TECHNICAL UNIVERSITY OF CRETE

GRADUATION THESIS

# Study and SDR implementation of OFDM modulation

*Author:*
Orestis ZEKAI

*Supervisor:*
Prof. Athanasios P. LIAVAS

**Thesis committee**
Prof. Athanasios P. Liavas (ECE)
Associate Professor George N. Karystinos (ECE)
Professor Michael Paterakis (ECE)

*A thesis submitted in fulfillment of the requirements for the degree of BSc of Engineering*

*in the*

Department Of
Electrical and Computer Engineering

October 2017

Technical University of Crete

# *Abstract*

Department Of
Electrical and Computer Engineering

BSc of Engineering

**Study and SDR implementation of OFDM modulation**

by Orestis ZEKAI

Digital bandpass modulation techniques can be broadly classified into two categories. The first is single-carrier modulation, where data is transmitted by using a single radio frequency (RF) carrier. The other is multicarrier modulation, where data is transmitted by simultaneously modulating multiple RF carriers. This thesis is concerned with a particular type of multi-carrier modulation known as orthogonal frequency division multiplexing (OFDM). OFDM has gained popularity in a number of applications including digital subscriber loops, wireless local area networks,etc. It is also the chosen technique for fourth generation (4G) cellular land mobile radio systems.OFDM transmits data in parallel by modulating a set of orthogonal subcarriers. It is attractive because it admits relatively easy solutions to some difficult challenges that are encountered when using single-carrier modulation schemes on wireless channels. The greatest benefit of using OFDM is that the modulation of closely-spaced orthogonal sub-carriers partitions the available bandwidth into a collection of narrow sub-bands. On the other hand, although OFDM waveforms are resilient to timing errors, they are highly sensitive to frequency offsets and phase noise at the transmitter and receiver RF and sampling clock oscillators.

In this thesis, we will deal with synchronization issues in OFDM systems. Also, we will study frequency offset estimation and cancellation and channel estimation. Finally, we will implement an OFDM system using USRP N200 kits.

# *Acknowledgements*

First of all, I would like to thank my supervising professor, Athanasios P. Liavas, for being patient and supportive and for sharing his knowledge and experience on telecommunication topics. Also, I would like to thank my family for supporting me and believing in me during my studies. Finally, I would like to thank my friends and colleagues for their help and for the enjoyable time I have experienced during my study at the Technical University of Crete.

# Contents

# List of Figures

# List of Abbreviations

| | |
|---|---|
| **USRP** | **U**niversal **S**oftware **R**adio **P**eripheral |
| **OFDM** | **O**rthogonal **F**requency **D**ivision **M**ultiplexing |
| **RF** | **R**adio **F**requency |
| **ISI** | **I**nter **S**ymbol **I**nterferecne |
| **MCM** | **M**ulti **C**arrier **M**odulation |
| **FPGA** | **F**ield **P**rogrammable **G**ate **A**rray |
| **ADC** | **A**nalog to **D**igital **C**onverter |
| **DAC** | **D**igital to **A**nalog **C**onverter |
| **UHD** | **U**niversal **H**ardware **D**river |
| **SDR** | **S**oftware **D**efined **R**adio |
| **CFO** | **C**arrier **F**requency **O**ffset |
| **SRRC** | **S**quare **R**oot **R**aised **C**osine |
| **QAM** | **Q**uadrature **A**mplitude **M**odulation |
| **DFT** | **D**iscrete **F**ourier **T**ransform |
| **IDFT** | **I**nverse **D**iscrete **F**ourier **T**ransform |
| **LS** | **L**east **S**quares |
| **DSW** | **D**ouble **S**liding **W**indow |
| **SNR** | **S**ignal to **N**oise **R**atio |

*Dedicated to my family…*

# Chapter 1

# Introduction

## 1.1 Thesis Outline

This thesis is organized as follows:

- In Chapter 1, we present a brief introduction to wireless communications, bandpass modulation, orthogonal frequency-division multiplexing and software defined radios.
- In Chapter 2, we describe the software tools that we used as well as the model and specifications of USRPs.
- In Chapter 3, we introduce and analyse theoretically an OFDM system. We consider both the analog and the digital model.
- In Chapter 4, we analyze the techniques used for coarse time synchronization, carrier frequency offset estimation and cancellation, fine time synchronization, channel estimation and demodulation.
- In Chapter 5, we provide a conclusion, followed by some suggestions and directions for future work.

## 1.2 Wireless Communications

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, wireless communication is a field that has been around for about a hundred years. Specifically, the 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 of the subscribers.

In the present days, wireless is one of the most important technology of the 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, infra red communication, broadcast radios, computers, Wi-Fi, 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

There are two main digital bandpass modulation techniques:

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

### 1.3.1 Single-carrier modulation

The single carrier transmission means one radio frequency carrier is used to carry the information. 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.

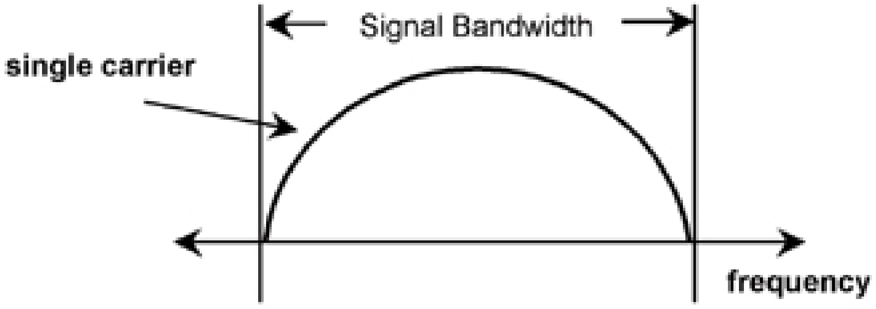


FIGURE 1.1: The single-carrier modulation

### 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.

The advantages of MCM include relative immunity to fading caused by transmission over more than one path at a time (multipath fading). In a single carrier system, a single fade can cause the entire system to fail, but in MCM systems only a small percentage of the sub carriers will be affected. Also, MCM systems have less susceptibility to interference caused by impulse noise, and are 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.

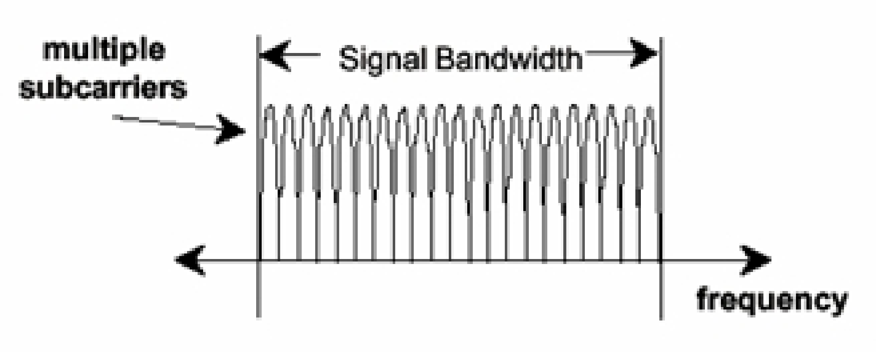


FIGURE 1.2: The multi-carrier modulation

## 1.4 Orthogonal Frequency Division Multiplexing (OFDM)

Orthogonal frequency-division multiplexing (OFDM) is a method of encoding digital data on multiple carrier frequencies. OFDM has become a popular scheme for wideband digital communication, used in applications such as digital television and audio broadcasting, DSL Internet access, wireless networks, powerline networks, and 4G mobile communications.

OFDM is a frequency-division multiplexing (FDM) scheme used as a digital multi-carrier modulation method. A large number of closely spaced orthogonal sub-carrier signals are used to carry data on several parallel data streams or channels. Each sub-carrier is modulated with a conventional modulation scheme at a low symbol rate, maintaining total data rates similar to conventional single-carrier modulation schemes in the same bandwidth.

The primary advantage of OFDM over single-carrier schemes is its ability to cope with severe channel conditions without complex equalization filters. Channel equalization is simplified because OFDM may be viewed as using many slowly modulated narrowband signals rather than one rapidly modulated wideband signal. The low symbol rate makes the use of a guard interval between symbols affordable, making it possible to eliminate ISI and utilize echoes and time-spreading to achieve a diversity gain.

## 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.

Software-defined radio (SDR) is a radio communication system where components that have been typically implemented in hardware (e.g. mixers, filters, amplifiers, modulators/demodulators, detectors, etc.) are instead implemented by means of software on a personal computer or embedded system. While the concept of SDR is not new, the rapidly evolving capabilities of digital electronics render practical many processes which used to be only theoretically possible.

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 hardware 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. The basic block diagram of a SDR transceiver is depicted in Figure 1.3.

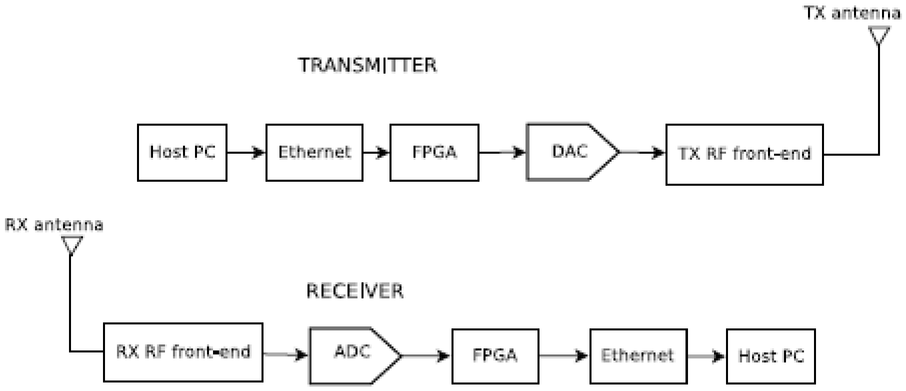


FIGURE 1.3: High lever block diagram of SDR transceiver

Data taken from [1],[3].

# 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 downconversion, 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 a 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. Similar components consist a USRP N200, that will be used in this thesis.

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. In this thesis, we will use GNU Radio, Matlab.

### 2.1.1 USRP N200 motherboard overview

The Ettus Research 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, dynamic 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.

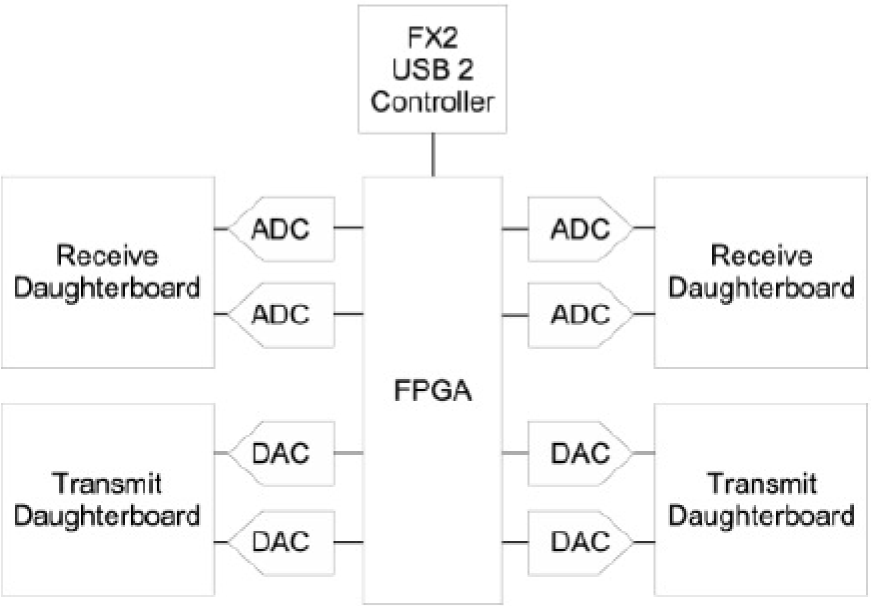


FIGURE 2.1: Block Diagram of a USRP 1 board

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 different 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 2.2, a photo of USRP N200 is shown.

FIGURE 2.2: Photo of 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.

- 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
  - 50 MHz of RF bandwidth with 8 bit samples
  - 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© WebPACKTM tools

**USRP N200 specifications** In this subsection, the specification of USRP N200 series is described as shown in the table below. Also, in Figure 2.3, the internal construction and the individual blocks are depicted.

| 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 |

### 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 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. The main daughterboard families are CBX and WBX. Both CBX and WBX family of daughterboards are complete RF transceiver systems.

#### 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.

#### 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.

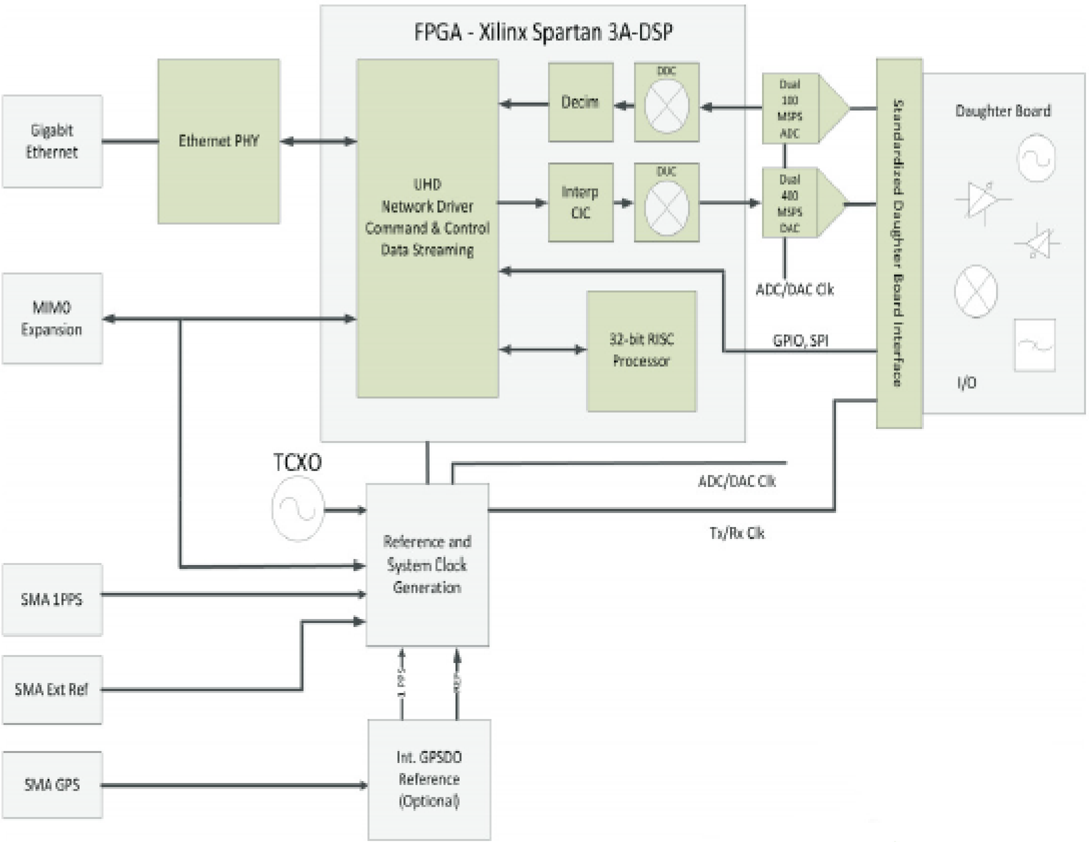


FIGURE 2.3: Internal construction of USRP N200

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.

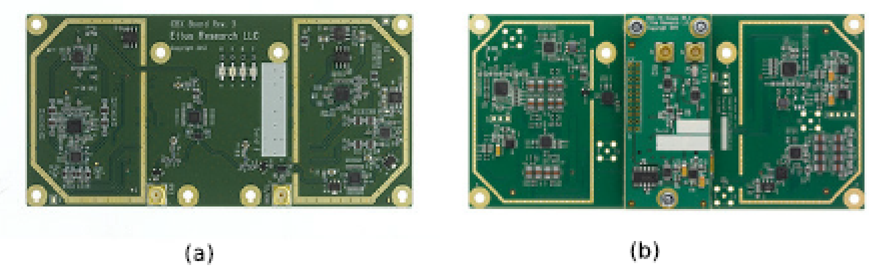


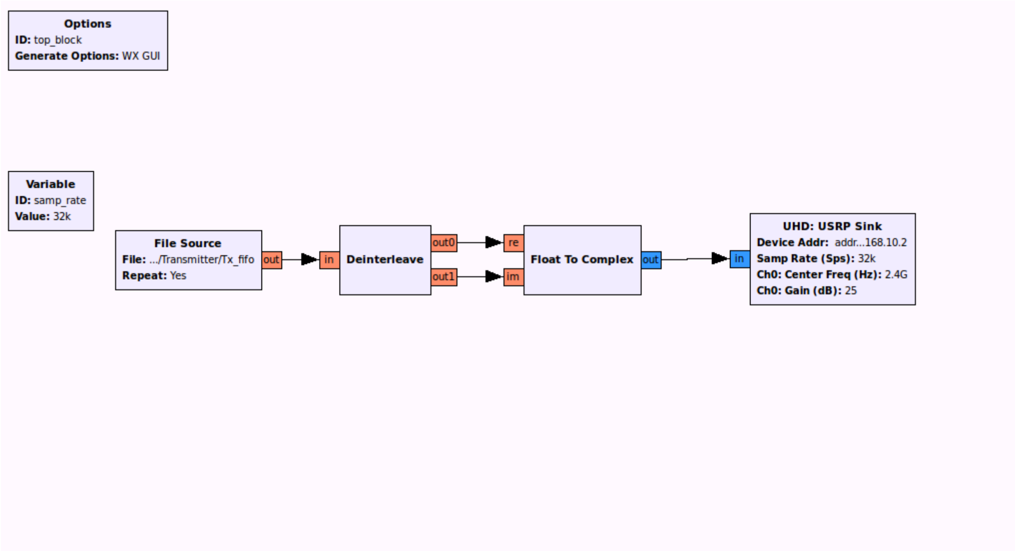
FIGURE 2.4: Daughterboard: (a) CBX (b) WBX

Figures and information was taken from *Ettus Research Website* [4].

## 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).

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. In this thesis, we 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 2.5 and a receiver is depicted in Figure 2.6. The transmitter reads 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.



Options
ID: top_block
Generate Options: WX GUI

Variable
ID: samp_rate
Value: 32k

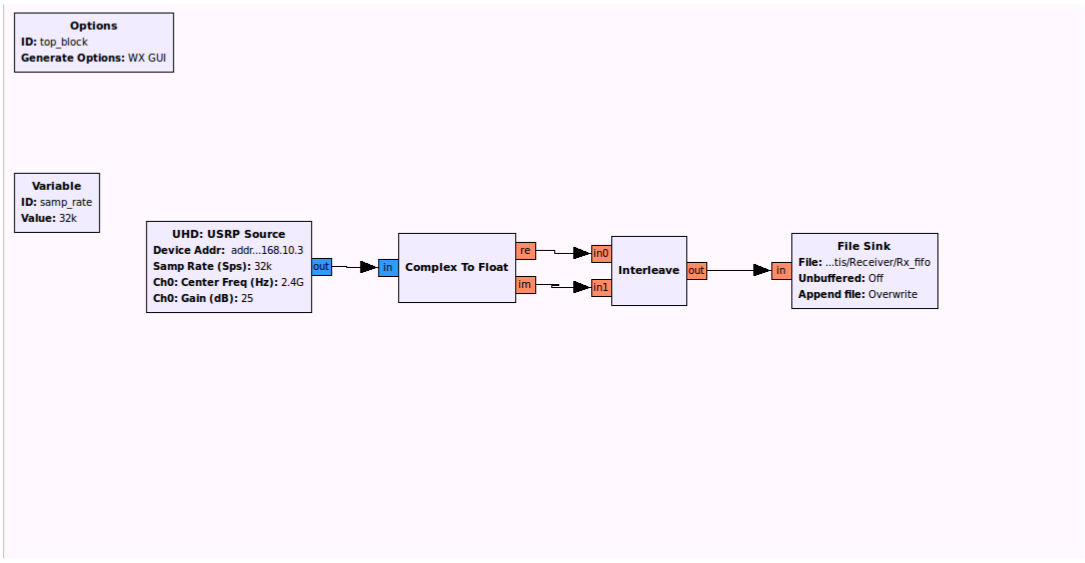
File Source
File: .../Transmitter/Tx_fifo
Repeat: Yes

Deinterleave

Float To Complex

UHD: USRP Sink
Device Addr: addr...168.10.2
Samp Rate (Sps): 32k
Ch0: Center Freq (Hz): 2.4G
Ch0: Gain (dB): 25

FIGURE 2.5: The GNU Radio transmitter we implemented



Options
ID: top_block
Generate Options: WX GUI

Variable
ID: samp_rate
Value: 32k

UHD: USRP Source
Device Addr: addr...168.10.3
Samp Rate (Sps): 32k
Ch0: Center Freq (Hz): 2.4G
Ch0: Gain (dB): 25

File Sink
File: ...tis/Receiver/Rx_fifo
Unbuffered: Off
Append file: Overwrite

FIGURE 2.6: The GNU Radio receiver we implemented

## 2.3 Matlab

All the mathematical processing done in this work has been initially written and tested thoroughly in MATLAB in order to find the best functioning algoritms and methods to create, modify and process the data that is sent/received from/by the corresponding USRPs.

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems.

# 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 divides the entire band of a frequency selective fading channel into many narrow bands at 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 Tc is much larger than the delay spread Td) 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.

**OFDM advantages**

- Makes efficient use of the spectrum by allowing frequency overlap.
- By dividing the channel into narrowband flat fading subchannels, OFDM is more resistant to frequency selective fading than single carrier systems are.
- Eliminates ISI and IFI through use of a cyclic prefix.
- Using adequate channel coding and interleaving one can recover symbols lost due to the frequency selectivity of the channel.
- Channel equalization becomes simpler than by using adaptive equalization techniques with single carrier systems.
- It is possible to use maximum likelihood decoding with reasonable complexity. This is because the DFT technique is used to implement the modulation and demodulation functions.
- OFDM is less sensitive to sample timing offsets than single carrier systems are.
- OFDM provides good protection against cochannel interference and impulsive parasitic noise.

**OFDM disadvantages**

- The OFDM signal has a noise like amplitude with a very large dynamic range, therefore it requires RF power amplifiers with a high peak to average power ratio.

- It is more sensitive to carrier frequency offset and drift than single carrier systems are due to leakage of the DFT. This is because some operations, in this case multiplication with a sinusoidal, create new frequency components that interfere with 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

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

Respectively, for a time-invariant channel we have

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

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 $\mu s$ 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 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

$$h\left(\tau\right) = \sum_i a_i \delta\left(\tau - \tau_i\right), \tag{3.3}$$

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

$$H\left(F\right) = \int_{-\infty}^{+\infty} h\left(\tau\right) e^{-j2\pi F\tau} d\tau = \sum_i a_i e^{-j2\pi f\tau_i}, \tag{3.4}$$

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 $W_c$.

## 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 are 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 transmitted by OFDM modulation. The OFDM signal can be expressed as

$$s\left(t\right) = \sum_{k=0}^{N-1} s_k e^{j2\pi f_k t} = \sum_{k=0}^{N-1} s_k \phi_k\left(t\right), \qquad 0 \leq t \leq T_s\,, \tag{3.6}$$

where:

$$f_k = f_0 + k\Delta f \;\; \text{and} \;\; \phi_k\left(t\right) = \begin{cases} e^{j2\pi f_k t} & \text{if } 0 \leq t \leq T_s \\ 0 & \text{otherwise}\,, \end{cases} \tag{3.7}$$

for k = 0,1,...,N-1, where

| | | |
|---|---|---|
| $s_k$ | : | the $k$-th complex symbol to be transmitted |
| N | : | the number of symbols |
| $\Delta f$ | : | 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 $\Delta f$ 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

$$\begin{aligned}
&\frac{1}{T_s}\int_0^{T_s} \phi_k\left(t\right)\phi_l^*\left(t\right)dt \\
&= \frac{1}{T_s}\int_0^{T_s} \left(e^{-j2\pi f_k t}\right)\left(e^{-j2\pi f_l t}\right)^* dt \\
&= \frac{1}{T_s}\int_0^{T_s} e^{-j2\pi(f_k-f_l)t}dt \\
&= \frac{1}{T_s}\int_0^{T_s} e^{-j2\pi(k-l)\Delta f t}dt \\
&= \delta\left[k-l\right] ,
\end{aligned} \tag{3.8}$$

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

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

Equation 3.3 shows that $\{\phi_k\left(t\right)\}_{k=0}^{N-1} = \{e^{j2\pi f_k t}\}_{k=0}^{N-1}$ is a set of orthogonal functions.Using this property and assuming ideal channel, the OFDM signal can be demodulated by

$$\begin{aligned}
&\frac{1}{T_s}\int_0^{T_s} s\left(t\right)e^{-j2\pi f_k t}dt \\
&= \frac{1}{T_s}\int_0^{T_s}\sum_{k=0}^{N-1} s_l\left(e^{j2\pi f_l t}\right)\left(e^{j2\pi f_k t}\right)^* dt \\
&= \sum_{k=0}^{N-1} s_l\delta[l-k] \\
&= s_k
\end{aligned} \tag{3.9}$$

## 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 $h_l, l=0,\ldots,N-1$. 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

$$y[m] = \sum_{l=0}^{L-1} h_l x[m-l] + w[m], \qquad m = 0,\ldots,N+L-1. \tag{3.10}$$

Let the data block of length N be

$$\tilde{\mathbf{d}} = [\tilde{d_0},\ldots,\tilde{d_{N-1}}]^T \tag{3.11}$$

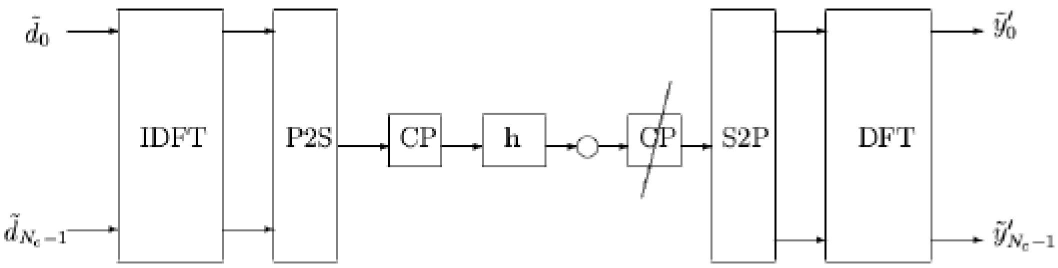


FIGURE 3.1: Digital OFDM System Model

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

$$d = IDFT\left(\tilde{\mathbf{d}}\right) = [d_0, \dots, d_{N-1}]^T \tag{3.12}$$

Using **d**, we construct the vector x, by inserting a cyclic prefix, of length L (equal or bigger of the channel length),

$$x = \begin{bmatrix} d_{N-L} \\ \vdots \\ d_{N-1} \\ d_0 \\ \vdots \\ d_{N-1} \end{bmatrix} = \begin{bmatrix} x_0 \\ \vdots \\ \vdots \\ \vdots \\ x_{N+L-1} \end{bmatrix} \tag{3.13}$$

Therefore, the equation 3.4 can be written as follows

$$y_m = \sum_{l=0}^{L-1} h_l x_{m-l} + w_m, \qquad m = 0, \dots, N+L-1 \tag{3.14}$$

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

$$y[m] = \sum_{l=0}^{L-1} h_l d[(m-L-l) \text{ modulo } N] + w[m] \tag{3.15}$$

After ignoring the first L output symbols, we use the N output symbols $y_m, m = L, \dots, N+L-1$ to construct the vector y′

$$\mathbf{y}' = \begin{bmatrix} y'_0 \\ \vdots \\ y'_{N-1} \end{bmatrix} = \begin{bmatrix} y_L \\ \vdots \\ y_{N+L-1} \end{bmatrix} \tag{3.16}$$

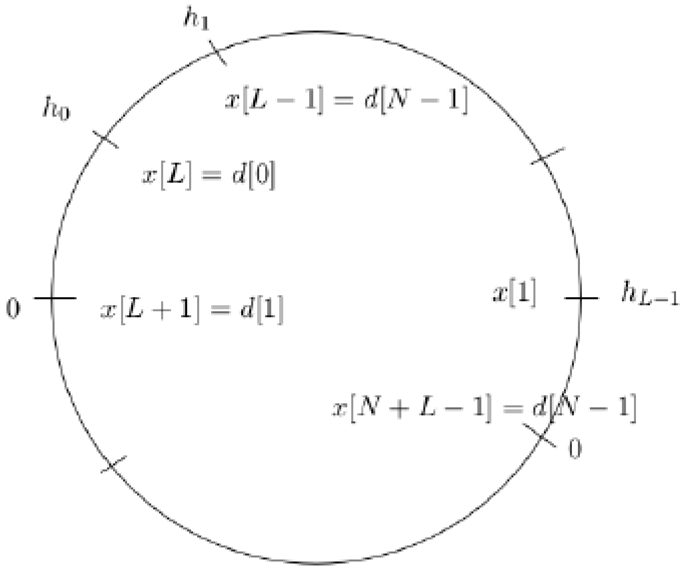


FIGURE 3.2: Convolution between the channel **h** and the input **x** formed from the data symbols **d** by adding a cyclic prefix.

and the vector of channel with length N

$$\mathbf{h} = \begin{bmatrix} h_0 \\ h_1 \\ h_{L-1} \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.17}$$

Now, equation 3.4 can be written as

$$\mathbf{y}' = \mathbf{d} \circledast_N \mathbf{h} + \mathbf{w} \tag{3.18}$$

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

$$y'_0 = y_L = \sum_{l=0}^{L-1} h_l x_{L-l} = h_0 d_0 + \sum_{l=1}^{L-1} h_l d_{N-1} \tag{3.19}$$

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

$$(\mathbf{d} \circledast_N \mathbf{h})_0 = h_0 d_0 + \sum_{l=1}^{L-1} h_l d_{N-1} \tag{3.20}$$

Respectively, we can prove the remaining relations and conclude that

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

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

$$\tilde{d}_n := \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} d[m] e^{\frac{-j2\pi nm}{N}}, \qquad n = 0, \ldots, N-1 \tag{3.21}$$

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

$$\begin{aligned} \tilde{y}' &= DFT\left(y'\right) = DFT\left(\mathbf{d} \circledast_N \mathbf{h} + \mathbf{w}\right) \\ &= DFT\left(\mathbf{d} \circledast_N \mathbf{h}\right) + DFT\left(\mathbf{w}\right) \\ &= \sqrt{N} DFT\left(\mathbf{h}\right) \odot DFT\left(\mathbf{d}\right) + DFT\left(\mathbf{w}\right) \end{aligned} \tag{3.22}$$

where

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \odot \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 y_1 \\ \vdots \\ x_n y_n \end{bmatrix} \tag{3.23}$$

is the element-wise or Hadamard product. Therefore,

$$\tilde{y}_n^{'} = \tilde{h}_n \tilde{d}_n + \tilde{w}_n, \qquad n = 0, \ldots, N-1, \tag{3.24}$$

where $\tilde{h}_n = \sum_{l=0}^{L-1} h_l e^{\frac{-j2\pi ln}{N}}$, n=0, ...,N-1 which is equal to 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.

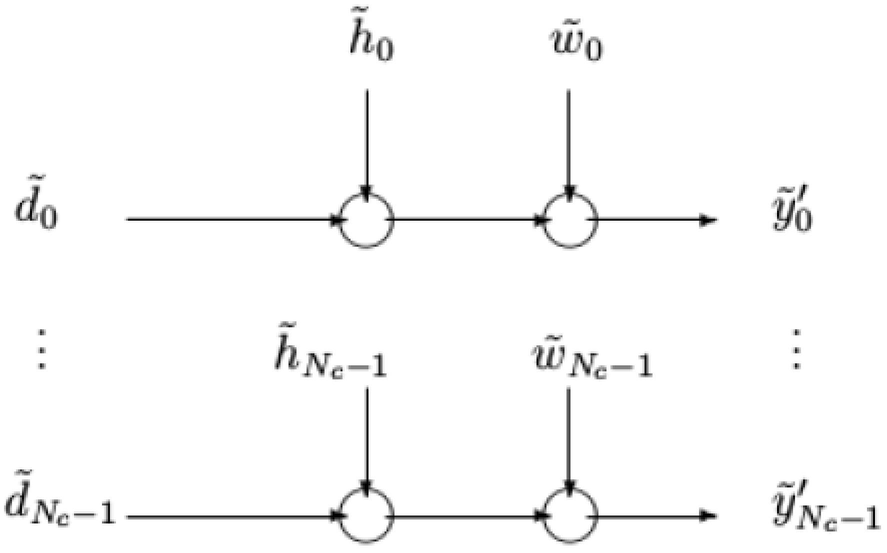


FIGURE 3.3: OFDM converts a wideband channel into a set of N parallel narrowband channels.

# Chapter 4

# OFDM Implementation

## 4.1 Synchronization in OFDM

Synchronizing in an OFDM system can be a great challenge. Schmidl proposed using the autocorrelation of a training symbol with two identical parts to estimate timing and fractional frequency offset. However, Schmidl's timing metric has an uncertainty plateau and the method can yield timing estimates which are well beyond the ISI-free region, thus leading to degradation in BER performance. Other autocorrelation methods have been proposed by Park, Minn and Shi using different uniquely-designed preamble patterns to obtain sharper timing metrics and improve the timing accuracy. However, they are not robust in fading and strong-ISI channels. When speaking of synchronization, most algorithms are based on two important principles, autocorrelation and crosscorrelation. In the autocorrelation process, the signal is correlated with itself, whereas in the crosscorrelation, the signal is correlated with a stored pattern known to the receiver. In this thesis, Minn's [2] synchronization will be used.

### 4.1.1 Coarse time synchronization

A preamble with four parts will be used. The first two parts are symmetric and the last two parts are the negative of the two first parts. This synchronization symbol is sent before every OFDM symbol in order to perform accurate sync every time an OFDM symbol is received. The sync symbol's structure is as follows:

$$S_{Minn} = \begin{bmatrix} d_{N/4} & d_{N/4} & -d_{N/4} & -d_{N/4} \end{bmatrix}$$

This preamble is created by generating the one forth of the preamble size and using it only every other tone in the OFDM symbol. By using IDFT, we get the result depicted in Figure 4.1.

The coarse timing estimate, where $y\left(d\right)$ is the received sequence, is derived as follows:

$$P\left(d\right) = \sum_{k=0}^{1} \sum_{m=0}^{\frac{N}{4}-1} y^{*}\left(d + k\frac{N}{2} + m\right) y\left(d + k\frac{N}{2} + m + \frac{N}{4}\right),$$

$$R\left(d\right) = \sum_{k=0}^{1} \sum_{m=0}^{\frac{N}{4}-1} \left| y\left(d + k\frac{N}{2} + m + \frac{N}{4}\right)\right|^{2}.$$

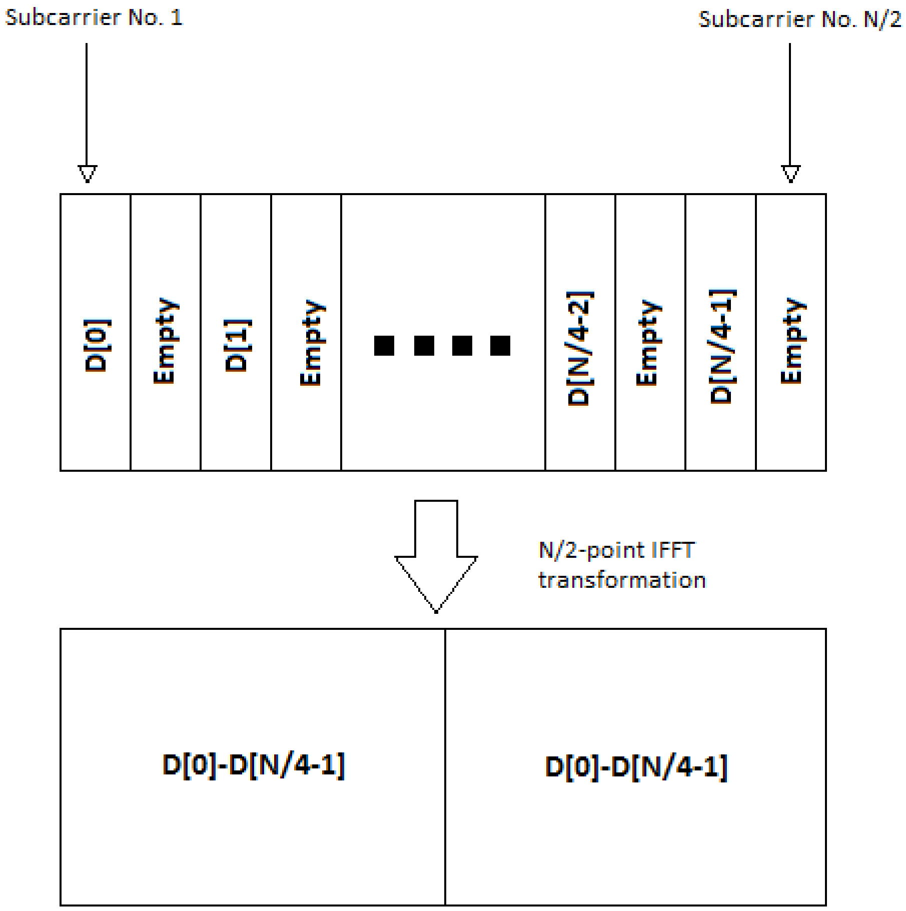


FIGURE 4.1: Structure of the synchronization symbol or preamble $S_{Minn}$.

The final timing metric is given by:

$$M\left(d\right)=\frac{\left|P\left(d\right)\right|^{2}}{\left(R\left(d\right)\right)^{2}}.$$

By using Minn's synchronization method, we bring negative values for the correlation function near the correct, thereby this method can eliminate the 'peak plateau phenomenon' by Schmidl's way. Figure 4.2 shows the results.

In order to achieve coarse timing synchronization, we select the sample at which the final metric has the highest value. Let's denote this specific sample $\hat{d}_c$.

**Note:**For sample spaced sequences, the quantifiers $P\left(d\right)$ and $R\left(d\right)$ are computed using the following equations:

$$P\left(d\right)=\sum_{k=0}^{1}\sum_{m=0}^{\frac{N}{4}-1}y^{*}\left(d+\left(k\frac{N}{2}+m\right)\cdot over\right)y\left(d+\left(k\frac{N}{2}+m+\frac{N}{4}\right)\cdot over\right), \tag{4.1}$$

$$R\left(d\right)=\sum_{k=0}^{1}\sum_{m=0}^{\frac{N}{4}-1}\left|y\left(d+\left(k\frac{N}{2}+m+\frac{N}{4}\right)\cdot over\right)\right|^{2}, \tag{4.2}$$

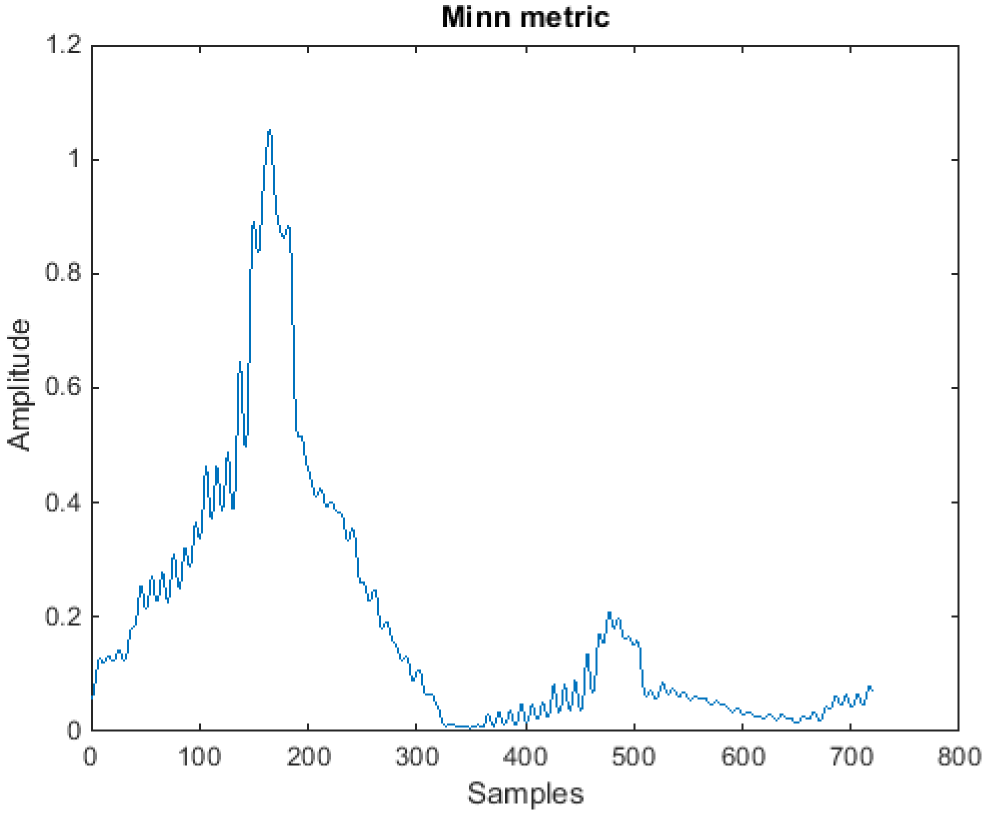


FIGURE 4.2: Coarse time synchronization statistic $M(d)$ of Minn's metric for a frequency selective channel.

where $over$ stands for oversampling factor.

Our coarse estimate for the length-N received synchronization sequence is:

$$\begin{aligned} y_0^{'} &= y_{\hat{d}_{c_1}} \\ y_1^{'} &= y_{\hat{d}_{c_1}+1} \\ &. \\ &. \\ &. \\ y_{N-1}^{'} &= y_{\hat{d}_{c_1}+N-1} \end{aligned}$$

Assuming that the received sequence is sample spaced, meaning that the estimate $\hat{d}_c$ was derived by equations 4.1 and 4.2, we can get the symbol spaced sequence $\{y_l^{'}\}_{l=0}^{N-1}$ by sampling at the correct time. Meaning:

$$\begin{aligned} y_l^{'} &= Y\left(\hat{d}_c T_s + lT\right) = Y(t)\,|_{t=\hat{d}_c T_s + lT} \\ &= e^{j\left(2\pi \Delta F\left(\hat{d}_c T_s + lT\right)+\phi\right)} \sum_n s_n h\left(\hat{d}_c T_s + lT - nT\right) + W\left(\hat{d}_c T_s + lT\right) \\ &= e^{j\left(2\pi \Delta F \hat{d}_c + \phi\right)} e^{j2\pi(\Delta F T)l} \sum_n s_n h_{l,n}^{\hat{d}_c} + w_l^{'} \end{aligned} \tag{4.3}$$

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

$$h_{l,n}^{'} = e^{j\left(2\pi\Delta f\hat{d}_c+\phi\right)} h_{l,n}^{\hat{d}_c} \tag{4.4}$$

then, for l=0,...,N-1,

$$y_l^{'} = e^{j2\pi(\Delta FT)l} \sum_n s_n h_{l,n}^{'} + w_l^{'} = e^{j2\pi\Delta f^{'} l} r_l^{'} + w_l^{'}, \tag{4.5}$$

where $\Delta f^{'} = \Delta F \cdot T$ and $r_l^{'} = \sum_n s_n h_{l,n}^{'}$.

### 4.1.2 Estimation and Correction of Carrier Frequency Offset (CFO)

Carrier frequency offset (CFO) is one of many non-idealities in baseband receiver design. CFO occurs when the local oscillator signal for down-conversion in the receiver does not synchronize with the carrier signal contained in the received signal. This phenomenon can be attributed to two important factors: frequency mismatch in the transmitter and the receiver oscillators and the Doppler effect as the transmitter or the receiver is moving.

When this occurs, the received signal will be shifted in frequency. For an OFDM system, the orthogonality among sub carriers is maintained only if the receiver uses a local oscillation signal that is synchronous with the carrier signal contained in the received signal. Otherwise, mismatch in carrier frequency can result in inter-carrier interference (ICI). The oscillators in the transmitter and the receiver can never be oscillating at identical frequency. Hence, carrier frequency offset always exists even if there is no Doppler effect.

So, our goal here is to mitigate the frequency offset in order to retrieve all our symbols successfully. We can easily take the samples that correspond to the received part of the preamble (without its CP) by using the coarse timing estimate $\hat{d}_c$.

Assuming that the received symbols corresponding to the first quarter of the training block are given by:

$$y_k^{'} = e^{i2\pi\Delta f^{'} k} r_k^{'} + w_k^{'}, \qquad \text{k =1,...,} \frac{N}{4}\ , \tag{4.6}$$

respectively, the symbols in the second quarter take the form

$$y_{k+\frac{N}{4}}^{'} = e^{i2\pi\Delta f^{'}\left(k+\frac{N}{4}\right)} r_{k+\frac{N}{4}}^{'} + w_{k+\frac{N}{4}}^{'}, \qquad \text{k =1,...,} \frac{N}{4}\ , \tag{4.7}$$

where $r^{'}_k$ and $r^{'}_{k+\frac{N}{4}}$ are identical.Consequently,

$$\begin{aligned}
\left(y^{'}_k\right)^* y^{'}_{k+\frac{N}{4}} &= \left(e^{j2\pi\Delta f^{'}k} r^{'}_k + w^{'}_k\right)^* \left(e^{j2\pi\Delta f^{'}(k+\frac{N}{4})} r^{'}_{k+\frac{N}{4}} + w^{'}_{k+\frac{N}{4}}\right) \\
&= \left(e^{-j2\pi\Delta f^{'}k}\left(r^{'}_k\right)^* + \left(w^{'}_k\right)^*\right)\left(e^{j2\pi\Delta f^{'}(k+\frac{N}{4})} r^{'}_{k+\frac{N}{4}} + w^{'}_{k+\frac{N}{4}}\right) \\
&= e^{-2\pi\Delta f^{'}k} e^{2\pi\Delta f^{'}(k+\frac{N}{4})}\left(r^{'}_k\right)^* r^{'}_{k+\frac{N}{4}} + e^{-2\pi\Delta f^{'}k}\left(r^{'}_k\right)^* w^{'}_{k+\frac{N}{4}} + \\
&\quad + e^{2\pi\Delta f^{'}(k+\frac{N}{4})} r^{'}_{k+\frac{N}{4}}\left(w^{'}_k\right)^* + \left(w^{'}_{k+\frac{N}{4}}\right) \\
&= e^{j2\pi\Delta f^{'}(k+\frac{N}{4}-k)}|r^{'}_k|^2 + \tilde{w}_k \\
&= e^{j\pi\Delta f^{'}\frac{N}{2}}|r^{'}_k|^2 + \tilde{w}_k,
\end{aligned}$$

where

$$\tilde{w}_k = e^{-2\pi\Delta f^{'}k}\left(r^{'}_k\right)^* w^{'}_{k+\frac{N}{4}} + e^{2\pi\Delta f^{'}(k+\frac{N}{4})} r^{'}_{k+\frac{N}{4}}\left(w^{'}_k\right)^* + \left(w^{'}_{k+\frac{N}{4}}\right) \tag{4.8}$$

Ignoring the noise part, if we take the argument of $y^*_k y_{k+\frac{N}{4}}$ and use all the samples of the preamble part, an estimate of CFO, $\hat{\Delta} f$, can be derived as:

$$\hat{\Delta} f = \frac{1}{\pi\frac{N}{2}} arg\left(\sum_{k=0}^{\frac{N}{4}-1}\left(y^{'}_k\right)^* y_{k+\frac{N}{4}}\right), \tag{4.9}$$

because,

$$arg\left(\sum_{k=0}^{\frac{N}{4}-1}\left(y^{'}_k\right)^* y_{k+\frac{N}{4}}\right) = arg\left(e^{j\pi\Delta f^{'}\frac{N}{2}}\sum_{k=0}^{\frac{N}{4}-1}|r^{'}_k|^2\right) = \pi\hat{\Delta} f\frac{N}{2}, \tag{4.10}$$

We derived that the sum of the product of the corresponding samples in the preamble is almost equal (since we ignored the noise part) to a sinusoidal with frequency equal to the unknown CFO, times a real constant. After calculating $\hat{\Delta} f$, we can use it to eliminate the effect of CFO by creating the following sequence:

$$ycor^{'}_k = e{-}j2\pi\hat{\Delta} f k y^{'}_k \tag{4.11}$$

where $ycor^{'}_k$ is the corrected sequence of all the samples of the packet that are only affected from the channel.

It would be interesting to see what would happen if the CFO was not cancelled. Every symbol would be multiplied by an exponential that would have unitary magnitude but a non-zero phase. This exponential is a function of time $t$, meaning that every symbol would be multiplied by a different factor. As a result the scatterplot, of a constellation whose CFO would not have been cancelled, would look circular, creating arches. Such a scatterplot is depicted in Figure 4.3.

### 4.1.3 Fine time synchronization

Our final step in the synchronization method is to accurately find the exact sample to start demodulating our OFDM symbols. So, we now have the frequency corrected

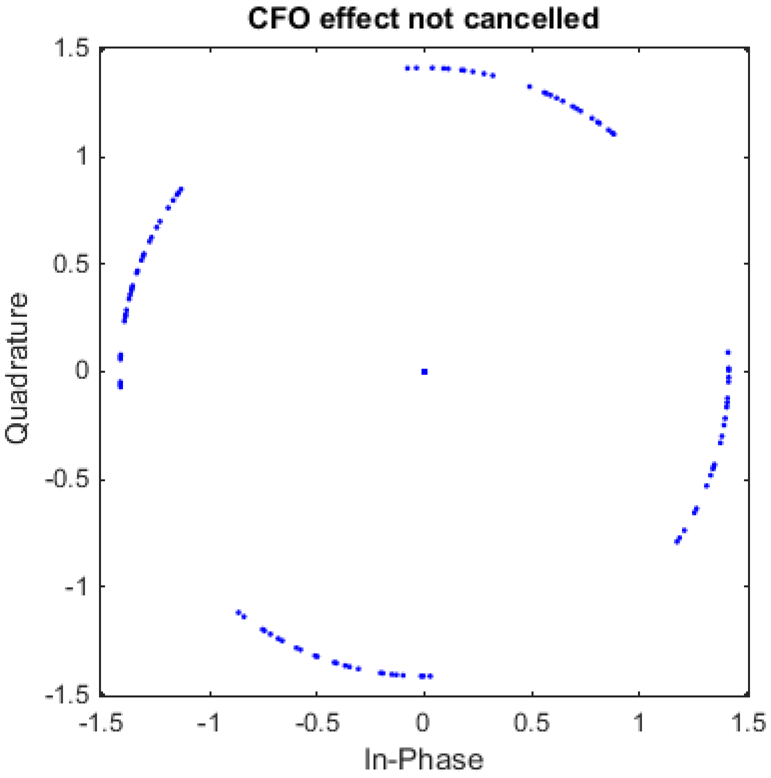


FIGURE 4.3: The scatterplot before CFO cancellation.

signal $y_{cor}^{'}$ and we apply the cross-correlation stage that is summarized as follows:

$$P_x\left(d\right) = \sum_{k=0}^{N-1} y_{cor}^{'}\left(d+k\right) S_{Minn}^{*}\left(k\right) \tag{4.12}$$

$$d \in \left\{\hat{d}_c - \frac{N}{2}, \hat{d}_c + \frac{N}{2}\right\} \tag{4.13}$$

Now, by using the Minn timing metric and the result from the equation above, we calculate the final statistic $M_{opt}$:

$$M_{opt} = \left|P_x\left(d\right)\right|^2 M\left(d\right) \ ,$$

$$d \in \left\{\hat{d}_c - \frac{N}{2}, \hat{d}_c + \frac{N}{2}\right\}$$

Finally, we acquire the fine timing estimate as:

$$\hat{d}_opt = argmax\{M_{opt}\} \, ,$$

$$d \in \left\{\hat{d}_c - \frac{N}{2}, \hat{d}_c + \frac{N}{2}\right\}$$

It is important to note that in cases of channels with frequency selective fading, we have to use a sliding window with size of M symbols that calculates the sum of the energies of the samples in it. In that way, we will be able to find the correct start of the OFDM symbol even if the $M_{opt}$ statistic has multiple peaks due to frequency selectivity. We have to gather the sums of energies from the areas of interest and then we set as the start, the first position of the window with the greatest sum. As a result, we are sure that we have taken all the great peaks of $M_{opt}$ in mind.

$$\text{fine sync win}\,(n) = \sum_{n=0}^{M-1} |M_{opt}\,(n+m)\,|^2 \tag{4.14}$$

$$n \in \{\hat{d}_c - \frac{N}{2}, \hat{d}_c + \frac{N}{2}\}$$

$$\hat{d}_opt = argmax\{\text{fine sync win}\,(n)\}$$

After getting the estimate $\hat{d}_{opt}$ from any method, we have to go $M$ symbols backwards in order to also get the CP of the preamble in the packet that we obtain.

In Figures below, we see the spectrum of a flat fading channel and the spectrum of a frequency selective channel as well as their corresponding synchronization estimates. In Figure 4.4, we see the spectrum of an OFDM symbol that is sent with rate equal to 100 Msamples/sec or 10 Msymbols/sec due to our oversampling factor that equals to 10. It is clear that we deal with a flat fading channel (or else channel with short delay spread) as the frequency response is flat, without notches. In Figure 4.5, we see the synchronization statistic $M_{opt}$ of an OFDM symbol that is sent through the channel described above. The fact that our channel is flat fading, can be confirmed from the one and only great peak that appears in the figure.

In Figure 4.6, we see the spectrum of an OFDM symbol that is sent with rate equal to 100 Msamples/sec or 10 Msymbols/sec due to our oversampling factor that equals to 10. It is clear that we deal with a frequency selective channel (or else channel with long delay spread) as the frequency response is not flat but presents a notch that degrades the frequencies that correspond to the notch. In Figure 4.7, we see the synchronization statistic $M_{opt}$ of an OFDM symbol that is sent through the channel described above. The fact that our channel is frequency selective can be confirmed from the two great peaks that appear in the figure.

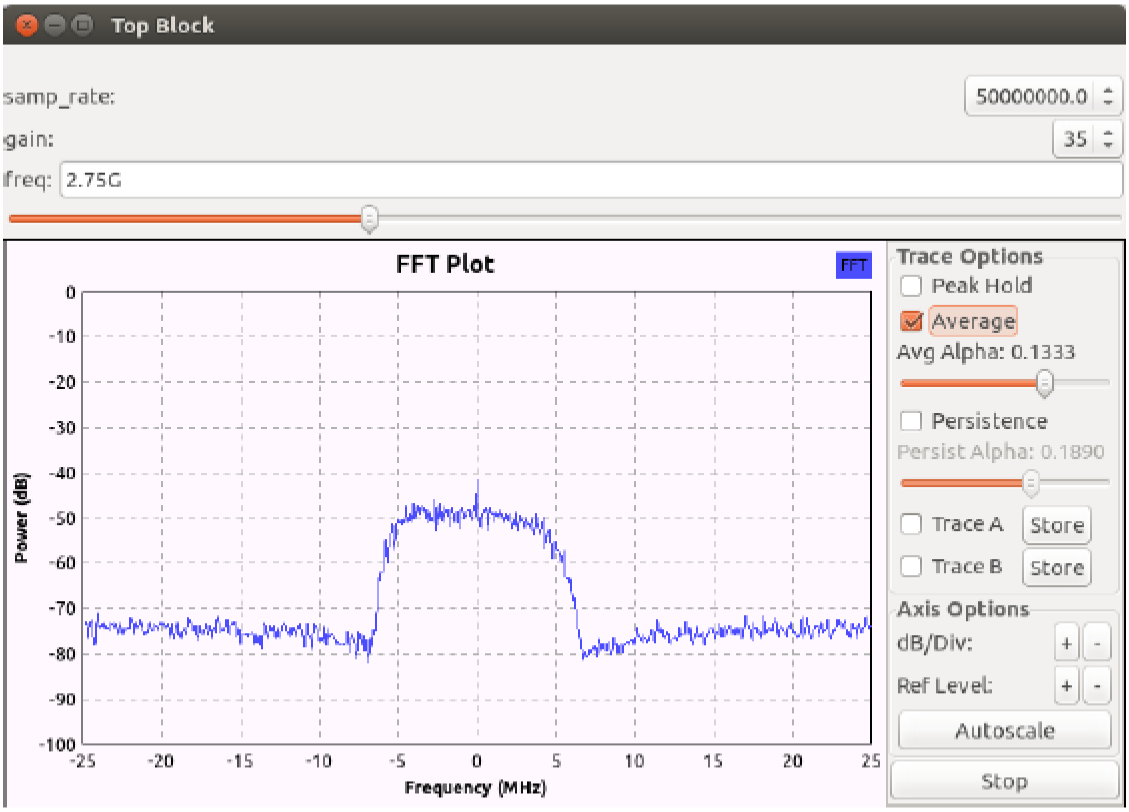
## 4.2 Channel Estimation

In our experiments, the channel can be either frequency flat or frequency selective. This depends on the configuration settings (e.g. transmission rate) or the environment conditions that lead to short or long delay spread of the transmitted signal. Thus, we have to introduce a way to reliably estimate the channel. In this section, we describe the channel estimation procedure. After time synchronization and CFO correction as mentioned before, the received synchronization symbol (preamble including CP) can be expressed as:

$$y_m^{'} = \sum_{l=0}^{M-1} h_l s_{m-l} + w_m, \qquad m = 0, ..., N+M-1\,, \tag{4.15}$$

or in vector form,

$$\mathbf{y} = \mathbf{h} * \mathbf{s} + \mathbf{w} \tag{4.16}$$



Top Block

samp_rate: 50000000.0

gain: 35

freq: 2.75G

Trace Options

☐ Peak Hold

☑ Average

Avg Alpha: 0.1333

☐ Persistence

Persist Alpha: 0.1890

☐ Trace A

☐ Trace B

Axis Options

FIGURE 4.4: Spectrum of received signal for a frequency flat channel.

Analytically, the above convolution can be written as (assuming the maximum channel's length $M$ is equal to 4):

$$y^{'} = \begin{bmatrix} h_0 s_0 \\ h_0 s_1 + h_1 s_0 \\ h_0 s_2 + h_1 s_1 + h_2 s_0 \\ h_0 s_3 + h_1 s_2 + h_2 s_1 + h_3 s_0 \\ \vdots \\ h_0 s_{N+M-1} + h_1 s_{N+M-2} + h_2 s_{N+M-3} + h_3 s_{N+M-4} \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_{N+M-1} \end{bmatrix}, \tag{4.17}$$

or in other form,

$$y^{'} = \begin{bmatrix} s_0 & 0 & 0 & 0 \\ s_1 & s_0 & 0 & 0 \\ s_2 & s_1 & s_0 & 0 \\ s_3 & s_2 & s_1 & s_0 \\ \vdots & \vdots & \vdots & \vdots \\ s_{N+M-1} & s_{N+M-2} & s_{N+M-3} & s_{N+M-4} \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_{N+M-1} \end{bmatrix} \tag{4.18}$$

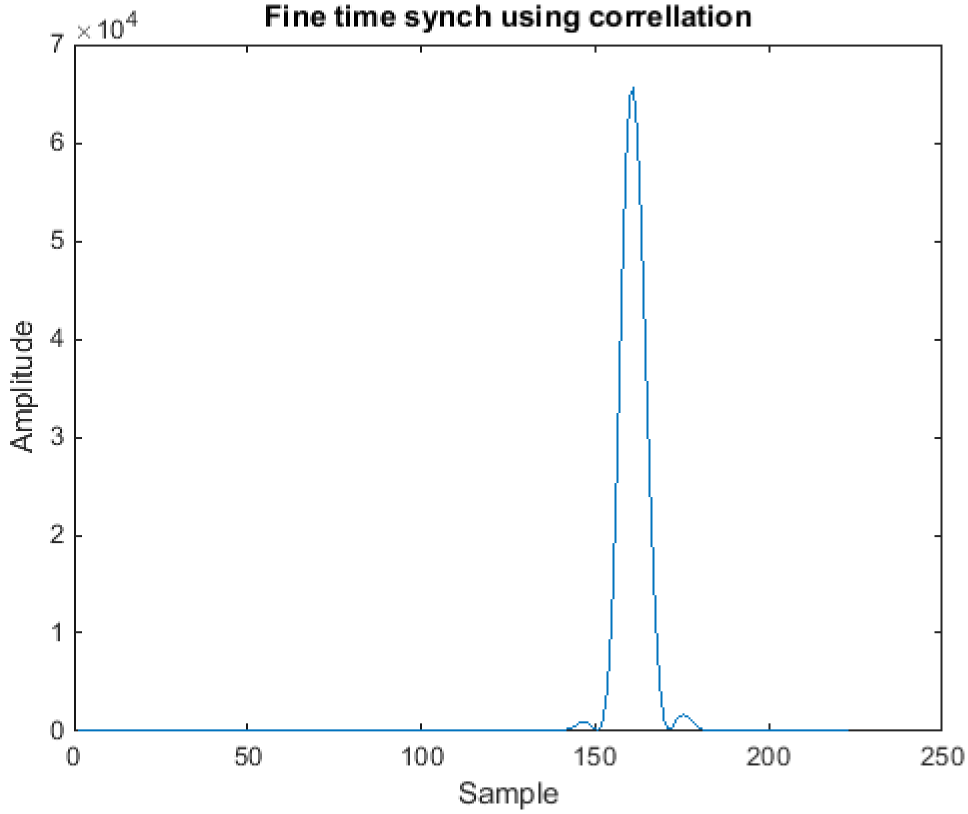


FIGURE 4.5: Synchronization statistic $M_{opt}$ in flat fading channel.

Assuming that $s$ is known (i.e. $s$ is the preamble used previously for synchronization), we have to create a Toeplitz matrix with is, as follows:

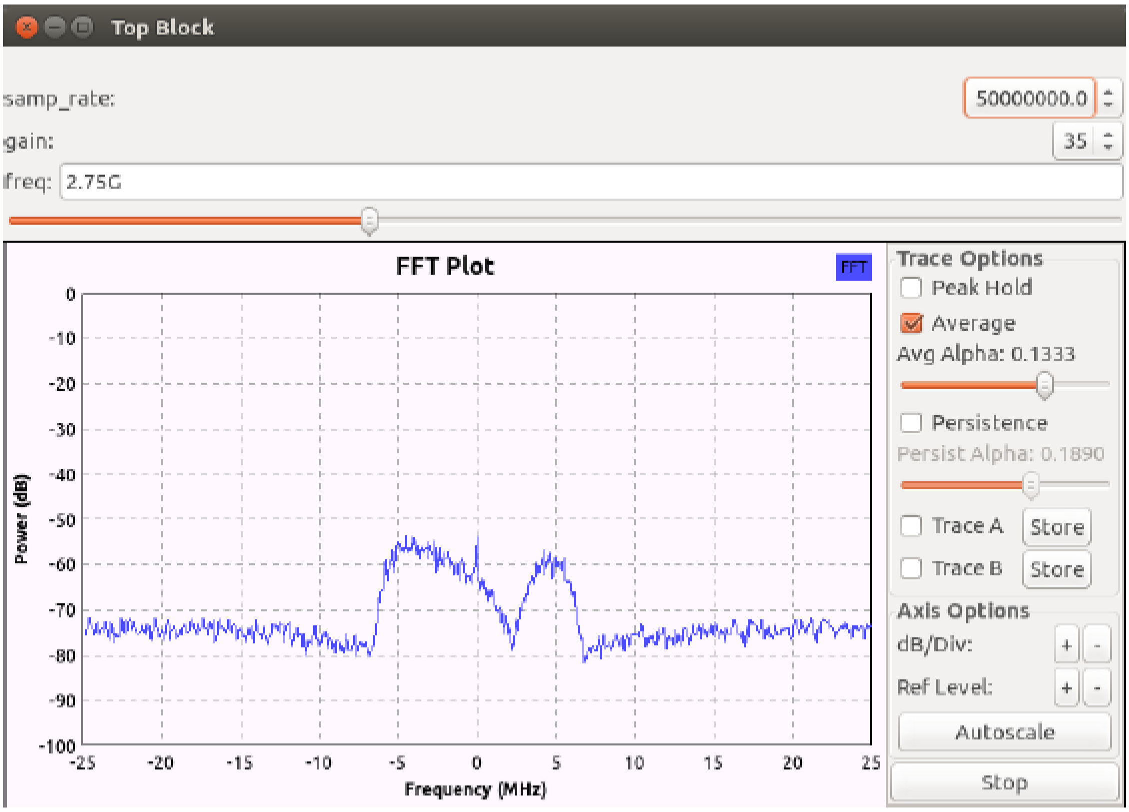
$$\mathbf{s}_{toep} = \begin{bmatrix} s_0 & 0 & 0 & 0 \\ s_1 & s_0 & 0 & 0 \\ s_2 & s_1 & s_0 & 0 \\ s_3 & s_2 & s_1 & s_0 \\ \vdots & \vdots & \vdots & \vdots \\ s_{N+M-1} & s_{N+M-2} & s_{N+M-3} & s_{N+M-4} \end{bmatrix}, \tag{4.19}$$

with $(N + M)$ rows and $M$ columns. The Toeplitz matrix is used in order to calculate all the channel's values (maximum $M$ values). In case of a frequency selective channel, some of these values (number equal to the channel's real length) will have comparable magnitude while, in case of a frequency flat channel, only one value will be noticeably greater than the other $M - 1$ values. This method operates also perfectly for flat channels as channels $h_1$,$h_2$ and $h_3$ will have zero or almost zero value, so only the first column of $y^{'}$ will have useful information to retrieve the value of $h_0$ (the other three channel values will return zero, or almost zero value, as expected from a flat channel).

Then, the Least Squares (LS) channel estimate is given by:

$$\mathbf{h}_{LS} = (\mathbf{s}_{toep}\mathbf{s}_{toep})^{-1}\,\mathbf{s}_{toep}y^{'}, \qquad \text{with size}\,(\mathbf{h}_{LS}) = M\,, \tag{4.20}$$

where $.^H$ denotes the Hermitian transpose.



samp_rate: 50000000.0

gain: 35

freq: 2.75G

☐ Peak Hold

☑ Average

Avg Alpha: 0.1333

☐ Persistence

Persist Alpha: 0.1890

☐ Trace A

☐ Trace B

FIGURE 4.6: Spectrum of received signal for a frequency selective channel.

Here, it is important to point out the impact of the way we choose the parameter $M$. If $M$ is chosen to be greater than our channel length, then some of these $M$ least squares channel estimates will be zero. In that case, the demodulation will be performed successfully, since the non-zero values are the ones who have impact on our transmitted symbols.

On the other hand, if the channel length is greater than our selected $M$, then our estimate will get the "important part" of the channel, meaning the one with the greatest energy. In that case, we will have problems when trying to demodulate using the least squares estimate, depending on how big the portion of the channel we lost is. In Figures 4.8 and 4.9 we show the LS estimates for a frequency selective channel. The continuous line is the actual channel, where the diracs are the estimate. It is clear that the channel length we have assumed, $M$, is greater than the actual length. This is why we see so many values close to zero.

## 4.3 Correction of channel effect and symbol estimation

After acquiring our OFDM symbol $y$ of length $N$ from the synchronized and frequency corrected sequence as described previously, we perform a $N$-point FFT transformation to it. In addition, we perform another $N$-point FFT transformation to the $M$ channel values that we calculated in the previous section.

$$\mathbf{Y} = fft\left(y, N\right), \tag{4.21}$$

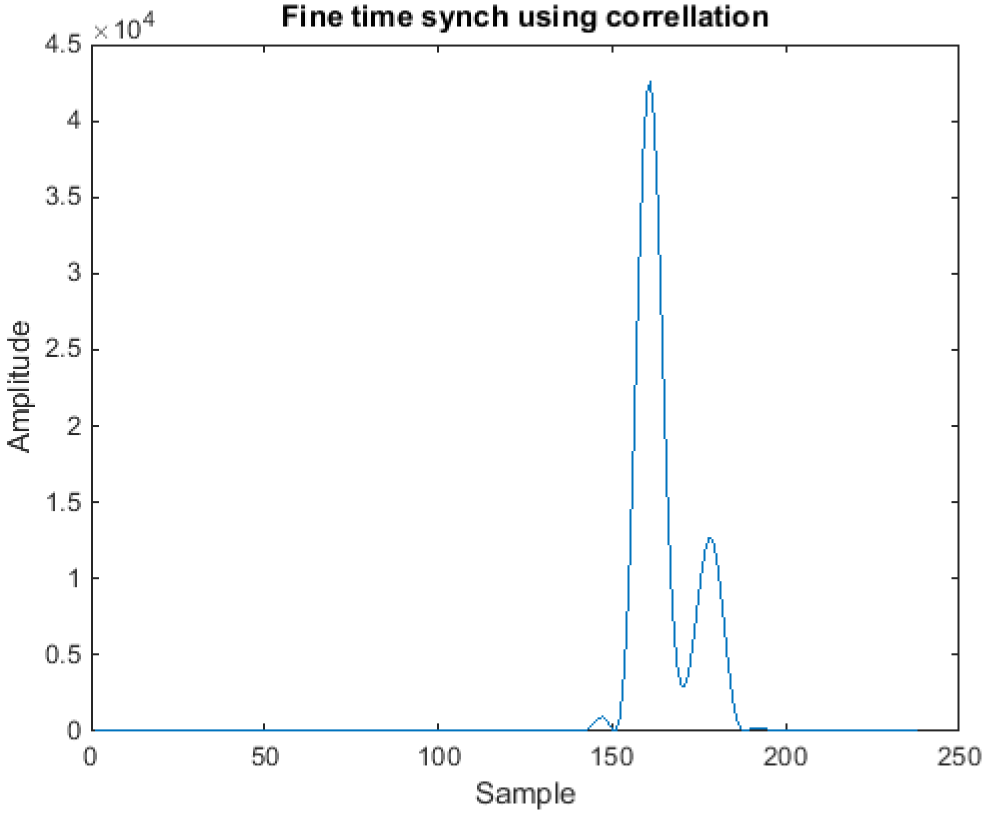


FIGURE 4.7: Synchronization statistic in frequency selective channel.

$$\mathbf{H}_{LS} = fft\left(\mathbf{h}_{LS}, N\right) . \tag{4.22}$$

Then, we can easily perform symbol-by-symbol channel correction

$$\text{rxsymbols} = \frac{\mathbf{H}_{LS}^{*} \cdot \mathbf{Y}}{\left|\mathbf{H}_{LS}\right|^{2}} , \tag{4.23}$$

with $(.^*)$ the element-by-element multiplication and $(.)^2$ the element-by-element exponentiation to 2 (the division is also element-wise).

The final step is to have the symbol decisions on the rx symbols vector's elements. Since we have equiprobable symbols and the noise is modelled as a zero mean normal random variable, we can use the nearest neighbour rule. This rule is applied easily by looking at the sign of the real and imaginary part of each symbol.

We can see a typical scatterplot of a received OFDM symbol that has emerged from 4-QAM modulated data symbols in Figure 4.10. It emerges after a complete equalization is performed.

## 4.4 Implementation Details

This section explains the technical details of the algorithms developed for the transmitter and the receiver.

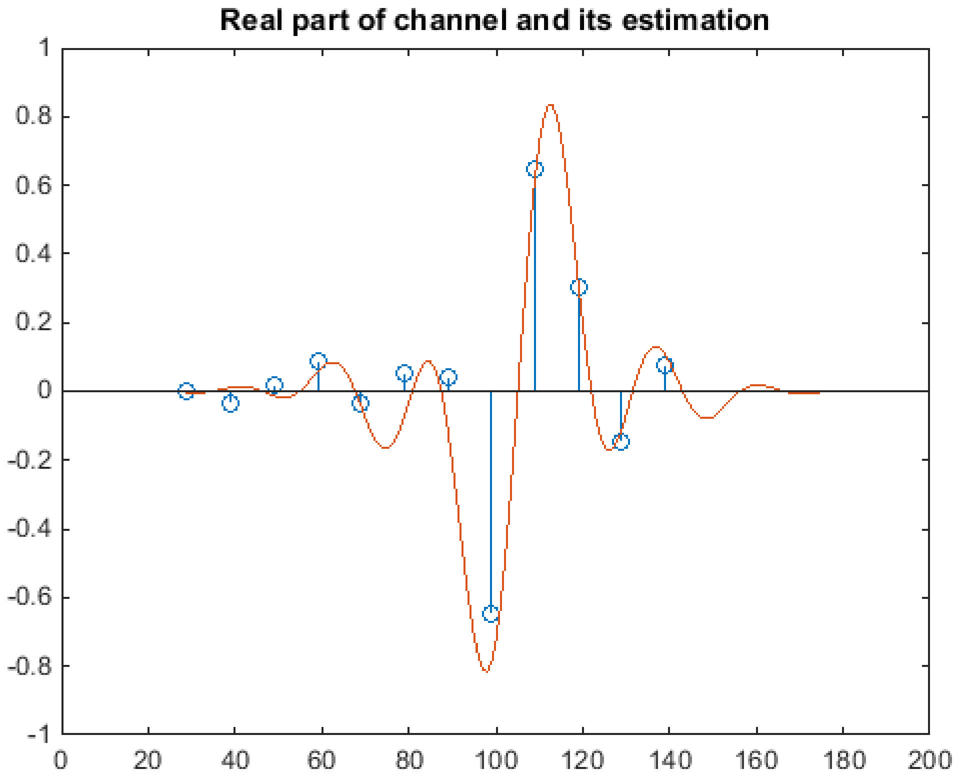


FIGURE 4.8: The real part of LS estimate of frequency selective channel.

### 4.4.1 Specifications

In our implementation, we use the preamble that was proposed by Minn. This preamble is better described in section 4.1.1. We use this synchronization symbol before each OFDM packet, which contains the useful data. The synchronization algorithm runs every time we receive a packet. Between each packet, we consider a gap of zeros, so we can detect each packet easily.

Every OFDM data packet consists of 128 symbols ($N$=128). These symbols are generated from 4-QAM source. We also consider that each preamble consists of $N$ symbols too. The preamble remains the same for each packet. The length of the guard interval we use is 16 ($M$=16). This means that the $M$ last symbols of the packet are concatenated in the beginning of the packet. This cycle prefix is used before the preamble too, so in total we have $(2M + 2N)$ symbols that is equal to 288 symbols. Figure 4.11 describes our packet more efficiently.

This packet is up sampled with an oversampling factor of 10. This means that after every symbol we consider a gap of 10 zeros. So, the total samples are $(2M + 2N) \cdot over$. Every upsampled packet is convolved with a Squared Root Raised Cosine (SRRC) filter which is described later on.

Then, the final signal is send by the transmitter to the receiver. There, an algorithm for detecting packets is used. This method is described later on. Finally, since we have detected the packet, the synchronization and equalization algorithms are used.

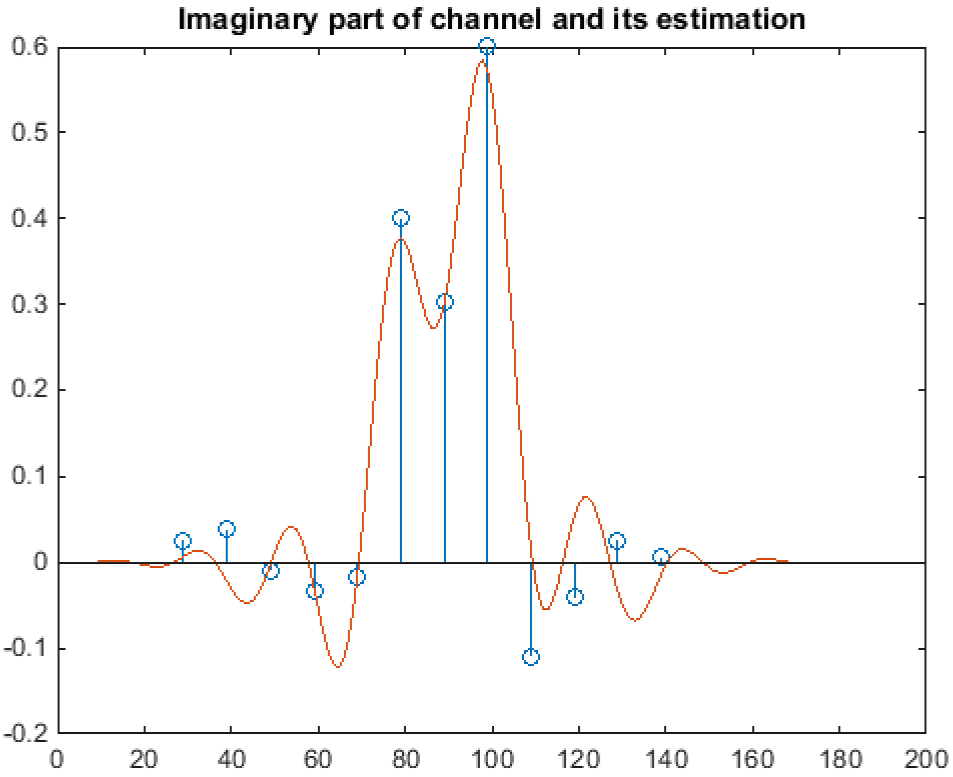


FIGURE 4.9: The imaginary part of LS estimate of frequency selective channel.

### 4.4.2 Square Root Raised Cosine (SRRC) filter

In signal processing, a Root Raised Cosine filter (RRC), sometimes known as Square Root Raised Cosine filter (SRRC), is frequently used as the transmitter and receiver filter in a digital communication system. A SRRC filter is used at the transmitter to perform spectrum-shaping transmission, therefore we call it pulse shaping filter, while in the receiver we call it as matched filter. The combined response of two such filters, is that of the raised-cosine filter, which does not suffer from ISI.

One of the parameters for creating a SRRC filter it the roll-ff factor. This parameter ranges from 0 to 1. The greater the roll-off factor the lesser the "ripples" the filter makes at the edges. In Figure 4.12 we plot the SRRC filter for some values of the roll-off factor.

### 4.4.3 Initial packet detection

The first and ceaseless job of a receiver is to be able to detect any incoming transmissions. This procedure, also known as packet synchronization, hat to overlook any possible peaks of noise and compute a rough estimate for the beginning of any incoming packet. In order to accomplish that, we use a Double Sliding Window (DSW) algorithm for packet detection. The DSW algorithm uses two consecutive sliding windows to calculate the sum of energies of the samples in each window. At each window movement, the ratio of these two sums is calculated and if it exceeds a designated threshold, a counter is increased by one. If this counter reaches a specific

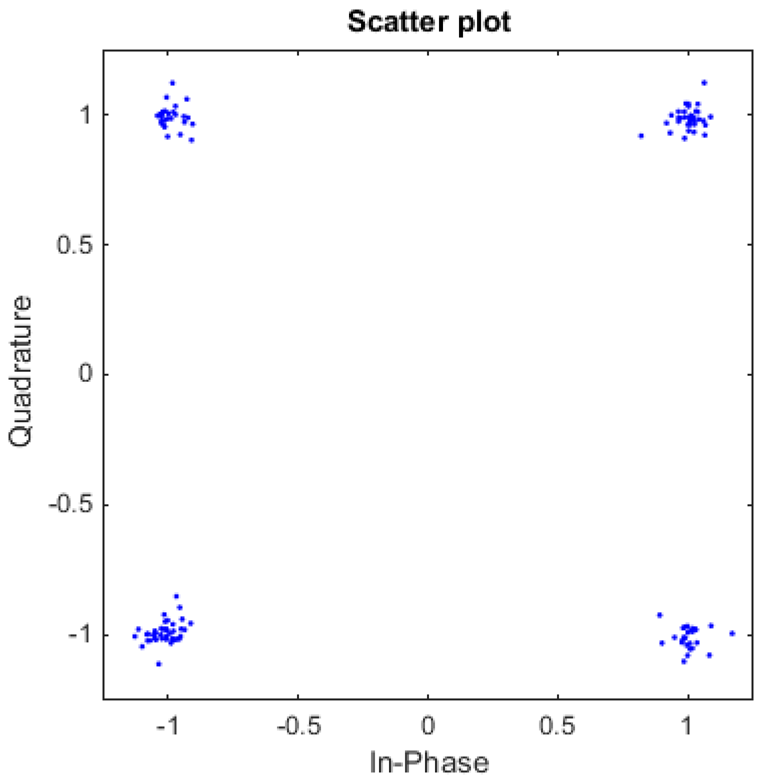


FIGURE 4.10: Scatterplot of a received OFDM symbol using 4-QAM constellation.

limit, then we know that we have found a packet and after estimating a rough start, its synchronization begins as described previously. On the other hand, each time the ratio of the two sums is lower than the threshold, the counter becomes zero.

The way we choose the energy ratio threshold and the counter limits is crucial in order to successfully detect a packet. The counter limit must be equal to the window length. This is because, when the right window gets completely inside the signal, then the second window will start coming inside the signal too. In that case, the ratio will start decreasing. Figure 4.13 explains this phenomenon. The energy ratio threshold depends on the distance of the transmitter and the receiver. In our case, the greater the distance of the USRPs the lesser the threshold, while when the USRPs are close (there is line of sight) the threshold is high.

The length of each of these windows is:

$$W_{len} = over \cdot A\,, \tag{4.24}$$

where $A$ is the half-period duration is symbols.

We start the window from the beginning of the signal we have to process. We calculate the energy of the samples in the right window and then the energy of the samples in the left window. We compute the ratio of those two energies. If the ratio is greater that the threshold set by the user, then we increase the counter, else we set it to zero. If the counter reaches the length of the window, as mentioned before, we assume that we have found a packet and we start the synchronization algorithm.

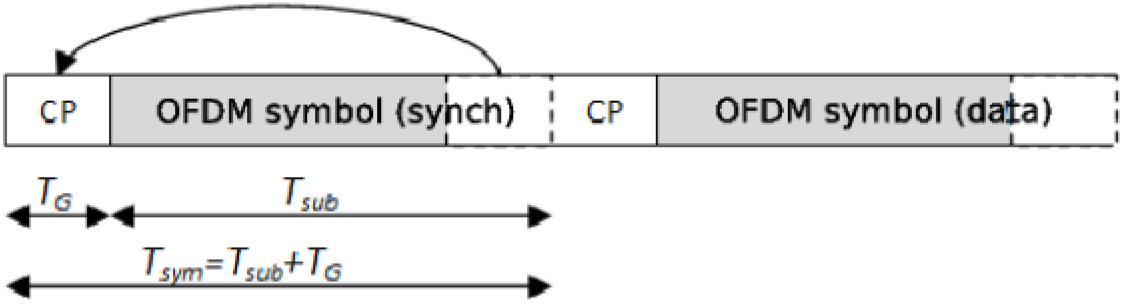


FIGURE 4.11: Structure of a packet in our implementation.

The equations used are:

$$\text{right window}[d] = \sum_{i=0}^{W_{len}-1} \left| y\left(d+i\right) \right|^2 \tag{4.25}$$

$$\text{left window}[d] = \sum_{i=0}^{W_{len}-1} \left| y\left(d+i+W_{len}\right) \right|^2 \tag{4.26}$$

$$\text{ratio}[d] = \frac{\text{left window}[d]}{\text{right window}[d]} \tag{4.27}$$

The process above is illustrated for better understanding.

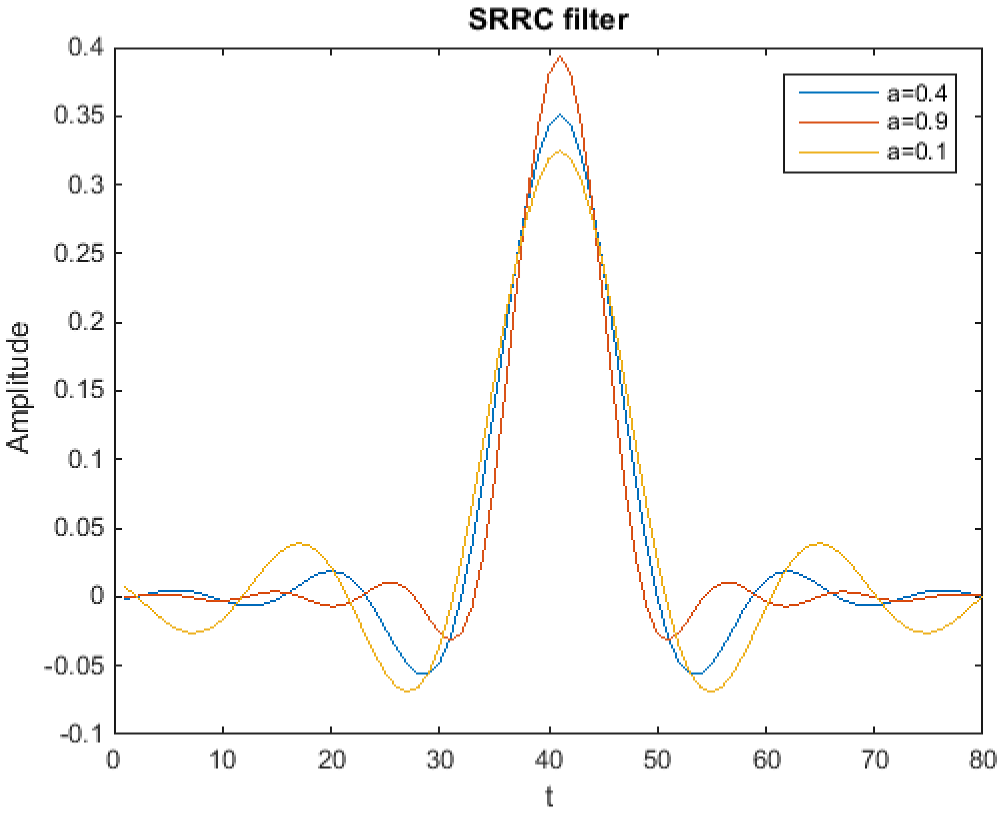


FIGURE 4.12: SRRC filter in time domain.

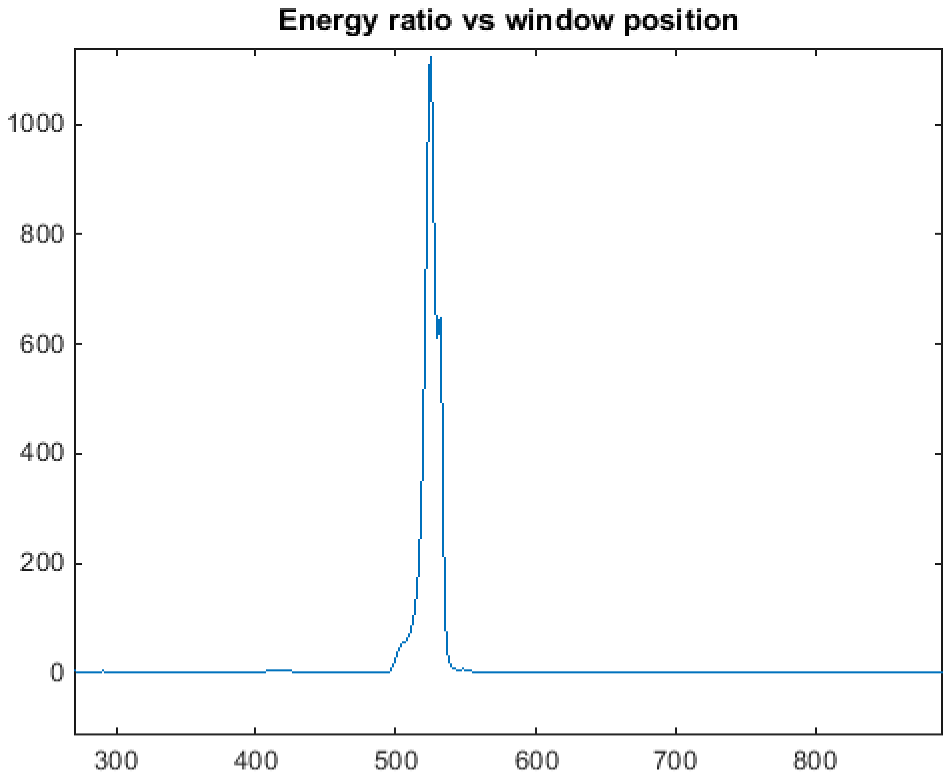


FIGURE 4.13: After the first window is completely inside the signal the ratio starts decreasing.

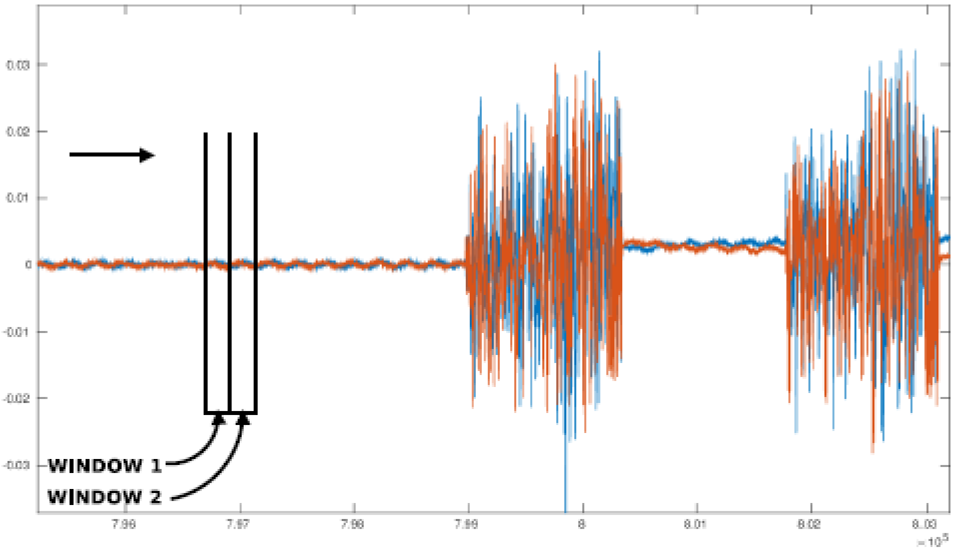


FIGURE 4.14: Initial set up of windows.

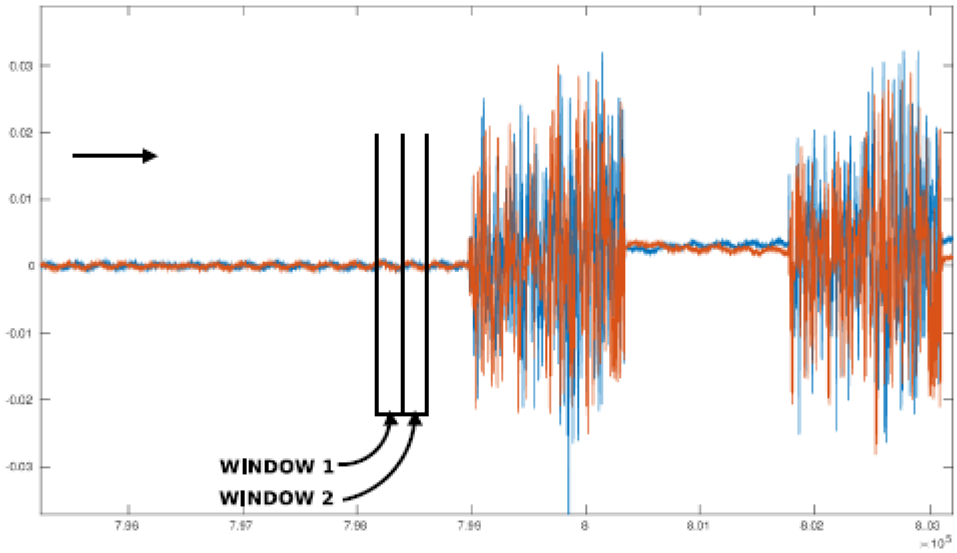


FIGURE 4.15: The two windows sliding towards a packet.

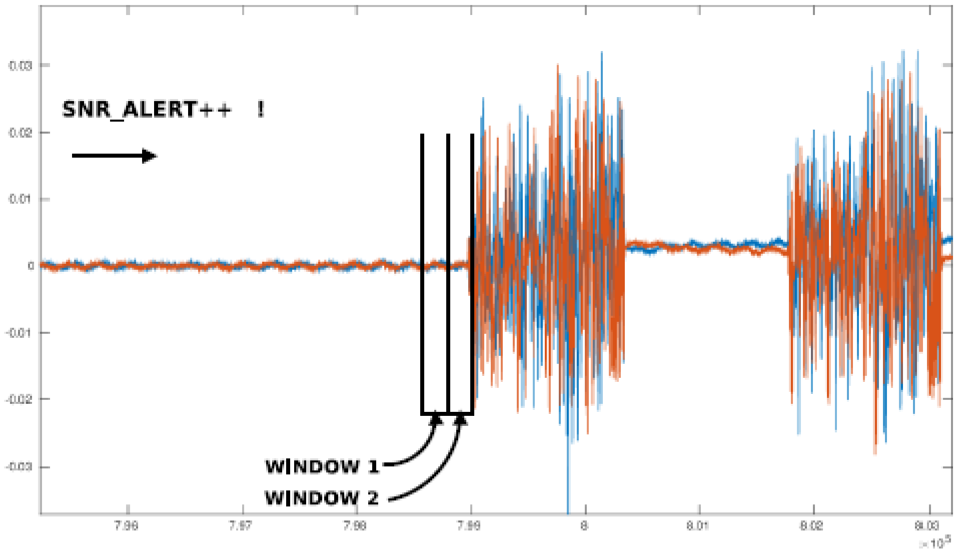


FIGURE 4.16: The right window gets a part of a packet, counter is raised.

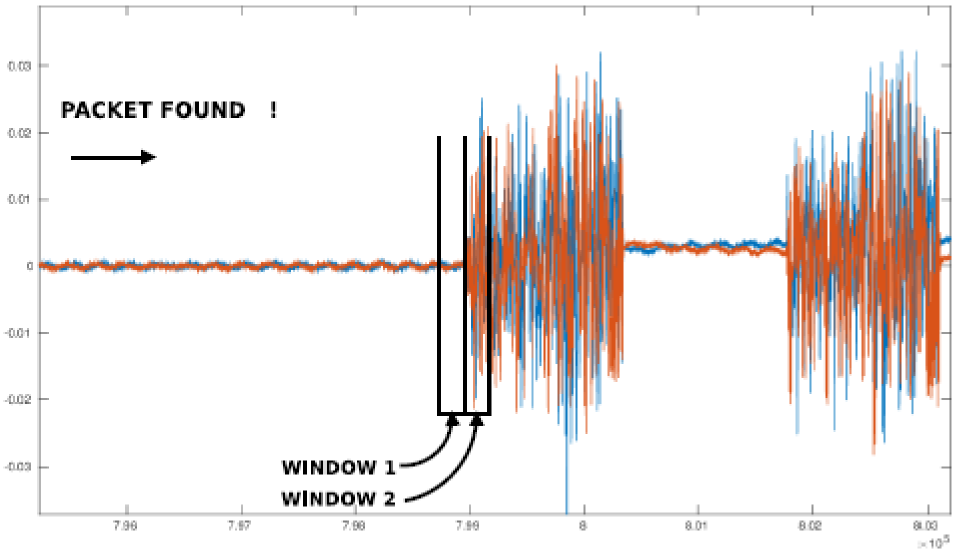


FIGURE 4.17: As the windows slide to the right, the counter reaches its limit. This means we found a packet.

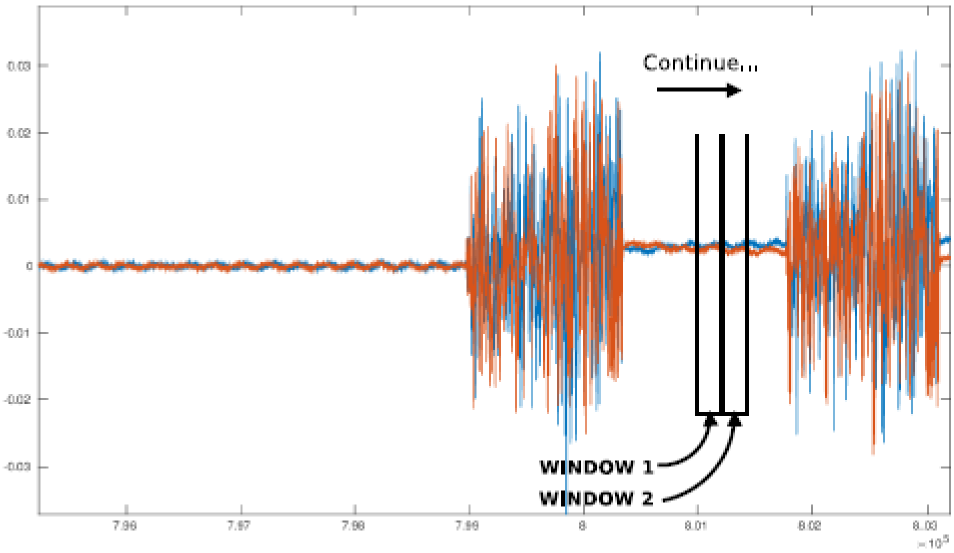


FIGURE 4.18: The windows get out of the packet to continue for further detections.

# Chapter 5

# Future Work

## 5.1 Future Work

There are many ways this thesis can be further improved and evolved.

For example we could expand this model so that it consists of more receiving or transmitting antennas (MIMO model).Alamouti's scheme is a popular technique that could be used in order to implement this goal.This is a way that the diversity could be enhanced.

Also, we could use error detecting and correction codes.In that way, we would achieve a better BER ratio at the receiver and the reliability of the whole system in general would improve.

Of course, these are just some ideas that could be used in order to improve the current thesis.The subject is vast and there are many more topics on which someone can work in the future.

# Bibliography

[1] Athanasios P. Liavas. Lecture notes on OFDM.
`Technical University Of Crete`

[2] Chao Chen, Zhanxin Yang. Three Timing Synchronization Methods Based on Two Same Preambles for OFDM Systems.
`Procedia Engineering`., 29:1656-1661, June 2012

[3] Athanasios Gkiolias. Implementation of Smart Radios with OFDM in SDR platform USRP N200 and Application in a Cognitive Radio Protocol
`Technical University of Crete`, November 2015

[4] Ettus Research Website.
`https://www.ettus.com`

[5] Mounir Ghogho, Ananthram Swami. Carrier Frequency Synchronization for OFDM Systems.
*Signal Processing for Mobile Communications Handbook*