Czech Technical University in Prague

Faculty of Transportation Sciences

Department of Transport Telematics

LDACS Test Platform Proposal

Master's thesis

Study program: Technology in Transportation and Telecommunications

Study field: Air Traffic Control and Management

Study project: Sattelite Navigation and Practical Use in Transportation

Supervisors: Ing. Petr Bureš Ph.D, Ing. Tereza Topková

Bc. David Levin

Prague 2022

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta dopravní
děkan
Konviktská 20, 110 00 Praha 1

K620..... Ústav dopravní telematiky

ZADÁNÍ DIPLOMOVÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

Bc. David Levin

Studijní program (obor/specializace) studenta:

navazující magisterský – PL – Provoz a řízení letecké dopravy

Název tématu (česky): **Návrh testovací platformy systému LDACS**

Název tématu (anglicky): LDACS Test Platform Proposal

Zásady pro vypracování

Při zpracování diplomové práce se řiďte následujícími pokyny:

- Cílem práce je navrhnout testovací platformu pro systém LDACS, včetně HW a SW a testovacích scénářů a zhodnotit jeho využití pro praxi. Za tímto účelem student zpracuje:
- podrobný popis využití, funkce a technických parametrů systému LDACS
- analýzu možných HW a SW nástrojů pro simulaci systému LDACS
- návrh testovacího nástroje při začlenění SW+HW platforem včetně testovacích scénářů
- provedení testů systému na dostupném SW a případně HW
- zhodnocení využití navržené testovací platformy

ČVUT
FD

Rozsah grafických prací: Dle pokynů vedoucího práce

Rozsah průvodní zprávy: minimálně 55 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)

Seznam odborné literatury: SESAR2020 – PJ14-02-01 - LDACS A/G Specification (Edition 00.02.02, 2019),

OFDM/OFDMA Audio Visual Tutorial for WiFi and WiMAX od Silicon DSP Corporation,

Y. Cho, J. Kim, W. Young Yang, C. G. Kang - MIMO-OFDM WIRELESS COMMUNICATIONS WITH MATLAB.

Vedoucí diplomové práce: **Ing. Petr Bureš Ph.D.**

Ing. Tereza Topková

Datum zadání diplomové práce: **15. července 2021**
(datum prvního zadání této práce, které musí být nejpozději 10 měsíců před datem prvního předpokládaného odevzdání této práce vyplývajícího ze standardní doby studia)

Datum odevzdání diplomové práce: **30. listopadu 2022**
a) datum prvního předpokládaného odevzdání práce vyplývající ze standardní doby studia a z doporučeného časového plánu studia
b) v případě odkladu odevzdání práce následující datum odevzdání práce vyplývající z doporučeného časového plánu studia

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE Fakulta dopravní

L. S.

Ing. Zuzana Bělinová, Ph.D.
vedoucí
Ústavu dopravní telematiky

prof. Ing. Ondřej Přibyl, Ph.D.
děkan fakulty

Potvrzuji převzetí zadání diplomové práce.

Bc. David Levin
jméno a podpis studenta

V Praze dne 16. května 2022

Declaration

I hereby submit for the evaluation and defence the master's thesis elaborated at the CTU in Prague, Faculty of Transportation Sciences.

I have no relevant reason against using this schoolwork in the sense of § 60 of Act No. 121/2000 Coll. on Copyright and Rights Related to Copyright and on Amendment to Certain Acts (the Copyright Act)

I declare I have accomplished my final thesis by myself and I have named all the sources used in accordance with the Guideline on ethical preparation of university final theses.

In Prague, November 2022

Bc. David Levin

Acknowledgement

I would like to thank my supervisor Ing. Petr Bureš Ph.D. and Ing. Tereza Topková for their professional advice, patience and support during the writing of the thesis. Also, I would like to thank SigNav laboratory of ENAC and CNS laboratory of CTU in Prague, for professional consultations, advice and providing the equipment required for work developing. Last, but not least, I would like to thank my family, friends and my girlfriend for the valuable support and faith in my strength during thesis creation.

Abstract

The LDACS is the upcoming digital communication system in aviation, that is able to perform navigation and even surveillance service. This system by the time of writing of this thesis LDACS is under certification standardization process by ICAO. This master thesis deals with the process of designing the LDACS testing platform in order to support future testing and performance evaluations different conditions. This thesis contain the detailed description of the processes that are going on in LDACS transmitter and reciever, evaluation of the SW and HW solutions that can be used for signal generation and signal transmission. Moreover the LDACS testing platform proposed and tested using different testing scenarios.

Key words:

LDACS, wireless communications, MATLAB, USRP, ADALM-Pluto, simulation

Abstrakt

LDACS je budoucí digitální komunikační systém v letectví který dokáže zajistit poskytování jak navigačních tak i přehledových služeb. Tento systém je současně ve standardizačním procesu u Mezinárodní organizace pro civilní letectví. Práce obsahuje podrobný popis procesů probíhajících ve vysílači a přijímači systému LDACS a zabývá se hodnocením SW a HW řešení, využitelných pro generování a přenos signálů. A navíc se zaměřuje na návrh testovací platformy, která by umožňovala testovaní, další vývoj a hodnocení výkonnosti daného systému v různých podmínkách.

Klíčová slova:

LDACS, bezdrátová komunikace, MATLAB, USRP, ADALM-Pluto, simulace

Acronyms

A2A Air-to-Air

A2G Air-to-Ground

ACARS Aircraft Communications Addressing and Reporting System

AeroMACS Aeronautical Mobile Airport Communication System

ANSP Air Navigation Service Providers

APNT Alternative Positioning Navigation and Timing

APT Airport channel

ATC Air Traffic Control

ATM Air Traffic Management

ATS Air Traffic Service

AWGN Additive white Gaussian noise

CNS Communication Navigation Surveillance

DME Distance Measuring Equipment

DSB Double Side Band

DSP Digital Signal Processors

ENR En-Route channel

FAA Federal Aviation Administration

FCI Future Communication infrastructure

FFT Fast Fourier Transform

FPGA Field Programmable Gate Arrays

GNSS Global Navigation Satellite System

GPP General Purpose Processors

GPS Global Positioning System

HFDL High Frequency Data Link

ICAO International Civil Aviation Organization

ICNSS Integrated CNS and spectrum

LDACS L-band digital aeronautical communication system

Loran long-range navigation

OFDM Orthogonal Frequency Division Multiplex

OOP Object-Oriented Programming

SATCOM Satellite Communications

SDR Software Defined Radio

SESAR Single European Sky ATM Research

SSR Secondary surveillance radar

TMA Terminal channel

VHF Very high frequency

VoIP Voice over Internet protocol

Contents

Introduction

The first communication systems took place even before the WWII. Over the years great progress has been made in order to make commutation systems more efficient and reliable. Today, the aviation communication infrastructure still has some challenges to overcome. These include lack of capacity, low security level and congestion of frequency spectrum in aeronautical bands. Due to these limitations, aviation industry organizations are interested in developing and implementing a new modern communication system.

L-band digital aeronautical communication system (LDACS) is one of the systems considered by International Civil Aviation Organization (ICAO) as a successor of current voice and digital communication solutions. Its base technology is the same as in a well-tested solution that has been implemented in the civil communication sector for years now - LTE and Wi-Max. This base technology is called Orthogonal Frequency Division Multiplex (OFDM). It is known as a spectrum efficient and high speed solution for wireless communication.

Nevertheless, every system in civil aviation should pass the certification process. Especially when the base technology of the system has not been used in civil aviation. The certification process includes testing of the new system in different kinds of environment and under different conditions. The example of these conditions can be interference with other systems in L-band, for example with Distance Measuring Equipment (DME). Due to technological progress there is a possibility to test the system on a smaller scale before building a full size ground station and aircraft avionics. The accuracy of this kind of testing is very high and the expenses are noticeably lower. Therefore, this thesis is focused on providing technical description of the LDACS system and also describes the process of designing the LDACS testing platform.

The first chapter of this master thesis describes issues of the current CNS infrastructure, especially on Communication component of CNS, and basically answers the question if LDACS is the solution at least for some of this problems. The second chapter proposes a detailed description of LDACS on physical level describing the processes that are taking

place in the transmitter and receiver. Then, thesis contains the analysis of hardware and software solutions that are currently available on the market and that can be used to design the LDACS testing platform. Then, the platform's design itself is proposed. In the last part of the thesis some testing scenarios that take place in a real system use are proposed. These scenarios are tested on available hardware and software to achieve the results in the form of transmission performance indicators. Then, the evaluation of the proposed solution is made in the form of discussion.

Chapter 1

Issues of today's CNS infrastructure

This chapter aims to describe today's Communication Navigation Surveillance (CNS) services from the operational perspective. From this point of view, there are several issues that are either already limiting the provision of the CNS service or may limit it in the future, assuming the future traffic growth. As well as with the introduction of the CNS vulnerabilities and issues, this chapter contains the introduction to existing projects and propositions for the future CNS infrastructure:

- Future Communication infrastructure (FCI)
- Alternative Positioning Navigation and Timing (APNT)
- Integrated CNS and spectrum (ICNSS)

Then, the requirements of the Air Navigation Service Providers (ANSP), that they will have together with the future CNS systems, are introduced. At the end of this chapter the question if an LDACS is capable to solve these issues is answered.

1.1 Today's communication vulnerabilities and Future Communication Infrastructure

Nowadays, communication service could be divided into two groups: voice communication and data communication. Both of them are used for Air-to-Ground (A2G) and Air-to-Air (A2A) communications. Moreover, as already mentioned in the introduction to this chapter,

those systems have their own limitations and vulnerabilities that nowadays are critical to air traffic. These issues are described in the following subsections. [50]

1.1.1 Data and voice protection

In recent decades air traffic management communications have been transitioning from analog voice communications to digital data link. The most notable one is the digital Aircraft Communications Addressing and Reporting System (ACARS), which uses VDL, High Frequency Data Link (HFDL) and Satellite Communications (SATCOM) transmission paths, graphically represented in figure 1.1. This network was established in 70s and originally had no security functions. Later on, after some modernisation certain cybersecurity protection was added. Therefore, only a few papers related to this problematics pointed out the existence of data leakage, data fraud, entity camouflage and denial of service attacks and other security threats to the ACARS system. Therefore, there is still no solution regarding the cybersecurity boost.[68, 62]

"Many papers discussed advanced security schemes using cryptographic algorithms, but these advanced security schemes hardly work in ACARS system, unless the next generation air transportation system will replace the current system"[79]

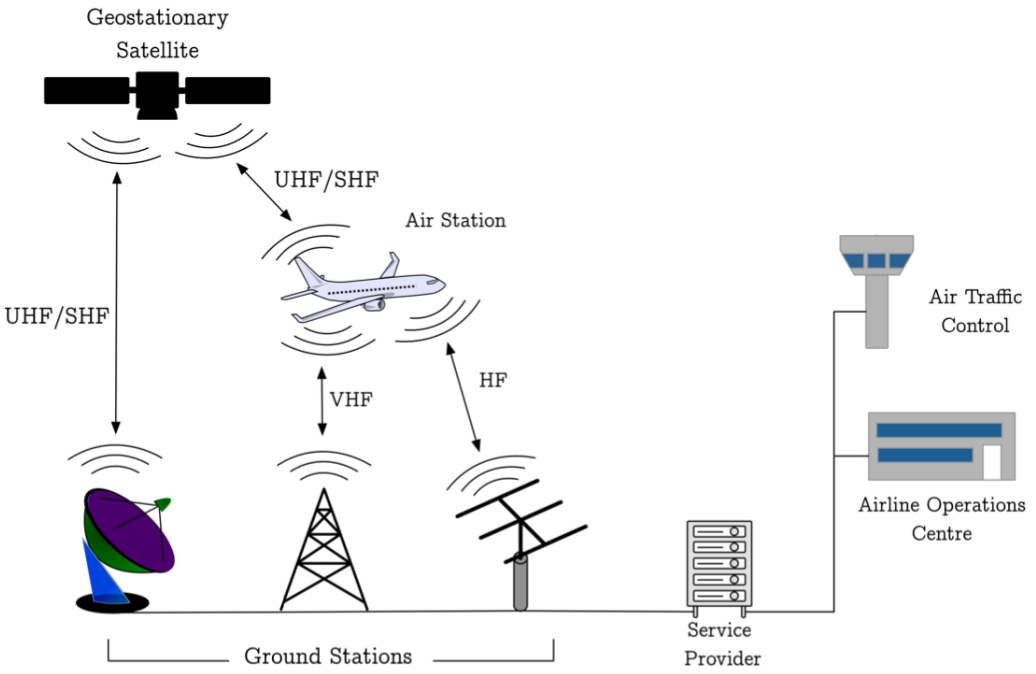

Figure 1.1: ACARS structure and transmission paths [68]

1.1.2 Inefficient usage of spectrum and lack of capacity

For terrestrial voice communication Very high frequency (VHF) radio in 118,000 MHz –136,975 MHz is implemented. It is based on Double Side Band (DSB) transmission with amplitude modulation. The main benefit of DSB is its resistance to interference. However, it is less spectrum efficient compared to single side band. After establishing voice communication services in 1947, parameters for transmission frequencies were defined as 118,000 MHz –136,975 MHz and channel separation as 200 kHz. This meant that there were only 70 channels available for transmission. Owing to technical progress at present the channel separation of 8.33 kHz is standardised and used in high traffic density areas such as the European region. 8.33 kHz channel separation theoretically gives the possibility to use 2280 channels for voice communication, but despite that there is still lack of available channels in some areas. [3, 2, 62, 50]

1.1.3 Future Communication Infrastructure

To overcome the limitations described in the subsections above, European project Single European Sky ATM Research (SESAR) is developing more sustainable technological basis for A2G and A2A communications, which is called FCI. For that purpose SESAR is considering

the Iris satellite link, Aeronautical Mobile Airport Communication System (AeroMACS) and L- band digital aeronautical communications system. These systems are intended for different use cases:

- Iris satellite link - enabling communications to aircraft in oceanic and remote areas.
- AeroMACS - enabling communications at large airports.
- LDACS - ensures long-haul terrestrial communications. [47]

The described concept is graphically represented in figure 1.2. [47]

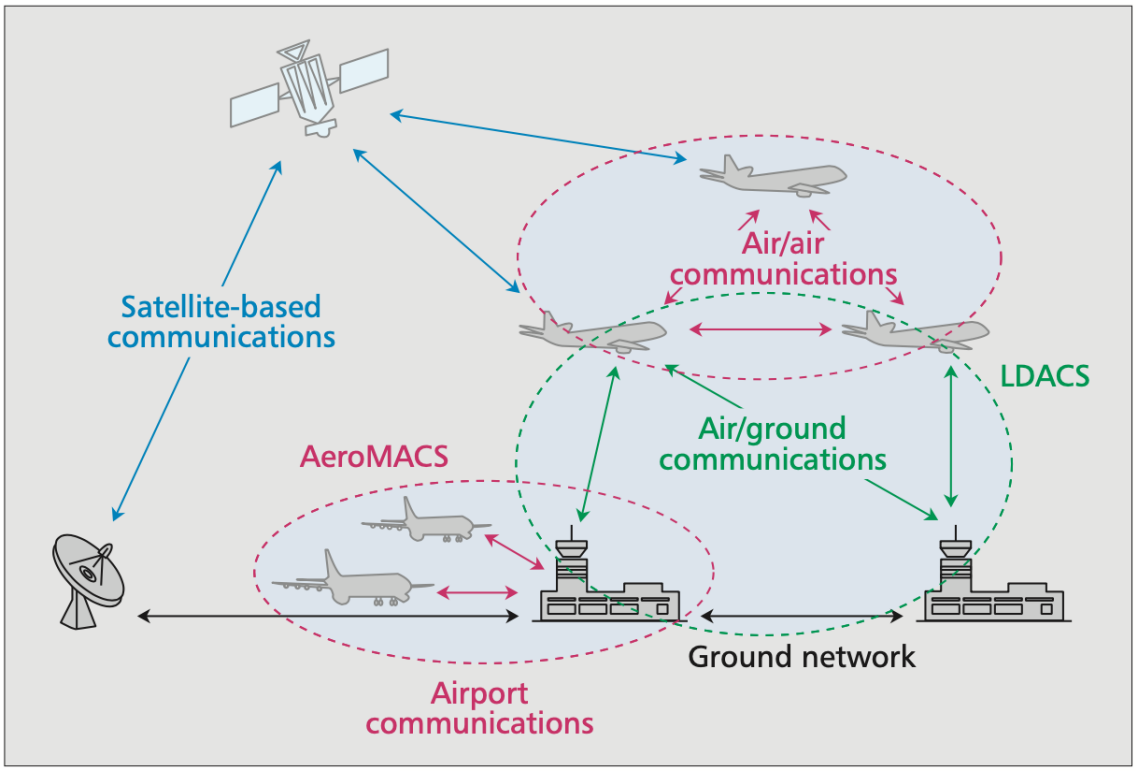

Figure 1.2: FCI concept structure and transmission paths between main components [62]

1.2 APNT for GNSS backup

At the moment, in most cases Global Navigation Satellite System (GNSS) is the primary source of aircraft position information. The GNSS architecture consists of three main segments: the space segment, which is represented by satellites, the user segment, which is represented by GNSS receivers, and the control segment, which is represented by control stations. The main function of the space segment is to generate and transmit position detection signals and to transmit a navigation message recorded by the control segment. The

GNSS space segments are made up of satellites orbiting the Earth in several orbits in an attempt to ensure that users have the necessary number of satellites in position at any time to determine their position. The control segment is responsible for the proper operation of GNSS. Its basic functions are: to manage and maintain the status and configuration of the satellite constellation, to predict ephemeris and satellite clock errors and to update navigation reports for all satellites. The user segment consists of GNSS receivers. Their main function is to receive GNSS signals, determine pseudorange and solve navigation equations in order to obtain coordinates and provide very accurate time. The principle of positioning using GNSS is based on the measurement of pseudorange, which differs from the classic distance by a time shift. By transmitting a navigation message stating their designation, position and time of transmission, the satellites can determine the distance between the satellite and the receiver, and more precisely the spherical surface on which the receiver can be located. After finding three pseudoranges, it is possible to calculate the intersection of these spherical surfaces on which the receiver is located. Another satellite is needed to detect the receiver clock error, which is the fourth unknown in the calculation. [41]

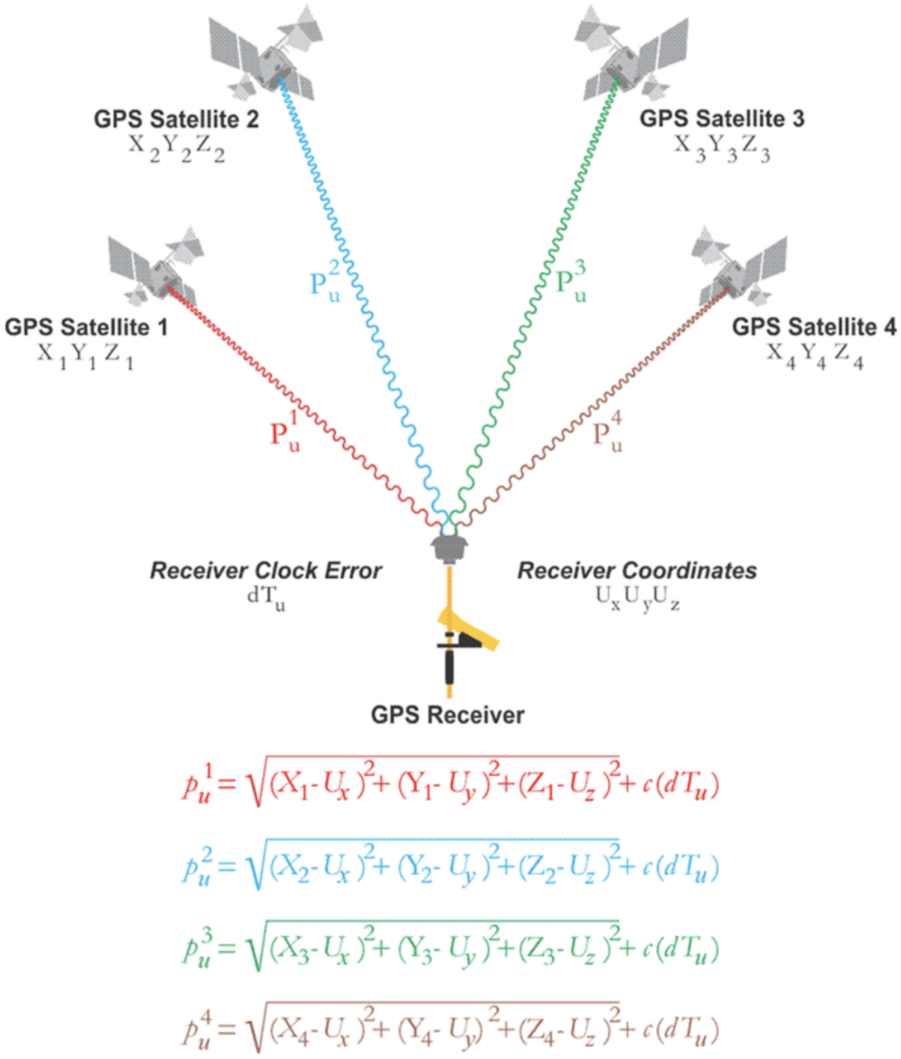

Figure 1.3: The principle of GNSS positioning [41]

The Global Positioning System (GPS) signal is a low power signal. Airbus defines the signal as being comparable to the energy emitted by a 60 W light bulb located more than 20,000 km from the Earth's surface. This means that the signal can be easily disturbed by any terrestrial source located near the receiver and operating in the GPS frequency band. Interference can be divided according to origin into non-anthropogenic and anthropogenic interference. Non-anthropogenic interference includes atmospheric effects, satellite clock errors, and multipath propagation, which results in reduced accuracy. There are a few types of anthropogenic GNSS interference. Jamming - is the most widespread type of interference, which causes impossibility of positioning. The other two main types of anthropogenic GNSS interference are spoofing and meaconing, which causes the same consequence - displaying a false position on the terminal device. Due to the vulnerability factors of the GNSS described

above, ICAO recommends that States actively support the development of a future APNT system. The aim of the APNT strategy should be to maintain the safety and, as far as possible, an acceptable level of air navigation services in the event of long-term GNSS outages or disruptions. In 2012, the ICAO Conference expressed the need to support alternative positioning, navigation and timing systems in order to keep air navigation services to a maximum in the event of a GNSS signal failure. In recent years, great progress has been made in the development of APNTs.

1.2.1 APNT consideration in EU

In the European region corporations such as Honeywell, DLR and Thales are developing systems that will be capable of meeting performance requirements defined by the SESAR Joint Undertaking. The project defined certain requirements for future APNTs and divided them into three categories: navigation support, ADS-B OUT support, and providing accurate time for on-board systems. Meanwhile, these requirements are defined for certain phases of the project: short, medium and long term. The subject of the project is not only to define the requirements, but also to assess the selected systems for their ability to support the set requirements. The European SESAR considers the following to be specific APNT systems: LDACS, DME/DME, Multi DME, eLoran[1]. These systems are intended for different implementation periods defined by Eurocontrol. For long term solution the use of new technologies, such as LDACS or eLoran are considered. In my bachelor thesis I worked on evaluation of APNT systems suggested for the European Union. The results of the assessment were similar except for the LDACS system, which has earned the lowest number of points. These results were influenced by the fact that the implementation process was included into the scoring, and since it is an entirely new system, it was disadvantaged by this evaluation. Nevertheless, LDACS has an important added value due to the provision of both communication and navigation services. So, in the future there are a lot of enhancement opportunities for this system. For instance, adding A2A communication and ranging or even surveillance services. The performance of the system is also very impressive, the precision reaches 20m, and the integrity is the same as GPS's integrity - $1 - 10^{-5}$. [41]

[1] Enchanced version of long-range navigation (Loran)

1.3 Integrated CNS and spectrum

Besides APNT activities the SESAR JU project has interest in integration of CNS services. According to the SESAR' s report to European Commission:
"Many aspects of CNS are currently characterised by lack of integration: multiple technologies are used where it could be possible to merge into one, both within and across domains [5] Improved integration would provide an opportunity to reach better safety, minimise operational costs and improve the efficiency of spectrum usage. The last aspect is critical for aviation, since most of the CNS systems are operating in one band (L-band). This band is defined by the radio spectrum from 1 GHz to 2 GHz. In principle, 960-1215 MHz is reserved to be used by civil and military, navigation and surveillance air services. Due to the quantity of systems already operating at those frequencies (shown in figure 1.4) it is difficult to implement new technologies and protect the existing ones at the same time. [27, 58]

The following figure 1.4 represents the L-band spectrum utilization (x axis represents the frequencies in MHz). This picture shows that besides the military systems (JTIDS/MIDS and RSBN), the greatest part is dedicated to DME [2]. DME can not be easily removed or substituted because of its importance and prevalence in the field of ground to air radio navigation. [32, 27, 58]

[2]DME is a radio navigation device that indicates the slant distance between an on-board interrogator and a ground transponder. DME works on the principle of transmitting and receiving electromagnetic pulses. These pulses are in the form of pulse pairs in the case of DME.

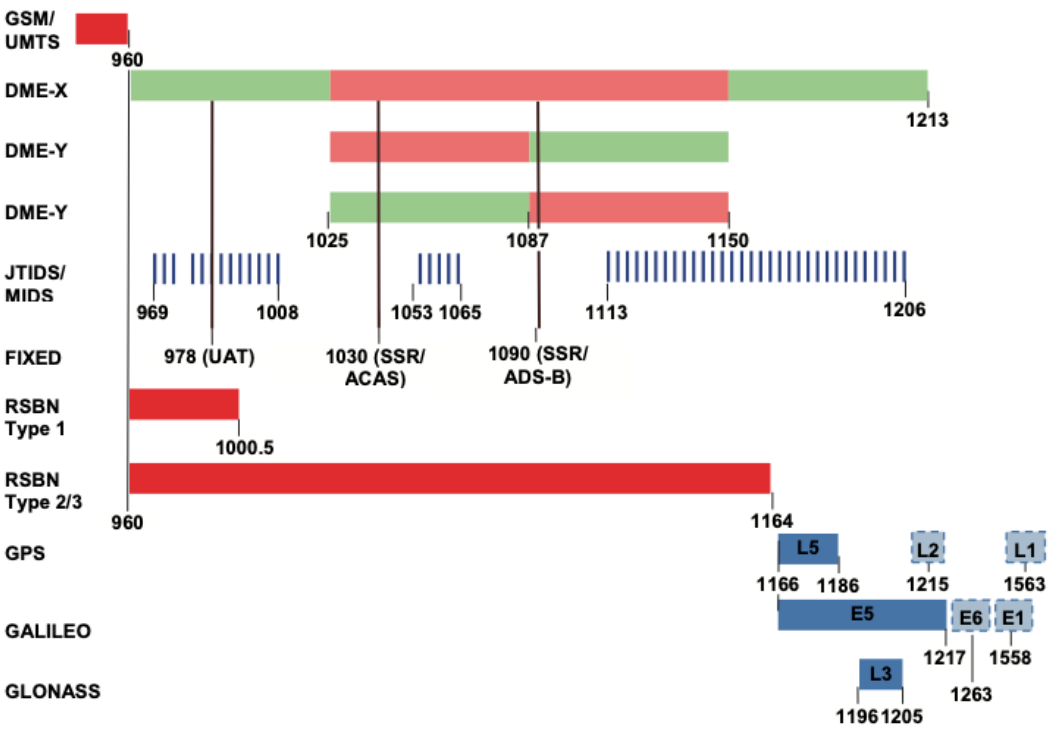

Figure 1.4: L-band spectrum utilisation [32]

1.4 Introduction to LDACS on operational level

In the previous sections LDACS has already been mentioned during the APNT and FCI project introduction. This section contains basic LDACS description on the operational level.

LDACS is the upcoming digital communication system, that is able to perform navigation service. It was developed by DLR (German Aerospace Center) and since this system is not based on any of the existing systems used in aviation, LDACS implementation might be a big step to the modernization of current ATC services.

LDACS operates on the basis of a network of ground stations operating at different frequencies. The communication between the ground station (GS) and the aircraft (A/C) is implemented through OFDM (Orthogonal Frequency Division Multiplexing) technology. LDACS adheres to the bidirectional transmission mode, where the band 985.5–1008.5 MHz is defined for the uplink and 1048.5–1071.5 MHz for the downlink with channel separation of 1 MHz and bandwidth of 500 kHz. This makes it possible to transmit the LDACS signal in-between DME pulse pairs, figure 1.5 With that feature LDACS is very compatible with the existing CNS infrastructure and creates fewer concerns about the interference with the

existing systems after the LDACS system is implemented. [41, 28]

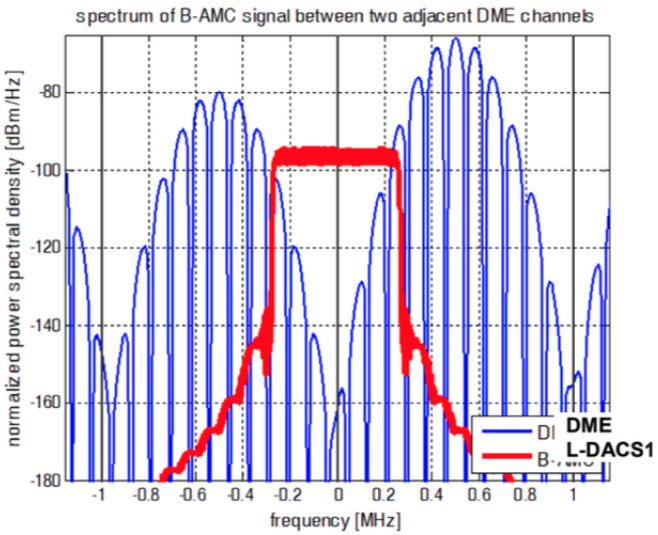

Figure 1.5: LDACS signal (in red) between DME pulse pairs (in blue). [71]

DME is an significant system in today's aeronautical navigation. So, interference issues with the future CNS systems is a crucial topic. Studies on interference between these two systems were carried out by the LDACS development team as well as by some independent researchers. The result of these studies shows that even though DME and LDACS affect each other, they still can safely share the same frequency range.This allows to overcome the issues of lack of available frequencies in L-band. Due to this fact, LDACS can be implemented without replacing any of the current systems. [71]

1.4.1 LDACS communication service

Owing to technologies used in LDACS, it is capable of achieving up to 260 times higher data rate per channel than modern communication systems such as VDL Mode 2. The benefits of the LDACS communication service deployment include: low-latency transmission through coordinated channel access, no co-channel interference, high continuity, and providing of both communication services - data link and digital voice. Along with the above-mentioned performance benefits, LDACS brings in secure transmission through built-in cyber security means such as mutual entity authentication, protection of confidentiality, integrity and authenticity of messages. And, last but not least, LDACS ensures availability and continuity of service which is critically important. [64, 65]

The LDACS Digital Voice communication technique is based on VoIP technology. The

main idea behind Voice over Internet protocol (VoIP) is transmission of voice traffic over the internet connection. On LDACS VoIP is deployed by first using wireless A2G connection and after that an ordinary VoIP sequence of actions (VoIP structure is described in figure 1.6.

Figure 1.6: Voice over Internet Protocol structure deployed on LDACS.[67]

Also the LDACS Digital Voice channels are independent from the physical LDACS deployment. This means that a LDACS ground station can serve several Air Traffic Control (ATC) voice sectors, and vice versa one ATC voice sector can use multiple LDACS ground stations for digital voice transmission. Furthermore, voice circuits can be completely independent of the current ATC voice sector. [67, 65]

1.4.2 LDACS further capabilities

LDACS is capable of overcoming the existing limitations. This system introduces other benefits and prepares ATM for modernisation according to the development vectors that were described at the beginning of this chapter: trajectory based operations, transition to broadband data link technologies and aircraft integration into SWIM (XML-based communications). For these purposes LDACS is ready to introduce a bunch of technologies and abilities, that can make Air Traffic Management (ATM) more robust, efficient and secure. To sum up the LDACS features and capabilities:

- LDACS is capable of covering all kinds of ATM communications: AOC, ATS, AAC.
- LDACS enables broadband applications such as 4D trajectories, weather maps, SWIM integration.
- LDACS contains ranging/navigation functionality.
- LDACS can be used as data link for GBAS (supplementing or replacing VDB) with benefits of more capacity and less latency.

- LDACS is extendable and future proof - it can be easily extended to higher capacity. [67]

With all that, LDACS is being under consideration as a primary future long-haul terrestrial communication system by SESAR JU and Federal Aviation Administration (FAA) NextGen. [41]

1.5 Requirements to the upcoming CNS systems

As a conclusion to this part, it has to be noted that LDACS fulfills all the requirements and even adds its own special features to the existing CNS infrastructure. On the other hand, the author sees a lot of complications that might occur once this system is implemented. The implementation process plan has to be very detailed and deal with special features of the Orthogonal Frequency Division Multiplex systems that include high sensibility to the Carrier Frequency Offset and frequency synchronization problems. Because of these drawbacks the system requires a long and diverse testing period. However, as soon as the system is tested and fully operable it will boost the possible Air Traffic Service (ATS) and ATC application variety that can be introduced. This includes the main vectors that were presented at the beginning of this chapter.[65, 66]

Apart from the main vectors of the CNS infrastructure and the development of applications, ANSP want the future systems to achieve certain acceptable performance in terms of:

- Performance itself - the service is provided with an acceptable quality, with certain navigation performance indicators (continuity, availability, accuracy and integrity).
- Capacity - the number of aircrafts this service can be provided for.
- Security - resistance against data loss
- Interoperability - requirement for compatibility with the existing and future infrastructure.
- Scalability - possibility of a future system extension without major technical changes
- Coast efficiency - the system should be compatible with business [67]

All those characteristics of the system are forming the safety aspect, which must match an appropriate level as well. LDACS matches all of these requirements. The only requirement that cannot be evaluated yet is coast efficiency (the system is not in service yet). [67]

Chapter 2

Orthogonal Frequency division Multiplexing

This chapter provides basic technical information about LDACS and describes OFDM in detail. OFDM is the basic technology of this system and a detailed description is required in order to understand the processes going on during the LDACS signal transmission and reception.

The first part of this chapter provides some basic information about the technical aspects of the LDACS system. Since LDACS uses the orthogonal frequency division multiplexing technique, the OFDM architecture is introduced in detail.

2.1 Introduction into OFDM architecture

The OFDM modulation technique is based on the multi-carrier modulation method (MCM), where despite the signal is transmitted within one carrier frequency and wide broadband, multiple sub-carriers with narrow broadband are used. This is achieved by dividing the broadband channel into a number of parallel narrow sub-channels. Each of those sub-channels has its own carrier frequency and carries a low data rate stream. Consequently, high data rate transmission is achieved by summing up those data rate streams. The difference between the signal with a single carrier structure and the OFDM sub-carrier structure is shown in figure 2.1. The multi-subcarrier structure is shown on the left, where N sub-carriers are placed with defined separation between the carriers across the same bandwidth, as single carrier signal on the right hand side of the picture. Besides high data rate transmission, there are also several advantages offered by this modulation technique. The most important

of them is resistance to channel dispersion and facilitation of phase and channel estimation in a time-varying environment. [40, 8]

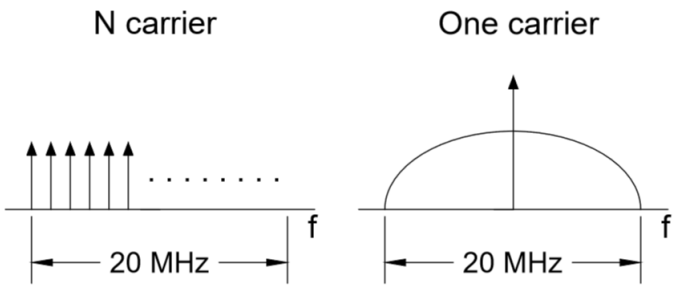

Figure 2.1: Difference between one carrier and MCM signal structure.[author]

As follows from the OFDM abbreviation, it uses the frequency division multiplexing technique to divide channels. However, unlike FDM, OFDM transmits the sub-carriers orthogonally, which means, that individual signals actually overlap. However, each sub-carrier has its maximum power at the point, where the other ones have zeros, figure 2.2. This feature does not cause any noticeable adjacent sub-carrier interference and makes it possible to use bandwidth more efficiently. [20]

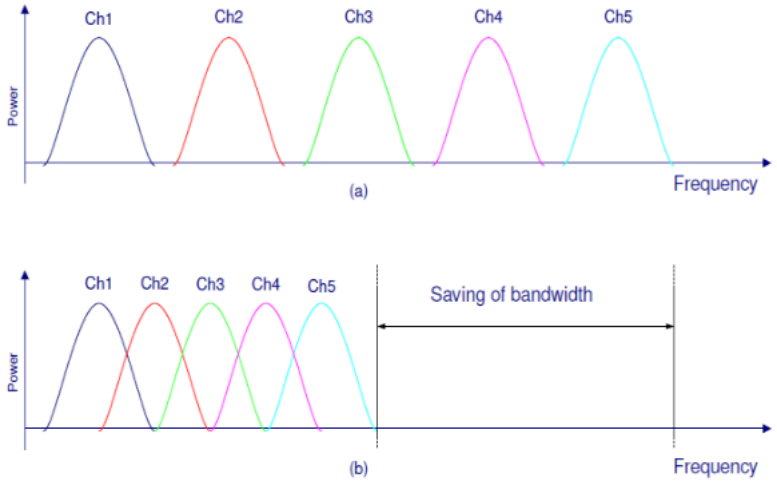

Figure 2.2: Orthogonality in OFDM technique compared to FDM technique [8]

In figure 2.2 the upper part (a) represents FDM, where the sub-carriers are placed one after another in a defined bandwidth. In the lower part (b) of figure 2.2 the sub-carriers are orthogonal to each other, so they occupy less bandwidth. [8, 20]

2.1.1 LDACS transmitter structure

This subsection introduces the transmitter of the LDACS system and describes the functionality of the transmitter structure. It outlines the signal definition process starting from raw binary data up to complex signal in time domain.

The main components of the OFDM-based transmitters are: data modulation which implies data transfer into the complex plane and Inverse Fast Fourier Transformation which converts the signal into the time domain. Nevertheless, in real OFDM systems, as well as in LDACS, some additional components are added. These might include synchronization algorithms, error correction or interleaving. This helps the OFDM communication systems become more resistant to real channel interference and add several features to those systems, for example Forward Error Correction (FEC) algorithms. In case of LDACS, the following procedures are added:

- Data Randomizer
- Reed-Solomon coding
- Block interleaver
- Convolutional coding
- Helical interleaver
- Windowing and cyclic postfix addition

Further on, the LDACS transmitter and receiver can be represented by the following scheme in figure 2.3. This scheme is further widely used as reference during the LDACS platform design proposition.

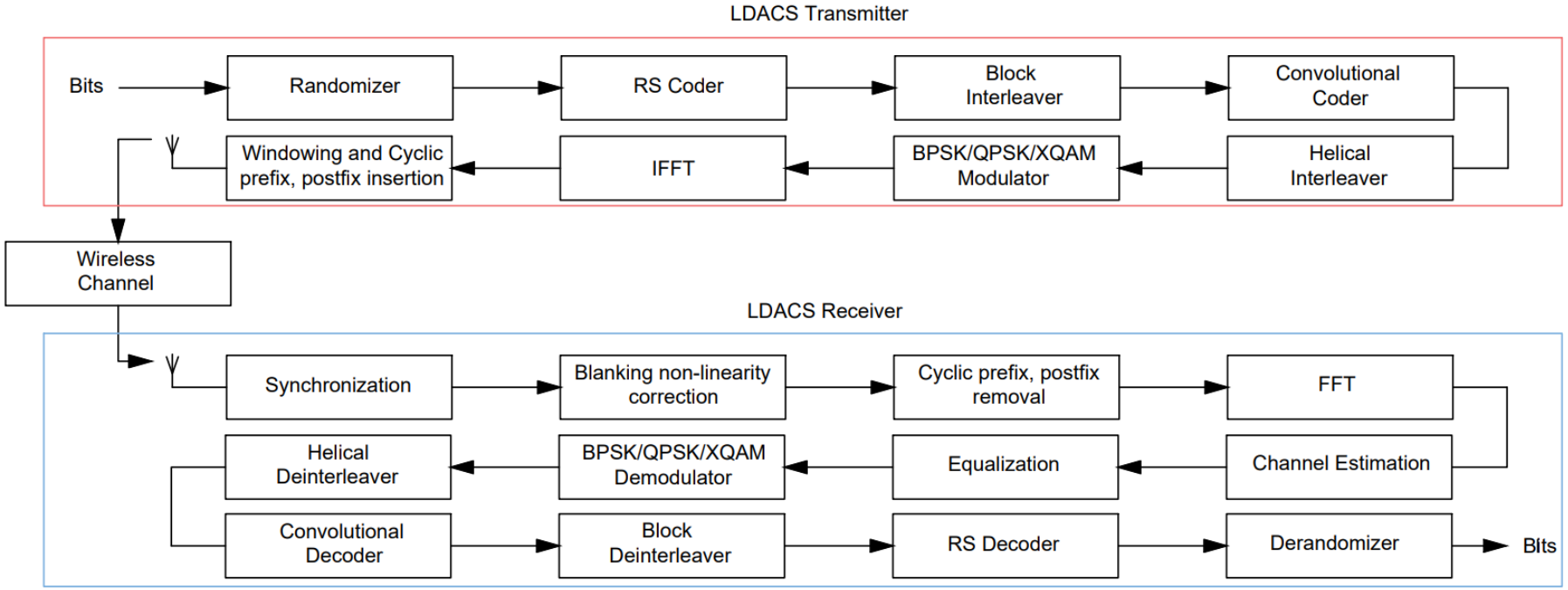

Figure 2.3: LDACS block scheme

In the scheme above, bits represent digital data for transmission, in case of the uplink - ground to aircraft message, in case of the downlink - aircraft to ground message. In the following parts individual components of this block scheme are described in detail.

2.1.2 Scrambler

Scrambling or randomizing of the data is a process of rearrangement of this data sequence in order to achieve better data safety and better synchronization efficiency. The main function of the scrambler is to get rid of a long string of ones and zeros from the binary data. There are different scrambling techniques. The following scrambler structure is proposed for LDACS (figure 2.4): [64, 63, 6]

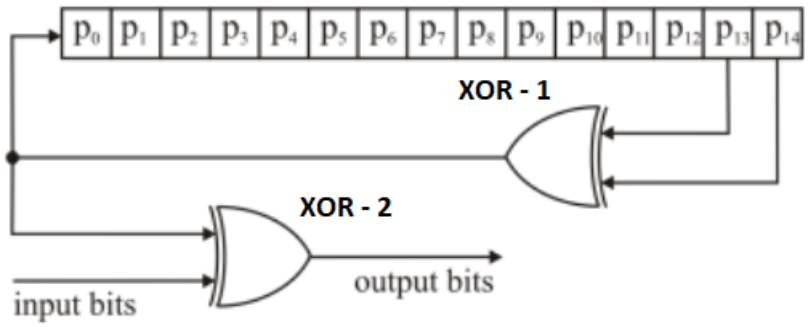

Figure 2.4: Scrambler structure for LDACS [64]

Where p_i has the following default values:

$$\{p_0, p_1, p_2, p_3 ..., p_{14}\} = \{0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1\} \tag{2.1}$$

The exact type of the scrambler is not defined in the LDACS documentation. Nevertheless, with high probability LDACS deals with the additive scrambler, where Linear Feedback Shift Register (LFSR) with length of 15 memory elements ($p_0 ... p_{14}$) and two XOR[1] operations are used. [64, 49]

In this sequence, XOR of the latter two memory elements are calculated and sent into two ways. The first way is the XOR - 2 with the input bit to obtain the output bit value. The second way (through XOR-1) is used for the purpose of registration of the shift. When the register shifts, the value of the first XOR operation becomes p_0 and all the bit values are shifted left. p_{13} becomes p_{14}, and original p_{14} is deleted. [64, 49]

[1] XOR - exclusive or - logical operation, takes two boolean operands and returns true if, and only if, the operands are different

This process is repeated until the whole data unit is scrambled, for the next data unit LFSR is set up with default $p_0 ... p_{14}$ values (eq: 2.1). On the receiver side the backward operation is performed to obtain the data stream. [64, 49, 6].

2.1.3 Reed-Solomon Coding

Reed-Solomon coding (RS coding) is part of the error correction algorithm that is used in many digital communication systems. The error correcting code itself can be described as:

"An algorithm for expressing a sequence of numbers so that any errors which are introduced can be detected and corrected (within certain limitations) based on the remaining numbers." [44]

For the LDACS system two error correction algorithms are proposed - RS coding and Convolutional coding (described later in this chapter). [64, 63]

The RS coding works with the field of finite elements (q), called Galois field (GF). Without going into detail, the main goal of RS coding may be defined as forming a code-word $c(x)$ from the message and the generator function. This is done by multiplying the message in the form of corresponding polynomial $m(x)$ with the generator function polynomial. The generator function polynomial for LDACS is: [18]

$$g(x) = \prod_{i=1}^{2F} (x + \lambda^i) \tag{2.2}$$

Where λ is equal to $x + 0$, and F is the greatest integer less than or equal to $\frac{N-K}{2}$. In this case, N is the number of coded bytes, and K is the number of uncoded bytes. [64, 63]

Then, since the code-word is $c(x) = g(x) * m(x)$, the code-word after transmission is represented by:

$$v(x) = c(x) + e(x) \tag{2.3}$$

Where $e(x)$ is an error polynomial. [18]

The error correction is made by defining the zeros of $g(x)$, that can be denoted as γ_1, γ_2, γ_3 ... γ_r. Then, evaluating of the received polynomial with the roots of $g(x)$:

$$v(\gamma_i) = c(\gamma_i) + e(\gamma_i) \tag{2.4}$$

Provided that $c(x) = g(x) * m(x)$, since $g(x) = 0$, the expression above can be written as:

$$v(\gamma_i) = e(\gamma_i) \tag{2.5}$$

The error polynomial can be written out in terms of coefficients e_j and γ_i^j with $j = 1, ..., r$, eq. 2.6

$$v(\gamma_i) = \sum_{i=0}^{n-1} e_j \gamma_i^j \tag{2.6}$$

Then, if this set of equations can be solved for e_j, the error pattern can be determined, with the following error correction application. [18]

2.1.4 Block Interleaver

In general, interleaving is a process or methodology to make a system more efficient, fast and reliable by arranging data in a non-contiguous manner. Then, in case of a burst error, where several bits in a row are corrupted, the package is not lost and it can be recovered with error correction algorithms, considering that the most error correction algorithms are way more productive in case of random errors. The general purpose of the interleaver is visualized in figure 2.5. It can be seen, that in case of a raw data stream more data from one package are corrupted by burst error and the probability that this data can be recovered by the error correction is lower than in case of interleaved data, where corrupted data are spread across different data packages.[17, 39]

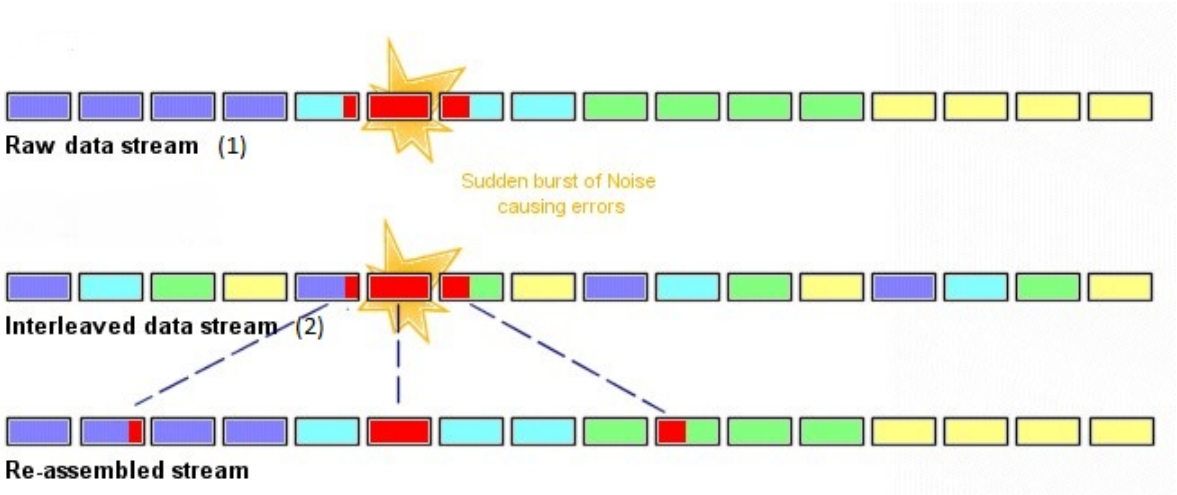

Figure 2.5: Interleaved vs raw data stream burst error consequences [39]

In LDACS two interleavers are considered - the block interleaver after the RS coding and the helical interleaver after convolutional coding. [64, 63]

The block interleaver works with a matrix, which in case of LDACS has a number of rows defined by the number of RS code-words and the number of columns defined by the number

of coded bytes per RS code-word. Bits are loaded into this matrix row-wise and read out column-wise. This ensures the rearranging of the data described above. [64, 63]

2.1.5 Convolutional Coder

The convolutional coder has the same goal as the previously described Reed-Solomon coding which is error correction. The convolutional coder (CC) has a rather similar structure to the scrambler. It has a certain number of LFSR, but instead of using the XOR logical operation, it uses modulo 2 operation. The main goal here is to form a predefined (known to the transmitter and the receiver) structure from the bit stream. The CC has one crucial indicator which is the code rate. The code rate defines how many output bits are produced with one bit on the input. The code rate (r_{cc}) of 1/2 means that for each input bit, two output parity bits are calculated. In fact, CC produces two or several bits for each input one which causes lower data rate transmission, but introduces error correction. Error correction is applied due to the known structure of the CC coder on the receiver side. Then, bits can be encoded at the receiver and error correction can be applied (the receiver knows all of the states the CC can be in). The parity bit is calculated with an algorithm of the CC for the system. The structure may vary from system to system, LDACS uses different CC structures for different types of transmission. [16, 64, 63]

- Native - for all except RL[2] RA+DC : CC with $r_{cc} = 1/2$ and 6 LFSR with a structure in figure 2.6 [64]

[2]For the description of the frame structure reffer to the "Subcarrier Allocation" subsection.

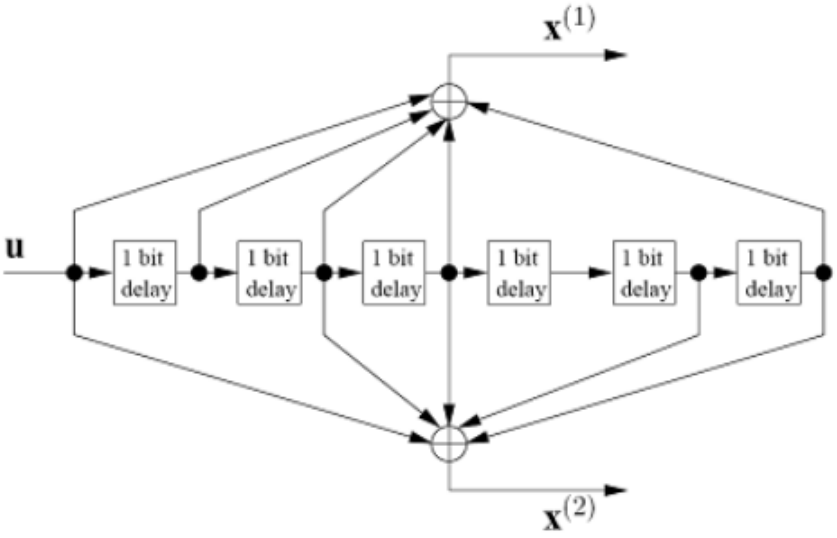

Figure 2.6: LDACS native convolutional coder structure [64]

- RL RA+DC [3]: CC with CC with $r_{cc} = 1/3$ and 6 LFSR with a structure in figure 2.7. [64]

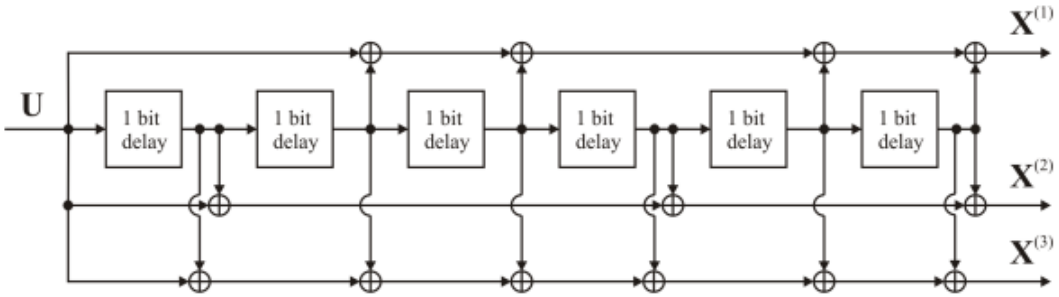

Figure 2.7: LDACS RL RA+DC convolutional coder structure [64]

2.1.6 Helical Interleaver

The last step before the coded data modulation is to interleave them one more time. This is performed by the Helical interleaver. The Helical interleaver aims to evenly spread the coded bits across the time-frequency plane. In the LDACS system the Helical interleaver has a matrix size of A x B, where A and B are linked to the number of coded bits. For example, for the FL transmission with CC $r_{cc} = 1/2$, 16QAM modulation and 6 data units per an interleaving block is 132x148. [64, 63]

[3] In this case, no RS encoding and block interleaving is performed.

Data are interleaved through the matrix with the following algorithm:

```
for l = 0:a-1
    for n = 0:b-1
       k=l*b+n+1
       m_k=b*(3*n+l)+n+1
    end
end
```

Modulation of data

For the purpose of QAM modulation, amplitude shift keying and phase shift keying are used, where bits are mapped into the complex plane using either QPSK, 16QAM or 64QAM constellation diagrams. These constellation diagrams are represented in figure 2.8 [70, 73, 74, 64]

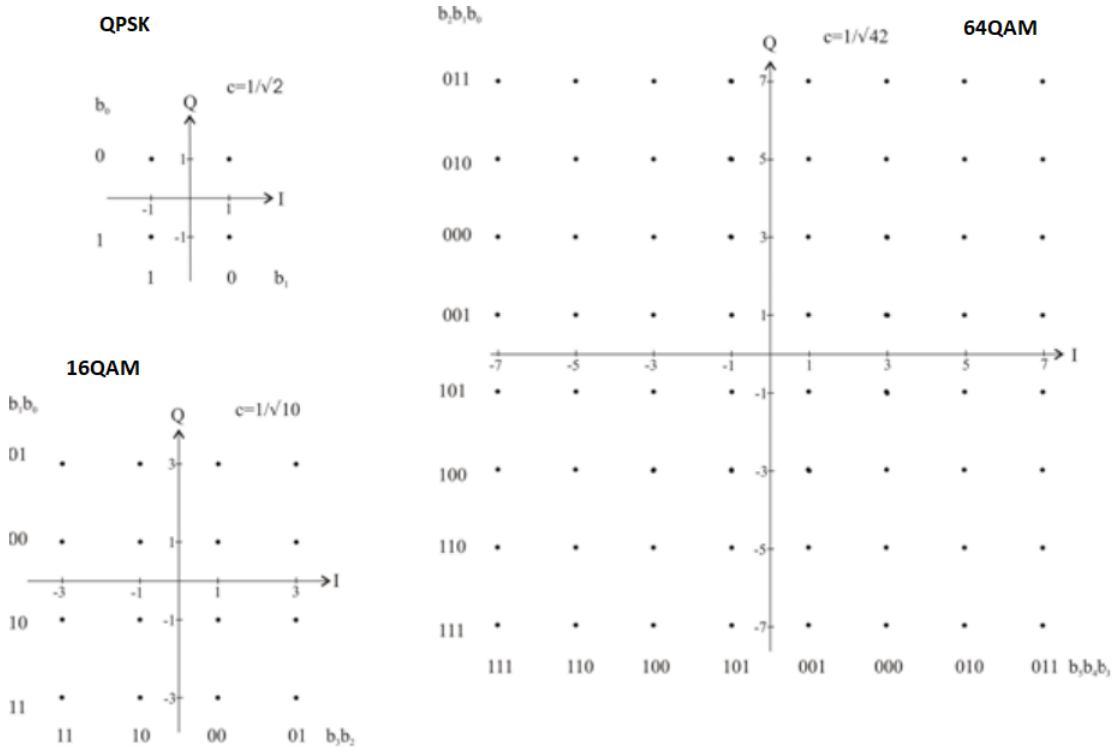

Figure 2.8: QPSK, 16QAM and 64QAM constellation diagrams [78]

The value that is transmitted by each sub-carrier is represented by: $X_k = I_k + iQ_k$, where I_k is a real part, and Q_k is an imaginary part of the signal. This X_k is called a modulation symbol. Different modulation techniques work with different input number of bits that can

be modulated into one modulation symbol:

- QPSK: 2 bits/modulation symbol
- 16QAM: 4 bits/modulation symbol
- 64QAM: 6 bits/modulation symbol [78, 64, 63]

2.1.7 Subcarrier allocation

One of the last parts of the signal process before the transmission is to shape the frames. LDACS has defined a frame structure that is based on the transmission of super frames (SF). The SF architecture differs for uplink and downlink. These frames are transmitted using the above-mentioned OFDM modulation. The SF is divided into 5 parts, which depend on the type of connection. The uplink SF content is 4 multi frames and 1 RA (Random Access), the content of the downlink SF is 4 multi frames and 1 BC frame (Broadcast). The strucrture of the uplink and downlink SF is represented in figure 2.9. [64, 63, 22]

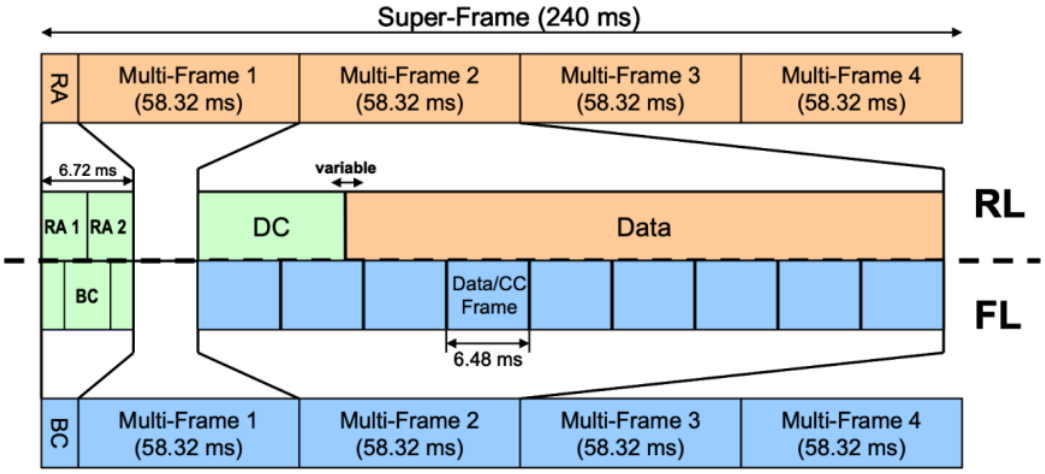

Figure 2.9: LDACS super frame structure. [64]

As shown in figure 2.9, the content of multi frames varies depending on the type of connection. On an uplink, every fourth multi frame contains nine Data / CC frames. In case of the downlink connections, every fourth multi frame contains DC (Dedicated Control) frames and Data frames the size of which may vary depending on the information being transmitted. Each Data / CC frame and BC frame, which is further subdivided into BC1, BC2 and BC3 frames contains synchronization symbols at the beginning. The synchronization symbols are used for the signal synchronization purposes. [64, 63, 22]

Looking closer to the symbol structure of the frames, the synchronization, PAPR reduction and the pilot symbols are places according to a certain pattern. The aim of these symbols is to make the LDACS signal more resilient to the interference. As mentioned before, the synchronization symbols are used during the synchronization process in the receiver. In figure 2.10 the position of the synchronization symbols is represented. [64, 63]

Figure 2.10: Synchronization symbols layout. [64]

The PAPR reduction symbols are placed according to a special pattern and aim to reduce the Peak to Average Power Ration[4]. The particular algorithm is not defined in LDACS specifications yet, but the pattern of these symbols is already available in LDACS documentation. The PARP symbol itself can be defined as one that carries no information and can be discarded at the receiver. [64, 63, 22]

The pilot symbols are inserted into the sub-frames as well. These pilot symbols provide an estimate of the channel at given locations within a sub-frame. There are a few techniques that can be used for these purposes. These techniques are often based on the assumption that the receiver is aware of the pattern of the pilot symbols, so the channel characteristic can be estimated by comparing the received data and the pattern itself. [64, 63, 21]

To form the frame, all the above-mentioned symbol types (synchronization, null, pilot, data, PAPR, AGC[5] and tile segmentation) are combined together according to a predefined pattern. Figure 2.11 represents the DC frame as an example of the final frame structure after the operations described above. [64, 63, 21]

[4]PAPR - Peak-to-Average Power Ratio is defined as the ratio of peak power to the average power of a signal.

[5]AGC - Automatic gain control symbol. Used during signal equalization.

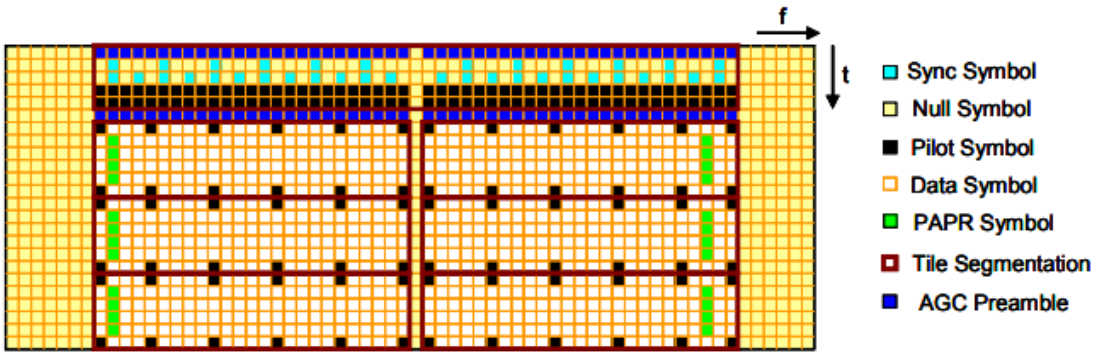

Figure 2.11: Structure of an RL DC slot [64]

2.1.8 Inverse Fast Fourier Transform

Once data are modulated and put into frames, it could be transformed using IFFT. The Inverse Fourier transform is used to convert the signal from the frequency domain to the time domain. To apply IFFT, a few mathematical manipulations need to be performed.

Considering the OFDM signal, the sub-carriers have to be shifted from the center frequency. This shift is described with $X(f-a)$, in the time domain, where a is a frequency shift. This also may be expressed as $x(n)e^{j2\pi at}$ in the frequency domain. However, the first window function[6] $g(t)$ is applied to cut out the unwanted parts of each sub-carrier signal.[20, 72] The next step is to sum up all these frequencies, figure 2.12.[20, 78]

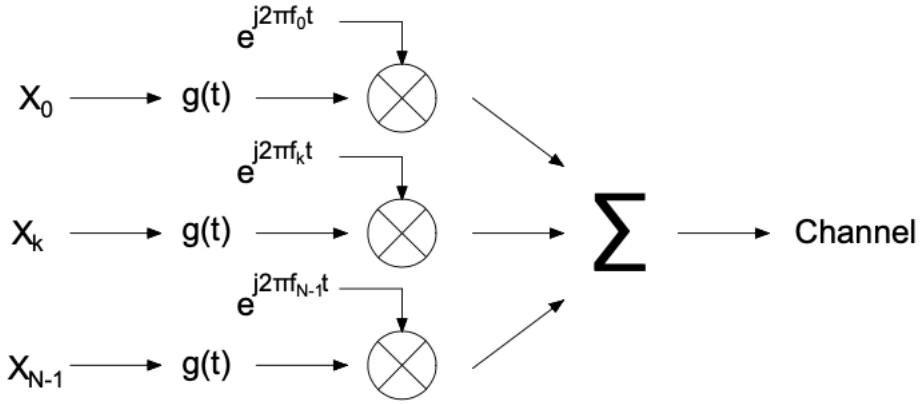

Figure 2.12: OFDM subcarriers summation [author]

After the said manipulations are completed, the signal is represented by an equation 2.7.[20]

[6] Window function is a mathematical function that is zero-valued outside of some chosen interval

$$x(n) = \sum_{k=0}^{N-1} X_k e^{j2\pi f_k t} \tag{2.7}$$

Then, the X_k is a time limited function of duration T of a particular carrier and each sub-carrier in this function is a frequency which is a multiple of the carrier spacing: $f_k = k\Delta f = \frac{k}{f}$. The carrier spacing is graphically represented in fig. 2.13. Then, substituting f_k with $\frac{k}{f}$, the signal is represented by eq. 2.8. [20, 23]

$$x(n) = \sum_{k=0}^{N-1} X_k e^{j2\pi t \frac{k}{T}} \tag{2.8}$$

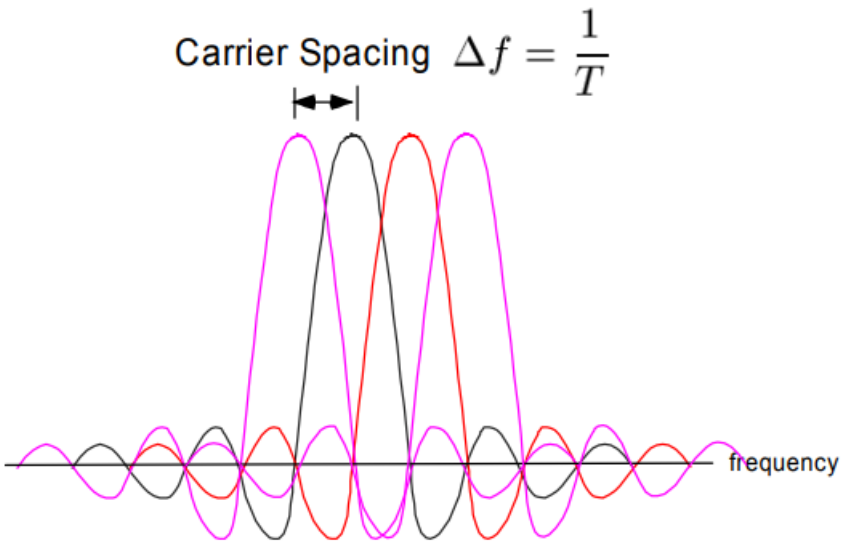

Figure 2.13: Carrier spacing in OFDM systems [20]

The next step is to sample this function (2.8) in the time domain with a sampling function. Once it is done, there is a possibility for further manipulations with this function. The aim of these manipulations is to get closer to the Discrete Fourier Transform formulation.

Sampling

Since the signal is sampled with a defined sampling rate, each sample is then separated by the equal intervals of Δt. Also, as already mentioned, the signal is time limited, and the duration of the signal is T_{sig}. Considering that in this interval of T_{sig} there are N points (samples), the between sample interval is described with $\Delta t = \frac{T_{sig}}{N}$. So, any interval between the samples can be expressed by: $t_n = n\Delta t$. This expression is the actual position of the sample on the time axis. And at each sample point, there is an actual value for x_n (sampled

version of $x(t)$ at that point). This means, that it is possible to replace t with t_n, eq.2.9. [20, 72, 23]

$$x_n = \sum_{k=0}^{N-1} X_k e^{j2\pi \frac{k}{T_{sig}} n\Delta t} \tag{2.9}$$

Δt may be also substituted for $\frac{T_{sig}}{N}$, eq. 2.10

$$x_n = \sum_{k=0}^{N-1} X_k e^{j2\pi \frac{k}{T_{sig}} n\frac{T}{N}} \tag{2.10}$$

So, after the T_{sig}s cancel out, the final expression (eq.2.11) matches the definition of Inverse DFT (eq.2.12) except $\frac{1}{N}$.

$$x_n = \sum_{k=0}^{N-1} X_k e^{j2\pi n \frac{k}{N}} \tag{2.11}$$

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j2\pi n \frac{k}{N}} \tag{2.12}$$

What is proposed for OFDM, is that IDFT is performed on N constellation points, and the receiver performs DFT on received samples x_n to recover the n complex constellation points X_k, which then can be demodulated back to the original bits, and the original bit-stream is recovered. The conceptual diagram of the actions between X_k before the transmission and X_k after the reception is graphically represented in figure 2.14 [72, 20, 23]

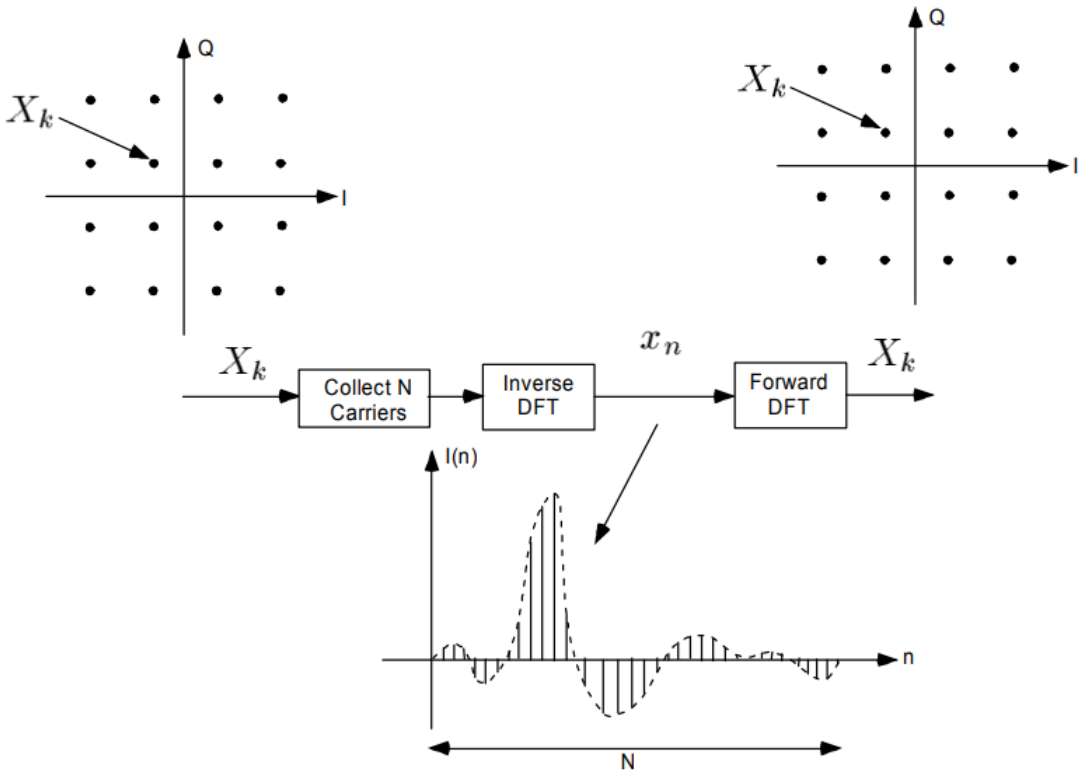

Figure 2.14: The actions between X_k before transmission and X_k after reception [20]

In OFDM systems, IFFT/FFT is used instead of DFT/IDFT. The intention is that FFT/IFFT requires fewer computational resources and provides the same output as the discrete version. [72, 20, 64]

2.1.9 Cyclic prefix insertion and Windowing

One of the last operations before the signal transmission is to window the signal. This is done in order to mitigate the influence of LDACS onto the existing L-band systems, mainly DME. The windowing function aims to smooth the sharp phase transition between the OFDM symbols. The equation 2.13 represents the conditions and the function of windowing. [64, 63]

$$w(t) = \begin{cases} \frac{1}{2} + \frac{1}{2}\cos(\pi + \frac{\pi t}{T_w}), \text{ for } 0 \leq t < T_w \\ 1, \text{ for } T_w \leq t < T_s \\ \frac{1}{2} + \frac{1}{2}\cos(\frac{\pi(t-T_s)}{T_w}), \text{ for } 0 \leq t < T_w \\ 0, \text{ else} \end{cases} \tag{2.13}$$

Where T_w represents the duration of the flanks of the window. The windowing is also

visualized in figure 2.15, [64, 63]

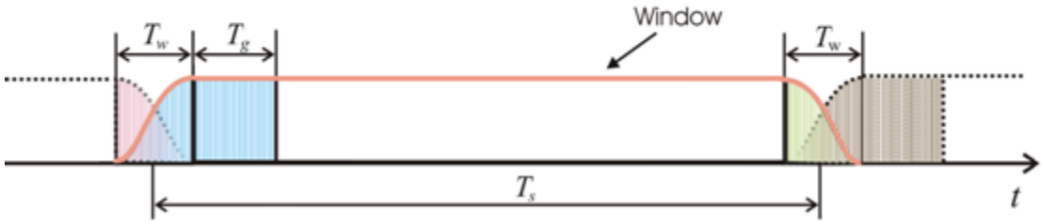

Figure 2.15: Windowing in the LDACS system. [64]

In figure 2.15, T_g is the guard time - the cyclic prefix (described in section 2.1.1), and T_u is the useful time of the symbol. [64]

Cyclic Prefix and postfix

The last step of the OFDM signal processing is adding a cycling prefix to the transmitted signal. In reality, OFDM is prone to inter-channel interference (ICI) and inter-symbol interference (ISI) due to the multi-path. Such types of interference come up when the signal on the receiver is received with a time offset. To overcome this problem, a cyclic prefix is applied. Basically, it is done by copying several last samples of the symbol to the beginning of that symbol. Then, the symbol sequence after the IFFT operation is represented by $x = [x(0) \quad x(1) \quad x(2) \quad x(3 \quad ... \quad x(N-1)]$. Considering the length of the cyclic prefix of $L-1$, the final OFDM symbol can be represented by the eq.2.14. [19, 9]

$$x = [x(N-L+1) \quad ... \quad x(N-2) \quad x(N-1) \quad x(0) \quad x(1) \quad ... \quad x(N-1)] \tag{2.14}$$

This process is graphically represented in figure 2.16, where some symbols from the end of the signal are moved to the beginning. This number of samples can vary due to the surrounding conditions of the multi-path, for instance, normal duration of CP in LTE[7] is 7% of the symbol. [76, 4]

[7] LTE - Long-Term Evolution is a standard for wireless broadband communication for mobile devices and data terminals which uses two types of cyclic prefixes with different lengths.[76]

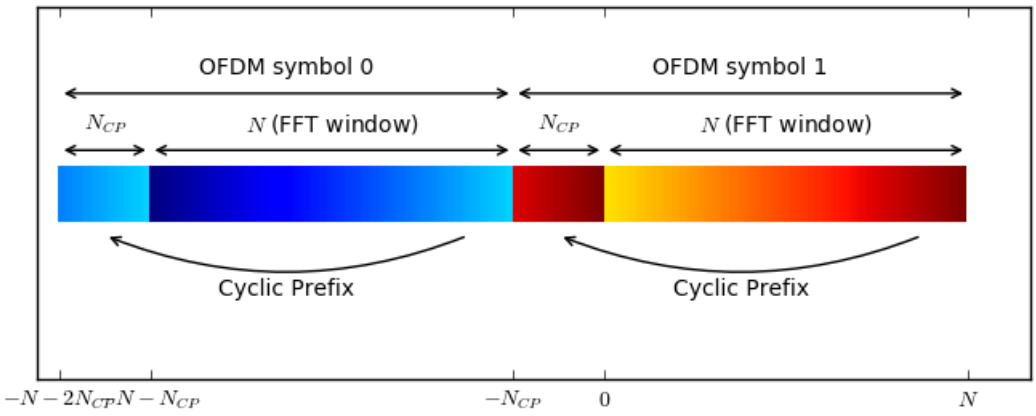

Figure 2.16: Cycling prefix conceptual scheme [35]

The introduction of the cyclic prefix has its own advantages and disadvantages. The main advantages include the ones that have already been mentioned above and also the improvement of its resilience to ICI and ISI. The main disadvantage is the data capacity reduction due to re-transmission of data. [9, 19]

As well as with a cyclic prefix, LDACS is a feature with a cyclic post-fix. A cyclic post-fix is a certain number of symbols coming from the beginning of the signal to the end (the opposite way compared to a cyclic prefix).

2.2 LDACS receiver structure

On the transmitter side the signal process is a bit more complicated. This is caused by the fact that the signal has to be synchronized and equalized in order to begin the demodulation and decoding process. This process includes:

- Synchronization
- Blanking non-linearity correction
- Cyclic prefix and postfix removal
- Fast Fourier Transform (FFT)
- Channel Estimation
- Equalization
- Demodulation

- Helical deinterleaving
- Convolutional Decoding
- Block deinterleaving
- RS decoding
- Derandomization[64, 63]

This section aims to describe the above-mentioned processes - from the signal reception, up to the decoded data stream. The structure of the transmitter is also described in figure 2.3.

2.2.1 Blanking non-linearity correction

The blanking non-linearity function is performed in order to achieve a greater signal to interference to noise ratio. In fact, signal parts with an amplitude greater than T_{BN} are blanked. The blanking function is represented in the following eq.2.15. [64, 63]

$$y_l = f(r_l) = \begin{cases} r_l, \text{ if } |\mathrm{r}_l| < T_{BN} \\ 0, \text{ else} \end{cases} \tag{2.15}$$

Where l takes values from 1 to 64. [64, 63]

2.2.2 Synchronization

The synchronization function is implemented in order to anticipate the errors caused by the sensitivity of the OFDM systems to timing and frequency offset. Since any synchronization error will degrade the performance of the OFDM system by the effect of ICI and ISI, the synchronization is an important step to obtain reasonable system performance indicators. The LDACS system employs CAZAC (constant amplitude zero autocorrelation) algorithms for the frequency and time synchronization. [64, 63]

The synchronization symbols, that are inserted during the framing, are used during the CAZAC synchronization sequence. The main goal of this synchronization sequence is to construct the preamble symbol. The property of the CAZAC sequence is not affected by IFFT and FFT. With that feature, the preamble symbol can be constructed by repeating the sequence after the IFFT operation. With these structure and synchronization symbol characteristics the timing and frequency synchronization can be performed according to (A

Novel Synchronization Algorithm Based on CAZAC Sequence for OFDM Systems). [64, 63, 45]

The calculation of the synchronization sequence varies depending on the OFDM symbol number. In case of the first row in figure 2.10, the synchronization sequences in the LDACS system shall be calculated by:

$$S_{sy1,k} = \sqrt{4}e^{j*\pi\frac{5k^2}{N_{sy1}}} \tag{2.16}$$

In case of the second OFDM symbol (lower row in figure 2.10), the calculation is made by eq. 2.17

$$S_{sy1,k} = \sqrt{2}e^{j*\pi\frac{k^2}{N_{sy2}}} \tag{2.17}$$

both of these equations are calculated for $k = 0, ..., N_{sy1/2} - 1$. In these equations $S_{sy1/2}$ is the synchronisation symbol for the first and the second OFDM synchronisation symbol and $N_{sy1/2}$ is the number of synchronisation symbols per OFDM synchronisation symbol (12 for the first OFDM synchronisation symbol and 24 for the second OFDM synchronisation symbol).[64, 63]

According to the synchronization results the signal is cut further and the cyclic prefix and post-fix are removed. Once it is done, the signal and its components are ready for further operations of channel estimation and error correction.[64, 63]

2.2.3 Channel estimation

Since the OFDM systems may be highly influenced by the ICI and ISI interference, the high channel estimation accuracy will be provided in order to maintain the performance of the OFDM system. The technique with the periodically inserted pilot symbols is a frequently used idea because of its relatively simple implementation and accurate and efficient performance. The pilot symbols allow to track the time variation and the frequency selectivity of the channel. At the receiver, the channel complex gain at the pilot symbol positions can be easily obtained from the received signal and the known pilot symbols. The interpolation is then applied to derive the estimation of the channel knowledge from the data symbol positions. [77, 26]

In the LDACS system the channel estimation is made with either the wiener interpolation or the linear interpolation. The detailed sequence of action of these types of interpolation as well as the MMSE estimation, that is used in the LDACS system are provided in "Linear

Interpolation in the Pilot Symbol Assisted Channel Estimation for the OFDM" paper. [77, 26]

2.2.4 Equalization

When the signal passes the channel, certain distortion is introduced. This distortion may be presented by means of amplitude and signal delay. The distortion creates a problem of ISI, so the detection of the signal at the receiver might be more complicated. [64, 52] The aim of the equalization is to correct the said distortion. The equalizer is basically a filter that scales the signal to the constellation points of the modulation alphabet. [64, 52]

2.2.5 Demodulation

The demodulation operation is a reverse operation of mapping the bits into complex values. The outcome is a matrix of a certain format filled with bits. These bits are further decoded in order to perform error correction. [20, 64]

2.2.6 Signal decoding

The signal decoding part of the process in the LDACS transmitter in fact makes the reverse process of channel coding in the transmitter. It includes the de-interleaving process, where data are rearranged back into the appropriate format to be further decoded. Then, the Veterbi decoder[8] is engaged. It aims to decode the data, so that the data is the same as before coming through the convolutional coder (in a perfect case). The latter part of the channel decoding is to use the Reed-Solomon decoder. The operation of decoding is described in detail at the beginning of this chapter in subsection 2.1.3. The data then represent the final received sequence of bits. Then, the errors in the bits are corrected so that the said bits can be used to obtain the binary information. [16, 18, 64]

2.3 OFDMA

Since in practice several aircrafts can be in connection with one ground station at the same time, the reception of several signals by one receiver has to be ensured. It is done by OFDMA. OFDMA - Orthogonal frequency division multiple access is a version of FDM (Frequency

[8] A Viterbi decoder uses the Viterbi algorithm for decoding a bit-stream that has been encoded using a convolutional code

division multiplexing). It is used to divide packets of information into separate bands that are carried by from different transmitters. It aims to ensure that the base station can receive signals from multiple transmitters. On the left-hand-side of figure 2.17 the downlink transmission from the base station to multiple subscriber stations is represented. The distance between the BS and each SS is different, so are the time and frequency offset of the signal. In this type of BS and SSs interconnection is less complex since the synchronization algorithms that are described in 2.2.2 are capable to handle it. [61]

In case of uplink (right-hand-side on figure 2.17), assuming that SSs are synchronized in time, the signal on BS will be received in different time and with different power level. The timing of the signal is crucial, since the OFDMA systems (as well as LDACS[9]) use the Time Division Duplex[10] model, so the delayed signal might not even fit into the time dedicated for signal reception by BS. Assuming that the SSs will transmit the signal in different time compared to each other, SS3 in figure 2.17 will transmit the signal with the offset compared to SS1, so the base station receives both signals simultaneously. [14, 61]

As already mentioned, the power level of the signal from SSs, located at different distance is going to be different, ex. SS1 and SS3 in figure 2.17. The signal from SS1 would dominate and cause very noticeable interference into the carriers. In view of the above, algorithms that will accomplish the timing and introduce the power control feature are needed.

[9] LDACS Reverse Link transmission is based on OFDMA-TDMA bursts, with silence phases between such bursts

[10] TDD - duplex communication links where uplink is separated from downlink by the allocation of different time slots in the same frequency band

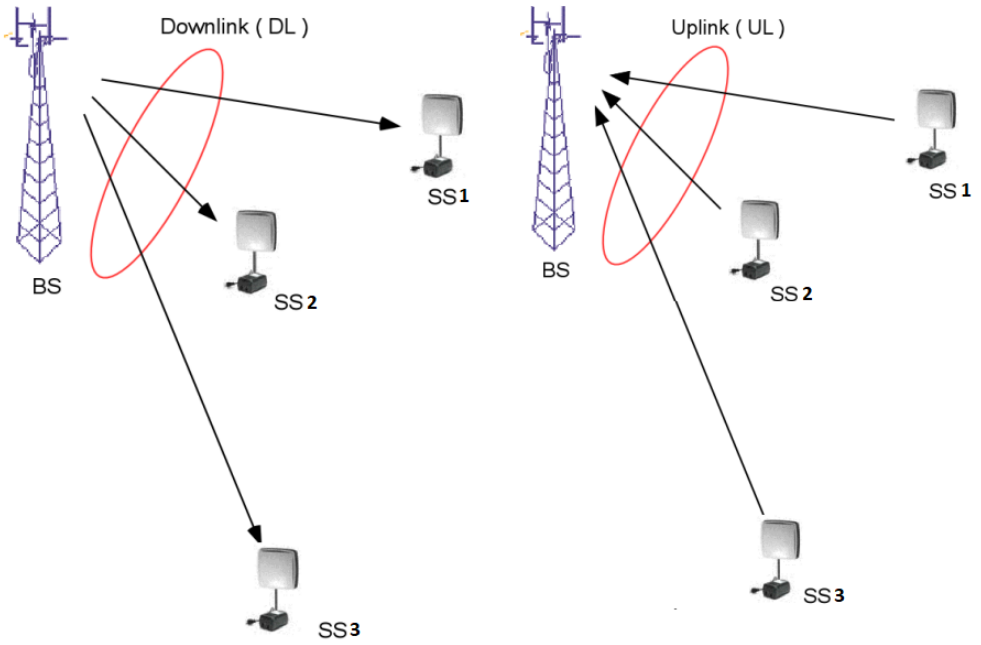

Figure 2.17: Uplink and downlink transmission [14]

Even though the OFDMA method is used on LDACS, it is out of scope of this thesis, since the testing platform is made in order to simulate the signal transmission between BS and the only one SS.

2.4 Summary

In this chapter the LDACS technical structure is described. It includes data coding and interleaving, modulation, IFFT, subcarrier allocation and cyclic prefix and postfix insertion. The structure of LDACS resembles other civil non-aeronautical systems such as LTE or WiMax with minor differences. Although this system appears to be quite unique in the aeronautical field, due to the structure described above in this chapter, it can ensure very high information transfer bitrate that cannot be even compared to the speed of the existing CNS systems. Moreover, because of the advanced error correction techniques, the system is capable to ensure solid bit-error-rate, sufficient even under moderate channel interference.

This detailed descriptions of the LDACS system transmitter and receiver will serve as theoretical basis for further pats of this thesis.

Chapter 3

Methodology

This chapter describes the methodology proposed for this thesis. The goal of this thesis is to propose the LDACS testing platform including HW, SW and testing scenarios and evaluate its use. Based on the said goal, it is clear that first of all, there is a need to define the HW and SW solution for the platform. There are two ways to create the signal: analog and software. Due to the complexity and price of the analog systems, this thesis has the LDACS signal defined on the software level. HW to send software defined signals is called Software Defined Radio (SDR)[1]. Since SDR has to be configured on the software level, there is a need to choose software with suitable environment for signal definition and signal processing. That SW solution also has to be compatible with different SDR types for cases when the chosen HW will be unavailable. In order to define the exact SW and HW solution, the research of the hardware and software options is done in the next chapter 4.

Once the HW + SW solution is defined, the following process is proposed:

- Create requirement framework for the software and hardware solutions that can be used. Choose the most suitable software for LDACS signal transmission and reception.
- Create the LDACS testing platform on the software level - create the LDACS transmitter and the receiver according to the theory described in chapter 2. The created model should contain all the processes that are going on during LDACS signal generation. To create the LDACS platform, the existing scripts can be used, e.g. the existing models of OFDM components (ex. CC coder/decoder or Helical interleaver/deinterleaver) can be used as "building blocks" in order to create the LDACS transmitter and receiver models. Then, the software level platform has to be tested. The simulation should also

[1] Software defined radio is described in detail in 4.2

have an ability to evaluate the performance of the transmission by calculation of the performance indicators.

- Integrate HW into the platform – add the option to transmit and receive the LDACS signal using SDR. The transmission with the hardware must be integrated into the LDACS software model, e.g. the simulation has to have customizable setting of software/hardware channel.

- Create testing scenarios and test the platform according to them. The scenarios should simulate the real usecase of the LDACS system. The scenarios have to be tested either by SW or SW + HW means.

- Evaluate the proposed platform – first, SW results feasibility and possible performance loss of SW + HW integration has to be discussed. Second, evaluate the platform by the means of usage and rebuild complexity, extension options and a variety of theoretical testing scenarios.

Chapter 4

Research of the tools for the LDACS test platform

This chapter presents various software and hardware tools that are capable of signal modelling, transmission and reception and therefore can be used as a basis for the LDACS test platform proposed later in this thesis. These software and hardware solutions are discussed and most convenient is then selected for the test platform proposal.

4.1 Software for the LDACS platform

Considering SDR for LDACS signal modeling, there is a need for environment in which signal would be defined and processed. Different SDR manufactures are implementing support of different software solutions for their products. However, there is a list of software that supports almost any SDR device and all of them can perform sophisticated calculation such as signal processing for transmission and reception. There are three main software solutions that are recommended and supported by the widely known companies. These software solutions are: GNU Radio, LabVIEW and Matlab/Simulink. [55]

4.1.1 GNU Radio

GNU Radio is free and open-source software that provides signal processing functions. This software is capable to define signals for SDR. It offers a possibility to use external software as well as create models without hardware in a simulation-like environment. GNU Radio has several advantages. These include a rich library with a number of blocks for signal processing, support of the main used operational systems (Windows, Linux, Mac OS), the

opportunity for a designer to process a signal stream in real time and a large community of users and designers. [48]

The library of GNU Radio includes an OFDM prefixer, an interleaver/deinterleaver, channel estimation, a randomizer/derandomizer, an OFDM transmitter/receiver, convolutional and RS coding/decoding blocks. These blocks can be used during the LDACS signal generation. Also, GNU Radio offers a possibility to create user-defined blocks to perform specific operations. The existing models of OFDM and systems similar to LDACS are available to be used to create the LDACS model.

GNU Radio can be downloaded from the official page or GitHub. The installation of this software is straightforward. [48]

4.1.2 MATLAB

MATLAB is a proprietary multi-paradigm programming language and a numeric computing environment developed by MathWorks. MATLAB is featured with a wide variety of built-in functions. Some functions, such as "fft()" (FFT of function in brackets), can be very useful for LDACS signal generation. Also, MATLAB can be used as a programming tool that supports Object-Oriented Programming (OOP). Another great benefit of MATLAB is Simulink. Simulink is a built-in graphical programming environment for model and simulations creation. This tool can simplify the simulation creation and setup.

As already mentioned, MATLAB is featured with several built-in functions that can be used for LDACS test platform creation on the software level. But what is more important, the LDACS development team has released an official MATLAB simulation.

4.1.3 LabView

LabView stands for Laboratory Virtual Instrument Engineering Workbench and it is a system-design platform and a development environment for a visual programming language from National Instruments. National Instruments is one of the greatest USRP manufactures, and LabView was created with native compatibility with their products. Regarding integration, LabVIEW has a native function for hardware set up. This feature supports a large list of hardware, especially USRPs, that could be used with this software. LabView is featured with everything that is needed for signal generation and transmission as well as with example models of configurable OFDM solutions. [12]

4.1.4 Analysis on SW decision for the LDACS platform

The ideal software for the LDACS testing platform should meet the following criteria:

- Availability for the end user - ideally the SW has to be free and open source.
- SW+HW compatibility - it should be compatible with a wide range of SDR solutions.
- SW usage simplicity - the software solution has to be intuitive and easy to use.
- SW popularity - the platform is proposed mainly for academical use, so the popularity of the SW in the academic environment plays a role.
- SW advancement in terms of signal processing - the SW should be featured with functions for signal processing.
- Existence of similar projects - since a similar existing code can be used, software with existing "building blocks" and strong community support is preferred to simplify the creation of the platform.

The software solutions are evaluated according to the criteria mentioned above using the multi-criteria analysis. The evaluation is shown below in table 4.1.4. Some of the criteria above are less critical than others. The evaluation criteria of availability for the end user, SW complexity and SW popularity are estimated with a weight factor of 0.5. These criteria do not influence the process of the LDACS platform creation. The other criteria have a weigh factor of 1. Each system is estimated with 1-3 points, where if the SW is estimated with 3, it means that it strongly meets the requirement described above, if the system SW is evaluated with 1, it weakly meets the requirement (ex. for the availability, GNU Radio scored 3 points, since it is a free and open source. MATLAB and LabView are available with student licence and evaluated by 2 points).

Table 4.1: Software sulutions comparison according to the criteria

Software solutions comparison			
Software	MATLAB	GNU Radio	LabView
Availability (0.5)	2	3	2
SW+HW compatibility (1)	3	2	2
SW usage simplicity (0.5)	2	1	2
SW popularity (0.5)	3	2	2
SW advanceness in terms of signal processing (1)	2	2	2
Existence of similar projects (1)	3	2	1
Score with respect to weight factor:	11.5	9	7.5

Due to the compatibility made in the form of add-ons for a wide range of HW solutions, SW popularity in the academic community and the existence of official LDACS simulation on MATLAB basis, MATLAB got the highest score during the evaluation. Thus, MATLAB is the best HW for the purpose of LDACS platform creation.

The MATLAB simulation for LDACS is created and distributed by DLR. It is a free and open source. The simulation is featured with the LDACS transmitter and receiver as well as with a simulated channel and L-band interference models. The transmitter and receiver are defined according to SESAR2020 – PJ14-W2-60 - Initial LDACS A/G Specification [64]. It is the latest specification for this system available at the time of writing this thesis. In the next section this simulation is represented in detail and its use for the purpose of creating the test platform is discussed. [29]

4.2 Hardware for the LDACS platform

Software Defined Radio (SDR) also known as Software Radio can be described with a number of definitions. The SDR Forum, that is working in collaboration with the Institute of Electrical and Electronic Engineers (IEEE), defines SDR as:

"Radio in which some or all of the physical layer functions are software defined" [25]

In this context radio has the meaning of a technology for signalling and communicating using radio waves, and the "physical layer functions" are characteristics of a transmitted or received electromagnetic wave. The main difference between SDR and hardware-based

radios is that the SDR system can be configured to transmit and receive signals across a large frequency range using programmable hardware devices. These devices may have difference in general use (transmitter, receiver or both) as well as in more detailed specifications, for example the range of frequencies. [25, 34]

SDR architecture

Since the concept of a Software-defined radio was to create a tool that would have a wide range of radio frequency bands and modes defined by software, this means it would consist only of an antenna capable to send/receive signals, DAC or ADC (depending on type of device - transmitter or receiver) and a programmable processor. In real SDR systems some additional components are needed. [25, 34]

Typically the SDR architecture is divided into two subgroups:

- Radio Frequency Front-End - the main function of this subgroup is to create an analog output out of a digital input from the digital back end. This is done by adding a set of amplifiers, filters and local oscillators to the antenna(s) and ADCs/DACs. The input digital signal is converted into the analog form using DAC. After that, the analog signal is filtered and synthesised using local oscillators to intermediate frequency (the frequency to which a carrier wave is shifted). The last step of signal transmission for a common SDR is to amplify the signal and lead it to the antenna. The receiver side of common SDR has a very similar structure to the transmitter, but the sequence of actions is the opposite. After the signal is received, amplified and its carrier frequency is converted into a information-bearing signal (low-intermediate frequency), the signal digitisation is performed. [25, 34, 24]

- Digital Back-End, or as it is called in some papers - Multi-Processor Subsystem, is a computational set for signal processing. It consist of General Purpose Processors (GPP), Digital Signal Processors (DSP)[1] and Field Programmable Gate Arrays (FPGA) [2]. [25, 34, 24]

The following figure 4.1 describes the SDR architecture in the form of a block diagram.

[1] Digital signal processors (DSPs) are specialized microprocessors that are made to fulfill fast operational needs and in case of SDR - to fulfil the needs of digital signal processing. [13]

[2] FPGA is a device that can be reprogrammed to a desired application or functionality requirements. [13, 7]

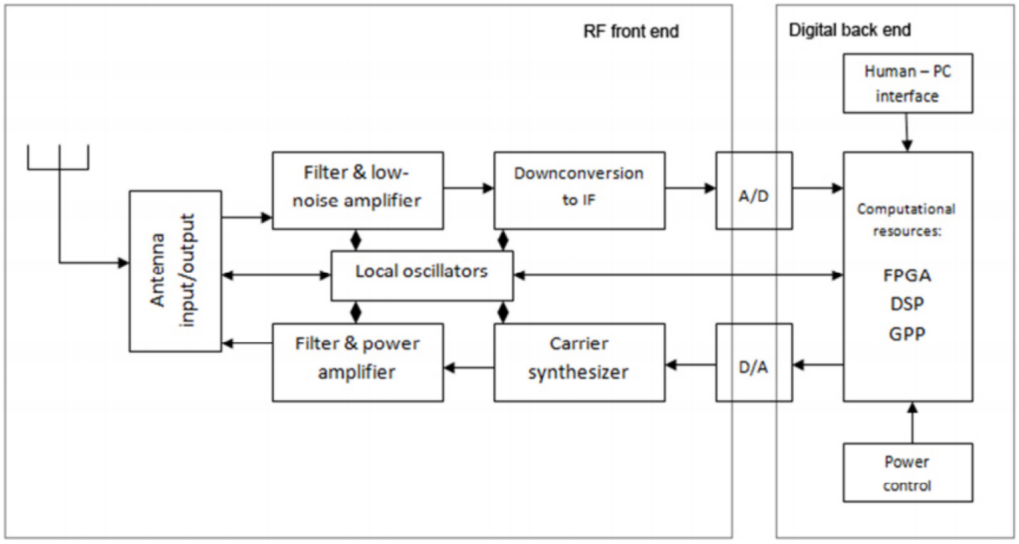

Figure 4.1: Software defined radio block diagram [24]

There are a lot of SDR solutions from different manufactures on the market. They differ in price and performance. The main limitations of SDRs is digital to analog/analog to digital converters. In case of low performance of a DAC/ADC chip, it may convert the signal from digital to analog form imprecisely which will cause great performance loss. Also, a DAC/ADC chip limits top frequencies that can be used by the digital section.

As an example, in the following picture 4.2 several SDR systems are represented. The difference between these SDRs is in HW computation, RF performance and price.

Figure 4.2: Overview of some SDR solutions available on market.[53, 54, 36, 59, 30]

From LDACS specifications, all these HW solutions can be used for LDACS signal transmission[3]. However, low resolution of DAC/ADC chips of the cheapest solutions may introduce mentionable errors. The hardware solution choice is influenced by the availability of these SDRs in the university laboratory. For the first steps of HW integration into the LDACS test platform, available hardware with highest performance is used. Later on, the possibility of HW downgrade is discussed.

[3]LDACS signal specification: center frequency - around 1 GHz, bandwidth - 498 kHz.

Chapter 5

LDACS test system proposal

The previous chapter dealt with the research of software and hardware solutions. This chapter describes the process of creating the LDACS testing platform. This process includes the determination of the software use (if the MATLAB simulation for LDACS can be used or if the LDACS system model has to be created separately), software interconnection and the resulting evaluation of the system efficiency.

5.1 MATLAB simulation for LDACS

In the previous sections of this thesis the MATLAB simulation for LDACS has already been mentioned several times. This section describes the official MATLAB LDACS simulation in detail. It also includes the way the simulation settings can be customized.

The simulation can be downloaded for free from the official LDACS website [29]. The downloaded simulation is archived in .rar format. So, in order to start the simulation, it has to be extracted. Then, it can be launched with `main_file.m`. The first step is to define the working folder and a Java pass on line 21 and line 23 of the `main_file.m`. This setup differs depending of where java and the simulation itself are saved. Afterwards, the simulation can be run with default settings (predefined by DLR) as an ordinary MATLAB script.

5.1.1 Simulation structure

MATLAB simulation for LDACS is complex. It contains more than 100 .m files with single or multiple functions inside. The main file of the simulation contains different functions such as the LDACS transmitter and receiver. The whole simulation in a simplified manner may be described by a block diagram, figure 5.1.

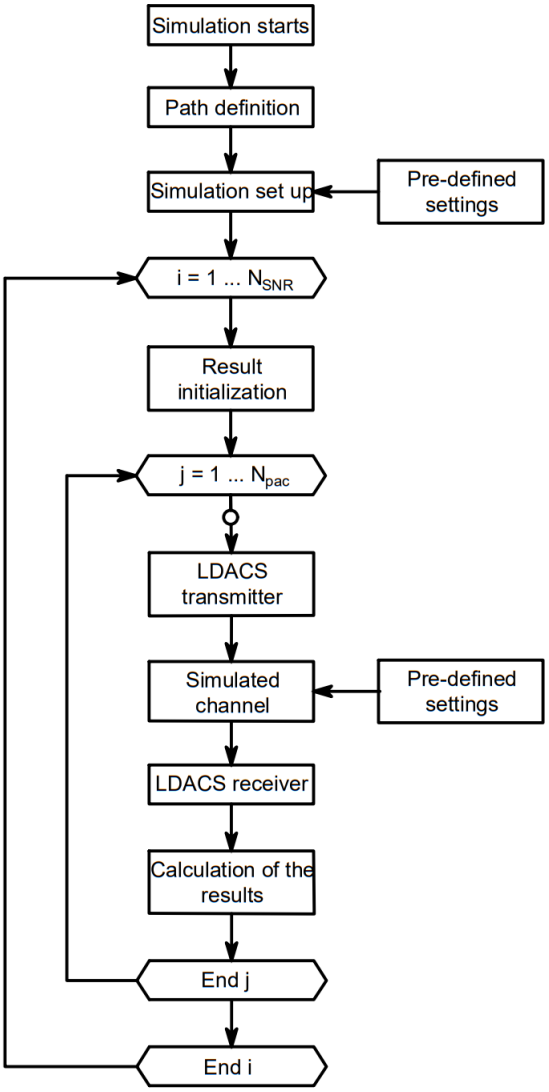

Figure 5.1: MATLAB LDACS simulation block diagram

According to the simulation algorithm, the signal transmission and reception as well as the result calculation is inside a double loop. The first loop indicates SNR values, the other one indicates frame quantity. This feature allows to generate the result of the transmission of N_{pac} packets for N_{SNR} SNR values.

Individual blocks of the algorithm in figure 5.1 are described in the following subsections.

Simulation set up

After defining the path for the simulation and before the transmission, three functions are run in order to initialize the main variables that are then used during the transmission. These functions are: `set_additional_parameters`, `display_option`, `set_parameters_fl`

(for forward link) or `set_parameters_rl` (for reverse link). The first function that defines the parameters is either `set_parameters_fl` or `set_parameters_rl`. They define the mode of the transmission - reverse or forward link. The difference between them is in the frame structure as well as in the frame length (more information in chapter 2.1.7). To choose the forward link transmission the user should comment the `[param struct, ... sim param] = set parameters rl;`, otherwise comment on `[param struct, sim ... param] = set parameters fl;`. These functions return structures with the parameters - `param_struct` and `sim_param`. The structure contains the most critical parameters for signal generation and transmission (OFDM parameters, channel type, interference, the number of frames, ect.) and `sim_param` which contains parameters of information about signal and interference strength (SNR, EbN0 [1], ect.)

This function contains the greatest number of customizable settings compared to other functions. The setup of these parameters is done by changing the value of the appropriate variable. The options for the settings are represented below and are also commented on inside the function code. The setting proposed in `set_parameters_fl` and `set_parameters_rl` are described in table 5.1.1.

Table 5.1: Overveiw of availible settings for `set_parameters_fl/rl`

Simulation parameter setup - set_parameters_fl/rl		
Variable	**Description**	**Options**
Simulation length		
sim_param.N_packets	Number of packets in one frame	Integer value
param_struct.N_frames	Number of frames	Integer value
Coding and modulation		
param_struct.codmod.mod	Modulation type	QPSK, 16QAM, 64QAM
param_struct.codmod.rate_cc	Convolitonal coder rate	0.5, 0.75, 0.67
framesparam_struct.codmod.N_frame_joint	Number of jointly encoded frames	Integer value
Channel parameters		
sim_param.SNR_vec_dB	Signal to noise ratio	Double value
param_struct.chan.channel	Channel model	AWGN, ENR, APT, TMA
param_struct.ce.chan_est	Interpolation type	linear, Wiener, perfect
param_struct.chan.interference	Added interference type	DME, GGI, off
param_struct.int_mit.td_mitigation	Interference mitigation parameter	blanking nonlinearity, off
Synchronization parameters		
param_struct.sync.sync_method	Synchronization method	perfect, real

[1] EbN0 - Energy per bit to noise power spectral density ratio

The options for these settings are made according to the theory represented in chapter 2.

Another function for the simulation setup (`set_additional_parameters`) sets further parameters for signal generation and transmission. It includes parameters for up-sampling on the receiver, ensuring synchronization, pilot symbols and receiver filters. The last function that is used during the simulation set up is `display_optoins`. In this function the performance indicators that will be displayed after the transmission are set.

Result initialization

MATLAB simulation for LDACS initializes the result of the previous transmission from the root folder. Also, in this step a new value of SNR is set for the run.

Simulated LDACS transmitter

The transmission vector generating the sequence starts from generating random data. These data are in the binary form. In this case they represent the real data that can be transmitted by LDACS. Random data are created with a function called `create_data`.

The data are further placed into the randomizer function `ldacs_randomizer`, which performs the operation described in section 2.1.2.

Once the data are scrambled, they can be encoded with the Reed-Solomon and Convolutional code and also interleaved. This is done by the function `channel_coding`. The coding and interleaving sequence is as follows:

- RS coder
- Block Interleaver
- Convolutional Coder
- Helical Interleaver

The functionality of this block is done according to the sections 2.1.3, 2.1.4, 2.1.5 and 2.1.6 respectively. Also the LDACS transmitter strictly follows the scheme in figure 2.3.

After the data are channel-coded and interleaved, the modulation is performed. The function for the data modulation is `modulation`, assuming that the modulation is chosen in function `set_parameters_fl`.

The next operation is to form the frames and add the additional sub-carriers as described in section 2.1.7. After the sub-carrier allocation, the cyclic prefix and postfix addition, the

windowing of IFFT can be performed. This is done by function `trans_vec_gen` and its functionality is described in the theoretical part of the thesis in section 2.1.9.

The results of the LDACS transmitter operations are stored in the `data_struct_transmit` structure. This structure contains the final transmission vector as well as the intermediate results of signal processing.

Simulated channel

MATLAB simulation for LDACS is featured with a simulated channel that adds noise and interference to the signal that is coming through it. The channel models have already been mentioned in the simulation setup description. These are: Additive white Gaussian noise (AWGN), En-Route channel (ENR), Airport channel (APT) and Terminal channel (TMA). The main difference between these channels lies in parametrizacion of the parameters that influence the signal. These parameters include the Rician factor[2], path delays, the number of samples per frame and other factors that influence the signal during its path propagation.

Simulated LDACS receiver

The receiver's main role is to process the received signal in the time domain up to the decoded and deinterleaved bit stream. First, the received data in the form of a complex vector are synchronized (described in section 2.2.2) with function `synchronisation`. This makes correction with a time offset of the signal that is caused by the channel. Then, blanking non-linearity correction is performed (function `blanking_block`). This function also removes the cyclic prefix and postfix and performs FFT. The last two steps before demodulation are to perform channel estimation and equalize the signal according to the theory described in sections 2.2.3 and section 2.2.4 respectively. There are two separate Matlab functions for that: `chan_est` and `equalizer`.

Then, demodulation is made with function `demod`. After the demodulation the data are in binary form. This binary data are then decoded and deinterleaved the opposite way as they were coded and interleaved on the transmitter side. This process includes:

- Helical deinterleaver
- Convolutional decoder

[2]Rician Factor - is defined as *"The ratio of signal power in dominant component over the (local-mean) scattered power."* [76]

- Block deinterleaver
- RS decoder

After these processes the data have to be derandomized. Then, the result data are achieved and stored into the `data_struct_rec` structure.

Calculation of the results

At the end of the simulation performance the indicators are calculated and the results are stored in the root folder. During the calculation process the following performance indicators are calculated:

- Bit Error Rate - is a ration between the bits that are incorrect after reception (compared to the original message) and the total number of bits for the transmission. For LDACS system BER is required to be lower than 1e-6. BER is calculated twice - after CC decoding and after RS decoding
- Packet Error Rate - The number of error packets after Forward Error Correction (FEC) (CC and RS decoding) divided by the total number of received packets. PER is calculated two times - after CC decoding and after RS decoding
- Mean Squad Error - is an average squared difference between the estimated values and the actual value.

The result of error calculation is stored in the `result_struct` structure.

Workspace structure

The operations described above generate and accept certain outputs and inputs. These structures and their content have already been mentioned in the previous subsections. The table below represents the overview of the function outputs and inputs that are saved into the MATLAB base workspace.

Table 5.2: Overveiw of availible settings for `set_parameters_fl/rl`

Overview of functions inputs and outputs		
Function	**Input parameter**	**Output parameter**
Simulation setup		
set_parameters_fl		param_struct, sim_param
set_paramters_rl		param_struct, sim_param
set_additional_parameters	param_struct, sim_param	param_struct, param_struct2, sim_param
display_option	param_struct	display_struct
Result initialization		
select_parameter	param_struct, sim_param, k	param_struct
init_result	param_struct	result_struct, param_struct
LDACS transmitter		
transmitter_ldacs	param_struct	data_struct_transmit
Channel model		
channel_model	data_struct_rec, result_struct	data_struct_transmit, param_struct, param_struct2
LDACS receiver		
receiver_ldacs	param_struct, param_struct2 data_struct_rec, result_struct	data_struct_rec, result_struct
Calculation and storing of the results		
error_calculation	result_struct, data_struct_rec data_struct_transmit, param_struct	result_struct

The most critical structures are: `param_struct` - contains the variables that describe the settings of the simulation which were set in `set_parameters_fl/rl`, `data_struct_transmit` - contains the final generated vector for the transmission, `data_struct_rec` - contains the received vector as well as decoded data.

5.1.2 Evaluation on MATLAB simulation for LDACS usability

The simulation described above is a very powerful tool to generate, transmit and receive the LDACS signal. It is featured with a wide variaty of set up options as well as with a mechanism to calculate the transmission results after it ends. Also, the simulation is distributed by the developer team of this system which means that it must pass the quality and testing control.

The only feature the author was lacking in this simulation is plotting the frequency and

time domain. Nevertheless, this issue can be solved. The frequency domain can be plotted with the MATLAB function `obw` with the input of signal and sampling frequency. The time domain is a plot of squared magnitude of the sample versus time (number of samples times sampling period). Plots of frequency and time domain of the LDACS signal are shown in figures 5.2 and 5.3 respectively.

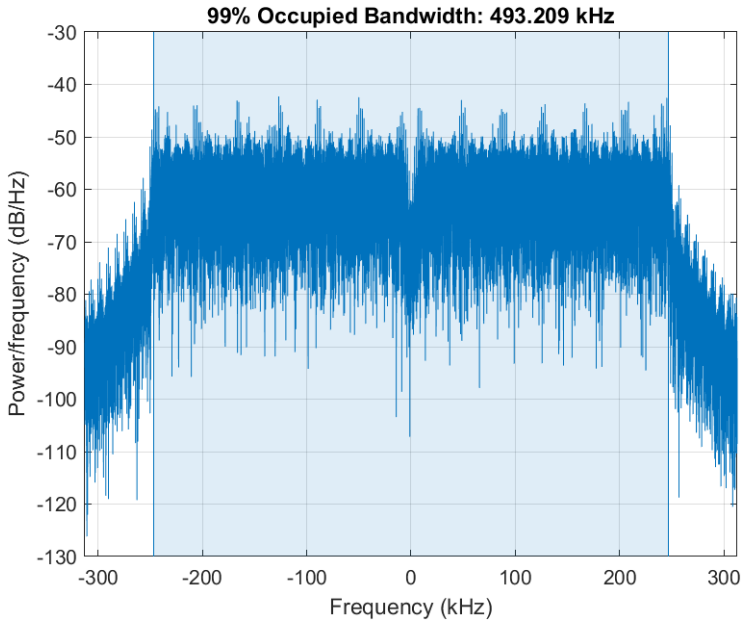

Figure 5.2: Frequency domain of LDACS signal

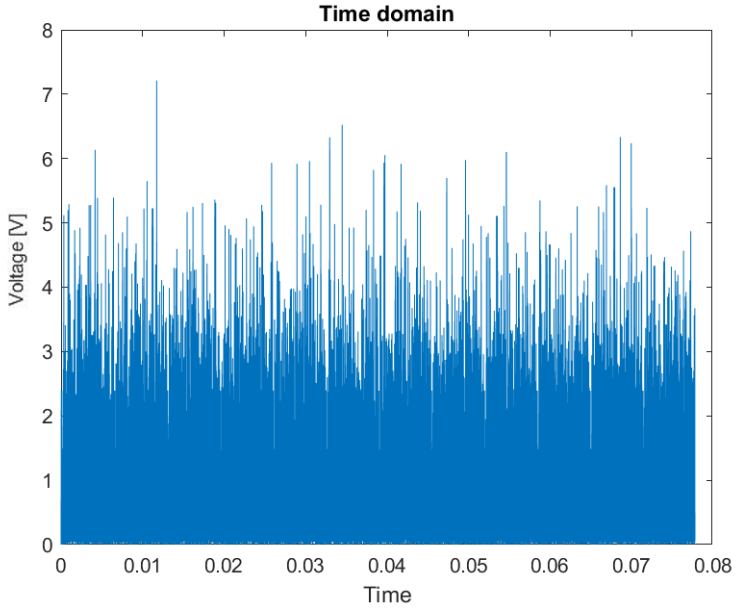

Figure 5.3: Time domain of LDACS signal

The time and frequency and time domain visually look as can be expected from the OFDM system.

5.2 Introducing the HW component to the MATLAB simulation

The main idea of the HW integration process into the LDACS test platform is first to set up the hardware equipment and after that change the MATLAB LDACS simulation software in order to use it to process the signal after SDR transmission. A simplified version of this process can be described with the following four steps:

1. SDR test platform is built
2. Transmission vector generated by a simulated LDACS transmitter is taken out of the simulation
3. Transmission vector is processed in order to be compatible with the SDR input standard
4. Transmission is established and the signal is received in another SDR
5. The received vector is saved and reshaped into the format that can be an input for the simulated LDACS receiver

Those steps are further presented in this chapter. Moreover, a detailed guideline on how to build such a platform is provided.

5.2.1 Hardware equipment

This section describes the ENAC SIGNAV laboratory equipment and also the interconnection of individual devices. SIGNAV is equiped with one NI USRP-2901, one Ettus Research USRP X310 and two Linux running computers. Both of these SDRs are high end solutions for the laboratory environment and their characteristics are more than enough for the purpose of LDACS signal transmission. The interconnection between these devices is in fact a one way loop with the following components:

- Linux driven PC on which the signal is prepared to be transmitted (further called as PC Alpha)

- USRP X310 - used as a signal transmitter (hereinafter referred to as called as USRP Alpha)
- USRP-2901 - used as a signal receiver (further called as USRP Beta)
- Linux driven PC in which the signal is then processed in order to obtain the results of the simulation

Interconnection between these types of hardware is ensured via different standards:

$$PCAlpha \xleftrightarrow{\text{Gigabit Ethernet}} USRPAlpha$$

$$USRPAlpha \xrightarrow{\text{SMA-M to SMA-M}} USRPBeta$$

$$USRPBeta \xleftrightarrow{\text{USB}} PCBeta$$

$$PCBeta \xleftrightarrow{\text{SMA-M to SMA-M}} PCAlpha$$

The structure described above is visualized in figure 5.4.

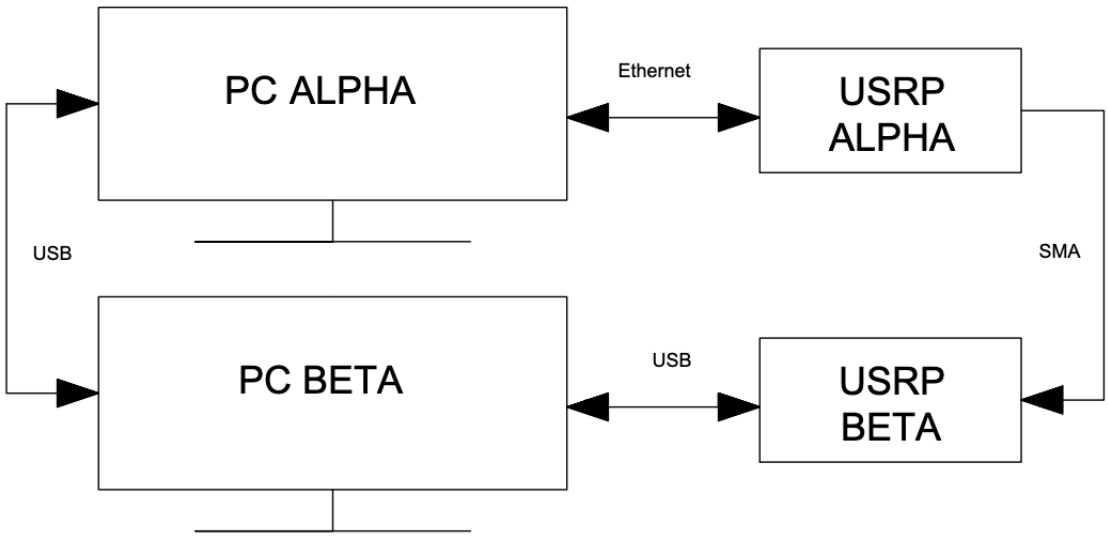

Figure 5.4: Hardware set up of the proposed LDACS platform

Figure 5.5: Photo of the proposed HW setup

After the platform was set, it was tested by sending a complex sine wave. The result of the test run were acceptable - the sine wave was seen from the time domain of the signal.

5.2.2 USRP-USRP Transmission Setup

This section aims to describe the method for USRP-USRP transmission. The described method uses a binary file. First, the generated signal is saved into it and then, USRP is made to transmit the data from it. The method is as follows:

- Transmission vector is ejected from the simulation after transmitter and saved into the binary file. Then the Workspace is saved.
- Transmission via USRP ALPHA
- Reception on USRP BETA
- Received signal is turned back into LDACS simulation format
- Received signal is cut and injected back into the simulation before the simulation channel. Workspace is loaded.

This method is proposed by head of SigNav laboratory of ENAC and also partially described in USRP documentation [56]. This method is supported by the fact that the binary files are the most efficient way to store data. On the other hand, this method influences the complexity of LDACS platform use. [42]

The USRP-USRP LDACS transmission method is also shown in a block scheme in figure 5.6. In the scheme below, the blue lines represent data before the transmission and the red

lines represent data after the transmission. The signal preparation for the transmission as well as the transmission itself are described in next subsections.

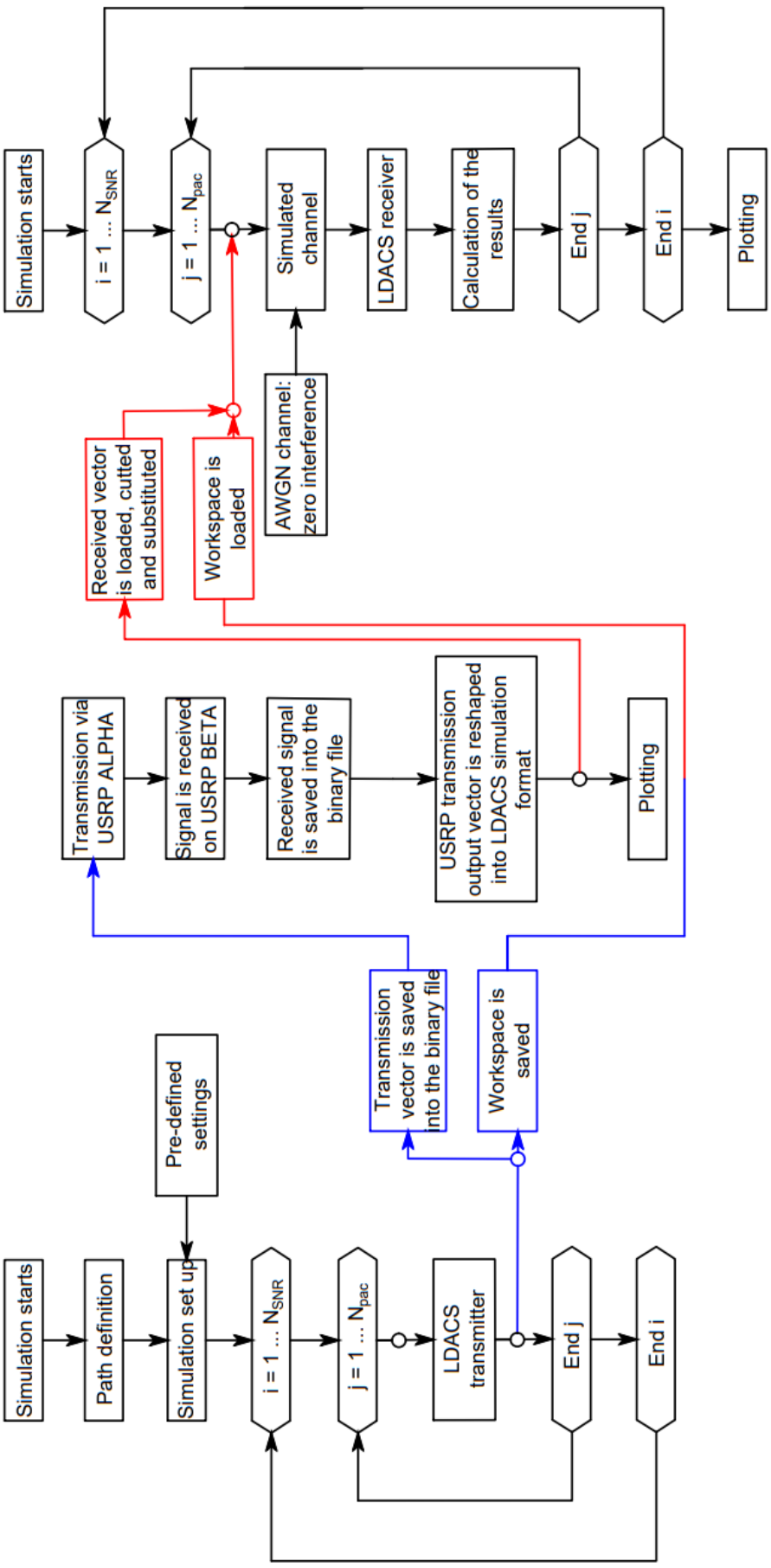

Figure 5.6: Block diagram of USRP-USRP transmission implementation into LDACS simulation.

Software simulation - preparation for the transmission

As mentioned in the previous section, the simulation must be updated with some attributes that will enable the signal to be sent to USRP. The first step is to create a .bin file with `file=fopen('data_for_usrp.bin','w');` (`'w'` defines that the data are going to be written into the file). The next step is to eject the transmission vector from the simulation and write it in the appropriate format into the binary file. The data must be reshaped into int16 format. The complex values of $X_k = I_k + iQ_k$ format shall be divided into two values where the real part is followed by the imaginary (IQ format). The data generated by the LDACS simulation are double values in the interval of [-5;5]. Int16 format assumes whole values in the interval of [-32768 ; 32767]. Considering the mentioned fact, the signal has to be scaled, otherwise the LDACS signal values would be just rounded. The scaling factor is determined as a higher bound of int16 interval divided by the highest value of the signal (scaling factor for real and imaginary parts are calculated separately). Then, the signal can be converted into int16 format using MATLAB function. Once the data are shaped into the transmission-required format they can be written into the binary file.

The last step before the physical USRP transmission is to save the workspace for each iteration. Since two PCs are used, the workspaces are needed to compare the received and transmitted data to measure the performance indicators (BER and PER).

Also, in order to have accurate performance indicators the additional interference has to be removed. For the USRP transmission the AWGN channel is chosen.

USRP-USRP transmission

Once the data are saved into the binary file, it can be transmitted with USRP. Considering that USRP is installed on a Linux-run PC, it can be called for transmission with the following command[3]:

sudo ./init_usrp –args=”type=x300,addr=192.168.40.2,second_adddr=192.168.30.2, enable_tx_dual_eth=1,skip_dram=1” –rate 1.6e-06 –wirefmt=”sc16” –freq 1156e6 –type=”short” –subdev=”B:0” –gain=0 –ref=’gpsdo’

In this command, the most important parts are:

- –args - Initialization of the USRP

[3]The command is written according to USRP X310 manual [57]. The command must be run from the file directory.

- –rate - Sampling rate of the signal
- –freq - Center frequency of the transmission

With this command the transmission is established. Since the L-band is a secured band (unauthorized transmission is prohibited) and a permission is needed to perform the transmission in it, the signal is transmitted via a wired SMA cable as shown in figure 5.4, 5.5.

The reception of the signal on the software level on USRP BETA is made with GNU Radio, where the signal is received and written into the file with a File Sink (block from the GNU Radio library, it write the signal into the binary file). The structure and setup of the GNU Radio block diagram is shown in figure 5.7

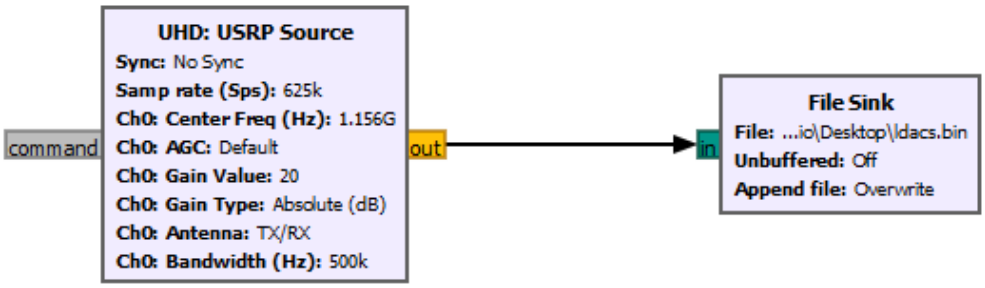

Figure 5.7: Structure and setting of the GNU Radio block diagram for signal reception.

The data output of the GNU Radio is in the form of a binary file and has the same format as the data reshaped during the transmission preparation, which means that the real part is followed by the imaginary part without the i. In order to put the data into the MATLAB LDACS receiver, the said data must be formed into the same format as the original transmission vector ($X_k = I_k + iQ_k$).

Cutting and injection of received signal into the simulation

The receiver and the transmitter are turned on simultaneously. Then, USRP BETA receive only noise before the moment USRP ALPHA starts to actually transmit data (it takes some milliseconds to initialize the input signal and start the transmission). Also, after all the frames are transmitted, USRP BETA receives only noise until its turned off. Summarising the above, the signal has to be properly cut in order to insert it back into the LDACS simulation. The structure of the received signal is shown in figure 5.8.

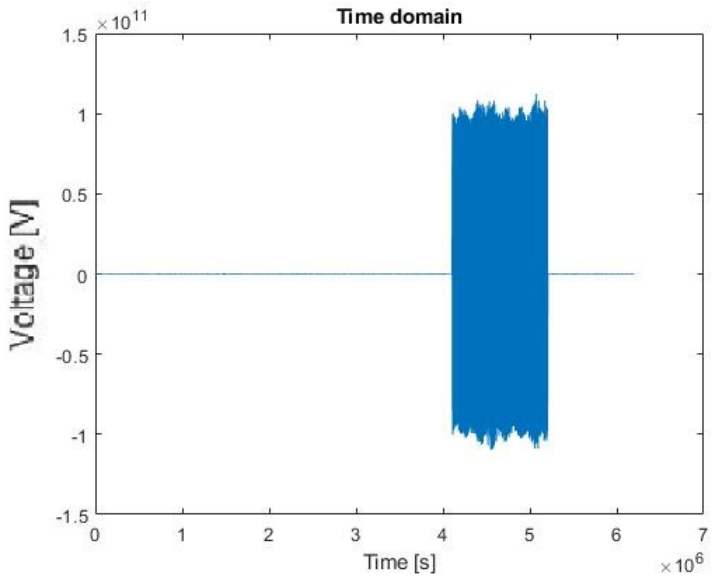

Figure 5.8: Time domain of the received signal before cutting.

The LDACS receiver is made to take only a certain number of values - 48600 samples - 1 packet for the after-reception signal process. The start of the signal has to be located first, and then there is a need to cut the signal received on USRP. The cutting is made as a floating window, where once there is a value that is greater than the borderline of the noise (defined according to plots of time domain for multiple runs), the signal is cut as: 'received signal = first symbol above borderline : first symbol above borderline + 48599' for single frame and 'received signal = first symbol above borderline + (number of packets - 1) * 48600: first symbol above borderline + 48600 * number of packets - 1' in case multiple frames are transmitted.

Then, the received signal can be loaded into the MATLAB LDACS simulation receiver. This is done by substituting the vector that enters the simulated channel by the packet received. Also the workspaces, that were saved during the transmission preparation, are loaded into the simulation in order to measure the signal transmission performance. Then, the simulation runs normally to the end until the performance indicators are displayed.

5.3 USRP-USRP LDACS transmission results

The result of the actual transmission is very poor. BER after decoding process is close to 0.5. This can be interpreted as the probability of a bit having the correct value of 50% after decoding. The frequency domain of the signal is visually correct and matches expectation of

OFDM signal with some AWGN noise, figure 5.9. But out of the time domain plot (figure 5.10) it can be seen that the sample values are muffled.

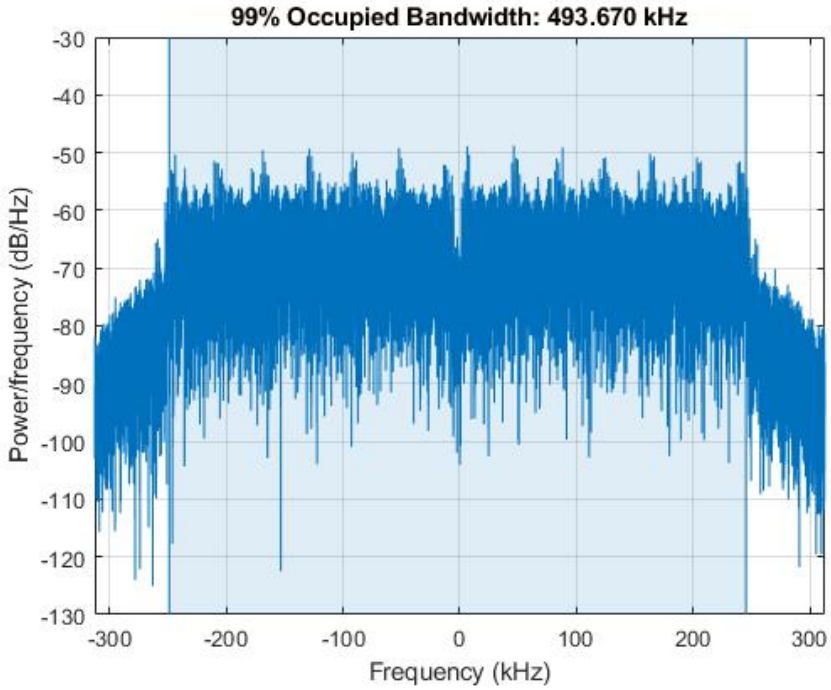

Figure 5.9: Frequency domain of the received signal after USRP-USRP transmission

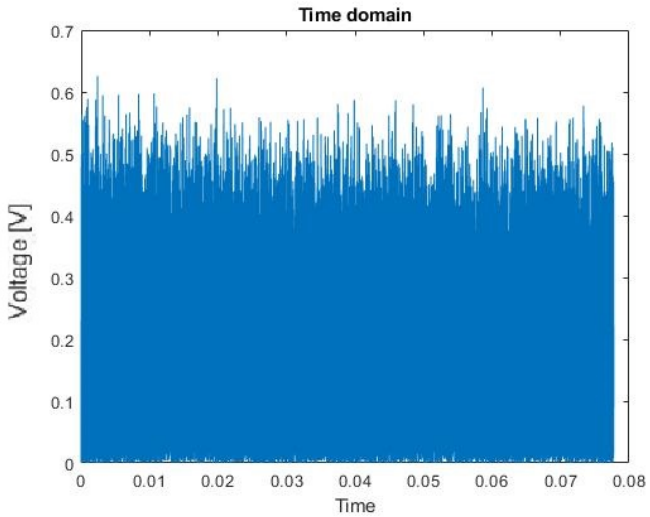

Figure 5.10: Time domain of the received signal after USRP-USRP transmission

This issue should have been solved by the LDACS equalizer, that is a part of the simulation, nevertheless this issue can be solved by normalization of data as well. The scaling factor for the received data is the same scaling factor that was used for signal downscaling

during the signal process before transmission.

Running the receiver with normalized data still gave an unacceptable result. BER of the transmission is still equal to 0.5. This result may be caused by lack of computational performance of HW or insufficient throughput rate of components that interconnect the hardware components. For this run 64QAM modulation technique was used. It has a very high bitrate. Due to this fact, it can be supposed, that one or multiple components that interconnect HW components or the components as such caused data loss due to insufficient performance. This component might have been the USB cable that was connecting USRP BETA with PC BETA since it is the "slowest" component in this installation.

The unacceptable result also may be caused by the author's mistake made during the hardware setup. Unfortunately, the author didn't have enough time in the laboratory in Toulouse so it was not possible to identify and correct this issue or try out another way of transmission. Because of that, there is a need to find the equipment for signal transmission elsewhere.

5.4 ADALM-Pluto hardware for Signal Transmission

The author of the present thesis got in touch with the department of air transport of the faculty of transportation sciences in CTU with a request to provide HW. The laboratory of the CNS systems provided the ADALM-Pluto SDR system. This HW solution has already been described in chapter 4.

5.4.1 ADALM-Pluto installation and transmission setup

Since the previous way for transmission proved to be unsuccessful, there is a need to try another way. After the research in the field of Software Defined Radio use, the following option was found:

- Reroute the ready-for-transmission vector from the LDACS simulated channel to the ADALM-Pluto transmitter on Simulink.
- Transmit the signal with ADALM-Pluto using the Simulink ADALM-Pluto transmitter
- Recieve the signal with ADALM-Pluto using the Simulink ADALM-Pluto receiver
- Cut the signal
- Inject the received signal into the simulation before the simulated channel

The signal process as well as the signal ejection, transmission and insertion are described as a block diagram in figure 5.11.

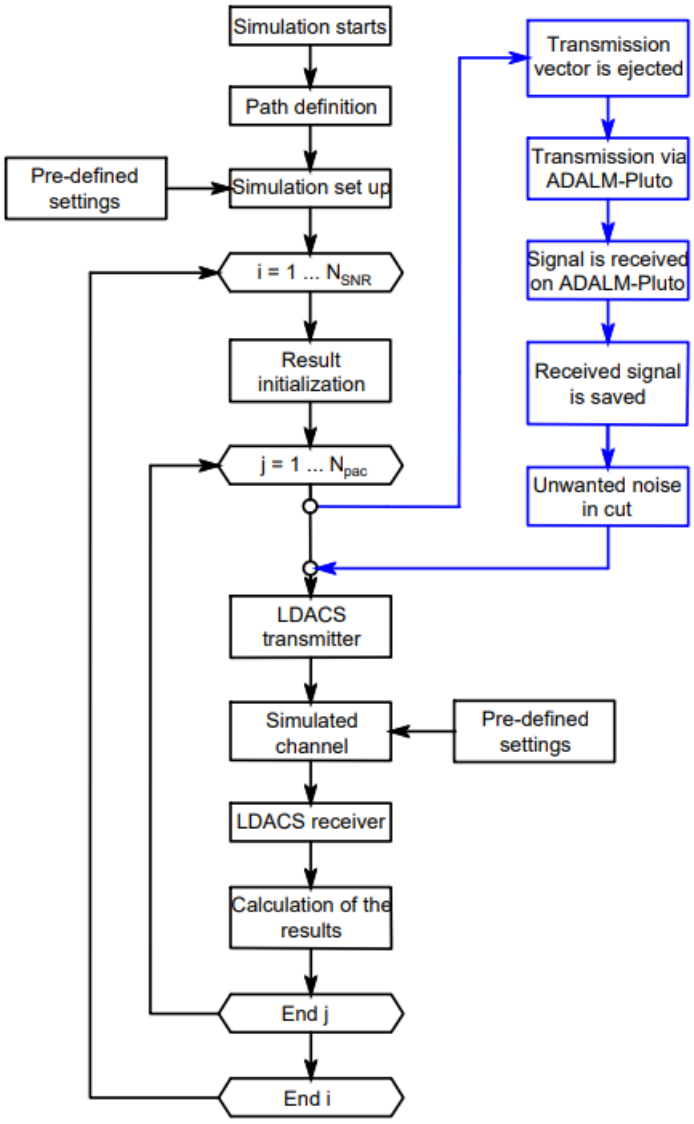

Figure 5.11: Block diagram of ADALM-Pluto transmission implementation into LDACS simulation.

This method simplifies the experiment, because there is no need for other software than MATLAB. Also, the hardware platform itself is less complicated and less expensive. On the other hand, the transmission from the binary file is a more efficient way. This fact may play a role in case of transmission of a large number of frames. Moreover, ADALM-Pluto has the DAC/ADC chip with lower resolution which leads to lower accuracy of digital to analog/analog to digital conversion which may cause worse transmission performance.

The experiment setup of the LDACS signal HW transmission is shown in figure 5.12

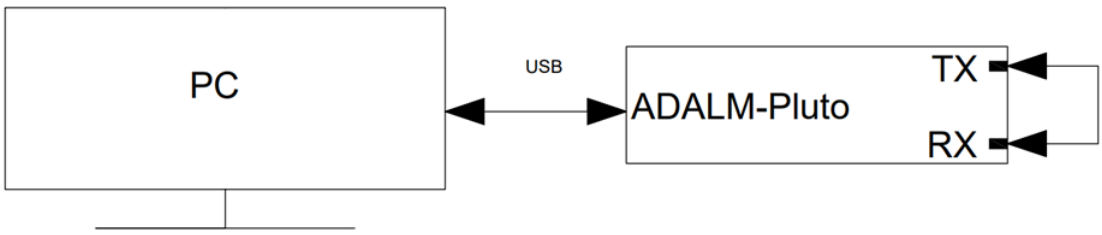

Figure 5.12: Interconnection of ADALM-Pluto RX and TX modules and host PC.

The interconnection of the devices on the software level is made with the help of the MATLAB Add-on - "Communications Toolbox Support packet for Analog Devices ADALM-Pluto Radio" that is a part of MATLAB Add-on library. For installation there is a need to supplement the add-on from the MATLAB Add-on library and click out to the end of the installation[4].

With installed add-ons the transmission can be established by means of Simulink blocks: the "ADALM-Pluto receiver" and the "ADALM-Pluto transmitter". However, before the transmission, these blocks have to be properly set according to the LDACS signal parameters.

5.4.2 Test transmission with ADALM-Pluto

In order to test the functionality of the platform, there is a need to test it with a less sophisticated signal. As a test signal $cos(t) + j * sin(t)$ complex vector of values is generated, where t is 48600 samples from 0 up to π (in order to achieve same number of samples as LDACS frame has). The received signal can be easily visually compared with the transmitted one and some conclusions about the platform's performance can be made. The plot of this signal on complex plane is shown in figure 5.13

[4]In case of other firmware than 0.31 MATLAB would ask you reinstall the firmware.

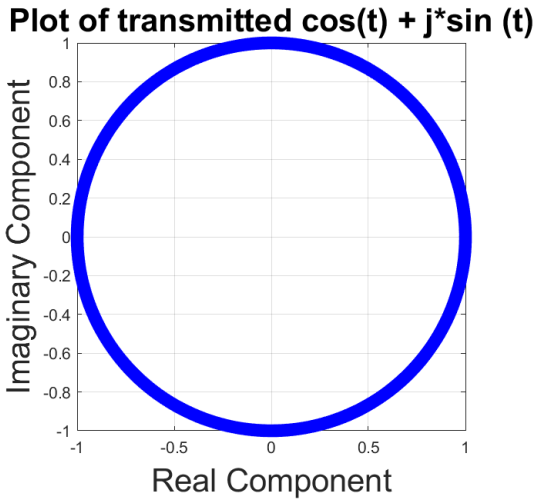

Figure 5.13: Plot of $cos(t) + j * sin(t)$ function for test transmission.

This sampled $cos(t) + j * sin(t)$ function is then injected into the Simulink with the "Signal from Workspace" block, that is connected with the ADALM-Pluto transmitter. The transmitter and the receiver are set in the same way as shown in figure 5.15, where baseband sample rate is equal to $1/T_{sa}$, where T_{sa} is a sampling period of LDACS that is equal to 1.6e-6 seconds. Then, connecting "To Workspace" Simulink block to the receiver will allow to save the received vector to the MATLAB. This simulink model[5] is shown in figure 5.14. The configuration of the transmitter and receiver is shown in figure 5.15.

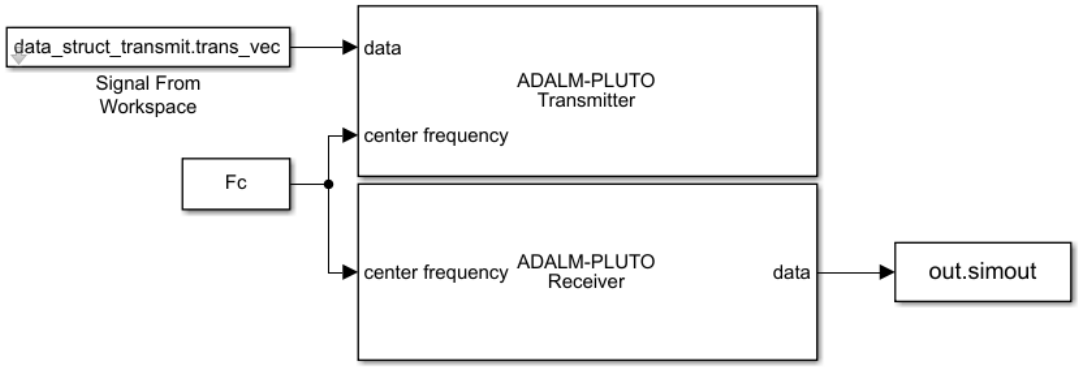

Figure 5.14: ADALM-Pluto transmitter and receiver Simulink model. `data_struct_transmit.trans_vec` is a signal for the transmission, `Fc` is center frequency.

[5]The Simulink model for ADALM-Pluto described in this chapter is provided in form of electronic appendix

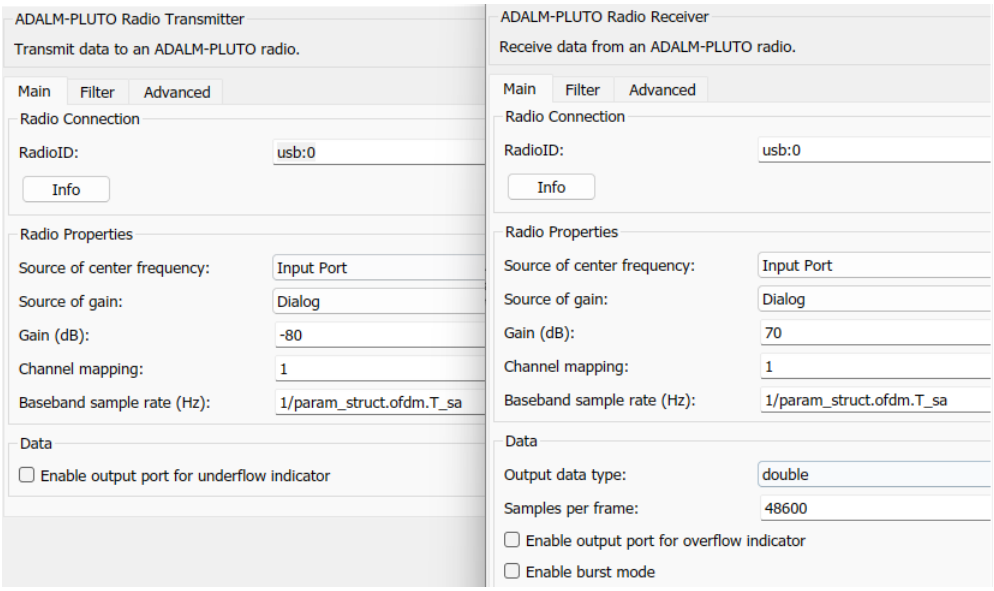

Figure 5.15: ADALM-Pluto transmitter and receiver Simulink blocks setup.

ADALM-Pluto offers a possibility of a configuration test with digital transmission from transmitter to receiver. To enable that "Digital TX to Digital RX" option of Loopback is set into the Advanced tab in the ADALM-Pluto transmitter and receiver block parameters. In this case the received vector must be the same as the transmitted one.

Having set the simulation time (2 seconds is enough) and run the simulation, one can assess the received vector after the simulation is over. When plotting both - the transmitted and the received vector, it is clear that the received signal is similar to the transmitted one, figure 5.16

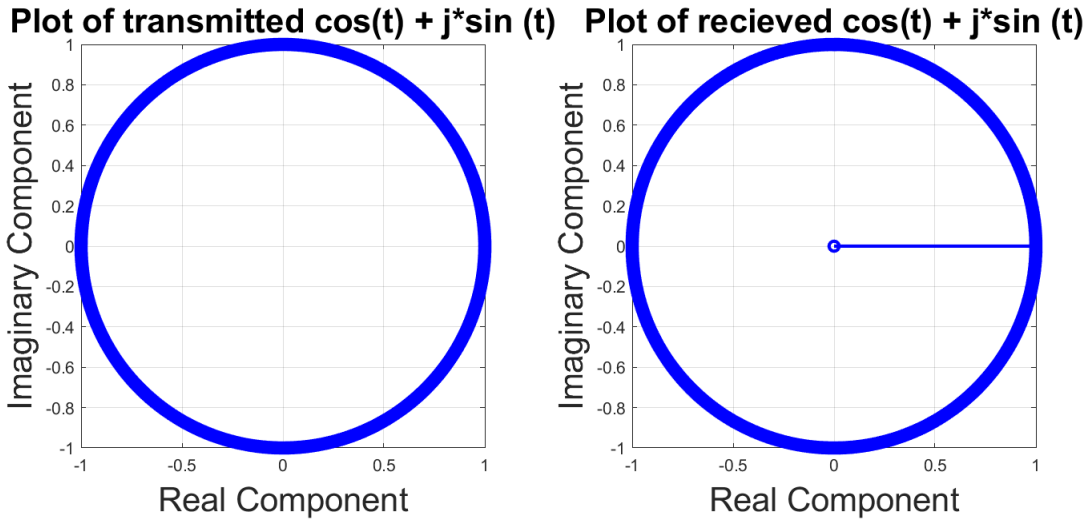

Figure 5.16: Plot of $cos(t) + j * sin(t)$ function before transmission (on the left) and received function through "Digital TX to Digital RX" (on right).

In case of the transmission by the SMA cable ("Default" is set in Loopback), the results of the transmission are less accurate. Out of the plot (figure 5.17) the amplitude of the received signal is noticeably lower than the amplitude of the transmitted one. This issue can be solved by applying the scaling factor to the signal. Also, there is a noticeable level of noise, which is quite high (the signal transmission is made by cable). In the end, the form of the received function is a bit corrupted, which may be caused by the HW bitrate non-linearity.

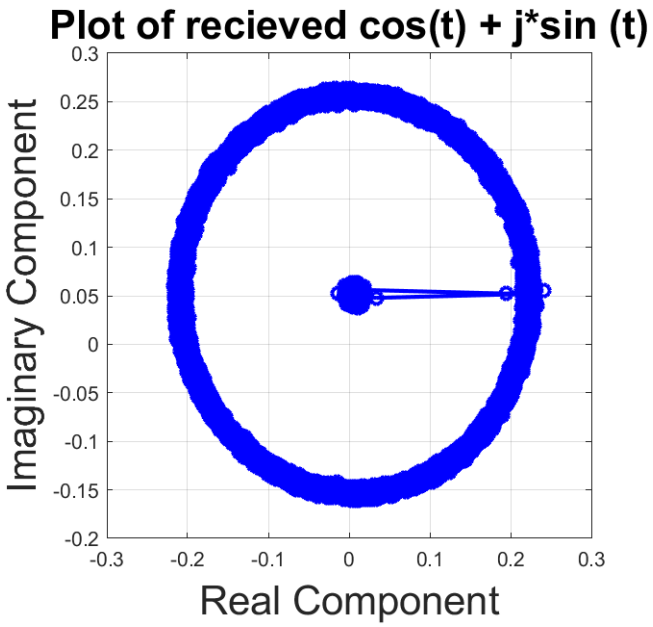

Figure 5.17: Plot of received $cos(t) + j * sin(t)$ function using RF transmission.

Also, the time and frequency domains of the signal prove the assumptions made above, figure 5.21, figure 5.20.

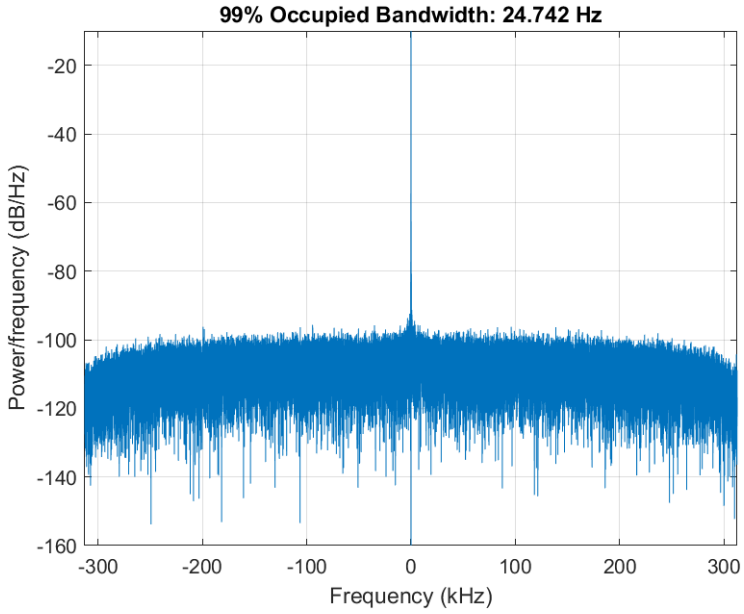

Figure 5.18: Frequency domain of received complex function.

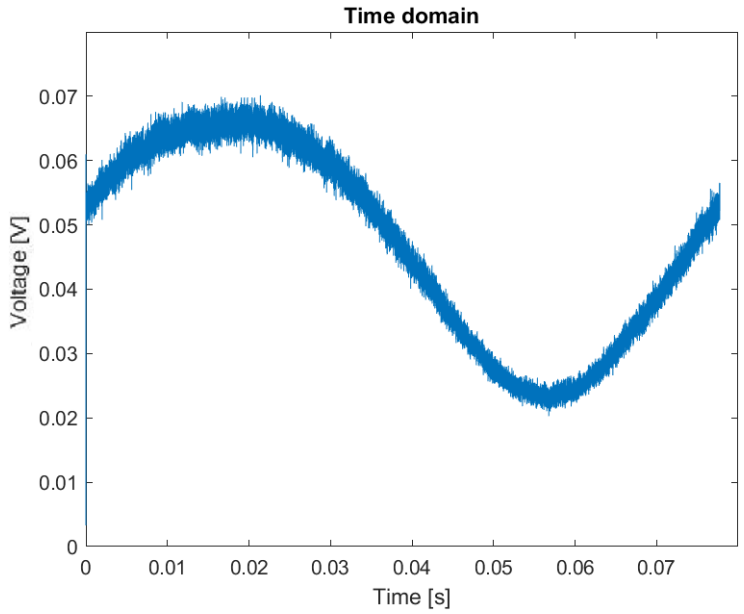

Figure 5.19: Time domain of received complex function.

From the frequency domain as well as from the plot of received samples, it can be seen that the signal is attenuated and also corrupted. This issue was also dealt with in[31, 33] and it can be considered as a hardware limitation. Nevertheless, it can be overcome with data normalization.

5.4.3 LDACS signal transmission with ADALM-Pluto

Once the hardware is prepared for the transmission and the platform is tested, there is a need to prepare and reroute the transmission vector from the LDACS MATLAB simulation to Simulink. The final vector which is generated by the simulation (`data_struct_transmit. ... trans_vec`) is further injected into Simulink in the same way as it was described in the previous subsection. Since the test signal has the same parameters as the LDACS signal, there is no need to change anything in the transmitter and receiver blocks compared to the settings presented in the previous subsection.

Test transmission of the LDACS signal using ADALM-Pluto

For the first run the LDACS signal is sent digitally from TX to RX bypassing the RF part of the HW. The time and frequency domains of the received LDACS signal are represented in figure 5.20, 5.21

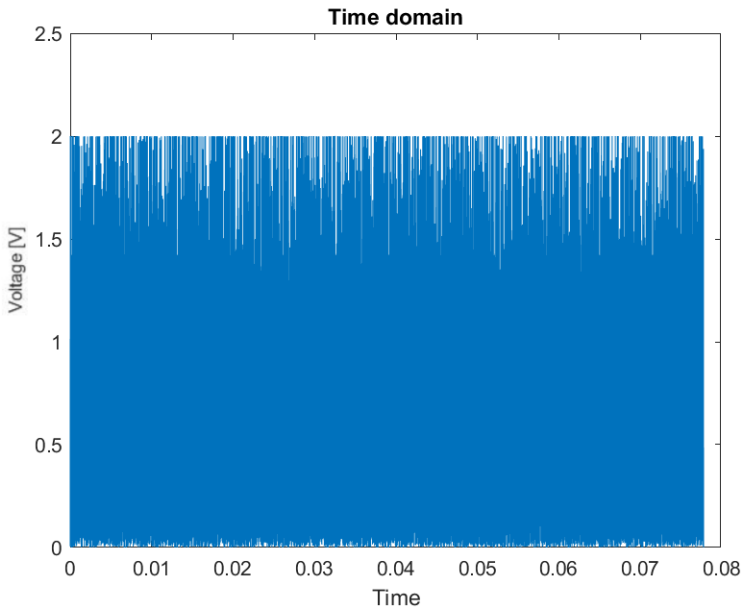

Figure 5.20: Time domain of received LDACS signal (bypassing RF part).

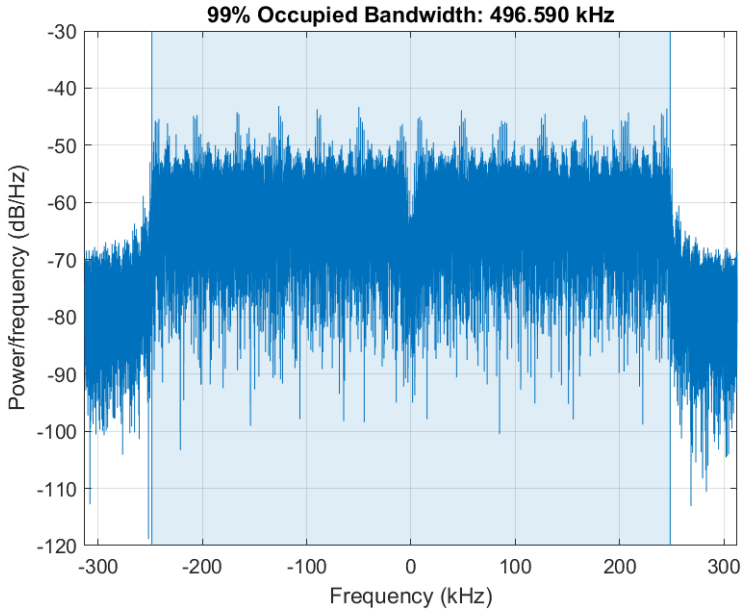

Figure 5.21: Frequency domain of received LDACS signal (bypassing RF part).

Out of the time and frequency domain plots, it is clear that all the values greater than 1 or lower than -1[6] are strongly corrupted. In fact, they are cut. The explanation for that is ADALM-Pluto limitations for signal transmission:

"Double-precision floating point with values scaled to the range of [–1,1]." [43]

Besides that, there is the second limitation of ADALM-Pluto:

"The AD936X RF chip has a 12-bit DAC. Only the 12 most significant bits of the transmitted data are used. Values of magnitude less than 0.0625 are lost."[43]

Table 5.4.3 represents the percentage of the samples generated by LDACS transmitter that were affected by the HW limitaions.

[6]Time domain represent squared magnitude of the signal versus time. The magnitude of complex number $(X_k = I_k + iQ_k)$ is calculated as $|X_k| = \sqrt{I_k^2 + Q_k^2}$

Table 5.3: Overveiw of availible settings for `set_parameters_fl/rl`

Magnitude test

Run	Total number of frames	Samples with magnitude higher than 0.0625	Percentage
1	48600	279	$\approx 0.6\%$
2		318	$\approx 0.7\%$
3		304	$\approx 0.6\%$

Interval test

Run	Total number of samples	Samples outside [-1;1] interval	Percentage
1	48600	4973	$\approx 10\%$
2		4999	$\approx 10\%$
3		4946	$\approx 10\%$

This issue can be hardly overcome, since in case of signal normalisation before the transmission, for example dividing the LDACS transmission vector by 3 (solving the interval limitation), the number of samples with magnitude lower than 0.0625 will increase as well. This problem can be solved without signal scaling by using more performaced HW with a higher resolution DAC chip. Nevertheless, the option of signal downscale should be considered, since the number of samples lower than 0.0625 would not increase by the same amount as the samples outside the interval [-1;1] would decrease.

This specific of the ADALM-Pluto also can be ignored. Since 16QAM and 64QAM modulations are amplitude modulation techniques, they are very sensible to changes in amplitude. To overcome this, the QPSK modulation can be used instead to achieve the result. QPSK is a phase modulation then, it would have noticeably lower error rate in case of signal amplitude cut compared to 16QAM and 64QAM. This modulation is also less demanding to the computational and throughput performance of hardware.

In order to define a better way, both of the options described above were tested for AWGN channel with no interference. From several simulation runs it was clear, that, ignoring the fact that the signal is cut, the approach of the QPSK modulation, is a better option in terms of performance indicators results.

LDACS signal transmission

The transmission of the LDACS signal is the same as for the test signal. From the time domain of the received signal (figure 5.23) it can be seen noticeable signal attenuation, but

otherwise the signal still remains the OFDM waveform. Injecting of this vector into the simulation would give an unacceptable result, as it was in case of transmission with USRPs.

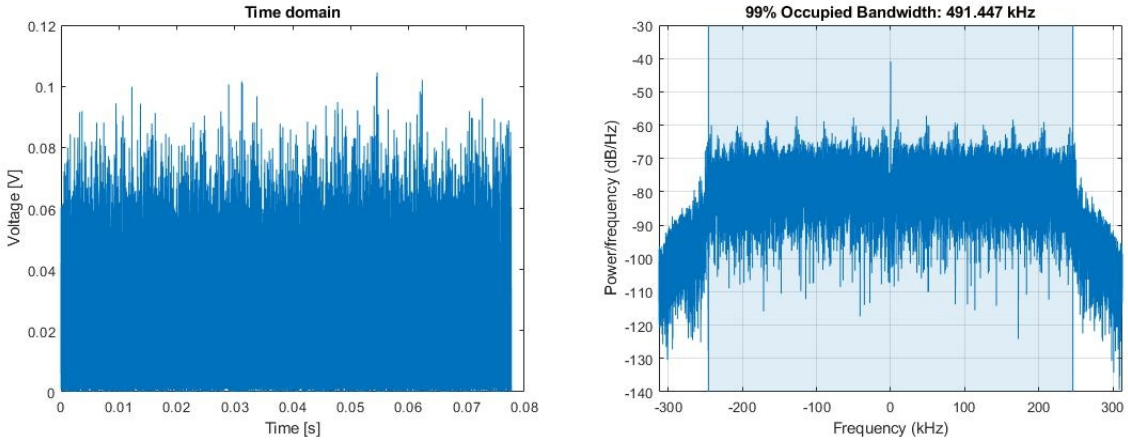

Figure 5.22: Time domain of received LDACS signal on the left, frequency domain of received LDACS signal on the right.

The attenuation issue has already been described in the previous sub-sections. To solve the mentioned issue, the received signal can be normalized or up-scaled with some scaling factor. This scaling factor is calculated as a ratio of the highest amplitude of the transmitted and received signal. This scaling factor takes value around 8 (calculated out of several runs of the simulation). After the normalization, the time and frequency domain are visually correct, figure 5.22.

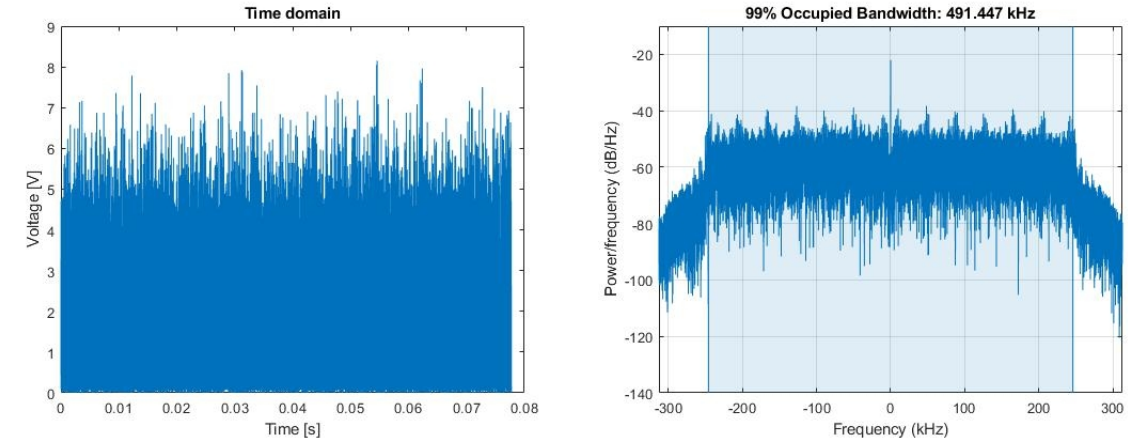

Figure 5.23: Time domain of normalized LDACS signal on the left, frequency domain of normalized LDACS signal on the right.

Post transmission signal process

Once the signal is received and assigned to the variable, it is possible to substitute the original LDACS transmission vector. This has to be done into the main file of the LDACS simulation after the transmitter function. The transmission output vector is injected into the simulation before the simulated channel. This is done in order to leave the option of signal noise and interference adding. The simulation can also be modified in order to run the Simulink model, cutting and plotting scripts from the main file.[7]

Result evaluation - transmission with ADALM-Pluto

Even considering certain HW limitations and the test transmission using Digital TX to Digital RX, the results are very good. The result is in compliance with the demanded BER value of 10e-6 - BER after CC as well as 0 after RS decoding of the transmission. This is true, because in case of the QPSK modulation, even in case of the amplitude loss for some samples, the information is not lost because QPSK is a phase modulation.

The result of real transmission through the RF module with the AWGN channel and SNR of 6 is: BER after CC - 0.0017189, BER after RS - 0. This result is influenced by the type of RF transmission. In this case, it is corrupted by the cable, namely the resistivity of the cable that caused amplitude attenuation and loss of some information. Still, the said result is in compliance with the specification (BER lower than 10e-06). This result is achieved with the AWGN channel model and SNR of 6.

The full MATLAB simulation version with all the supplements described above is available in the electronic appendix.

[7] Modified main file and functions for plotting, cutting and Simulink model run can be found in electronic appendix. `main_file.m` - main file, `tnf_plot.m` - time and frequency domain plotting, `sim_run.m` - Simulink model run and signal cutting

Chapter 6

Testing scenarios

The proposed LDACS testing platform might be used for the different testing of the LDACS system. This may include a wide variety of testing scenarios. This chapter describe several proposed sceanrios that can be made with the proposed platform.

In the scenarios the certain conditions of the channel are proposed. With the certain scenario the performance of the system under certain conditions is measured. In this thesis the proposed scenarios for the testing of the LDACS system are following:

- Signal to Noise Ratio value change
- Different modulation techniques
- Simulated channel model application
- Interference with other system

Since platform proposition was successful and performance indicators are on zero level, the scenarios proposed in this chapter are defined by LDACS simulation settings. Also one of proposed testing scenarios contain suggestion for the extended version of the platform.

6.1 Scenario 1: Signal to Noise Ratio

Signal to noise is basic indicator of signal strength compared to the environment conditions. For the simulation purpose, in order to track the changes applying interference, SNR has to be set in the way where BER is very close, but not equal to zero. Even in case the signal is sent using real HW on the proposed LDACS test platform, SNR still influences the signal since the simulation is not featured with a zero interference channel.

Signal to Noise Ration can be set by changing `sim_param.SNR_vec_dB` variable in `set_parameters_fl` for forward link and in `set_parameters_rl` for reverse link.

In real environment SNR value of the signal depends on the distance between the transmitter and the receiver. The recommended value of SNR starts from 20dB [11]. The default value of SNR in the LDACS simulation is 6 dB. On this level BER and PER are on zero after the RS decoding assumed AWNG simulated channel and no HW. In figure 6.1 BER and PER results with different values of SNR with HW and QPSK modulation and without them are shown.

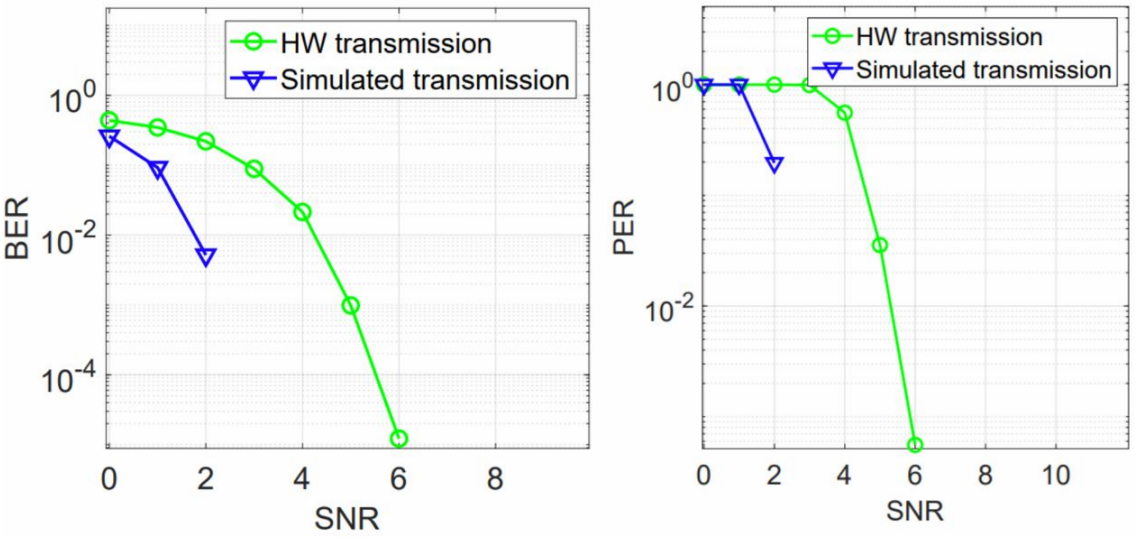

Figure 6.1: BER and PER for different transmission option versus SNR

From the plot above it can be seen that in case of HW use for signal transmission, BER and PER have higher values. This is caused by the ADALM-Pluto limitations described in the previous chapter and by the fact that HW also adds some AWGN noise. Nevertheless, the BER result is more than acceptable - BER after RS decoding is equal to zero on SNR 7 dB.

6.2 Scenario 2: Different modulation techniques

The second scenario proposes a change in the modulation technique. LDACS supports 3 modulation techniques - QPSK, 16QAM and 64QAM. The constellation diagrams for these modulation as well as the principle itself are described in chapter 2. Different modulation techniques can support different bitrate, since different number of bits are modulated into one symbol. Nevertheless, higher modulation techniques, ex. 64QAM, require higher preci-

sion precision to enable acceptable BER and PER. The BER and PER results for QPSK, 16QAM and 64QAM using ADALM-Pluto for signal transmission and AWGN channel with no interference are shown in figure 6.2 for BER and in figure 6.3 for PER.

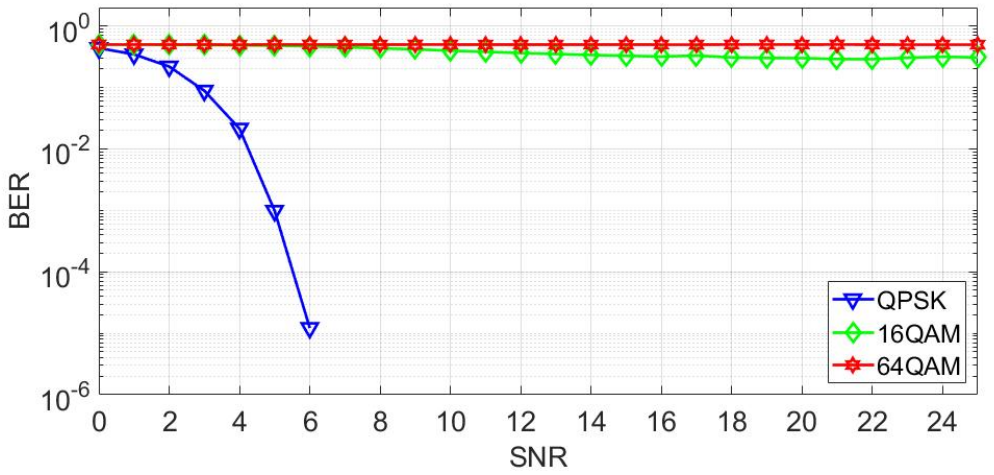

Figure 6.2: BER of different modulation techniques changes depending on SNR value.

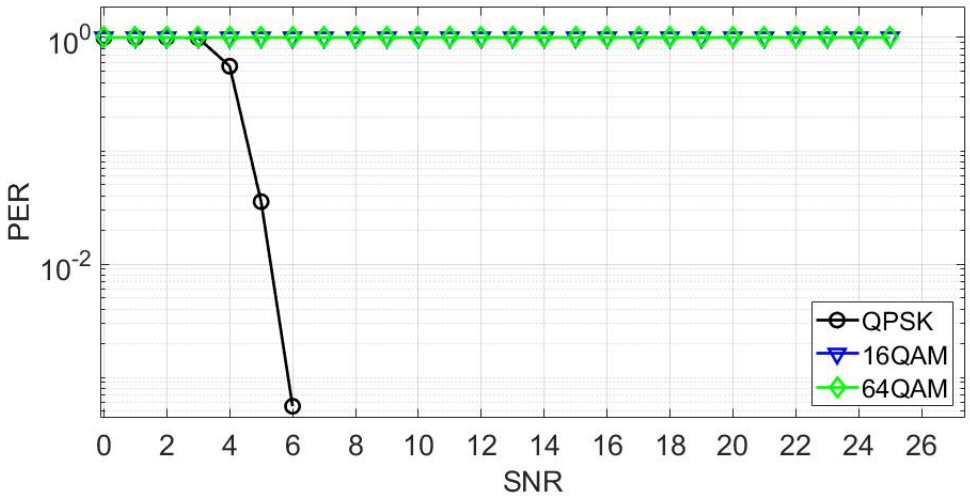

Figure 6.3: PER of different modulation techniques depending on SNR value.

From the figures above, certain performance loss can be noted in case of using 16QAM and 64QAM modulation techniques. This result is expected due to the transmission approach, where the amplitude cutting by HW is ignored. Nevertheless, this can be fixed using HW with a higher resolution DAC chip.

6.3 Scenario 3: Simulated Channel Model

The third considered scenario is the application of a diffident simulated channel. There are 3 channel models that represent real transmission models - ENR, TMA and APT. These channels have already been mentioned above in section 5.2.2. They can be supplicated during the simulation setup by changing `param_struct.chan.channel` variable in function `set_parameters_fl` for forward link transmission or in function `set_parameters_rl` for reverse link. The functionality and interference parameters of these functions are described in the following subsection.

- En-route channel - The en-route conditions may be described as high ground speed and high altitude of the aircraft. The en-route parameters proposed by the LDACS developers include such parameters as Doppler spread, Rician factor, path delay and number of harmonics of the signal. According to the proposed parameter values in MATLAB simulation, the en-route channel is the least challenging for the LDACS system.

- Terminal channel - The terminal channel case is faced when the aircraft is in phases of take off or climb and also during the approach and landing. The air speed and the altitude of the aircraft in these phases is distinctly lower compared to the en-route case. Nevertheless, the multi-path effect is higher. The characteristics of this channel case are more demanding. The Doppler effect and the number of harmonics are still taking place with lower values, but the Rician factor is noticeably higher.

- Airport channel - The proposed airport channel is the most demanding among the others. The ground speed and the altitude conditions are lower, but on the other hand, the multi-path effect of the real airport channel is way stronger.

The changing of the channel setting is made in `channel_parameters.m` file, which contains the `channel_parameter` function. After the setup is done and the simulated channel is either chosen or chosen and redefined, the simulation can be run in a normal way. The simulation ends with the calculation of the performance indicators, which in case of En-route, Terminal and APT channel may drastically differ.

The results of BER under channel interference conditions are represented in figure 6.5

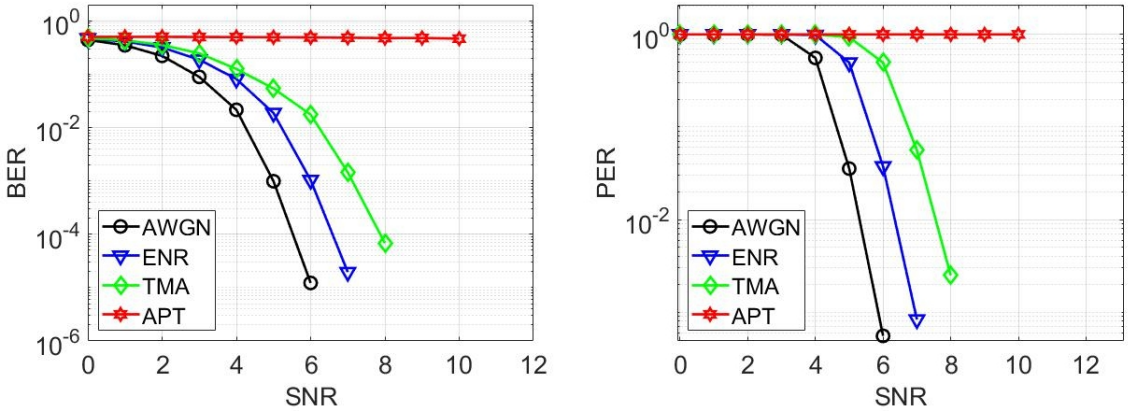

Figure 6.4: BER and PER of different channel models changes depending on SNR value.

From the plot of BER and PER versus SNR, it can be seen that most assumptions of the results described above have been confirmed. But the APT channel results are way worse than expected. Even if the simulation is set to SNR 30, the BER is around 50% and PER is equal to 1.

Taking a closer look at the simulated channel definition on `channel_parameters.m` function, it can be seen that the setting that differs from other channel settings (ENR and TMA) the most is Rician factor. It is equal to -100 compared to 10 and 15 for TMA and ENR channel respectively. According to paper "Analytical Calculation of Rician K-factor for Indoor Wireless Channel Models" there are no upper or lower bounds for the Rician factor, other than this factor cannot be negative. This leads to the assumption that the Rician factor is set incorrectly in the simulation.

The Rician factor characterizes multi-path fading channels in the real environment. Then, it can be considered that the ratio of signal power over the scattered power would be lower on APT channel compared to ENR and TMA channels. This is because in the airport environment there are more obstacles located closer to the receiver that may reflect the signal are theoretically. Then, based on the author's estimations, the Rician factor for the APT channel can be equal to 7,5[1]. Changing the Rician factor to the estimated value in the simulation, the results of BER and PER versus different SNR values are shown in figure 6.5

[1]The Rician factor may differ depending on individual case. Terrain and the buildings around the airport play a big role.

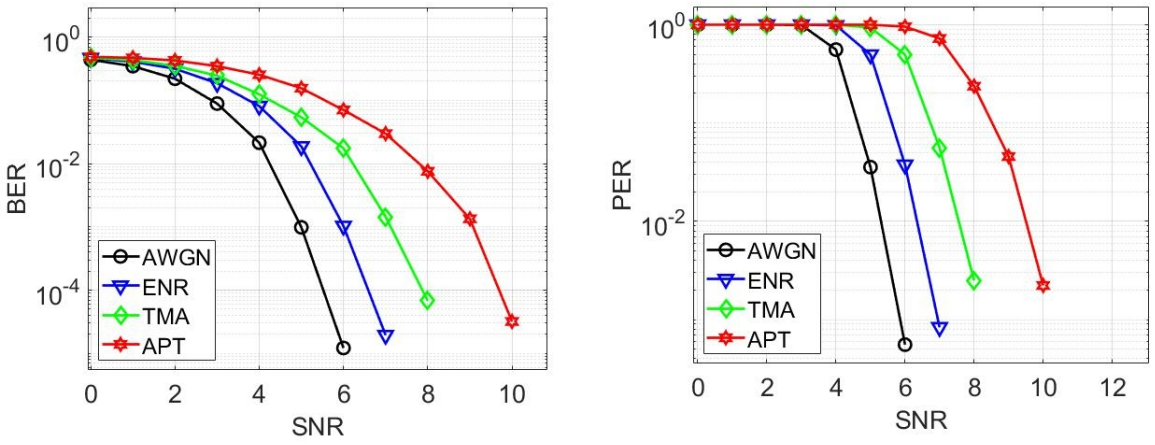

Figure 6.5: BER and PER of different channel models changes depending on SNR value (after APT channel model change).

After changing the Rician factor, the transmission results over APT channel look as they could be expected from the real airport channel.

6.4 Scenario 4: Interference caused by other system

In this scenario interference with one or more CNS system is proposed. Possible interference with DME has been already described in chapter 1. Nevertheless, DME is not the only system that operates in L-band and could possibly interfere with LDACS. These systems include Secondary surveillance radar (SSR) Mode S and GNSS. Even on the operation channel these systems are around 70 MHz apart of the LDACS, interference scenario is considered by ICAO [37]. Assumption of interference with GNSS is based on consultation with head of SigNav laboratory in Toulouse. The DME interference has already been studied in several papers. The output of these studies and sometimes software simulations or even sometimes testing were that the DME cause low impact on the LDACS signal. The interference of LDACS and other systems in L-band is taken in two ways. First, if these systems did not corrupt the LDACS signal, and second if LDACS did not corrupt the signal of other critical CNS systems.[1]

In order to test the interference with other systems in the L-band proposed testing platform should be extended. There is a need to add another HW piece. The additional HW can be used to transmit DME/SSR Mode S/GNSS signal into the LDACS reception HW (similar structure was proposed in [51]). Schematic HW interconnection is represented in

figure 6.6

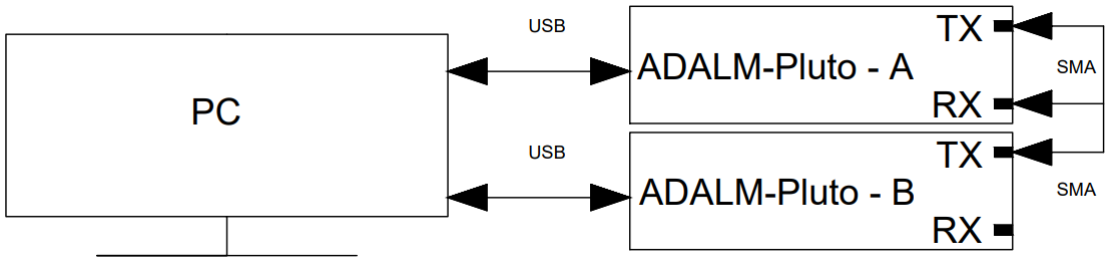

Figure 6.6: Interconnection of ADALM-Pluto RX and TX modules and host PC.

After the platform extension, DME/SSR Mode S/GNSS signal has to be defined in MATLAB and transmitted in the same way as the LDACS signal. As an example of signal creation, the DME signal can be defined according to [75]. Also, since LDACS MATLAB simulation offers a possibility of DME interference on a simulated channel can be either taken from the LDACS simulation (`create_dme` function).

This approach offer more realistic channel interference with other systems in the L-band. Furthermore, the DME/SSR Mode S/GNSS signal definition may be changed. This feature is critical to DME, since there are several proposals regarding the DME enchantments. These enchantments include the slight changes in pulse shape. According to [71] the DME pulse shape change offers more precise ranging, but since the shape of the pulses is changed it may cause the effect on interference with the LDACS system.

As mentioned, the DME interference can be implemented in the simulation. The interference type can be set using `param_struct.chan.interference` variable in `set_parameters_fl/rl` function. In this function the DME interference itself can be customized. The MATLAB simulation for LDACS offers a possibility to set the number of pulse pairs per second, interference-to-noise ratio, the frequency offset between DME and LDACS, and also the number of DME stations that influence the LDACS signal. These parameters can be set by `du_cy_vec`, `INR_vec` and `fc_vec` variable change of `set_parameters_fl/rl` function respectively. The number of DME stations is defined as vector length of these parameters. For this scenario 2 DME with following parameters are considered:

- Number of pulse pairs per second - 2700 pps in case of one DME, 2700 and 3600 pps in case of 2 DMEs
- Interference to noise ratio - 10 dB in case of one DME, 10 and 12 in case of 2 DMEs Frequency offset between DME and LDACS centere frequencies - 0.5 MHz in case of

one DME, 0.5 MHz and -0.5 MHz in case of 2 DMEs

In figure 6.7 the result of interference add are shown.

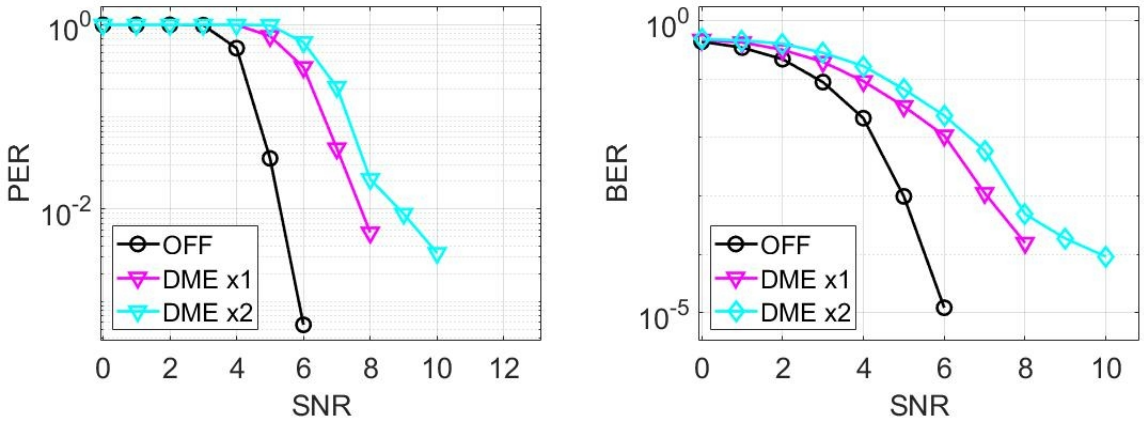

Figure 6.7: PER and BER curves for different interference events

From the figures above the influence of DME to LDACS can be seen. In both cases the performance loss can be seen. However, it is not strong, and the LDACS signal still meets the requirement of BER in the typical SNR interval.

Evaluation of the proposed platform/Discussion

In this part author would like to discuss the proposed LDACS platform. Speaking of the platform's software level, the app, author chosen, assume usage of official simulation. This approach has its own drawbacks. For example, in case of multiple frame transmission, the simulation generates data for each frame separately. Author tried to overcome this issue by saving multiple frames into the binary file during the USRP-USRP transmission. However, this method showed poor performance, compared to the transmission of one frame with ADALM-Pluto. Also, since the simulation is not featured with a zero noise channel, it is impossible to achieve the same result multiple times one after another. The next issue is the inability of native real digital message transmission - now the simulation is generating random bits for every frame. To solve this issue, the simulation can be rebuilt using the edited LDACS transmitter and receiver functions. Enabling the continuous transmission of frames that contain actual data will bring the result of the transmission closer to reality.

Speaking of the hardware available for the purpose of proposition of the LDACS testing platform, author had the opportunity to try two different hardware options for the platform design. As mentioned in Chapter 4, when author had USRPs for my experiments, the result had poor performance. This is because author had limited time in the laboratory during my study in France. And yet the result on the brink of chance is caused by the mistake during the experiment establishing. The procedure of the transmission of a binary file using USRPs should be possible, even with respect to the fact that the original data vector (generated by LDACS transmitter) is converted into different data formats several times during the signal process before and after transmission. This platform has another issue - there is 1 PC per USRP. This makes it hard to automate the transmission process, which is important in the laboratory environment. Applying it to the platform that author had in Toulouse, both

USRPs should have been connected to one PC, where 2 MATLAB instances are running simultaneously.

On the other hand, the transmission using ADALM-Pluto showed great result in terms of performance. Even if the transmission can be evaluated as realistic, the limitation of hardware makes it hard to recommend it for the LDACS test platform. Nevertheless, the way how LDACS signal is processed can be recommended - it is relatively simple and can be easily automated (as proposed in chapter 5).

Summarizing all the above, the ideal LDACS teat platform should have one host PC with two USRPs. As for the software for signal process and platform establishment, author would recommend to use the Simulink receiver and transmitter for USRP. These blocks are available after installation of the Communications Toolbox Support Package for USRP Radio add on to MATLAB. The setup of USRP transmitter receiver is very similar to the setup of ADALM-Pluto Simulink blocks (shown in figure 5.15). Then, the signal received on USRP can be processed and plotted using the code that is used for ADALM-Pluto transmission.

As the last part of this discussion, author want to mention some possible enchantments and possible tests for the proposed platform. Since the LDACS system is operated in the L-band, there is not only DME system interference with which it can be tested. During the consultation with the head of the SigNav laboratory in Toulouse we discussed the future of the LDACS system. His opinion is that the LDACS system may cause interference with GNSS at some frequencies. This assumption can be tested with the addition of a GNSS antenna to the platform installation. To bring the platform closer to reality, author can also propose to test transmitting the signal using a wireless channel. This requires managing permission to transmit in L-band (secured band) and suitable antennas for the transmission. This would allow interference and noise to be added through the wireless channel (one USRP for signal transmission and reception and another one for interference transmission), measure the SNR required for the real world and evaluate the signal performance under some other conditions other than those proposed in the official LDACS simulation.

Conclusion

The motivation for this master thesis was to propose a solution for the LDACS testing that would be capable to simulate the behavior of the real signal assuming the use of hardware for signal transmission. The design of this test platform can be used in a laboratory environment for further experiments with different scenarios. The platform can be used for further LDACS testing to reveal all drawbacks of this system. Due to existing options for simulation customization, the number of scenarios that are available for testing is large.

At the beginning of this thesis the difficulties that the CNS infrastructure is facing are described, as well as the project and propositions regarding the solution of these issues. Taking that into account, it was defined, that there is a demand for a high performance and efficient solution that supports multi functionality in its use. In the next part of this thesis, a detailed description of the LDACS system was proposed. This part was featured with the description of processes that are going on in the transmitter and receiver of the LDACS system.

In the third part of the thesis the methodology for the LDACS test platform design was introduced. According to this methodology, the research on the best suitable software and hardware solution was made. In the mentioned research MATLAB, GNU Radio and LabView software were evaluated by means of availability for the end user, SW+HW compatibility, SW usage simplicity, SW popularity, SW advanceness in term of signal processing and also by the existence of similar project. According to this evaluation, it has been found, that the MATLAB is the best solution, especially due to the fact that DLR introduced MATLAB simulation for LDACS. Then, the overview of the hardware available on the market was proposed. Regarding the hardware solution, the author was limited to the available solutions at the university laboratory.

In a fifth chapter the design of the platform itself was proposed. This chapter described the MATLAB simulation for LDACS in detail and some attributes in the form of time and frequency domain plots were added. It was decided to use the MATLAB simulation for

LDACS as a SW solution for the testing platform. The first attempt to build the testing platform was made in cooperation with French Civil Aviation University, that provided their SIGNAV laboratory for the LDACS test platform designing. The mentioned attempt was not successful, since the error rate of the transmission was around 50%. The second attempt was made with the ADALM-Pluto SDR system which was borrowed from the CNS laboratory in CTU in Prague. This time the results were way better - BER and PER of the transmission were on zero level in certain SNR values.

The last part of the thesis was featured with introducing certain testing scenarios. In this part four different scenarios were proposed. All of them were performed on the proposed testing platform. The results were within the expectation.

The author concluded that the proposed platform solution with ADALM-Pluto can be used in the laboratory environment for the LDACS system testing. As for further research recommended by the author, this platform can be modernized to more performances hardware in order to achieve more precise results. Then, the permission to transmit the signal in L-band can be obtained in order to send the signal in a wireless way.

With such a modernized platform, it is possible to test the designed test scenarios with results that more closely correspond to the real environment. This can achieve a cost-efficient simulation that evaluates the possibility of the LDACS system functioning in a real environment and its effect on the surrounding systems and vice versa, the effect of the LDACS system on the already existing CNS systems. Of course, later on, it is necessary to carry out real testing with a system prototype as part of the standardization process.

Bibliography

[1] Mitigation of dme interference in ldacs1-based future air-to-ground (a/g) communications. Availible at: https://www.tandfonline.com/doi/full/10.1080/23311916.2018.1472199.

[2] Understanding 8.33khz transmit/receive frequencies and their specific channel number. Availible at: https://www.ofcom.org.uk/__data/assets/pdf_file/0023/144590/Understanding-8.33kHz-frequencies-and-their-specific-channel-number.pdf.

[3] White paper: Ubiquitous aviation connectivity with ldacs. Availible at: https://www.frequentis.com/sites/default/files/support/2021-02/Frequentis_ATM_White-paper_LDACS_0.pdf.

[4] Yong soo cho, jaekwon kim, won young yang, chung g. kang. Availible at: https://onlinelibrary.wiley.com/doi/book/10.1002/9780470825631.

[5] Communication,navigation and surveillance (cns). Availible at: https://cordis.europa.eu/programme/id/H2020_Sesar-08-2015, 26 May 2015.

[6] Laith Abdul-Rahaim. Proposed realization of modified scrambling using 2d-dwt based ofdm transceivers. Availible at: https://www.researchgate.net/publication/266674755_Proposed_Realization_of_modified_Scrambling_using_2D-DWT_Based_OFDM_Transceivers/citations, 09 2009.

[7] Inc. Advanced Micro Devices. Field programmable gate array (fpga). Availible at: hhttps://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.htmlttps://dspillustrations.com/pages/posts/misc/the-cyclic-prefix-cp-in-ofdm.html.

[8] Majed Albogame and Khaled Elleithy. Enhancement orthogonal frequency division multiplexing (ofdm) in wireless communication network by using pts(partial transmit

sequences) technique. Availible at: https://core.ac.uk/download/pdf/52956625.pdf, 03 2014.

[9] Jean Armstrong. Ofdm for optical communications, Feb 2009.

[10] A. Bilzhause, B. Belgacem, M. Mostafa, and T. Graupl. Datalink security in the l-band digital aeronautical communications system (ldacs) for air traffic management, 2017.

[11] Cisco. Signal-to-noise ratio (snr) and wireless signal strength. Availible at: https://documentation.meraki.com/MR/Wi-Fi_Basics_and_Best_Practices/Signal-to-Noise_Ratio_(SNR)_and_Wireless_Signal_Strength.

[12] National Instruments Corp. What is labview? Availible at: https://www.ni.com/cs-cz/shop/labview.html.

[13] Bela G. LiptakSilicon DSP Corporation. Instrument engineers' handbook, vol. 2: Process control and optimization, 2nd edition. Availible at: https://nivelco.com.ua/documents/technical%20publications%20docs/Instrument%20Engineers%20Handbook%2C%20Volume%202%20Process%20Control%20and%20Optimization%2C%20Fourth%20Edition%20%28Volume%202%29.pdf, 2006.

[14] Silicon DSP Corporation. Orthogonal frequency-division multiple access. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/ofdma.pdf, 2007.

[15] Silicon DSP Corporation. Carrier frequency offset estimation and correction. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/cfo.pdf, 2021.

[16] Silicon DSP Corporation. Forward error correction (fec), convolutional encoder, viterbi decoder. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/interleaver.pdf, 2021.

[17] Silicon DSP Corporation. Interleaving. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/interleaver.pdf, 2021.

[18] Silicon DSP Corporation. Introduction to reed solomon coding. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/IntroductionReedSolomonCoding.pdf, 2021.

[19] Silicon DSP Corporation. Ofdm cyclic prefix. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/cyclic_prefix.pdf, 2021.

[20] Silicon DSP Corporation. Ofdm fundamentals. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/ofdm.pdf, 2021.

[21] Silicon DSP Corporation. Physical layer frame organization. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/plcp.pdf, 2021.

[22] Silicon DSP Corporation. Pilot tracking. Availible at: https://silicondsp.com/OFDM_Tutorial_PDF_Files/pilots.pdf, 2021.

[23] Fabio da Costa Pinto, Fernando Sérgio Oliveira Fernando Sérgio Oliveira Scoralick, Fabrício Pablo Virginio de Campos, Zhi Quan, and Moisés Vidal Ribeiro. A low cost ofdm based modulation schemes for data communication in the passband frequency. Availible at: https://ieeexplore.ieee.org/document/5764434, 2011.

[24] Kresimir Dabcevic. Intelligent jamming and anti-jamming techniques using cognitive radios. Availible at: https://www.researchgate.net/publication/279749856_Intelligent_jamming_and_anti-jamming_techniques_using_Cognitive_Radios, 04 2015.

[25] SDRF Cognitive Definitions. Sdrf cognitive radio definitions. Availible at: http://www.sdrforum.org/pages/documentLibrary/documents/SDRF-06-R-0011-V1_0_0.pdf, 2007.

[26] Xiaodai Dong, Wu-sheng Lu, and Anthony C.K. Soong. Linear interpolation in pilot symbol assisted channel estimation for ofdm. Availible at: https://ieeexplore.ieee.org/document/4202196, 2007.

[27] Robert Emilien. Cns evolution: roadmap & strategy. Availible at: https://www.sesarju.eu/sites/default/files/documents/events/wac2019/wac2019-day2-cns.pdf.

[28] Ulrich Epple, Felix Hoffmann, and Michael Schnell. Modeling dme interference impact on ldacs1. Availible at: https://ieeexplore.ieee.org/document/6218404, 2012.

[29] Deutsches Zentrum für Luft und Raumfahrt. Ldacs software - matlab simulation for ldacs. Availible at: https://www.ldacs.com/software/, 2007.

[30] GREAT SCOTT GADGETS. Hackrf one. Availible at: https://greatscottgadgets.com/hackrf/one/.

[31] Praveen Gorla, T. Sanguankotchakorn, and Poompat Saengudomlert. Object localization using rf based ieee 802.11 frame transceiving with fer based rssi controlled calibration. Availible at: https://osf.io/preprints/indiarxiv/whu2t/download, 03 2020.

[32] Bernhard Haindl, Stefan Müller, Holger Arthaber, Josef Meser, Miodrag Sajatovic, Thomas Faseth, and Michael Zaisberger. Ldacs1 conformance and compatibility assessment. Availible at: https://ieeexplore.ieee.org/document/6979589, 2014.

[33] Robert Helbet, Paul Bechet, Vasile Monda, Simona Miclaus, and Iulian Bouleanu. Low-cost sensor based on sdr platforms for tetra signals monitoring. *Sensors (Basel, Switzerland)*, 21, 05 2021.

[34] Tassadaq Hussain, Musawir Khan, Mujeeb Ur Rehman, Wasim Akram, Khalid Anayat, Ahtisham Arshad, Ahsan Akbar, and Asad Habib. A high performance software defined radio system architecture and development environment for a wide range of applications. Availible at: https://ieeexplore.ieee.org/document/8346424, 2018.

[35] DSP Illustrations. The cyclic prefix for ofdm. Availible at: https://dspillustrations.com/pages/posts/misc/the-cyclic-prefix-cp-in-ofdm.html.

[36] Ananlog Devices Inc. Adalm-pluto - software-defined radio active learning module. Availible at: https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html#eb-overview.

[37] Simon Dunkley Ian CasewellDSP Illustrations John Micallef, Richard Womersley. Fci technology investigations: L band compatibility criteria and interference scenarios study deliverables s1-s7: L-band interference scenarios. Availible at: https://www.eurocontrol.int/sites/default/files/2019-05/25082009-lcis-s1tos5-compatibility-scenarios-v10.pdf.

[38] Mohamed Essam Khedr. Orthogonal frequency division multiplexing and related technologies. Availible at: .

[39] Kitz.co.uk. Interleaving explained. Availible at: https://kitz.co.uk/adsl/interleaving.htm, 2015.

[40] Arun Kumar and Manisha Gupta. A review on ofdm and papr reduction techniques. Availible at: https://thescipub.com/pdf/ajeassp.2015.202.209.pdf, 02 2015.

[41] David Levin. PŘehled nÁvrhŮ alternativnÍch navigaČnÍch systÉmŮ a posouzenÍ jejich vÝkonnosti. Availible at: https://dspace.cvut.cz/handle/10467/90535?show=full, 2020.

[42] Dr. Marc Lichtman. Pysdr: A guide to sdr and dsp using python. Availible at: https://pysdr.org/content/iq_files.html#binary-files.

[43] MathWorks. Matlab documentation: Pluto transmitter - transmit data to analog devices adalm-pluto radio. Availible at: https://www.mathworks.com/help/supportpkg/plutoradio/ref/plutotransmitter.html.

[44] Wolfram MathWorld. Error-correcting code. Availible at: https://mathworld.wolfram.com/Error-CorrectingCode.html.

[45] Jingbo Meng and Guihua Kang. A novel ofdm synchronization algorithm based on cazac sequence. Availible at: https://www.sciencedirect.com/topics/engineering/cyclic-prefix, 2010.

[46] Sandeep Mukherjee, Suvra Sekhar Das, Aritra Chatterjee, and Sourav Chatterjee. Analytical calculation of rician k-factor for indoor wireless channel models. Availible at: https://ieeexplore.ieee.org/document/8030982, 2017.

[47] Nils Mäurer and Arne Bilzhause. A cybersecurity architecture for the l-band digital aeronautical communications system (ldacs). Availible at: https://ieeexplore.ieee.org/abstract/document/8569878, 10 2018.

[48] Official page of GNU Radio. Availible at: https://www.gnuradio.org/.

[49] Rakesh Palisetty, Amit Kumar Panda, and Kailash Chandra Ray. Secure ofdm based on coupled linear congruential generator and its fpga prototype. Availible at: https://ieeexplore.ieee.org/document/8549826, 2018.

[50] Stanislav Pleninger. Cns (communication-navigation-surveillance) systémy. Prague, CTU in Prague.

[51] Christos Politis, Sina Maleki, Juan Merlano Duncan, Jevgenij Krivochiza, Symeon Chatzinotas, and Björn Ottesten. Sdr implementation of a testbed for real-time interfer-

ence detection with signal cancellation. Availible at: https://core.ac.uk/download/pdf/154761365.pdf, 2018.

[52] Waris Junaid Pratibha Mishra, Mohd. Ayub Khan. Analysis of equalization in digital communication. Availible at: https://ieeexplore.ieee.org/document/697958?denied=, 1992.

[53] Ettus Research. Products - usrp b200mini-i. Availible at: https://www.ettus.com/all-products/usrp-b200mini-i-2/.

[54] Ettus Research. Products - usrp x310. Availible at: https://www.ettus.com/all-products/x310-kit/.

[55] Ettus Research. Sdr software. Availible at: https://www.ettus.com/sdr-software/l.

[56] Ettus Research. Usrp hardware driver and usrp manual. Availible at: https://files.ettus.com/manual/.

[57] Ettus Research. Usrp hardware driver and usrp manual. Availible at: https://files.ettus.com/manual/page_usrp_x3x0.html.

[58] Christoph Rihacek. Ldacs - from single function to integrated cns. Availible at: https://www.sesarju.eu/sites/default/files/documents/webinars/webinar%20cns%20vision%20From%20single%20function%20to%20integrated%20CNS%20-%20Christoph%20Rihacek%2C%20Frequentis.pdf.

[59] RTL-SDR. Rtl-sdr dongles (rtl2832u). Availible at: https://www.rtl-sdr.com/buy-rtl-sdr-dvb-t-dongles/.

[60] M.N.O. Sadiku and C.M. Akujuobi. Software-defined radio: a brief overview. Availible at: https://ieeexplore.ieee.org/abstract/document/1343223, 2004.

[61] Hikmet Sari and Georges Karam. Orthogonal frequency-division multiple access and its application to catv networks. Availible at: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.4460090605, 1998.

[62] Michael Schnell, Ulrich Epple, Dmitriy Shutin, and Nicolas Schneckenburger. Ldacs: future aeronautical communications for air-traffic management. Availible at: https://ieeexplore.ieee.org/document/6815900?reloa=, 2014.

[63] EUROCONTROL SESAR Joint Undertaking. Updated ldacs1 system specification. Availible at: http://www.ldacs.com/wp-content/uploads/2014/02/LDACS1-Updated-Specification-Proposal-D2-Deliverable.pdf.

[64] EUROCONTROL SESAR Joint Undertaking. Ldacs a/g specification. Availible at: https://www.ldacs.com/wp-content/uploads/2013/12/SESAR2020_PJ14_D3_3_030_LDACS_AG_Specification_00_02_02-1_0.pdf, 2019.

[65] EUROCONTROL SESAR Joint Undertaking. Sesar ju webinar: Introducing ldacs (l-band digital aeronautical communication system). Availible at: https://www.sesarju.eu/node/3809, 2021.

[66] EUROCONTROL SESAR Joint Undertaking. Sesar ju webinar: Ldacs deployment. Availible at: https://www.sesarju.eu/node/3852, 2021.

[67] EUROCONTROL SESAR Joint Undertaking. Sesar ju webinar: Technical details and capabilities of ldacs. Availible at: https://www.sesarju.eu/node/3832, 2021.

[68] Matthew Smith, Daniel Moser, Martin Strohmeier, Vincent Lenders, and Ivan Martinovic. Analyzing privacy breaches in the aircraft communications addressing and reporting system (acars). Availible at: https://arxiv.org/pdf/1705.07065.pdf, 05 2017.

[69] department of computer science Stanislaus State. Multiplexing and demultiplexing (channelization). Availible at: https://www.cs.csustan.edu/~john/classes/previous_semesters/CS3000_Communication_Networks/2018_02_Spring/Notes/chap11.html, 2018.

[70] Muh Taufiqurrahman, I Gede Astawa, and Amang Sudarsono. Performance analysis of circular 8-qam constellation with mmse equalizer for ofdm system using usrp. Availible at: https://emitter.pens.ac.id/index.php/emitter/article/view/148, 12 2016.

[71] Valeriu Vitan, Gerhard Berz, Luca Saini, Jean-Pierre Arethens, Boubeker Belabbas, and Petr Hotmar. Research on alternative positioning navigation and timing in europe. Availible at: https://ieeexplore.ieee.org/document/8384887, 2018.

[72] S. Weinstein and P. Ebert. Data transmission by frequency-division multiplexing using the discrete fourier transform. Availible at: https://ieeexplore.ieee.org/abstract/document/1090705, 1971.

[73] Bob Witte. Digital modulation basics, part 1. Availible at: https://www.5gtechnologyworld.com/digital-modulation-basics-part-1/.

[74] Colin Yao. What is qam modulation. Availible at: https://pt.slideshare.net/fiberoptics4sale/what-is-16-qam-modulation.

[75] Omar Yeste-Ojeda and Rene Jr Landry. Software defined radio approach to distance measuring equipment. Availible at: https://www.researchgate.net/publication/269297416_Software_defined_radio_approach_to_distance_measuring_equipment, 05 2014.

[76] A. Zaidi, F. Athley, J. Medbo, U. Gustavsson, G. Durisi, and X. Chen. 5g physical layer: Principles, models and technology components. Availible at: https://www.sciencedirect.com/topics/engineering/cyclic-prefix, 2018.

[77] Aida Zaier and Ridha Bouallegue. Channel estimation study for block - pilot insertion in ofdm systems under slowly time varying conditions. Availible at: https://www.researchgate.net/publication/220486321_Channel_Estimation_Study_for_Block_-_Pilot_Insertion_in_OFDM_Systems_Under_Slowly_Time_Varying_Conditions/citations, 02 2012.

[78] Tomáš Zelinka, Martin Šrotýř, and Zdenek Lokaj. Telekomunikace a místní sítě, 2020. Prague, CTU in Prague.

[79] Ru Zhang, Gongshen Liu, Jianyi Liu, and Jan P. Nees. Analysis of message attacks in aviation data-link communication. Availible at: https://ieeexplore.ieee.org/document/8086144, 2018.

List of Figures

List of Tables