FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Wireless Leash

Vasco Miguel Gonçalves Correia

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Cândido Duarte (PhD)

Second Supervisor: Filipe Santos (PhD)

July 14, 2017

Abstract

The emergence of Wireless Sensor Networks, and the paradigm of the Internet of Things, during the twenty first century, has led to new market opportunities. Among the explored applications, ranging has been one of the most interesting ones. Despite the scientific community best efforts, it still exists a quite extensive discussion in terms of the best solution, capable of offering a cost-effective system, high accuracy and wide coverage range.

In this work, a novel architecture for precise ranging accuracy is proposed. By exploiting the freed spectrum, in the sub-GHz region (the so called TV White Spaces), and combining it with a spectral efficient OFDM modulator, high throughput and node scalability is achieved, while mitigating multipath propagation. Combined with the Symmetrical Double-Sided Two-Way Ranging algorithm, a custom solution can be designed to achieve high location resolution, while maintaining a wide coverage range.

The main focus of the hardware development of this dissertation was given to the core module of the OFDM modulator and demodulator, i.e.: the FFT and IFFT, respectively. An efficient design of a custom Fast Fourier Transform, with an embedded CORDIC unit, in fixed-point notation, was proven, capable of achieving 0.0284 Msamples/s/slice, when synthesized for FPGA implementation.

Resumo

O aparecimento das redes de sensores sem fios, e do paradigma da Internet das Coisas, durante o século XXI, levou a que surgissem novas oportunidades de mercado. Entre as aplicações exploradas, localização tem sido um dos mais interessantes. Apesar dos esforço da comunidade científica, ainda existe uma discussão decorrente em qual a melhor solução, capaz de oferecer um sistema economicamente viável, grande precisão e alcance abrangente.

Neste trabalho, uma nova arquitectura para localização é proposta para medições precisas. Utilizando o espectro deixado livre, na região sub-GHz (os denominados espaços brancos de TV), e combinado-o com um modulador e desmodulador OFDM, espectralmente eficiente, alta taxa de débito e escalabilidade de nós pode ser alcançada, mitigando a propagação multi-caminho, em simultâneo. Combinado com o algoritmo "Symmetrical Double Sided Two-Way Ranging", uma solução personalizada pode ser desenhada para obter grande precisão de localização, mantendo, simultaneamente, uma cobertura de sinal abrangente.

O principal foco do desenvolvimento em hardware desta dissertação foi dada ao módulo fulcral para a modulação OFDM, ou seja, a FFT. Um projecto eficiente de uma solução personalizada da Transformada Rápida de Fourier, com uma unidade CORDIC embebida, em notação em vírgula fixa, foi provada, capaz de alcançar 0.0284 milhões de amostras/s/fatia, quando sintetizado para implementação em FPGA.

Agradecimentos

Após 3 batalhas perdidas, eis que finalmente venço a guerra!

Primeiro que tudo, gostaria de agradecer aos meus pais, sem os quais nem estaria, neste momento, a apresentar este documento, pela oportunidade que me ofereceram de estudar na Faculdade de Engenharia da Universidade do Porto.

Em segundo ao meu orientador e co-orientador, o professor Cândido Duarte e o Filipe Santos, pelo tema de tese oferecido, e, sobretudo, ao primeiro, pelas enxaquecas que lhe devo ter causado algures entre Junho de 2016 e Janeiro de 2017.

À Diana Soares, Pedro "Espaço" Diogo, Nuno Cardoso e Filipe Gonçalves, pela *wake-up call*, o meu grande obrigado.

À Joana Santos Leite e Carolina Sousa, pelos café nocturnos, a meio da semana, para ganhar mais um bocado de coragem para continuar a batalhar no dia seguinte.

Ao Bernardo "Maquetista" Silva, Pedro "Pintor" Aguiar, Bruno "Corrector" Ferreira, Cristhian "Rodriguez" Caldeira e Rui "Chuck" Feio pelos 5 a 7 anos de amizade, 5 deles, pelo menos, a partilhar um curso que, para mim, termina com este documento.

A todos aqueles que não mencionei neste documento, mas que sabem que me representam algo, vocês não ficaram esquecidos, simplesmente já custa a escrever...

Vasco Correia

"If I have seen further it is
by standing on the shoulders of Giants."

Isaac Newton

"Words build bridges
into unexplored regions."

Contents

List of Figures

List of Tables

Abbreviations and Acronyms

AP	Access Point
ASIC	Application Specific Integrated Circuit
BPSK	Binary Phase-Shift Keying
BS	Base Station
BW	Bandwidth
CIR	Channel Impulse Response
CORDIC	COordinate Rotation DIgital Computer
CR	Cognitive Radio
CRLB	Cramer-Rao Lower Bound
DB	Database
DFT	Discrete Fourier Transform
DIF	Decimation-In-Frequency
DIT	Decimation-In-Time
DSP	Digital Signal Processing
DUT	Design Under Test
FCC	Federal Communications Commission
FDM	Frequency Division Multiplexing
FFT	Fast Fourier Transform
FIFO	First-In First-Out
FMCW	Frequency-Modulated Continuous-Wave
FPGA	Field-Programmable Gate Array
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System
HDL	Hardware Description Language
ICI	Inter-Channel Interference
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
ISI	Inter-Symbol Interference
ISM	Industrial, Scientific and Medical
IoT	Internet of Things
LOS	Line-of-Sight
LSR	Linear Shift-Register
MAC	Medium Access Control
OFCOM	Office of Communications
OFDM	Orthogonal Frequency Division Multiplexing
PHY	Physical Layer

QAM	Quadrature-Amplitude Modulation
RF	Radio Frequency
RMS	Root-Mean Squared
RSS	Received Signal Strength
RTOF	Round-trip Time-of-Flight
RU	Remote Unit
SDF	Single-Delay-Feedback
SDS-TWR	Symmetric Double-Sided Two-Way Ranging
SNR	Signal-to-Noise Ratio
TOA	Time-of-Arrival
TOF	Time-of-Flight
TV	Television
TVBD	TV Band Device
TVWS	TV White Spaces
TWR	Two-Way Ranging
UHF	Ultra-High Frequency
UWB	Ultra-Wideband
VHF	Very-High Frequency
WLAN	Wireless Local Area Networks
WSN	Wireless Sensor Networks

Chapter 1

Introduction

With the emergence of Wireless Sensor Networks (WSN) and the Internet of Things (IoT), during the twenty first century, new context-aware applications systems have been developed, to provide unforeseen market opportunities. Such examples include inventory and people tracking, surveillance systems, virtual immersion and augmented reality applications, among others. Such market is forecasted to grow to as much as 1.8 billion US dollars, by 2024 [1].

Among these newly found systems, wireless positioning systems have become very popular. From detection of products stored in a warehouse, location detection of medical personnel or equipment in a hospital, the widespread access to wireless information has led to a high demand for accurate positioning in wireless networks, for both indoor and outdoor environments [2].

Global Positioning System (GPS) is, currently, the most widely used, satellite-based, positioning system, which offers the maximum coverage (theoretically, worldwide). However, it can not be deployed, for example, in environments where line-of-sight transmission between satellites and receivers is not feasible. For example, in river valleys, mountain slopes decrease the number of achievable GPS satellite signals, required to obtain an accurate ranging estimate. On the other hand, for certain applications, higher accuracy than GPS may be of the essence, while keeping infrastructure cost, size and power consumption at a minimum.

To accomplish such objective, the scientific community has been using very well known radio frequency (RF) systems, like Wireless Local Area Networks (WLAN) transceivers, by modifying them to address ranging and localization. However, there are various obstacles (like walls, equipment and, even, human beings) which influence the propagation of electromagnetic waves. These combined effects, lead to severe environmental conditions, such as multipath propagation and additional path losses, which, combined with interference and noise sources (from other wired or electrical equipment), degrade the quality and accuracy of positioning measurements [3].

1.1 Context

Under the Project ROMOVI, autonomous machinery is being deployed in the Douro river vineyards of Porto [4–7]. The main purpose of this project is to target and manage threats, like plagues,

to the grape cultivation. These machinery must be capable of accurately track their position to properly roam along the slopes of the river valley, in an efficient and safe manner. Unfortunately, GPS coverage is limited, and all other current solutions (mostly based on simulatenous localization and mapping and sensor fusion algorithms) have not been capable of offering sufficient ranging precision, either due to inherent inaccuracy or insufficient coverage range. Furthermore, the dense vineyard vegetation and river slopes, offer a rich multipath environment, leading to the requirement of proper techniques (or custom solutions) to accommodate both ranging and communication requirements.

1.2 Objective of this work

In this work, a novel architecture is proposed, capable of achieving high location accuracy, while attempting to minimize the device complexity, cost and size. The proposed system is intended to aid the, aforementioned, vineyard autonomous machinery, through the design of a custom RF transceiver, capable of handling both data and ranging packet transmission, with minimal effort.

As such, this work proposes the use of Orthogonal Frequency Division Multiplexing (OFDM) modulation, in the baseband digital signal processing chain, in order to minimize multipath fading effects. By employing sub-GHz carrier frequencies, in the so-called *TV White Spaces* (TVWS), signal penetration (of the dense vineyard vegetation) due to lower attenuation coefficients [8], improves path propagation losses.

This work does not intend to demonstrate the complete solution, but an analysis of the requirements for the development of such an architecture and the steps performed to achieve that purpose.

1.3 Document structure

Given a brief presentation of the work developed, the remaining document is structured as follows:

1. Chapter 2: presentation of a brief overview of currently used geolocation techniques and algorithms;
2. Chapter 3: an overview on TVWS and OFDM will be explored and the proposed architecture explained;
3. Chapter 4: demonstration and implementation of an area-efficient FFT, capable of being integrated in a FPGA or ASIC solution;
4. Chapter 5: presentation of the tests performed on the implemented modules and respective results;
5. Chapter 6: conclusions and future work.

Chapter 2

Fundamentals on Positioning Systems

In this chapter, a brief overview will be presented on the most well known technologies and aspects of positioning systems. Since the development of the Global Navigation Satellite System (GNSS), many different applications have grown, such as enhancement of communication protocols, inventory tracking or even virtual immersion for augmented reality. All of these were only achieved due to the recent developments of geo-location techniques, either for indoor or outdoor usage.

2.1 Global navigation satellite system

Among the existing global navigation satellite systems (GNSS), the global positioning system (GPS) is the most known. Developed during the 1970s [9, 10], due to the nuclear threat during the Cold War arms race, it was initially purposed as a system which would enable ballistic missile submarines to acquire a precise fix on their position, before launching the projectiles.

It consists on 24 nominal satellites (currently, 31 are operational), spread on a constellation that ensures worldwide signal coverage of at least 4, at any given moment in time, orbiting in the medium earth orbit (MEO), yielding an orbital period of 12 hours. Communication is performed by modulating a carrier signal in the L-Band, with a Binary Phase Shift Keying (BPSK) digital modulation scheme. Since all satellites operate, continuously, in parallel and on the same frequency, code division multiple access (CDMA) is used to encode the broadcasted communication signal.

To ensure precise location, very stable atomic clocks are used, with an accuracy of at least 1ns, with further techniques being applied to account for special relativity. By employing the Time of Arrival (TOA) technique[1] to at least three signals, a GPS receiver can compute its current position, by comparing the time delays to the expected almanac (i.e.: the current state of the GPS satellites).

[1] More information on TOA, will be given in 2.2.3.1.

2.2 Geolocation techniques

Typically, geolocation techniques can be sorted into 3 types, according to the type of measure performed, to obtain the distance between a base station (BS) and remote unit (RU)[2] device:

1. Received Signal Strength (RSS) to distance (path loss models);
2. RSS or signal correlation to angle of arrival (usually, using directional antennas);
3. Delay time to distance (proportional to the speed of light or sound, according to the medium).

2.2.1 Received signal strength

Probably the most known method in literature, the Received signal strength (RSS) technique makes usage of the well known free space path loss equation. It is also the most used in geolocation devices, since most transceivers already have a RSS Indicator (RSSI), for employing automatic gain control (AGC). The used path loss model quantifies the received power, at a RU, with the delivered power (by the BS) to estimate the distance between the two, according to:

$$P(d)\ [dB] \quad = \quad P_{d_0}\ [dB] - \gamma * 10\log_{10}\left(\frac{d}{d_0}\right) + S_n \tag{2.1}$$

where $P(d)$, P_{d_0} and γ represent the current received power, the received power at a reference distance d_0 of the transmitter and the path loss exponent, respectively. S_n is a Gaussian-distributed random variable, with zero mean and standard deviation σ_S, which represents the variation of the path loss around its local mean, due to shadowing or fast-fading Doppler processes. An example evolution of the path loss versus distance between devices, is depicted in fig. 2.1.

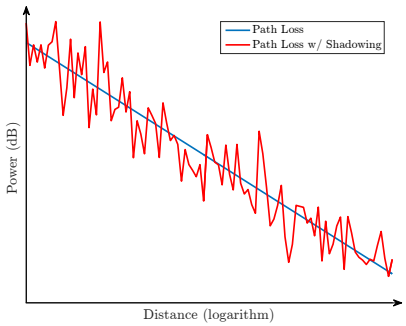


Figure 2.1: Evolution of the received signal strength with distance.

2.2.1.1 Fingerprinting

Due to the static nature of the path loss model (if, shadowing is ignored), some authors have employed what is named the fingerprinting method. An array of similar devices is spread out in

[2]To facilitate readability on this section, base station will refer to the device requesting a ranging metric and remote unit as the device being interrogated.

different known positions and a database is created with the evaluated RSS measurements, from at least 4 access points (AP), also at known locations. When a new node appears, its signal power signature is compared in all APs, and a position estimation is created (usually employing a K-Nearest Neighbours, KNN, algorithm). An example scenario can be found in fig. 2.2, where the black dots represent the array of devices used to create the RSS database signatures.

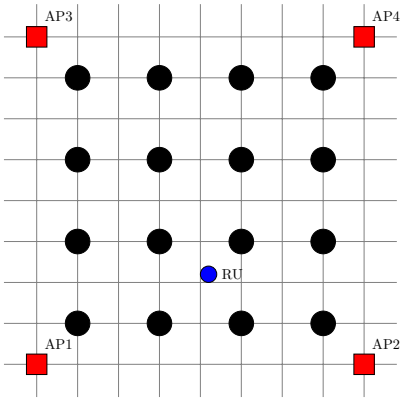


Figure 2.2: Example of a RSS fingerprinting scheme.

2.2.1.2 Angle of arrival

Through the usage of the RSS metric, a different position estimate can be performed. By comparing the signal strength of a set of BS (with directional antennas), a remote unit can assess its own position, by evaluating the different angles of arrival (AOA) with respect to the signal power received, as can be seen in fig. 2.3.

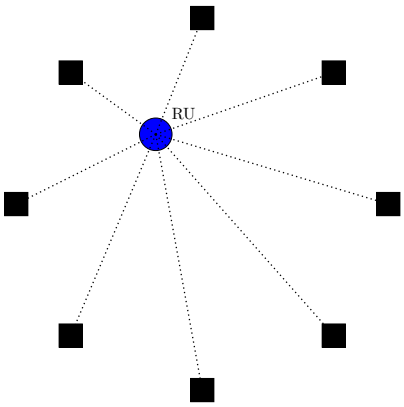


Figure 2.3: Example of an angle of arrival scenario.

2.2.2 Phase of arrival

With some similarities to the AOA, the phase of arrival (POA) technique compares the phase offset between two time-delayed received replicas, of the same signal (for example, through a node with two separate antennas). A simplified example of a phase offset measure can be seen in fig. 2.4. It can easily be demonstrated that the time delay (and, consequently, the distance

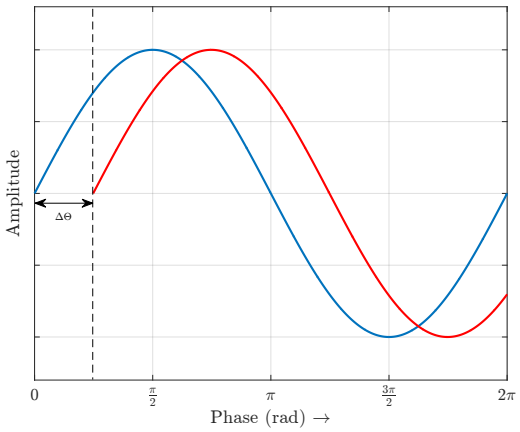


Figure 2.4: Phase offset between two known signals.

measurement) is proportional to the phase offset, according to:

$$\Delta T = \frac{\Delta\theta}{2\pi f}, \quad \Delta\theta \in]0, 2\pi] \tag{2.2}$$

where ΔT, $\Delta\theta$ and f represents the time delay, the corresponding phase offset between the two replicas and the frequency of operation of the said signal. Despite being substantially easy to implement and offering a very accurate measure, for ranging, this technique is incapable of differentiating phase offsets larger than 360 degrees, due to redundancy issues. This leads this method to either be employed with small carrier frequencies or on very narrowband devices [11].

2.2.3 Time-delay techniques

Time delay techniques are currently, the most reliable method for precise location precision. By performing a time measurement, the position of a device can easily be achieved by multiplying the former by the velocity of propagation of the medium (the speed of light, for example, for RF systems).

2.2.3.1 Time-of-arrival

In time-of-arrival techniques, both the target and measuring unit are assumed to be similar transceiver with synchronized clocks. At reference times, or through time-stamping [2], the latter performs a signal query, which will be used for correlation at the target. The one-way propagation time, as seen in fig. 2.5a, is measured and the distance between the devices assessed. When only

2 devices are used (i.e.: the mobile and target unit), the set of possible solutions lies in a circle centered around the former.

2.2.3.2 Time difference of arrival

In time-difference-of-arrival (TDOA), the relative position of the mobile transmitter is attained by examining the difference in time, at which a *beacon* signal arrives to multiple measuring units. Alternatively, the mobile node can be equipped with two separate antennas (similar to POA) as seen in fig. 2.5b, and the time difference is calculated through the cross-correlation of signals received at both, according to:

$$\hat{R}_{x_i,x_j}(\tau) = \frac{1}{T}\int_0^T x_i(t)x_j(t-\tau)dt, \tag{2.3}$$

where $\hat{R}_{x_i,x_j}(\tau)$, $x_i(t)$ and $x_j(t)$ correspond to the cross-correlation function and the signal received at antenna i and j, respectively. The timing measurement is obtained by finding the global maximum of $\hat{R}_{x_i,x_j}$. Unlike TOA, the set of possible solutions becomes the intersection of hyperbolas.

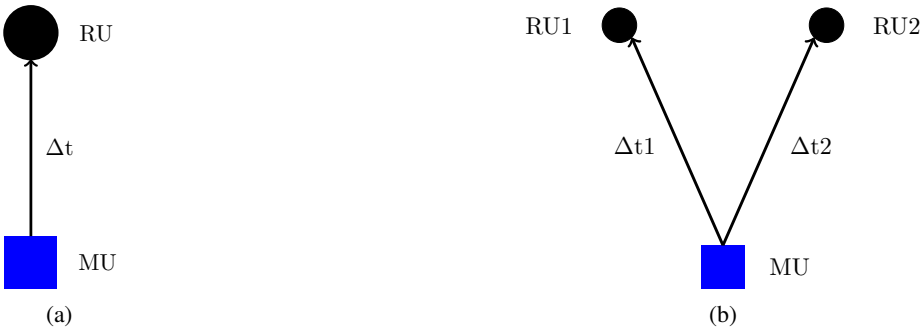


Figure 2.5: Comparison between (a) TOA and (b) TDOA.

2.2.3.3 Round-Trip Time of Flight

Round-trip time of flight (RTOF), like its TOA counterpart, requires two similar transceivers. However, instead of one time measurement, two are performed. After receiving a ranging request signal, from a base station, the RU will respond with a similar modulated signal. Therefore, no device synchronization is required. The distance between the two devices is proportional to the time it takes for the requesting signal to propagate to a second device and to come back, i.e., the time-of-flight (TOF). If we consider that A is the base station and B the remote unit, then t_{AB} is the aforementioned TOF from A to B, and t_{BA} the reversed version. Therefore, the RTOF measurement leads to:

$$t_{TOF} = \frac{t_{AB}+t_{BA}}{2} \tag{2.4}$$

It should be noted, that all time-delay techniques have one common disadvantage. They require very high precise time measurements, to enable proper ranging. As an example, a 10ns time error leads to an error of 3m. Due to this particularity, this technique has been commonly employed in combination with sound or ultrasound modulators, since the lower propagation speed of these waves reduces the ranging error. However, they do not provide enough capability in terms of coverage range and object penetration.

2.3 Position estimation

Until this point, only the techniques required to perform one dimensional measurements have been presented. In the following subsections, an overview of employed base algorithms for two or three dimensional position estimation is provided[3].

2.3.1 Trilateration

Based on the principle of nodes lying in a circle around the measuring unit, comes trilateration, as seen in fig. 2.6a. Usually, the straightforward approach uses geometric methods to compute the

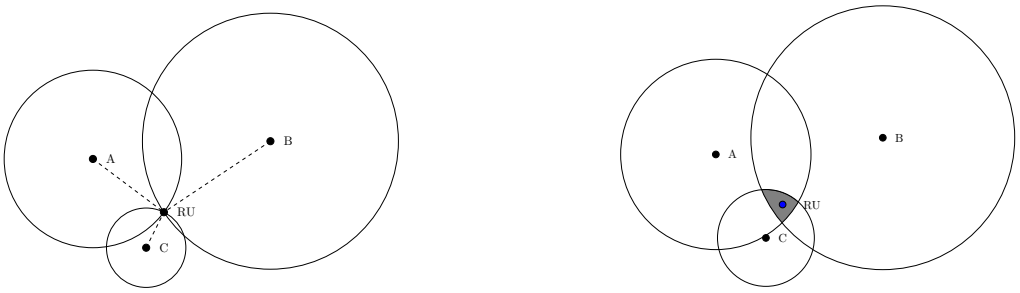


Figure 2.6: Comparison between (a) an ideal situation for trilateration computation and (b) a typical real scenario.

intersection points of the circles. Alternatively, a least-squares algorithm can be employed to minimize the following equation:

$$F(x) \quad = \quad \sum_{i=1}^{N} \alpha_i^2 \; f_i^2(x), \tag{2.5}$$

where α_i is merely a weight given to reflect the reliability of a signal, and $f_i(x)$ corresponds to:

$$f_i(x) \quad = \quad v_p(t_i - t) - \sqrt{(x_i - x)^2 + (y_i - y)^2}, \tag{2.6}$$

where v_p and t_i correspond to the propagation speed of the medium, and the time measurement of unit i, respectively, and x_i and y_i to the coordinates to be estimated.

[3]It should be noted that this implies more than 2 or 3 device measurements.

2.3.2 Multilateration

Similar to trilateration, the multilateration method is employed in TDOA scenarios, where the known ranging metric, as aforementioned, lies along a hyperbola, instead of a circle. Like the former method, it requires at least three devices (but, only two measurements) to perform two-dimensional position estimation, according to:

$$R_{i,j} = \sqrt{(x_i - x)^2 + (y_i - y)^2} - \sqrt{(x_j - x)^2 + (y_j - y)^2}, \tag{2.7}$$

where $R_{i,j}$ would be the function to minimize, in 2.5, in detriment of $f_i(x)$.

2.3.3 Triangulation

In triangulation, the geometric properties of triangles is explored, as can be seen in fig. 2.7. In the general case, two dimensional angulation requires two angle measurements and one distance measurement (for example, the distance between the two reference points, A and B), which could be fixed. If three dimensional positioning is required, the following measurements must be performed: two angle measurements, one distance and one azimuth measurement [12].

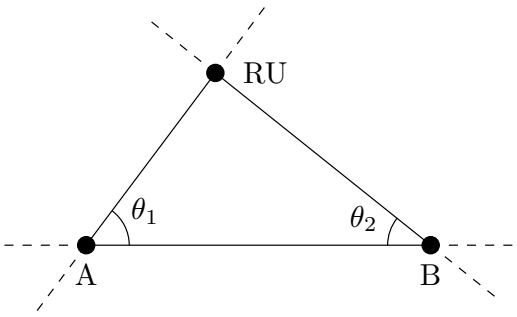


Figure 2.7: Triangulation.

2.4 Overview

A small comparison between the characteristics of the different specified geolocation techniques can be found in table 2.1 [13].

Table 2.1: Geolocation techniques characteristics

	COM [a]	**INF** [b]	**SCA** [c]	**RT** [d]	**ENC** [e]	**FUS** [f]
RSS	small	medium	low	long	low	low
AOA	high	medium	high	long	medium	low
TDOA	high	high	high	long	low	low
TOA	small	medium	low	short	high	high
RTOF	high	low	high	medium	high	high

As can be seen, all the aforementioned techniques have their pros and cons. Despite this, the scientific community appears to be in close agreement, that full-custom or proprietary solutions usually offer the best accuracy results, as can be seen in fig. 2.8 [14]. On the other hand, if ease

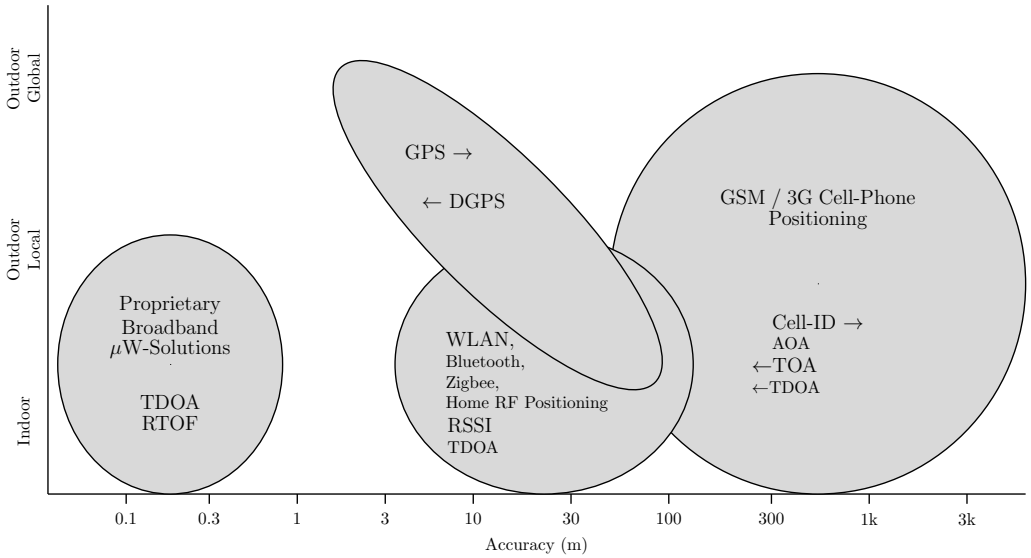


Figure 2.8: Overview of current wireless local-positioning systems.

of use is preferred and lower infrastructure (or product development) costs are of the essence, one might equate to re-use, for example, a *Wi-Fi* or Bluetooth module. In order to properly compare the different techniques (and the corresponding devices that support them), two figures of merit are usually used: the mean error and the root mean squared (RMS) error. Recently, some authors also started to employ the cumulative distribution function, for a set of measures, since it takes into account a more reliable representation of the two aforementioned quantities, under different system and environmental characteristics (such as signal-to-noise ratio - SNR - or existence of a line-of-sight - LOS - component).

2.5 Literature review

During the dissertation, a brief literature review was performed on the most interesting works, developed by scientific community, and summarized in table 2.2. Due to the scope of this work, it should be noted that the emphasis was mainly given to works employing the RTOF technique.

[a] Communication Effort.
[b] Infrastructure Requirements.
[c] Effort for scaling the system.
[d] Response time.
[e] Energy consumption of the mobile unit.
[f] Sensor fusion efficiency.

2.5.1 Commercial solutions

At the authors best knowledge, most available commercial solutions are based on ultra-wideband (UWB) transceivers [15]. Despite their high accuracy, they are incapable of achieving high coverage range, being usually limited to 100-200 meters maximum, in line-of-sight. On the other hand, there is also the possibility of operating these devices in the sub-GHz region, for achieving higher communication distances. However, the low power spectral density (PSD) requirements in Europe, for this class of operation, turns it into an infeasible task.

Table 2.2: Comparison between multiple ranging devices in literature.

	Work	Technology	Technique	Carrier Frequency	Bandwidth	Max Range Reported	Max Mean Error	RMSE
[16]	TIM'15	RFID	PoA and RSS [a]	866.9 MHz	N/A	< 2.5m	< 5cm [b,c]	< 2cm [b,c]
[17]	SENSORCOMM'09	ZigBee	RSS	2.4 GHz	N/A	4.5m	2.5m (@ 2m)	N/A
[18]	TWC'12	CSS	RTOF	2.4 GHz	100 MHz	10m	< 1m	1m
[19]	SIES'12	CSS	RTOF	2.4 GHz	100 MHz	5m	< 4m	50cm
[20]	TIE'09	CSS	RTOF	2.4 GHz	100 MHz	< 3m	< 20cm	N/A
[21]	GeMIC'14	N/A	RTOF	2.44 GHz	60 MHz	200m	< 33cm (@ 90%)	N/A
[22]	RFID'13	FMCW	RTOF	34.5 GHz	500 MHz	10 m	5 cm	N/A
[23]	WPNC'12	FMCW	RTOF	5.8 GHz	150 MHz	19m	< 1.2m (@ 11m)	0.15m
[24]	IVS'11	FMCW	RTOF	5.8 GHz	150 MHz	20 m	N/A	2.5cm
[25]	ICWITS'12	FMCW	RTOF	2.45 GHz	N/A	2m	< 0.2m	< 0.3m
[26]	ICC'14	OFDM	RTOF	5.5 GHz	36.6 MHz	< 30m	< 1m (@ 75%)	< 20cm
[27]	MASS'10	OFDM	RTOF	2.45 GHz	20 MHz	10 m	< 1.17m	< 2.6m
[28]	WPNC'13	3GPP-LTE	RTOF	N/A	30.72 MHz	N/A	N/A	< 10 cm
[29]	MTT'13	UWB	RTOF	9 GHz	6 GHz	55cm	3.5cm	3.5cm
[30]	WISES'06	ZigBee	RTOF	2.4 GHz	N/A	16m	5.5m (@ 4m)	2.6m
[31]	MSN'11	802.15.4a PHY	SDS-TWR	N/A	N/A	40m	< 1m	N/A
[32]	GLOCOM'08	FSK	SDS-TWR	2.4 GHz	2 MHz	N/A	N/A	1m

[a] EIRP of 2W.
[b] Under LOS conditions.
[c] After 2D/3D Mapping.
[d] Employing SDS-TWR technique.

Chapter 3

Proposed Architecture

In this chapter, a novel architecture, based on Orthogonal Frequency Division Multiplexing (OFDM), is proposed for enabling ranging applications in outdoor scenarios. Sub-GHz RF transmission, in TV White Spaces, is explored as an alternative approach, due to its material penetration and propagation characteristics. Key aspects like data throughput, node scalability and, specially, the surrounding environment itself were considered for the design choices.

Due to the characteristics of the Porto vineyards slopes, as stated in chapter 1, a rich multipath environment is expected. As such, this chapter will start with an introductory remark on TV White Spaces, followed by a study of the different channel models, and its effects on wireless data transmission, and an analysis of the requirements of OFDM modulation. Only then will the complete architecture and the ranging algorithm proposal be described.

3.1 TV White Spaces

With the switch from analog TV to Digital Video Broadcasting (DVB) and Digital Terrestrial Television (DTT), selected channels, ranging from 54MHz to 862MHz, became worldwide available. These spectral holes, named *TV White Spaces* (TVWS) or *Interleaved Spectrum*, in the very high frequency (VHF) and ultra high frequency (UHF) range, unused by licensed services, became an unique opportunity for the development (or enhancement) of a niche set of RF communication systems and applications, such as [33]:

- Machine-to-Machine Communications (M2M), which faced problems from current ISM band solutions, due to low data-rates;
- *Super Wi-Fi*, by enabling Internet access in rural or high population density regions;
- In-Home Distribution;
- Video Surveillance, by reducing costs in wired infra-structures;
- Disaster Planning: deployment of communication systems, in case of natural disasters, which can incapacitate public communications.

Due to its lower frequency of operation, they inherently enable much better propagation capabilities than the more common systems deployed in Industrial, Scientific and Medical (ISM) bands. This allows a more reliable operation in rural and metropolitan applications, while being cost-effective and offering better coverage range (as high as dozens of kilometers) [34].

However, operation in TVWS is conditioned, under the ruling that devices do not cause harmful interference to licensed users and incumbents (such as, wireless microphones). Several cognitive radio technologies have been developed, to ensure proper protection of these services. Two of the most common technologies are:

1. Geo-location positioning, combined with access to a database of incumbent systems;
2. Spectrum sensing.

In the former, a TVWS device attempts to calculate its location, through the usage, for e.g., of a GPS receiver. Then, it accesses a TVWS database which contains information related to TV broadcasting stations, like its location and power transmission, in an attempt to compute a set of available channels, on which no harmful interference is caused. In the latter, a system capable of measuring the RF TV channels, must be embedded in the device, for determining which channels are occupied by incumbent services. Both devices are commonly termed as cognitive radios (CR).

3.1.1 Regulatory requirements

Despite the initial transitions to digital broadcasting having begun, mostly, in the early years of the millennium, it took almost until 2010 for most communication regulators to widely accept the idea of license exempting these spectral holes. Multiple works on the availability of these unused spectrum have already emerged, such as [35, 36], for different countries, with some also performing field trials [37].

Unfortunately, at the authors best knowledge, most countries tend to adopt solutions similar to the Federal Communications Commission (FCC), at a much slower rate. As such, in the following subsections, a brief explanation of different rulings will be presented, with main emphasis on the requirements in the United States of America. Nevertheless, the following differences should be noted: while TVWS in North America is composed of 6MHz channels, with a maximum power spectral density limited to the equivalent value of a single channel, the remaining countries adopted a more traditional 8MHz bandwidth, and the capability of multiple channel allocation without PSD restrictions. This can be seen, in fact, as a major drawback for implementation of CR in the former.

3.1.1.1 United States of America

As already explained, the switch-over from analog to digital TV broadcast (usually, termed National and Advanced Television System Committee - NTSC and ATSC, respectively, in North America) led to the availability of unused frequency spectrum. In June 2002, the Spectrum Policy Task Force was established, to assist the FCC in identifying and evaluate changes is spectrum policy that would increase the public benefit, through the usage of radio spectrum. In fact, the

final report [38] states that, in certain districts of the United States of America, many bands below 1GHz were unoccupied or only used part of the time. The work of the task force, led to the development of the FCC Report and Order (R&O) [39], in 2008, and the Second Memorandum Opinion and Order [40], in 2010, which defined the current requirements for operation in TVWS.

Two different classes of devices are defined under the FCC rulings: fixed devices and personal/portable devices (which can operate either in mode I or mode II, according to their geolocation capabilities). Only fixed devices are allowed to operate in VHF, except in channels 3-4. In UHF, personal/portable devices are not allowed to used channels 14-20, since they are used for public land and commercial mobile radio services. Furthermore, fixed devices are not allowed to operate in channels 36 and 38, and channel 37 is reserved for radio astronomy applications. This channel distribution can be found summarized in table 3.1.

Table 3.1: TV White Spaces channel distribution, in the USA.

TV Channel		Frequency (MHz)	TVWS allowed devices
VHF	2	54-60	Fixed
	5-6	76-88	Fixed
	7-13	174-216	Fixed
UHF	14-20	470-512	Fixed
	21-35	512-602	Fixed & personal/portable
	36	602-608	Personal/portable
	38	614-620	Personal/Portable
	39-51	620-698	Fixed & personal/portable

The maximum transmit power for fixed devices is 1W (30dBm), but can also be equipped with an antenna, with a maximum gain of up to 6dBi. For personal/portable devices, the allowed maximum is 100mW, but it is reduced to 40mW if operating adjacently to a TV broadcast channel, with maximum PSD given according to table 3.2 and the required compliance spectral mask can be seen in fig. 3.1. In case the TV band device (TVBD) relies on spectrum sensing, for incumbent protection, it has to be capable of sensing the following signal levels:

- -114dBm, averaged over 6MHz (ASTC signal);
- -114dBm, averaged over 100 kHz (NSTC signal);
- -107dBm, averaged over 200 kHz (wireless microphones).

Table 3.2: TVWS maximum power transmission, in the USA.

Device Class	Maximum Power Output	PSD Limit (100kHz)	Adjacent Channel PSD Limit (100kHz)
Fixed	30 dBm (1W)	12.6 dBm	-42.8 dBm
Personal/Portable[a]	20 dBm (100mW)	-1.4 dBm	-56.8 dBm
Personal/Portable[b]	16 dBm (40mW)	-0.4 dBm	-55.8 dBm
CR with Spectrum Sensing	17 dBm (50mW)	2.6 dBm	-52.8 dBm

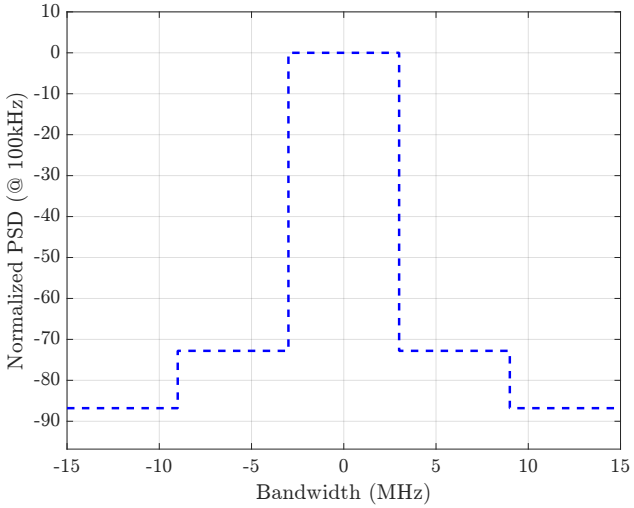


Figure 3.1: TVWS spectral mask, as required by the FCC.

3.1.1.2 Europe

In Europe, the switchover do DVB, partially freed up the spectrum from 470 to 862MHz, in blocks of 8MHz, as aforementioned, in what became known as the *Digital Dividend* (the simultaneous transmission of multiple signals over a shared bandwidth). The European harmonized channel table can be found in table 3.3. Despite the recommendations of the International Telecommunications Union (ITU), during the 2012 World Radiocommunication Conference [41], frequencies ranging from 790-862MHz, corresponding to channels 61-69, were decided to be allocated to fixed and mobile communications [42, 43]. This turn of events made it a matter of each individual country to formulate the requirements to be applied, in terms of channel interference [44].

Table 3.3: European harmonized DTT channels [45].

Channel Number	21	22	23	...	58	59	60
Frequency Range (MHz)	470-478	478-486	486-494	...	766-774	774-782	782-790

In the remaining frequency ranges, the European Conference of Postal and Telecommunications Administrations (CEPT) has allowed the usage of cognitive radio techniques, if subject to not causing interference to incumbent radio services, such as:

- Broadcasting services;
- Program Making and Special Event (PMSE), composed (mainly) of wireless microphones;

[a] Not operating in channel adjacent to a TV Broadcast.
[b] Operating in channel adjacent to a TV Broadcast.

- Radio Astronomy and Services (RAS), which occupies channel 38, from 608 to 614Mhz, in most European countries;
- Aeronautical radionavigation (ARNS);
- Mobile fixed services adjacent to freed bands.

However, despite initial reports indicating three possible approaches for dynamic spectrum access (two of them already mentioned, and the third being the usage of a beacon to transmit information of readily available channels) [46,47], both the Electronics Communication Committee (ECC) and the Office of Communications (OFCOM), in the United Kingdom, at the time of writing, have ruled that TVWS devices should only rely on the access to a geo-location database (DB), or white space database (WSDB), in a Master-Slave configuration. Through this requirement, a master device must acquire its current latitude and longitude, and query a DB for unallocated channels and maximum power available for transmission. It should then relay that information to one or more slave devices, which must comply to the aforementioned metrics.

Access to the DB should be performed on an hourly basis, and transmissions should be stopped, in case of a negative response [48–50]. It should be noted, that they have not completely ruled out the possibility of spectrum allocation through spectrum sensing. The decision was made on the basis that current technologies were incapable of achieving the required sensing thresholds, below -120dBm, after OFCOM performed field trials [51], in the United Kingdom[1]. In fact, coexistence scenarios have already been under discussion for more than five years, at the time of writing, by the scientific community [8, 53, 54]. In terms of adjacent channel leakage, similar ruling as the FCC apply, i.e., a 55dB margin for adjacent channel operation.

3.2 Standards and Regulations

With the increasing need for new communication systems, new standards have also emerged, in an attempt to satisfy public and industrial demands. In the following sub-sections, a brief overview of such standards, developed by the Institute of Electrical and Electronics Engineers (IEEE) will be explained.

3.2.1 IEEE 802.11af/ah

After the development of requirements for TVWS operation, the IEEE 802.11af standard was released, as the fifth amendment to the original 802.11 standard, for medium access control (MAC) and PHY specifications, in wireless local area networks (LAN) [55]. This standard also exploited the propagation characteristics of sub-GHz operation (and inherent possible higher multipath propagation), by changing the normalized Wi-Fi parameters for OFDM. However, most

[1] This is actually a current matter of study. Most European countries, still do not processes any WSDB, and, the usage of the database alone, does not guarantee incumbent protection for PMSE devices, as an example. On the other hand, there are already works which were capable of achieving the required thresholds, by merely using using software-defined radios at the physical layer [52], instead of full-custom solutions.

works developed, found in literature, referencing it, rely on the conventional wireless definitions, by employing direct down-conversion of 2.4GHz signals, from commercially available modems, to sub-GHz frequency bands.

On the other hand, in 2017, the *ah* revision, was published, which attempts to offer higher coverage ranges, by exploiting the re-usage of the lower ISM bands, of each region. The main accepted purpose, for this specification, is to enable rural and high-density urban areas open-access to the Internet.

3.2.2 IEEE 802.15

Among existing standards, IEEE 802.15 has always been the most interesting for the deployment of IoT and M2M communication systems. Developed with the purpose of enabling wireless personal area networks (WPAN), the original standard and its successive amendments, have successively tried to achieve higher throughputs and scalability, in the typical ISM frequency bands. It is based on these set of standards, that some of the most known technologies, in wireless sensor networks (WSN), such as *Zig-Bee* and UWB were developed. Among the known amendments, the following are emphasized:

- 802.15.3: Added additional PHY and MAC specifications for high data-rate WPANs, enabling wireless connectivity at low-cost in multimedia-capable portable consumer electronic devices [56];
- 802.15.4: This amendment, added to its scope, specifications for UWB operation, including in sub-GHz frequencies, enabling the possibility of ranging application, in particular with the Two-Way-Ranging (TWR) algorithm, based on RTOF [57];
- 802.15.4m: This particular amendment, added additional PHY and MAC layer specifications, to enable communication of low-rate devices in TVWS.

3.3 Channel models

Two of the most typical impairments to RF systems are the frequency selective nature of channels, due to the presence of scatterers and reflections, and time dispersion (which leads to a frequency shift in the received carrier frequency), due to the mobility of nodes. In this section, an overview of both effects will be presented, due to its importance for characterizing OFDM systems.

3.3.1 Multipath propagation

Multipath propagation is the result of propagation in a channel where reflectors lead to the creation of multiple versions of a transmitter signal, each received with different amplitude, phase and even angle of arrival [58, 59]. Depending upon the phase, these multiple copies may result

in either an increase or decrease input power, at the receiver. This uncontrollable nature of RF transmission, led to the development of different channel models, in order to aid researchers and designers in predicting the different circumstances under which communication is still feasible.

3.3.2 Tapped delay line models

In a real scenario, multipath propagation channels have a continuous impulse and frequency response. However, due to the sampling nature of baseband RF signals, its analysis is simplified by assuming it as a finite set of discrete impulses (i.e.: as a discrete channel impulse response - CIR),

$$h(t,\tau) = \sum_{i=1}^{N} c_i(t)\delta(t-\tau_i) \tag{3.1}$$

where $c_i(t)$ and $\delta(t)$ represent a complex coefficient (i.e., a phase shift and magnitude deviation) and the Dirac Delta function, respectively, and τ_i represents the time at which a signal reflection occurs, i.e., a multipath component (MPC). In fact, any continuous time impulse response can be approximated by the summation of delta functions, given they are sufficiently close, according to the bandwidth of the signal of interest [60].

3.3.2.1 Exponential decay

Until recently, the typical approach for channel modeling was the assumption of an exponential decay model, which, like the name implies, assumes that the magnitude of coefficient $c_i(t)$, in 3.1, has an exponential decay, as can be seen in fig. 3.2b, according to

$$h(t,\tau) = \sum_{i=1}^{N} \beta_i e^{(j\phi_k)}\delta(t-\tau_i) \tag{3.2}$$

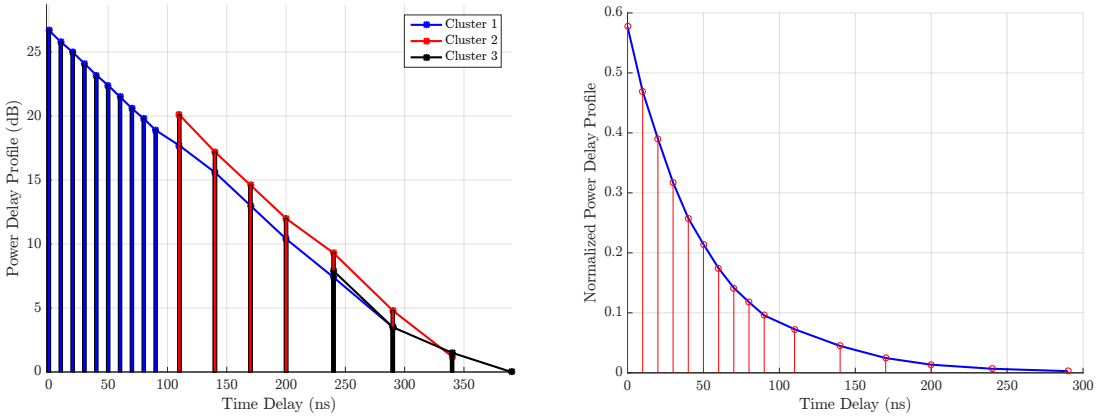
where β_i and $e^{j\phi_k}$ represent the exponential magnitude decay and arbitrary phase shift, in replacement of coefficient $c_i(t)$.

3.3.2.2 Saleh-Valenzuela

Similar to the previously explained model, Saleh-Valenzuela (named after the empirical studies of its authors [61]) includes four more assumptions:

- Reflections occur in clusters, poisson distributed according to parameter Λ;
- Clusters have an exponential decay time Γ;
- Each cluster can generate additional MPCs, according to a poisson process of parameter λ;
- MPCs, due to a given cluster, have an exponential decay time γ.

An example channel, which follows the Saleh-Valenzuela model, can be found in fig. 3.2a [62].



(a) (b)

Figure 3.2: 802.11 channel model D, (a) with 3 visible clusters and (b) overview of the first cluster.

3.3.3 Power delay profile

Despite initially assuming a complex CIR, usual models are provided in what is known as the power delay profile (PDP), according to:

$$P(t,\tau) \equiv |h(t,\tau)|^2. \tag{3.3}$$

Instead of defining a complex characteristic, models are provided as losses, in decibels, per MPC as can be seen, as example, in table 3.4. The complex representation of the channel (or, more

Table 3.4: ITU Recommendations for vehicular test environments, channel A [63].

Tap	**Relative Delay (ns)**	**Average Power Decay(dB)**
1	0	0.0
2	310	-1.0
3	710	-9.0
4	1090	-10.0
5	1730	-15.0
6	2510	-20.0

correctly, the phase shift) is assumed to be random, according to three possible distributions:

- Rician: assumes the first MPC has a much higher magnitude than all the others, making it useful to model multipath propagation when a LOS component is available;

- Rayleigh: a special case of the Rician distribution, assumes that the magnitude and phase of the MPCs are uncorrelated and normal distributed. It is usually used to model NLOS scenarios;

- Nakagami-m: a purely empirical model, not based on any results derived from physical considerations. It is usually used because, by changing its parameter m, we can obtain the aforementioned distributions, in a closed form [64].

3.3.3.1 Delay spread metrics

From the power delay profile model, three metrics of interest can be derived:

1. Mean delay spread, $\bar{\tau}$;
2. Max delay spread, τ_{Max};
3. Root mean squared delay spread, τ_{RMS};

From the three, usually the latter is the one of interest for the design of RF systems, and it is expressed according to:

$$\tau_{RMS} = \sqrt{\frac{\int_0^\infty (t-\bar{\tau})^2 \, P(t,\tau) \, dt}{\int_0^\infty P(t,\tau) \, dt}} \tag{3.4}$$

or, in its discrete form:

$$\tau_{RMS} = \sqrt{\frac{\sum_{i=0}^\infty (\tau_i-\bar{\tau})^2 \, P[\tau_i]}{\sum_{i=0}^\infty P[\tau_i]}} \tag{3.5}$$

Usually a channel is considered flat (or frequency non-selective) if the symbol bandwidth (BW_{sym}) is such that:

$$BW_{sym} \geq \frac{1}{5\tau_{RMS}}, \tag{3.6}$$

in other words, the symbol duration is at least 5 timers longer than the resulting accumulation of reflections, where the value $1/\tau_{RMS}$ is usually named the coherence bandwidth. An example channel frequency response, can be seen in fig. 3.3, where we can find large signal variations along its spectrum.

3.3.3.2 Doppler spread

Motion of intermediate objects (or between a transmitter and receiver RF device), leads to frequency shifts of the carried signal. This frequency shifts are assumed bounded, for free space propagation, according to:

$$f_{d,MAX} = \pm \frac{v \, f_c}{c}, \tag{3.7}$$

where f_d, v, f_c and c correspond to the Doppler frequency shift, the relative movement speed between two objects, the carrier frequency of the RF signal and the speed of light, respectively.

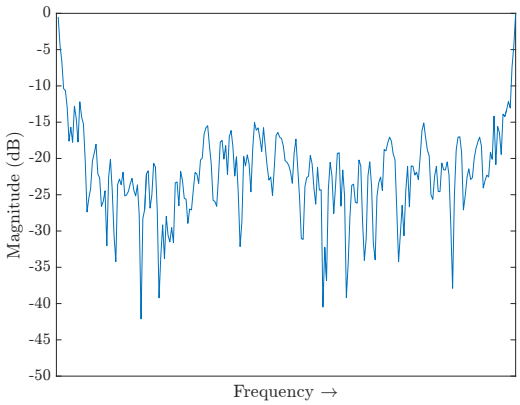


Figure 3.3: Example multipath channel frequency response.

This frequency shift, may lead to a time-variant channel frequency response, if the total symbol duration is larger than, what is known as, the coherence time, usually given by:

$$T_C = \frac{1}{f_{d,MAX}}. \tag{3.8}$$

3.3.4 Overview

The above presented characteristics, for channel propagation, lead to the formulation of four sets of channels:

- Flat fading, if a channel is assumed to have a constant frequency response, for a bandwidth greater than the transmitter signal bandwidth;
- Frequency selective, if the signal bandwidth is greater than the coherence bandwidth, leading to fading of frequency components, at different extents;
- Fast fading, if the signal duration is larger than the coherence time, leading to a changing CIR, along time;
- Slow fading, if the CIR changes at a rate much slower than the transmitted baseband signal.

3.4 Orthogonal Frequency Division Multiplexing

With the evolution of cellular networks, a high requirement for high spectral efficient RF communication emerged. Orthogonal Frequency Division Multiplexing appeared as one of those techniques, enabling multiple channel allocation, usage of high order constellations, such as Quadrature Amplitude Modulation (QAM) and avoiding multipath propagation. It is currently one of

the most used technologies for RF transmission, found in Wi-Fi, Long-Term-Evolution and DVB systems and, until recently, was still a strong candidate for 5G applications[2].

Multi-carrier systems can be dated back more than one century, when telegraph subscription lines carried more than one signal, at low data rates, using a separate carrier frequency each [67]. Carrier frequencies were spaced sufficiently far apart, so that the signal spectrum would not overlap. Empty spectral regions assured that signals could be differentiated with realizable filters. It was only during the 1960s that the concept of adapting parallel data transmission and frequency division multiplexing (FDM) was published.

In the classical view of FDM, as described, N non-overlapping frequency channels are required to transmit the total signal bandwidth, where each sub-channel is modulated with a different symbol. As can easily be seen in fig. 3.4a, this leads to very inefficient use of the available spectrum [68]. By combining sub-channels, each with a signal rate of b and also spaced apart b in frequency, high data-rate can be achieved, using overlapping channels, capable of avoiding high-speed equalization requirements, impulsive noise and multipath distortion, while fully using all of the available bandwidth, as shown in fig. 3.4b. In OFDM, the signal sub-carriers are rearranged, such that the sidelobes of the individual subcarriers[3] overlap, but each individual signal is still received without adjacent interference. This leads to a mathematical orthogonality criterion, from which the name of the modulation scheme is derived.

3.4.1 Principle of operation

The basic principle, as aforementioned, is to split a high-rate datastream into N lower rate streams, that are transmitted simultaneously. Because the overall symbol duration increases with the decrease in frequency rate, the dispersion in time, due to multipath propagation, is decreased, when compared to single-carrier systems. Even when under frequency selectivity, OFDM outperforms the latter in terms of channel equalization, since it can be implemented as a single-tap per frequency bin.

Taking all parameters of the OFDM scheme, a symbol starting at time $t = t_s$, can be denoted as

$$s(t) = \begin{cases} \mathfrak{Re}\left\{ \sum_{i=\frac{-N_s}{2}}^{\frac{N_s}{2}-1} d_{i+Ns/2}\, \mathrm{e}^{j2\pi(fc-\frac{i+0.5}{T})(t-ts)} \right\}, & t_s \le t \le t_s + T \\ 0, \quad t < t_s \wedge t > t_s + T \end{cases} \tag{3.9}$$

or, alternatively, in its equivalent complex baseband notation,

$$s(t) = \begin{cases} \sum_{i=\frac{-N_s}{2}}^{\frac{N_s}{2}-1} d_{i+Ns/2}\, \mathrm{e}^{j2\pi(\frac{i}{T})(t-ts)}, & t_s \le t \le t_s + T \\ 0, \quad t < t_s \wedge t > t_s + T \end{cases}, \tag{3.10}$$

[2] At the time of writing, Filter-Bank Multi-Carrier (FBMC) was already under testing, mainly due to its superior adjacent channel leakage ratio (ACLR), for base station operation. For more information, in FBMC, we recommend reading [65, 66].

[3] Due to the nature of frequency division multiplexing modulation schemes, the terms subcarrier and subchannel will be used inter-changeably, without loss of generality.

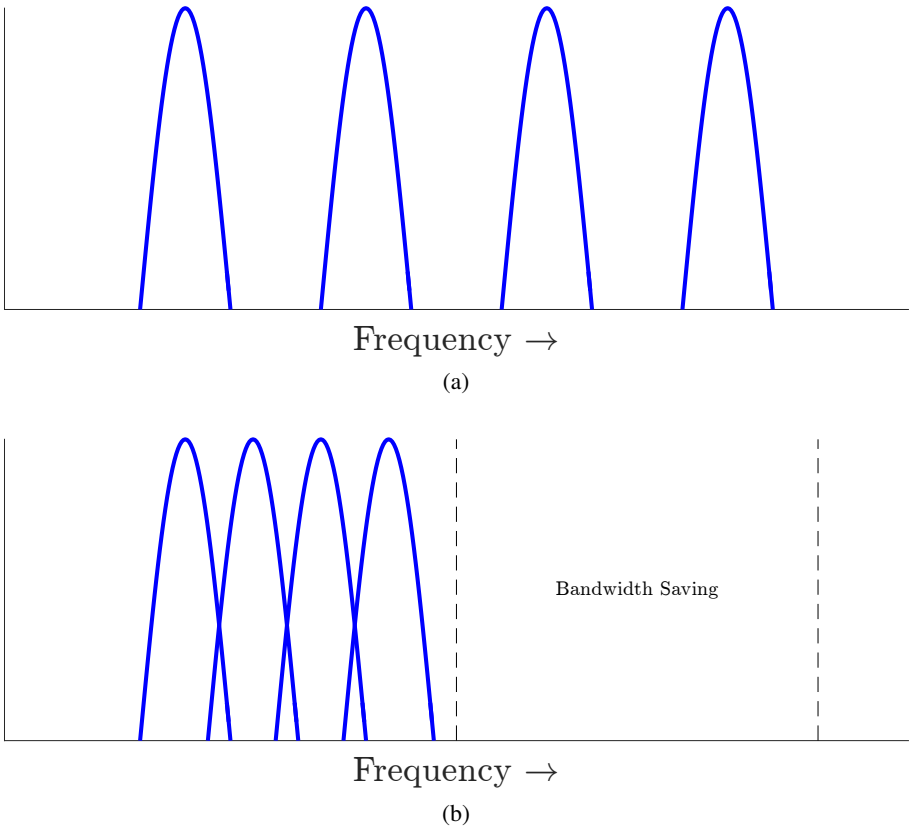


Figure 3.4: Comparison between (a) classical multi-carrier FDM systems and (b) OFDM spectral usage.

where N_s, T, d_i, and fc represent the number of subcarriers allocated, the symbol duration, the complex modulated data symbols and the RF carrier frequency, respectively. In the latter representation, the real and imaginary components correspond to the in-phase and quadrature parts of the OFDM signal, which should be upconverted using a quadrature modulator. An example functional block diagram, of the OFDM modulation, can be seen in fig. 3.5. As an example OFDM modulation, we can see in fig. 3.6a and fig. 3.6b four subcarriers in frequency and time domain, respectively. The input modulated data is assumed to be a square pulse of length T, hence the equal amplitude and phase. It should be further noticed that each subcarrier has exactly an integer number of periods in the time interval and the number of cycles between each adjacent subchannel differs by exactly one, like aforementioned, for the orthogonality criterion. If, for instance, we attempt to demodulate the $j-th$ sub-carrier from 3.10 by downconverting the signal and then integrating it over a period of T seconds, the result is as follows:

$$\int_{t_s}^{t_s+T} \mathrm{e}^{(-j2\pi\frac{j}{T}(t-t_s))} \sum_{i=-\frac{N_s}{2}}^{\frac{N_s}{2}-1} d_{i+(Ns/2)} \mathrm{e}^{(j2\pi\frac{i}{T}(t-t_s))}\, dt \tag{3.11}$$

$$= \sum_{i=-\frac{N_s}{2}}^{\frac{N_s}{2}-1} d_{i+(Ns/2)} \mathrm{e}^{(j2\pi\frac{i-j}{T}(t-t_s))}\, dt = d_{j+(N_s/2)} T \tag{3.12}$$

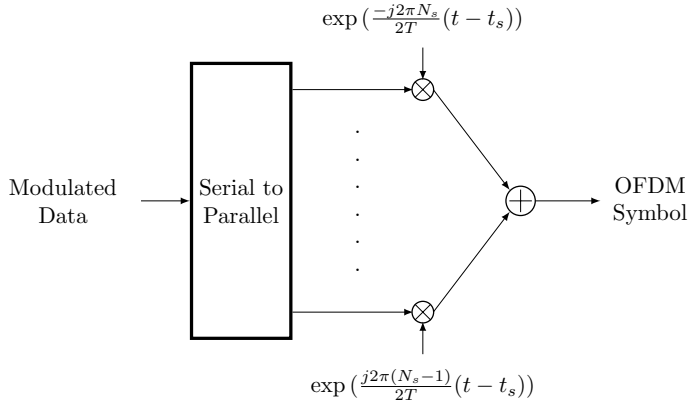


Figure 3.5: Simplified OFDM Modulator

With this analysis, it is easy to see that for values $i \neq j$, the integration always results in zero, due to the difference $i-j$ always producing an integer number of cycles, within the integration interval T. On the other hand, for $i = j$, the value to be integrated becomes $e^0 = 1$, leading to the expected result, the $j-th$ modulated data symbol, with a duration of T.

Despite this analysis, a different approach can be demonstrated, through analysis of fig. 3.6a. According to 3.10, each OFDM symbol contains subcarriers that are non-zero, over a T-second interval. Hence, the spectrum of a single symbol can be seen as the convolution of a group of Dirac pulses, each located at the subcarrier frequencies, with the spectrum of a square pulse of duration T. The spectral representation of a square pulse of duration T is $sinc(\pi fT)$, which is zeros for all frequencies that are an integer multiple of $1/T$. At the maximum value of each subcarrier, all others are zero. Since the OFDM receiver calculates the spectrum values at those particular values (with the integration from t_s to t_s+T), it can effectively demodulate each subcarrier, free from interference from any other. This can also be seen as the Nyquist criterium, applied in the frequency domain, for an intersymbol interference free pulse shape, i.e: intercarrier intereference (ICI) free.

The complex baseband notation, for the OFDM symbol, represented in 3.10, can be seen as the inverse Fourier transform (IFT) of N_s complex modulated symbols. In fact, all modulator implementations, use its discrete equivalent, the inverse discrete Fourier transform (IDFT), where time t is replaced by sample number n and the symbol duration T by the number of samples, N_s, according to:

$$s[n] = \sum_{i=0}^{N_s-1} d_i e^{(j2\pi \frac{in}{N_s})}. \tag{3.13}$$

In other words, the OFDM demodulation and modulation, basic procedure, can be performed through the usage of the discrete Fourier transform (DFT) and its inverse, respectively. As will be seen in 4.1, the DFT, despite its apparent complexity, has specific algorithms to decompose it into rather simple algorithms.

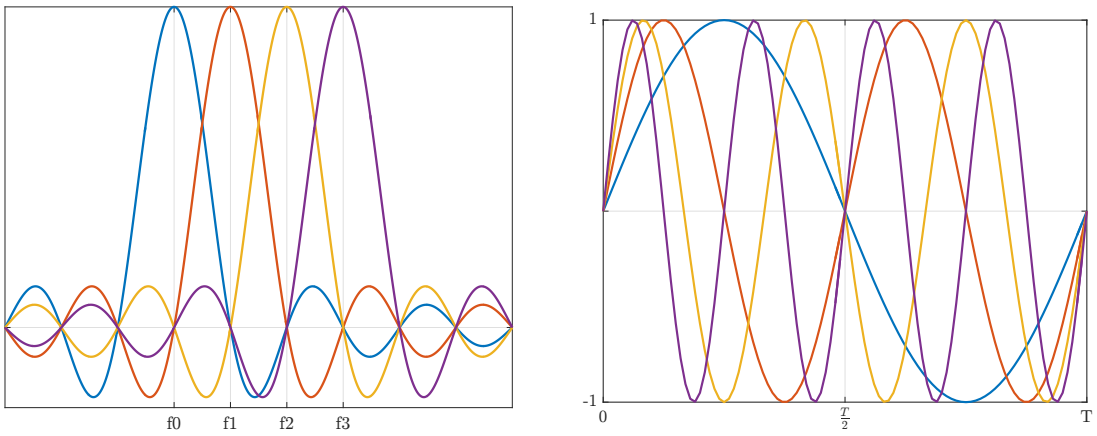


Figure 3.6: Example orthogonal subcarriers in (a) frequency and (b) time domain.

3.4.2 Cyclic extension

As already explained, one of the most important features of OFDM, is its capability to deal with multipath propagation. By dividing an input datastream in N_S subcarriers, the symbol duration is, effectively reduced N_s times, which minimizes the relative multipath delay spread each individual carrier is subject to. However, this is not sufficient to guarantee inter-symbol interference (ISI) free communications. Due to the nature of multipath propagation, one OFDM symbol can still overlap with the next, incoming, symbol. To eliminate ISI almost completely, a guard time is added to the beginning of each symbol, The duration of this guard time should be chosen larger than the expected delay spread. In fact, the usual metric, as already explained, is the RMS delay spread.

The typical approach, for the guard time, is to cyclically extend the last N_{CP} samples of the OFDM symbol, from 3.13, as can seen in fig. 3.7, to reduce ICI due to cross-talk, between delayed subcarriers. This way, a transmission affected by multipath, will always see the symbol as the sum of sine waves, but of different phase shifts.

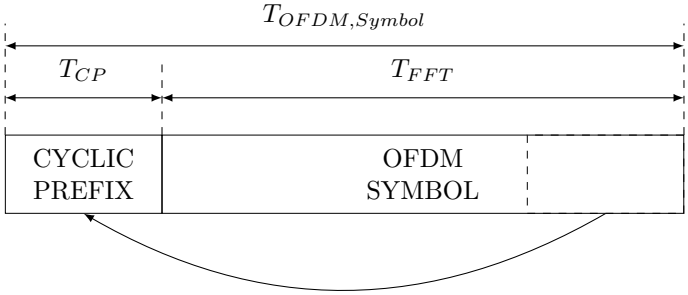


Figure 3.7: Cyclic prefix addition to OFDM symbol.

3.4.3 Parameter selection

The choice of parameters for OFDM modulation, is usually a trade-off between conflicting requirements. According to [68], there are usually three main requirements to start with:

1. Available bandwidth;
2. Bit-rate;
3. Delay spread.

Since bit-rate is usually dependent on the symbol modulation and coding rate employed, it will be ignored for the following analysis. As a rule of thumb, the cyclic extension should be 2 to 4 times the RMS delay spread, depending on the sensitivity of the system to ICI and ISI. After setting the guard interval (GI), the actual symbol duration can be set. In order to minimize the SNR loss (due to the cyclic extension), the actual symbol duration should be much larger than its prefix. However, it must not be chosen arbitrarily, since a larger symbol leads to:

1. Higher number of subcarriers, with smaller spacing between them (and, consequently, an increase in the peak-to-average-power ratio - PAPR);
2. Higher complexity, due to the required IFFT size to accommodate the aforementioned point;
3. More sensitivity to phase noise and frequency offsets.

A typical design choice is to make the symbol duration at least five times larger than the GI, which only implies a 1-dB SNR loss. After fixing the symbol and guard time duration, the subcarrier spacing (Δf) can be calculated as

$$\Delta f = \frac{1}{T_{OFDM} - T_{CP}} \tag{3.14}$$

and the corresponding number of subchannels,

$$\#Subcarriers = \frac{BW_{available}}{\Delta f}, \tag{3.15}$$

where T_{OFDM}, T_{CP}, and $BW_{available}$ corresponds to the complete symbol and cyclic prefix duration, and the available spectrum -3dB bandwidth, respectively. The size of the required DFT, can then be found as the nearest number, power of 2, in respect to the number of subcarriers. In case the size of the required DFT exceeds the number of allowed subcarriers, zero padding is performed on the input signal, to ensure no signal spectral leakage occurs in adjacent channels. This process of zero padding is, usually, referred as guard band insertion.

Taking as an example, the IEEE 802.11 specification, found in table 3.5 [69], we can find that all parameters follow this basic principle, where the only unknown variable is the cyclic prefix. In the worst case scenario (for a 20MHz channel bandwidth), the GI is $0.8\mu s$. Through a

Table 3.5: IEEE 802.11 OFDM specification for different channel bandwidths.

	20 MHz Channel Bandwidth	**10 MHz Channel Bandwidth**	**5 MHz Channel Bandwidth**
FFT Size	64	64	64
Data Subcarriers	52	52	52
Guard band Subcarriers	12	12	12
Subcarrier Spacing (kHz)	312.5	156.25	78.125
IFFT Period (μs)	3.2	6.4	12.8
Cyclic Extension (μs)/samples	0.8/16	1.6/16	3.2/16
OFDM Symbol (μs)	4	8	16

quick analysis of [62], we can find that the largest expected RMS delay spread, for IEEE 802.11 channels, is actually $350ns$, so it is inside the described specifications, as expected.

The last step, is to find the required sampling frequency for the output samples:

$$F_{sampling} = \frac{N_{FFT} + N_{CP}}{T_{OFDM}}, \tag{3.16}$$

where N_{FFT}, N_CP and T_{OFDM} correspond to the required size of the FFT, the number of samples of the cyclic prefix, and the complete OFDM symbol duration, respectively.

3.4.4 Digital signal processing chain

Through the previous sub-sections, the basic implementation requirements for OFDM were described, through the implementation of the IFFT and cyclic extension. In fig. 3.8a and fig. 3.8b, we can find the typical signal chain for transmission (Tx) and reception (Rx), respectively.

We can see, as aforementioned, that both the FFT and IFFT blocks are present in both diagrams. If no full-duplex operation is required (which is the case, for RF transmission), the same FFT block can be used for the upstream and downstream link, since the IFFT can be computed by conjugating the input and output samples of the former, and dividing the output by its size, as can be demonstrated:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, \mathrm{e}^{-\frac{j2\pi nk}{N}} \tag{3.17}$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]\, \mathrm{e}^{\frac{j2\pi nk}{N}} \tag{3.18}$$

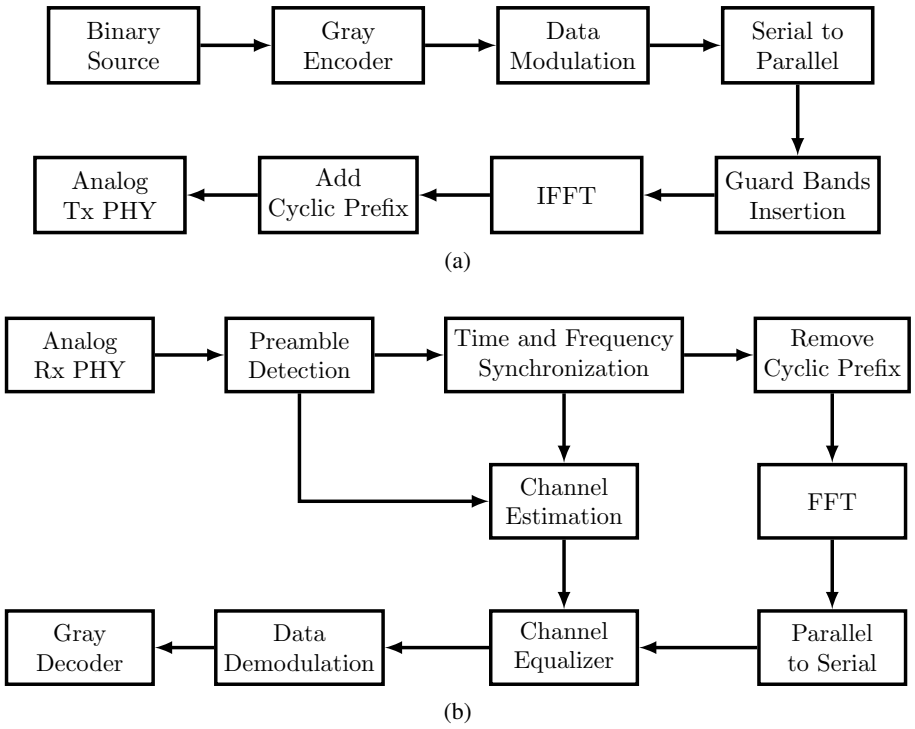


Figure 3.8: Typical OFDM (a) modulator and (b) demodulator block diagram.

$$\begin{aligned}(x[n])^* &= \frac{1}{N}\sum_{k=0}^{N-1}\left(X[k]\,\mathrm{e}^{\frac{j2\pi nk}{N}}\right)^* \\ (x[n])^* &= \frac{1}{N}\sum_{k=0}^{N-1}(X[k])^*\,\mathrm{e}^{-\frac{j2\pi nk}{N}},\end{aligned} \tag{3.19}$$

where $()^*$ denotes the conjugate operator.

3.5 Architecture decisions

Taking into account the former sections, a novel architecture is proposed, capable of employing OFDM at sub-GHz radio frequencies, to both mitigate multipath propagation[4], while achieving high data-throughput and coverage range[5]. For the ranging metric acquirement, we propose the usage of the symmetrical-double-sided two-way ranging (SDS-TWR) algorithm, based in RTOF. By embedding, in the physical layer, a high-resolution timer[6] with a proper preamble detection mechanism [71] (for performing time-stamping operations), high timing accuracy can be attained[7], in a manner similar to TOA algorithms (which are usually based in signal correlation). According to the Cramer-Rao Lower Bound (CRLB) for an unbiased estimator, the minimum variance of a

[4]Doppler Spread was not taken into account, at the moment, due to the low mobility of the vineyard machinery.

[5]Assuming only free-space path loss, a maximum transmit power of 40mW and a typical receiver sensitivity of -120dBm, with 0dBi antennas, more than 40km could be achieved.

[6]In fact, this idea has already been patented, for usage in IEEE 802.15.4 [70].

[7]For example, the 802.11 data link layer uses an embedded clock, with a period of 1μs, which can lead to a precision error of 300m.

TOA measure is [72]:

$$\sigma^2_{TOA} \geq \frac{1}{8\,\pi^2\,SNR\,BW^2}, \tag{3.20}$$

where σ^2, SNR and BW represent the variance of the estimator, the mean signal-to-noise ratio and the used bandwidth, respectively. From the CRLB, we can see why UWB devices are widely accepted as high-accuracy ranging devices, since the variance of the estimator is inversely proportional to the SNR, but decreases by a squared factor with the used bandwidth.

To further reduce the variance, of the ranging metric, multiple measurements can be performed, simultaneously, between two similar devices, in what is known as the Burst-Mode SDS-TWR ranging algorithm [30–32]. A functional block diagram of the proposed architecture can be seen in fig. 3.9a, separated in its individual layers, as according to the simplified ISO-OSI 5 layer model (found in fig. 3.9b). From this work point-of-view, the proposed architecture is only

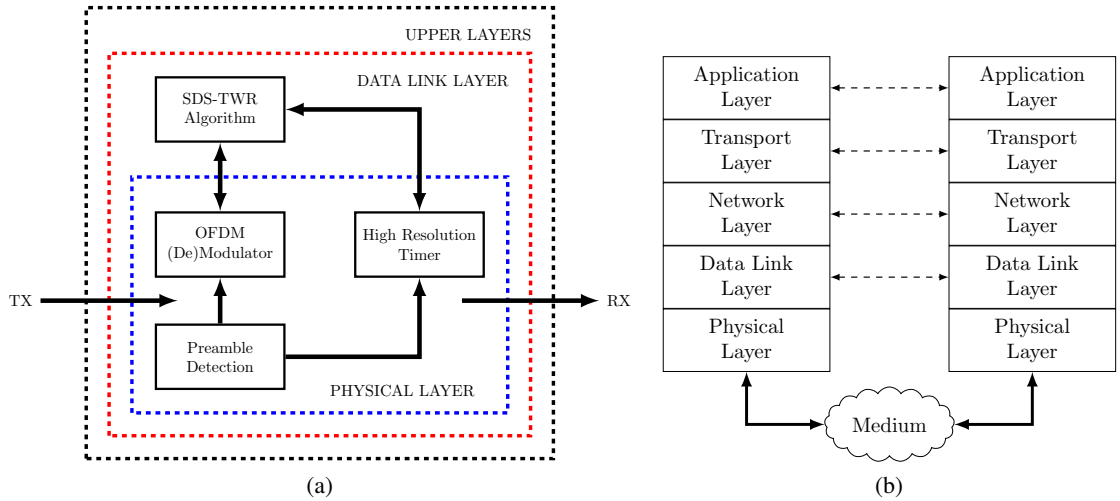


Figure 3.9: Proposed Architecture (a) functional diagram, according to (b) the ISO-OSI model.

concerned in performing relative distance measurements. Capability of performing 2-dimensional or 3-dimensional measurements, is not inside the scope of this document, although it can be required and performed either at the data-link or application layer. Given this, as can be seen, the main changes only affect the physical and data-link layer[8]. Another work is being performed, in the same context of this document, for the development of a quadrature modulation transceiver chain, which will lead to the implementation and integration of the analog and digital physical layers, according to fig. 3.10. Spectrum sensing or geo-location database techniques are not being considered, at the moment, due to the highly questionable worldwide regulatory requirements.

Further enhancements, like the time-reversal technique [73, 74] or usage of super resolution

[8] In fact, by moving, the SDS-TWR algorithm closer to the physical layer, the only requirements needed for the latter are the medium access control mechanisms, which at the time of writing are not being considered.

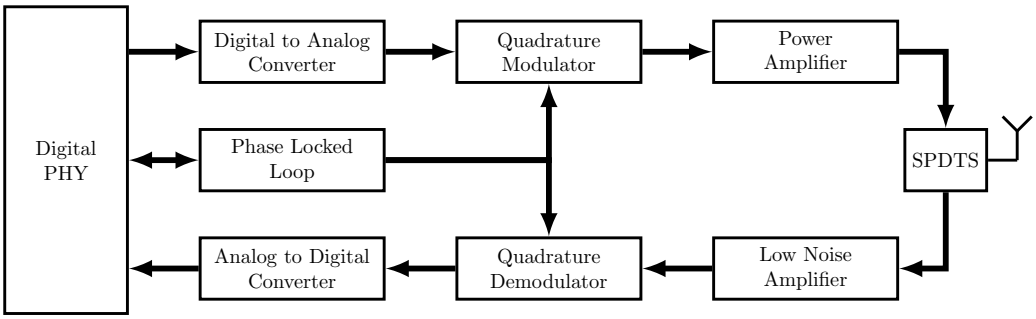


Figure 3.10: Interconnection of the digital and analog physical layers.

algorithms (SRA)[9] can be used to increase the accuracy of the ranging metric, if operating under NLOS conditions. Furthermore, since the proposed architecture is based in OFDM, multiple-input multiple-output (MIMO) techniques can be used to either increase data throughput, or the location precision, due to beam-forming at the antenna or of the RF sinal.

3.5.1 OFDM parameter selection

From reviewing a select subset of available channel models, in literature, the expected RMS delay spread that the system must be capable of handling is around 1.2μs. This value was particularly chosen, according to the typical hilly environments channel models [75], which is expected to offer a similar CIR to the steep slopes of the Douro vineyards. Given this knowledge and since the typical available bandwidth in a TVWS channel, in a worldwide view, is of 6MHz, the following parameters have been initially selected, according to the description in 3.4.3 (although, with a cyclic prefix 5 times larger than the RMS delay spread, to accommodate a possibly larger channel spread):

- Guard interval: 6 μs;
- OFDM symbol duration: 30 μs;
- OFDM symbol of interest duration: 24 μs;
- Subcarrier spacing: 41.6667 kHz;
- # Data subcarriers: 144;
- FFT size: 256;
- # Guard band subcarriers: 112;
- Sampling frequency: 10.6667 MHz;

[9] Typical SRA used for ranging applications in NLOS scenarios include the Matrix Pencil, ESPRIT, Root-MUSIC and Extended Kalmann Filter.

Although not mentioned (until this point), as can be seen, there is also a trade-off between the guard interval duration and the system sampling frequency. Since the architecture is expected to employ digital technology (either application-specific integrated circuit - ASIC - or FPGA[10]), a low uncertainty clock-rate is of the essence, which can not be achieved with the aforementioned values. To correct this, the guard interval was increased to 8 μs, leading to the following parameters:

- OFDM symbol duration: 40 μs;
- OFDM symbol of interest duration: 32 μs;
- Subcarrier spacing: 31.250 kHz;
- # Data subcarriers: 192[11];
- FFT size: 256;
- # Guard band subcarriers: 64;
- Sampling frequency: 8 MHz;

Given a proper selection of the OFDM modulator, we find the expected data-rates, for different coding rates, in table 3.6.

Table 3.6: Achievable Data Rates with the proposed architecture.

Data Modulation	**Coding Rate**	**Data Throughput**
BPSK	1/2	2.4 Mbps
QPSK	1/2	4.8 Mbps
8PSK	1/2	7.2 Mbps
16-QAM	1/2	9.2 Mbps
16-QAM	3/4	14.4 Mbps

3.5.2 Symmetrical double-sided two-way ranging

Similar to RTOF, symmetrical double-sided two-way ranging (SDS-TWR) algorithm adds an extra frame in the communication link (frame 4 in fig. 3.11), to enable both remote units to perform a ranging measurement. The first frame is usually optional, if the device is only used for positioning purposes. In this work's specific context it becomes mandatory, since it will act as an enabling signal for the timing modules, while disabling any possible data transmission, prioritizing the location metric.

[10] In the case of this document, the latter.

[11] It should be noted, that this represents the absolute maximum value for a 6MHz bandwidth. The number of used sub-channels can be further increased if operating, for example, in Europe, or decreased, if the output spectrum does not comply with the required PSD mask.

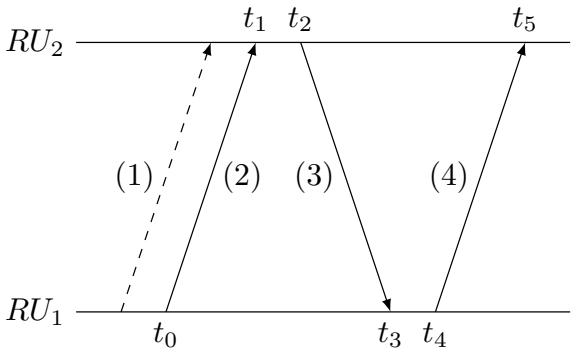


Figure 3.11: Symmetrical double-sided two-way ranging communication link.

From fig. 3.11, the analysis becomes trivial that RTOF, for both devices, can be estimated from:

$$RTOF_{,RU_1} = \frac{(t_3 - t_2) + (t_1 - t_0)}{2} \tag{3.21}$$

$$RTOF_{,RU_2} = \frac{(t_5 - t_4) + (t_3 - t_2)}{2}, \tag{3.22}$$

where t_i, represent time-stamping operations performed at each device. The time differences $(t_2 - t_1)$ and $(t_4 - t_3)$ are due to the required processing overhead, at each device.

3.6 Summary

In this chapter, a novel architecture was proposed, capable of employing OFDM in TVWS, to mitigate multipath propagation, while achieving high data-throughput. An analysis was performed, on the requirements of the OFDM modulation DSP chain. Furthermore, SDS-TWR was defined as the ranging algorithm to be employed, in the architecture. Enhancements, like moving a high-resolution timer to PHY layer, were demonstrated, due to their advantages for high ranging accuracy.

Chapter 4

Hardware Implementation

In this chapter, the required algorithms for the implementation of a fixed-point FFT will be presented. Since the transform requires a set of multiplications, which can be seen as a rotation, along the Cartesian plane, the usage of the COrdinate DIgital Computer (CORDIC) will be explored, due to its rotation functionality.

After the complexity of the procedures is analyzed, the developed HDL modules will be presented, in detail, with all of the design choices taken into account.

4.1 Discrete Fourier Transform

The discrete Fourier transform (DFT) is among one of the most widely used functions in DSP algorithms and applications. It consists of an alternative representation of the discrete Fourier series (DFS) of an aperiodic discrete sequence. The resulting value of its computation is, itself, a discrete sequence, and not a continuous function, of equally spaced samples, in the frequency domain [76], according to

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}, \quad n,k \in \mathbb{N}_0 \tag{4.1}$$

where $X[k]$ and $x[n]$ represent the finite set of samples in frequency and time domain, respectively, and

$$W_N^{nk} = \mathrm{e}^{\frac{-j2\pi}{N}} \tag{4.2}$$

refers to the N complex, harmonically related, exponential coefficients of the DFS. Equation 4.1 is usually referred as the analysis equation. Its dual, the synthesis equation, or IDFT, can then be written as:

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]W_N^{-nk}, \quad n,k \in \mathbb{N}_0, \tag{4.3}$$

implying the following relationship:

$$x[n] \overset{\mathbf{DFT}}{\longleftrightarrow} X[k] \tag{4.4}$$

where $x[n]$ and $X[k]$ may both be complex values.

Through a direct evaluation of the analysis equation, one can find that the typical DFT equation has a high computational requirement. For each k frequency component, the basic algorithm, requires:

- N complex multiplications;
- $N-1$ complex additions.

By expressing (4.1) in terms of operations on real numbers, we obtain:

$$\begin{aligned} X[k] = \sum_{n=0}^{N-1} & \left[\mathfrak{Re}\{x[n]\}\mathfrak{Re}\{W_N^{nk}\} - \mathfrak{Im}\{x[n]\}\mathfrak{Im}\{W_N^{nk}\} + \right. \\ & \left. \mathrm{j}(\mathfrak{Re}\{x[n]\}\mathfrak{Im}\{W_N^{nk}\} + \mathfrak{Im}\{x[n]\}\mathfrak{Im}\{W_N^{nk}\}\right] \end{aligned} \tag{4.5}$$

Taking into account that each complex multiplication requires 4 real multiplications and 2 real additions, and each complex addition requires 2 real additions, one can find that the total processing cost is:

- for each value of k:
  - $4N$ real multiplications;
  - $4N-2$ real additions;
- for the complete transform:
  - $4N^2$ real multiplications;
  - $N(4N-2)$ real additions.

4.1.1 Fast Fourier Transform

In 1965, Cooley and Tukey [77] published an algorithm for the efficient computation of the discrete Fourier transform, always applicable when N is a composite number (i.e.: the product of two or more integers). This led to the development of a multiple number of highly efficiently computational algorithms, that came to be know as the Fast Fourier Transform (FFT) [78]. In the

following subsections, an explanation of the two most commonly known algorithms is shown and explained, which reduce the complexity of the DFT from $O(N^2)$ to $O(N\log(N))$. Both algorithms are based on the principle of decomposing either the input or output sequence into successively smaller DFT computations.

4.1.1.1 Decimation-in-time

The basic principle of the decimation-in-time (DIT) algorithm, is based on the decomposition of the set $x[n]$ into smaller subsequences. Considering the special case where N is an integer power of 2, N is also an even number, so we can compute $X[k]$ by separating $x[n]$ into two $(N/2)$-point sequences, consisting of the even and odd-numbered samples of $x[n]$, as follows.

$$X[k] = \sum_{n\ even} x[n]W_N^{nk} + \sum_{n\ odd} x[n]W_N^{nk}. \tag{4.6}$$

Performing the substitution $n = (2r)$ for n *even* and $n = (2r+1)$ for n *odd*, the equation can be separated into two sums:

$$\begin{aligned} X[k] &= \sum_{r=0}^{(N/2)-1} x[2r]W_N^{2rk} + \sum_{r=0}^{(N/2)-1} x[2r+1]W_N^{(2r+1)k}. \\ &= \sum_{r=0}^{(N/2)-1} x[2r](W_N^2)^{rk} + (W_N^k)\sum_{r=0}^{(N/2)-1} x[2r+1](W_N^2)^{rk} \end{aligned} \tag{4.7}$$

and since

$$W_N^2 = \mathrm{e}^{-2\frac{j2\pi}{N}} = \mathrm{e}^{-\frac{j2\pi}{(N/2)}} = W_{N/2}, \tag{4.8}$$

we can write (4.7) as

$$\begin{aligned} X[k] &= \sum_{r=0}^{(N/2)-1} (x[2r] + W_N^k x[2r+1])W_{N/2}^{rk} \\ X[k] &= G[k] + W_N^k H[k], \quad k \in \{0,1...,N-1\} \end{aligned} \tag{4.9}$$

where G[k] and H[k] correspond to the $(N/2)$ DFTs of the even and odd samples, respectively, as exemplified in fig. 4.1, for $N = 8$. Even though k ranges from 0 to $N-1$, each of the sums only needs to be computed for k between 0 and $(N/2)-1$, since $G[k]$ and $H[k]$ will be periodic sequences, of period $(N/2)$.

With the restructured decomposition, one can easily see that the computation of (4.9) requires at most $N + N^2/2$ complex multiplications and additions, which for $N > 2$ will always be less than the previously mentioned N^2 computational cost. Furthermore, the previous equation corresponds to breaking the original DFT into 2. If (N/2) is an even number, (4.9) can be further decomposed into two (N/4) DFTs, which can then be combined, to yield the same result as before. This decomposition can be represented, for $G[k]$ as follows:

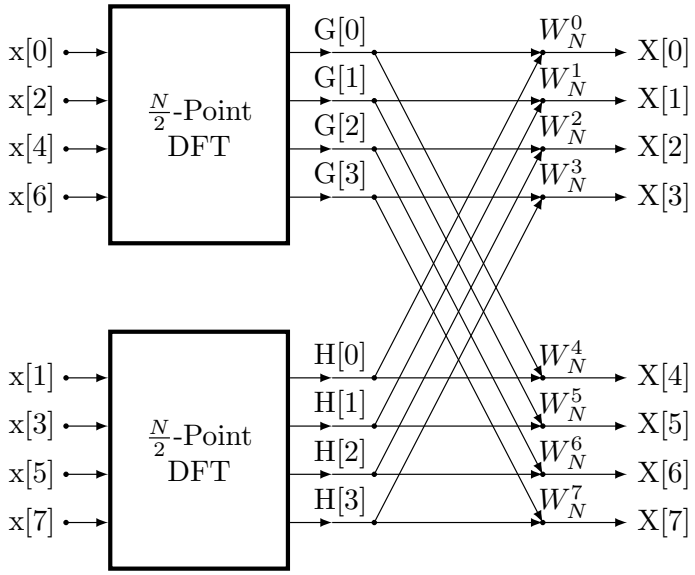


Figure 4.1: Flow-graph of the DIT decomposition of an 8-point DFT into two $(N/2)$-point DFT computations.

$$\begin{aligned} G[k] &= \sum_{r=0}^{(N/2)-1} g[r]W_{(N/2)}^{rk} \\ &= \sum_{l=0}^{\frac{N}{4}-1} g[2l]W_{(N/2)}^{2lk} + g[2l+1]W_{(N/2)}^{(2l+1)k} \end{aligned} \tag{4.10}$$

or

$$G[k] = \sum_{l=0}^{\frac{N}{4}-1} g[2l]W_{\frac{N}{4}}^{lk} + W_{(N/2)}^{k} g[2l+1]W_{\frac{N}{4}}^{lk}. \tag{4.11}$$

Similarly, $H[k]$ would be represented as

$$H[k] = \sum_{l=0}^{\frac{N}{4}-1} h[2l]W_{\frac{N}{4}}^{lk} + W_{(N/2)}^{k} h[2l+1]W_{\frac{N}{4}}^{lk}. \tag{4.12}$$

Consequently, the previous $(N/2)$-point DFT can be obtained by combining the $(N/4)$-point DFTs of the sequences $g[2l]$ and $g[2l+1]$, and $h[2l]$ and $h[2l+1]$, as represented in fig. 4.2.

By combining the decompositions observed in fig. 4.1 and 4.2, we obtain the resulting flow-graph for the computation of the 8-point DFT, observed in fig. 4.3.

In the general case, we would proceed to decompose the $N/4$-point DFTs into $N/8$ (and $N/16$, reciprocally), until we are left with only 2-point transforms. It can easily be seen, that if N is a composite number, such that $N = 2^v$, by employing this decomposition $v = \log_2(N)$ times, the

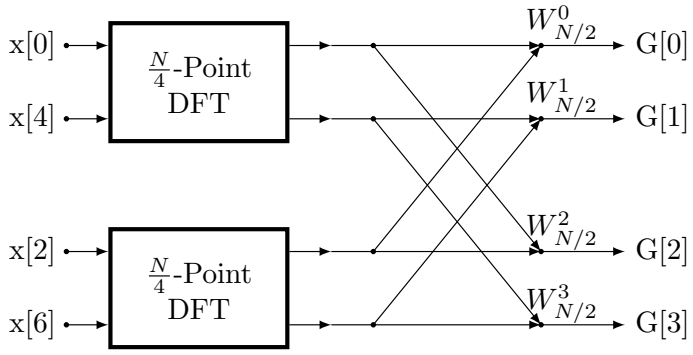


Figure 4.2: Decomposition of an $(N/2)$-point DFT into two $(N/4)$-point DFTs.

maximum number of complex multiplications and additions would be equal to $N_v = N\log_2(N)$, whose operations can be explicitly found in fig. 4.4. By counting each branch, we note that every stage has N complex multiplications and N complex additions.

The computational requirements can be further reduced, by exploiting the symmetry and periodicity properties of the coefficient W_N^r. By noticing, that when proceeding from one stage to the next, the basic computation is in the form of fig. 4.5a, the coefficients are always powers of W_N and separated by $N/2$. This flow-graph is usually called, in literature, a *butterfly*, due to its shape. According to the symmetry property,

$$W_N^{N/2} = \mathrm{e}^{(-j\frac{2\pi}{N}(N/2))} = \mathrm{e}^{-j\pi} = -1, \tag{4.13}$$

so, the factor $W_N^{r+(N/2)}$ can be written as

$$W_N^{r+(N/2)} = W_N^{N/2}W_N^r = -W_N^r. \tag{4.14}$$

With this observation, the butterfly computation can be simplified to the form of fig. 4.5b, leading to the simplified flow-graph in fig. 4.6.

4.1.1.2 Decimation-in-frequency

Similar to the DIT algorithm, the decimation-in-frequency (DIF) attempts to reduce the computational requirements, by decomposing the output sequence into smaller sets, if, for simplicity of explanation, N is a composite number and power of 2. Taking into account (4.1), one can accomplish that purpose by writing, the even and odd numbered frequency samples as

$$\begin{aligned} X[2k] &= \sum_{n=0}^{N-1} x[n]W_N^{n(2k)} \\ X[2k+1] &= \sum_{n=0}^{N-1} x[n]W_N^{n(2k+1)}, \end{aligned} \tag{4.15}$$

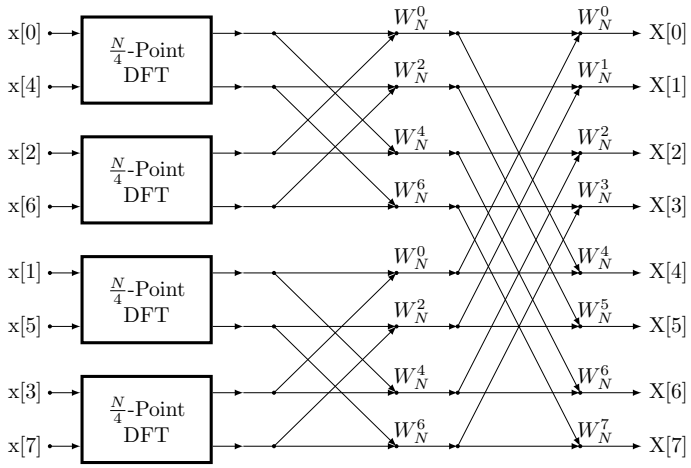


Figure 4.3: Substitution of fig. 4.2 into fig. 4.1.

which can be express in the following form:

$$\begin{aligned} X[2k] &= \sum_{n=0}^{(N/2)-1} x[n]W_N^{n(2k)} + \sum_{n=(N/2)}^{N-1} x[n]W_N^{n(2k)} \\ X[2k+1] &= \sum_{n=0}^{(N/2)-1} x[n]W_N^{n(2k+1)} + \sum_{n=(N/2)}^{N-1} x[n]W_N^{n(2k+1)}. \end{aligned} \tag{4.16}$$

Performing a variable substitution on the second term of the sum, we obtain

$$\begin{aligned} X[2k] &= \sum_{n=0}^{(N/2)-1} x[n]W_N^{n(2k)} + \sum_{n=0}^{(N/2)-1} x[n+(N/2)]W_N^{(2k)(n+(N/2))} \\ X[2k+1] &= \sum_{n=0}^{(N/2)-1} x[n]W_N^{n(2k+1)} + \sum_{n=0}^{(N/2)-1} x[n+(N/2)]W_N^{(2k+1)(n+(N/2))}. \end{aligned} \tag{4.17}$$

By re-exploring the symmetry and periodicity properties of W_N^{nk}, we can rearrange the above equations to

$$\begin{aligned} X[2k] &= \sum_{n=0}^{(N/2)-1} (x[n]+x[n+(N/2)])\, W_{N/2}^{nk} \\ X[2k+1] &= \sum_{n=0}^{(N/2)-1} (x[n]-x[n+(N/2)])\, W_N^n\, W_{N/2}^{nk} \end{aligned} \tag{4.18}$$

From the above equations, we can already see a difference between the DIT and DIF algorithms, were the *butterfly* appears, explicitly, in the inner summation. On the other hand, by writing

$$\begin{aligned} g[n] &= (x[n]+x[n(+N/2)] \\ h[n] &= (x[n]-x[n(+N/2)]\, W_N^n \end{aligned} \tag{4.19}$$

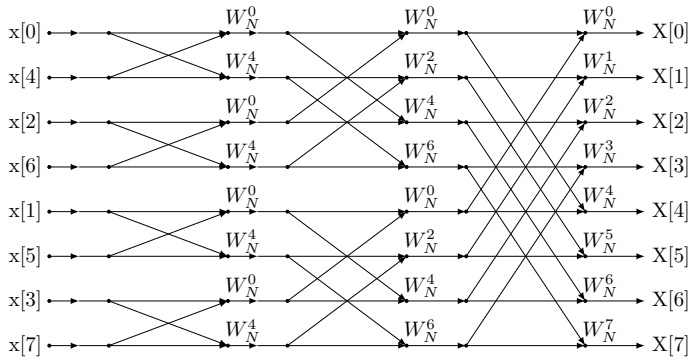


Figure 4.4: Flow-graph of complete DIT decomposition of an 8-point DFT.

the process of decomposition into even smaller subsets of the DFT, becomes trivial, since we are left with

$$\begin{aligned} X[2k] &= \sum_{n=0}^{(N/2)-1} g[n]\, W_{N/2}^{nk} \\ X[2k+1] &= \sum_{n=0}^{(N/2)-1} h[n]\, W_{N/2}^{nk}. \end{aligned} \tag{4.20}$$

We can easily see, that both expressions, in (4.20), correspond two performing 2 DFTs of size $N/2$, each. If $N = 8$, and we continue to decompose the DFT, the above results would be represented as in the flow-graph of fig. 4.7.

4.1.2 Fast Fourier Transform radix-4

As explained above, the complexity of the DFT can be reduced, by decomposing it into smaller subsets. Until this point, only decomposition for N power of 2 was explored. The subset of algorithms, following this property, are usually referred as radix-2. In the original work by Cooley and Tukey, it was actually proven, that different decompositions could offer similar or better computational effort results. In fact, they proved that decomposing N into powers of 3 (i.e.: a radix-3 algorithm), actually offered the best computational effort. However, this idea was rapidly abandoned, due to the binary nature of computer technology and the gains in using binary representations.

A possible approach, would be to decompose the DFT into 4 subsets, i.e., perform a FFT radix-4 (assuming N is a power of 4), as follows, using the DIF procedure[1]:

$$\begin{aligned} X[4k] &= \sum_{n=0}^{N-1} x[n]\, W_N^{n(4k)} \\ X[4k+1] &= \sum_{n=0}^{N-1} x[n]\, W_N^{n(4k+1)} \end{aligned}$$

[1] A similar approach can be taken into account, using DIT.

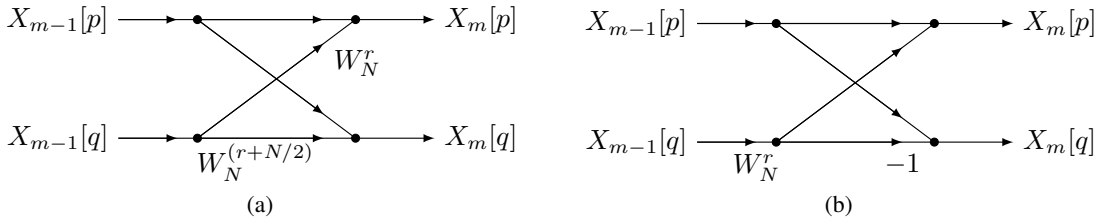


Figure 4.5: Butterfly (a) basic computation and (b) with only one multiplication.

$$\begin{aligned} X[4k+2] &= \sum_{n=0}^{N-1} x[n]\, W_N^{n(4k+2)} \\ X[4k+3] &= \sum_{n=0}^{N-1} x[n]\, W_N^{n(4k+3)} \end{aligned} \tag{4.21}$$

which, after simplifications, can be written as:

$$\begin{aligned} X[4k] &= \sum_{n=0}^{N/4-1} (x[n]+x[n+(N/4)]+x[n+(N/2)]+x[n+(3N/4)])\, W_{N/4}^{nk} \\ X[4k+1] &= \sum_{n=0}^{N/4-1} (x[n]-jx[n+(N/4)]-x[n+(N/2)]+jx[n+(3N/4)])\, W_N^n\, W_{N/4}^{nk} \\ X[4k+2] &= \sum_{n=0}^{N/4-1} (x[n]-x[n+(N/4)]+x[n+(N/2)]-x[n+(3N/4)])\, W_N^{2n}\, W_{N/4}^{nk} \\ X[4k+3] &= \sum_{n=0}^{N/4-1} (x[n]+jx[n+(N/4)]-x[n+(N/2)]-jx[n+(3N/4)])\, W_N^{3n}\, W_{N/4}^{nk} \end{aligned} \tag{4.22}$$

taking into account similar properties as in 4.13 and 4.14:

$$\begin{aligned} W_N^{n(4k)} &= \mathrm{e}^{(-j\frac{2\pi(4nk)}{N})} = \mathrm{e}^{(-j\frac{2\pi nk}{N/4})} \\ &= W_{N/4}^{nk} \end{aligned} \tag{4.23}$$

$$\begin{aligned} W_N^{(n+(N/4))(4k)} &= \mathrm{e}^{(-j\frac{2\pi(4nk)}{N})}\, \mathrm{e}^{(-j\frac{2\pi(Nk)}{N})} = \mathrm{e}^{(-j\frac{2\pi(4nk)}{N})} \\ &= W_{N/4}^{nk} \end{aligned} \tag{4.24}$$

$$\begin{aligned} W_N^{(n+(N/4))(4k+1)} &= \mathrm{e}^{(-j\frac{2\pi(4nk)}{N})}\, \mathrm{e}^{(-j\frac{2\pi n}{N})}\, \mathrm{e}^{(-j\frac{2\pi Nk}{N})}\, \mathrm{e}^{(-j\frac{2\pi(N/4)}{N})} \\ &= -j\, \mathrm{e}^{(-j\frac{2\pi nk}{N/4})}\, \mathrm{e}^{(-j\frac{2\pi n}{N})} \\ &= -jW_{N/4}^{nk}\, W_N^n. \end{aligned} \tag{4.25}$$

As already explained before, the DIF algorithm has the particularity of explicitly demonstrating the butterfly equation. In this case, since we are decomposing the DFT into 4 subsets, it

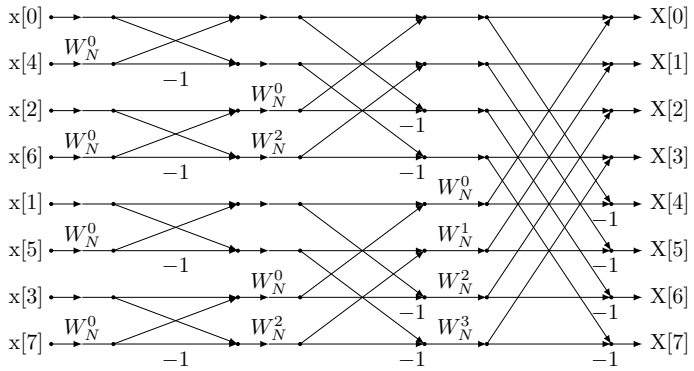


Figure 4.6: Flow-graph of 8-Point DFT using the simplified butterfly computation.

requires 4 branches, instead of 2, as seen in fig. 4.8, and can be translated to the following matrix:

$$BF = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -j & -1 & +j \\ +1 & -1 & +1 & -1 \\ +1 & +j & -1 & -j \end{bmatrix}, \tag{4.26}$$

which means that the inner sums, of the FFT, can be seen as a complex vectorial operation.

Simultaneously, we can write:

$$\begin{aligned} e[n] &= x[n]+x[n+(N/4)]+x[n+(N/2)]+x[n+(3N/4)] \\ f[n] &= (x[n]-jx[n+(N/4)]-x[n+(N/2)]+jx[n+(3N/4)])\, W_N^n \\ g[n] &= (x[n]-x[n+(N/4)]+x[n+(N/2)]-x[n+(3N/4)])\, W_N^{2n} \\ h[n] &= (x[n]+jx[n+(N/4)]-x[n+(N/2)]-jx[n+(3N/4)])\, W_N^{3n}, \end{aligned} \tag{4.27}$$

and, by combining 4.27 into 4.22:

$$\begin{aligned} X[4k] &= \sum_{n=0}^{N/4-1} e[n]\, W_{N/4}^{nk} \\ X[4k+1] &= \sum_{n=0}^{N/4-1} f[n]\, W_{N/4}^{nk} \\ X[4k+2] &= \sum_{n=0}^{N/4-1} g[n]\, W_{N/4}^{nk} \\ X[4k+3] &= \sum_{n=0}^{N/4-1} h[n]\, W_{N/4}^{nk}, \end{aligned} \tag{4.28}$$

we can see that if $N/4$ is still a power of 4, we can continue to decompose the DFT, according to 4.21. This iterative procedure can be exemplified, according to fig. 4.9.

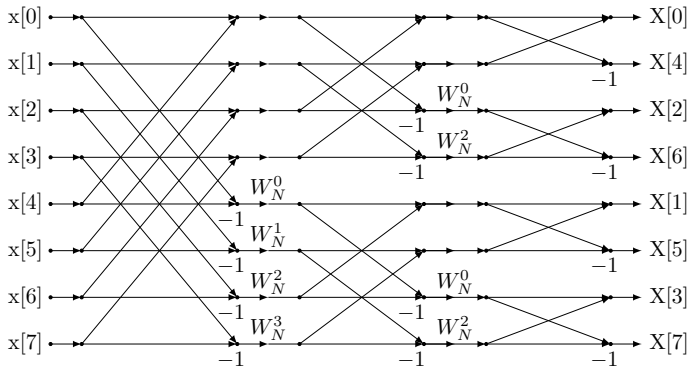


Figure 4.7: Flow-graph of 8-Point DFT using the DIF algorithm.

4.1.3 Digit-Reversal

A particular characteristic of the DIT and DIF algorithms is that any permutation of the order inside a given stage, will still yield the same output result, just in a different order. In fact, the main difference between the DIF and DIT algorithms is the order of the input and output sequences. Typically, if we follow the presented flowgraphs, we will find that an input sequence in order will result in a output, whose samples are in (what is known as) digit-reversed order[2], according to the applied radix.

Typically, if we want the output sequence to be in order, we would digit-reverse the input sequence and use the DIT algorithm, for computing the FFT. On the other hand, if we can not interfere with the order of the input samples (due to architecture restrictions, for example), we would use the DIF and then digit-reverse the output. If we choose either of the procedures, it will always be required to perform the digit-reversal procedure, either on the input or output samples. An example can be found in tables A.1 and A.2, for radix-2 and radix-4 respectively.

4.1.4 Overview

As can be seen, the process of successively decomposing the DFT into smaller subsets, usually referred as stages, leads to a more efficient computation effort. Instead of calculating the result, as a single iterative process, smaller iterations are performed, in each stage, and then combined to yield the same result. This is only achievable due to the periodic nature, of the coefficient W_N^{nk}, rom now on reffered as *twiddle*. Each stage, takes as an input a set of samples, and instead of performing a rather expensive multiplication, they merely have to perform complex number additions, according to their respective butterflies.

[2]For radix-2 algorithms, it is usually named bit-reversed order.

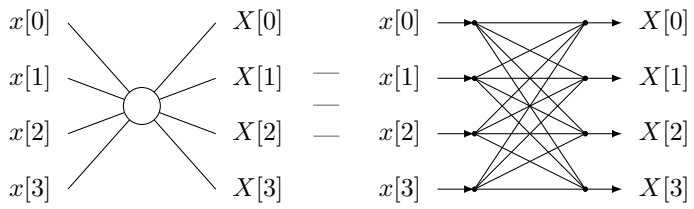


Figure 4.8: Radix-4 butterfly.

4.2 CORDIC

The COordinate Rotation DIgital Computer (CORDIC), developed by Jack Volder, in 1959 [79], is, as stated by its developer, "a highly efficient method to compute elementary functions" [80], By employing fixed point notation, the CORDIC is specially suited for solving trigonometric relationships, vector rotations in the Euclidean plane and conversion from rectangular to polar coordinates. The system can be employed in two different subsets:

- Rotation Mode: suitable for vector rotations or computing the *cosine* and *sine* function;
- Vectoring Mode: usually used to perform cartesian to polar coordinate transformations.

Due to the nature of this work, we will only focus on the former and, more specifically, in its functionality for performing angle rotations.

4.2.1 Rotation mode

The CORDIC rotation mode, finds its basic structure on the rotation matrix, which can be written as

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}. \tag{4.29}$$

By taking an input vector, defined by its $\mathbb{R}^2$ coordinates (x,y) and an input rotation angle (θ), we can find the coordinates of the resulting rotation (x',y') as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ y \end{bmatrix} \Leftrightarrow \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{4.30}$$

as can be seen in fig. 4.10.

This can easily be proven, by taking into account that

$$\begin{aligned} (x,y) &= \begin{cases} R\cos(\phi) \\ R\sin(\phi) \end{cases} \\ (x',y') &= \begin{cases} R\cos(\psi) \\ R\sin(\psi) \end{cases} \Leftrightarrow \begin{cases} R\cos(\phi+\theta) \\ R\sin(\phi+\theta) \end{cases} \end{aligned} \tag{4.31}$$

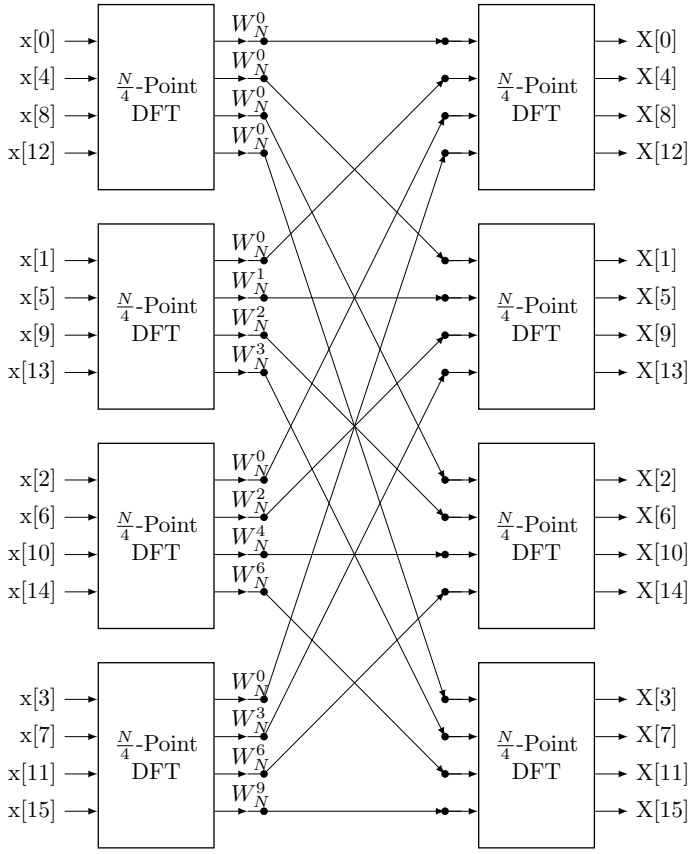


Figure 4.9: Block diagram of a 16-point DFT, using the DIF radix-4 algorithm.

but, simultaneously, using the trigonometric identities of the *sine* and *cosine* function:

$$(x',y') = \begin{cases} R(\cos(\phi)\cos(\theta)-\sin(\phi)\sin(\theta)) \\ R(\sin(\phi)\cos(\theta)+\cos(\phi)\sin(\theta)) \end{cases} \tag{4.32}$$

which equates to

$$(x',y') = \begin{cases} x\cos(\theta)-y\sin(\theta)) \\ y\cos(\theta)+x\sin(\theta)) \end{cases} \tag{4.33}$$

If instead of using θ as the rotation angle, we employ the summation of smaller angles α_i, i.e.:

$$\theta = \sum_{i=0}^{\infty} \alpha_i \tag{4.34}$$

and apply it to (4.30) and (4.33), we can find that the resulting rotation matrix, becomes the product of the individual matrices

$$R(\theta) = \prod_{i=0}^{\infty} \begin{bmatrix} \cos(\alpha_i) & -\sin(\alpha_i) \\ \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix}, \tag{4.35}$$

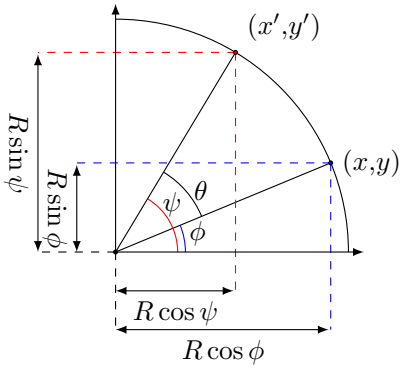


Figure 4.10: Example rotation in the Euclidean plane, around a circle of radius R.

and the rotated vector:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=0}^{\infty} \begin{bmatrix} \cos(\alpha_i) & -\sin(\alpha_i) \\ \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{4.36}$$

By dividing all terms of the inner matrix by $cos(\alpha_i)$,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=0}^{\infty} \cos(\alpha_i) \begin{bmatrix} 1 & -\tan(\alpha_i) \\ \tan(\alpha_i) & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{4.37}$$

and performing the variable substitution $\alpha_i = \arctan(2^{-i})$, we are left with the following result:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \prod_{i=0}^{\infty} \cos(\arctan(2^{-i})) \begin{bmatrix} 1 & -2^{-i} \\ 2^{-i} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{4.38}$$

where,

$$\begin{aligned} \cos(\arctan(x)) &= \frac{1}{\sqrt{1+x^2}} \\ \Rightarrow \cos(\arctan(2^{-i})) &= \frac{1}{\sqrt{1+2^{-2i}}} \end{aligned} \tag{4.39}$$

As can be seen, the decomposition performed by the CORDIC, can accomplish a perfect computation of a vector rotation, by the simple process of summing and shifting variables, iteratively. The final result, is then multiplied by a constant factor K, which is equal to

$$K = \prod_{i=0}^{\infty} \frac{1}{\sqrt{1+2^{-2i}}}. \tag{4.40}$$

which converges to

$$\lim_{i\to\infty} K \approx 0.6072529... \tag{4.41}$$

Until this point, only the theoretical point of view of the CORDIC implementation has been demonstrated. In practice, to ensure proper convergence of the method, the smaller angles α_i, henceforth named micro-rotations, must either be positive, or negative. In fact, for a proper result, the input rotation angle is constrained to:

$$\theta = \pm\sum_{i=0}^{\infty}\arctan(2^{-i}), \tag{4.42}$$

which tends towards

$$\lim_{i\to\infty}\theta \approx \pm 99.88296^{\circ} = \pm 1.743286\ rads. \tag{4.43}$$

For popular reasons, it is usually assumed that the rotation angle is bounded by $]-\pi/2,\pi/2]$. By performing this assumption, the first iteration of the CORDIC algorithm usually consists in setting the rotation angle to the first and fourth quadrant of the Cartesian plane, as can be found in (1).

To proper constrain the convergence of the algorithm, an additional variable is required, to select between a positive and negative rotation:

$$\theta = \sum_{i=0}^{\infty} d_i\ \arctan(2^{-i}) \tag{4.44}$$

or, in its iterative form

$$z_{i+1} = z_i - d_i\ \arctan(2^{-i}), \tag{4.45}$$

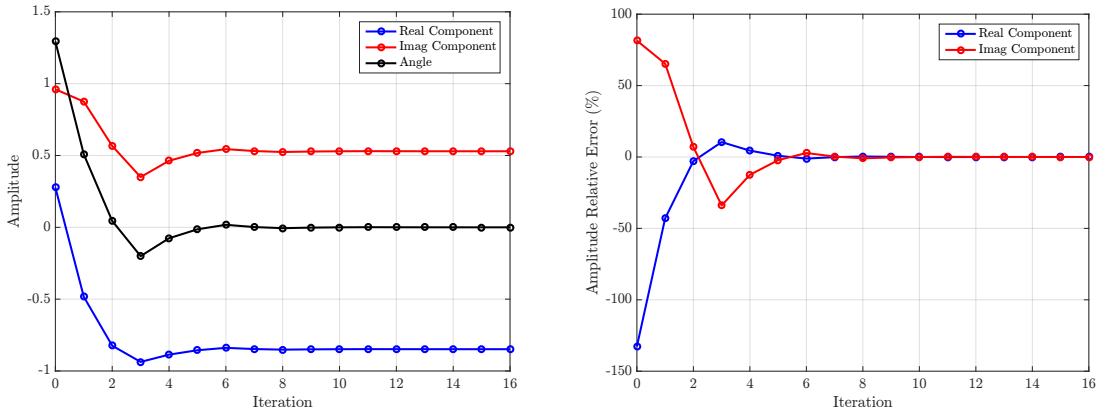
where z_{i+1} and z_i correspond to the missing rotation step for iterations $i+1$ and i, respectively, and

$$d_i = \begin{cases} -1 & , \text{if } z_i < 0 \\ 1 & , \text{if } z_i > 0 \end{cases}. \tag{4.46}$$

Similarly, the rotation matrix has to be adjusted to account for the sign of the rotation, resulting in the following iterative procedure:

$$\begin{aligned} x_{i+1} &= x_i - y_i\ d_i\ 2^{-i} \\ y_{i+1} &= y_i + x_i\ d_i\ 2^{-i} \\ z_{i+1} &= z_i - d_i\ \arctan(2^{-i}) \end{aligned} \tag{4.47}$$

The complete iterative procedure is demonstrated in (2) and an example application of the CORDIC algorithm and its relative error (for each iteration) can be found in fig. 4.11a and fig. 4.11b, respectively. It should be noted, that despite its seemingly large initial error, during the first iterations, we can see in table. (B.1) that after only 8 iterations, the algorithm converges to a very low relative error, as it is expected.



(a) (b)

Figure 4.11: CORDIC (a) example iterational procedure and (b) corresponding relative error.

4.3 FPGA implementation

Due to the time restrictions of this dissertation, the main focus of the hardware implementation was given to the core module of the OFDM modulator and demodulator (i.e.: the IFFT and FFT). In this section, a presentation will be performed for an area-efficient implementation of the aforementioned module, capable of being synthesized for FPGA or application specific integrated circuits (ASIC) usage. Low-level design choices, like the required number of bits will be demonstrated and explained. Further considerations, like pipelining will be explored and demonstrated.

4.3.1 Fixed-point notation

For an efficient hardware implementation, in terms of both area and time constraints, the implemented modules were all coded using fixed-point notation, according to fig. 4.12, in two's complement representation, which means 1 bit is always reserved for sign representation. In order to guarantee convergence of all the procedures required, two bits were used for the integer part, to account for any possible signal growth, on reception in the DSP chain. It should be noted, that this

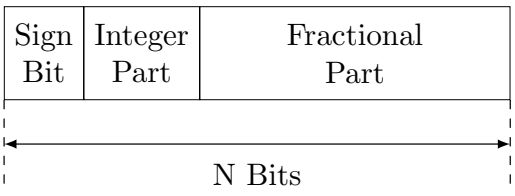


Figure 4.12: Fixed point mantissa.

amounts to a maximum and minimum value of 3 and -3, respectively, for the integer part alone. Additionally, all the constellations considered for transmission, do not exceed the value 1.2, when all symbols are normalized to unity mean power.

4.3.2 Golden models

To properly test the design choices, both floating-point and fixed-point models (of the FFT and CORDIC) were written, before starting the HDL coding. The former model was written in MATLAB^{TM}, according to the radix-4 demonstration performed, to verify all equations, and computational requirements. The latter was coded in C, in a manner similar to HDL, to facilitate any possible debugging requirements.

In order to access the performance of the FFT, under different constraints (size and fixed point word length), a qualitative analysis is of the essence, to select a suitable number of bits for the final design of the baseband processor implementation, which does not affect performance or induces any possible SNR increase. The methodology to be used, for such an analysis, can be found in fig. 4.13, in a similar approach to [81]. At the time of writing, a quantitative analysis, for error

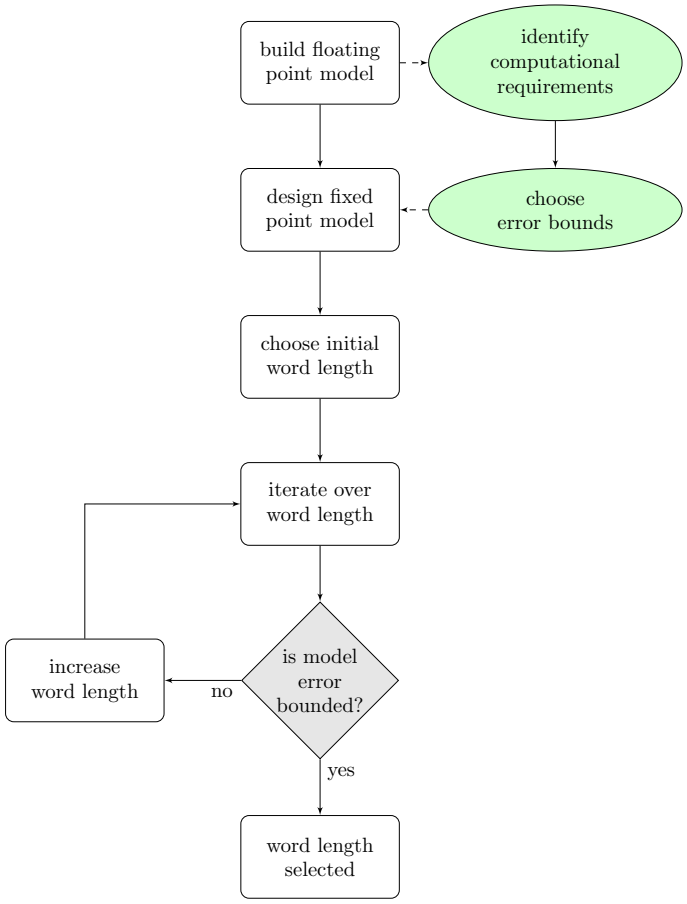


Figure 4.13: Qualitative analysis procedure for fixed-point implementations.

assessment, has already been performed, using the developed fixed-point models. Such analysis includes metrics like the SNR (which can be found in fig. 4.14a and 4.14b for the CORDIC and FFT, respectively), mean error, error variance and max absolute error for each component[3]. These metrics were obtained, by feeding to each developed algorithm a random uniformly dis-

[3]Some additional figures for these last metrics can be found in (C).

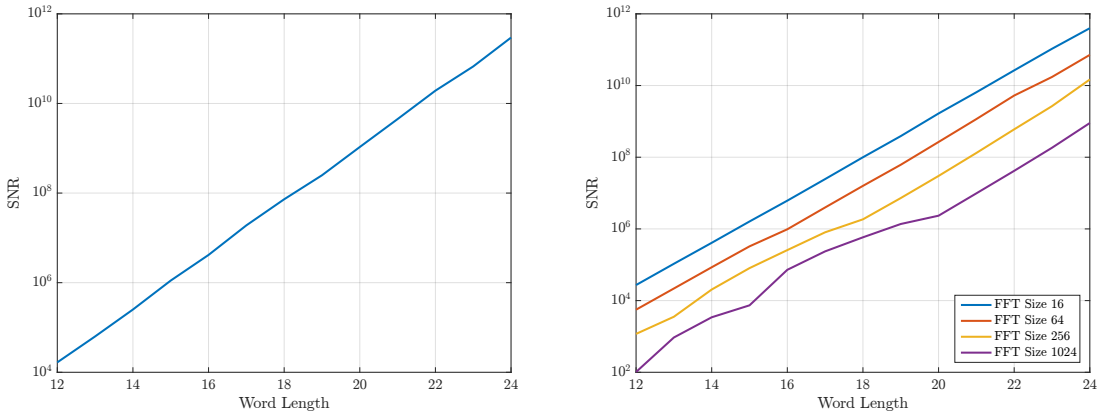


Figure 4.14: Expected signal-to-noise ratio of the (a) CORDIC and (b) FFT unit, for different wordlengths.

tributed input sequence, with values ranging from -3 to 3 (in floating point). These values where then changed to the required fixed-point notation, according to 4.3.1, and compared against their expected result, in floating point, as can be seen in fig. 4.15.

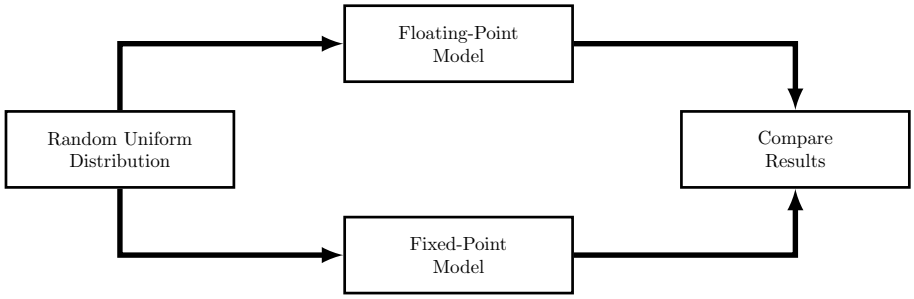


Figure 4.15: Statistical analysis procedure.

According to [76], increasing the FFT size by a factor of 2, leads to an half-bit increase of the required input wordlength, to achieve the same SNR. It is interesting to note that the obtained SNR curves, follow this behavior (in this case, increasing the FFT size by a factor of 4, requires an increase of 1 bit, to achieve a similar result). It should be noted, that the actual SNR can be quite different, according to the input sequences used. For example, by feeding a BPSK modulated signal, input values will always be constrained to either -1 or 1.

4.3.3 Pipelined CORDIC

As mentioned in 3.5, a minimum sampling frequency of 80MHz, for the clock domain, is required. According to the shown procedures, to be developed, it is expected that the CORDIC operation translates into the highest computational requirements. Due to its iterative nature, a

pipelined version can be built, in a trade-off between area resources (mostly in required registers) and functional speed is required. The basic structure of a CORDIC iteration can be seen in fig. 4.16.

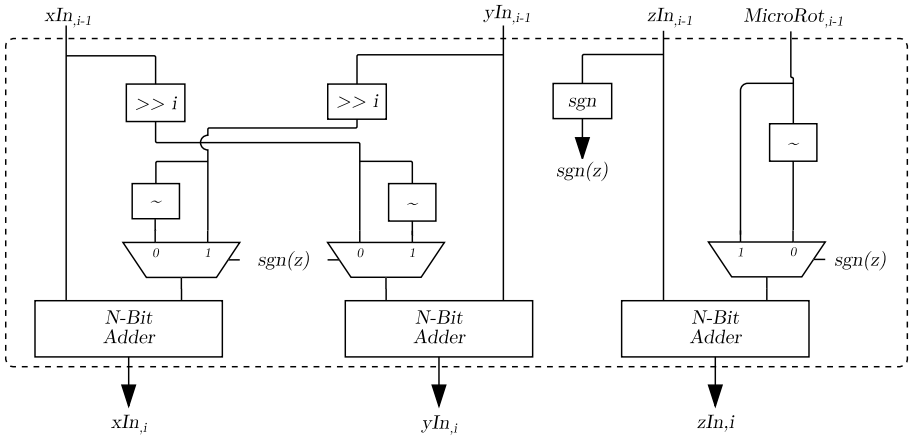


Figure 4.16: Functional blocks of a single CORDIC iteration.

The pipelined version can be designed by interconnecting multiple stages, interchangeably with registers. An example of such a combination can be found in fig. 4.17.

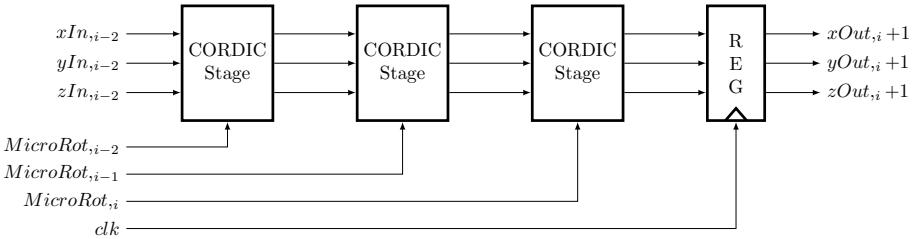


Figure 4.17: Example CORDIC pipeline stage, with 3 iterations.

4.3.3.1 Approximation considerations

According to 4.45, the remaining angle rotation should decrease $\arctan(2^{-i})$, in each successive iteration, i.e.: by the same angle that the input vector was rotated. It is common practice, in literature, to change the aforementioned value to simply 2^{-i}. This is a valid approximation, if we take into account the Taylor polynomial approximation of $\tan(x)$, around zero:

$$\frac{\partial \tan(x)}{\partial x} = \sec(x)^2 \tag{4.48}$$

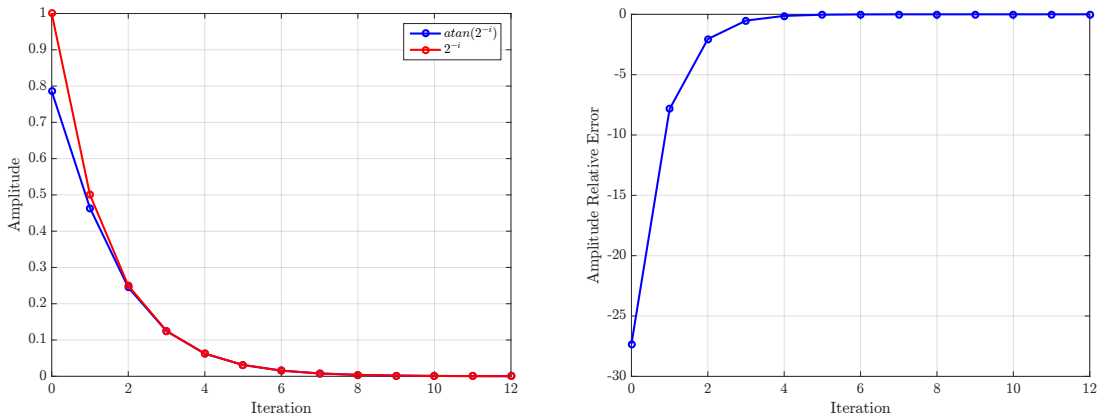
$$\Rightarrow \tan(x) \approx \tan(0) + \frac{\sec(0)^2}{1}(x-0)$$

$$= x \tag{4.49}$$

$$\Rightarrow \tan(2^{-i}) \approx 2^{-i} \tag{4.50}$$

$$\Rightarrow \arctan(2^{-i}) \quad \approx \quad \arctan(\tan(2^{-i})) = 2^{-i}. \tag{4.51}$$

In fact, there are works that exploit this property [82, 83], when a high number of iterations is required. The approximation and its relative error, for different iteration values can be seen in fig. 4.18a and fig. 4.18b, respectively. At the authors view, the gains in performing this approxi-



(a) (b)

Figure 4.18: CORDIC (a) micro-rotation comparison between $atan(x)$ and 2^{-x} and (b) corresponding relative error.

mation, for a radix-2 algorithm[4], are not sufficiently high, compared to a LUT (or ROM) solution, when less than 16 iterations are performed. Furthermore, when multiple CORDIC modules are required, the LUT can actually be re-used by all. As such, the latter became the preferred option. An example representation of each micro-rotation, contained in the LUT, can be found in table 4.1.

4.3.3.2 Number of iterations

Another consideration to be taken into account is the number of iterations to perform. For most cases, it can be found that, for an input angle representation of $N-bits$, $N-2$ iterations are required for the algorithm to properly converge, under fixed-point[5]. Given this, the opted solution was to perform as many iterations as the number of bits used in the basic FFT computations (i.e.: the butterflies). In other words, and as an example, given a 16-bit input sequence to the FFT, each CORDIC operation will require 16 iterations. To improve SNR and to account for the increase in magnitude (according to 4.40), the input values, x and y, must be sign-extended by 1 bit.

4.3.4 FFT radix-4 DIF single-delay-feedback

Among the most common algorithms, for efficient implementation of the FFT, the Single-Delay-Feedback, proposed in [84], is among the most interesting ones. By adding a delay chain (of equivalent size to the FFT being performed), to each stage, the same butterfly can be re-used in a

[4] Like the FFT, the CORDIC algorithm can be performed in multiple steps at once, hence the indication of the radix.
[5] Assuming the required initialization is performed, as explained beforehand.

Table 4.1: Example micro-rotation LUT.

Iteration	**Expected Micro-rotation (°)**	**16-Bit Fixed-Point Representation**
0	45.0000	16'b0110010010000111
1	26.5651	16'b0011101101011000
2	14.0362	16'b0001111101011011
3	7.1250	16'b0000111111101010
4	3.5763	16'b0000011111111101
5	1.7899	16'b0000001111111111
6	0.8952	16'b0000000111111111
7	0.4476	16'b0000000011111111
8	0.2238	16'b0000000001111111
9	0.1119	16'b0000000000111111
10	0.0560	16'b0000000000011111
11	0.0280	16'b0000000000001111
12	0.0140	16'b0000000000001000
13	0.0070	16'b0000000000000100
14	0.0035	16'b0000000000000010
15	0.0017	16'b0000000000000001

similar manner to a pipeline. This effectively reduces the total number of butterflies and multipliers per stage from $N/4$ to 1, in a radix-4 implementation, at the expense of N additional registers[6]. The exponential multiplication, by the twiddle factor, can explicitly be seen implemented using the CORDIC unit.

The basic functional unit, for this particular algorithm can be seen in fig. 4.19, where *BF*, and *LSR* (which could also be seen as a first-in first-out - FIFO - memory) refer to the butterfly unit and linear shift register, respectively.

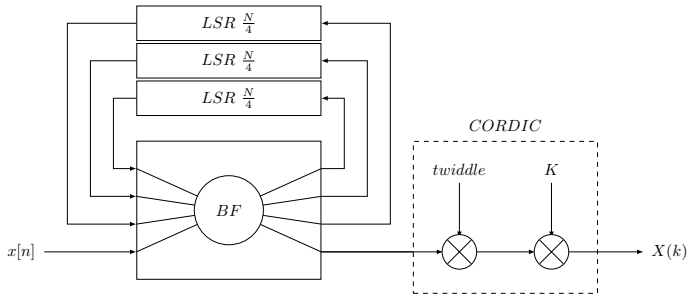


Figure 4.19: Functional diagram of a FFT radix 4 stage.

By pipelining multiple stages, (but with decreasing lengths of the LSR chain, according to $N/4$), an area-efficient design can be achieved, with the obvious trade-off of, at least, a N clock cycles delay, since the first input sample is fed to the module and an output sample is computed.

[6]It should be noted that all implementations of the FFT, actually, have a similar or higher memory logic cost.

This trade-off, however, is negligible, for the application at hand, since input data is assumed to always be received, in bursts, at a 80MHz clock rate.

4.3.4.1 Control Logic

Despite not, initially, intuitive, the SDF algorithm has an actually interesting manner to deal with the required decimation. By defining four different modes of operation, the input samples (received in-order) are driven to their proper positions, by the following manner:

- Mode 0: $N/4$ samples ($x[n]$) are fed to each stage and inputted to the first LSR chain;
- Mode 1: $N/4$ samples ($x[n+(N/4)]$)are fed to each stage and inputted to the second LSR chain;
- Mode 2: $N/4$ samples ($x[n+(N/2)]$) are fed to each stage and inputted to the third LSR chain;
- Mode 3: $N/4$ samples ($x[n+(3N/4)]$) are fed to each stage and combined with the first output samples of the FIFO to be inputted to the butterfly.

As an example, for the first stage of a 16-point FFT, after 12 clock cycles, we can find in table 4.2, the expected state of the LSRs.

Table 4.2: Linear shift-register state, for the first stage of a 16-Point FFT.

	Positition #			
	1	**2**	**3**	**4**
LSR 1	x[0]	x[1]	x[2]	x[3]
LSR 2	x[4]	x[5]	x[6]	x[7]
LSR 3	x[8]	x[9]	x[10]	x[11]

From its analysis, we can see that when the thirteenth sample is driven ($x[12]$), to the FFT stage, sample 0, 4 and 8 are at the output of the LSRs. As stated before, the module will begin operating on mode 3, and the four samples will be fed to the butterfly. The resulting computational output is then re-fed to the LSRs, to re-order the results, at the exception of the first sample, which is driven out of the FFT stage and into the CORDIC, for performing the proper rotation.

After N clock cycles, the module returns to mode 0, where the output sample of the first LSR corresponds to the second sample driven by the butterfly. An example of the required interconnections, for an individual stage, can be found in figs. 4.20a, 4.20b, 4.20c and 4.20d, for each operational mode. This iterative process, can be repeated the amount of times required, to process multiple FFTs, consecutively. This control logic is rather easy to implement, only requiring a registered counter of size $\log_2(N)$ bits, according to each stage. The current, required, mode of operation can be found on the 2 most significant bits (MSB) of the counter.

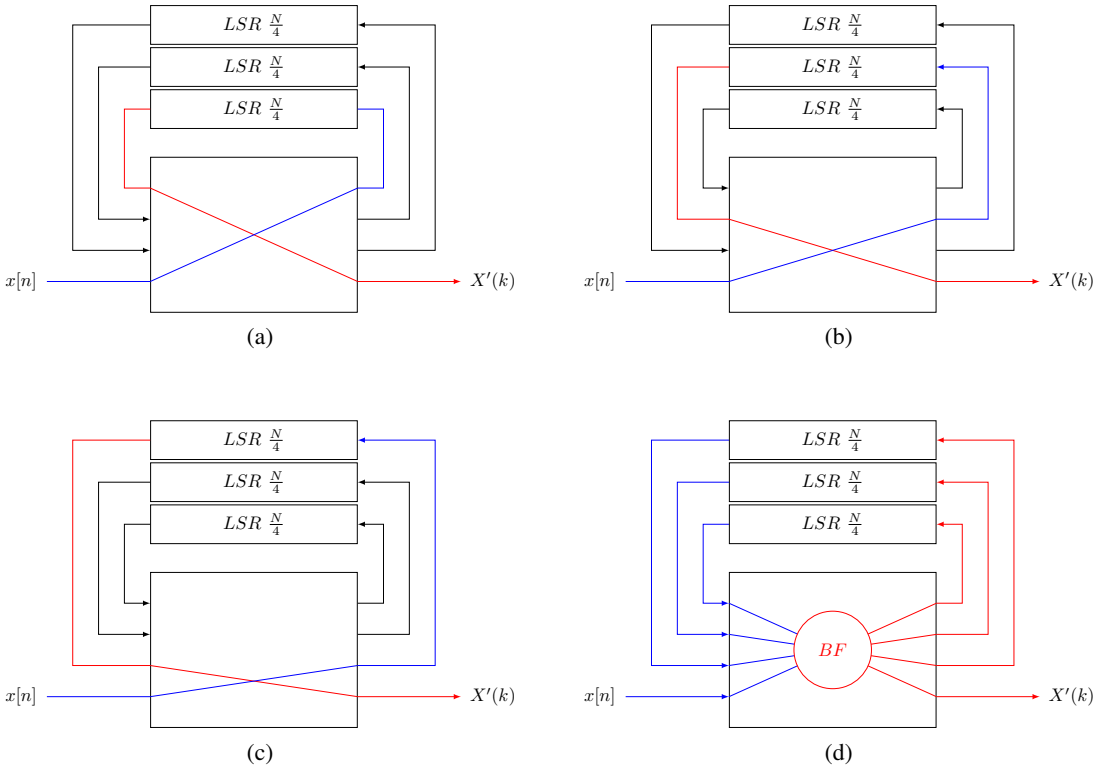


Figure 4.20: FFT radix 4 stage different modes of operation: Mode 0 (a), 1 (b), 2 (c) and 3 (d).

4.3.4.2 Twiddle Generator

To provide the *Twiddle* value to the FFT, most works in literature usually settle on adding a look-up table (LUT) to each individual stage. An alternative approach was pursued, due to the characteristics of this value. Indeed, a LUT was used, but only for the base twiddle value, in each stage, i.e.: W_N^1, where N corresponds to the size of the FFT being computed, in a particular stage. In other words, we only require $N_{Stages} - 1$ twiddle values to be stored in memory (since the last stage does not perform any rotation), as can be found in 4.3[7]. The ensuing values are computed using a registered counter, which adds the base value to itself, on each clock cycle.

Table 4.3: Base twiddle values, for a arbitrary sized radix-4 FFT.

Stage #	Base Twiddle Value	18-Bit Binary Representation
N-1	22.500°	18'b000011001001000011
N-2	5.6250°	18'b000000110010010000
N-3	1.4062°	18'b000000001100100100

To implement the rest of the twiddle values, W_N^{2n} and W_N^{3n}, two multipliers are required, for

[7] It should be noted that the table was written in descending order, on purpose. No matter the size of the FFT, the base twiddle values, for the represented stages, will always be the same, for that particular number of representation bits.

a radix-4 implementation: a multiplication by 2 and 3, which can just be seen as shifting 1 bit left and shifting 1 bit left and re-adding the input value, respectively. To select the corresponding twiddle, for each mode of operation, a mere 4-to-1 multiplexer is needed. An example block diagram, of the implemented twiddle generator can be seen in fig. 4.21. The first multiplexer was used to ensure that the first rotation, at the start of each mode, is always 0.

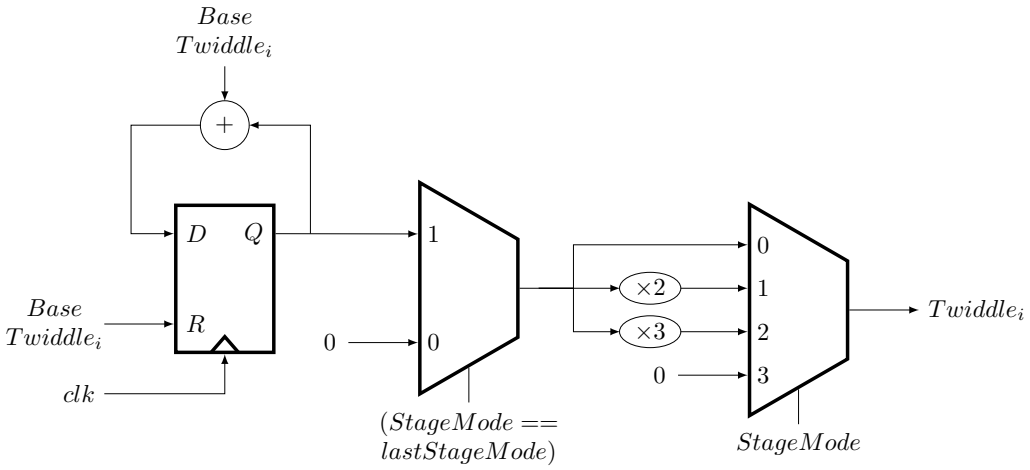


Figure 4.21: Register level block diagram, for generating the FFT twiddle values.

4.3.4.3 Pipeline Considerations

As aforementioned, to compute a complete FFT, of arbitrary size N, $log_4(N)$ stages are required to be connected in series. A pipeline division, at the stage level, was combined with the designed CORDIC, to ensure proper functional behavior and clock frequency requirements.

Power and synchronization constraints were also taken into account, by separating enabling signals, of each individual stage. When a stage reaches (for the first time) mode 3 of operation, it will enable its corresponding CORDIC unit. Only after the complete CORDIC pipeline is followed, will stage $N-1$ drive its *DataReady* signal, enabling the subsequent stage. This procedure is followed in all stages, until a final *DataReady* signal is driven by the last stage (which does not require any rotation). An example interconnection of two stages, can be found in fig. 4.22. Furthermore, since the butterfly operation, if left unchecked, can lead to sample overflow, each FFT stage sample must be priorly truncated 2-bits.

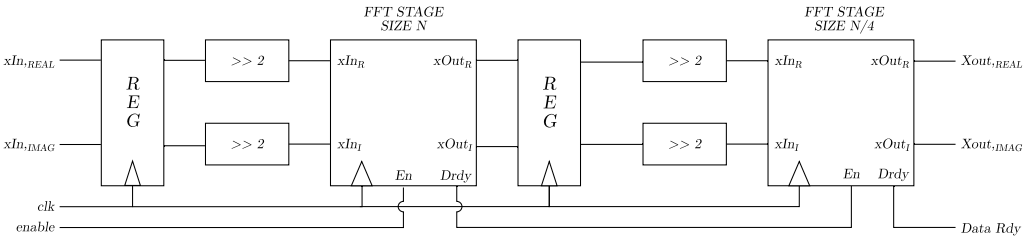


Figure 4.22: Interconnection between two FFT Stages.

Chapter 5

Test and validation

In this chapter, the set of tests performed on the developed hardware modules will be presented. From behavioral verification to real testing, in a FPGA or application specific integrated circuit (ASIC), a long time is required to properly verify the design under test (DUT). All the required steps were taken, for different implementation possibilities, to assess the actual constraints that must be met, in terms, for example, pipeline stages.

It should be noted, that in this section an additional requirement is added to the design. The initially assumed sampling frequency, for the OFDM modulator, as referred in 3.5.1, is 8MHz. To account for a possible interpolation factor[1] of upmost 10 times the baseband signals sampling frequency (i.e.: 80MHz), the required developed modules, must at least meet this value, in terms of maximum clock frequency. Usually, such requirements can be controlled by using multiple clock domains. However, due to the high precision required, for enabling ranging accuracy and improving the final design itself, the lack of usage of synchronization barriers is of the essence.

5.1 Behavioral verification

To properly test the developed modules, a testbench was written in Verilog-2001 to access their behavioral functionality. By employing the aforementioned written fixed-point models, different sets of input sequences were used as stimulus for the design under test (DUT), i.e.: the FFT and CORDIC modules. An input and output FIFO were also added to the design to emulate the expected behavior of the FFT, when tested in a FPGA, as will be explained further ahead. The data driven by the output FIFO was compared against the expected values, through assertions.

An example waveform dump can be found in figs. 5.1, 5.2 and 5.3, with all signals of interest, for an FFT of size 256, with a data-bus width of 16 bits. In particular, we can see the expected enabling interconnection between different stages, and the comparison between the output driven data and the expected result. At the same time, it can be verified that 256 assertions are performed, from 8.78 to 10.74 μs (in simulation time), for a clock signal with a period of $10ns$.

[1] Interpolation is a major requirement, in RF systems, to reduce ADCs and DACs harmonic distortion and facilitate the design of the low-pass analog filters.

The software used for compilation (and simulation) and waveform dumping were, respectively:

1. Icarus Verilog v10.1.1 [85];
2. GTKWave Analyzer v3.3.80 [86].

Figure 5.1: Example DUT testbench waveform dump.

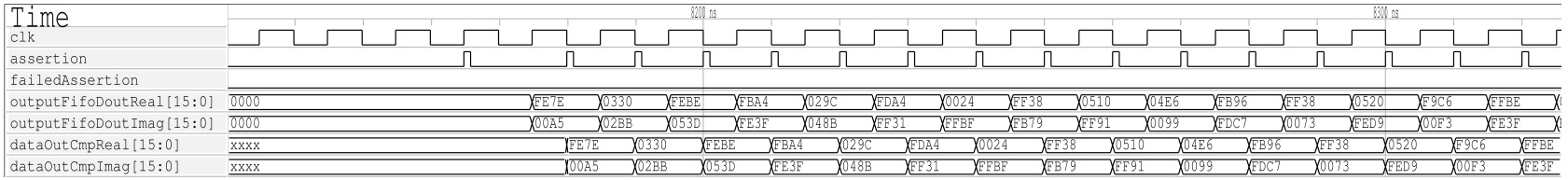


Figure 5.2: Testbench assertion example, comparing the DUT output to the expected values.

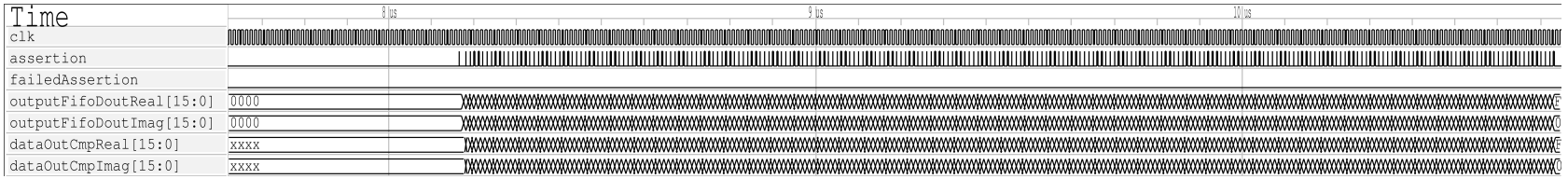


Figure 5.3: 256 assertions being performed on the testbench.

5.2 Coverage results

One of the most important metrics, for the validation of a true hardware design, is the coverage analysis. Besides explicitly demonstrating if a DUT is being properly stimulated, it can also enable the developer to find possible bugs in their coding, by demonstrating conditions that are never verified.

The basic functionalities a coverage tool is required to have are:

- Line Coverage: verifies if all lines of code were executed;
- Toggle Coverage: verifies if a bit value changes from 0 to 1 and back to 0;
- Finite State Machine (FSM) Coverage: verifies if all states of a FSM are achieved;
- Combinational Logic Coverage: evaluates if all possible combinations for setting a signal were met.

For the aforementioned purposes, Covered (v.0.7.10) [87], an open-source Verilog coverage tool, was used to verify the implementation of the designed testbench. We can find in table 5.1, a summary of the obtained results[2].

Table 5.1: Coverage report of the implemented DUT.

Line Coverage				**Combinational Logic Coverage**			
Hit	Miss	Total	Percent Hit	Hit	Miss	Total	Percent Hit
263	24	287	92%	411	151	562	73%

Toggle Coverage (0 to 1)				**Toggle Coverage (1 to 0)**			
Hit	Miss	Total	Percent Hit	Hit	Miss	Total	Percent Hit
1654	3164	4818	34%	654	3164	4818	34%

It should be noted that toggle coverage is usually among the hardest to meet. In our particular scenario, a 256-point FFT, with a 16-bit data-bus (per real and imaginary component), requires 2^{32} signal input combinations, for the stimulus only. If we take into account the size of the module, besides the combinations, 256! permutations are required, for a complete set of stimulus. This is obviously an over estimation, since different stimulus may result in equal outputs, but it is just to give an idea of how difficult it is to properly test a design.

5.3 Synthesis Results

To evaluate the area requirements of the implementation, synthesis was performed in XILINX^{TM} ISE (v14.7) [88], for a Spartan-6 family FPGA [89] XC6SLX45CG324C, for each individual module. On the ensuing tables, a report of the required logic, for different possible configurations, is shown.

Table 5.2: FFT radix-4 butterfly synthesis report.

Wordlength	**Max Combinational Path Delay (ns)**	**Slices-LUTS / (%)**
16	7.963	384 (1.41)
18	8.001	432 (1.58)
20	8.039	480 (1.76)

Table 5.3: CORDIC synthesis report.

Iterations / Pipeline Stages	**$F_{clk,MAX}$ (MHz)**	**Slice Regs (%)**	**Slice LUTS (%)**	**LUT-FF Pairs (%)**
16 / 4	74.486	335 (0.61)	1244 (4.56)	239 (17.83)
16 / 8	131.667	562 (1.03)	1600 (5.86)	459 (26.95)
18 / 6	93.142	481 (0.88)	1698 (6.22)	387 (21.60)
18 / 9	128.608	669 (1.23)	1992 (7.30)	567 (27.08)
20 / 5	72.252	479 (0.88)	1913 (7.01)	372 (18.47)
20 / 10	129.321	828 (1.52)	2461 (9.02)	707 (27.38)

Table 5.4: 16-bit data-bus radix-4 stage synthesis report with 4 CORDIC pipeline stages.

FFT Size	**$F_{clk,MAX}$ (MHz)**	**Slice Regs (%)**	**Slice LUTS (%)**	**LUT-FF Pairs (%)**
16	74.486	808 (1.48)	2069 (7.58)	421/2456 (17.14)
64	74.486	756 (1.39)	2263 (8.29)	616/2403 (25.51)
256	74.486	902 (1.65)	2361 (8.65)	521/2742 (19.00)

Table 5.5: 16-bit data-bus radix-4 stage synthesis report with 8 CORDIC pipeline stages.

FFT Size	**$F_{clk,MAX}$ (MHz)**	**Slice Regs (%)**	**Slice LUTS (%)**	**LUT-FF Pairs (%)**
16	131.667	1034 (1.89)	2495 (9.14)	680 (23.87)
64	131.667	982 (1.80)	2689 (9.85)	871 (31.11)
256	131.667	1128 (2.07)	2787(10.21)	777 (24.76)

As can be seen, like it was expected, the component which leads to the maximum combinational effort and required logic is the CORDIC unit. In fact, an increase of the input data-bus, almost does not affect the computational requirements of the butterfly operation. Furthermore, we can observe that to achieve the 80MHz clock frequency goal, and assuming SNR is not a requirement, it is almost preferable to use a 18-bit data-bus for the FFT, with 6 CORDIC pipeline stages than using the 16-bit configuration with 8 stages, just due to the memory logic requirements.

To further verify the complete area constraints, synthesis was performed to the complete DUT, under different configurations. A summary of the results obtained can be found on the ensuing tables[3]:

[2]FSM coverage results were not reported, since the DUT is free of any state machine.

[3]FFT Configuration refers to: FFT size / input data-bus / number of CORDIC pipeline stages / number of iterations performed per CORDIC pipeline stage.

Table 5.6: DUT synthesis results, for FFT sizes 16 and 64.

FFT Configuration	$F_{clk,MAX}$ (MHz)	Slice Regs (%)	Slice LUTS (%)	LUT-FF Pairs (%)
16 / 16 / 8 / 2	131.667	1452 / (2.66)	3116 / (11.42)	817 / (23.06)
64 / 16 / 4 / 4	72.737	1748 / (3.20)	4942 / (18.11)	1179 / (21.39)
64 / 16 / 9 / 2	131.321	2201 / (4.03)	5796 / (21.24)	1678 / (26.55)
64 / 18 / 6 / 3	93.142	2204 / (4.04)	6254 / (22.92)	1630 / (23.87)
64 / 18 / 9 / 2	128.608	2574 / (4.72)	6692 / (24.52)	2001 / (26.56)

Table 5.7: DUT synthesis results, for FFT size 256.

FFT Configuration	$F_{clk,MAX}$ (MHz)	Slice Regs (%)	Slice LUTS (%)	LUT-FF Pairs (%)
256 / 16 / 4 / 4	72.737	2628 / (4.82)	7287 / (26.70)	1690 / (20.55)
256 / 16 / 8 / 2	123.440	3274 / (6.00)	8569 / (31.40)	2437 / (25.91)
256 / 18 / 6 / 3	93.142	3282 / (6.01)	9225 / (33.81)	2355 / (23.20)
256 / 18 / 9 / 2	128.608	3833 / (7.02)	10282 / (37.68)	2609 / (23.28)
256 / 20 / 5 / 4	72.252	3426 / (6.28)	10266 / (37.62)	2362 / (20.85)
256 / 20 / 10 / 2	128.840	4481 / (8.21)	12231 / (44.82)	3428 / (25.81)

5.4 FPGA-in-the-Loop

The final step to ensure a proper hardware implementation, is the validation either in ASIC (which is an expensive procedure, for initial proof of concepts) or FPGA. To perform the latter, a script was written in MATLAB^{TM}, to create a closed simulation loop, according to fig. 5.4a, in what is usually known as *FPGA-in-the-Loop simulation*.

A top level Verilog module was developed, containing a RS232 (UART) serial port protocol implementation, two FIFOs (for combining the DUT input and output data, respectively), the implemented FFT, in the same configuration used as in the testbench (i.e.: size 256 with a 16-bit input data bus) and a FSM to control the state of the aforementioned blocks, as can be seen in fig. 5.4b. A Digilent Atlys development board [90] was used, which contains the aforementioned Spartan-6 FPGA embedded in a test platform with a USB to serial-port converter system.

The four FSM states are used to control the following subset of operations:

1. State 1 (*READ DATA*):
   - Receive four 8-bit samples (corresponding to the MSBs and LSBs of the real and imaginary input samples), though the serial port;
   - Concatenate the samples and write them to the input FIFO;
2. State 2 (*FILL FFT*):
   - Enable the read operation of the input FIFO;
   - Enable the DUT;

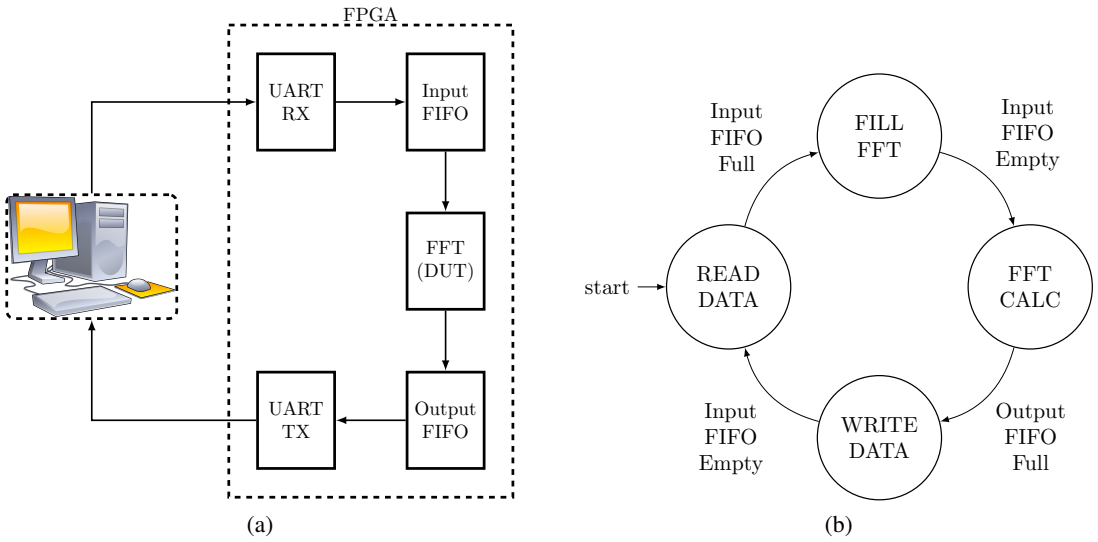


Figure 5.4: FPGA-in-the-Loop (a) block design and (b) corresponding finite state machine.

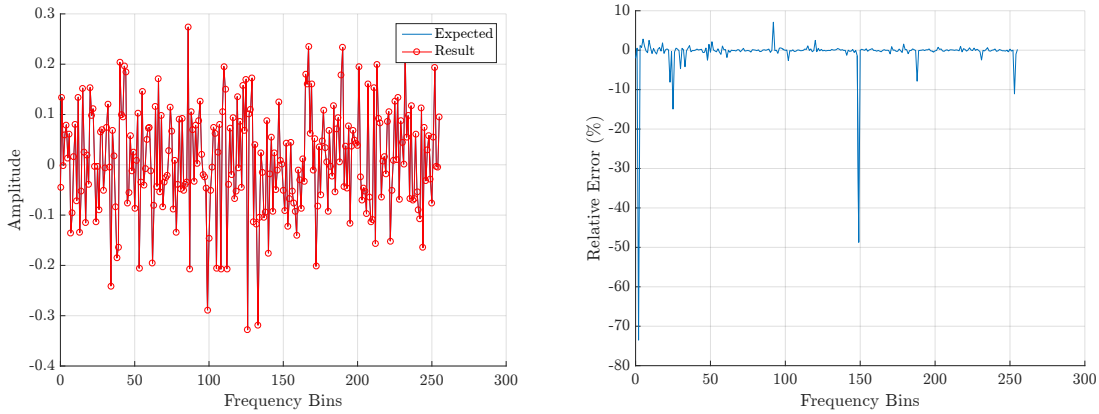
3. State 3: (*FFT CALC*):
   - Disable the read operation of the input FIFO;
   - Enable the write enabling signal of the output FIFO, when the DUT drives its *Data Ready* output;
4. Stage 4: (*WRITE DATA*):
   - Read data from the output FIFO;
   - Separate the read data into four 8-bit samples;
   - Transmit the data through the serial port to be read by the MATLAB^{TM} script.

After the script receives the resulting data, it was plotted against the expected (floating-point) result and the samples relative error computed. An example comparison can be found on the following figures: fig. 5.5a, 5.5b, 5.6a and 5.6b.

As can be seen, the expected value and the DUT output are in close concordance. The peaks in the relative error are expected behavior, due to the nature of fixed-point arithmetic. As a value tends towards zero, the number of bits required for the mantissa become insufficient to contain its representation, as can be seen in fig. 5.7a, with its corresponding relative error in fig. 5.7b.

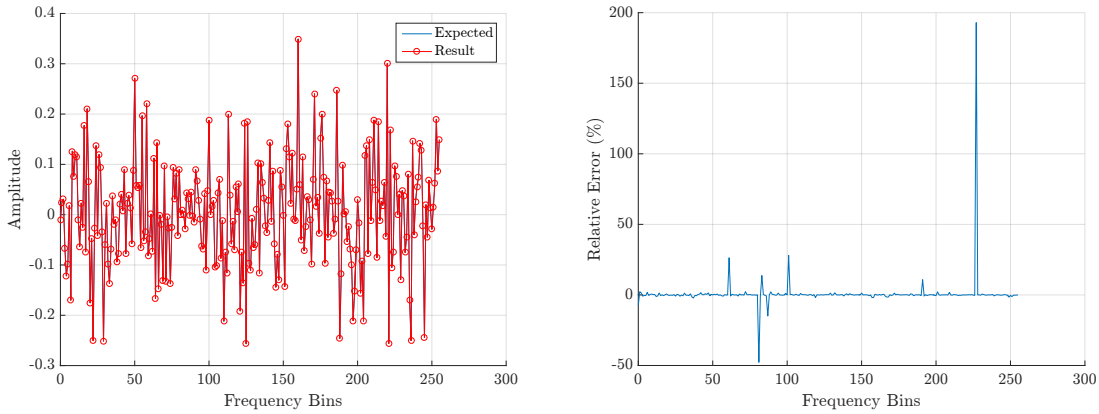
5.5 Summary

In this chapter, the required testing of the developed modules was demonstrated. As aforementioned, a clock frequency of 80MHz should be met. Since, the CORDIC unit, as expected, delivered the biggest trade-off in terms of area, we can see, that by selecting a proper pipeline size, a 256-point FFT with a 18-bit data-bus is achievable (with a maximum clock frequency of



(a) (b)

Figure 5.5: Comparison of the expected and DUT FFT (a) real component and (b) corresponding relative error.

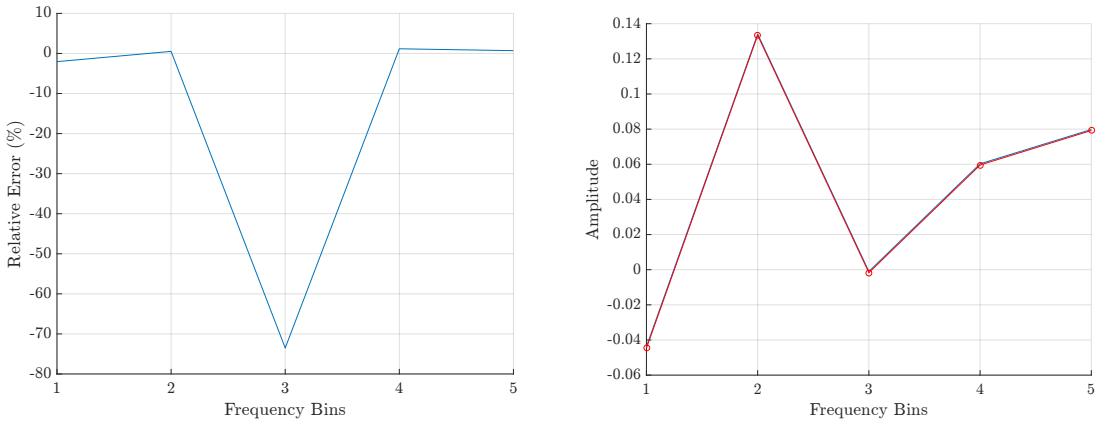


(a) (b)

Figure 5.6: Comparison of the expected and DUT FFT (a) imaginary component and (b) corresponding relative error.

93.142MHz), while still meeting a low area requirement. This equates to a maximum combinational effort, capable of achieving a throughput of 0.0284 Msamples/s/slice[4].

[4]This is the usual metric used for FFT implementations, in FPGA.



(a) (b)

Figure 5.7: DUT values tending (a) towards zero in the DUT and (b) its effect on the relative error.

Chapter 6

Conclusions

In this work, a novel architecture was proposed, with a proper analysis of the system requirements. By exploring the usage of OFDM, in TV White Spaces, wide coverage range can be achieved, while maintaining performance in multipath propagation environments. The SDS-TWR was proposed, for ranging applications, connected with a custom-made high-resolution timer. According to the Cramer-Rao Lower Bound, by performing multiple measurements, an highly accurate time estimate can be performed, enhancing the autonomous vineyard machinery location position evaluation. Despite the lack of knowledge of an expected channel delay spread, the selection of OFDM parameters, can now be seen as a straightforward approach.

Furthermore, a design of an area-efficient FFT with an embedded CORDIC was performed and demonstrated in FPGA, taking into account all possible constraints, and meeting the expected values of the developed fixed-point algorithms. The next steps to be taken, in order to complete the proposed architecture, can be found on the ensuing section.

It should be remarked that this work was partially presented in the Symposium on Electrical and Computer Engineering of the 2^{nd} Doctoral Congress in Engineering, held at the Faculty of Engineering University of Porto, at 8^{th} and 9^{th} of June, 2017.

In short, the main contributions of this work were:

1. The demonstration of a new concept for ranging applications;
2. The development of a custom FFT and IFFT solution;
3. The empirical study required to perform OFDM modulation.

6.1 Future Work

Since no work is ever really completed, there are still an ammount of points on the performed implementation, to be addressed. In terms of the FFT, the selection of an adequate wordlength which can still exhibit low area requirements, while keeping SNR bounded at expected values, is still missing. This selection is dependent on the remaining components of the physical layer, like the analog-to-digital and digital-to-analog converters, which can exhibit lower SNR values,

therefore becoming the design main bottleneck. Furthermore, a study of a possible radix-2^2 implementation could be done, mainly due to its advantages for mixed-radix designs (in case a reconfigurable FFT is required).

In terms of the OFDM modulator itself, a study of filtering and clippling techniques is required, to improve the high expected peak-to-average-average power ratio (PAPR). Furthermore, the implementation of a suitable preamble is of the essence, both for data transmission and ranging accuracy. Despite not demonstrated, a feasible possibility would be the usage of a Frank-Zadoff-Chu constant amplitude, zero auto-correlation, sequence. A channel estimation and channel equalizer system, should also be developed to further address multipath propagation.

As for the proposed architecture, a suitable study of the channel impulse response, in real vineyard scenarios, should be performed, to facilitate (and accommodate) the final design. Integration of the developed modules with an analog interface, for testing in a real outdoor scenario is expected to occur on the ensuing months.

Appendix A

Digit-Reversal

Table A.1: Bit-reversal for 8-Point Radix-2 FFT.

Input Address	Binary Representation	Bit Reversed	Output Address
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	6
7	111	111	7

Table A.2: Digit-reversal for 16-Point Radix-4 FFT.

Input Address	**Binary Representation**	**Bit Reversed**	**Output Address**
0	0000	0000	0
1	0001	0100	4
2	0010	1000	8
3	0011	1100	12
4	0100	0001	1
5	0101	0101	5
6	0110	1001	9
7	0111	1101	13
8	1000	0010	2
9	1001	0110	6
10	1010	1010	10
11	1011	1110	14
12	1100	0011	3
13	1101	0111	7
14	1110	1011	11
15	1111	1111	15

Appendix B

CORDIC

B.1 Procedures

Algorithm 1 Typical CORDIC initialization.

Require: x, y, θ: input coordinates and rotation angle.
Ensure: x_0, y_0, θ_0: input coordinates and rotation angle subject to CORDIC constraints.

```
function CORDIC INITIALIZATION(x, y, θ)
    if θ > π/2 then
        θ_0 ← θ − π/2
        x_0 ← −y
        y_0 ← x
    else if θ < −π/2 then
        θ_0 ← θ + π/2
        x_0 ← y
        y_0 ← −x
    else
        θ_0 ← θ
        x_0 ← x
        y_0 ← y
    end if
    return θ_0, x_0, y_0
end function
```

Algorithm 2 Typical CORDIC iteration.

Require: x, y, z, i: input coordinates, rotation angle and iteration number.
Ensure: x', y', z': output coordinates of the rotated vector, and missing rotation

```
function CORDIC ITERATION(x, y, z, i)
    if z > 0 then
        x' ← x − (y >> i)
        y' ← y + (x >> i)
        z' ← z − (1 >> i)
    else
        x' ← x + (y >> i)
        y' ← y − (x >> i)
        z' ← z + (1 >> i)
    end if
    return x', y', z'
end function
```

B.2 Example application

Table B.1: CORDIC input values, and corresponding relative error, as seen in fig. 4.11a.

Iter	X_i	X_i **Relative Error (%)**	Y_i	Y_i **Relative Error (%)**	Z_i	**Magnitude Error** (K)**++**
00	+0.27692	-132.64186	+0.96089	+81.50326	+1.29348	+1.00000
01	-0.68397	-42.99179	+1.23782	+65.32959	+0.50809	+0.70711
02	-1.30288	-2.87093	+0.89583	+7.02018	+0.04444	+0.63246
03	-1.52683	+10.42652	+0.57011	-33.92533	-0.20054	+0.61357
04	-1.45557	+4.45951	+0.76097	-12.48677	-0.07618	+0.60883
05	-1.40801	+0.84954	+0.85194	-2.21540	-0.01377	+0.60765
06	-1.38139	-1.10563	+0.89594	+2.78473	+0.01747	+0.60735
07	-1.39539	-0.11562	+0.87435	+0.29629	+0.00185	+0.60728
08	-1.40222	+0.37029	+0.86345	-0.95723	-0.00596	+0.60726
09	-1.39884	+0.12810	+0.86893	-0.32970	-0.00206	+0.60725
10	-1.39715	+0.00643	+0.87166	-0.01650	-0.00010	+0.60725
11	-1.39630	-0.05455	+0.87303	+0.13995	+0.00087	+0.60725
12	-1.39672	-0.02405	+0.87235	+0.06174	+0.00039	+0.60725
13	-1.39693	-0.00881	+0.87200	+0.02262	+0.00014	+0.60725
14	-1.39704	-0.00119	+0.87183	+0.00306	+0.00002	+0.60725
15	-1.39709	+0.00262	+0.87175	-0.00672	-0.00004	+0.60725
16	-1.39707	+0.00071	+0.87179	-0.00183	-0.00001	+0.60725

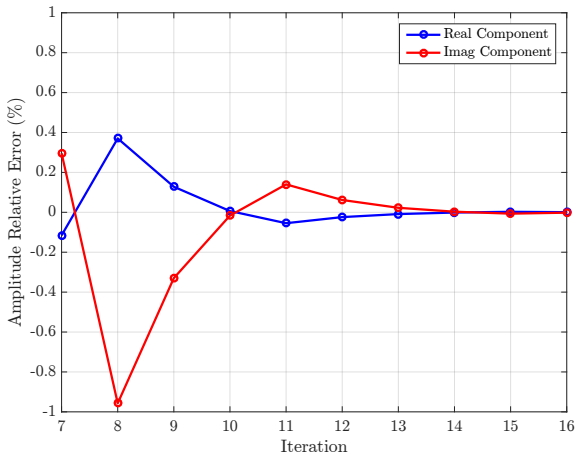
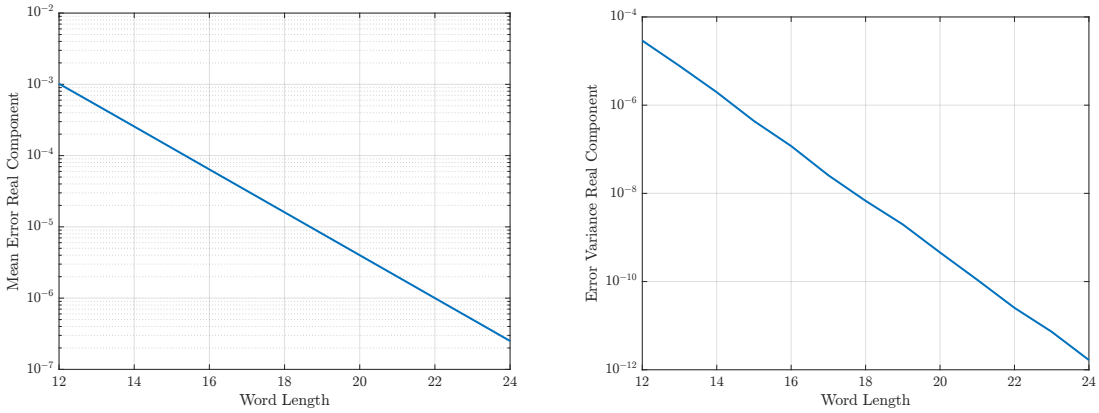


Figure B.1: CORDIC relative error during the last iterations.

Appendix C

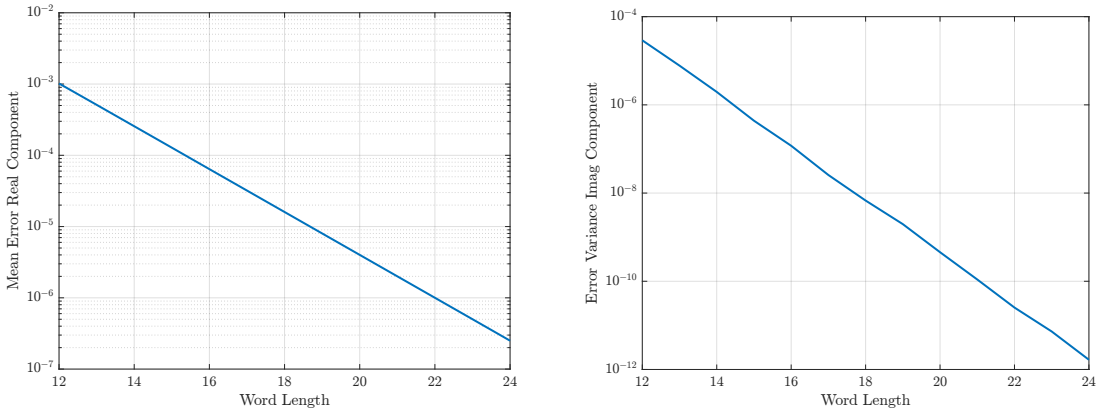
Statistical analysis

C.1 CORDIC



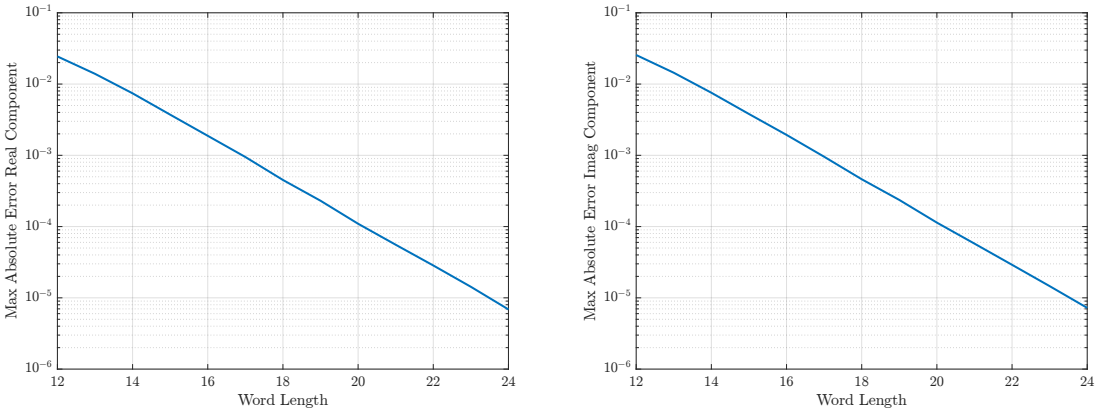
(a) (b)

Figure C.1: CORDIC mean error (a) and respective variance (b), for the real component.



(a) (b)

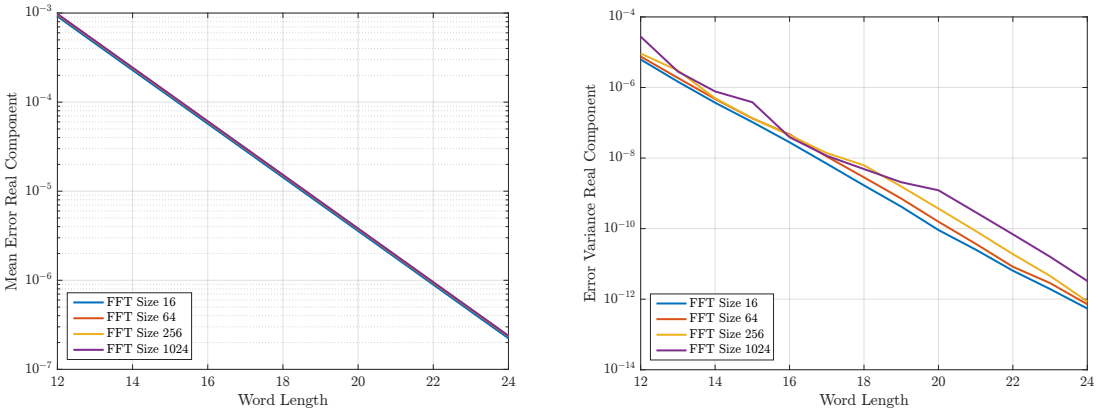
Figure C.2: CORDIC mean error (a) and respective variance (b), for the imaginary component.



(a) (b)

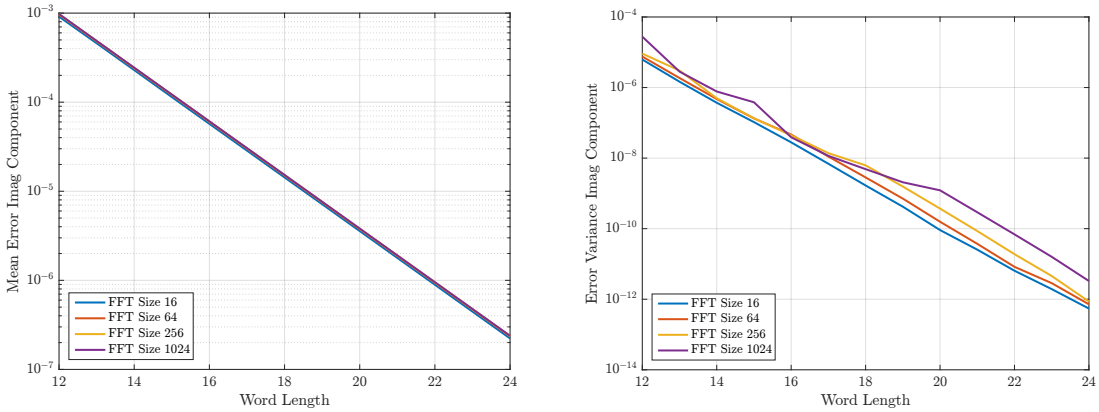
Figure C.3: CORDIC max absolute error for the real (a) and imaginary (b) component.

C.2 FFT



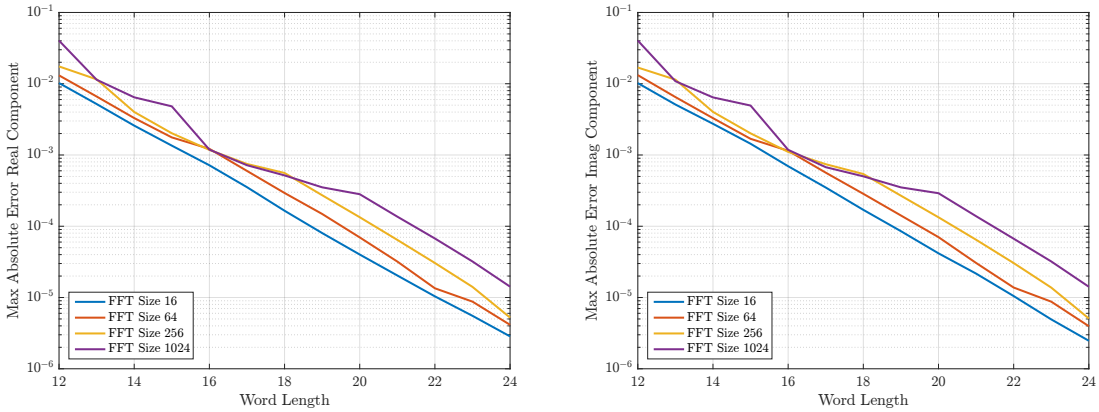
(a) (b)

Figure C.4: FFT mean error (a) and respective variance (b), for the real component.



(a) (b)

Figure C.5: FFT mean error (a) and respective variance (b), for the imaginary component.



(a) (b)

Figure C.6: FFT max absolute error for the real (a) and imaginary (b) component.

Bibliography

[1] IDTechEx, "Wireless Sensor Networks (WSN) 2014-2024: Forecasts, Technologies, Players," July 2015, Last Checked: June 2017. [Online]. Available: http://www.idtechex.com/research/reports/wireless-sensor-networks-wsn-2014-2024-forecasts-technologies-players-000382.asp

[2] H. Liu, H. Darabi, P. Barnejee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man and Cybernetics-Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, November 2007.

[3] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 1, pp. 13–32, First Quarter 2009.

[4] INESC TEC, "AGROB V1X - Robotics for Agriculture and Forestry | That works in steep slope viticulture," Last Checked: June 2017. [Online]. Available: http://agrob.inesctec.pt

[5] M. Duarte, F. N. dos Santos, A. Sousa, and R. Morais, "Agricultural Wireless Sensor Mapping for Robot Localization," in *Robot 2015: Second Iberian Robotics Conference: Advances in Robotics*, vol. 1, December 2016, pp. 359–370.

[6] F. N. dos Santos, H. Sobreira, D. Campos, R. Morais, A. P. Moreira, and O. Contente, "Towards a Reliable Monitoring Robot for Mountain Vineyards," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, April 2015, pp. 37–43.

[7] ——, "Towards a Reliable Monitoring Robot for Steep Slope Vineyards Monitoring," in *Journal of Intelligent and Robotic Systems*, February 2016, pp. 1–16.

[8] A. B. Flores, R. E. Guerra, E. W. Knightly, P. Ecclesine, and S. Pandey, "IEEE 802.11af: A standard for TV white space spectrum sharing," *IEEE Communications Magazine*, vol. 51, no. 10, pp. 92–100, October 2013.

[9] P. Daly, "Navstar GPS and GLONASS: global satellite navigation systems," *IET Electronics and Communications Engineering Journal*, vol. 5, no. 6, pp. 349–357, December 1993.

[10] J. J. S. Jr., P. Axelrad, and B. W. Parkinson, *Introduction And Heritage Of Navstar, The Global Positioning System - Global Positioning System: Theory and Applications*, 1st ed. American Institute of Aeronautics and Astronautics, January 1996, vol. Volume I.

[11] N. Patwari, J. N. Ash, S. Kyperountas, A. O. H. III, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, July 2005.

[12] M. A. Al-Ammar, S. Alhadhrami, A. Al-Salman, A. Alarifi, H. S. Al-Khalifa, A. Alnafessah, and M. Alsaleh, "Comparative survey of indoor positioning technologies, techniques, and algorithms," in *2014 International Conference on Cyberworlds*. IEEE, Oct 2014, pp. 245–252.

[13] P. Gulden, S. Roehr, and M. Christmann, "An overview of wireless local positioning system configurations," in *2009 IEEE MTT-S International Microwave Workshop on Wireless Sensing, Local Positioning, and RFID*, September 2009, pp. 1–4.

[14] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, C. Hoffmann, and P. Heide, "Wireless local positioning," *IEEE Microwave Magazine*, vol. 4, no. 4, pp. 77–86, December 2003.

[15] A. R. J. Ruiz and F. S. Granja, "Comparing Ubisense, BeSpoon, and DecaWave UWB Location Systems: Indoor Performance Analysis," in *IEEE Transactions on Instrumentation and Measurement*, vol. PP, no. 99, April 2017, pp. 1–12.

[16] M. Scherhaufl, M. Pichler, and A. Stelzer, "UHF RFID Localization Based on Evaluation of Backscattered Tag Signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 11, pp. 2889–2899, November 2015.

[17] G. Gomes and H. Sarmento, "Indoor Location System Using ZigBee Technology," *2009 Third International Conference on Sensor Technologies and Applications*, June 2009.

[18] D. Oh, Myungkyun, and J.-W. Chong, "A Subspace-Based Two-Way Ranging System Using a Chirp Spread Spectrum Modem, Robust to Frequency Offset," *IEEE Transactions on Wireless Communications*, vol. 11, no. 4, pp. 1478–1487, April 2012.

[19] P. Pivato, S. Dalpez, and D. Macii, "Performance Evaluation of Chirp Spread Spectrum Ranging for Indoor Embedded Navigation Systems," *7th IEEE International Symposium on Industrial Embedded Systems (SIES'12*, pp. 307–310, June 2012.

[20] H. Cho and S. W. Kim, "Mobile Robot Localization Using BiasedChirp-Spread-Spectrum Ranging," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 8, pp. 2826–2835, August 2010.

[21] B. Schaffer, G. Kalverkamp, and E. Biebl, "A 2.4 GHz high precision local positioning system based on cooperative roundtrip time of flight ranging," *GeMiC 2014; German Microwave Conference*, pp. 1–4, March 2014.

[22] C. Carlowitz, M. Vossiek, A. Strobel, and F. Ellinger, "Precise Ranging and Simultaneous High Speed Data Transfer Using mm-Wave Regenerative Active Backscatter Tags," *2013 International Conference on RFID*, pp. 253–260, April 2013.

[23] N. Joram, J. Wagner, A. Strobel, and F. Ellinger, "5.8 GHz Demonstration System for Evaluation of FMCW Ranging," *2012 9th Workshop on Positioning, Navigation and Communication*, pp. 137–141, March 2012.

[24] L. Marcaccioli, E. Sbarra, L. Urbani, R. V. Gatti, and R. Sorrentino, "An accurate indoor ranging system based on FMCW radar," *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 981–986, June 2011.

[25] T. Ussmueller, D. Brenk, J. Essel, J. Heidrich, G. Fischer, and R. Weigel, "Roundtrip-Time-of-Flight Based Localization of Passive Multi-Standard RFID-Tags," *2012 IEEE International Conference on Wireless Information Technology and Systems (ICWITS)*, pp. 1–4, November 2012.

[26] E. Staudinger and A. Dammann, "Round-Trip Delay Indoor Ranging Experiments with OFDM Signals," *ICC14 - W10: Workshop on Advances in Network Localization and Navigation*, pp. 150–156, June 2014.

[27] S. Konig, M. Schmidt, and C. Hoene, "Precise Time of Flight Measurements in IEEE 802.11 Networks by Cross-Correlating the Sampled Signal with a Continuous Barker Code," *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2010)*, pp. 642–649, November 2010.

[28] E. Staudinger and A. Dammann, "Sparse Subcarrier Allocation for Timing-based Ranging with OFDM Modulated Signals in Outdoor Environments," *2013 10th Workshop on Positioning, Navigation and Communication (WPNC)*, pp. 1–6, March 2013.

[29] R.-E.-A. Anee and N. C. Karmakar, "Chipless RFID Tag Localization," *IEEE Transaction on Microwave Theory and Techniques*, vol. 61, no. 11, pp. 4008–4017, November 2013.

[30] S. Lanzisera, D. T. Lin, and K. S. J. Pister, "RF time of flight ranging for wireless sensor network localization," in *2006 International Workshop on Intelligent Solutions in Embedded Systems*, June 2006, pp. 1–12.

[31] A. I. Baba and M. M. Atia, "Burst mode symmetric double sided two way ranging," in *2011 IFIP Wireless Days (WD)*, October 2011, pp. 1–3.

[32] S. Lanzisera and K. S. J. Pister, "Burst mode two-way ranging with cramér-rao boud noise performance," in *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, November, 2008, pp. 1–5.

[33] S. W. Oh, Y. Ma, M.-H. Tao, and E. Peh, *TV White Space The First Step Towards Better Uttilization of Frequency Spectrum*, 1st ed. IEEE Press, 2016.

[34] R. A. Saeed and S. J. Shellhammer, *TV White Space Spectrum Technologies Regulations, Standards and Applications*, 1st ed. CRC Press, 2012.

[35] J. van de Beek, J. Riihijarvi, A. Achtzhen, and P. Mahonen, "TV white space in europe," in *IEEE Transactions on Mobile Computing*, vol. 11, no. 2, February 2012, pp. 178–188.

[36] K. Harrison, S. M. Mishra, and A. Sahai, "How much white-space capacity is there?" in *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, April 2010, pp. 1–10.

[37] *Proposed Regulatory Framework for TV White Space Operations in the VHF/UHF Bands*, Info-Communications Development Authority of Singapore, 2013, IDA-June-2013.

[38] *Spectrum Policy Task Force Report*, Federal Communications Commission, November 2002, 02-135.

[39] *Second Report and Order and Memorandum Opinion and Order*, Federal Communications Commission, 2008, 08-260.

[40] *Second Memorandum Opinion and Order*, Federal Communications Commission, 2010, 10-174.

[41] *World Radiocommunication Conference, Geneva 2012*, International Telecommunication Union, 2012, ITU-WRC2012.

[42] K. Mazor and I. D. Lutilsky, "Spectrum pricing overview for digital dividend in europe," in *2015 57th International Symposium ELMAR (ELMAR)*, September 2015, pp. 53–56.

[43] A. K. Mucalo, K. Mazor, and I. D. Lutilsky, "Impacts and benefits of transition to DVB-T2 and assignment of second digital divided," in *2016 58th International Symposium ELMAR (ELMAR)*, September 2016, pp. 43–46.

[44] *World Radiocommunication Conference, Geneva 2015*, International Telecommunication Union, 2015, ITU-WRC2015.

[45] *White Space Devices (WSD); Wireless Access Systems operating in the 470 MHz to 790 MHz TV broadcast band*, European Telecommunications Standards Institute, April 2014, ETSI EN 301 598 (V1.1.1).

[46] *Digital dividend: cognitive access - Statement on licence-exempting cognitive devices using interleaved spectrum*, Office of Communications, 2009.

[47] *Technical and Operational Requirements for the Possible Operation of Cognitive Radio Systems in the White Spaces of the Frequency Band 470-790 MHz*, Electronics Communications Committee, 2011, ECC Report 159.

[48] *OFCOM Implementing TV White Spaces*, Office of Communications, June 2015.

[49] *OFCOM Implementing TV White Spaces - Annexes 1 to 12*, Office of Communications, February 2015.

[50] *Technical and operational requirements for the operation of white space devices under geo-location approach*, Electronics Communications Committee, 2013, ECC Report 186.

[51] O. Holland, "Some Are Born With White Space, Some Achieve White Space, and Some Have White Space Thrust Upon Them," *IEEE Transactions on Cognitive Communications and Networking*, vol. 2, no. 2, pp. 178–193, June 2016.

[52] M. Kloc, S. Riess, J. Brendel, S. Linz, M. Gardill, R. Weigel, and G. Fischer, "Let's Make Them Cognitive Cognitive Radio Technology Applied to Professional Wireless Microphone Systems," *IEEE Microwave Magazine*, vol. 17, no. 1, pp. 70–78, January 2016.

[53] C. Ghosh, S. Roy, and D. Cavalcanti, "Coexistence challenges for heterogeneous cognitive wireless networks in TV white spaces," *IEEE Wireless Communications*, vol. 18, no. 4, pp. 22–31, August 2011.

[54] E. Axxel, G. Leus, E. G. Larson, and H. V. Poor, "Spectrum Sensing for Cognitive Radio : State-of-the-Art and Recent Advances," *IEEE Signal Processing Magazine*, vol. 29, no. 3, pp. 101–116, 2012.

[55] *IEEE Standard for Information Technology - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 5: Televeision White Spaces (TVWS) Operation*, IEEE Computer Society, 2013, IEEE Std 802.11 af^{TM}-2013.

[56] *IEEE Standard for Information technology - Local and metropolitan area networks – Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs)*, IEEE Computer Society, 2003, IEEE Std 802.15.3 ^{TM}-2003.

[57] *IEEE Standard for Local and metropolitan area networks - Part 15.4: Low-Rate Wireless Personal Area Networks LR-WPANs*, IEEE Computer Society, 2011, IEEE Std 802.15.4 ^{TM}-2011.

[58] R. Jain, "Channel models - a tutorial," 2003, Last Checked: June 2017. [Online]. Available: http://www.cse.wustl.edu/~jain/cse574-08/ftp/channel_model_tutorial.pdf

[59] A. Goldsmith, *Wireless Communications*, 1st ed. Cambridge University Press, 2005.

[60] A. Meijerink, "On the physical interpretation of the saleh–valenzuela model and the definition of its power delay profiles," in *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 9, 2014, pp. 4780–4793.

[61] A. A. M. Saleh and R. A. Valenzuela, "A statistical model for indoor multipath propagation," in *IEEE Journal on Selected Areas in Communications*, vol. SAC-4, no. 2, February 1987, pp. 128–137.

[62] I. TGn, *IEEE P802.11 Wireless LANs: TGn Channel Models*, IEEE, January 2004, 802.11-03/940r1.

[63] ITU, *Guidelines for evaluation of radio transmission technologies for IMT-2000*, ITU, 1997, Recommendation M.1225.

[64] M. Nakagami, "The m-distribution – a general formula of intensity distribution of rapid fading," in *Statistic Methods of in Radio Wave Propagation*, P. Press, Ed., 1960, pp. 3–36.

[65] B. Farhang-Boroujeny, "Ofdm versus filter bank multicarrier: Development of broadband communication systems," *IEEE Signal Processing Magazine*, vol. 28, no. 92-112, p. 3, May 2011.

[66] B. Farhang-Boroujeny and H. Moradi, "OFDM Inspired Waveforms for 5G," in *IEEE Communications Surveys and Tutorials*, vol. 18, no. 4, Fourth Quarter 2016, pp. 2474–2492.

[67] A. R. S. Bahai and B. R. Saltzberg, *Multi-Carrier Digital Communications Theory and Applications of OFDM*, 1st ed., ser. Information Technology: Transmission, Processing, and Storage, J. K. Wolf, Ed. Kluwer Academic Publishers, 2002.

[68] R. van Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, 1st ed. Artech House, 2000.

[69] *IEEE Standard for Information Technology - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Computer Society, March 2012, IEEE Std 802.11^{TM}-2012.

[70] S. Wang and T.-W. Ahn, "MAC layer timestamping approach for emerging wireless sensor platform and communication architecture," US Patent Application 12/213,286, December 2009.

[71] G. Piccinni, G. Avitabile, and G. Coviello, "An improved technique based on zadoff-chu sequences for distance measurements," in *2016 IEEE Radio and Antenna Days of the Indian Ocean (RADIO)*, Octover 2016, pp. 1–2.

[72] S. Gezici, Z. Tian, G. B. Giannaki, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization viw ultra-wideband radios: A look at positioning aspects of future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, July 2005.

[73] A. Haghparast, T. E. Abrudan, and V. Koinuven, "OFDM ranging in multipath channel using time reversal method," in *2009 IEEE 10th Workshop on Signal Processing Advances in Wireless Communications*, June 2009, pp. 568–572.

[74] T. E. Abrudan, A. Haghparast, and V. Koinuven, "Time synchronization and ranging in OFDM systems using time-reversal," in *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 12, 2013, pp. 3276–3290.

[75] M. Failli and C. of the European Communities, *Digital Land Mobile Radio Communications. COST 207*, 1st ed., ser. Information Technologies and Sciences. Office for Official Publications of the European Communities, September 1989.

[76] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Prentice Hall, 1999.

[77] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, April 1965.

[78] J. W. Cooley, P. A. W. Lewis, and P. D. Welch, "Historical Notes on the Fast Fourier Transform," *IEEE Proceedings*, vol. 55, no. 10, pp. 1675–1677, October 1967.

[79] J. E. Volder, "The CORDIC Trigonometric Computing Technique," *IRE Transaction on Electronic Computers*, vol. EC-8, no. 3, pp. 330–334, September 1959.

[80] ——, "The Birth of CORDIC," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 25, no. 2, pp. 101–105, June 2000.

[81] T.-D. Chiueh and P.-Y. Tsai, *OFDM Baseband Receiver Design for Wireless Communications*, 1st ed. John Wiley and Sons (Asia) Pte Ltd, 2007.

[82] P. K. Meher, J. Valls, T.-B. Juang, K. Sridharan, and K. Maharatna, "50 Years of CORDIC: Algorithms, Architectures and Applications," *IEEE Transactions on Circuits and Systems -I: Regular Papers*, vol. 56, no. 9, pp. 1893–1907, September 2009.

[83] B. Lakshmi and A. S. Dhar, "CORDIC Architectures: A Survey," *VLSI Design*, vol. 2010, pp. 1–19, 2010.

[84] A. M. Despain, "Fourier Transform Using CORDIC Iterations," *IEEE Transactions on Computers*, vol. C-23, no. 10, pp. 993–1001, October 1974.

[85] A. Stevens, "Icarus verilog," Last Checked: June 2017. [Online]. Available: http://iverilog.icarus.com

[86] GTKWave, "GTKWave Analyzer," Last Checked: June 2017. [Online]. Available: http://gtkwave.sourceforge.net

[87] T. Williams, "Covered - Verilog Code Coverage Analyzer," Last Checked: June 2017. [Online]. Available: http://covered.sourceforge.net

[88] XILINX^{TM}, "ISE WebPACK Design Software," Last Checked: June 2017. [Online]. Available: https://www.xilinx.com/products/design-tools/ise-design-suite/ise-webpack.html

[89] ——, *Spartan-6 Family Overview*, XILINX^{TM}, October 2011, DS160 (v2.0).

[90] Digilent, *Atlys^{TM} Board Reference Manual*, Digilent, August 2013.