TECHNICAL UNIVERSITY OF CRETE
ELECTRONIC AND COMPUTER ENGINEERING
DEPARTMENT

TELECOMMUNICATIONS DIVISION

Implementation of distributed Alamouti space time code in Orthogonal frequency-division multiplexing (OFDM) system using USRP N200

by

© Nikolaos Karamolegkos

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

ELECTRONIC AND COMPUTER ENGINEERING

THESIS COMMITTEE

Professor Athanasios P. Liavas, Thesis Supervisor

Associate Professor George N. Karystinos

Professor Michael Paterakis

December 2015

Abstract

Orthogonal frequency division multiplexing (OFDM) is a modulation technique which is now used in most wired and wireless communication systems, because it is an effective solution to cope with intersymbol interference caused by the channel. In wireless communication, the concept of parallel transmission of symbols is used to achieve high throughput and better transmission quality. OFDM is one of the techniques for parallel transmission. OFDM splits the total transmission bandwidth into a number of orthogonal subcarriers in order to transmit the symbols using these subcarriers in parallel. It is a widely used modulation and multiplexing technology, which has become the basis of many telecommunications standards including 4G, wireless local area networks (LANs), digital terrestrial television (DTT) and digital radio broadcasting in much of the world.

Multiple-input multiple-output (MIMO) technology significantly enhances system performance. The extra degrees of freedom afforded by the multiple antennas can be used for increasing bit rates through spatial multiplexing or for improved diversity order through space-time coding techniques.

In this thesis, we combine OFDM and MIMO systems. Specifically, we design and implement an OFDM system with two transmitters and one receiver with Alamouti space time code (STC) using USRP. At the beginning, we create an OFDM transmitter and receiver with N subcarriers in Matlab and GNU radio and then we build it using C++ for real time processing. The two transmitters modify the data, as described, by Alamouti STC and they send them simultaneously. The receiver has to receive and process the data in order to detect the transmitted symbols. Time synchronization, CFO cancellation and channel estimation are such processes. The transmitters are two USRPs N200, which were synchronized to send the Alamouti symbols at the same time. Also, the receiver is a USRP N200. Before the Alamouti scheme, we implemented an OFDM system with one transmitter and one receiver. Finally, we measure some performance metrics, like SNR and BER.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

OFDM ..... Orthogonal Frequency-Division Multiplexing

MCM ..... Multi-Carrier Modulation

ADSL ..... Asymmetric Digital Subscriber Line

WLAN ..... Wireless Local Area Network

RF ..... Radio Frequency

ISI ..... Inter-Symbol Interference

ICI ..... Inter-Carrier Interference

SDR ..... Software-Defined Radio

STC ..... Space time code

FPGA ..... Field Programmable Gate Arrays

SoC ..... System on Chip

GPP ..... General Purposes Processors

DSP ..... Digital Signal Processors

ADC ..... Analog to Digital Converter

DAC ..... Digital to Analog Converter

USRP ..... Universal Software Radio Peripheral

TCXO ..... Temperature Compensated Crystal Oscillator

ppm ..... parts per million

GRC ..... GNU Radio Companion

QAM ..... Quadrature Amplitude Modulation

PSK ..... Phase Shift Keying

DFT ..... Discrete Fourier Transform

IDFT ..... Inverse Discrete Fourier Transform

SRRC ..... Square Root Raised Cosine

DSW ..... Double Sliding Window

CFO ..... Carrier Frequency Offset

CP ..... Cyclic Prefix

LS ..... Least Squares

MIMO ..... Multiple Input Multiple Output

SISO ..... Single Input Single Output

SNR ..... Signal to Noise Ratio

BER ..... Bit Error Rate

Acknowledgements

I would like to express my sincere gratitude to my mentor Prof. Athanasios P. Liavas, for sharing his knowledge and experience with me and for giving me the opportunity to delve into such an interesting and challenging topic. His assistance and guidance were invaluable. Moreover, I would like to thank my family for supporting me wholeheartedly throughout that journey. Next, I wish to thank my friends that were there for me every step of the way, especially A. Gkiolias for his great contribution.

To my mentor, my family, and all those who believed in me ...

Chapter 1

Introduction

1.1 Thesis Outline

The thesis is organized as follows:

- In Chapter 1, we present a brief introduction to wireless communications, bandpass modulation, orthogonal frequency-division multiplexing, software-defined radio, and space time codes.
- In Chapter 2, we describe the software tools that we used as well as the model and specifications of USRPs.
- Chapter 3 introduces and analyzes theoretically an OFDM system. We consider both the analog and the digital model.
- In Chapter 4, we describe the structure of an OFDM system in C++ (transmitter and receiver), like channel estimation, packet detection, and CFO estimation.
- In Chapter 5, we describe the way that we built a system using Alamouti STC and OFDM using two USRPs as transmitters and one as receiver. Also, we simulate a system with two transmitters and two receivers using matlab. Moreover, the Signal to Noise (SNR) of these systems.
- Finally, in Chapter 6, a conclusion is provided, followed by some suggestions and directions for future work.

1.2 Wireless Communication

Wireless is a term used to describe telecommunications in which electromagnetic waves carry the signal over part or all of the communication path. Wireless communications is the fastest growing segment of the communication industry and is rapidly evolving and playing an increasing role in the lives of people all over the world. Despite the hype of the popular press, is a field that has been around for about a hundred years. Specifically, term wireless communication was introduced in the 19th century (with Marconi's wireless telegraphy) and the wireless technology has been developed over the subsequent years.

Wireless communication is now applied mostly in cellular telephone systems and networks which are extremely popular worldwide. A cellular network consists of

a large number of wireless devices that can be used in cars, buildings, and almost anywhere. There are also a number of fixed base stations, arranged to provide coverage (electromagnetic) of the subscribers.

In the present days, wireless is one of the most important media of transmission of information from one device to other devices. Via this technology, the information can be transmitted through the air without requiring cables, wires, or other electronic conductors. Wireless communication systems are used in every aspect of our life such as cell phones, infrared communication, broadcast radios, computers, WIFI, bluetooth technology, and satellite communications in which wired communication is not feasible. As a result, the research in this area becomes even more important and several projects studying wireless networks with different extents of coverage are underway.

1.3 Digital bandpass modulation

Two important digital bandpass modulation techniques are

- Single-carrier modulation, in which data is transmitted by using one single radio frequency (RF) carrier.
- Multi-carrier modulation, in which data is transmitted by simultaneously modulating multiple RF carriers.

1.3.1 Single-carrier modulation

Single-carrier modulation techniques use only one sinusoidal wave at all times. Basic single-carrier modulation techniques modify only one of the next three parameters, amplitude, frequency or phase of the sinusoidal wave, according to the information symbol to be transmitted. An overview of the signal bandwidth is depicted in Figure 1.1.

Single-carrier has been used in wireless as well as in wireline applications such as voiceband modems. However, single-carrier modulation is sensitive to inter-symbol interference (ISI) when it is used over frequency-selective channels. Thus, it is important to deal with ISI while at the same time exploit the inherent frequency diversity of the channel. This can be done using linear and non-linear processing at the receiver (i.e., using Viterbi algorithm for optimal detection).

1.3.2 Multi-carrier modulation

Multi-carrier modulation (MCM) is a method of transmitting data by splitting it into several components, and sending each of them over separate frequency bands. An overview of the signal bandwidth is depicted in Figure 1.2. The individual carriers have narrow bandwidth, but the composite signal can have broad bandwidth.

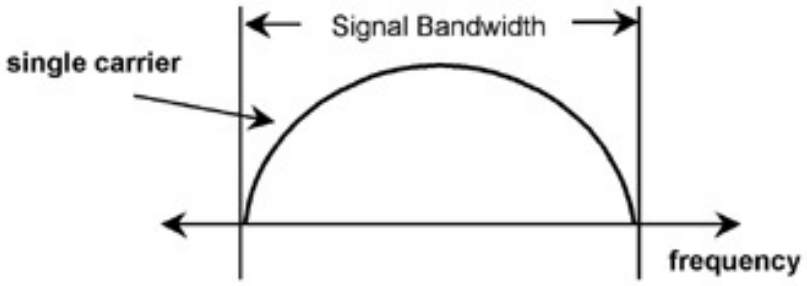


Figure 1.1: Single-carrier modulation.

The advantages of MCM include relative immunity to fading caused by transmission over more than one path at a time (multipath fading), less susceptibility to interference caused by impulse noise, and enhanced immunity to ISI than single-carrier systems. Limitations include difficulty in synchronizing the carriers under marginal conditions and a relatively strict requirement that amplification must be linear.

MCM was first used in analog military communications in the 1950s. Recently, MCM has attracted attention as a means of enhancing the bandwidth of digital communications over media with physical limitations. The scheme is used in some audio broadcast services. This technology is vital for digital television and is used as a method of obtaining high data speeds in asymmetric digital subscriber line (ADSL) systems. MCM is also used in wireless local area networks (WLANs). In this thesis we consider *Orthogonal frequency-division multiplexing (OFDM)* modulation, which is a particular type of MCM.

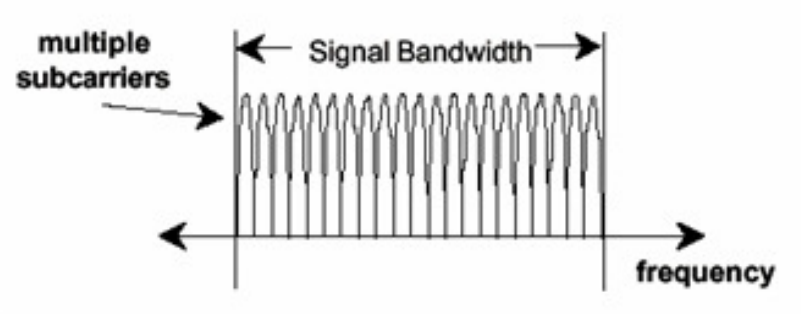


Figure 1.2: Mutli-carrier modulation.

1.4 Orthogonal frequency-division multiplexing (OFDM)

OFDM is a method of encoding digital data on multiple-carrier frequencies. OFDM transmits data in parallel, by modulating a set of orthogonal subcarriers. The subcarriers have the minimum frequency separation required to maintain orthogonality. One of the greatest benefits of this method is that the modulation of closely-spaced orthogonal sub-carrier divides the available bandwidth into a collection of narrow sub-bands. As a result, this technique can be readily used to increase the overall bandwidth efficiency in a frequency selective channel.

OFDM has been developed and used as a popular scheme in wideband digital communication as well as in applications such as digital television and audio broadcasting, DSL Internet access, wireless networks, powerline networks, and lately in 4G mobile communications. It is clear that OFMD has become the definitive modulation scheme in current and future wireless communication systems.[1, 2, 6]

1.5 Software-defined radio (SDR)

A radio is any kind of device that wirelessly transmits or receives signals in the radio frequency part of the electromagnetic spectrum to facilitate the transfer of information. In today's world, radios exist in a multitude of items such as cell phones, computers, car door openers, vehicles, televisions, etc.

Traditional hardware based radio devices limit cross-functionality and can only be modified through physical intervention changing the hardware. It is easy to understand that any change of the harware is not only expensive but also impossible sometimes. In this way, the flexibility of a system is limited. By contrast, software defined radio technology provides an efficient and comparatively inexpensive solution to this problem, allowing multi-mode and multi-functional wireless devices that can be enhanced using upgrades or changes of the software.

SDR defines a collection of hardware and software technologies in which some or all of the radio operating functions are implemented through modifiable software or firmware operating on programmable processing technologies. These devices include field programmable gate arrays (FPGA), digital signal processors (DSPs), general purpose processors (GPPs), programmable System on Chip (SoC), mixers, filters, amplifiers, modulators/demodulators, detectors, Analog to Digital converters (ADCs). In Figure 1.3, a high level block diagram of a SDR transceiver is depicted. The use of these technologies allows new wireless features and capabilities to be added to existing radio systems without requiring new hardware. In this thesis, the SDR system consists of a Linux computer and USRP N200 for transceiver.

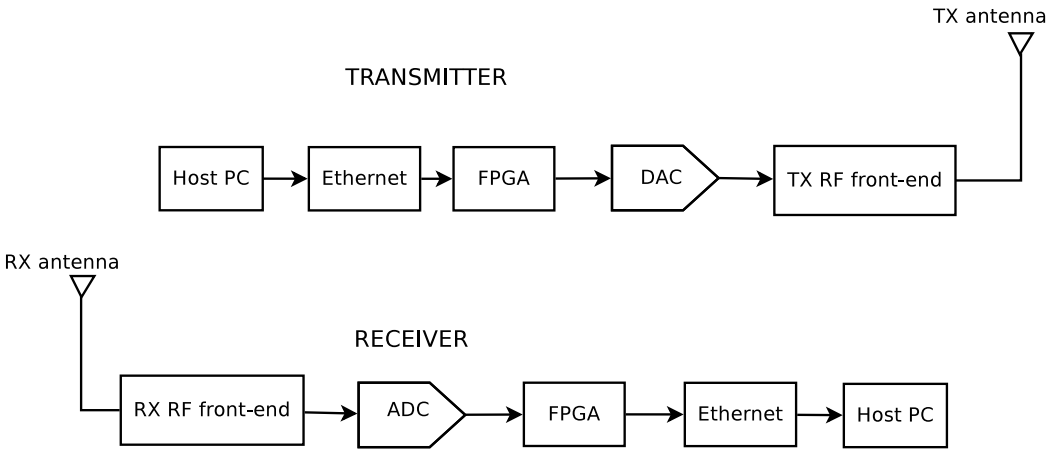


Figure 1.3: Block diagram of a SDR transmitter and receiver.

1.6 Space-time coding (STC)

Space time coding is host of techniques that are used in wireless communications to transmit multiple copies of a data stream across a number of antennas and to exploit the various received versions of the data to improve the reliability of data-transfer. The fact that the transmitted signal must traverse a potentially difficult environment with scattering, reflection, refraction and so on and may then be further corrupted by thermal noise in the receiver means that some of the received copies of the data will be "better" than others. This redundancy results in a higher chance of being able to use one or more of the received copies to correctly decode the received signal. In fact, space-time coding combines the copies of the received signal in an optimal way to extract as much information from each of them as possible.

There have been a lot of research activities in this area under the rubric of space-time coding. In this thesis, we will discuss the simplest and yet one of the most elegant space-time codes the so-called Alamouti scheme. This scheme is a transmit diversity scheme proposed in several third-generation cellular standards. Alamouti scheme is designed for 2 transmit antennas and M receive antennas.[2]

Chapter 2

Software Tools and USRP

2.1 Universal Software Radio Peripheral (USRP)

The Universal Software Radio Peripheral is the most common hardware used with GNU Radio to build a SDR system. USRP is a family of hardware devices created by Matt Ettus. It is composed of two main sub-devices, a motherboard and various daughterboards. This modularity permits the USRP to serve applications (transmit & receive) that operate from DC to up to 6 GHz. The daughterboards can easily be exchanged. In this way, USRP can work at a variety of frequency bands.

All of the high speed general purpose operations, like digital-up and down-conversion, interpolation and decimation, are done on the FPGA. The code for the FPGA is open-source and can be modified to allow high-speed, low-latency operations to occur in the FPGA. Apart from the operations described before, the FPGA does not perform any advanced digital signal processing, such as modulation, demodulation and coding. Such processing operations are always done in the host computer. USRPs connection to a host computer is through a high-speed USB or Gigabit Ethernet link. The host-based software controls the USRP hardware and operates with it. In Figure 2.1, the basic parts of a USRP 1 board such as ADC, Digital to Analog Converter (DAC), FPGA, daughterboards, as well as the way they are connected, are shown.

In order to achieve communication between the USRP and the host PC, we need some software tools. The UHD is a device driver provided for use with any product of the USRP family by Ettus Research. The platforms which are supported are Linux, MacOS, and Windows. Several frameworks, including GNU Radio, LabVIEW, MATLAB and Simulink, use UHD. The functionality provided by UHD can also be accessed directly with the UHD API, which provides native support for C++. Any other language that can import C++ functions can also use UHD. In this thesis, we will use GNU Radio, Matlab and C++.[8]

2.1.1 USRP N200 motherboard overview

The Ettus Research^{TM} USRP N200 is the highest performing class of hardware of the USRP family of products, which enables engineers to rapidly design and implement powerful and flexible software radio systems. Using a Gigabit Ethernet interface, they provide higher dynamic range and higher bandwidth than the bus series. The N200 is ideally suited for applications requiring high RF performance and large bandwidth. Such applications include physical layer prototyping, dy-

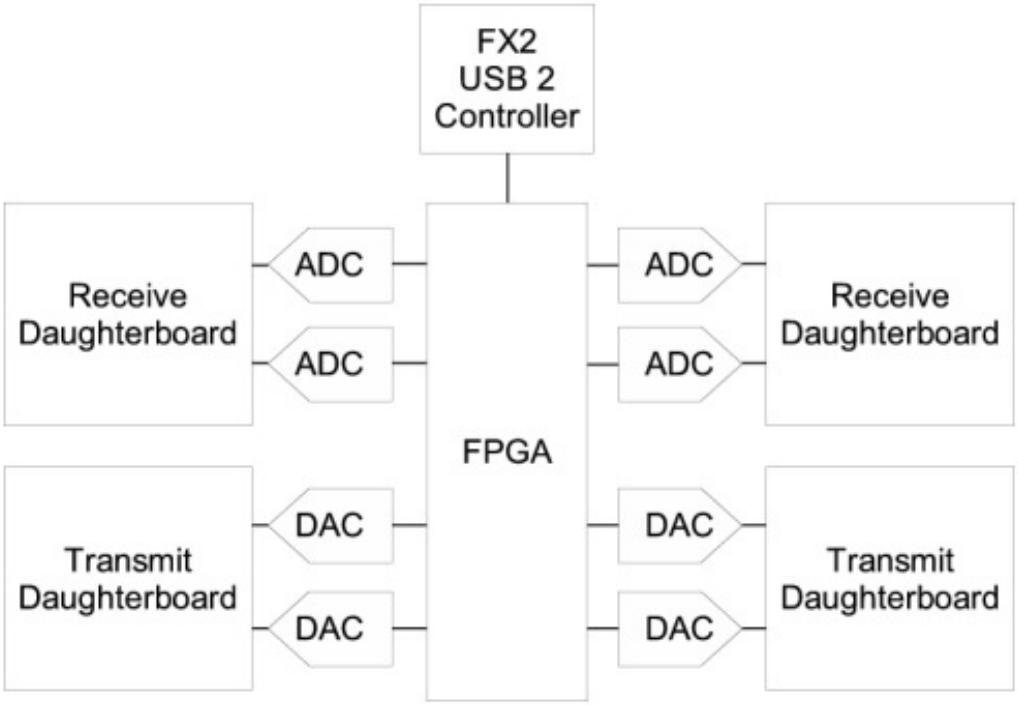


Figure 2.1: Block diagram of a USRP 1 board.

namic spectrum access and cognitive radio, spectrum monitoring, record and playback, and even networked sensor deployment. This series also provides a MIMO expansion port (located on the front panel of each unit) which can be used to synchronize two devices from this series. This is the recommended solution for MIMO systems.

Specifically, the product's architecture includes a Xilinx© Spartan© 3A-DSP 1800 FPGA, dual 14-bit, 100 MS/s ADC and dual 16-bit 400 MS/s DAC. A modular design allows the USRP N200 to operate from DC to 6 GHz using diffent daughterboards. The devices in the networked series can transfer up to 50 MS/s of real-time bandwidth in the receive and transmit directions, simultaneously (full duplex). An optional GPDSO module can also be used to discipline the USRP N200 reference clock to within 0.01 ppm of the worldwide GPS standard. Finally, USRP N200 has the ability to lock to external 5 or 10 MHz clock reference and has 2.5 ppm Temperature Compensated Crystal Oscillator (TCXO) frequency reference. In Figure 2.3, a photo of USRP N200 is shown. [8]

Figure 2.2: USRP N200.

USRP N200 features

In this subsection, we will briefly mention the features of USRP N200 series and we will describe how ADC, DAC and FPGA operate.[8]

- Use with GNU Radio, LabVIEW and Simulink
- Modular Architecture: DC-6 GHz
- Dual 100 MS/s, 14-bit ADC which converts the continuous-analog signal to a digital signal. The conversion involves quantization of the input (14-bit), so it necessarily introduces a small amount of error. Furthermore, instead of continuously performing the conversion, an ADC does the conversion periodically, sampling the input. The result is a sequence of digital values that have been converted from a continuous-time and continuous-amplitude analog signal to a discrete-time and discrete-amplitude digital signal.
- Dual 400 MS/s, 16-bit DAC which converts digital data (usually binary) into an analog signal. An ADC performs the reverse function. Unlike analog signals, digital data can be transmitted, manipulated, and stored without degradation.
- Spartan 3A-DSP 1800 FPGA, in which all the ADCs and DACs are connected. Specifically what a FPGA does is to perform high bandwidth mathematics with purpose to perform operations like digital-up and down-conversion, interpolation and decimation.
- DDC/DUC with 25 mHz Resolution
- Up to 50 MS/s Gigabit Ethernet Streaming
    1. 50 MHz of RF bandwidth with 8 bit samples
    2. 25 MHz of RF bandwidth with 16 bit samples

- Fully-Coherent MIMO Capability
- Gigabit Ethernet Interface to Host
- 2 Gbps Expansion Interface
- 1 MB High-Speed SRAM
- Auxiliary Analog and Digital I/O
- 2.5 ppm TCXO Frequency Reference
- 0.01 ppm with GPSDO Option
- Ability to lock to external 5 or 10 MHz clock reference
- FPGA code can be changed with Xilinx ISE© WebPACK™ tools

USRP N200 specifications

In this subsection, the specification of USRP N200 series is described as shown in the Table 2.1. Also, in Figure 2.3, the internal construction and the individual blocks are depicted.[8]

Specification	Type	Unit
Power		
Dc Input	6	V
Current Consumption	1.3	A
Using WBX Daughterboard	2.3	A
Conversion Performance and Clocks		
ADC Sample Rate	100	MS/s
ADC Resolution	14	bits
ADC Wideband SFDR	88	dBc
DAC Sample Rate	400	MS/s
DAC Resolution	16	bits
DAC Wideband SFDR	80	dBc
Host Sample Rate (8 bit/16 bit)	50/25	MS/s
Frequency Accuracy	2.5	ppm
GPSDO Reference	0.01	ppm

Table 2.1: USRP N200 specifications.

2.1.2 Daughterboards

The USRP family features a modular architecture with interchangeable daughterboard modules that serve as the RF front end. In the motherboard of USRP N200 there is a slot to install the desired daughterboard. There are several classes of

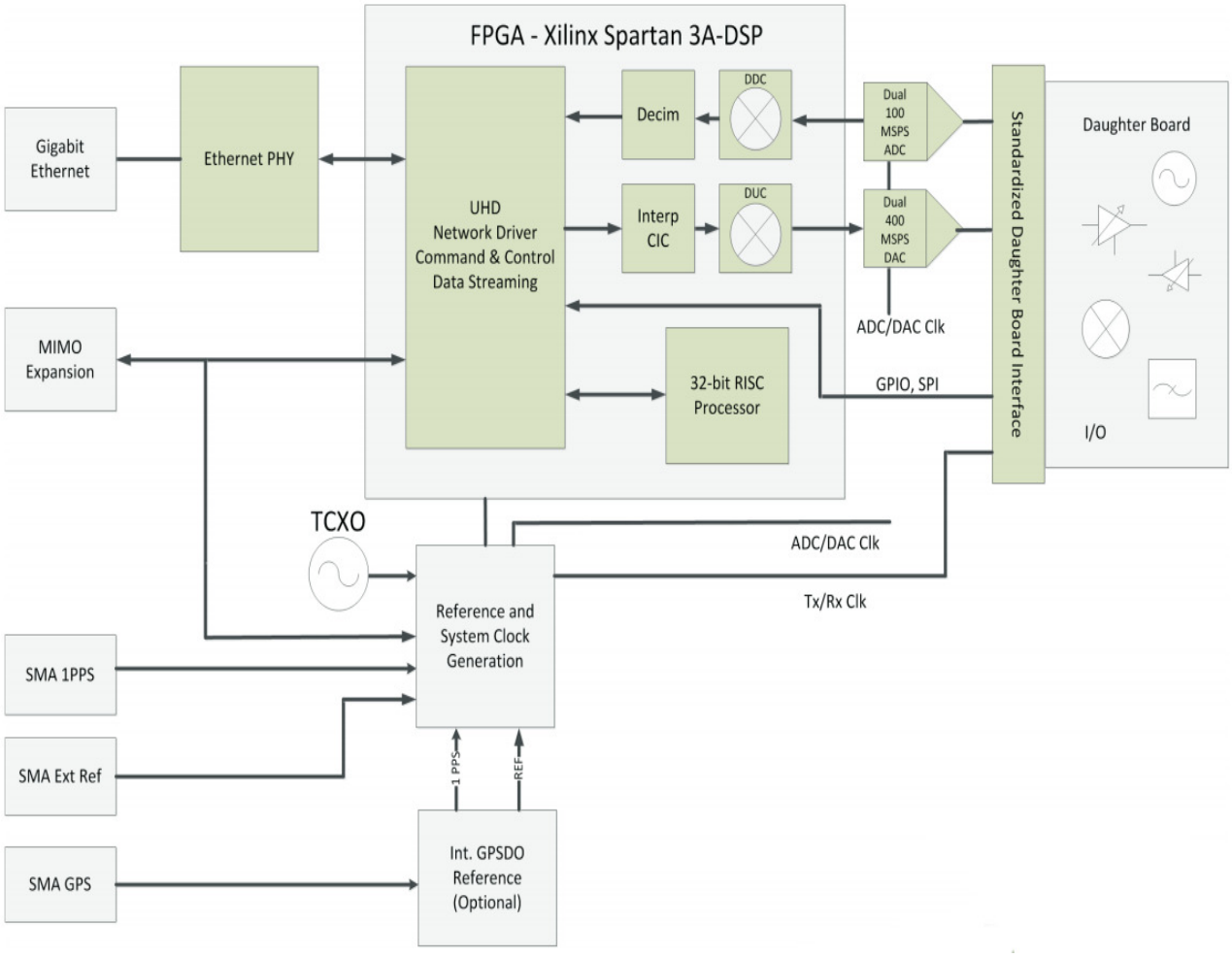


Figure 2.3: Internal construction of USRP N200.

daughterboard modules which can be used depending on the functionality (transmitter, receiver, transceivers) we want to achieve. Each type of daughterboard has different technical characteristics such as operating frequency, bandwidth, and gain range. In this thesis, we used two different types of daughterboard, the CBX and the WBX. Both CBX and WBX family of daughterboards are complete RF transceiver systems.[8]

CBX daughterboard

The CBX is a full-duplex, wideband transceiver that covers a frequency band from 1.2 GHz to 6 GHz with a instantaneous bandwidth of 40 MHz. The CBX can serve a wide variety of application areas, including WiFi research, cellular base stations, cognitive radio research, and RADAR. The local oscillators for the receive and transmit chains operate independently, allowing multi-frequency operation. The

references for the local-oscillators are derived from the USRP master-clock, which permits coherent operation when combined with MIMO-capable USRP models such as the USRP N200. At the left side of Figure 2.4, a CBX daughterboard is shown.[8]

WBX daughterboard

The WBX is a wide bandwidth transceiver that provides up to 100 mW of output power and a noise figure of 5 dB. The local oscillators for the receive and transmit chains operate independently, but can be synchronized for MIMO operation. The WBX provides 40 MHz of bandwidth capability and is ideal for applications requiring access to a number of different bands within its range from 50 MHz to 2.2 GHz. Example application areas include land-mobile communications, maritime and aviation band radios, cell phone base stations, PCS and GSM multi-band radios, coherent multi-static radars and wireless sensor networks. At right side of the Figure 2.4, a WBX daughterboard is shown.[8]

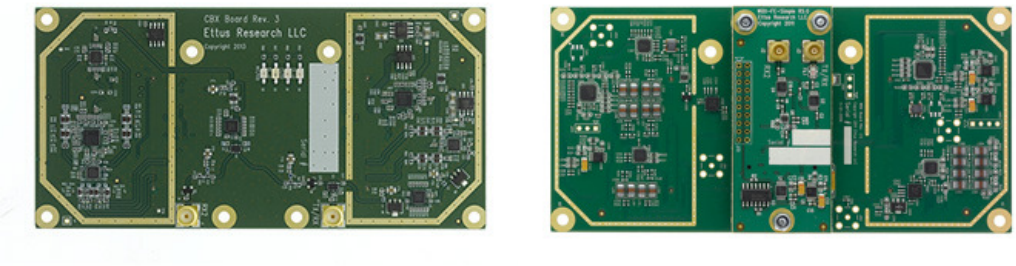
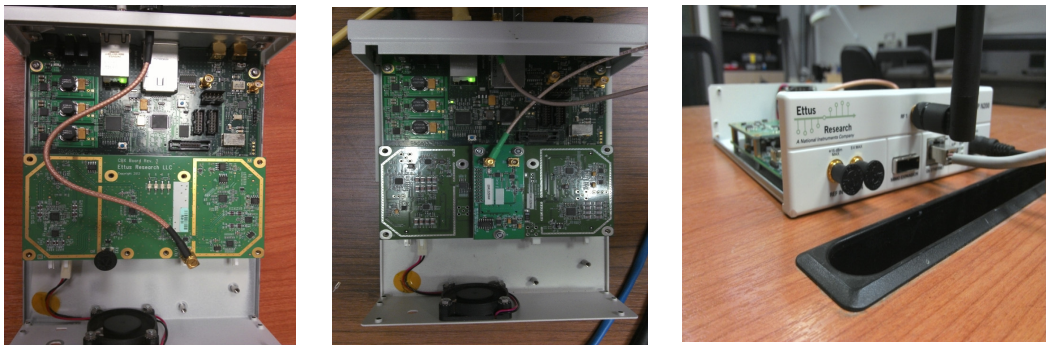


Figure 2.4: Daughterboard: (a) CBX, (b) WBX.

2.2 GNU Radio

GNU Radio was founded by Eric Blossom with the intention of creating an application programming interface for SDR platforms. Today, it is a free and open-source software development toolkit that provides signal processing blocks to implement software radios. It can be used with readily-available low-cost external RF hardware to create software-defined radios, or without hardware in a simulation-like environment. It is widely used in hobbyist, academic and commercial environments to support both wireless communications research and real-world radio systems and it works on all major platforms (Linux, Windows and Mac). [9]



(a) (b) (c)

Figure 2.5: Telecom lab hardware: (a) USRP N200 with CBX, (b) USRP N200 with WBX, (c) USRP N200.

GNU Radio companion (GRC) performs all the signal processing and it can be used to write applications to receive data out of digital streams or to push data into digital streams, which are then transmitted using hardware. GNU radio has several processing blocks like filters, channel codes, synchronization elements, equalizers, demodulators, vocoders, decoders, and many other elements which are typically found and widely used in radio systems. More importantly, it includes a method of connecting these blocks and then manages how data is passed from one block to another. Also, the user of the GNU radio can create and built blocks to extend the overall functionality. The new blocks can be written in either C++ or in Python. Each block can be edited, upgraded or even implemented independently, without interfering with the whole communication chain [9]. In this thesis, we firstly used GRC combined with Matlab. Specifically, we created/processed the OFDM frames-packets in Matlab and then we sent/received them with USRP N200 using GNU radio. A GNU radio transmitter is depicted in Figure 2.6 and a receiver is depicted in Figure 2.7. The transmitter read data from a FIFO file built by Matlab and sent them. The receiver using USRP received the transmitted data and then saved them at an other FIFO file which was processed by Matlab. As it becomes understood, the overall processing in not real-time.

2.3 Implementation in C++

As we mentioned before, the system we built using GNU radio and Matlab was not working in real-time, so we processed the data before transmission and after reception in Matlab. This method is not efficient for two reasons. The first is that the user had to open GNU radio and collect the data in a FIFO and then to use them in Matlab. The second and most important reason is Matlab's performance which is bad for our system, because we need real-time processing. In contrast,

C/C++ is a lower-level language and generates relatively optimized machine code when we compare it with any other high-level language like Matlab. For this reason, we started prototyping with Matlab where debugging is easier. When we confirmed that the code is fully functional we translated the project to C++ to gain a 10 to 100 times performance improvement. In C++, we didn't use GNU radio for transmission/reception. We transmitted and received the data calling the ready USRP C++ functions[1] send/receive. Similarly, we processed the data with the C++ code we built. As a result, our system was faster and it was running in real-time without significant losses. In this thesis, when we used USRPs everything was implemented in C++ code.

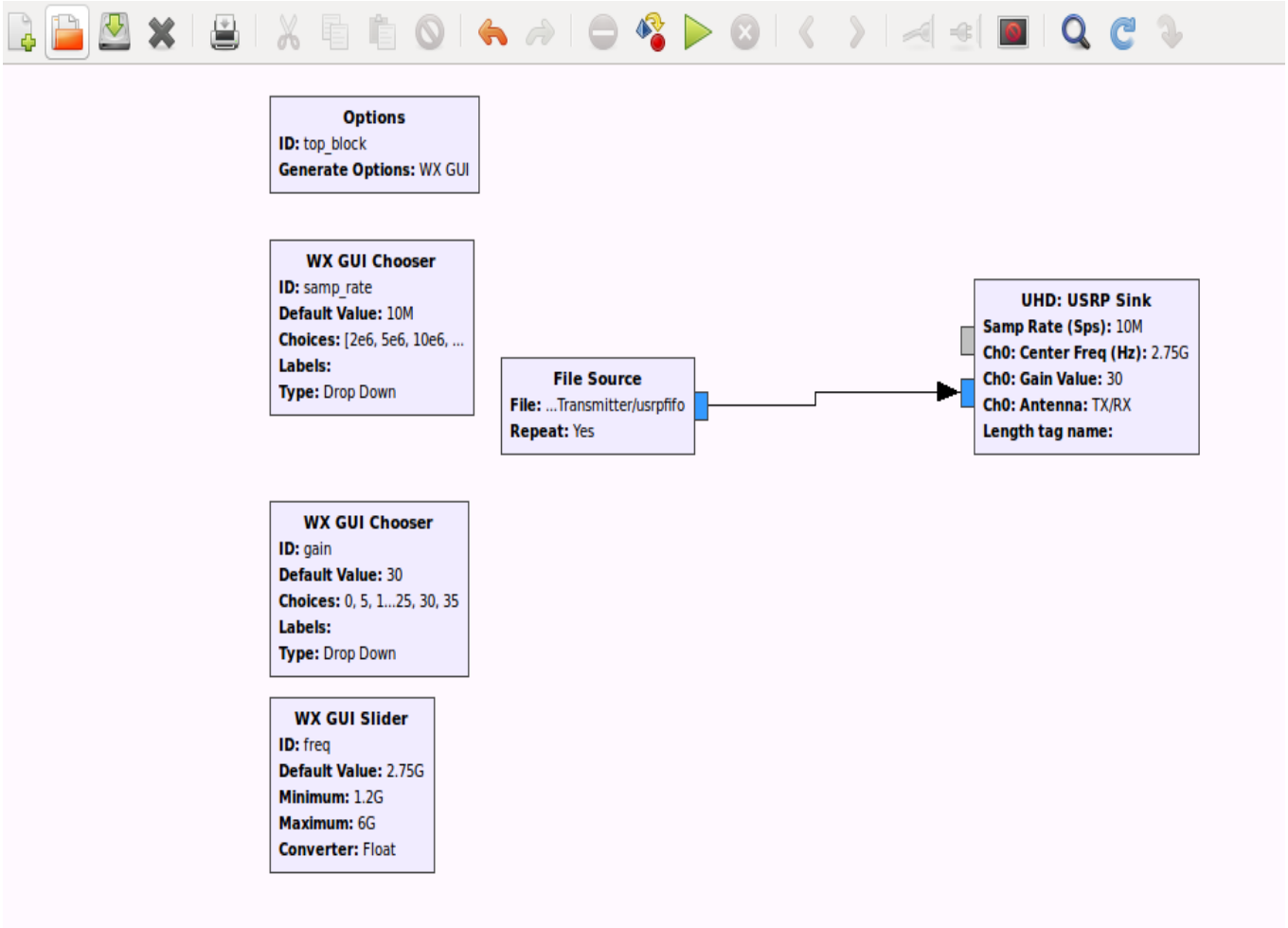


Figure 2.6: GNU radio transmitter modules.

[1]Provided by Ettus.

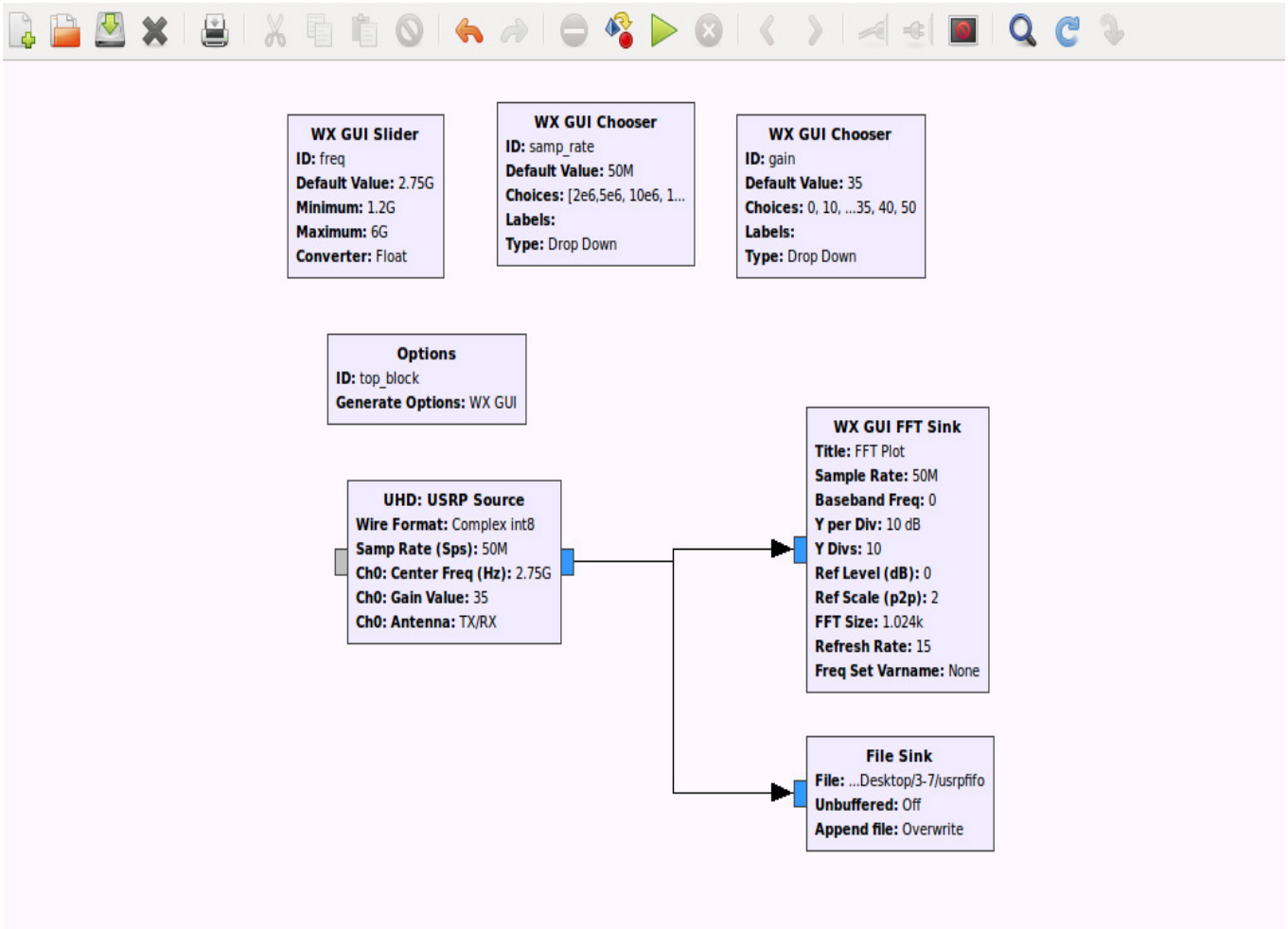


Figure 2.7: GNU radio receiver modules.

Chapter 3

A brief description of OFDM

3.1 Basic OFDM

High data-rate is desired in many applications. However, as the symbol duration is reduced with the increase of the data-rate, the systems using single-carrier modulation suffer from severe ISI caused by the dispersive fading of wireless channels, thereby needing complex equalization techniques. OFDM techniques divides the entire frequency selective fading channel into many narrow band flat fading subcarriers/subchannels in which high-bit-rate data is transmitted in parallel and does not undergo ISI due to the long symbol duration. If the channel is under-spread (the coherence time T_c is much larger than the delay spread T_d) and is therefore approximately time-invariant for a sufficiently long time-scale, then transformation into the frequency domain can be a fruitful approach to communication over frequency-selective channels. This is the basic concept behind OFDM. In the next two paragraphs the advantages and disadvantages of OFDM are summarized [1, 2].

OFDM advantages

- *Immunity to selective fading*, because it divides the overall channel into multiple narrowband signals that are affected individually as flat fading subchannels.
- *Makes efficient use of the spectrum*, by allowing overlap of neighboring subbands.
- *Resilience to interference*, interference appearing on a channel may be bandwidth limited and in this way it will not affect all the sub-channels and consequently it will not affect all the data.
- *Resilient to ISI*, the most important advantage of OFDM is that it is very resilient to inter-symbol and inter-frame interference through use of a cyclic prefix.
- *Simpler channel equalization*, complex equalization techniques are used in a single carrier system.
- *Less sensitive to sample timing offsets*, than single carrier systems are.
- *High transmission bit rates.*

OFDM disadvantages

- *Sensitive to carrier frequency offset and drift*, than single carrier systems are, due to leakage of the DFT.

Therefore, OFDM modulation has been chosen for many standards, including Digital Audio Broadcasting (DAB), terrestrial TV in Europe, and wireless local area network (WLAN). Moreover, it is also an important technique for high data-rate transmission over mobile wireless channels. In the next sections, we will introduce the basic ideas and mathematical models (analog and digital) which are describing an OFDM system.

3.2 Delay Spread and Coherence Bandwidth

Before we move to further description of an OFDM system we present an important general parameter of a wireless system, which is the multipath delay spread T_d, defined as the difference in propagation time between the longest and shortest path.
For a time-varying channel, we have

$$T_d := \max_{i,j} |\tau_i(t) - \tau_j(t)|. \tag{3.1}$$

Respectively, for a time-invariant channel we have

$$T_d := \max_{i,j} |\tau_i - \tau_j|. \tag{3.2}$$

If in a wireless system or cell the distance (line of sight) between the transmitter and the receiver is a few kilometers, then it is very unlikely to have path lengths that differ by more than 300 to 600 meters. Thus, path delays of one or two μs are created. As cells become smaller, T_d also shrinks. Typical wireless channels are under-spread, which means that the delay spread T_d is much smaller than the coherence time T_c as we mentioned above.

The receiver, in order to perform the demodulation and the detection, has to estimate the values of channel taps. The taps are estimated using the transmitted and received waveforms. The receiver does not have any information about individual path delays and path strengths. All we really need is the aggregate values of gross physical mechanisms such as Doppler spread, coherence time, and multipath spread.

The delay spread of the channel dictates its frequency coherence. Wireless channels change both in time and frequency. The time coherence shows us how quickly the channel changes in time and, similarly, the frequency coherence shows how quickly it changes in frequency. We assume a simple case where the wireless channel is linear time-invariant with impulse response

$$h(\tau) = \sum_i \alpha_i \delta(\tau - \tau_i) \tag{3.3}$$

where a_i is the overall attenuation from the transmitter to the receiver on path i. So the frequency response[2] is

$$H(F) = \int_{-\infty}^{\infty} h(\tau) e^{-j2\pi F\tau} \mathrm{d}\tau = \sum_i \alpha_i e^{-j2\pi f \tau_i} \tag{3.4}$$

The contribution of a particular path has linear phase. For multiple paths, there is a differential phase, $2\pi f(\tau_i - \tau_k)$. This differential phase causes selective fading in frequency. The coherence bandwidth, W_c is given by

$$W_c \approx \frac{1}{T_d}. \tag{3.5}$$

.

When the bandwidth $B \approx \frac{1}{T}$ (where T is the symbol period) of the baseband signal is considerably less than W_c, the channel is usually referred to as *flat fading*. In this case, the delay spread T_d is much less than the symbol period T , and a single channel tap is sufficient to represent the channel. When the bandwidth B is much larger than W_c , the channel is said to be *frequency-selective*. In this case, the delay spread T_d is much larger than the symbol period T and the channel description needs more than one taps. We can notice that flat or frequency-selective fading is not a property of the channel alone, but of the relationship between the bandwidth B and the coherence bandwidth W_c. [2, 3]

3.3 Analog OFDM System Model

An OFDM carrier signal is the sum of a number of orthogonal subcarriers, with baseband data on each subcarrier being modulated using quadrature amplitude modulation (QAM) or phase-shift keying (PSK). In an OFDM system more than one signal is carried over a relatively wide bandwidth channel using a separate carrier frequency for each signal. The carrier frequencies are spaced adequately so that the signals do not overlap. Guard bands/cyclic prefix are placed between the signals to ensure that the carriers were orthogonal, so that the carrier interference in the receiver can be avoided.

Let $\{s_k\}_{k=0}^{N-1}$ be the complex symbols to be trasmitted by OFDM modulation. The OFDM signal can be expressed as [1]

$$s(t) = \sum_{k=0}^{N-1} s_k \mathrm{e}^{j2\pi f_k t} = \sum_{k=0}^{N-1} s_k \varphi_k(t), \quad 0 \le t \le T_s$$

where:

$$f_k = f_0 + k\Delta f \quad \text{and} \quad \varphi_k(t) = \begin{cases} e^{j2\pi f_k t}, & \text{if } 0 \le t \le T_s, \\ 0, & \text{otherwise,} \end{cases}$$

for $k = 0, 1, \ldots, N-1$, where

[2]Similar way for time-varying channels

s_k : the k-th complex symbol to be transmitted,

N : the number of symbols,

Δf : subcarrier space of OFDM,

T_s : symbol duration of OFDM,

f_k : the frequency of the k-th subcarrier.

From the *orthogonality condition*, we have that the relation between T_s and Δf has to be $T_s \Delta f = 1$ in order the receiver to detect and demodulate the OFDM signal. This practically means that the duration of the symbols must be long enough. Because of the orthogonality condition, we have

$$\begin{aligned} &\frac{1}{T_s}\int_0^{T_s} \varphi_k(t)\varphi_l^*(t)\mathrm{d}t \\ &= \frac{1}{T_s}\int_0^{T_s} (e^{j2\pi f_k t})(e^{j2\pi f_l t})^*\mathrm{d}t \\ &= \frac{1}{T_s}\int_0^{T_s} e^{j2\pi(f_k - f_l)t}\mathrm{d}t \\ &= \frac{1}{T_s}\int_0^{T_s} e^{j2\pi(k-l)\Delta f t}\mathrm{d}t \\ &= \delta[k-l] \end{aligned} \tag{3.6}$$

where $\delta[k-l]$ is the delta function defined as

$$\delta[n] = \begin{cases} 1, & n = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Equation (3.6) shows that $\{\varphi_k(t)\}_{k=0}^{N-1} = \left\{e^{j2\pi f_k t}\right\}_{k=0}^{N-1}$ is a set of orthogonal functions. Using this property and assuming ideal channel, the OFDM signal can be demodulated by

$$\begin{aligned} &\frac{1}{T_s}\int_0^{T_s} s(t)e^{-j2\pi f_k t}\mathrm{d}t \\ &= \frac{1}{T_s}\int_0^{T_s} \sum_{l=0}^{N-1} s_l(e^{j2\pi f_l t})(e^{j2\pi f_k t})^*\mathrm{d}t \\ &= \sum_{l=0}^{N-1} s_l \delta[l-k] \\ &= s_k. \end{aligned} \tag{3.7}$$

3.4 Digital OFDM System Model

In this section, we assume ideal timing and frequency synchronization. Consider a wideband wireless channel with a discrete-time impulse response of length L given by $h_l, l = 0, \ldots, L-1$. We can prove without loss of generality that in an OFDM system the transmitter can be implemented using IDFT and respectively the receiver using DFT. We assume that the channel remains constant over the time period of interest. If $x[m]$ is the input at the time instant m, the discrete-time baseband model is given from the following convolution [2, 3, 7].

$$y[m] = \sum_{l=0}^{L-1} h_l x[m-l] + w[m], \quad m = 0, \ldots, N+L-1. \tag{3.8}$$

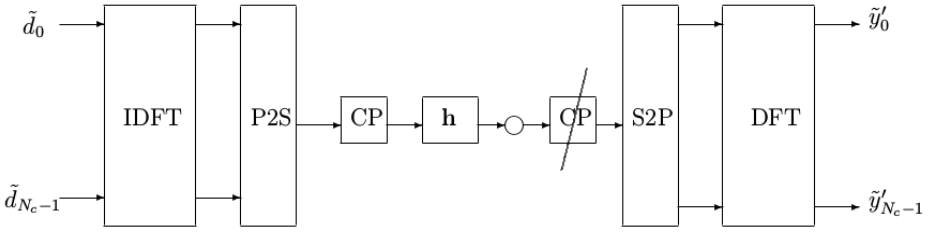


Figure 3.1: Digital OFDM System Model.

Let the data block of length N be

$$\tilde{\mathbf{d}} = [\tilde{d}_0, \ldots, \tilde{d}_{N-1}]^T.$$

Taking the inverse Discrete Fourier Transform (IDFT) of $\tilde{\mathbf{d}}$, the data block is expressed as

$$\mathbf{d} = IDFT(\tilde{\mathbf{d}}) = [d_0, \ldots, d_{N-1}]^T.$$

Using $\mathbf{d}$, we construct the vector $\mathbf{x}$, by inserting a cyclic prefix, of length L (equal or bigger of the channel length),

$$\mathbf{x} = \begin{bmatrix} d_{N-L} \\ \vdots \\ d_{N-1} \\ d_0 \\ \vdots \\ d_{N-1} \end{bmatrix} = \begin{bmatrix} x_0 \\ \vdots \\ \\ \\ \vdots \\ x_{N+L-1} \end{bmatrix}.$$

Therefore, the equation (3.8) can be written as follows

$$y_m = \sum_{l=0}^{L-1} h_l x_{m-l} + w_m, \quad m = 0, \ldots, N+L-1.$$

The ISI extends over the first L symbols and the receiver ignores it (removes CP) by considering the output over the time interval $m \in [L, N+L-1]$, see Figure 3.2. Due to the additional cyclic prefix, the output over this time interval (of length N) is

$$y[m] = \sum_{l=0}^{L-1} h_l d[(m - L - l)\ modulo\ N] + w[m]. \tag{3.9}$$

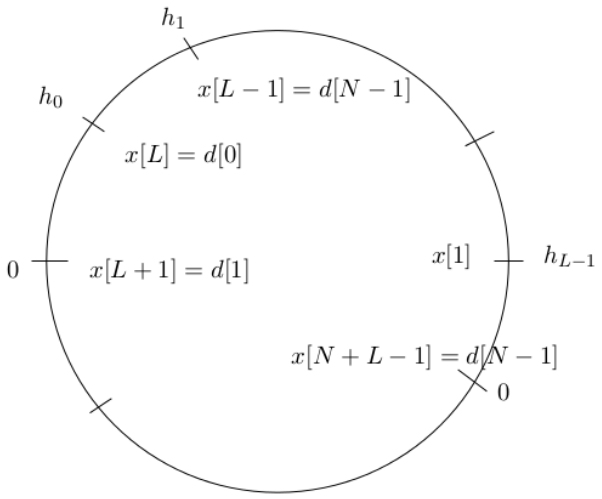


Figure 3.2: Convolution between the channel $\mathbf{h}$ and the input $\mathbf{x}$ formed from the data symbols $\mathbf{d}$ by adding a cyclic prefix.

After ignoring the first L output symbols. We use the N output symbols y_m, $m = L, \ldots, N+L-1$ to construct the vector $\mathbf{y}^{'}$

$$\mathbf{y}^{'} = \begin{bmatrix} y_0^{'} \\ \vdots \\ y_{N-1}^{'} \end{bmatrix} = \begin{bmatrix} y_L \\ \vdots \\ y_{N+L-1} \end{bmatrix}.$$

and the vector of channel with length N

$$\mathbf{h} = \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{L-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Now, equation (3.9) can be written as

$$\mathbf{y}^{'} = \mathbf{d} \otimes_N \mathbf{h} + \mathbf{w},$$

where $\mathbf{w} = [w_L, \ldots, w_{N+L-1}]^T$ and $\mathbf{d} \otimes_N \mathbf{h}$ is the circular convolution as mentioned before, of length N, of vectors $\mathbf{d}$ and $\mathbf{h}$. We will prove the above assertion as follows. Consider,

$$y_0^{'} = y_L = \sum_{l=0}^{L-1} h_l x_{L-l} = h_0 d_0 + \sum_{l=1}^{L-1} h_l d_{N-l}.$$

So, the first term of the circular convolution is expressed as

$$(\mathbf{d} \otimes_N \mathbf{h})_0 = h_0 d_0 + \sum_{l=1}^{L-1} h_l d_{N-l}.$$

Respectively, we can prove the remaining relations and conclude that

$$\mathbf{y}^{'} = \mathbf{d} \otimes_N \mathbf{h} + \mathbf{w}$$

The discrete Fourier transform (DFT) of d is defined to be[3]

$$\tilde{d_n} := \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} d[m] e^{\frac{-j2\pi nm}{N}}, \quad n = 0, \ldots, N-1$$

Taking the Discrete Fourier Transform (DFT) of both sides, we obtain

$$\begin{aligned} \tilde{\mathbf{y}}^{'} &= DFT(\mathbf{y}^{'}) = DFT(\mathbf{d} \otimes_N \mathbf{h} + \mathbf{w}) \\ &= DFT(\mathbf{d} \otimes_N \mathbf{h}) + DFT(\mathbf{w}) \\ &= \sqrt{N} DFT(\mathbf{h}) \odot DFT(\mathbf{d}) + DFT(\mathbf{w}) \end{aligned} \tag{3.10}$$

where

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \odot \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 y_1 \\ \vdots \\ x_n y_n \end{bmatrix}$$

is the element-wise or Hadamard product. Therefore,

$$\tilde{y_n^{'}} = \tilde{h}_n \tilde{d}_n + \tilde{w}_n, \quad n = 0, \ldots, N-1,$$

where $\tilde{h}_n = \sum_{l=0}^{L-1} h_l e^{-\frac{j2\pi ln}{N}}$, $n = 0, \ldots, N-1$ which is equal to the frequency response of the channel at frequency $f = \frac{nW}{N}$. Thus, using OFDM, we convert a wideband channel into a set of N parallel narrowband channels as shown in Figure 3.3. As a result, no equalization is required, which has a high computational cost, but a symbol-by-symbol decision for each information symbol. In Figure 3.1, the overall process as described above is depicted.

[3]We should multiple with $\frac{1}{\sqrt{N}}$ so much in DFT as in IDFT.

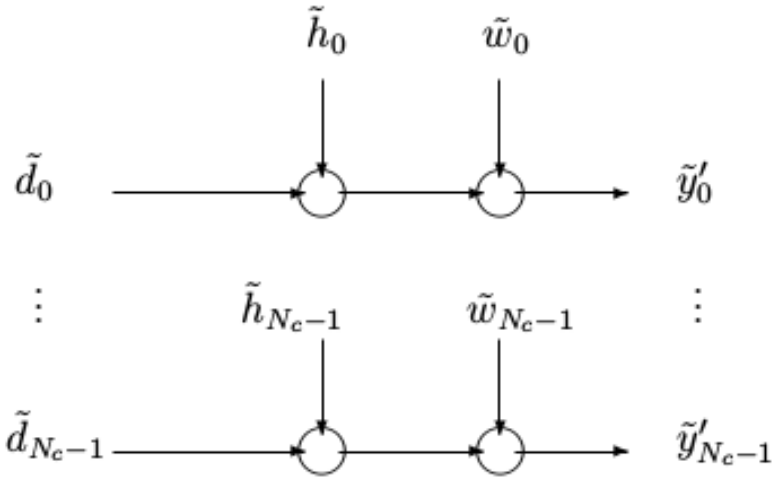


Figure 3.3: OFDM converts a wideband channel into a set of N parallel narrowband channels.

Chapter 4

Structure of OFDM

In this chapter, we will describe the overall structure of the OFDM system that we built. Specifically, we will mention the structure and the operations of transmitter and receiver. Such operations are the data package creation, the time synchronization, the channel estimation, etc.

4.1 Transmitter

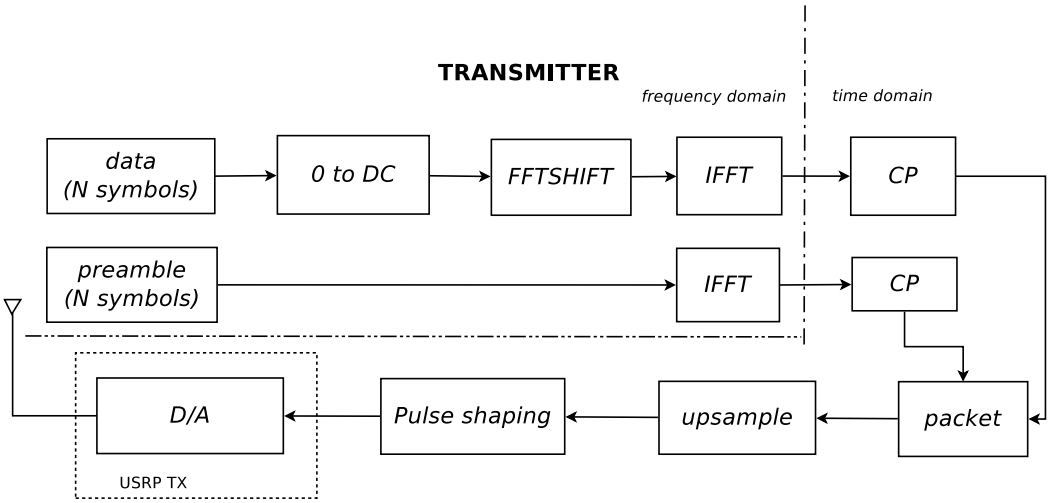


Figure 4.1: OFDM transmitter structure.

In this section, we will present the operation that an OFDM transmitter does. At first, we generate randomly the symbols that we want to transmit. The complex symbols have length N and are modulated with 4-QAM modulation. We assume that first symbol is in position 0. Then, we modify the symbol in DC position ($\frac{N}{2}$) to be 0 and we apply fftshift with purpose to avoid transmitting in DC frequency which is usually undesirable and unreliable. After these, we apply IFFT to data and add cyclic prefix (CP) of length L at the beginning of the data packet as a guard interval to cope with ISI. In a similar way, we construct the preamble symbols with length N. The preamble are constructed with symbols according to 4-QAM in the even-numbered positions and with zeros in the odd-numbered positions. As a result, when we apply IFFT the preamble have the following structure. The

first half $(0 : \frac{N}{2} - 1)$ is exactly the same as the second half $(\frac{N}{2} : N - 1)$ which is important for the receiver as we shall see later (time & freqeuncy synchronization). Finally, we add cyclix prefix (CP) as we do in data.

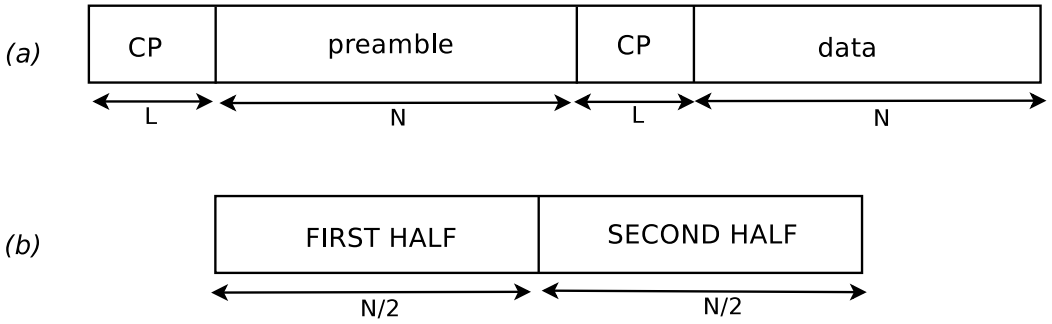


Figure 4.2: (a) Packet of OFDM symbols with length $2(N + L)$, (b) Preamble structure (before the CP) with length N.

Once we create preamble and data, we create the packet by merging preamble and data, as shown in Figure 4.2. Each packet is upsampled with *over* factor and is passed through a pulse shaping filter, which is a square root raised cosine (SRRC) filter. We set g_T as the SRRC filter and x_n as the transmitted symbols.
Thus, the transmitted signals is

$$X(t) = \sum_{n=0}^{N-1} x_n g_T(t - nT).$$

In the end of the transmitted signal, we add a number of zeros in order to avoid interference of consecutive packets so that the receiver can separate them. In Figure 4.1, the operations of the transmitter are depicted. Also, in Figure 4.3, the real and the imaginary part of the transmitted signal is shown.

4.2 Receiver

We will now describe how the receiver works. The overall process is depicted in Figure 4.12.

4.2.1 Packet Detection

When the receiver receive the transmitted signal the first operation is to find where the valid transmission begins. Specifically, the receiver saves data into a buffer which has to process (in real time) in order to find the real transmitted information-data and to separate the noise. This operation is called *packet detection* and is performed with a *Double Sliding Window (DSW)* algorithm. The DSW algorithm

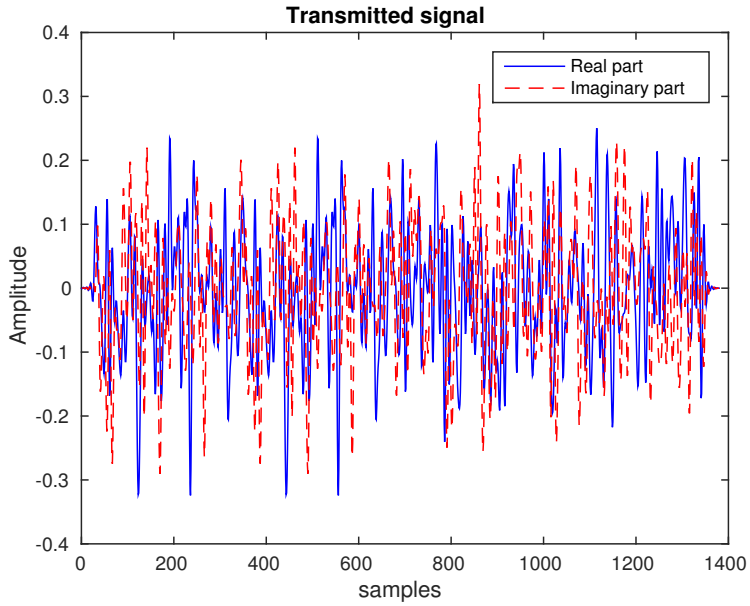


Figure 4.3: Transmitted OFDM Signal.

uses two consecutive sliding windows as depicted in Fig 4.5. Windows have fixed length $W_{len} = floor(\frac{2(N+L)}{8})$ each one, in which N is the number of transmitted symbols and L the length of cyclic prefix as mentioned before. The **first** time, if r is the received signal (in the buffer), we calculate the energy $E[i]$ of symbol i as follows

$$E[i] = |r(i)|^2 = r^2_{real}(i) + r^2_{imag}(i), \tag{4.1}$$

We define as useful signal S always the right window and as noise N_{oi} always the left. Thus, using equation (4.1) we have,

$$S[2W_{len} - 1] = \sum_{k=0}^{W_{len}-1} E(W_{len} + k), \quad \text{right window}$$

and

$$N_{oi}[2W_{len} - 1] = \sum_{k=0}^{W_{len}-1} E(k), \quad \text{left window}$$

Let the ratio M as

$$M = \frac{S[2W_{len} - 1]}{N_{oi}[2W_{len} - 1]}$$

if M is larger than a threshold for $C_{times} = floor(\frac{W_{len}}{2})$ consecutive time instants, we understand that we find useful signal and therefore there is a packet in the start

of the buffer.
If the packet is not, and usually is not, in the start of the buffer we have to move (right) one position/time instant our windows and repeat the overall process as we will describe.
For the new time instant $i = 2W_{len}$ we calculate again the energy as shown in equation (4.1). Now, the signal-right window S is

$$S[i] = S[i-1] + E[i] - E[i - W_{len}], \quad i = 2W_{len} \dots end$$

and the noise-left window N_{oi} is

$$N_{oi}[i] = N_{oi}[i-1] + E[i - W_{len}] - E[i - 2W_{len}], \quad i = 2W_{len} \dots end$$

where $rspb$ is the number of samples in the buffer. As we did previously, we calculate the ratio M for the new position i and if it is larger than the threshold $thresh$ for C_{times} consecutive time instants, we decide that a packet had been detected.
We repeat the above process until the end of the buffer. In the Figure 4.4, the received signal is depicted. When we find the packet we perform matched filtering.

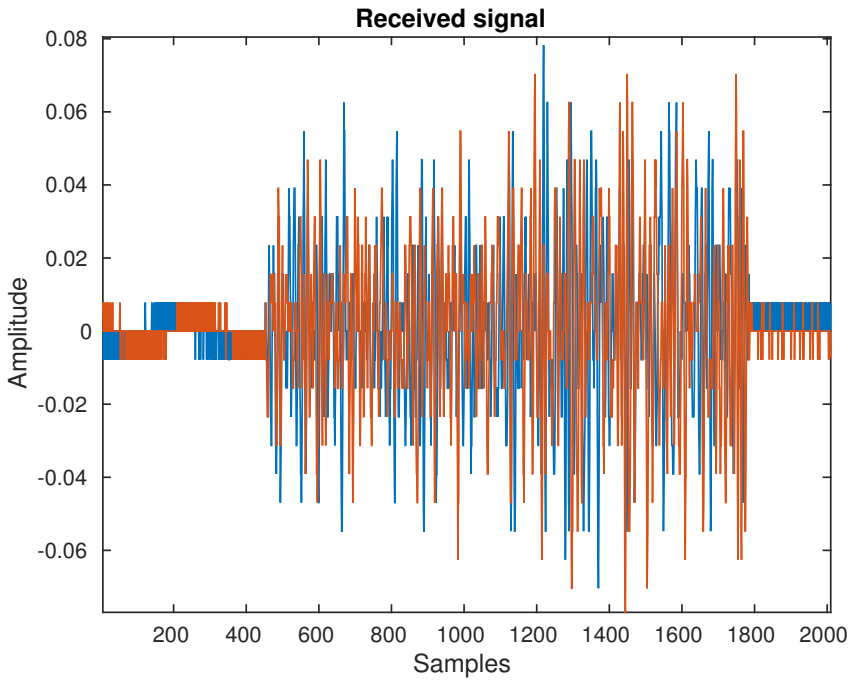


Figure 4.4: Received OFDM Signal.

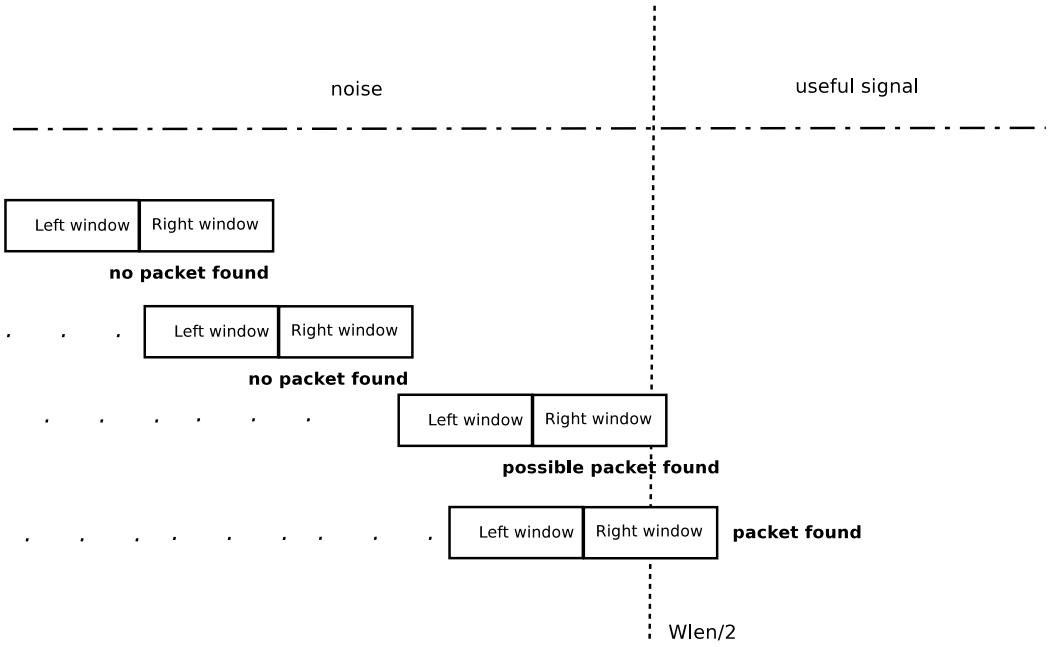


Figure 4.5: Double Sliding Window packet detection.

4.2.2 Received Signal

Before we move to further analysis of the receiver operations, we will describe a detailed signal model for OFDM. Specifically, The transmitted baseband OFDM signal that will pass through the channel is [4]

$$X(t) = \sum_n x_n g_T(t - nT), \tag{4.2}$$

where $g_T(t)$ is the pulse shaping filter (SRRC). Also, we assume that the continuous time channel is

$$c(t) = \sum_l c_l \delta(t - \tau_l) \tag{4.3}$$

Using (4.2) and (4.3), the received signal, affected by noise, in channel output is

$$\begin{aligned} Y(t) &= c(t) * X(t) + W(t) \\ &= c(t) * \left(\sum_n x_n g_T(t - nT) \right) + W(t) \\ &= \sum_n x_n h(t - nT) + W(t), \end{aligned} \tag{4.4}$$

where $h(t) = c(t) * g_T(t)$ and $W(t)$ the noise. If the signal has been affected by analog CFO ΔF and phase offset ϕ, then the received signal from equation (4.4)

becomes

$$Y(t) = e^{j(2\pi\Delta Ft+\phi)} \sum_n x_n h(t - nT) + W(t). \tag{4.5}$$

If we sample signal $Y(t)$ with period $T_s = \frac{T}{over}$, where *over* is a positive integer (the oversampling factor), we obtain the *sample-spaced* sequence

$$\begin{aligned} y_k = Y(kT_s) &= e^{j(2\pi\Delta FkT_s+\phi)} \sum_n x_n h(kT_s - nT) + W(kT_s) \\ &= e^{j\phi} e^{j2\pi\Delta fk} \sum_n x_n h_{k,n} + w_k, \end{aligned} \tag{4.6}$$

where $\Delta f := \Delta F T_s$ and $h_{k,n} := h(kT_s - nT)$. Then, y_k can be expressed as

$$y_k = e^{j\phi} e^{j2\pi\Delta fk} r_k + w_k,$$

where

$$r_k := \sum_n x_n h_{k,n}.$$

Thus, the discrete time baseband equivalent signal model with the additive white Gaussian noise of equation (3.8) can be expressed as

$$y_k = y[k] = e^{j\phi} e^{j2\pi\Delta fk} \sum_{l=0}^{L-1} h_l x[k-l] + w[k], \quad k = 0, \ldots, N + L - 1 \tag{4.7}$$

where L is the length of cyclic prefix (greater than channel length) and N is the symbols length.

4.2.3 Time synchronization and CFO cancellation

OFDM, like any other digital communication system, requires synchronization. The OFDM receiver needs to perform two important synchronization tasks.

The first is the *time synchronization* in which it has to identify the starting position of each received OFDM block in order to apply the N-point FFT. The goal is to have an efficient time-offset estimator with reasonable accuracy in order to avoid residual phase error and therefore to make the overall system efficient.

The second is the *frequency synchronization* in which it has to estimate and align its local carrier frequency as closely as possible to the transmitter carrier frequency. In OFDM systems, the term orthogonal dictates an exact mathematical relationship between the frequencies of the subcarriers. The orthogonality of the OFDM-based systems depends on the situation that transmitter and receiver work with the same frequency reference. If this doesn't happen, the orthogonality of the subcarriers will be destroyed leading to inter-carrier interference (ICI). Strictly speaking, all of the OFDM communication systems are sensitive to the frequency synchronization in form of carrier frequency offset (CFO) which occurs due to Doppler shift and the differences between the oscillators at the transmitter and

receiver side caused by the system noise temperature and oscillator inaccuraciess. As it is understood the CFO degrades severely the performance of the OFDM systems. Therefore, it is important to estimate and cancel it.

Coarse Time Synchronization

The proposed method is multi-stage. In this method, we combine a simple autocorrelation technique and a cross-correlation technique to achieve an efficient time synchronization and an enhanced estimation performance [5]. Moreover, we exploit the fact that the preamble have two identical parts in time-domain. Specifically, the first half is equal to the second half. This structure of preamble is chosen since it provides for low-complexity coarse timing, fractional frequency estimation. We should notice that the receiver has complete knowledge of the preamble symbols (after ifft-in time domain).
Let the preamble vector be **a** with

$$\mathbf{a} = \left[a_0 \dots a_{\frac{N}{2}-1} \quad a_{\frac{N}{2}} \dots a_{N-1}\right] \tag{4.8}$$

where N is the number of preamble symbols. The coarse timing estimate is derived as follows. First, we calculate the statistic

$$P_{Sch}(d) = \sum_{k=0}^{\frac{N}{2}-1} y^*(d+k)y(d+k+N/2), \quad d = 1 \dots \tag{4.9}$$

where y_k is the received signal affected by CFO and P_{Sch} is Schmidl's autocorrelation stage. Then, we calculate the statistic

$$M_c(d) = \frac{1}{L+1} \sum_{k=0}^{L} |P_{Sch}(d-k)|^2, \quad d = 1 \dots \tag{4.10}$$

where Schmidl's autocorrelation (4.9) is computed over the length L of cyclic prefix in order to eliminate its uncertainly plateau. Therefore, our estimate for the position of the first preamble symbols is

$$\hat{d}_c = \underset{d}{\text{argmax}}\{M_c(d)\},$$

In this way, we achieve a coarse timing metric M_c whose peak indicates the coarse timing estimate $\hat{d}_c$, see Figure 4.6.
Subsequently, we can use $\hat{d}_c$ for frequency synchronization as we will see in the next paragraph.

CFO estimation and cancellation

Once we find $\hat{d}_c$, we can use the received sequence of the preamble symbols a_{rec} vector in order to estimate the CFO. The symbol-spaced received preamble sequence

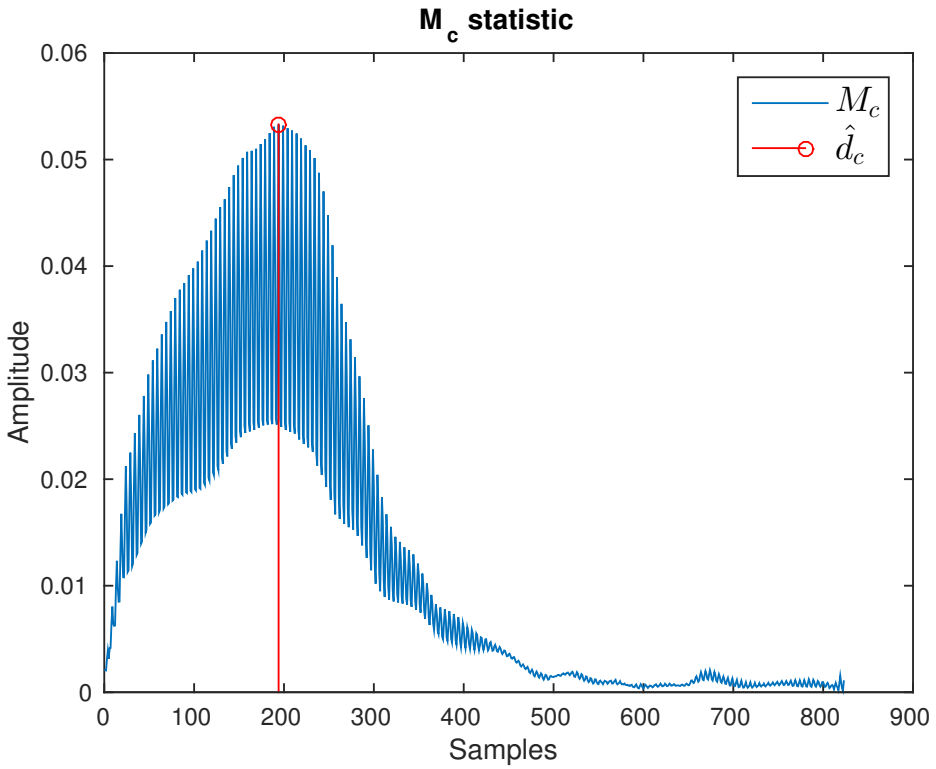


Figure 4.6: M_c statistic.

$\{a_{rec}\}_{k=0}^{N-1}$ (according to equation (4.5)) is

$$\begin{aligned} a_{rec}(k) &= y_{\hat{d}_c+k} = Y(\hat{d}_c T_s + kT) = Y(t)|_{t=\hat{d}_c T_s+kT} \\ &= e^{j(2\pi\Delta F(\hat{d}_c T_s+kT)+\phi)} \sum_n x_n h(\hat{d}_c T_s + kT - nT) + W(\hat{d}_c T_s + kT) \\ &= e^{j(2\pi\Delta f\hat{d}_c+\phi)} e^{j2\pi(\Delta F\cdot T)k} \sum_n x_n h_{k,n}^{\hat{d}_c} + w_k^{'}, \end{aligned} \tag{4.11}$$

where $h_{k-n}^{\hat{d}_c} := h(\hat{d}_c T_s + kT - nT)$. If we incorporate the constant term $e^{j(2\pi\Delta f\hat{d}_c+\phi)}$ into the channel and, for simplicity, denote

$$h_{k,n}^{'} = e^{j(2\pi\Delta f\hat{d}_c+\phi)} h_{k,n}^{\hat{d}_c}, \tag{4.12}$$

then, for $k = 0, \ldots, N-1$,

$$a_{rec}(k) = e^{j2\pi(\Delta F\cdot T)k} \sum_n x_n h_{k,n}^{'} + w_k^{'} = e^{j2\pi\Delta f^{'}k}\, r_k^{'} + w_k^{'}, \tag{4.13}$$

where $\Delta f^{'} = \Delta F{\cdot}T$ and $r_k^{'} = \sum_n x_n h_{k,n}^{'}$. In order to estimate the CFO, the receiver exploits the special structure of the OFDM preamble. As we have mentioned, the

second half of the preamble is equal to the first half. It is important to note that the main difference between the two halves of the received preamble will be a phase shift.
Therefore, using the (4.13) the first half of the received preamble sequence is

$$z_k = \{a_{rec}\}_{k=0}^{\frac{N}{2}-1} = e^{j2\pi\Delta f^{'} k} r_k^{'} + w_k^{'}, \quad k = 0, \ldots, \frac{N}{2} - 1. \tag{4.14}$$

Similarly the second half is

$$\{a_{rec}\}_{k=\frac{N}{2}}^{N-1} = e^{j2\pi\Delta f^{'} k} r_k^{'} + w_k^{'}, \quad k = \frac{N}{2}, \ldots, N-1 \tag{4.15}$$

For simplicity, we can write the second half as follows

$$z_{k+\frac{N}{2}} = \{a_{rec}\}_{k=\frac{N}{2}}^{N-1} = e^{j2\pi\Delta f^{'}(k+\frac{N}{2})} r_{k+\frac{N}{2}}^{'} + w_{k+\frac{N}{2}}^{'}, \quad k = 0, \ldots, \frac{N}{2} - 1, \tag{4.16}$$

where $r_k^{'}$ and $r_{k+\frac{N}{2}}^{'}$ are identical. Consequently, using equation (4.14) and (4.16), we have,

$$\begin{aligned}
&z_k^* z_{k+\frac{N}{2}} \\
&= (e^{j2\pi\Delta f^{'} k} r_k^{'} + w_k^{'})^* (e^{j2\pi\Delta f^{'}(k+\frac{N}{2})} r_{k+\frac{N}{2}}^{'} + w_{k+\frac{N}{2}}^{'}) \\
&= (e^{-j2\pi\Delta f^{'} k} (r_k^{'})^* + (w_k^{'})^*)(e^{j2\pi\Delta f^{'}(k+\frac{N}{2})} r_{k+\frac{N}{2}}^{'} + w_{k+\frac{N}{2}}^{'}) \\
&= e^{-j2\pi\Delta f^{'} k} e^{j2\pi\Delta f^{'}(k+\frac{N}{2})} (r_k^{'})^* r_k^{'} + \\
&+ e^{-j2\pi\Delta f^{'} k} (r_k^{'})^* w_{k+\frac{N}{2}}^{'} + e^{j2\pi\Delta f^{'}(k+\frac{N}{2})} r_k^{'} (w_k^{'})^* + (w_k^{'})^* w_{k+\frac{N}{2}}^{'} \\
&= e^{j2\pi\Delta f^{'}(k+\frac{N}{2}-k)} |r_k^{'}|^2 + \tilde{w}_k \\
&= e^{j\pi\Delta f^{'} N} |r_k^{'}|^2 + \tilde{w}_k
\end{aligned} \tag{4.17}$$

where

$$\tilde{w}_k = e^{-j2\pi\Delta f^{'} k} (r_k^{'})^* w_{k+\frac{N}{2}}^{'} + e^{j2\pi\Delta f^{'}(k+\frac{N}{2}} r_k^{'} (w_k^{'})^* + (w_k^{'})^* w_{k+\frac{N}{2}}^{'}.$$

Before we move further, we should observe that if we ignore the noise we have

$$\begin{aligned}
&\arg\left(\sum_{k=0}^{\frac{N}{2}-1} z_k^* z_{k+\frac{N}{2}}\right) = \arg\left(\sum_{k=0}^{\frac{N}{2}-1} e^{j\pi\Delta f^{'} N} |r_k^{'}|^2\right) \\
&= \arg\left(e^{j\pi\Delta f^{'} N} \sum_{k=0}^{\frac{N}{2}-1} |r_k^{'}|^2\right) = \pi\Delta f^{'} N.
\end{aligned} \tag{4.18}$$

Therefore, we can estimate the CFO $\hat{\Delta}f$ as

$$\hat{\Delta}f = \frac{1}{\pi N} \arg\left(\sum_{k=0}^{\frac{N}{2}-1} z_k^* z_{k+\frac{N}{2}} \right). \tag{4.19}$$

Once $\hat{\Delta}f$ has been computed, it can be used to eliminate the effect of CFO in received signal by creating a sequence

$$y_k^{corr} = e^{-j2\pi\hat{\Delta}fk} \cdot y_k \tag{4.20}$$

where y_k^{corr} is the signal with canceled CFO [4].

Fine time synchronization

Once the frequency offset (CFO) has been corrected in the received signal, we will perform a cross-correlation technique for fine time synchronization. The restricted cross-correlation stage is summarized as follows [5]

$$P_x(d) = \sum_{k=0}^{N-1} y^{corr}(d+k)\ \mathbf{a}^*(k), \quad d = 1, \ldots, \tag{4.21}$$

and

$$M_{opt}(d) = |P_x(d)|^2 \cdot M_c(d) \quad d = 1, \ldots, \tag{4.22}$$

where y^{corr} is the corrected from CFO signal as we mentioned in equation (4.20), $\mathbf{a}$ is the known transmitted preamble, see (4.8), $P_x(d)$ is the restricted cross-correlation, M_c is the statistic we calculated before using the equation 4.10 and $M_{opt}(d)$ is the filtered timing metric. The statistic $M_{opt}(d)$ for flat fading channel is depicted in Figure 4.7, and for frequency selective channel is depicted in Figure 4.8. We observe that the statistic $M_{opt}(d)$ is the composite channel.

In order to decide the optimal starting point we used a *energy window* of length L to ensure that all relevant timing points that could be ideal timing were tracked. Thus, the window G is computed as

$$G(d) = \sum_{k=d}^{d+(L-1)} |M_{opt}(k)|^2, \quad d = 1, \ldots\ . \tag{4.23}$$

Using equation (4.23) we decide that the optimal start of the packet $\hat{d}_{opt}$ is the point in which we have the maximum energy and consequently it is the first preamble symbol without CP. However, in channel estimation we will use the preamble with CP for this reason we move back L symbols. Therefore, we define as

$$\hat{d}_{opt} = \underset{d}{\text{argmax}}\{G(d)\}$$

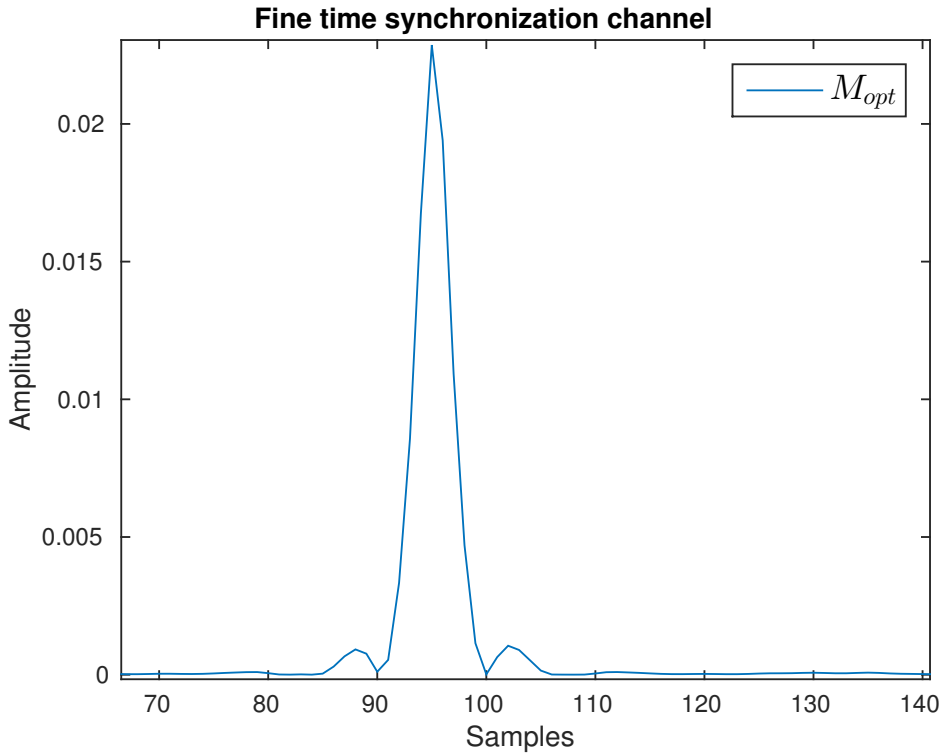


Figure 4.7: M_{opt} statistic for flat fading channel.

and we move back $\hat{d}_{opt} = \hat{d}_{opt} - L$.
Now, we can detect that the received symbol-spaced sequence of length $2N + 2L$ (preamble with CP, data with CP) is

$$\begin{aligned} R_0^{symbols} &= y^{corr}_{\hat{d}_{opt}}, \\ R_1^{symbols} &= y^{corr}_{\hat{d}_{opt}+1}, \\ &\vdots \\ R_{2(N+L)-1}^{symbols} &= y^{corr}_{\hat{d}_{opt}+2(N+L)-1}. \end{aligned} \tag{4.24}$$

We can express this sequence as

$$R_k^{symbols} = y^{corr}(\hat{d}_{opt}T_s + kT) = y^{corr}(t)|_{t=\hat{d}_{opt}T_s+kT}, \quad k = 0, \ldots, 2(N+L)-1,$$

where T is the symbol period.

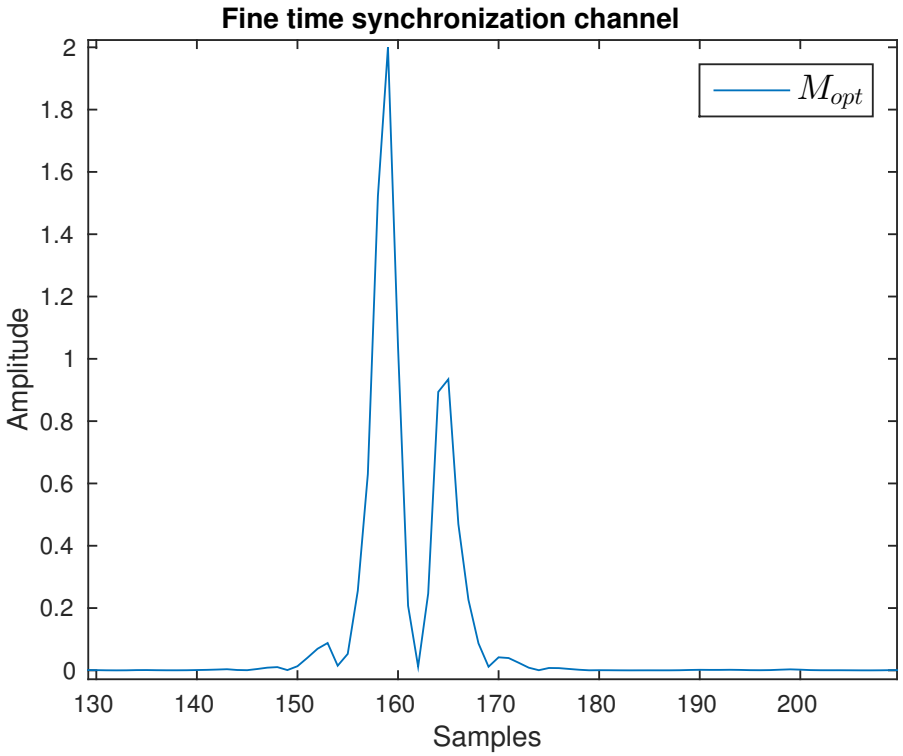


Figure 4.8: M_{opt} statistic for frequency selective channel

4.2.4 Channel Estimation

Once CFO has been canceled and we are synchronized, we have to estimate the channel using the preamble. The channel can be either flat fading as shown in Figure 4.9 or either frequency selective as shown in Figure 4.10. For this reason, our technique has to cover both cases. The received preamble sequence with CP in the receiver is the following

$$\mathbf{R}^{pr} = \{R_k^{symbols}\}_{k=0}^{k=N+L-1}.$$

Also, the receiver as we mentioned before has complete knowledge of the transmitted preamble symbols, vector $\mathbf{a}$, as shown in (4.8). So the receiver can create the known preamble with CP vector, $\mathbf{b}$. We assume for simplicity that $L = 4$. Then, we have

$$\mathbf{b} = [a_{N-L}, a_{N-L+1}, a_{N-L+2}, a_{N-L+3}, a_1, a_2, \ldots, a_{N-1}]$$

where $\mathbf{b}$ has length $N + L$. Before we move further lets mention that generally the input-output relation for a **flat fading channel** is

$$\mathbf{y} = \mathbf{x}h_0 + \mathbf{w}$$

where h_0 is only one value thus we can write the above relation as

$$\begin{aligned} y_0 &= h_0 x_0 + w_0 \\ y_1 &= h_0 x_1 + w_1 \\ &\vdots \\ y_{N-1} &= h_0 x_{N-1} + w_{N-1} \end{aligned} \tag{4.25}$$

and for **frequency selective channel** the input-output relation is

$$\mathbf{y} = \mathbf{x} * \mathbf{h} + \mathbf{w}$$

where $\mathbf{h}$ the channel has $L = 4$ values thus we can write the above relation as

$$\begin{aligned} y_0 &= h_0 x_0 + w_0 \\ y_1 &= h_0 x_1 + h_1 x_0 + w_1 \\ y_2 &= h_0 x_2 + h_1 x_1 + h_2 x_0 + w_2 \\ y_3 &= h_0 x_3 + h_1 x_2 + h_2 x_1 + h_3 x_0 + w_3 \\ &\vdots \\ y_{N-1} &= h_0 x_{N-1} + h_1 x_{N-2} + h_2 x_{N-3} + h_3 x_{N-4} + w_{N-1} \end{aligned} \tag{4.26}$$

Therefore, in our case the input-output relation for a frequency selective channel can be written as follows

$$\begin{bmatrix} R_0^{pr} \\ R_1^{pr} \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ R_{N+3}^{pr} \end{bmatrix} = \begin{bmatrix} a_{N-4} & 0 & 0 & 0 \\ a_{N-3} & a_{N-4} & 0 & 0 \\ a_{N-2} & a_{N-3} & a_{N-4} & 0 \\ a_{N-1} & a_{N-2} & a_{N-3} & a_{N-4} \\ a_0 & a_{N-1} & a_{N-2} & a_{N-3} \\ a_1 & a_0 & a_{N-1} & a_{N-2} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N-5} & a_{N-6} & a_{N-7} & a_{N-8} \\ a_{N-4} & a_{N-5} & a_{N-6} & a_{N-7} \\ a_{N-3} & a_{N-4} & a_{N-5} & a_{N-6} \\ a_{N-2} & a_{N-3} & a_{N-4} & a_{N-5} \\ a_{N-1} & a_{N-2} & a_{N-3} & a_{N-4} \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ w_{N+L-1} \end{bmatrix}$$

which can be written in matrix-vector form

$$\mathbf{R}^{pr} = \mathbf{B}\mathbf{h} + \mathbf{w}$$

where $\mathbf{R}^{pr}$ is the received preamble sequence in time domain, $\mathbf{B}$ is the toeplitz matrix constructed by vector $\mathbf{b}$ for $L = 4$ (respectively we can create the matrix for any L), $\mathbf{h}$ is the vector of channels and $\mathbf{w}$ is the vector of noise.

As we can observe, the output is the same as in equation (4.26) for frequency selective channels and also it is the same as in equation (4.25) where h_1, h_2, h_3 are equal to zero. As a result, we can estimate the channel either for flat fading channels, or for frequency selective.

As we mentioned, $\mathbf{B}$ is known to receiver (because the receiver has the knowledge to create it), so we can estimate the channel using Least Squares (LS)

$$\mathbf{h}_{LS} = \left(\mathbf{B}^H\mathbf{B}\right)^{-1}\mathbf{B}^H\mathbf{R}^{pr}, \tag{4.27}$$

where $\cdot^H$ denotes Hermitian transpose and $\mathbf{h}_{LS}$ is the channel estimate in time domain (before FFT) with length L.

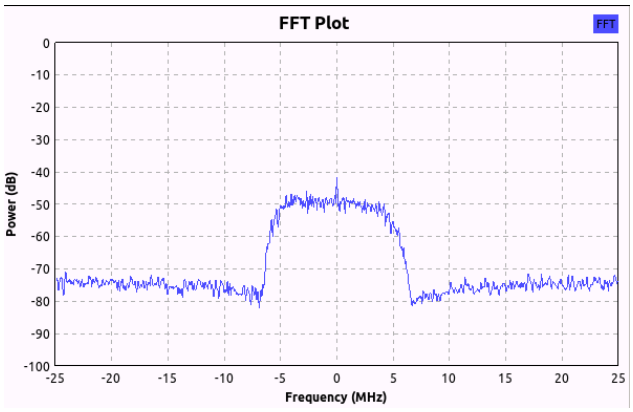


Figure 4.9: Spectrum of the received signal for flat fading channel (one tap).

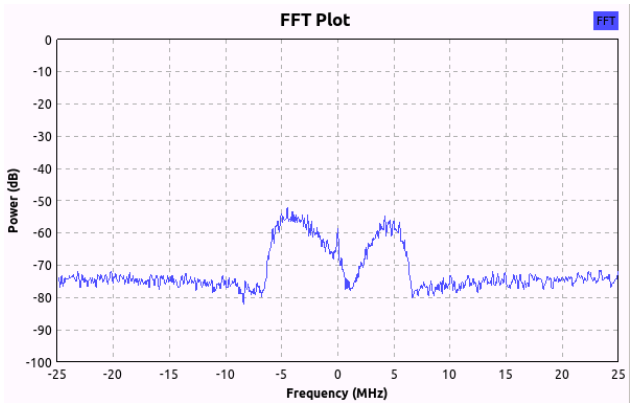


Figure 4.10: Spectrum of the received signal for for frequency selective with two taps.

4.2.5 Detection

As we described before, using OFDM we convert a wideband channel into a set of N parallel narrowband channels. As a result, no complicated equalization is required, but a symbol-by-symbol decision for each information symbol. Therefore, the received information symbols (data) without CP in time domain are

$$\mathbf{R}^{da} = \{R_k^{symbols}\}_{k=N+2L}^{k=2(N+L)-1}.$$

We have to make the decision in frequency domain. For this reason, first we apply FFT in $\mathbf{R}^{da}$ and we take $\tilde{\mathbf{R}}^{da}$ and then we apply FFT to channel vector $\mathbf{h}_{LS}$ in N-points and we take $\mathbf{H}_{LS}$, then we apply fftshift to $\tilde{\mathbf{R}}^{da}$ and to $\mathbf{H}_{LS}$ in order to cancel the fftshift that the transmitter applied. Now, we can demodulate as follows

$$\mathbf{S}^{est} = \frac{1}{|\mathbf{H}_{LS}|^2}\mathbf{H}_{LS}^{*}\tilde{\mathbf{R}}^{da}$$

and we decide using the sign for the real and the imaginary part because we used 4-QAM and the symbols have the same probability. We should notice that receiver didn't make a decision for the symbols in DC carrier because the transmitter sent zero to this position. The constellation of the received information symbols before the decision is depicted in Figure 4.11.

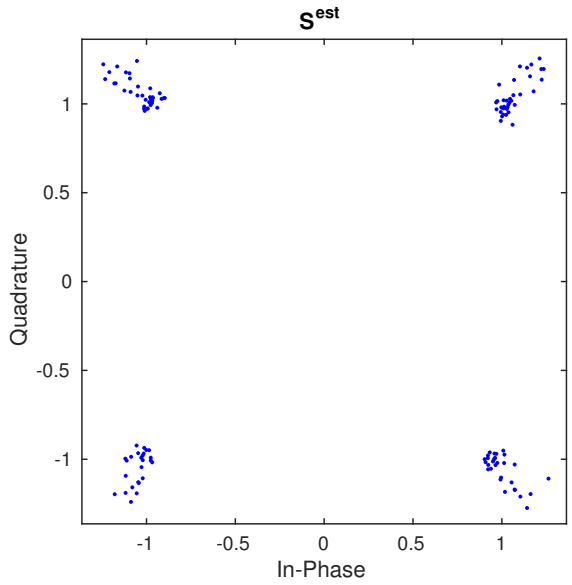


Figure 4.11: Constellation of received information symbols before the decision.

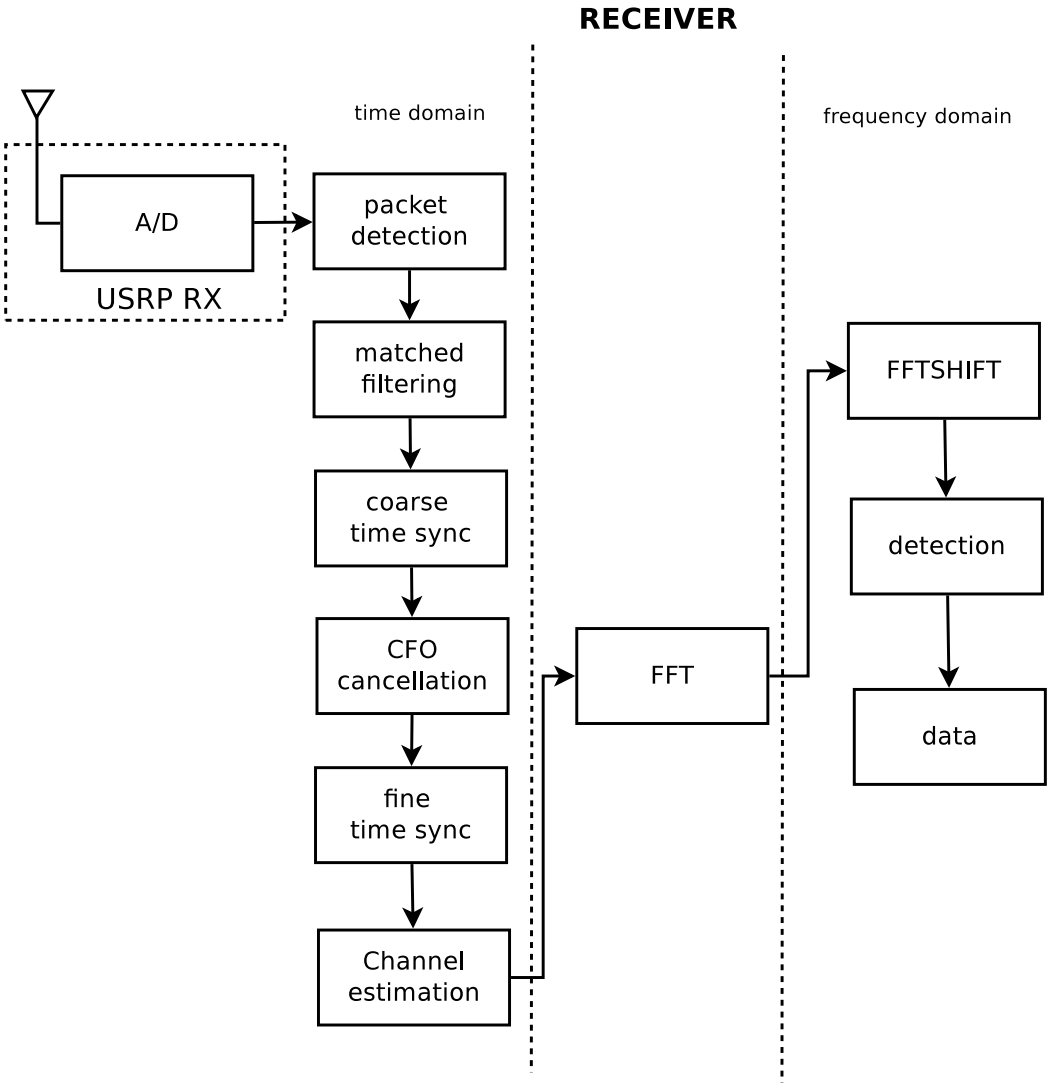


Figure 4.12: OFDM receiver structure.

Chapter 5

Implementation of distributed Alamouti STC with OFDM

Fast evolution of mobile communication faces users' growing demands for throughput, quality of service and coverage. Basic drawback of the mobile systems is their low throughput. The most promising data transmission technology, which is able to fulfill these requirements, is Multiple Input Multiple Output (MIMO). MIMO, in comparison to Single Input Single Output (SISO), allows us to increase radio channel capacity and quality of service, exploiting the spatial diversity and overcoming the effects of multipath fading.

5.1 Description of Alamouti STC

Alamouti Space-time Coding or Alamouti scheme is the first space-time block code scheme that provides full transmit diversity for systems with two transmit and M receive antennas. Alamouti STC is a rate-1 code. It takes two time-slots to transmit two symbols. The receiver decoding is provided by determining channel estimation matrix and maximum-likelihood detector. In this chapter, we will develop the Alamouti STC in an OFDM system with flat fading channel using first two transmitters and one receiver and then two transmitters and two receivers. Also, we will present some experimental results of the system performance such as signal to noise ratio (SNR) estimation and bit error rate.

Alamouti STC with 2 transmitters and 1 receiver 2x1 Scheme
As we mentioned before, we assume flat fading channel which means that we had one channel value for each transmitter-receiver pair. The Alamouti scheme transmits two complex symbols X_1 and X_2 over two symbol times slots. At time slot 1, the first transmitter sends X_1, and the second transmitter sends X_2. At time slot 2, the first transmitter sends $-X_2^*$ and the second transmitter sends X_1^*. We assume that the channel coefficients h_1, h_2 remain constant over the two symbol times. Then, the input-output relation of the system which uses Alamouti for two time slots is

$$\begin{aligned} Y_1 &= h_1 X_1 + h_2 X_2 + W_1 \\ Y_2 &= h_1(-X_2^*) + h_2 X_1^* + W_2 = h_2 X_1^* - h_1 X_2^* + W_2 \end{aligned} \tag{5.1}$$

which is shown in Figure 5.1.

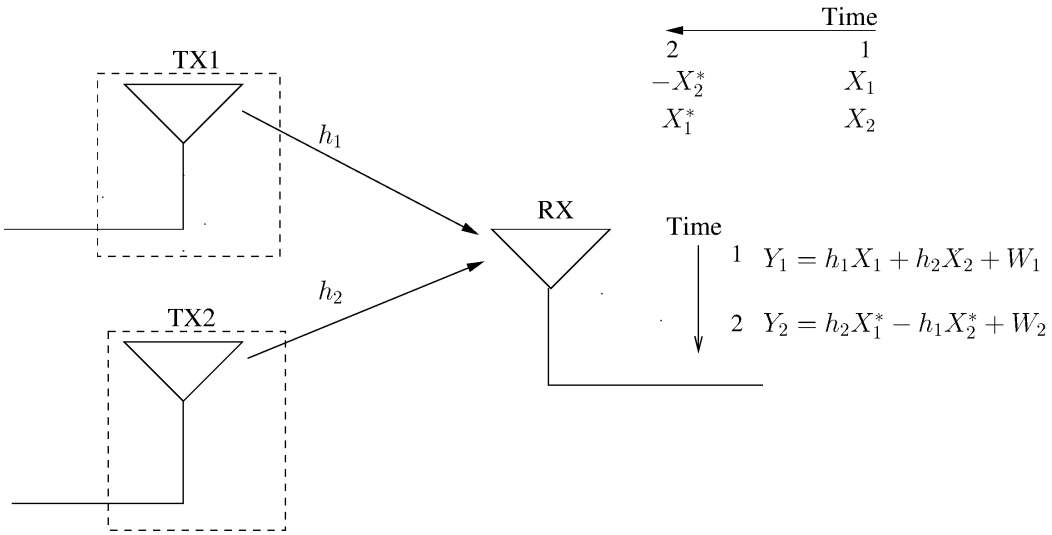


Figure 5.1: Alamouti STC with 2 transmitters and 1 receiver.

At the receiver, we construct the vector $\mathbf{Y}$ [2, 3],

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2^* \end{bmatrix} = \begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} W_1 \\ W_2^* \end{bmatrix}. \tag{5.2}$$

If we set $\mathbf{h} = [h_1 \; h_2]^T$, then the matrix

$$\mathcal{H} := \frac{1}{||\mathbf{h}||} \begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix} \tag{5.3}$$

is unitary, therefore the detection problem for X_1, X_2 decomposes into two separate scalar problems. Specifically, if we multiply, the vector $\mathbf{Y} = [Y_1 \; Y_2^*]^T$ with $\mathcal{H}^H$ we have

$$\begin{aligned} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} &:= \mathcal{H}^H \begin{bmatrix} Y_1 \\ Y_2^* \end{bmatrix} \\ &= \mathcal{H}^H \begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \mathcal{H}^H \begin{bmatrix} W_1 \\ W_2^* \end{bmatrix} \\ &= \begin{bmatrix} ||\mathbf{h}||X_1 \\ ||\mathbf{h}||X_2 \end{bmatrix} + \begin{bmatrix} W_1' \\ W_2' \end{bmatrix} \end{aligned} \tag{5.4}$$

where W_1' and W_2' are independent. Moreover, the input symbols X_1 and X_2 are independent, consequently we can decide for X_1 and X_2 using R_1 and R_2, respectively. Finally, if h_1 and h_2 are independent, then the Alamouti STC offers diversity 2.

Alamouti STC with 2 transmitters and 2 receivers 2x2 Scheme

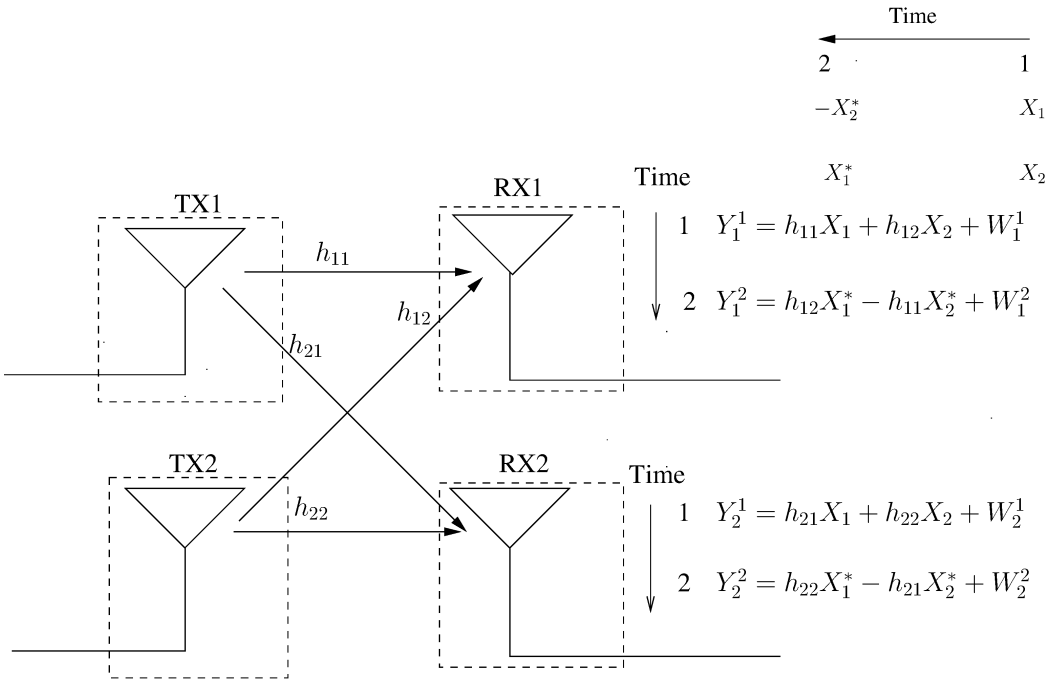


Figure 5.2: Alamouti STC with 2 transmitters and 2 receivers.

Similarly, for two transmitters and two receivers, the input-output relation of the system which uses Alamouti is

For time slot 1:

$$\begin{aligned} Y_1^1 &= h_{11}X_1 + h_{12}X_2 + W_1^1, \quad \textit{receiver 1} \\ Y_2^1 &= h_{21}X_1 + h_{22}X_2 + W_2^1, \quad \textit{receiver 2} \end{aligned} \tag{5.5}$$

and in vector form

$$\begin{bmatrix} Y_1^1 \\ Y_2^1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} W_1^1 \\ W_2^1 \end{bmatrix}. \tag{5.6}$$

For time slot 2:

$$\begin{aligned} Y_1^2 &= h_{12}X_1^* + h_{11}(-X_2^*) + W_1^2 = h_{12}X_1^* - h_{11}X_2^* + W_1^2, \quad \textit{receiver 1} \\ Y_2^2 &= h_{22}X_1^* + h_{21}(-X_2^*) + W_2^2 = h_{22}X_1^* - h_{21}X_2^* + W_2^2, \quad \textit{receiver 2} \end{aligned} \tag{5.7}$$

and in vector form

$$\begin{bmatrix} Y_1^2 \\ Y_2^2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} -X_2^* \\ X_1* \end{bmatrix} + \begin{bmatrix} W_1^2 \\ W_2^2 \end{bmatrix}. \tag{5.8}$$

Now we can combine (5.6) and (5.8) into a single matrix equation

$$\mathbf{Y} = \begin{bmatrix} Y_1^1 \\ Y_2^1 \\ Y_1^{2*} \\ Y_2^{2*} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} W_1^1 \\ W_2^1 \\ W_1^{2*} \\ W_2^{2*} \end{bmatrix}. \tag{5.9}$$

If we set $\mathbf{h} = [h_{11}\ h_{12}\ h_{21}\ h_{22}]^T$, then the

$$\mathcal{H} := \frac{1}{||\mathbf{h}||} \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix}, \tag{5.10}$$

is unitary.
Similarly, as the previous paragraph we multiply the vector $\mathbf{Y} = [Y_1^1\ Y_2^1\ Y_1^{2*}\ Y_2^{2*}]^T$ with $\mathcal{H}^H$ and we have

$$\begin{aligned} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} &:= \mathcal{H}^H \begin{bmatrix} Y_1^1 \\ Y_2^1 \\ Y_1^{2*} \\ Y_2^{2*} \end{bmatrix} \\ &= \mathcal{H}^H \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \mathcal{H}^H \begin{bmatrix} W_1^1 \\ W_2^1 \\ W_1^{2*} \\ W_2^{2*} \end{bmatrix} \\ &= \begin{bmatrix} ||\mathbf{h}||X_1 \\ ||\mathbf{h}||X_2 \end{bmatrix} + \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \end{aligned} \tag{5.11}$$

where Z_1 and Z_2 are independent. Moreover, the input symbols X_1 and X_2 are independent, consequently we can decide for X_1 and X_2 using R_1 and R_2, respectively. Finally, if h_{11}, h_{12}, h_{21} and h_{22} are independent, then the Alamouti STC offers diversity 4. In Figure 5.2, the Alamouti $2x2$ scheme is depicted.

Generally, an Alamouti system with two transmit antennas and M receive antennas has diversity $2M$ if the channels are independent.

5.2 Impementation of Alamouti 2x1 in USRP with C++

In this section, we describe the implementation of a 2x1 Alamouti OFDM system with USRPs N200. In order to achieve this, we had to synchronize the two transmitters to transmit simultaneously as we will describe later. In Figure 5.8, the layout of USRPs in Alamouti scheme (2 transmitters, 1 receiver) is shown.

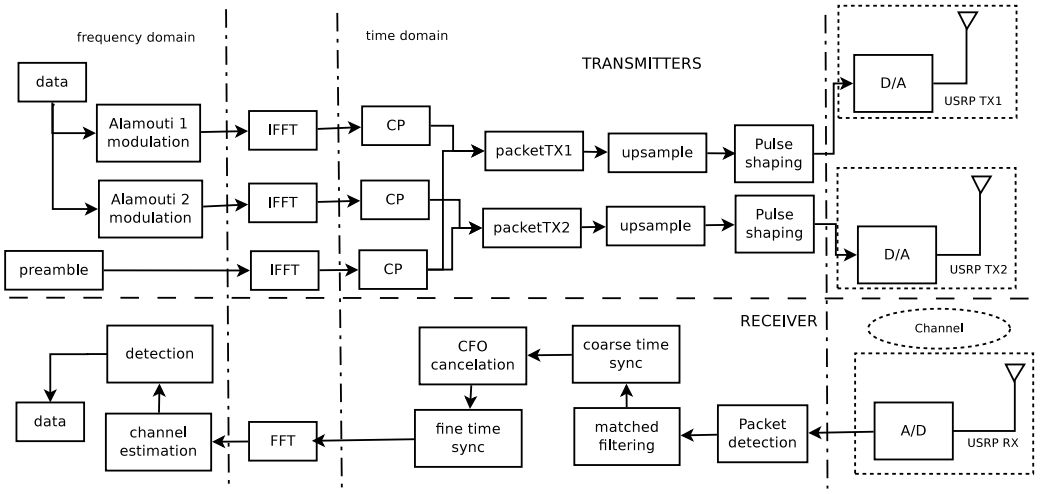


Figure 5.3: Structure of OFDM system with two transmitters and one receiver using Alamouti STC.

5.2.1 Transmitters

The two transmitters are exactly the same as described in the previous chapter in section 4.1. The only difference is that each transmitter sends different information symbols which are structured according to Alamouti STC. For this reason, we use the DC carrier (we don't send 0 as before) in order not to affect the Alamouti structure. As it is understood, the two transmitters have to be synchronized with accuracy much smaller than the symbol period. For the synchronization, we connect the USRPs with a MIMO cable in order to have the same clock time. We assume that the information data have length N. Then the two transmitters cooperate using Alamouti.
If the data consists of information symbols

$$\mathbf{X} = [X_1,\ X_2,\ X_3, \dots\ , X_N],$$

then each transmitter according to (5.1) sends the symbols described in Table 5.1 for any time slot.

In Figure 5.3, we can observe the overall process that a OFDM transmitter follows. The blocks "Alamouti 1 modulation" and "Alamouti 2 modulation" modify the information data as described in Table 5.1. Also, the preamble symbols sequence is not modified using Alamouti. In this way, the two transmitters send the same preamble symbols. In Figure 5.4, the structure of the OFDM packet for each transmitter using Alamouti STC is shown.
We should notice that the receiver has knowledge not only about the preamble symbols (in time domain-after IFFT) but also about the first $\frac{N}{4}$ transmitted data symbols (as training symbols in frequency domain-before IFFT) in order to estimate the flat fading channels for Alamouti.

	Transmitter 1	**Transmitter 2**
time t + T	X_1	X_2
time t + 2T	$-X_2^*$	X_1^*
time t + 3T	X_3	X_4
time t + 4T	$-X_4^*$	X_3^*
$\vdots$	$\vdots$	$\vdots$
time t + (N − 1)T	X_{N-1}	X_N
time t + NT	$-X_N^*$	X_{N-1}^*

Table 5.1: Alamouti transmitted data in N time slots

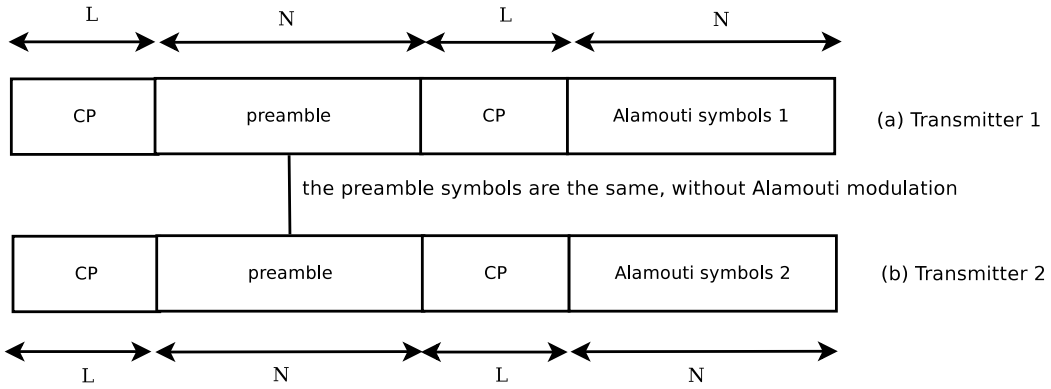


Figure 5.4: Structure of OFDM packet with Alamouti STC for (a) Transmitter 1, (b) Transmitter 2.

5.2.2 Receiver

In this subsection, we describe how the receiver works. The overall process is depicted in Figure 5.3. In general, the receiver with Alamouti STC is almost the same with the OFDM receiver without Alamouti that was described in section 4.2. Specifically, the packet detection is exactly the same and there are some small changes in time synchronization, channel estimation and detection.

Time synchronization and CFO cancellation

The coarse time synchronization is exactly the same as in the previous chapter. Similarly, using equation (4.10) we can estimate a coarse timing metric M_c whose peak indicates the coarse timing estimate $\hat{d}_c$ as

$$\hat{d}_c = \underset{d}{\text{argmax}}\{M_c(d)\}.$$

Now, we can separate the preamble symbols and estimate the CFO as in (4.19). Once $\hat{\Delta}f$ is calculated, it can be used to eliminate the effect of CFO in the received

signal by creating a sequence

$$y_k^{corr} = e^{-j2\pi\hat{\Delta}fk} \cdot y_k \tag{5.12}$$

where y_k is the signal affected by CFO and y_k^{corr} is the signal with canceled CFO [4].

After the coarse time synchronization and the correction of CFO, we should, as before, synchronize and find the optimal start of the packet. For this reason, we use the fine time synchronization, as described in the subsection 4.2.3, with some changes. Specifically, we calculate the statistic $M_{opt}(d)$ (Figure 5.5) that is shown in (4.22) and the optimal starting position is $\hat{d}_{opt}$ where

$$\hat{d}_{opt} = \underset{d}{\text{argmax}}\{M_{opt}\}.$$

As we can observe, we didn't use the energy window as before because our channel is flat fading and is enough to estimate as optimal start the position that the statistic $M_{opt}(d)$ is maximized. Then, we move back (although the preamble symbols are not used in channel estimation now) $\hat{d}_{opt} = \hat{d}_{opt} - L$.
Now, we can detect that the received symbol-spaced sequence of length $2N + 2L$ (preamble with CP, data with CP) is

$$\begin{aligned} R_0^{symbols} &= y_{\hat{d}_{opt}}^{corr}, \\ R_1^{symbols} &= y_{\hat{d}_{opt}+1}^{corr}, \\ &\vdots \\ R_{2(N+L)-1}^{symbols} &= y_{\hat{d}_{opt}+2(N+L)-1}^{corr}. \end{aligned} \tag{5.13}$$

Therefore, the information data symbols without CP in time domain are

$$R_z^{da} = \{R_k^{symbols}\}_{k=N+2L}^{k=2(N+L)-1}, \quad z = 1, \ldots, N \tag{5.14}$$

We should notice that if we had used the energy window as in the previous chapter we would have had synchronization problems sometimes. Moreover, because of the MIMO cable the synchronization of the two transmitters was not exact. As a result a small peak was appearing in the M_{opt} statistic (see Figure 5.6) which created problems while we were using the window. Therefore, we took advantage of the fact that the channel was flat fading and we estimated the optimal start position as described before.

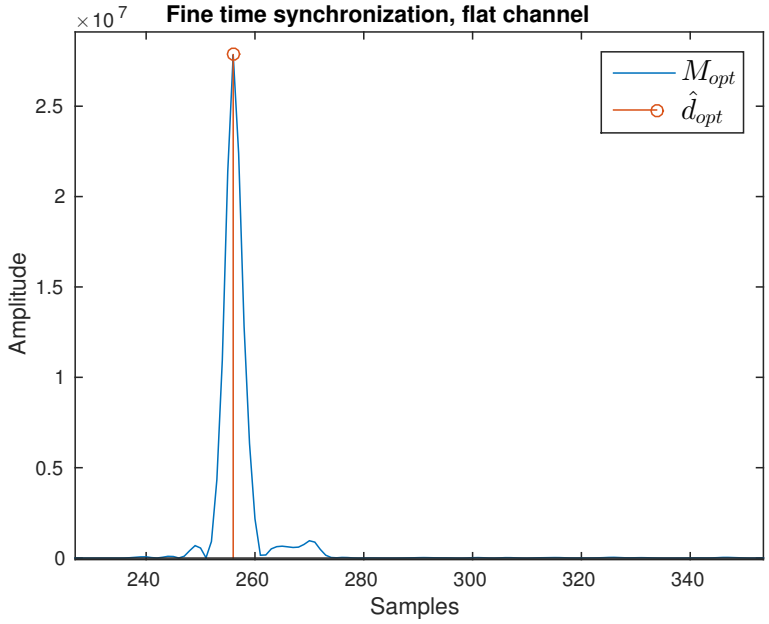


Figure 5.5: M_{opt} statistic for flat fading channel using Alamouti.

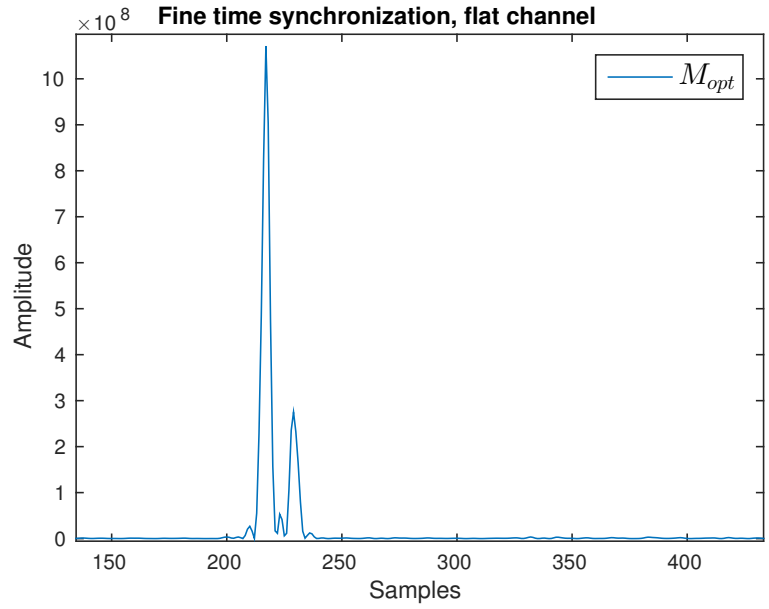


Figure 5.6: M_{opt} statistic for flat fading channel using Alamouti, with second peak because of bad TX synchronization.

Channel Estimation

The channel estimation is completely different from the subsection 4.2.4 because we use two transmitters that send simultaneously symbols that are modulated as Alamouti STC. Moreover, the channels were estimated after the FFT in the receiver and not before the FFT as in 1x1 scheme. Therefore, we applied FFT to $\mathbf{R}^{da}$ see equation (5.14) and we had $\tilde{\mathbf{R}}^{da}$ in frequency domain. As we mentioned, the receiver had complete knowledge of the first $\frac{N}{4}$ transmitted data (training) symbols (modified with Alamouti) in frequency domain (before IFFT). Let the training symbols be

$$\mathbf{TR} = \begin{bmatrix} X_1 & X_2 \\ -X_2^* & X_1^* \\ \vdots & \vdots \\ X_{\frac{N}{4}-1} & X_{\frac{N}{4}} \\ -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* \end{bmatrix}, \tag{5.15}$$

where the first column consists of the symbols sent by the first transmitter and the second column consists of the symbols sent by the second transmitter, respectively. (see Table 5.1).

Also, the received training sequence of length $\frac{N}{4}$ in frequency domain (after FFT) is the following

$$\tilde{\mathbf{R}}^{tr} = \{\tilde{R}_z^{da}\}_{z=1}^{k=\frac{N}{4}}.$$

Therefore, in our case for the training symbols the input-output relation for flat fading channels can be written as follows

$$\begin{bmatrix} \tilde{R}_1^{tr} \\ \tilde{R}_2^{tr} \\ \vdots \\ \tilde{R}_{\frac{N}{4}-1}^{tr} \\ \tilde{R}_{\frac{N}{4}}^{tr} \end{bmatrix} = \begin{bmatrix} X_1 & X_2 \\ -X_2^* & X_1^* \\ \vdots & \vdots \\ X_{\frac{N}{4}-1} & X_{\frac{N}{4}} \\ -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_{\frac{N}{4}-1} \\ W_{\frac{N}{4}} \end{bmatrix}, \tag{5.16}$$

which can be written in matrix-vector form

$$\tilde{\mathbf{R}}^{tr} = \mathbf{TRh} + \mathbf{W}.$$

Now, we can estimate the channels using Least Squares (LS)

$$\mathbf{h}_{LS} = \left(\mathbf{TR}^H\mathbf{TR}\right)^{-1}\mathbf{TR}^H\tilde{\mathbf{R}}^{tr}, \tag{5.17}$$

where $\cdot^H$ denotes Hermitian transpose. The $\mathbf{h}_{LS}$ is a vector of length 2 which contains the channels for the two transmitters h_1, h_2.

Detection

The detection should take place in frequency domain (after FFT). The received information data are in frequency domain so we can use them to decide as shown in equation (5.4)

$$\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} := \mathcal{H}^H \begin{bmatrix} \tilde{R}^{da}_{\frac{N}{4}+1} \\ \tilde{R}^{da^*}_{\frac{N}{4}+2} \end{bmatrix},$$

for the $N - \frac{N}{4} = \frac{3N}{4}$ time slots (we don't decide for the trainings). The decision for the first two symbols is taken in the first two time slots and similarly for the next two symbols, in the next two time slots, etc. We decide using the sign for the real and the imaginary part because we used 4-QAM and the symbols have the same probability.

The constellation of the received information symbols before the decision is depicted in the Figure 5.7.

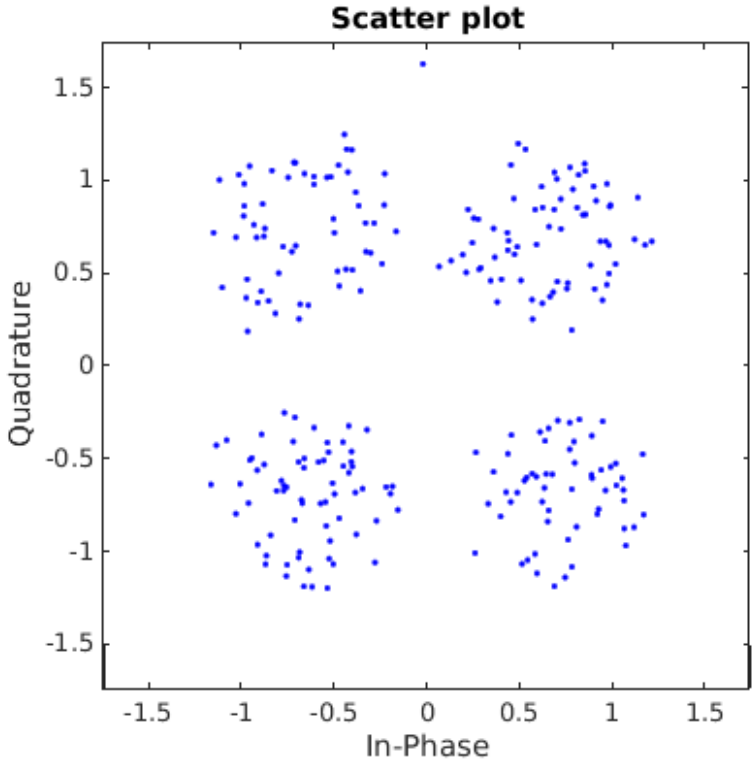


Figure 5.7: Constellation of received information symbols before the decision with the two transmitters.

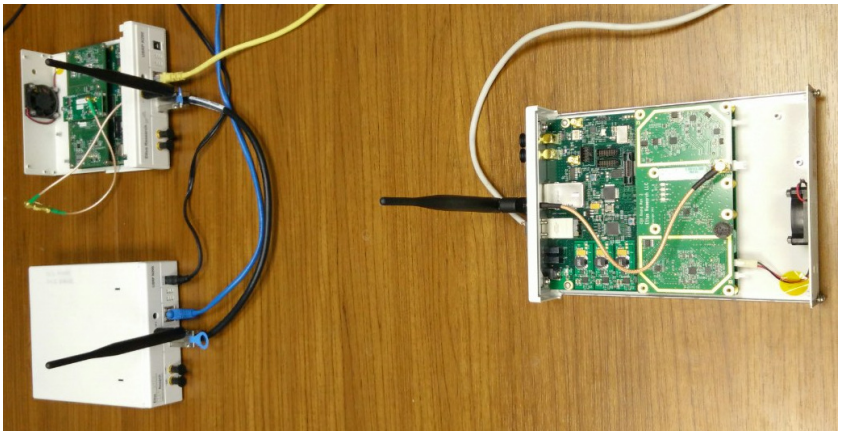


Figure 5.8: USRPs N200, two transmitters connected with MIMO cable for synchronization and one receiver.

5.3 Simulation of Alamouti 2x2 in Matlab

In this section, we describe the simulation in matlab a 2x2 Alamouti OFDM system without USRPs. We did these in order to understand how Alamouti 2x2 scheme works in a OFDM system. In Figure 5.9, the processing blocks implemented by transmitters and receivers are depicted. We will describe these processes in detail in the following subsections.

5.3.1 Transmitters

The transmitters are almost the same as described in subsection 5.2.1. The only difference is that we create-apply the CFOs and the channels.

5.3.2 Receivers

Now we have two receivers. Each one does exactly the same processes (packet detection, time synchronization, CFO cancellation) as mentioned in the subsection 5.2.2. The only difference is the channel estimation and the packet detection in which the two receivers have to combine their received data in order to make the decision.

Channel estimation

We have to estimate four channels which are h_{11}, h_{12}, h_{21}, h_{22}. As before (see subsection 5.2.2) the receivers have knowledge for the first $\frac{N}{4}$ symbols as trainings at frequency domain (before IFFT). Also, lets assume for simplicity that $\tilde{\mathbf{Y}}_1$ and $\tilde{\mathbf{Y}}_2$ are the received data of receiver 1 and receiver 2 respectively after FFT. Therefore, in our case, for the training symbols the input-output relation for the two receivers

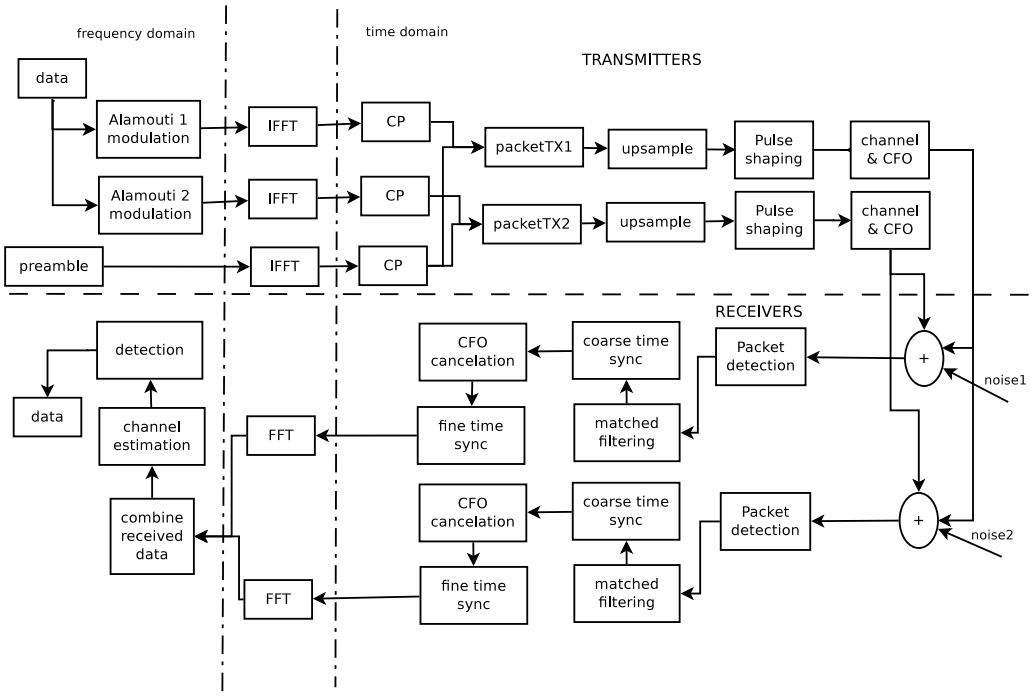


Figure 5.9: Structure of OFDM system with two transmitters and two receivers using Alamouti STC.

and for flat fading channels can be written as follows (the exponent shows the time slot and the indicator shows the receiver)

$$\begin{bmatrix} \tilde{Y}_1^1 \\ \tilde{Y}_2^1 \\ \tilde{Y}_1^2 \\ \tilde{Y}_2^2 \\ \vdots \\ \tilde{Y}_1^{\frac{N}{4}-1} \\ \tilde{Y}_2^{\frac{N}{4}-1} \\ \tilde{Y}_1^{\frac{N}{4}} \\ \tilde{Y}_2^{\frac{N}{4}} \end{bmatrix} = \begin{bmatrix} X_1 & X_2 & 0 & 0 \\ 0 & 0 & X_1 & X_2 \\ -X_2^* & X_1^* & 0 & 0 \\ 0 & 0 & -X_2^* & X_1^* \\ \vdots & \vdots & & \\ X_{\frac{N}{4}-1} & X_{\frac{N}{4}} & 0 & 0 \\ 0 & 0 & X_{\frac{N}{4}-1} & X_{\frac{N}{4}} \\ -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* & 0 & 0 \\ 0 & 0 & -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{21} \\ h_{22} \end{bmatrix} + \begin{bmatrix} W_1^1 \\ W_2^1 \\ W_1^2 \\ W_2^2 \\ \vdots \\ W_1^{\frac{N}{4}-1} \\ W_2^{\frac{N}{4}-1} \\ W_1^{\frac{N}{4}} \\ W_2^{\frac{N}{4}} \end{bmatrix}, \tag{5.18}$$

which can be written in vector form

$$\tilde{\mathbf{Y}} = \mathbf{TRh} + \mathbf{W},$$

where

$$\mathbf{TR} = \begin{bmatrix} X_1 & X_2 & 0 & 0 \\ 0 & 0 & X_1 & X_2 \\ -X_2^* & X_1^* & 0 & 0 \\ 0 & 0 & -X_2^* & X_1^* \\ \vdots & \vdots & & \\ X_{\frac{N}{4}-1} & X_{\frac{N}{4}} & 0 & 0 \\ 0 & 0 & X_{\frac{N}{4}-1} & X_{\frac{N}{4}} \\ -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* & 0 & 0 \\ 0 & 0 & -X_{\frac{N}{4}}^* & X_{\frac{N}{4}-1}^* \end{bmatrix}, \tag{5.19}$$

As we can observe the first column corresponds to the first transmitter and to the first receiver, the second column corresponds to the second transmitter and to the first receiver, the third column corresponds to the first transmitter and to the second receiver and the fourth column corresponds to the second transmitter and to the second receiver. Now, we can estimate the channels using Least Squares (LS)

$$\mathbf{h}_{LS} = \left(\mathbf{TR}^H \mathbf{TR}\right)^{-1} \mathbf{TR}^H \tilde{\mathbf{Y}}, \tag{5.20}$$

where $\cdot^H$ denotes Hermitian transpose. The $\mathbf{h}_{LS}$ is a vector of length 4 which contains the channels for the two transmitters and two receivers h_{11}, h_{12}, h_{21}, h_{22}.

Detection

The detection should take place in frequency domain (after FFT). The received information data are in frequency domain so we can use them to decide as shown in equation (5.11)

$$\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} := \mathcal{H}^H \begin{bmatrix} \tilde{Y}_1^{\frac{N}{4}+1} \\ \tilde{Y}_2^{\frac{N}{4}+1} \\ \tilde{Y}_1^{*\frac{N}{4}+2} \\ \tilde{Y}_2^{*\frac{N}{4}+2} \end{bmatrix},$$

for the $N - \frac{N}{4} = \frac{3N}{4}$ time slots (we don't decide for the trainings). The decision for the first two symbols is taken in the first two time slots and similarly for the next two symbols, in the next two time slots etc. We decide using the sign for the real and the imaginary part because we used 4-QAM and the symbols have the same probability. The constellation of the received information symbols before the decision is depicted in the Figure 5.10.

5.4 SNR estimation

In this section, we will try to estimate the SNR in the receiver for the two implemented systems. The first system was the OFDM system with one transmitter

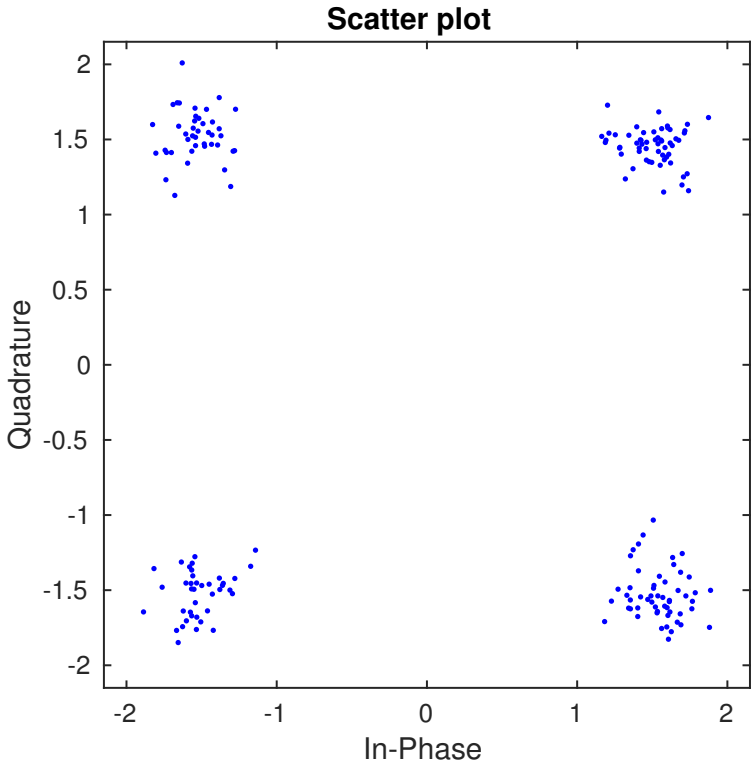


Figure 5.10: Constellation of received information symbols before the decision with two transmitters and two receivers.

and one receiver without Alamouti STC (1x1 scheme). The second system is the Alamouti with two transmitters and one receiver (2x1 scheme). Both, systems were implemented using USRP as we described previously. In order to estimate the SNR in the receiver we use the following formula

$$SNR_{dB} = 10 \cdot log_{10} \frac{\text{Power of useful signal}}{\text{Noise power}} \tag{5.21}$$

5.4.1 SNR in 1x1 scheme

As we mentioned in subsection 4.2.4, once we calculate the vector h_{LS} of length L using least squares (for flat fading or frequency selective channel) we apply FFT to N points and we create the new vector H_{LS} in frequency domain with length

N. So we can calculate the SNR according to equation(5.21)

$$
\begin{aligned}
SNR_{dB} &= 10 \cdot log_{10} \frac{E\{|H_{LS}(k)x(k)|^2\}}{E\{|n(k)|^2\}} \\
&= 10 \cdot log_{10} \frac{E\{|H_{LS}(k)x(k)|^2\}}{E\{|Y(k) - H_{LS}(k)x(k)|^2\}} \\
&= 10 \cdot log_{10} \frac{\sum_{k=1}^{N} p_{x_k} |H_{LS}(k)x(k)|^2}{\sum_{k=1}^{N} p_{x_k} |Y(k) - H_{LS}(k)x(k)|^2} \quad \text{symbols with same probability} \\
&= 10 \cdot log_{10} \frac{\sum_{k=1}^{N} |H_{LS}(k)x(k)|^2}{\sum_{k=1}^{N} |Y(k) - H_{LS}(k)x(k)|^2}
\end{aligned}
\tag{5.22}
$$

where $x(k)$ is the $k-th$ transmitted symbol and $Y(k)$ is the output affected by noise in receiver for the $k-th$ transmitted symbol. Specifically, we assume that the receiver knows the transmitted symbols $x(k)$ and when the channels are calculated we can estimate the noise power as

$$E\{|n(k)|^2\} = E\{|Y(k) - H_{LS}(k)x(k)|^2\}$$

Now equation (5.22) can be written

$$
\begin{aligned}
&SNR_{dB} = \\
&= 10 \cdot log_{10} \frac{\sum_{k=1}^{N} |H_{LS}(k)x(k)|^2}{\sum_{k=1}^{N} |Y(k) - H_{LS}(k)x(k)|^2} \\
&= 10 \cdot log_{10} \frac{|H_{LS}(1)x(1)|^2 + \ldots + |H_{LS}(N)x(N)|^2}{|Y(1) - H_{LS}(1)x(1)|^2 + \ldots + |Y(N) - H_{LS}(N)x(N)|^2}
\end{aligned}
\tag{5.23}
$$

5.4.2 SNR in Alamouti 2x1 scheme

Once we have calculated the channels h_1 and h_2 as described in subsection 5.2.2, we can calculate the SNR in the receiver similarly as we did before for 1x1 scheme. Now, we have flat fading channel for the two transmitters. Also, h_1 and h_2 were estimated, if the receiver knows the transmitted symbols $x(k)$ can estimate the

SNR for N time slots as

$$SNR = \frac{\sum_{k=k+2}^{N} |F_k|^2 + |G_k|^2}{\sum_{k=k+2}^{N} |Y(k) - F_k|^2 + |Y(k+1) - G_k|^2}$$

for simplicity $\mathrm{Y(k)} = Y_k$

$$= \frac{|F_1|^2 + |G_1|^2 + \ldots + |F_{N-1}|^2 + |G_{N-1}|^2}{|Y_1 - F_1|^2 + |Y_2 - G_1|^2 + \ldots + |Y_{N-1} - F_{N-1}|^2 + |Y_N - G_{N-1}|^2} \tag{5.24}$$

where

$$F_k = h_1 x(k) + h_2 x(k+1)$$
$$G_k = -h_1 x^*(k+1) + h_2 x^*(k)$$

and $Y(k) = Y_k$ is the output in time slot k. As we can observe in the sum we move with step 2 in order to be simpler to write the input for the Alamouti symbols. In dB

$$SNR_{dB} = 10 \cdot log_{10} SNR$$

5.4.3 Results

We measured the SNR in 4 different distances for Alamouti 2x1 scheme, for Alamouti 2x1 scheme with half power and for 1x1 scheme without Alamouti. The results are shown in table 5.2 As we can easily understand, in the Alamouti 2x1 scheme

Distance	**Alamouti scheme 2x1**	**Alamouti scheme 2x1 with half power**	**1x1 scheme without Alamouti**
1.5 m	17.2 dB	14.6 dB	14.19 dB
3.0 m	16.94 dB	14.21 dB	14.15 dB
4.5 m	16.8 dB	13.13 dB	14.12 dB
6.0 m	13.4 dB	11.18 dB	10.601 dB

Table 5.2: SNR for different distances between transmitters and receiver.

with two transmitters the transmitted energy is twice as the energy in 1x1 scheme. For this reason we transmit with the half power in Alamouti 2x1 scheme in order to compare the results with 1x1 scheme easier. For the transmission with the half transmitted power we multiply the Alamouti symbols in 2x1 scheme with $\frac{\sqrt{2}}{2}$. Moreover, comparing the results, we can observe that the SNR in the Alamouti 2x1 scheme with the half power is close to the SNR of 1x1 scheme. Also, the Alamouti

2x1 scheme has approximately $2.5 - 3.5$ dB difference with the other two schemes because of the double transmitted power (2 transmitters). Finally, while the distance increases, the SNR value decreases.

Theoretically, the Alamouti 2x1 scheme and the Alamouti 2x1 scheme with half power should have a deviation 3 dB factor. In Alamouti 2x1 scheme we have

$$SNR = \frac{T}{N_0}$$

where T is the transmitted power (from two transmitters) and N_0 is the noise power. We assume that the noise power is almost the same in both systems. Then for the Alamouti 2x1 scheme with the half transmitted power ($\frac{T}{2}$), we have

$$SNR^{half} = \frac{\frac{T}{2}}{N_0}$$

Comparing these two, we take

$$\frac{SNR}{SNR^{half}} = \frac{\frac{T}{N_0}}{\frac{\frac{T}{2}}{N_0}} = \frac{TN_0}{\frac{T}{2}N_0} = \frac{2TN_0}{TN_0} = 2$$

so

$$SNR = 2SNR^{half}$$
in dB we have
$$10 \cdot log_{10}SNR = 10 \cdot log_{10}2SNR^{half}$$
$$10 \cdot log_{10}SNR = 10 \cdot log_{10}(2)10 \cdot log_{10}SNR^{half}$$
$$SNR_{dB} \approx 3 + SNR_{dB}^{half}$$

5.5 Bit Error Rate (BER) using matlab

In order to examine and study the behavior of our system, we simulate and calculate the bit error rate (BER) in matlab for Alamouti 2x1 and Alamouti 2x2. We use SNR values from 0 to 20 with step 2 for Alamouti 2x1 scheme and SNR values -10 to 4 with step 2 Alamouti 2x2 scheme. For each SNR value we performed $15 \cdot 10^3$ loops while in every loop we send a different OFDM packet of length N (e.g. $N = 256$). We should mention that the simulation takes only place in matlab, without using USRP, for two reasons. The first reason is that the implementation is simpler and the second and most important one is that we can have full control of the system. Specifically, we created the channels, the noise in the receiver and we applied zero CFO for simplicity. Firstly, in Figure 5.11, we present the BER while the receiver has knowledge of the channels or estimates them, for fixed time synchronization or using the synchronization algorithm [4]. Comparing the results,

[4]Synchronization algorithm is the algorithm that we used for time synchronization.

we see that when the channels are known, the BER is always a slightly better. Also, when the synchronization is fixed (we know the start), the BER is a little improved compare to the BER of synchronization algorithm. However, the BERs do not present significant differences. In Figure 5.12, we compare the BERs of the two transmitter when they are fully and partially synchronized[5]. We assume that the channels, as well as the optimal starting point (fixed time synchronization), are known at the receiver. Comparing the results, we observe that the BER, when the transmitters are fully synchronized, is much better. In this way, we understand/realize the importance of the synchronization between the transmitters[6]. For both of the figures above, from SNRs higher or equal of 20 dB, the BER is 0 or almost 0. Similarly, in Figure 5.13, we present the same BERs for the Alamouti 2x2 scheme with the only difference that now we estimate the channels and the optimal start. The comparison of the results is the same. Finally, the BER in this scheme is 0 above 4 dB.

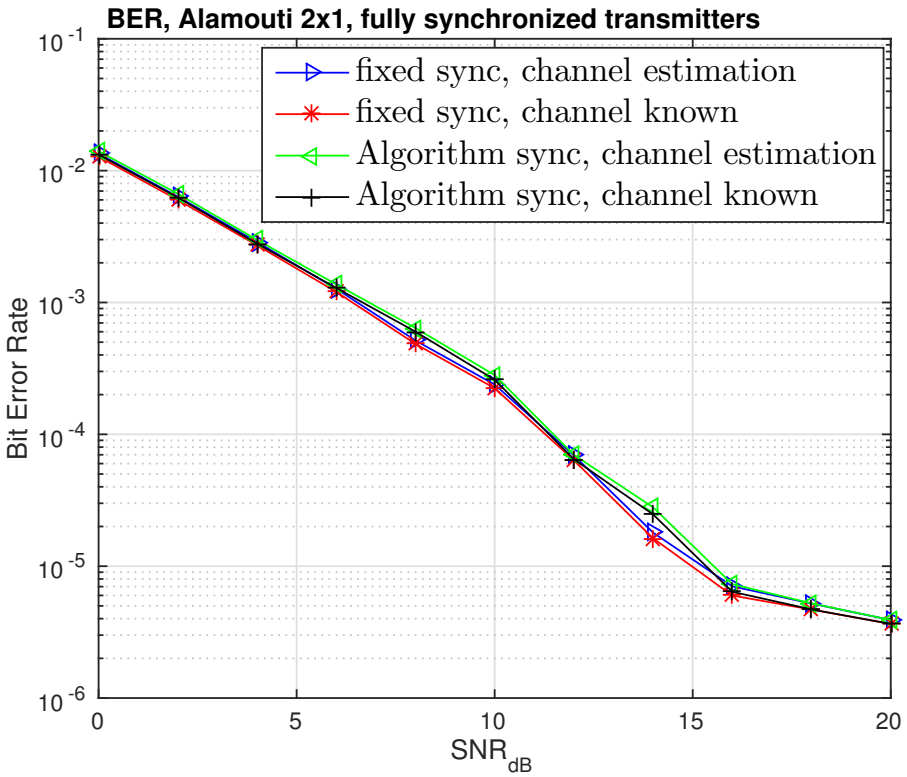


Figure 5.11: BER for Alamouti 2x1 scheme, with channels known or estimated, with fixed synchronization or using the synchronization algorithm.

[5]We mean the synchronization of the transmitters in order to send simultaneously.
[6]For optimal synchronization we can use GPS.

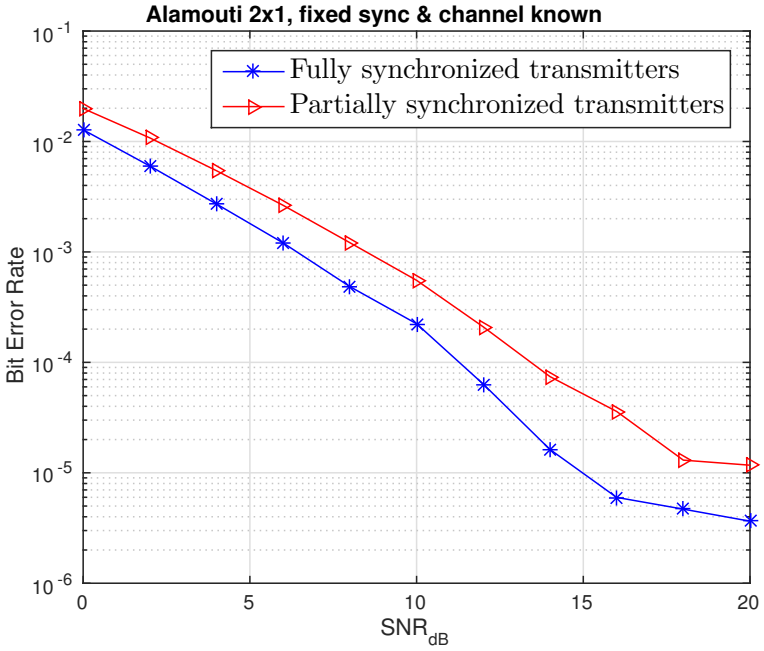


Figure 5.12: BER for Alamouti 2x1 with fully and partly synchronized transmitters.

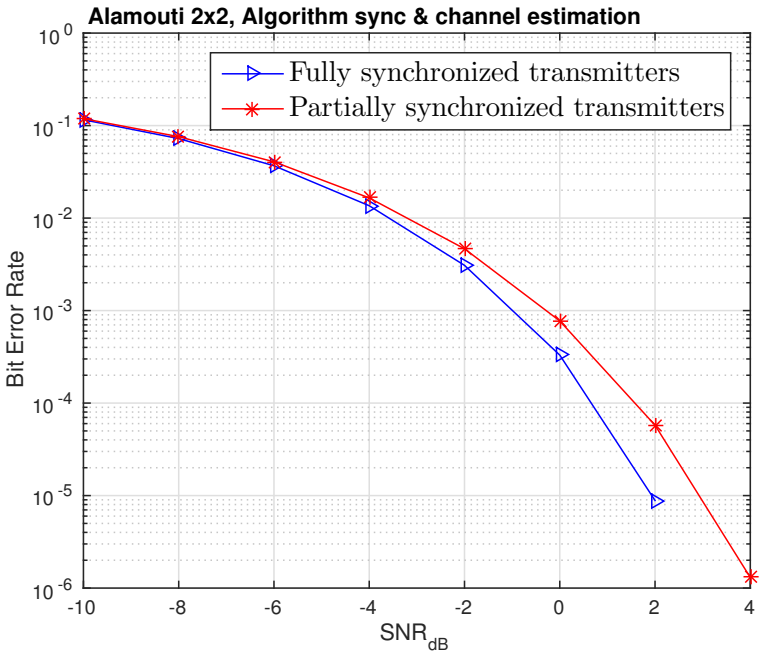


Figure 5.13: BER for Alamouti 2x2 with fully and partly synchronized transmitters.

Chapter 6

Conclusions

6.1 Conclusion

Nowadays, wireless communication is in every aspect of our life. Thus, it is important to study them. In this thesis, we studied and implemented an OFDM system using USRP N200. Moreover, we tried to append the Alamouti STC in a OFDM system in order to increase the reliability of the system. Also, we described techniques for channel estimation, CFO cancellation and time synchronization using only the preamble symbols. Finally, the results show that USRP N200 enable real-time implementation of MIMO link such as Alamouti with relatively high throughput. Such a platform-system is a low cost solution which is fully flexible and easy to modify, because of the fully software implementation.

6.2 Future Work

Future work could target to extend the capabilities of the developed system. Specifically, it will be useful to extend the Alamouti 2x1 scheme implemented with two USRPs as transmitters and one as receiver to the Alamouti 2x2 scheme using two USRPs as transmitters and two USRPs as receivers. Also, it is important the channel estimation, in Alamouti scheme, to take place using the preamble symbols and not new trainings symbols. In this way, we will gain more bandwidth for the information symbols. As a result, using only the preamble symbols of length N, the CFO cancellation, the time synchronization and the channel estimation will be succeed, as in the OFDM system without Alamouti. Moreover, what could be essential is a further study and research about CFO when we have more than one transmitter. Finally, it could be challenging a implementation of Alamouti in OFDM using frequency selective channels while the synchronization of the transmitters could be achieved with GPS for extra accuracy.

Bibliography

[1] G. Stuber and Y. Li, *Orthogonal frequency division multiplexing for wireless communications*, Springer, 2006

[2] D. Tse, and P. Viswanath, *Fundamentals of wireless communications*, Cambridge uni- versity press, August 13, 2004

[3] A.P. Liavas, *Lectures of wireless communications*, Technical University of Crete, 2014

[4] A.P. Liavas, and I. Kardaras, *Software-Defined Radio Implementation of an OFDM Link*, Technical University of Crete, 2010

[5] A. Awoseyila, C. Kasparis and B. Evans, *Improved preamble-aided timing estimation for OFDM systems*, IEEE Communications Letters, Nov. 2008

[6] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao *Timing recovery for OFDM transmission*, IEEE Journal Selected Areas Communications, vol. 18, pp. 2278-2290, Nov. 2000

[7] T. Magounaki, *A Software-Defined implementation of an OFDM-Adaptive OFDMA system using USRPs 1*, Bachelor Thesis, Oct. 2014

[8] USRP-Ettus Research webpage, *A National Instruments Company*

[9] GNU Radio webpage, *The free and open software radio ecosystem*