHD_003.009.007-5-ge8202128


Built in Release mode using commit 5343b33f on branch master.

---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
Opening USRP with args: type=x300,master_clock_rate=184.32e6
-- X300 initialization sequence...
-- Determining maximum frame size... 1472 bytes.
-- Setup basic communication...
-- Loading values from EEPROM...
-- Setup RF frontend clocking...
-- Radio 1x clock:184.32
-- Initialize Radio0 control...
-- Performing register loopback test... pass
-- Initialize Radio1 control...
-- Performing register loopback test... pass
Setting frequency: DL=2685.0 Mhz, UL=2565.0 MHz
Setting Sampling frequency 23.04 MHz

==== eNodeB started ===
Type <t> to view trace
```

Figure 4.10: eNB initialized

Once the eNB is connected to the EPC, the connection can be verified seeing the srsepc console and expected output is shown in figure 4.11.

```
rslab@rslab-1:~/srsLTE/srsepc$ sudo srsepc

Built in Release mode using commit 5343b33f on branch master.


---  Software Radio Systems EPC  ---

Reading configuration file /home/rslab/.config/srslte/epc.conf...
HSS Initialized.
MME S11 Initialized
MME GTP-C Initialized
MME Initialized. MCC: 0xf001, MNC: 0xff01
SPGW GTP-U Initialized.
SPGW S11 Initialized.
SP-GW Initialized.
Received S1 Setup Request.
S1 Setup Request - eNB Name: srsenb01, eNB id: 0x19b
S1 Setup Request - MCC:001, MNC:01, PLMN: 61712
S1 Setup Request - TAC 7, B-PLMN 0
S1 Setup Request - Paging DRX 2
Sending S1 Setup Response
```

Figure 4.11: EPC console output with eNB connected

Once the core network EPC and basestation eNB is runnning, run UE on the second machine using the "sudo srsUE" command. The expected console output is shown in figure 4.12. The UE will search the available cells with a particular Public Land Mobile Network (PLMN) as per the international mobile subscriber identity (IMSI) number specified in the soft SIM configurations of 'ue.conf' file.

```
rslab@rslab:~/srsLTE/srsue$ sudo srsue ue.conf
[sudo] password for rslab:
linux; GNU C++ version 7.3.0; Boost_106501; UHD_003.009.007-5-ge8202128

Reading configuration file ue.conf...

Built in Release mode using commit 5343b33f on branch master.

---  Software Radio Systems LTE UE  ---

Opening 1 RF devices with 1 RF channels...
Opening USRP with args: type=x300,master_clock_rate=184.32e6
-- X300 initialization sequence...
-- Determining maximum frame size... 1472 bytes.
-- Setup basic communication...
-- Loading values from EEPROM...
-- Setup RF frontend clocking...
-- Radio 1x clock:184.32
-- Initialize Radio0 control...
-- Performing register loopback test... pass
-- Initialize Radio1 control...
-- Performing register loopback test... pass
Waiting PHY to initialize...
...
Attaching UE...
6859250
Searching cell in DL EARFCN=3400, f_dl=2685.0 MHz, f_ul=2565.0 MHz
```

Figure 4.12: UE Initialized

The UE will connect to the network as soon as it finds the available network. An IP address is assigned to the UE and it is released once it disconnects from the network. The srseNB console output once the UE is connected is shown in figure 4.13. and the console output of srsEPC is shown in figure 4.14.

```
rslab@rslab-1:~/srsLTE/srsenb$ sudo srsenb enb.conf
linux; GNU C++ version 7.4.0; Boost_106501; UHD_003.009.007-5-ge8202128


Built in Release mode using commit 5343b33f on branch master.

---  Software Radio Systems LTE eNodeB  ---

Reading configuration file enb.conf...
Opening USRP with args: type=x300,master_clock_rate=184.32e6
-- X300 initialization sequence...
-- Determining maximum frame size... 1472 bytes.
-- Setup basic communication...
-- Loading values from EEPROM...
-- Setup RF frontend clocking...
-- Radio 1x clock:184.32
-- Initialize Radio0 control...
-- Performing register loopback test... pass
-- Initialize Radio1 control...
-- Performing register loopback test... pass
Setting frequency: DL=2685.0 Mhz, UL=2565.0 MHz
Setting Sampling frequency 23.04 MHz

==== eNodeB started ===
Type <t> to view trace
RACH:  tti=3711, preamble=36, offset=5, temp_crnti=0x46
User 0x46 connected
```

Figure 4.13: srseNB console (UE Connected)

```
HSS Initialized.
MME S11 Initialized
MME GTP-C Initialized
MME Initialized. MCC: 0xf001, MNC: 0xff01
SPGW GTP-U Initialized.
SPGW S11 Initialized.
SP-GW Initialized.
Received S1 Setup Request.
S1 Setup Request - eNB Name: srsenb01, eNB id: 0x19b
S1 Setup Request - MCC:001, MNC:01, PLMN: 61712
S1 Setup Request - TAC 7, B-PLMN 0
S1 Setup Request - Paging DRX 2
Sending S1 Setup Response
Initial UE message: LIBLTE_MME_MSG_TYPE_ATTACH_REQUEST
Received Initial UE message -- Attach Request
Attach request -- IMSI Style Attach request
Attach request -- IMSI: 001010123456789
Attach request -- eNB-UE S1AP Id: 1
Attach request -- Attach type: 1
Attach Request -- UE Network Capabilities EEA: 11100000
Attach Request -- UE Network Capabilities EIA: 01100000
Attach Request -- MS Network Capabilities Present: false
PDN Connectivity Request -- EPS Bearer Identity requested: 0
PDN Connectivity Request -- Procedure Transaction Id: 1
PDN Connectivity Request -- ESM Information Transfer requested: false
Downlink NAS: Sending Authentication Request
UL NAS: Received Authentication Response
Authentication Response -- IMSI 001010123456789
UE Authentication Accepted.
Generating KeNB with UL NAS COUNT: 0
Downlink NAS: Sending NAS Security Mode Command.
UL NAS: Received Security Mode Complete
Security Mode Command Complete -- IMSI: 001010123456789
Getting subscription information -- QCI 7
Sending Create Session Request.
Creating Session Response -- IMSI: 1010123456789
Creating Session Response -- MME control TEID: 1
Received GTP-C PDU. Message type: GTPC_MSG_TYPE_CREATE_SESSION_REQUEST
SPGW: Allocated Ctrl TEID 1
SPGW: Allocated User TEID 1
SPGW: Allocate UE IP 172.16.0.2
Received Create Session Response
Create Session Response -- SPGW control TEID 1
Create Session Response -- SPGW S1-U Address: 127.0.1.100
SPGW Allocated IP 172.16.0.2 to IMSI 001010123456789
Adding attach accept to Initial Context Setup Request
Initial Context Setup Request -- eNB UE S1AP Id 1, MME UE S1AP Id 1
Initial Context Setup Request -- E-RAB id 5
Initial Context Setup Request -- S1-U TEID 0x1. IP 127.0.1.100
Initial Context Setup Request -- S1-U TEID 0x1. IP 127.0.1.100
Initial Context Setup Request -- QCI 7
Received Initial Context Setup Response
E-RAB Context Setup. E-RAB id 5
E-RAB Context -- eNB TEID 0x460003; eNB GTP-U Address 127.0.1.1
UL NAS: Received Attach Complete
Unpacked Attached Complete Message. IMSI 1010123456789
Unpacked Activate Default EPS Bearer message. EPS Bearer id 5
Received GTP-C PDU. Message type: GTPC_MSG_TYPE_MODIFY_BEARER_REQUEST
Sending EMM Information
```

Figure 4.14: srsEPC console (UE Connected)

Once the UE is connected to the network. We can see the PLMN, cell Radio Network Temporary Identifier (c-RNTI), IP assigned, mode, PRB and other information on the console output of the srsUE. Figure 4.15 shows the expected output.

It is important to mention that the TX and RX gain settings of srsue and srsenb play an important role. Too low or too high gain settings may cause problem in synchronization or correctly detecting the signal. The gain value depends on the distance between the srsenb and srsue. In our case one meter SMA cables were used for TX/RX connection and Tx_gain value was set to 10 and Rx_gain was set to automatic gain control (AGC) mode. Further more 30 dB attenuators were used

to protect the SDR from high RF power.

```
rslab@rslab:~/srsLTE/srsue$ sudo srsue ue.conf
linux; GNU C++ version 7.3.0; Boost_106501; UHD_003.009.007-5-ge8202128

Reading configuration file ue.conf...

Built in Release mode using commit 5343b33f on branch master.

---  Software Radio Systems LTE UE  ---

Opening 1 RF devices with 1 RF channels...
Opening USRP with args: type=x300,master_clock_rate=184.32e6
-- X300 initialization sequence...
-- Determining maximum frame size... 1472 bytes.
-- Setup basic communication...
-- Loading values from EEPROM...
-- Setup RF frontend clocking...
-- Radio 1x clock:184.32
-- Initialize Radio0 control...
-- Performing register loopback test... pass
-- Initialize Radio1 control...
-- Performing register loopback test... pass
Waiting PHY to initialize...
...
Attaching UE...
7227633
Searching cell in DL EARFCN=3400, f_dl=2685.0 MHz, f_ul=2565.0 MHz
.
Found Cell:  Mode=FDD, PCI=1, PRB=100, Ports=1, CFO=0.4 KHz
Found PLMN:  Id=00101, TAC=7
Random Access Transmission: seq=36, ra-rnti=0x2
RRC Connected
Random Access Complete.     c-rnti=0x46, ta=5
Network attach successful. IP: 172.16.0.2
Software Radio Systems LTE (srsLTE)
```

Figure 4.15: srsUE console After connection to the Network

As soon as srsUE successfully attaches to a network a virtual tunnel interface "tun_srsue" is created to send and receive data traffic. Figure 4.16 shows the UDP traffic on our "tun_srsue" during the iperf testing.

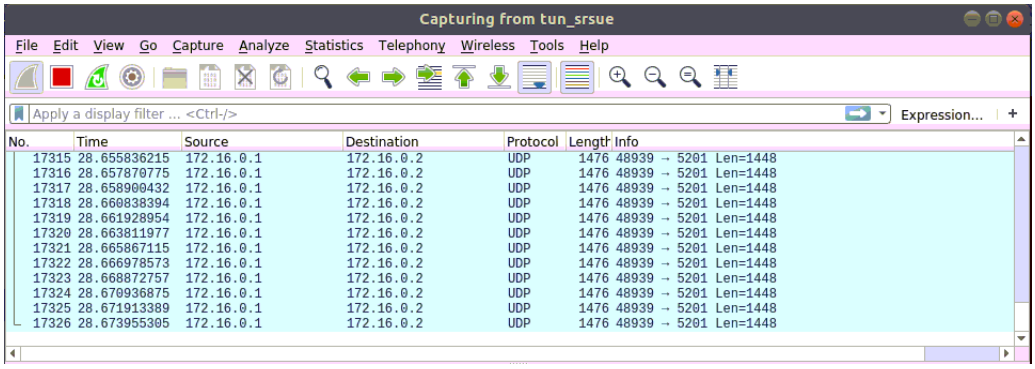


Figure 4.16: UDP Traffic on Tun_srsue interface - UE

On the network side or the machine running the srsepc and srsenb a tunnel interface "srs_spgw_sgi" is created by the srsepc to send and receive packets from the UE via eNB. Figure 4.17 shows the UDP traffic on our "srs_spgw_sgi" during the iperf testing.

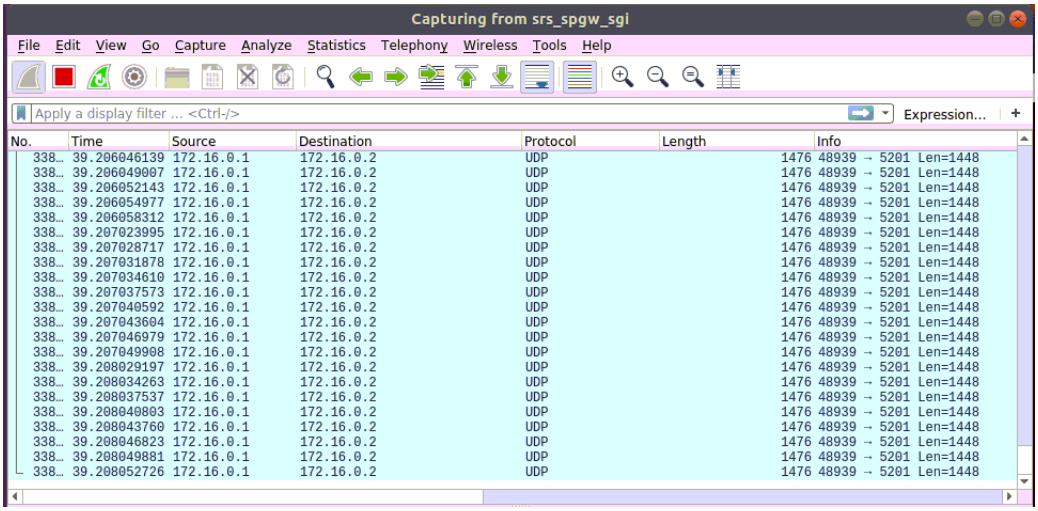


Figure 4.17: UDP Traffic on srs_spgw_sgi interface - EPC

4.3 Configurations

The srsLTE all the components srsue, srsepc and srsenb have the detail configuration files. The configurations of each component are briefly discussed below.

4.3.1 UE Configurations

The UE default configurations are listed in "ue.conf.example" file in the "/srslte/srsue" folder. The UE configuration file is divided in different parts consisting of RF, PCAP, Log, USIM, RRC, NAS and expert configurations.

In the RF configurations, parameters related to operating frequencies, TX/ RX power gains and device antenna modes are specified. In next part PCAP, MAC-layer packet capture properties are defined. In the log configuration different levels can be selected like debug, info, warning or error option to enable log generation in the 'tmp' folder. Furthermore, soft USIM parameters are defined in the USIM configurations. Option to choose the PC/SC mode or soft sim mode is also defined in this part. UE category and release is defined in the RRC part. NAS related parameters like apn, integrity algorithm and ciphering algorithms are defined in the NAS section. Last section is the expert section which defines different parameters.

All the configurations in the config file can be edited using the command line like for changing the EARFCN use "sudo srsue /.config/srslte/ue.conf –rf.dl_earfcn 2700".or just go to folder "/srslte/srsue" and run "cp ue.conf.example ue.conf" this will create a copy of all the default configurations in "ue.conf" file. Now if any of the parameter is changed ue.conf the new settings can be executed using "sudo srsue ue.conf" command.

4.3.2 eNB Configurations

The default eNB configurations can be found in "enb.conf.example" file. It includes the eNB configurations like eNB id, cell id, MCC, MNC etc. Next it has the supporting configurations file section containing sib.conf file to configure the system information blocks (SIBs), rr.conf for radio resource configurations and drb.conf file

for data bearers configurations. RF configuration sections parameters are used to specify operating frequencies, transmit/ receive power levels and device antenna modes. Enb.conf also has the log file configuration option and PCAP options as described for UE. The UL and DL MCS values can be specified in the scheduler section of the enb.conf. Lastly there is a section defining the expert options.

It is important to mention while running the eNB with PRB 6 change the "prach_freq_offset" value from 2 to 0 in sib.conf file. And as mentioned in above section all the configurations in the config file can be edited using the command line or by changing the configurations files in the "/srslte/srsenb" folder and execute using the "sudo srsenb enb.conf" command.

4.3.3 EPC Configurations

The EPC configurations are stored in two main files epc.conf containing the general configurations for the MME, HSS and GW parameters. Second file is the "user_db.csv" which contains the users information (IMSI, authentication algorithms, K, OP or OPc, etc) and is used by the HSS. In this project lab tests are conducted using the default EPC configurations.

Chapter 5

Results and Analysis

This chapter will analyze the results obtained during the testing of the srsLTE. The test metric includes DL data rate test, SNR test, RF characteristics and interference analysis. For data rate and RF power output tests, the achieved results are plotted and in case of data rate, a comparison is drawn with the standard requirements. Whereas, to observe the interference generated by the SDR the RF spectrum was monitored using a spectrum analyzer. Further more a comparison analysis is presented between USRP and limeSDR in section 5.3.

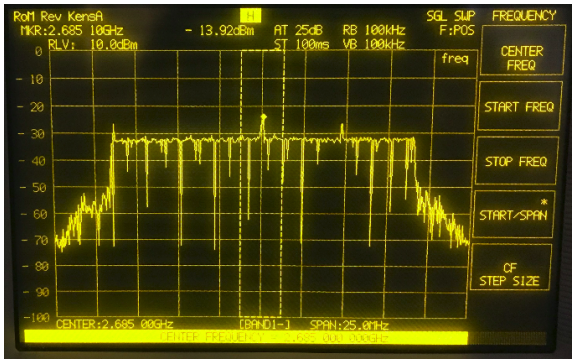
5.1 eNB

After the successful implementation, the RF performance characterisation was done for the srseNB part and the srsUE. RF power output plays a important role in overall system performance [32]. So to full-fill the transceiver design requirements tests like power off test, minimum/maximum output power and spurious emissions test are performed. Further more, it is important to mention that the output power and performance varies depending on the frequency band and the hardware platform used.

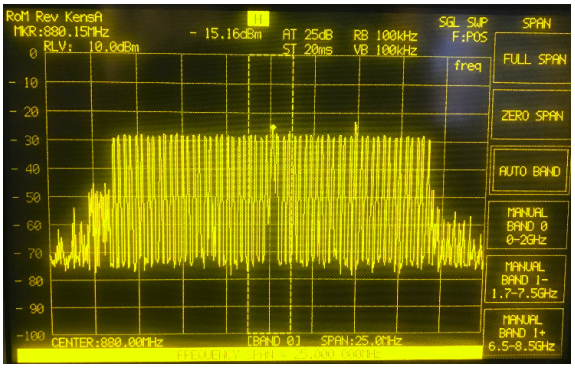
In this project we have used USRP x310 as the eNB base station for our test setup so RF test results are discussed in below section.

5.1.1 RF Power Output

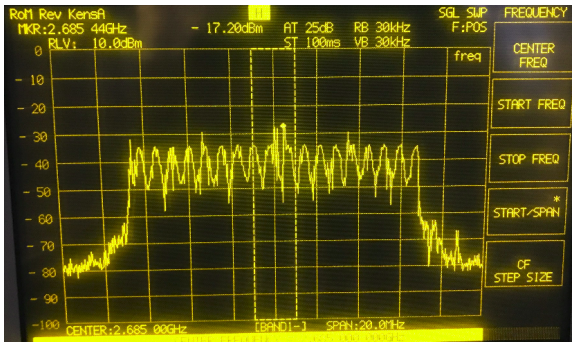
First to check the srseNB generates the OFDM signals for different PRB settings, we did the setup as described in section 3.2.1.2. The output was observed in two bands i.e. 800 MHz band and the 2.6 GHz band. The output spectrum plots are shown below in figure 5.1. We can see that in 800 MHz band the generated OFDM signals looks more refine than the 2.6 GHz band. During our lab testing we have used one meter SMA TX/RX cables between the UE and eNB. Therefore, this change in RF characteristics may not effect on the overall performance. However, in real long range scenarios results might differ for both the frequencies.



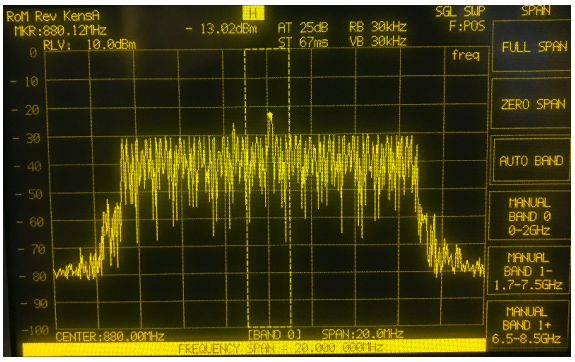
(a) PRB 100 - 2.6GHz Band



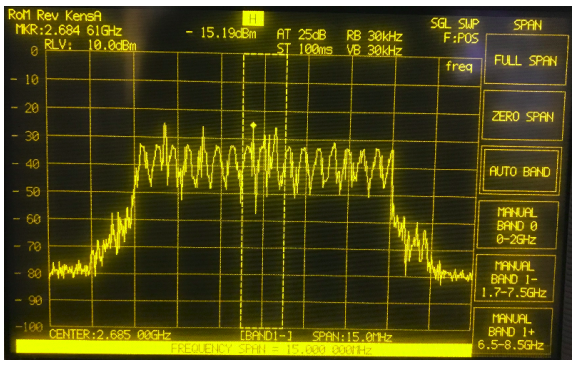
(b) PRB 100 - 800MHz Band



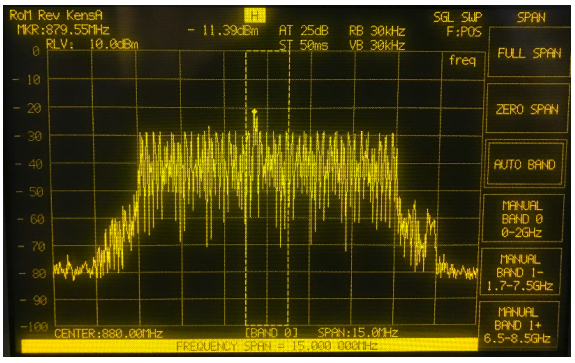
(c) PRB 75 - 2.6GHz Band



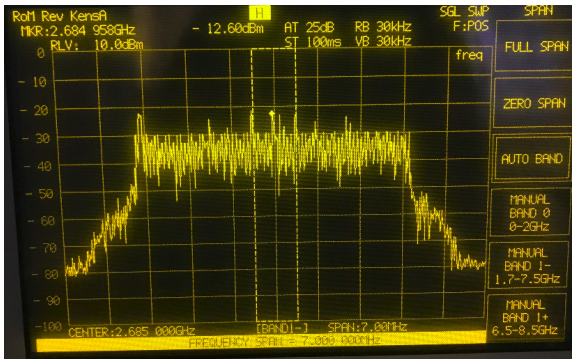
(d) PRB 75 - 800MHz Band



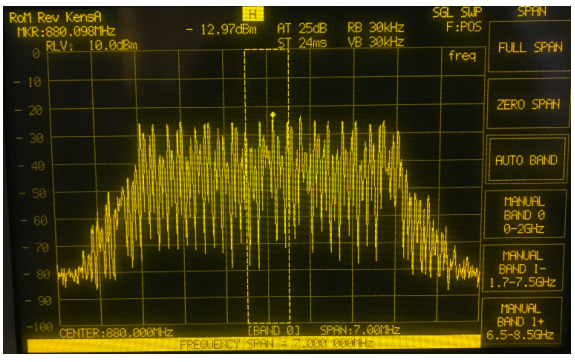
(e) PRB 50 - 2.6GHz Band



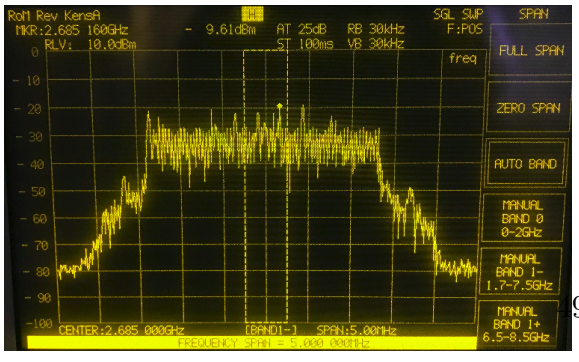
(f) PRB 50 - 800MHz Band



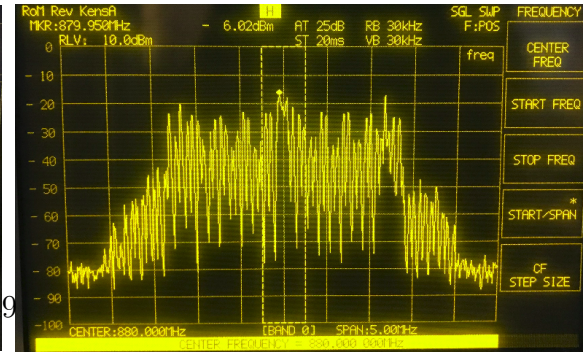
(g) PRB 25 - 2.6GHz Band



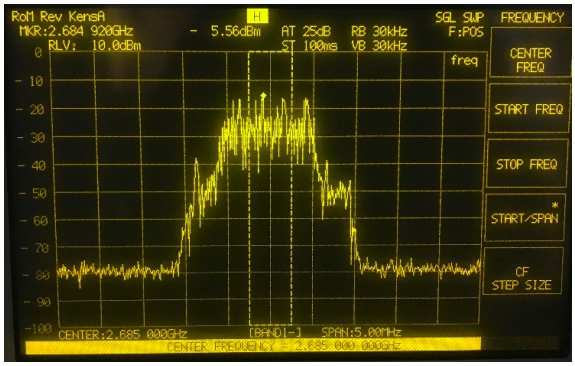
(h) PRB 25 - 800MHz Band



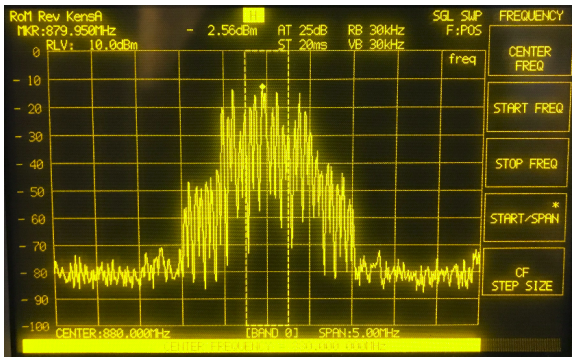
(i) PRB 15 - 2.6GHz Band



(j) PRB 15 - 800MHz Band



(k) PRB 6 - 2.6GHz Band



(l) PRB 6 - 800MHz Band

Figure 5.1: A figure that contains three subfigures

Moving on with the RF tests, next step is to find the minimum and maximum output power. And to locate the saturation point of the eNB RF power output. This will help to find the ideal eNB output power for our tests. The test results for different PRB settings are shown in figure 5.2. The test was conducted in 2.6 GHz band and it can be seen that the RF output power becomes almost constant after Tx_gain value of 30 for all bandwidths/PRB settings. This shows that gain values higher than 30 may generate more spurious signals or may cause inter-modulation/interference issues. Further, we can see that the achieved output power decrease as we increase the bandwidth.

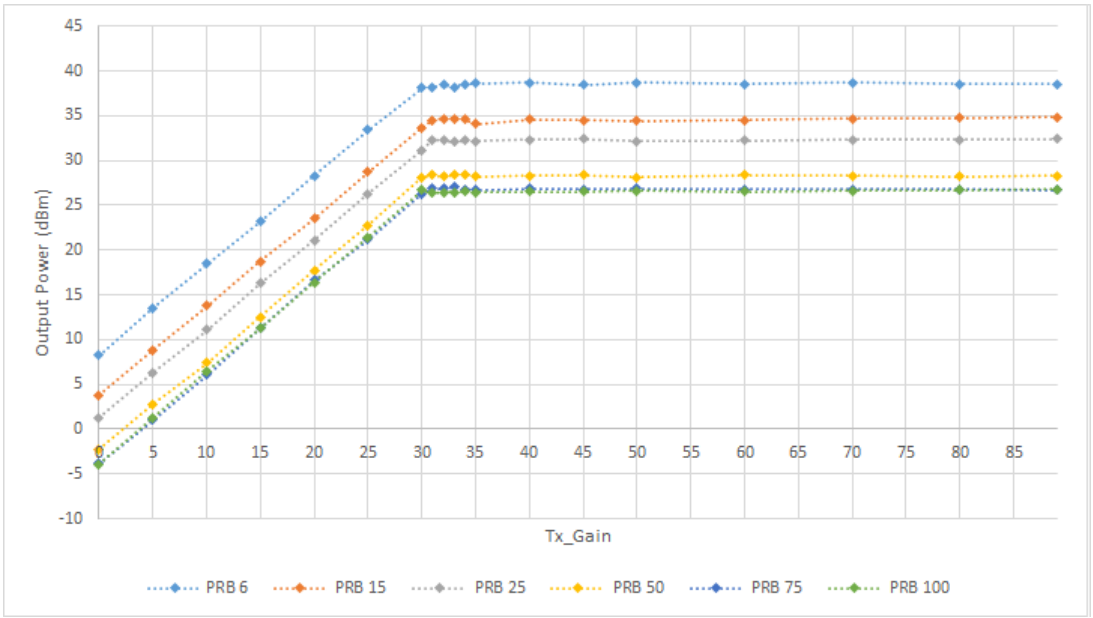


Figure 5.2: srseNB Output Power Test using USRP

5.2 UE (USRP X310)

The SDR based LTE UE Cat 4 modem is evaluated based on DL data rates for each bandwidth and MCS settings, the SNR of the received signal and RF performance analysis was performed. The results obtained are shown and analyzed below.

5.2.1 Downlink Data Rate

The DL data rate is one of the key element in evaluating any LTE UE. So for the lab tests and to observe how close is the performance of SDR based LTE modem with the 3GPP standard in terms of down link data rates the setup was done as shown in figure 3.1.

It is important to know that in an LTE based system the achieved data rate majorly depends on three factors bandwidth, wireless channel quality and number of active users. As we know that the LTE based communication system can support six different bandwidth configurations i.e. 1.4 MHz, 3 MHz, 5 MHz, 10 MHz, 15 MHz and 20 MHz. The bandwidth plays an important role as higher the bandwidth, higher is the resulting data rate. In LTE system bandwidth is divided into number of resource blocks (RB) so higher the bandwidth more are the number of resource blocks. The mapping is shown in table 5.1 below:

Table 5.1: LTE BW and PRB Mapping

Bandwithd (MHz)	1.4	3	5	10	15	20
PRB	6	15	25	50	75	100

Next element which affects the data rate is the channel quality or propagation medium quality. LTE uses adoptable date rate scheme and eNB selects the MCS based on the radio channel conditions. The channel quality is reported by the UE in CQI message to the eNB. Better the channel, higher is the CQI and higher is the data rate achieved. eNB selects the MCS based on the CQI reported as defined in section 7.2.3 of 3GPP standard 36.213 [3]. Next the UL/DL data rates also depend on number of active users as the resources are divided among the active users, meaning higher the number of users low is the data rate achieved for each user.

The overall throughput of the system can further be increased by using multiple antenna techniques like 2x2 or 4x4 MIMO. In this project only SISO configuration has been used by manually configuring the available bandwidth and MCS in "enb.conf" file of srseNB. Secondly the CQI reporting was also fixed to maximum i.e. 15 in the "ue.conf" file of srsUE. One subscriber i.e. SDR based LTE UE modem (srsUE) was attached to check the maximum data rate achieved for each bandwidth and MCS settings.

To observe the maximum DL data rate achieved iperf3 was used. The UE was run in server mode and eNB in client mode. The experiment was run for one hundred thousand iterations for each MCS settings and the average was taken. The achieved results along with the theoretical achievable values and are plotted in figure 5.3 and 5.4. The dotted plots shows the maximum achieve able data rate and the solid line plots show the actual achieved data rates, which is the average of one hundred thousand continuous iterations. The theoretically achievable values were obtained by using the table 7.1.7.1-1 and 7.1.7.2.1-1 of TS 36.213 [3] and procedure is explained in appendix C.

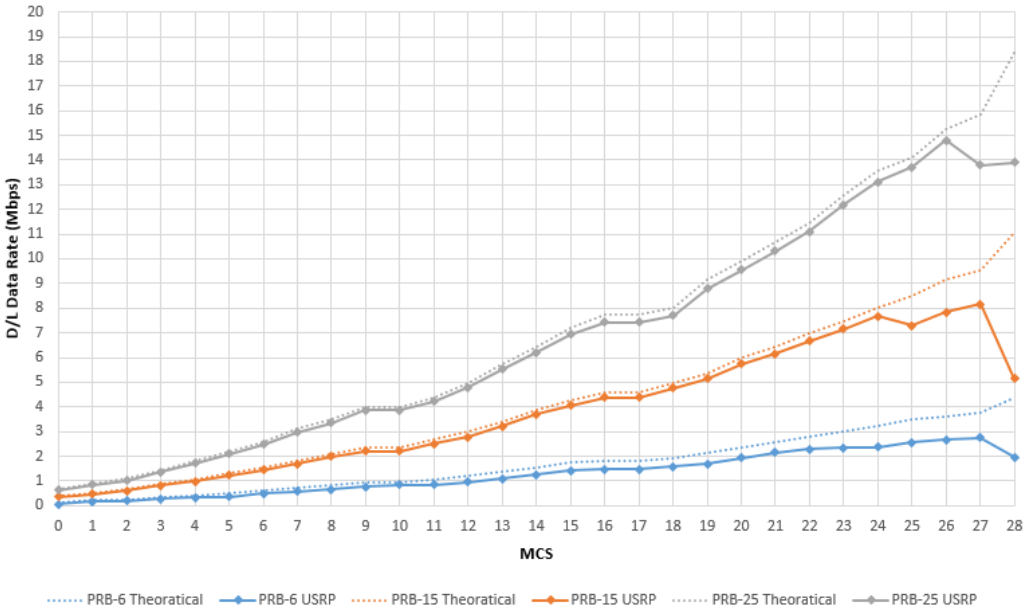


Figure 5.3: UE Downlink Data Rate (USRP) - 1/2

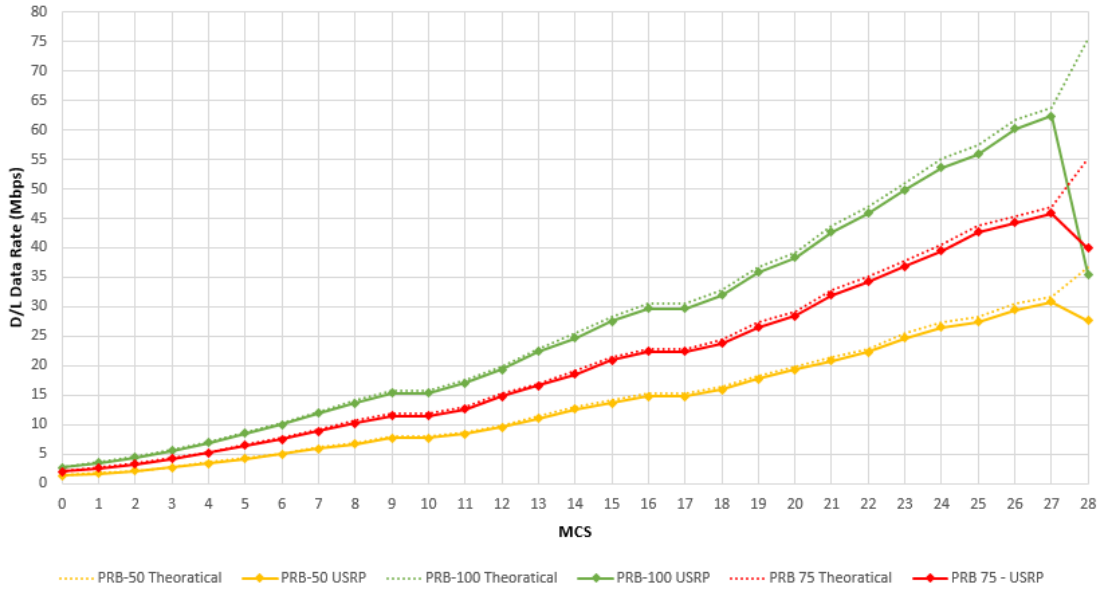


Figure 5.4: UE Downlink Data Rate (USRP) - 2/2

It can be seen from the plots that the srsUE performance in terms of DL data rate is more close to the 3GPP standard requirements for higher bandwidths (i.e. PRB 50, 75 and 100), with an average difference of around 3 to 4 percent between the data rate achieved and theoretical possible. Whereas, high difference in data rate achieved and theoretically possible is observed for PRB 6. Same is depicted in figure 5.5 and 5.6. Moreover if we see figure 5.5, there are fluctuations in the difference percentage for maximum bandwidth achievement for PRB 6. For PRB 15 the difference increases starting from MCS 25 till MCS 28. Similarly for PRB 50 we see the fluctuations for MCS values 1 to 3 and than the difference increases in achieved data rate for MCS 27 and 28. However, for PRB values 50, 75 and 100 the difference in data rate achieved and expected is quite low as well as stable till MCS

27. But the difference increases exponentially for MCS value of 28. This sudden increase in the difference between achieved and theoretically required DL data rate at MCS 28 happened for all PRB settings.

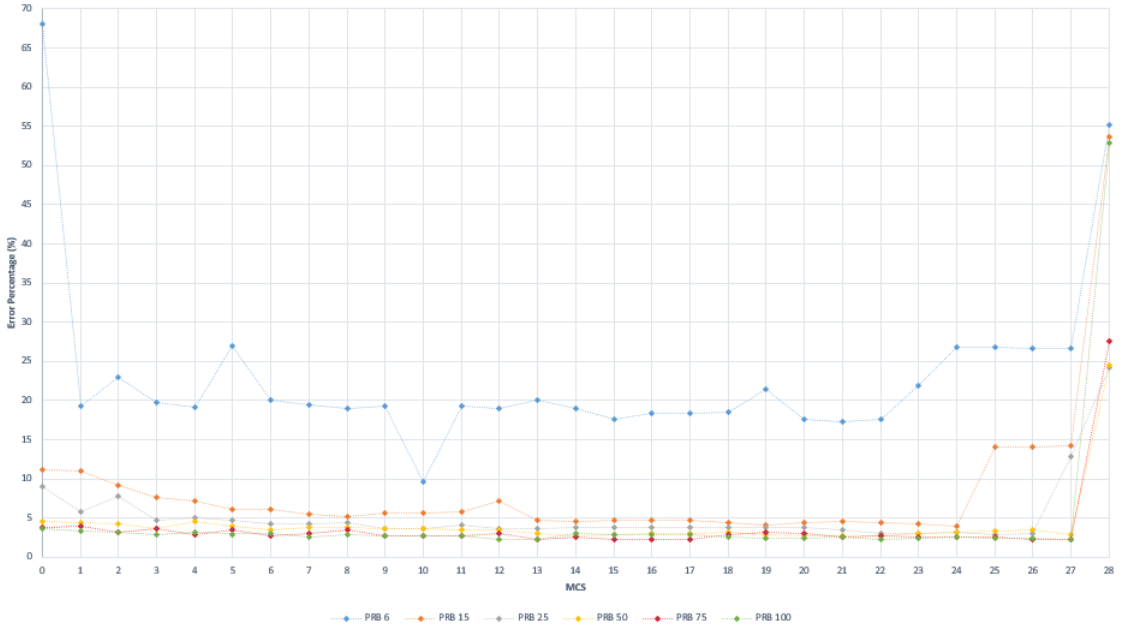


Figure 5.5: Error percentage plot

The figure 5.6 shows the average difference in data rate achieved and expected against each PRB setting. The graph is the average of values plotted in figure 5.5. The red plot is the average taken including all MCS values from 0 to 28. Whereas, the grey plot shows the average difference after removing the one outlier which is MCS 28 for each bandwidth setting. The lowest average error percentage is observed for PRB 100 and maximum for PRB 6 after removing the outlier value at MCS 28. This sudden reduction in achieved data rate for MCS 28 for all the bandwidth shows the limitation of this application to process higher Transport Block Size (TBS). The issue seems to lie at the application end as if it was happening due to the channel conditions than the trend would not be the same for all the bandwidth settings.

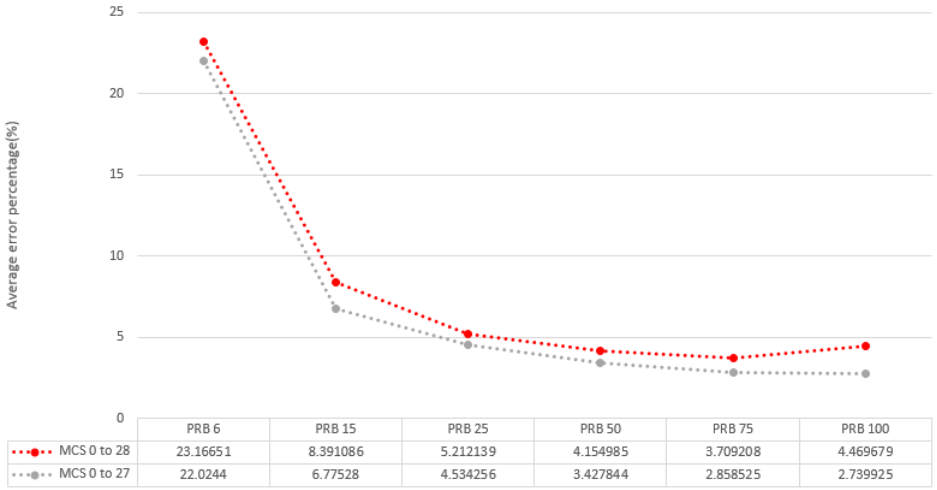


Figure 5.6: Average error percentage plot against PRB values

5.2.2 SNR

The signal to noise ratio of a wireless signal plays an important role in the over all performance of a mobile communication system. If the SNR drops below the certain level the demodulation cannot be performed or more errors will occur [32]. To measure the SNR variations of the received signal the values were recorded in a CSV file. The SNR values were recorded against each bandwidth and MCS settings while the data rate tests were run. The average of the recorded values were taken and are plotted in figure 5.7.

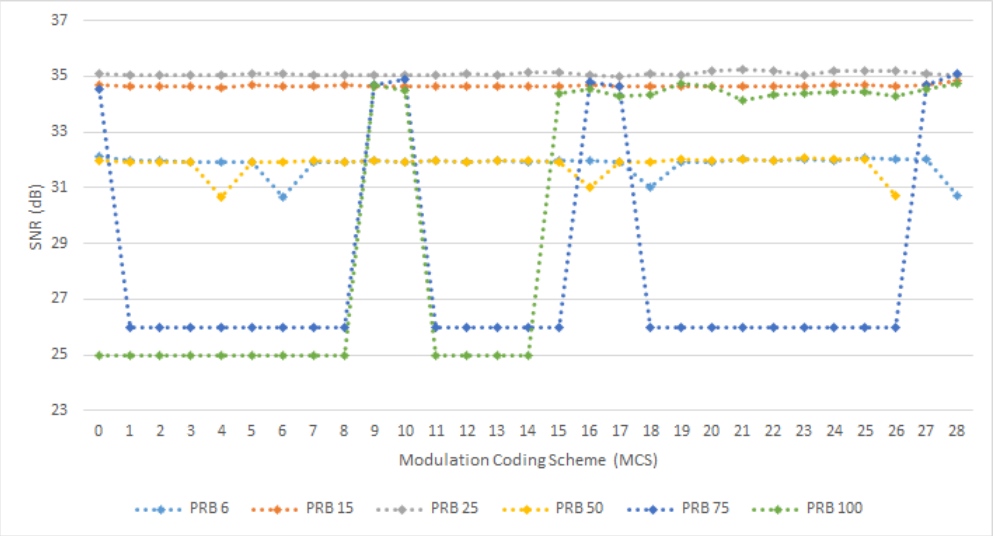


Figure 5.7: Downlink SNR at UE

We can see that SNR remains almost constant for PRB settings 6, 15, 25 and 50 with a few fluctuations less than 3 dB. However, huge fluctuations in the SNR values were observed for PRB settings 75 and 100. Ideally this should not happen as one meter coaxial cables were used as a Tx/Rx connection between the eNB and UE. These fluctuations shows the limitation of doing signal processing tasks on a GPU. The low SNR values of the received signal for PRB 75 and 100 may cause issue as it affects the effective coverage area of an eNB [32].

5.2.3 RF Power Output

The RF power output trend of the UE was observed by changing the Tx_gain value in the 'ue.conf' file. The setup for the test is shown in figure 3.3. As per [5, 32] the maximum and minimum output power tests should be performed at the maximum LTE BW i.e. 20 MHz. However, in our case we measured RF power of the PRACH transmitted by the UE to measure srsUE output power. The test was done in 800 MHz and 2.6 GHz band and the results are plotted in figure 5.8. We can see that the output power of the UE is linear and the maximum output power is within prescribed limits as per 3GPP standard [5, 32]. However, if we look at the minimum output power, it is bit too high as compare to the standard which requires it to be -39 dBm. Overall the RF output power achieved in 2.6 GHz band is 3 to 4 dB better than the 800 MHz band. So to have a better idea on power range the test should

be conducted in all the bands supported by LTE as described in 3GPP TS 36.521-1 [5].

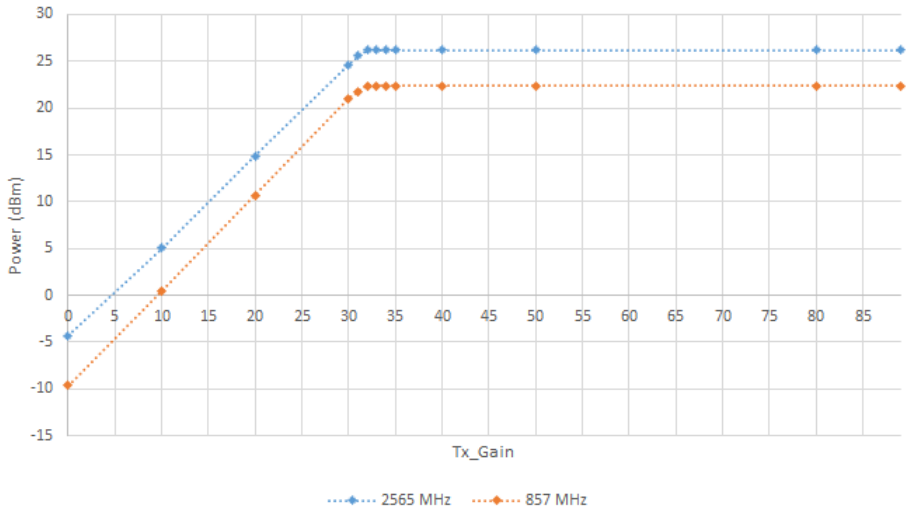


Figure 5.8: UE RF output Power

5.3 USRP x310 vs LimeSDR

As we know that cost is an important factor when it comes to the large scale usage of any technology. So to have get some idea how a cost effective SDR platform will perform as compared to a expensive high end USRP. The limeSDR was the choice as mentioned in chapter 2. The UE peak DL data rate, SNR and RF power output tests were conducted using the limeSDR. The results are plotted in figures below to draw a comparison between the two SDR platforms.

First we can see in the figure 5.9 and 5.11 that the maximum achieved data rate and corresponding average SNR values are almost equal to the USRP. Except for one exception for PRB 6 in case of peak DL data rate.

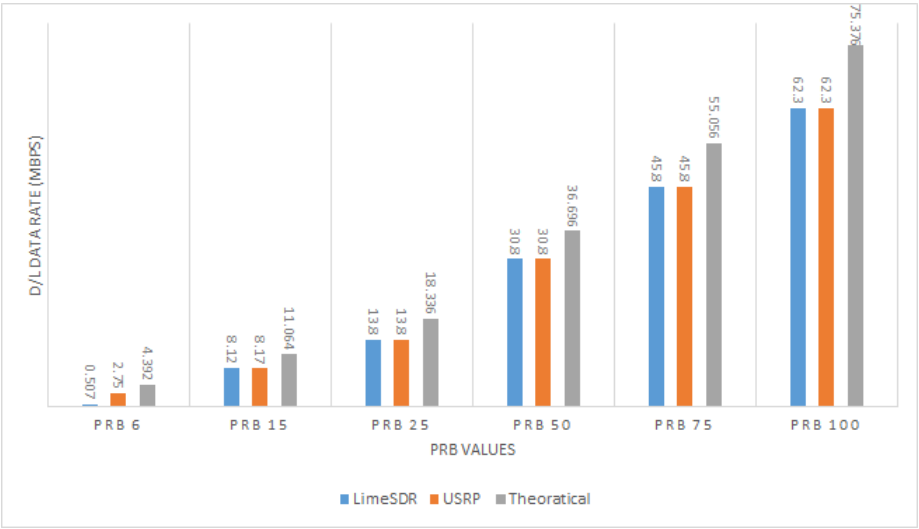


Figure 5.9: UE - LimeSDR Peak D/L Data Rate Comparison

The difference in achieved peak data rates in case of USRP and limeSDR as compared to the standard requirement, is shown in figure 5.10. The red dotted line shows the difference percentage in data rate achieved for limeSDR and the orange line represents the USRP x310 peak data rate difference. We can see that the difference in the peak data rate achieved is almost the same for both the SDRs, except for PRB 6 settings where it is extremely high in case of limeSDR.

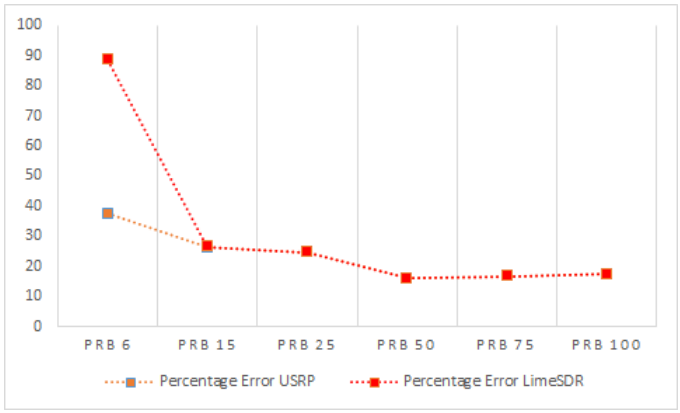


Figure 5.10: Peak data rate error percentage USRP vs LimeSDR

Next if we look at the SNR plot the difference of around 1 to 4 dB is seen between the values. Which means limeSDR sensitivity value is a little higher, more power at the input maybe required to achieve the same performance as of USRP. This high input sensitivity value directly affects the coverage area [32].

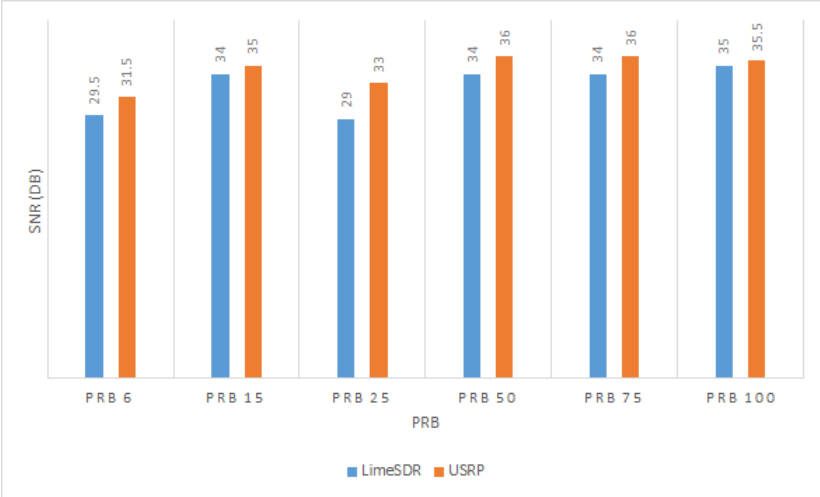


Figure 5.11: UE - LimeSDR SNR Comparison

Furthermore, the RF power output trend of the limesdr based UE was observed by changing the Tx_gain values. The test was done in 800 MHz and 2.6 GHz band. The setup for the test was same as for the usrp. The test results are plotted in figure 5.12. We can see that the output power of the UE is very non-linear and the maximum output power also exceeds the prescribed limits as per 3GPP standard [5, 32]. This non linearity in the output power may distort the output signal by causing the interference between the sub carriers. Overall if we see the RF output power

achieved in 800 MHz band is 3 to 4 dB better than the 2.6 GHz band which is exact opposite of USRP. So to have a better idea on power range the detail tests should be conducted in different bands and settings as described in 3GPP TS 36.521-1 [5].

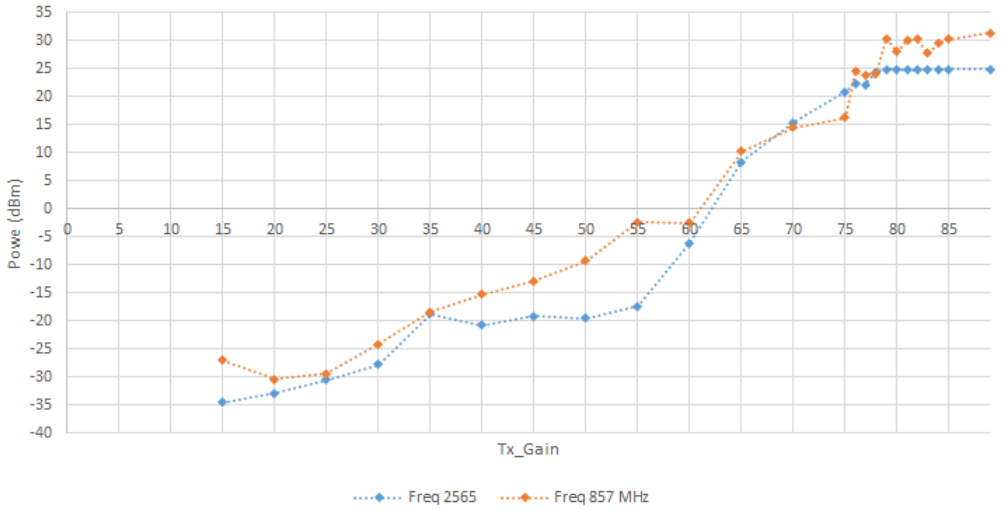


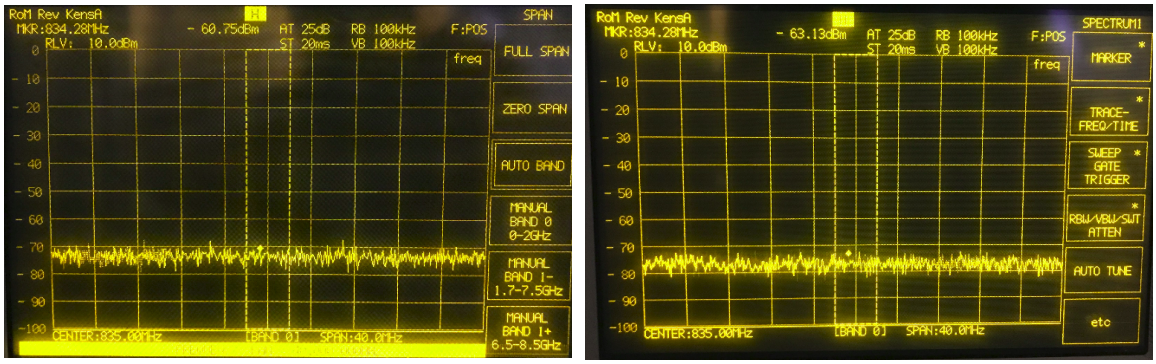
Figure 5.12: UE - LimeSDR Output Power test

5.4 RF Conformance Test

The conformance tests are done to validate if the UE RF characteristics comply with the 3GPP standards and other regulatory authorities. The testing is usually done by the standard organizations or some third party organizations to validate the compliance.

The LTE UE transmitter and receiver conformance specifications along with the requirements and test procedures are defined in 3GPP Specification TS 36.521-1 Radio Transmission and Reception LTE User Equipment [5]. The RF tests include transmit power tests, power control, out of band transmission analysis, spurious emission, transmit inter-modulation tests and receiver sensitivity tests. Specialized equipment's are required for detail RF conformance testing [5, 32]. However, to get some insight we have performed a few tests using the spectrum analyzer to analyze the RF characteristics of srsUE with USRP x310 and limeSDR.

The tests done are transmit power off, UE minimum & maximum output power and interference or spurious emissions analysis. The setup for the tests is described in section 3.2.1.3. First transmit off test results are shown in figure 5.13. As per requirement in section 6.3.3 of [5], the mean power when the transmitter is off should be $\leq$ -48.5 dBm for carrier frequency f $\leq$ 3.0GHz and $\leq$ -48.2 dBm for carrier frequency 3.0GHz $<$ f $\leq$ 4.2GHz. Both SDR hardware platforms full-fill the requirement as the mean power when the transmit is off is around -60 dBm in-case of USRP and around -63 dBm for limeSDR.



(a) UE - USRP x310 Transmit Off

(b) UE - LimeSDR Transmit Off

Figure 5.13: UE Transmit Off

Further more, minimum and maximum transmit power of the UE was measured with limeSDR and the USRP. As per the 3GPP standard [5], the maximum output power is 23 dBm with a tolerance of ± 2.7 dB for class 3 and 5 UE. Whereas, the minimum power requirement is −39 dBm with a tolerance of 1dB for frequency f ≤ 3.0 GHz and a tolerance of 1.3dB for frequencies above 3.0 GHz and less than 4.2 GHz [5].

The maximum and minimum RF output power values achieved with limeSDR and USRP are summarized in table 5.2. We can see that the maximum required values are close to the requirements for both limeSDR and the USRP. However, big difference is seen in the minimum required output power. This UL power control is an important factor in cellular communication as it limits the interference generated and maximizes the power of the desired received signal.

Table 5.2: UE USRP x310 vs limeSDR RF Parameters [5]

	Power output - USRP x310 (dBm)	Power output - LimeSDR (dBm)	Standard Requirements
Minimum	-9.6 (800MHz) -4.3 (2.6GHz)	-26.97 (800 MHz) -34.5 (2.6 GHz)	-39 dBm
Maximum	22.4 (800MHz) 26.19 (2.6GHz)	31.47 (800 MHz) 24.88 (2.6 GHz)	23 ± 2.7 dBm
Power off	-60 dBm	-63 dBm	< ≤ -48.5 dBm

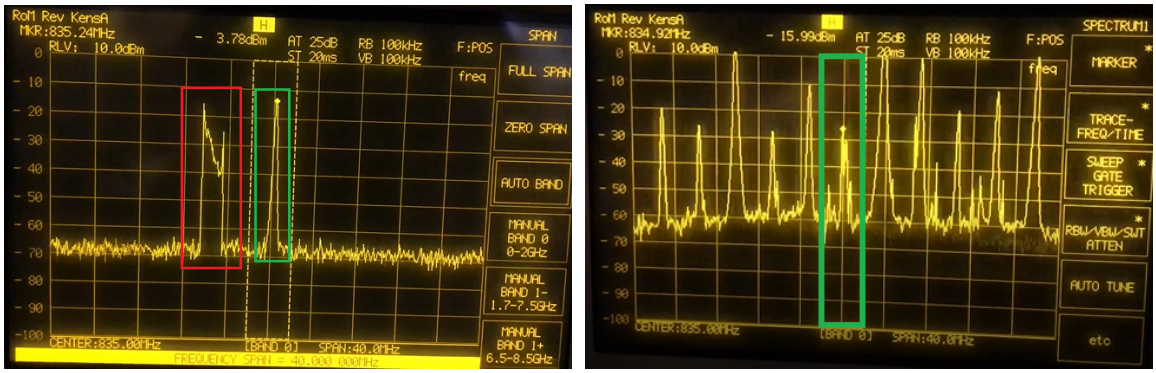
5.4.1 Interference Analysis

Spurious emissions, inter-modulation products or out of band emissions are one of the major issues in the area of wireless communications. These types of unwanted signals not only interfere with the user desired signal but may cause degradation to other neighbouring users.

The figure 5.14 below shows the interference generated by the limeSDR and the USRP at maximum power output. The test was conducted in 800 MHz and span was 40 MHz. The plots are taken while keeping Tx_gain value to 89. It was observed that the amplitude of the interference increases with gain as it is increase beyond saturation point i.e. approximately Tx_gain value of 30.

It can be seen that limeSDR based UE generates a lot of inter-modulation interference along with the spurious emission and suppressing the desired signal.

Whereas, in case of USRP the desired signal is not affected by the spurious emissions probably caused by LO leakage. The limits of spurious emissions is described in section 6.6.3 of TS 36.521-1.[5]. Further more, another issue observed with limeSDR was RF power not turning off at high power even after srsue application is turned off.



(a) UE - Interference generated by USRP (b) UE - LimeSDR Max Power Output

Figure 5.14: USRP & LimeSDR interference Analysis

5.5 Testing with Commercial LTE basestaion simulator

To verify the implemented solution is as per the standard. The prototype was tested with the LTE commercial base station simulator CMW 500 by R&S. The setup is show in figure 3.4. The test was done using both soft USIM and PCSC mode.

First the UE soft sim parameters were configured as per the CMW 500 database as shown in figure 5.15 below:

```
[usim]
mode = soft
algo = milenage
opc  = 00000000000000000000000000000000
k    = 000102030405060708090A0B0C0D0E0F
imsi = 001010123456063
imei = 353490069873319
```

Figure 5.15: srsUE soft SIM configurations for testing with CMW 500

The UE was tested in soft sim and PCSC mode with authentication, AS security and NAS security enabled and disabled. The UE was connected successfully and an IP was assigned in-case when the security and authentication was turned off. However, the UE receives the Scheduling Request (SR) failed message as shown in figure 5.16. Which means the UE requested for accessing the PUSCH resource to send data and the response was negative from the eNB side. The reason could be that the UE SR request did not reach in time to the eNB, scheduler type did not match on each side or maybe the system may lost synchronization and the SR is transmitted at incorrect time slots. As soon as SR request failed the UE lost the RRC connection and started the RA procedure again. The same behaviour was observed for both PCSC and soft SIM mode of the srsUE.

```
Attaching UE...
Searching cell in DL EARFCN=3300, f_dl=2675.0 MHz, f_ul=2555.0 MHz
..
Found Cell:  PCI=0, PRB=6, Ports=1, CFO=-0.3 KHz
Found PLMN:  Id=00101, TAC=1
Setting PDN protocol to IPv4
Random Access Transmission: seq=39, ra-rnti=0x7
Random Access Complete.     c-rnti=0x46, ta=0
RRC Connected
Network attach successful. IP: 192.168.48.129
Scheduling request failed: releasing RRC connection...
Random Access Transmission: seq=26, ra-rnti=0x9
```

Figure 5.16: UE Attached To CMW 500

Next, with security and authentication enabled and the UE in PCSC mode the UE PCSC mode settings are shown in figure 5.17.

```
[usim]
mode = pcsc

reader = HID Global OMNIKEY 6121 Smart Card Reader [OMNIKEY 6121 Smart Card Reader]
pin  = 0000
```

Figure 5.17: srsUE PC/SC Configurations

The UE was rejected connection and received an attach reject message with cause #3 as shown in figure 5.18. Which means either the UE IMSI number did not matched the database or the authentication failed because of RES mismatch as described in annex A of [7]. In this case the issue was authentication as message displayed on the CMW 500 was "authentication failure".

```
Attaching UE...
Searching cell in DL EARFCN=3300, f_dl=2675.0 MHz, f_ul=2555.0 MHz
..
Found Cell:  PCI=0, PRB=25, Ports=1, CFO=0.4 KHz
Found PLMN:  Id=00101, TAC=1
Setting PDN protocol to IPv4
Random Access Transmission: seq=35, ra-rnti=0x9
Random Access Complete.     c-rnti=0x93, ta=0
RRC Connected

Warning: Network authentication failure
Received Attach Reject. Cause= 03
Scheduling request failed: releasing RRC connection...
Random Access Transmission: seq=11, ra-rnti=0x9
```

Figure 5.18: UE Authentication Reject CMW 5000

Further more, in soft SIM mode of srsUE and security & authentication enabled NAS reject was recieve in the UE log with cause #18. Which happens if the UE request the circuit switched (CS) service and it is not supported by the MME [7].

The issues faced while testing the srsUE with the commercial base station simulator highlighted some of the limitations of the srsUE. More tests maybe done to exploit the the full potential and limitations of the implemented SDR based UE modem.

Chapter 6

Conclusion and Future Work

In this project a feasibility study has been performed to build a LTE Cat 4 modem using the SDR technology, keeping in view a use case of large scale commercial implementation for trucks connectivity. The most high-end SDR platform USRP x310 and a relatively cost effective option limeSDR is used to implement and analyze the performance of the the available open source LTE software suite i.e., srsLTE. The results show that the performance is dependent on both the CPU processing power and the selected SDR hardware platform. High CPU power is required to process higher bandwidths [18]. First the detail performance analysis is performed using the USRP x310. The metric used for performance analysis includes DL data rates measurements, SNR, RF power output and analyzing the interference generated.

The results show that the DL data rates achieved are better for higher bandwidths i.e 10 MHz, 15 MHz and 20 MHz as compared to the lower ones i.e. 1.4 MHz, 3 MHz, and 5 MHz. The average error difference from the standard requirement is around 2 percent for higher bandwidths. Whereas, for lower bandwidths 1.4 MHz to 5MHz the error difference varies from 5 to 23 percent. The highest error rate i.e. 23 percent was observed for 4 MHz or PRB 6 which shows that application is not suitable to operate in such lower bandwidths. Another strange phenomena is observed for MCS value of 28, the data rate falls drastically at this value. The issue seems to be the limitation of the application as the pattern was repeated for all bandwidths/PRB settings. Next the SNR recorded is stable for most of the BW/PRB settings. However, large fluctuations in SNR were observed in higher bandwidths i.e. 15 MHz and 20 MHz. Which shows the limitation of the implemented SDR based UE in terms of coverage for higher bandwidth settings. The RF power output test of the USRP yielded satisfactory results as per standard in terms of maximum output power and transmit off test. However, minimum output power was quite high as compared to standard requirement. Detail RF conformance testing maybe performed in different bands to draw a final conclusion.

The second tests were performed with the cost effective option i.e. limeSDR to draw a comparison and look into the limitations. First the peak DL data rate test shows that the peak achievable data rates almost the same as the USRP for each PRB/bandwidth setting with an exception of PRB 6 were the data rate achieved was extremely low i.e. 0.507 Mbps as compared to 2.75 Mbps in case of USRP. The SNR recorded was slightly lower than the USRP with a difference ranging from 1dB to 4dB. The low SNR effects the effective coverage area of an eNB as more TX power is required to achieve the desired data rates. Further more the limeSDR based UE

RF power output shows huge non-linearities which could cause signal distortion. The RF interference generated by the limeSDR based UE is also one of the concerns which needs to be looked into. Overall the performance of the USRP x310 based UE is better than the limeSDR. Especially the RF performance which is an important factor while designing any wireless communication system. The results shows that a detail RF conformance testing should be performed while choosing a suitable SDR platform for a specific application.

6.1 Discussion

SDR based systems are evolving with a higher pace keeping in view the increasing customer demands and advancement in cellular technology. We tried to look into the issues related to performance, limitations, and interoperability of this technology in an existing commercial setup. The quantitative results in terms of data rates are close to the standard requirements with a few exceptions. However, the RF performance is a challenge. Although the tests are performed in a lab setup without the bench mark equipment as defined in 3GPP standard, the results can still show the limitations or issues in the RF performance especially in case of limeSDR.

The test was done in SISO mode with a maximum data rate achieved is around 62.3 Mbps. But in today's modern world when we talk about hundreds of Mbps, the technology could be challenging to implement on an SDR based system. It happens because all the signal processing tasks are being done on a central processing unit instead of a specialized digital signal processing module. Host computers with higher processing resources will be required to full fill the requirement. Another important aspect is latency, although currently the issue is out of scope of this project but still for upcoming scenarios in LTE like V2X communication or autonomous driving could be challenging because of the processing and data transfer delays from host to the SDR platform. However, The use of graphics processing units and technologies like Radio Frequency Network on Chip (RF-NoC) can be a potential solution to these challenges. Lastly for commercial implementation cost is an important factor which needs to be kept in consideration. The cost of SDR based LTE UE is very high as compared to the commercially available LTE UE devices.

6.2 Future Work

The evaluation in terms of data rate, RF characteristics and sensitivity of the developed LTE UE needs to be continued in future. Sections below describes the tests or related work to be done in future.

6.2.1 Data rate tests in Different Frequency bands and Using MIMO configurations

The SDR based Cat 4 UE is tested for FDD SISO configurations. Whereas, srsUE application suite [18] also offers the MIMO and TDD support upto UE Cat 6 specifications in the latest release. So further testing should be performed to know how the UE behaves with these settings. Further more, testing should be conducted in different LTE bands to know the exact limitations of the system.

6.2.2 CAT 4 LTE modem test with USRP based eNB

To draw a comparison of the performance of the SDR based UE with a standard Cat 4 LTE modem, a test setup can be implemented by connecting a standard Cat 4 LTE modem with srseNB. The setup is shown in figure 6.1.

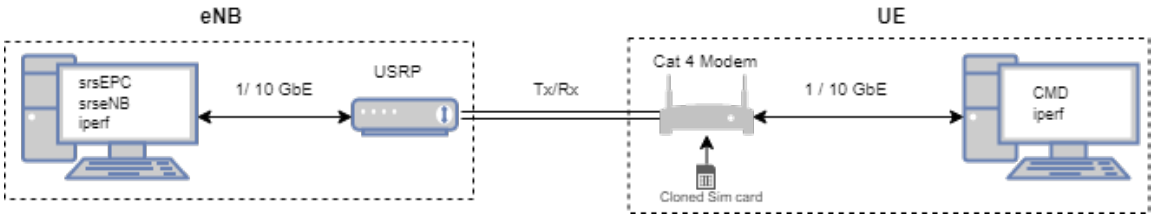


Figure 6.1: Cat 4 modem test setup

An empty sim card needs to be cloned with the parameters according to parameters stored in "user_db.csv" file of srsepc or vice versa. One can connect the Cat 4 modem to a host PC to run the iperf for data rate calculation.

6.2.3 RF Conformance tests

Detail radiation and reception performance of the SDR based UE should be performed. Relevant tests are to be chosen which are applicable to a particular UE that is intended to support the particular functionality or functionalities [5]. A list of all the applicable tests should be made along with the equipment required by referring the relevant 3GPP documents TS 36.521-1 [5] and TR 37.977 [10]. Tests like anechoic and reverberation chamber test provide the opportunity to test the UE by simulating different multi-path fading channels and get to know the performance of a UE in a real scenario [10].

References

[1] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Technical specification*. Technical Specification (TS) 36.300. Version 15.6.0. 3rd Generation Partnership Project (3GPP). URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2430.

[2] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification*. Technical Specification (TS) 36.321. Version 15.6.0. 3rd Generation Partnership Project (3GPP). URL: https://www.3gpp.org/ftp/Specs/archive/36_series/36.321/.

[3] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*. Technical Specification (TS) 36.213. 3rd Generation Partnership Project (3GPP). URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2427.

[4] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification*. Technical Specification (TS) 36.331. 3rd Generation Partnership Project (3GPP). URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2440.

[5] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) conformance specification; Radio transmission and reception; Part 1: Conformance testing*. Technical Specification (TS) 36.521-1. 3rd Generation Partnership Project (3GPP). URL: https://www.3gpp.org/ftp/Specs/archive/36_series/36.521-1/.

[6] 3GPP. *LTE ue-Category*. http://www.3gpp.org/keywords-acronyms/1612-ue-category, Last accessed on 2019-03-08. 2016.

[7] 3GPP. *Technical Specification Group Core Network and Terminals; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS);* Technical Specification (TS) 24.301. 3rd Generation Partnership Project (3GPP). URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=1072.

[8] 3GPP. *The Evolved Packet Core*. https://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core, Last accessed on 2019-07-21. 2017.

[9] 3GPP. *The Mobile Broadband Standard*. http://www.3gpp.org/technologies/keywords-acronyms/98-lte, Last accessed on 2019-02-21. 2017.

[10] 3GPP. *Universal Terrestrial Radio Access (UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRA); Verification of radiated multi-antenna reception performance of User Equipment (UE)*. Technical Reference (TR) 37.977. 3rd Generation Partnership Project (3GPP). URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2637.

[11] Amarisoft. *eNodeB*. https://www.amarisoft.com/technology/enodeb/, Last accessed on 2019-08-22. 2019.

[12] Fabian Eckermann, Philipp Gorczak, and Christian Wietfeld. "tinyLTE: Lightweight, Ad Hoc Deployable Cellular Network for Vehicular Communication". In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. IEEE. 2018, pp. 1–5.

[13] Feron Technologies P.C. *LTE-Sidelink*. http://www.feron-tech.com/products/, Last accessed on 2019-08-20. 2016.

[14] Zhiming Geng, Xingguang Wei, Haitao Liu, Rongtao Xu, and Kan Zheng. "Performance analysis and comparison of GPP-based SDR systems". In: *2017 7th IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies (MAPE)*. IEEE. 2017, pp. 124–129.

[15] Ismael Gomez-Miguelez, Andres Garcia-Saavedra, Paul D Sutton, Pablo Serrano, Cristina Cano, and Doug J Leith. "srsLTE: an open-source platform for LTE evolution and experimentation". In: *Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization*. ACM. 2016, pp. 25–32.

[16] Francesco Gringoli, Paul Patras, Carlos Donato, Pablo Serrano, and Yan Grunenberger. "Performance assessment of open software platforms for 5G prototyping". In: *IEEE Wireless Communications* 25.5 (2018), pp. 10–15.

[17] *iPerf - The ultimate speed test tool for TCP, UDP and SCTP*. https://iperf.fr/, Last accessed on 2019-08-22. 2018.

[18] Ismael Gomez-Miguelez, Andres Garcia-Saavedra, Paul D. Sutton, Pablo Serrano, Cristina Cano and Douglas J. Leith. *srsLTE*. https://github.com/srsLTE/srsLTE, Last accessed on 2019-08-20. 2019.

[19] Rahul Krishnan, R Ganesh Babu, S Kaviya, N Pragadeesh Kumar, C Rahul, and S Santhana Raman. "Software defined radio (SDR) foundations, technology tradeoffs: A survey". In: *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*. IEEE. 2017, pp. 2677–2682.

[20] Ludovic Rousseau. *PCSC lite project*. https://pcsclite.apdu.fr/, Last accessed on 2019-08-22. 2018.

[21] Nikos Makris, Agorastos Dimitrios Samaras, Virgilios Passas, Thanasis Korakis, and Leandros Tassiulas. "Measuring LTE and WiFi coexistence in Unlicensed spectrum". In: *2017 European Conference on Networks and Communications (EuCNC)*. IEEE. 2017, pp. 1–6.

[22] myriadrf. *Lime Suite*. https://wiki.myriadrf.org/Lime_Suite, Last accessed on 2019-08-22. 2019.

[23] Navid Nikaein, Mahesh K Marina, Saravana Manickam, Alex Dawson, Raymond Knopp, and Christian Bonnet. "OpenAirInterface: A flexible platform for 5G research". In: *ACM SIGCOMM Computer Communication Review* 44.5 (2014), pp. 33–38.

[24] Open Air Interface. *Towards Open Cellular Ecosystem*. https://www.openairinterface.org/?page_id=864#introduction, Last accessed on 2019-08-20. 2019.

[25] Suresh Paudel. "Investigation, Analysis and Implementation of Open Source Mobile Communication Software". MA thesis. NTNU, 2016.

[26] Pothosware. *Welcome to the SoapySDR project*. https://github.com/pothosware/SoapySDR/wiki, Last accessed on 2019-08-22. 2019.

[27] André Puschmann, Paul Sutton, and Ismael Gomez. "Implementing nb-iot in software-experiences using the srslte library". In: *arXiv preprint arXiv:1705.03529* (2017).

[28] Anders Charly Rasmussen and Mathias RÃ¸nholt Kielgast. "Embedded Massive MTC Device Emulator for lte using Software Defined Radios". MA thesis. Denmark: Aalborg University, 17.

[29] Ettus Research. *USRP Hardware Driver and USRP Manual*. 2019. URL: http://files.ettus.com/manual/page_build_guide.html.

[30] Devarpita Sinha, Anish Kumar Verma, and Sanjay Kumar. "Software defined radio: Operation, challenges and possible solutions". In: *2016 10th International Conference on Intelligent Systems and Control (ISCO)*. IEEE. 2016, pp. 1–5.

[31] Software Radio Systems. *SRSLte1*. https://www.softwareradiosystems.com/products/#srslte, Last accessed on 2019-08-20. 2019.

[32] Leena A Sonkusare and Sudhir N Dhage. "Analysis of LTE UE RF parameters for 3GPP specification". In: *2015 International Conference on Computers, Communications, and Systems (ICCCS)*. IEEE. 2015, pp. 82–86.

[33] Ben Wojtowicz and Dennis M Senyonjo. *OpenLTE*. URL: https://sourceforge.net/projects/openlte/.

[34] Nathan H Yee. "Performing a Practical Paging Attack on the LTE Network". In: (2017).

Appendix A

Installation Steps

Before moving on with the installations steps, it is necessary to know that these softwares require some supporting libraries like cmake, libfftw, PolarSSL/mbedTLS, Boost, lksctp, git, python etc which needs to be installed before installing the main sofware.

A.1 UHD Installation

Installation steps for UHD driver, adopted from [29], section "Building and Installing UHD from source". First, On Ubuntu 18.04, one can install the required supporting libraries using following command:

sudo apt-get install libboost-all-dev libusb-1.0-0-dev python-mako doxygen python-docutils cmake build-essential

git clone https://github.com/EttusResearch/uhd.git
Cd uhd
git checkout UHD-3.9.LTS
Cd host
Mkdir build
Cd build
Cmake ..
Make
Make test
Sudo make install
Sudo ldconfig

A.2 Updating USRP image

Download images:
sudo uhd_images_downloader
Connect the USRP to the host PC and run:
uhd_image_loader –args="type=x300,addr=¡IP address of USRP¿,fpga=HG"
Power-cycle the USRP once the image is updated.
https://files.ettus.com/manual/page_usrp_x3x0.html

A.3 LimeSuite Installation

First install required supporting libraries:

sudo apt install libsoapysdr-dev libi2c-dev libusb-1.0-0-dev git g++ cmake libsqlite3-dev libwxgtk3.0-dev freeglut3-dev

git clone https://github.com/myriadrf/LimeSuite.git
cd LimeSuite
git checkout stable
git pull
cd builddir
make clean
cmake ../
make
sudo make install
sudo ldconfig
cd ../udev-rules
sudo bash install.sh

A.4 SoapySDR Installation

Supporting libraries installation:

sudo apt-get install cmake g++ libpython-dev python-numpy swig
git clone https://github.com/pothosware/SoapySDR.git
cd SoapySDR
mkdir build
cd build
cmake ..
make -j4
sudo make install
sudo ldconfig
SoapySDRUtil –info

A.5 srsLTE Build and Install

Supporting libraries installation:

sudo apt-get install cmake libfftw3-dev libmbedtls-dev libboost-program-options-dev libconfig++-dev libsctp-dev
git clone https://github.com/srsLTE/srsLTE.git

cd srsLTE
git checkout release_19_03 ....As the project is done using this version
mkdir build
cd build
cmake ../
make
make test
sudo make install srslte_install_configs.sh user
.. Install the UE config in the home dir srslte_install_configs.sh service
.. Install the epc and enb config in the user dir

Appendix B

Below is the comparison list of the available SDR platforms which support LTE frequencies and were considered before finalizing the hardware used.

Table B.1: **SDR Supporting LTE Frequencies**

	RTL-SDR	**LimeSDR**	**UmTRX 2.3.1**	**USRP B200 mini**	**ASR-2300**
Frequency Range	500 Khz to 1766MHz	100kHz - 3.8 GHz	300MHz to 3.8GHz	70 MHz - 6 GHz	300 MHz to 3.8 GHz
RF Bandwidth	2.4 MHz	61.44 MHz	1 MHz to 28 MHz	56 MHz	28 MHz
DAC/ADC bits	ADC : 8 bits	12	12	12	12
Sample Rate	2.4 MHz	61.44 MSPS	40 MSPS	64.11 MSPS	-
Transmitter Channels	Nil	2	2	1	2
Receiver Channels	1	2	2	1	2
Full / Half Duplex	Half Duplex	Full Duplex	Full Duplex	Full Duplex	Full Duplex
Interface	USB 2.0	USB 3.0	GbE,RS-232 debug port	USB 3.0	SuperSpeed USB 3, or USB 2.0
Chipset	R820T	LMS7002M	LMS6002D	AD9364	LMS6002D
FPGA	Nil	Cyclone IV EP4CE40F23C8N	Spartan 6 LX75 FPGA	Xilinx Spartan-6 XC6SLX75	Xilinx Spartan 6
Processor	Nil	Nil	Nil	Nil	Nil
Open Source	Complete	Complete	-	Schematic, Firmware	complete
TDD/ FDD Supported	No	Yes	Yes	Yes	Yes
Oscillator Precision	± 2 ppm	± 1 ppm	± 100 ppb	± 2 ppm	± 1 ppm
Transmit Power	Receive only	10 dBm	20dBm	>10 dBm	6 dBm
Power Required	270-280 mA, 4.5V	USB or external power supply.	8-36V DC input	USB power 5V	5Volts(USB)
Price ($)	25	299	1300	810 (board only) $827 (Incl. cover)	

	AD-FMCOMMS4-EBZ	bladeRF 2.0 micro	Crimson TNG	ADALM-PLUTO	HackRF One
Frequency Range	70 MHz - 6 GHz	47 MHz - 6 GHz	DC âFIXME“ 6.0 GHz	325 MHz to 3.8 GHz	1 MHz to 6 GHz
RF Bandwidth	¡200 kHz to 56 MHz	56 MHz	325 MHz	20 MHz	20 MHz
DAC/ADC bits	12	12	16	12	8
Sample Rate	61.44 MSPS	61.44 MSPS	325 MSPS DAC, 325 MSPS ADC	61.44 MSPS	8 - 20 MSPS
Transmitter Channels	1	2	4	1	1
Receiver Channels	1	2	4	1	1
Full / Half Duplex	Full Duplex	Full Duplex	Full Duplex	Full Duplex	Half Duplex
Interface	USB or GbE.	USB 3.0 SuperSpeed	Dual 1/10G SFP+1GbE and USB	USB 2.0	High Speed USB 2.0
Chipset	AD9364	AD9361	ADF4355	AD9363	RFFC5071/5072
FPGA	Nil	Altera 49KLE Cyclone V	Altera Arria V (5ASTMD3E3F31I3N)	XC7Z010-1CLG225C4334	Nil
Processor	Nil	200 MHz ARM926EJ-S	dual-core ARM Cortex-A9	Nil	Nil
Open Source	Schematic, Firmware	Schematic, Firmware	Schematic, Firmware	Schematic, Firmware	Complete
TDD/ FDD Supported	Yes	Yes	Yes	Yes	Not sure
Oscillator Precision	± 15.0 ppm	± 5 ppm	± 10ppb	± 25 ppm	± 20 ppm
Transmit Power	6.5 dBm (Max output power of AD 9364)	6.5 dBm (Max output power of AD 9361)	Low -30 - 18 dBm High -10 - 15 dBm	7 dBm	15 dBm at 2150 MHz to 2750 MHz
Power Required	From FPGA board	DC 5V (1.5-3 A)	IEC320 C14 Power	DC Input (USB) 4.5V to 5.5V	USB 2.0 bus power
Price	399	480	13500	149	244.95

	Matchstiq S12	**USRP N200 and N210**	**USRP X300 and X310**
Frequency Range	¡1 MHz to 6GHz	DC - 6 GHz	DC - 6 GHz
RF Bandwidth	56 MHz	28 MHz	50 MHz
DAC/ADC bits	12	14 bit ADC 16 bit DAC	14 bit ADC 16 bit DAC
Sample Rate	200 KSPS to 61.44 MSPS	50 MSPS	200 MSPS
Transmitter Channels	1	1	2
Receiver Channels	1	1	4
Full / Half Duplex		Full Duplex	Full Duplex
Interface	GbE, USB 2.0 OTG, and HDMI	GbE	1 and 10 GbE
Chipset	-	Compatible Daughterboards	Compatible Daughterboards
FPGA	Xilinx Spartan 6 LX45T FPGA	Spartan 3A-DSP 1800/3400 FPGA	USRP X300 - XC7K325T USRP X310 âFIXME“ XC7K410T
Processor	Quad-core ARM Cortex A9 CPU @ 800 MHz	Nil	Nil
Open Source	-	Schematic, Firmware	Schematic, Firmware
TDD/ FDD Supported	-	Yes	Yes
Oscillator Precision	± 1 ppm	± 2.5 ppm	± 2.5 ppm
Transmit Power	TBD	15 dBm	¿10 dBm
Power Required	-	DC 6 V, 2.3 A	DC Input 12 V (45 W)
Price	4500	N200: $1715 N210: $1,943	X300: $4,403 X310: $5,422

Appendix C

Throughput calculation LTE

The theoretical LTE throughput was calculated using the following steps:

1. Pick number of resource blocks for specific bandwidth as shown in table 5.1.
2. Select the TBS value against the selected MCS value using table "Table 7.1.7.1-1: Modulation and TBS index table for PDSCH" in TS 36.213 [3].
3. Next refer to "Table 7.1.7.2.1-1: Transport block size table" in TS 36.213 [3] to determine how many bits can be transmitted in a subframe as subframe duration is 1ms so multiply it by 1000 to get bps.

Below table shows the theoretical max throughput for LTE. Calculated using above steps.

MCS	TBS	PRB-6 (Mbps)	PRB-15 (Mbps)	PRB-25 (Mbps)	PRB-50 (Mbps)	PRB 75 (Mbps)	PRB-100 (Mbps)
0	0	0.152	0.392	0.68	1.384	2.088	2.792
1	1	0.208	0.52	0.904	1.8	2.728	3.624
2	2	0.256	0.648	1.096	2.216	3.368	4.584
3	3	0.328	0.872	1.416	2.856	4.392	5.736
4	4	0.408	1.064	1.8	3.624	5.352	7.224
5	5	0.504	1.32	2.216	4.392	6.712	8.76
6	6	0.6	1.544	2.6	5.16	7.736	10.296
7	7	0.712	1.8	3.112	6.2	9.144	12.216
8	8	0.808	2.088	3.496	6.968	10.68	14.112
9	9	0.936	2.344	4.008	7.992	11.832	15.84
10	9	0.936	2.344	4.008	7.992	11.832	15.84
11	10	1.032	2.664	4.392	8.76	12.96	17.568
12	11	1.192	2.984	4.968	9.912	15.264	19.848
13	12	1.352	3.368	5.736	11.448	16.992	22.92
14	13	1.544	3.88	6.456	12.96	19.08	25.456
15	14	1.736	4.264	7.224	14.112	21.384	28.336
16	15	1.8	4.584	7.736	15.264	22.92	30.576
17	15	1.8	4.584	7.736	15.264	22.92	30.576
18	16	1.928	4.968	7.992	16.416	24.496	32.856
19	17	2.152	5.352	9.144	18.336	27.376	36.696
20	18	2.344	5.992	9.912	19.848	29.296	39.232
21	19	2.6	6.456	10.68	21.384	32.856	43.816
22	20	2.792	6.968	11.448	22.92	35.16	46.888
23	21	2.984	7.48	12.576	25.456	37.888	51.024
24	22	3.24	7.992	13.536	27.376	40.576	55.056
25	23	3.496	8.504	14.112	28.336	43.816	57.336
26	24	3.624	9.144	15.264	30.576	45.352	61.664
27	25	3.752	9.528	15.84	31.704	46.888	63.776
28	26	4.392	11.064	18.336	36.696	55.056	75.376

TRITA-EECS-EX-2019:841