Imperial College London

POLITECNICO DI TORINO

MASTER OF SCIENCE THESIS

VHDL implementation of communication methods for the next-generation of wearable and implantable devices

Imperial College Supervisors:
Prof. Pantelis GEORGIOU
Mr Daryl Ma

Politecnico di Torino Supervisor:
Prof. Danilo DEMARCHI

Author:
Giuliana EMMOLO

Master of Science degree in

Electronics Engineering
Department of Electronics and Telecommunications

A.Y. 2020/2021

"Success is no accident. It is hard work, perseverance, learning, studying, sacrifice and most of all, love of what you are doing or learning to do"

Pele

POLITECNICO DI TORINO

Abstract

Electronics Engineering
Department of Electronics and Telecommunications

Master of Science in Electronics Engineering

VHDL implementation of communication methods for the next-generation of wearable and implantable devices

by Giuliana EMMOLO

Advancements in the field of wearable and implantable technologies have the potential to improve significantly the healthcare monitoring and the therapy efficiency, while reducing medical costs. To achieve a continuous, unobtrusive and local monitoring of physiological parameters, it is essential the development of highly miniaturized and energy efficient devices equipped with sensing, computing and wireless communication capabilities.
Recent research effort has raised the prospect of moving beyond the well-established concept of implantable and wearable systems, through the conceptualization of the "Smart Dust" in the field of human sensing. The tight resource-constraints of such systems generate many challenges, which can be addressed through new design approaches and optimization strategies, especially concerning size and power consumption.
The proposed thesis project aims to examine the feasibility of communication methods for the next-generation of wearable and implantable devices. In particular, the problem of the transmission of multiple sensed signals from the same node towards the wireless channel is addressed. One promising solution consists in the implementation of digital mixing as a method to combine multiple input signals into a single output stream. The innovative aspect of the work developed consists in the detailed investigation of the trade-offs in terms of power consumption versus error performances of digital mixing, to demonstrate its suitability for extremely resource-constrained applications. The main objective is the research of the sources of error and power consumption concerning the logical behavior, along with the study of their impact on the system performance. This is achieved with dedicated error and power analysis, both relying on the VHDL implementation of the method under discussion. Specifically, the study of the error performance is carried out with the aid of functional simulations, whereas the power analysis is performed through the synthesis of the digital circuit.
The wireless communication capabilities require to address also the issue of the choice of a suitable modulation technique for such devices. Due to the lack of research works in this field, the proposed project focuses on the study of suitable digital modulation techniques for wireless microsensor applications, where the major system level metric is energy-efficiency, with the objective of providing the means for the choice of a suitable modulation scheme for the class of devices under discussion. A preliminary literature review showcases that Frequency-Shift-Keying (FSK) and On-Off-Keying (OOK) are viable solutions.

The final bit-error-rate (BER) analysis, aided by MATLAB simulations, demonstrates that FSK performs better under Additive White Gaussian Noise (AWGN). Lastly, open issues are discussed for the development of the future work.

Acknowledgements

The completion of this undertaking could not have been possible without the participation and the assistance of many people.

First, I would like to express my deep sense of gratitude to Professor Danilo Demarchi for his precious help and constant support throughout the last year. Then, I would like to offer my special thanks to Professor Pantelis Georgiou for the opportunity to work and collaborate with Imperial College. The accomplishment of this project would not have been possible without his assistance. I would like to express also my sincere gratitude and warm appreciation to Daryl Ma, who guided and motivated me with great care, enthusiasm, and patience.

I would like also to thank Politecnico di Torino and in particular the Department of Electronics and Telecommunications (DET) for the excellent education provided and for the opportunity to join several projects during the MSc studies. My special thanks goes to Professor Maurizio Zamboni and to Professor Maria Grazia Graziano. The dedication and passion profused in their work motivated me to go in for my studies with great interest and enthusiasm.

The most deserved acknowledgments go to my family and firends, whose wholehearted support was essential for the completion of this objective.

Contents

List of Figures

List of Tables

List of Abbreviations

IoT	**I**nternet **o**f **T**hings
IoMT	**I**nternet **o**f **M**edical **T**hings
IC	**I**ntegrated **C**ircuit
CMOS	**C**omplementary **M**etal-**O**xide **S**emiconductor
MEMS	**M**icro-**E**lectro-**M**echanical **S**ystems
BMI	**B**rain **M**achine **I**nterface
NDM	**N**eural **D**ust **M**otes
EM	**E**lectro-**M**agnetic
US	**U**ltra **S**ound
ADC	**A**nalog-to-**D**igital **C**onverter
BC	**B**ackscatter **C**ommunication
FF	**F**lip **F**lop
RTL	**R**egister **T**ransfer **L**evel
VHDL (HSIC-HDL)	**V**ery **H**igh **S**peed **I**ntegrated **C**ircuit **H**ardware **D**escription **L**anguage
EDA	**E**lectronic **D**esign **A**utomation
TX	**T**ransmitter
RX	**R**eceiver
DUT	**D**evice **U**nder **Test**
GUI	**G**raphical **U**ser **I**nterface
FOM	**F**igure **O**f **Merit**
BER	**B**it **E**rror **R**ate
ASK	**A**mplitude **S**hift **K**eying
FSK	**F**requency **S**hift **K**eying
PSK	**P**hase **S**hift **K**eying
BASK	**B**inary **A**mplitude **S**hift **K**eying
OOK	**O**n-**O**ff **K**eying
BFSK	**B**inary **F**requency **S**hift **K**eying
BPSK	**B**inary **P**hase **S**hift **K**eying
QPSK	**Q**uadrature **P**hase **S**hift **K**eying
QAM	**Q**uadrature **A**mplitude **M**odulation
AWGN	**A**dditive **W**hite **G**aussian **N**oise
SNR	**S**ignal-to-**N**oise **R**atio
WSNs	**W**ireless **S**ensor **N**etwork**s**
CW	**C**ontinuous **W**ave
MEC	**M**inimum **E**nergy **C**oding
WNSNs	**W**ireless **N**ano-**S**ensor **N**etwork**s**

List of Symbols

f	frequency	Hz (s^{-1})
T	period	s
P	power	W ($J \cdot s^{-1}$)
V	voltage	V
C	capacitance	F ($C\,V^{-1}$)
C_{eff}	effective capacitance	F ($J\,s^{-1}$)
BW	bandwidth	Hz (s^{-1})
E	energy	J
N	power spectral density	J ($W \cdot s$)

Chapter 1

Introduction

The Internet of Things (IoT) is a revolutionary and fast-changing field that penetrates deeper and deeper into many spheres of modern society [1]. It can be defined as a global infrastructure that interconnects physical and virtual objects (also called “things”), thus developing advanced services [2] and solutions for a wide variety of applications in various fields [3].
The Heathcare Industry represents one of the prominent applications. The integration of the IoT with the available medical technologies has provided significant benefits to the patients resulting from the improved medical assistance, the minimum medical costs and treatment time, and from the most satisfactory health services [4]. A broad range of healthcare IoT applications exists today, ranging from mobile applications to wearable and implantable devices.
However, recent research effort has raised the prospect of moving beyond the well-established concept of implantable and wearable systems, through the conceptualization of the so called "Smart Dust" in the field of human sensing. This revolutionary idea has been conceived two decades ago, along with all the potential applications. From the early development stages, the idea of producing these autonomous sensing, computing and communication systems that create together a massive distributed sensor network, outlined the main design efforts, concerning size, power and communication capabilities [5]. The production and fabrication of Smart Dust motes of sub-mm dimensions for medical health-care has yet to be achieved. Nevertheless, there exist studies that demonstrate the feasibility of the concept close to the target requirement, although the near future research has still many challenges to face.
To achieve the possibility of turning these conceptual devices into actual applications for the medical field, new design approaches and optimization strategies in terms of area, power consumption and data transmission capability must be introduced.
The integration of wireless data transmission capability represents an obvious requirement to ensure telemetry properties. However, the tight size and power constraints of the design require to seek solutions for the implementation of energy efficient wireless communication methods, as the power consumed in a chip is directly proportional to the occupied area and volume. As a consequence, the resource limitations may considerably impact the system error performances.

1.1 Thesis Objectives

The proposed thesis project aims to examine the feasibility of communication methods for the next-generation of wearable and implantable devices, as well as extremely resource-constrained systems. In particular, the work developed addresses the problem of transmitting multiple sensed signals from the same node towards a wireless channel. This is achieved with a separate study of (1) a multiple input data acquisition and transmission method, and of (2) digital modulation techniques for wireless signal transmission, where in both the cases the major system level metrics are power and error performances. The core of (1) is the study of "digital mixing" as a method to combine multiple input signals into a single output stream. The innovative aspect of the work developed consists in the detailed investigation of the trade-offs in terms of power consumption versus error performances of digital mixing, in order to demonstrated its suitability for extremely resource-constrained applications. This is achieved through dedicated (I) error and (II) power analysis of the system.
In (2), a preliminary introduction to the concept of digital modulation and related Figures-Of-Merit provides some useful details for the selection of a subset of existing digital modulation techniques in the field of wireless microsensor applications, which can be seen as a "precursor" of the Body Dust project. A dedicated literature review demonstrates that frequency-Shift-Keying (FSK) and On-Off-Keying (OOK) can be considered desirable modulation schemes. The final bit-error-rate (BER) estimation allows to compare the selected modulation schemes in terms of performance.

1.2 Thesis overview

The proposed thesis project consists of eight chapters, that can be grouped into two subsets: the first one (Ch. 2, 3, 4, 5) deals with (1), whereas the second subset (Ch. 6) relates to (2). The complete list of chapters is listed in the following, along with a brief description of each:

- Chapter 2, which contains a literature review of the Body Dust concept.
- Chapter 3, which provides an insight into digital mixing, and discuss the first example of application.
- Chapter 4, which deals with the analysis of the error performances of digital mixing.
- Chapter 5, which deals with the analysis of the power consumption of digital mixing. It contains also the study of the power versus error trade-off for the design of the input frequency levels in digital mixing.
- Chapter 6, which provides a deep insight into the field of digital modulations and defines the major Figure-Of-Merits(FOM) for the choice of a suitable modulation scheme for a generic target application.
- Chapter 7, which aims to investigate the error performance versus power trade-offs of viable digital modulation techniques for wireless microsensors

applications. The final section of this chapter in particular is dedicated to the simulation of the BER in MATLAB of OOK and FSK.

- Chapter 8, which contains the final considerations and the discussion of the open issues for the development of the future work.

Chapter 2

Body Dust: next generation of wearable and implantable devices

2.1 The "Smart Dust" Project

The concept of "Smart Dust" has been conceived nearly 20 years ago, when the advancements and the fast convergence of key technologies, such as (1) digital circuitry, (2) wireless communications, and (3) Micro Electro-Mechanical Systems (MEMS), demonstrated the possibility to integrate sensing, communication, and power supply into an "inch-scale" device using only off-the-shelf technology [6]. In particular, the MEMS technology enabled the development of low cost and low power sensors, while the IC scaling process provided increasing functionalities in even smaller area, and wireless communication allowed for simultaneous collection of data from several sensors.
The Smart Dust project was developed at the University of California at Berkeley, and its goal was to "explore whether an autonomous sensing, computing, and communication system could be packed into a cubic-millimeter mote, which was the basic building block of integrated, massively distributed sensor networks" [7]. These "Networked Sensors" [8] were supposed to communicate with another or with a central base station to perform a large sensing task, thus enabling innovative methods to interact with the environment, where more information was provided less intrusively from several distributed sites. To this end, many potential applications were proposed, especially those in which wired sensors were unusable [9]. The key requirement, as well as the biggest challenge, was minimum energy consumption in such a small size while performing all these functions [10]. Carrara in [5] summarizes with a detailed analysis of several research projects the main milestones along the pathway of the Smart Dust project. In particular, he identifies two different stages of the research:

1. The early development stages, concerning the proposition of the project, and the first demonstration with a full functioning device.

2. The second research period, where most of the research activities focused on state-of-art technology, and where the key constrains were power ad area, sometimes without succeeding in reaching the original goal of a real fabrication of a "*cubic-millimiter*" sized mote.

However, the work of revision of the Smart Dust concept in [5] is conceived to rise a new perspective in the field of human sensing, paving the way for the conceptualization of the *next generation of implantable and wearable devices*. The next section

recalls the crucial aspects of the research work developed by Carrara in [5] concerning the application of the Smart Dust concept in the field of human sensing.

2.2 The concept of Smart Dust in the medical field

Even though the application to biology of the Smart Dust project has been conceived in the early development stages for the monitoring of the movement of small birds and insects [6], the research has prospected in more recent years the possibility to apply such concept to the field of human health-monitoring. The idea was to provide telemetry from inside the body through a distributed sensing active network consisting of thousands of dust motes spread in the human body. The research in this field deals in particular with two main branches:

1. Brain-Machine Interfaces, where the dust particles are often referred to as "Neural Dust".

2. Metabolism Monitoring, where the dust motes take the name of "Body Dust".

The Neural Dust was conceptualized earlier than the Body dust. The first research was published in 2013, by Jan M. Rabaey and Michel Maharbiz [11], who introduced the concept of Neural Dust as a "revolutionary" application in the field of brain monitoring. The Body Dust project, proposed a few years later by Sandro Carrara and Pantelis Georgiou, was conceived instead to provide a new paradigm for the monitoring of human metabolic conditions.

2.2.1 Application to the Brain-Machine Interfaces (BMI)

Neural Recording

The work developed by Jan M. Rabaey and Michel Maharbiz in 2013 [11] aimed to overcome one of the major obstacles in Brain-Machine Interfaces (BMI), as well as the "lack of an implantable neural interface system that remains viable for a lifetime", thus providing the path for a truly "chronic BMI". To achieve this goal, two technological innovations were required. The first one concerned the development of the neural dust motes, consisting of thousands of free-floating, autonomous, ultra-miniature devices, with lateral size of 10-100 µm, that could detect and transmit local electro-physiological data. The other technological innovation concerned the development of a "sub-cranial interrogator" that could send power via a ultrasound link, and receive data via the same ultra-sound channel through backscatter communication. The work developed by Rabaey and Maharbiz identified from the beginning the main challenges in this field. In particular, they investigated the ultimate limits in scaling size, power and bandwidth, as well as the relevant system trade-offs for the development of such devices. The choice of powering and communicating via ultra-sonic channel was preferred over the other existing methods to meet the size constraints of neural dust. However, even though the ultrasonic communication resulted feasible at that size scales in simulations, there were still some challenges to be faced such as the fabrication of a piezo-electric transducer at the same scale.
The following studies contributed significantly to the advancement of the research in this field [12],[13], [14], [15]. However, two main issues has still to be reasonably

proven. On one hand, the true feasibility of the concept at the sub-mm scale has yet to be demonstrated. Additionally, the demonstration of real applications of the neural dust motes (NMD) for neural monitoring has yet to be achieved.

Neural Stimulation

For what concerns neural stimulation, Cagnan et al.[16] proposed the concept of "ultrasound-linked neural dust" as a means to improve the specificity of implant while ensuring less invasiveness, using hybrid methods that combines implantable and wearable systems. The main disadvantages of the existing battery powered stimulators increased the size and weight of the implant, and required battery replacement or charging, while wired stimulators might cause infection [17], thus limiting the movement of stimulators within the tissues, or even leading to foreign body reactions [18]. However, similarly to the case of neural recording, the existing research work follows essentially a simulative approach to demonstrate the feasibility of the concept. Moreover, even if new wireless technologies have enabled the development of miniaturized neural stimulators, the real fabrication of such devices is still at the mm-scale [19].

2.2.2 Application to Metabolism Monitoring

S. Carrara and P. Georgiou[20] have prospected the possibility of developing a drinkable, remotely powered, diagnostic system, consisting of a multitude of autonomous small particles, each one with specific size constraints, that could provide "in-body" diagnostic telemetry, favouring a monitoring approach at the molecular level. They termed such system as "Body Dust". To this end, they analysed the theoretical feasibility of the first ever conceived sub-10 μm CMOS integrated circuit (IC) with bio-sensing capability, that once self-located in human tissues (i.e. a tumor mass), could transmit wirelessly the diagnostic information outside the body. This CMOS cube was the result of the integration in a highly compact integrated-circuit (IC) of (1) sensing capability of specific molecules, (2) data-telemetry properties, (3) reduced power consumption, and (4) bio-compatibility. Such tight size-constraints (sub-10 μm^3) were essential to exploit the mechanism of "internalization": after being drunk by the patient, the body dust particles should pass through the gut-wall barrier, and "free-circulate in human tissues". Even with the present state-of-the art CMOS technology, such size requirements seems to be currently unfeasible, so that there are still some doubts concerning the real viability of so small devices [5]

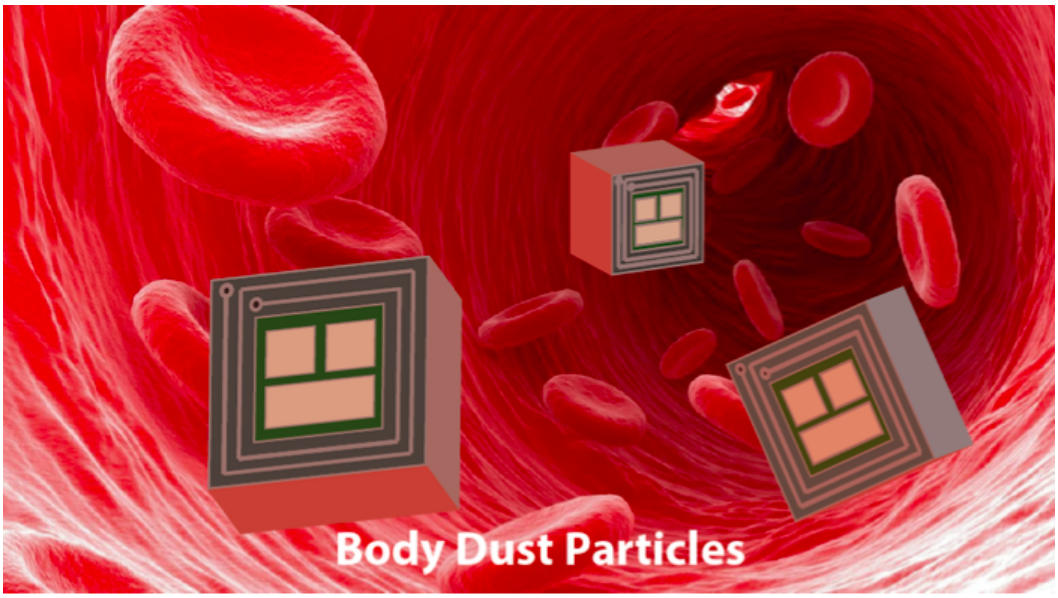


FIGURE 2.1: Illustration of concept of Body Dust particles for metabolism monitoring. The target size of these ultra-small particles is the blood cells size, in order to favour internalization process, so that they can self-identify the body-region source of the disease, and send wirelessly the sensed diagnostic data outside the body.

However, the existing research works demonstrates the feasability of the concept *close* to the target size and power constraints[5]. A first full circuit architecture for glucose amperometric biosensing in CMOS technology has been design by Jan Snoeijs, Sandro Carrara and Pantelis Georgiou [21]. Specifically, the work developed focused exclusively on the design of the architecture to interface to the biosensor and to the wireless transmitter. In particular, it took inspiration from a Google project published in 2012 [22], especially for some parts and features of the system conceived to gain space and efficiency. One of this was *Backscattering*, one of the most compact ways to wirelessly (1) power the system and (2) transmit data to an external device simultaneously [1] Another important feature was *Current to frequency conversion*, an efficient approach for encoding the sensed information to save area and power [2]. The results obtained from simulations demonstrated that the required chip area was of 10x10 μm^2, which is still too large considering that it refers to the biosensor front-end only.
The work developed a few years later by G. Barbruni et al. [26] focused on the study of the feasibility an ultra-sound (US) communication circuit for the body dust mote. The proposed architecture extended from the bio-sensor read-out front-end, which was assumed to be the one discussed in [21], up to the transmitter back-end. The main objective was to transmit the maximum number of source signals while assuring low power consumption and minimum occupation area. To implement the data acquisition and processing system an event-based approach, already

[1] It is important to remark that backscatter communication (BC) has emerged as a potential solution to solve the problems concerning the hardware complexity limitation [23], [24], [25].

[2] Current-to-frequency converters convert the analog data provided by the sensor into a digital transmittable information, without an explicit analog to digital converter (ADC)

discussed and demonstrated in [27], has been followed, while the low power transmission outside the body was achieved through a On-Off Keying (OOK) modulation and backscatter communication. The final simulation results demonstrate that the total chip area was 43x44 μm^2 and the power consumed by the chip was less than 10 μW.

2.2.3 Open Challenges

The near future research has to face several challenges to achieve the true realization of so small CMOS motes for body diagnosis at the molecular level. These challenges can be listed as follows:

1. design of extremely small bio-sensors, i.e. at the sub-micrometer (or nanometer) scale, and suitable bio/CMOS interfaces.
2. design of CMOS circuits and systems for ultra-low-power biosignal acquisition and processing, characterized by extremely small area;
3. design of powering methods feasible at extremely reduced sizes, which seems at the moment the most difficult task.
4. design of small and low power data links systems.
5. in-body demonstration, which remains actually the hardest challenge.

The proposed thesis project aims to contribute to the research in this field focusing on points 2. and 4. of the list above. In particular, the concept of digital mixing is proposed as an alternative approach to the event-driven digital architecture proposed in [26] for the acquisition and transmission of multiple input data. The second part of this project is dedicated to the analysis of the BER performances of OOK and FSK modulation techniques for the implementation of energy-efficient communication systems in the field of wireless microsensor applications, while attempting to extend the analysis to the body dust concept.

Chapter 3

Multiple Input, Single Output Frequency Mixing Communication Technique for Low Power Data Transmission

Digital mixers have been analysed in depth multiple times in literature. However, the concept of digital frequency mixing as a method to transmit multiple signals through a single output for low power wireless applications has not been analysed in great detail yet, especially for what concerns the analysis of the error generated in the output of the mixer because of the variation of the frequency of the input signals. In particular, this project offers a deep insight in the study of the simplest case of digital mixing, which involves two inputs only.
This chapter starts with a literature review of digital mixing, with a particular focus to the method used for the analysis in different research works. This is of great interest, as one innovative aspect of the proposed thesis project is the adoption of a peculiar test methodology used for the study of the error performance of digital mixing. Then, the concept of digital mixing as a method for multiple signal transmission is introduced, along with the first example of application. The latter in particular will be revised to introduce the concept of the error performance associated to the proposed method, which will be discussed in depth in the following chapter (Ch. 4).

3.1 Literature review of digital mixing

The concept of digital mixing and the study of the D flip-flop (D-FF) as a differential mixer has been discussed yet in literature in a variety of research activities along with the potential applications.
In 1969, K. P. Roby [28] demonstrated that when applying two input square-waves to the D and T inputs of a D-FF, the frequency of the output signal was the difference of the input frequencies, thus showing that the D-FF worked as a differential mixer. With this peculiar property, the D-FF was then employed for the design of measurement systems, for the detection of small frequency increments [29], [30], [31]. Bennet et al. [32] studied the logic behaviour of the D-FF using Fourier analysis to demonstrate its operation as a frequency changer, while Zhang et al. [33] studied the conditions in terms of input frequency ratio for which the D-FF can be used as a mixer, still by means of a Fourier analysis. Dong Jinfeng et. al [34]

studied in theory the output characteristics of the D-FF mixer and the output error performances [1] with a time-base analysis, and in experiments with test methodologies based on measurements against real hardware devices.
Zang et al. [35] studied in theory and experiments the behaviour of different logic gate mixer, when two quasi-digital signals are connected to the inputs, and the results showed that XOR and XNOR logic gate mixers performs better than the other logic gates.

3.2 Digital Mixing as a data transmission method for low power wireless communication

Figure 3.1 shows the conceptual architecture for the implementation of a communication method based on digital mixing.

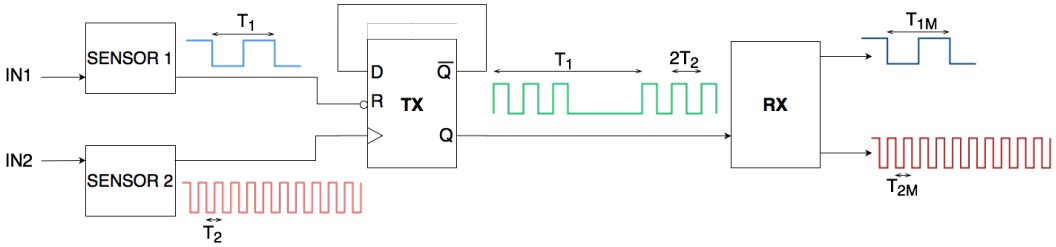


FIGURE 3.1: Architecture of a communication method based on digital mixing, which employs at the transmitter side a D flip-flop used in a divide-by-two counter setup with asynchronous active low reset, and at the receiver end a generic architecture that performs frequency measurement on the input mixed signal

The D flip-flop represents one of the possible circuit configuration for the signal mixing, and in this case it is used in a divide-by-two counter setup with asynchronous active-low reset. This simplifies the demodulation process at the receiver side, since it ensures that the Duty Cycle of the output signal is kept at the 50%.
The input signals are two square-waves produced by dedicated sensing nodes, and the information which is intended to be transmitted is the frequency of these time-continuous signals. Such signals are usually generated by the so called "quasi-digital sensors" and they have features of analog and digital signals at the same time [36].
With the mixing, the signal characterized by the highest frequency is conveyed to the clock port of the D flip-flop, while the one with the lowest frequency is fed into the reset port, and the signal generated by the mixer carries simultaneously the frequency information of both the inputs.
The two original frequencies will be then reconstructed at the receiver side, which performs essentially a frequency measurement on the input mixed signal. The specific method employed, as well as the design of the receiver architecture, will be chosen according to the target application.
The fact that the input signal information are simultaneously transmitted with a

[1] Here the expression "error performance" refers to the conditions in which the D-FF do not work as a differential mixer

single output stream allows for a true **continuous real-time monitoring** of the input sensed parameters: there is no need of switching in time between the different sources, as in the case of a typical multiplexed solution for example, where the inputs signal are selected for specific time slots. Additionally this data transmission method supports wireless power and data transfer as demonstrated [37]. However, the most attractive feature of such method lies in the **hardware simplicity** required for the circuit implementation.
All these features demonstrate that digital mixing is a promising solution for the transmission of multiple signals for extremely resource-constrained applications, such as the Body Dust.

3.3 First example of application: The "DAPPER"

The first application of this data transmission method can be found in the "DAPPER" designed by Ma et al. [37]. The system has been conceived is an analog front-end that performs concurrent potentiometric and amperometric sensing for the detection of bio-fluids. This system has been envisaged to be placed in the oral cavity for the continuous monitoring of salivary bio-molecules. The system will be roughly analysed in terms of its internal building blocks in the following, as it serves to introduce the concept of the error performance in digital mixing and the resulting constraints on the input frequency ratio, which will be studied in great detail in the following chapter.

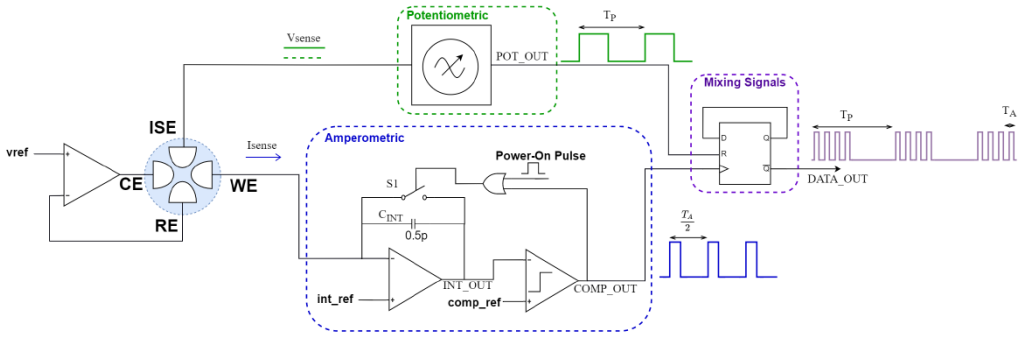


FIGURE 3.2: Block diagram of the Dapper

The block diagram of the DAPPER is shown in Figure 3.2. It consists of four main building blocks:

- Electrochemical cell, which consists of four electrodes, where in particular the working electrode (WE) is used for the amperometric reading, the ion-sensitive electrode (ISE) for the potentiometric sensing.
- Amperometric circuit, which transduces the information related to the input current into a frequency. The output signal is a square-wave with a 20% duty-cycle.
- Potentiometric circuit, which essentially converts the input voltage between RE and ISE into the frequency of a square-wave with a 50% duty cycle.

- D-FF digital mixer. The signal from the potentiometric circuit is fed into the reset port, whereas the one coming from the amperometric circuit goes into the clock port.

3.3.1 The constraint on the input frequency ratio

Ma et al. in [37] have precisely motivated the choice of the input frequency ranges, along with the constraints on the input frequency ratio, through the analysis of a possible situation arising with the signal mixing, which is reported in Fig. 3.3. A certain phase shift is assumed between the potentiometric signal and the amperometric one. Let t_a be the time when the potentiometric signal goes high, t_e the time when the mixed signal (the output of the D-FF) goes high, and t_b the point where the potentiometric output goes low.

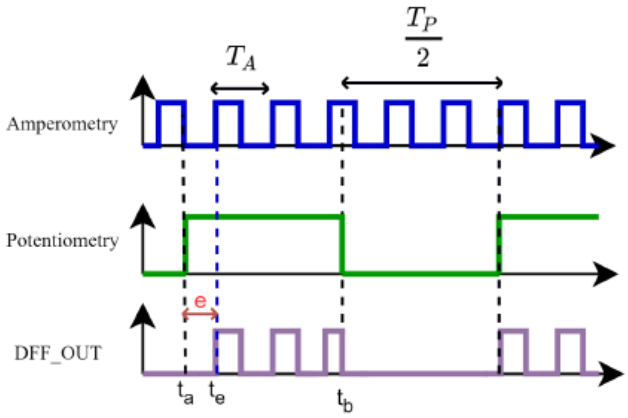


FIGURE 3.3: Timing Diagram of the signal mixing, where the amperometric and potentiometric signals are the inputs of the mixer. T_A is the period of the amperometric signal, T_P is the period of the potentiometric signal. "e" is the error generated in the mixed signal concerning the potentiometric period because of the input phase shift.

The reconstruction of the amperometric and potentiometric data in the mixed signal can be performed with the calculation of the edge-to-edge time difference in the mixed signal. To ensure the detection of at least two rising edges of the amperometric signal in the mixed signal, T_P has to be chosen to be at least 4 times greater than T_A. For the potentiometric signal, which is the one characterised by a lower frequency, an error arises in the mixed signal because of the phase shift with the signal mixing. This error can be computed as follows. The received period T_P can be expressed in terms of e according to the following equation:

$$\frac{T_{p,received}}{2} = t_b - t_e = \frac{T_p}{2} - e \tag{3.1}$$

where $T_{p,received}$ is the period of the potentiometric signal at the receiver side, while T_p is the actual potentiometric period, and e is the error generated on T_p in the mixed signal.
Considering:

$$t_e - t_a < T_A \tag{3.2}$$

And substituting Eq. 3.2 into Eq. 3.1, it results:

$$\frac{t_e - t_a}{T_p} < \frac{T_A}{T_p} \tag{3.3}$$

Eq. 3.3 allows us to deduce the maximum percentage error on the potentiometric period, given a certain input frequency ratio. In particular, if a maximum error of the 10% on $T_{p,received}$ is considered acceptable, it comes out that the amperometric frequency should be at least one order of magnitude (x10) larger than the potentiometric frequency. This initial results suggest that the smaller the ratio T_A/T_P the smaller the error generated in the mixed signal, which can be rephrased as "the greater the input frequency ratio f_A/f_P (or f_1/f_2 with reference to Fig. 3.1), the smaller the error associated with T_2 (and consequently with f_2) generated in the mixed signal".

Chapter 4

Characterization of Digital Mixing in terms of error performances

The aim of this chapter is the characterization of the error performance in digital mixing. This is achieved through the examination of the main sources of error, and through the characterization of their impact on the system performance. The main focus of the following analysis is the study of the ideal logical model of the system, which is by definition independent on the implementation/technology/-timing details, that usually are linked to the target application. This is of great importance, as it allows to create the basis for the full characterization of digital mixing.

4.1 Study of the error sources with preliminary time-base analysis

The first step consists in the identification of the sources of error in digital mixing through a preliminary time-base analysis of the output characteristics of the D flip-flop mixer.
As explained in Chapter 3, the information associated to each sensing node is embedded in the frequency (or period) of the square-waves they generate. In general, no synchronization between these nodes is assumed, and as a consequence there may exist a certain phase shift between the different square-waves.
With the signal mixing, the frequency information of the input signals are simultaneously present in the output of the D flip-flop. However, in some cases, an error is generated in the mixed signal, which causes the loss of information of the source signals.
Figure 4.1 shows one possible timing diagram of the signals involved in the mixing process. In this case, no phase shift between the input signals S_1 and $S_2$is assumed, and the input frequency ratio is equal to 7. T_2 coincides with T_{2_m}, while T_1 is equal to T_{1_m}, so no error is generated in the output signal for the information associated to both T_1 and T_2.
Suppose to change one of the input frequencies, e.g. f_2, while keeping constant the other one (i.e. f_1), and still assuming no phase shift among the input signals, as depicted in Fig.4.2. The only difference with respect to the case described in Fig. 4.1 is the input frequency ratio, which causes an error on T_2 in the mixed signal,

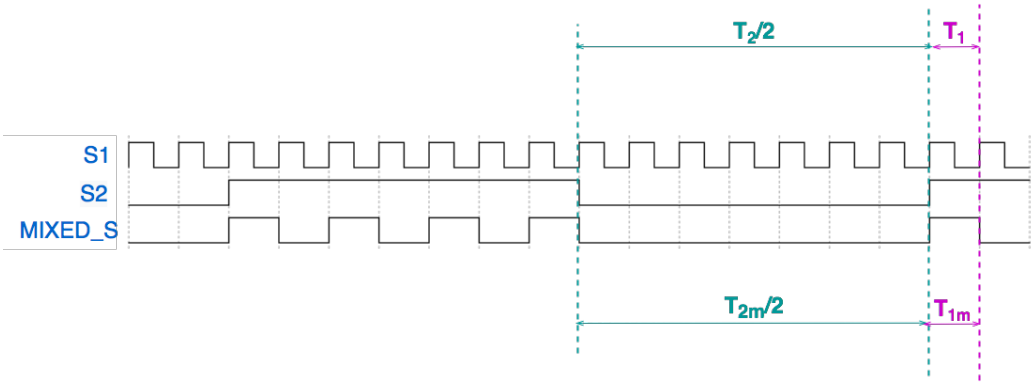


FIGURE 4.1: Timing diagram of the relevant signals of digital mixing. T_2 and T_{2_m} are respectively the period of the input signal with the lower frequency (i.e. S_2), which is fed into the reset port of the D flip-flop, and the period of the same signal in the mixed signal, (i.e. MIXED_S). Equivalently, T_1 and T_{1_m} represents respectively the actual period of S_1, which is fed into the clock port of the D flip-flop, and the period of S_1 in the mixed signal (MIXED_S). No phase shift is present, and for this specific input frequency ratio ($f_1/f_2 = 7$), no error is generated.

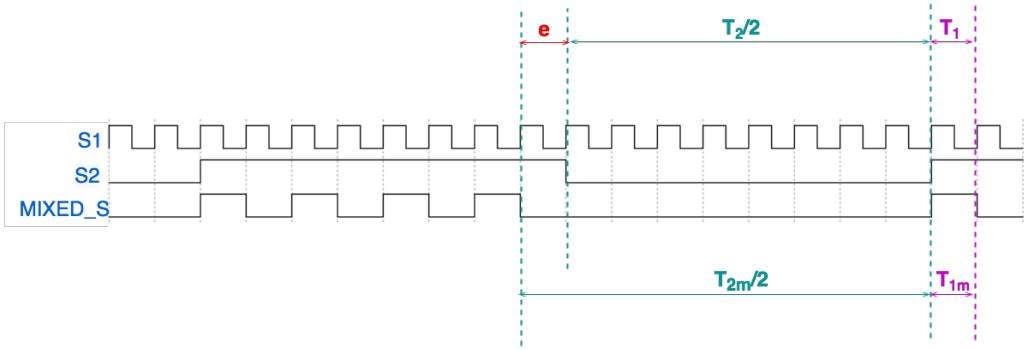


FIGURE 4.2: Timing diagram of the relevant signals of digital mixing. No phase shift is present and $f_1/f_2 = 8$. An error of the 12.5% is generated on T_2.

that can be expressed as follows:

$$e_{r_{T_{2\%}}} = \left| \frac{T_2 - T_{2_m}}{T_2} \right| \cdot 100 \tag{4.1}$$

In particular, $e_{r_{T_{2\%}}}$ is the relative percentage error on T_2, which is the period of the input signal with the lower frequency, and T_{2_m} is the information associated to T_2 in the mixed signal. For what concerns T_1, instead, no error arises in the mixed signal.
Considering that $\frac{f_1}{f_2} = 8$, it is possible to evaluate $e_{r_{T_{2\%}}}$ according to Eq.4.1 as follows:

$$e_{r_{T_{2\%}}} = \left| \frac{T_2 - T_{2_m}}{T_2} \right| \cdot 100 = \left| \frac{8 \cdot T_1 - 9 \cdot T_1}{8 \cdot T_1} \right| \cdot 100 = 12.5\% \tag{4.2}$$

Figure 4.3 showcases the timing diagram of the signals involved in digital mixing, where specifically $\frac{f_1}{f_2} = 7$, as in the case depicted in Fig.4.1, and $\Delta\phi$, which is the relative input phase shift between S_1 and S_2, is of the 50%[1].

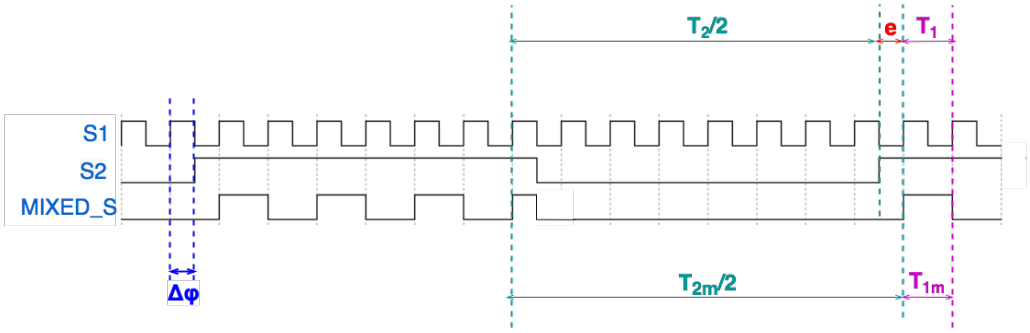


FIGURE 4.3: Timing diagram of the relevant signals of digital mixing. In this case, $\frac{f_1}{f_2} = 7$ and $\Delta\phi = 50\%$. An error is generated in the mixed signal for both the input periods.

In this case, errors are generated in the mixed signal for both the input periods. Fortunately, for S_1, which is the signal with the higher frequency, the larger amount of pulses compensates for the error on T_1 in the mixed signal. In contrast, for the lower frequency signal S_2 the period information is irrecoverably lost, because of the low amount of samples in the output of the D flip-flop.
The error on T_2 in the mixed signal can be computed following again Eq. 4.1 as:

$$e_{r_{T_{2\%}}} = \left| \frac{T_2 - T_{2_m}}{T_2} \right| \cdot 100 = \left| \frac{7 \cdot T_1 - 7.5 \cdot T_1}{7 \cdot T_1} \right| \cdot 100 \simeq 7\% \tag{4.3}$$

In Figure 4.4, $\frac{f_1}{f_2} = 8$, as in the case of 4.2, and $\Delta\phi = 50\%$. In this case, the error in the mixed signal arises only for T_2, which can be computed as:

$$e_{r_{T_{2\%}}} = \left| \frac{T_2 - T_{2_m}}{T_2} \right| \cdot 100 = \left| \frac{8 \cdot T_1 - 9 \cdot T_1}{8 \cdot T_1} \right| \cdot 100 = 12.5\% \tag{4.4}$$

[1] $\Delta\phi$ is expressed as a percentage of the period of S_1

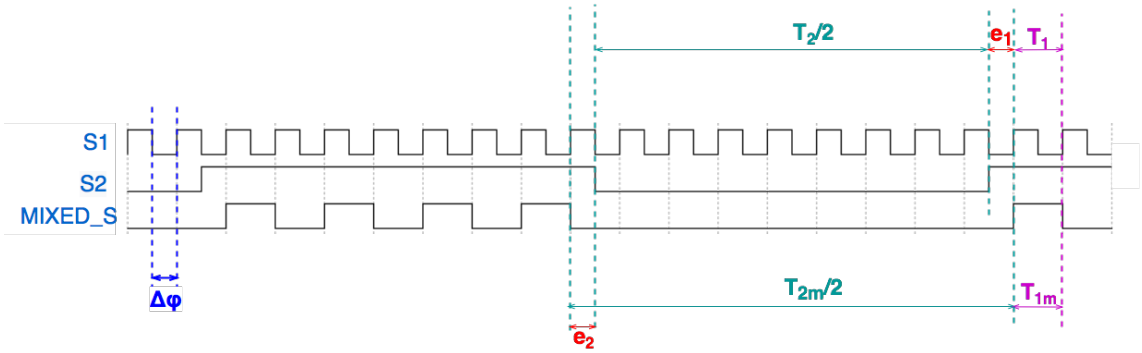


FIGURE 4.4: Timing diagram of the relevant signals of digital mixing. In this case, $\frac{f_1}{f_2} = 8$ and $\Delta\phi = 50\%$. An error is generated in the mixed signal for T_2 only.

Error on T_2 in the mixed signal		
f_1/f_2	$\Delta\phi$	$e_{r_{T_{2\%}}}$
7	0%	0%
8	0%	12.5%
7	50%	7%
8	50%	12.5%

TABLE 4.1

Table 4.1 summarises the cases described in Fig.4.1, 4.2, 4.3, 4.4, resulting form the time-base analysis of digital mixing. It suggests that the error on T_2 ($e_{r_{T_{2\%}}}$) in the mixed signal is the result of the **combination** of the frequency ratio of input signals (f_1/f_2), and their relative phase shift ($\Delta\phi$): for specific f_1/f_2, the variation of $\Delta\phi$ makes the error generated on T_2 vary as well, while for other f_1/f_2 it has no effect on $e_{r_{T_{2\%}}}$.

4.2 The Test Method

To achieve a complete characterization of the error performances, which takes into account different combinations of input frequency ratios and relative phase shifts, and to analyse and quantify the effects of the error sources, the approach exclusively based on a time-base analysis can be time-consuming, especially when accuracy is required.
One of the innovative aspect of the proposed thesis project consists in the definition and use of a peculiar test methodology based on **automated functional simulations** for the characterization of the error performance in digital mixing, which results in a rather effective and fast approach.
Functional simulations are traditionally performed to check the design functionalities [38] of digital circuits. In this case, they are rather employed to predict the logical circuit behaviour, in order to study the variation of the error generated in the

mixed signal when the input frequencies and relative phase shift vary. Additionally, functional simulations do not take into account any timing/technology/implementation details, since the HDL description of the circuit is not synthesized yet [39], which is of great interest for this analysis, as it focuses on the study of the error performance of digital mixing regardless the technological issues and their error contribution.
The entire test method for the characterization can be subdivided into two main parts:

1. **Simulation Stage**, whose main objective consists in testing the device in different possible combinations of f_1, f_2, and $\Delta\phi$.

2. **Processing Stage**, which handles with the results collected at the simulation stage to examine in great detail the effect of the error sources on the error generated in the mixed signal.

The key feature is **automation**: on one hand it allows to run *automatically* and *iteratively* several simulations, where a large amount of data is generated; on the other hand it permits to manage and process very efficiently and rapidly all these data, thus optimizing significantly the time required for the entire test process while assuring a high accuracy.

4.2.1 The Testbench

Functional simulations are based on the development of a Testbench, which is HDL code that is used to provide a documented, repeatable set of stimuli to the Device-Under-Test (DUT), and to check the system response.
The Testbench used for the characterization of digital mixing depicted in Figure 4.5 is conceptually structured in three main parts:

1. **Stimuli generation**. It is to remember that the inputs and the output of the digital mixer are "quasi-digital" signals, as specified in Chapter 3. So the stimuli generated in the Testbench are two square-waves with different frequencies and with a specific relative phase shift, produced by the VHDL code.

2. **DUT**, which consists of two sub-blocks, that are (1) the mixer, and (2) an additional block. The latter converts the informative parameter in the mixed signal produced by the D flip-flop (i.e. frequency or period) into a numerical quantity, and it is connected to the mixer in master-slave configuration, where the mixer works as a Transmitter (TX) and the additional block as a Receiver (RX).

3. **System Response**, which is the actual mixed signal[2], while the output of the receiver represents the numerical information associated to the mixed signal, and its content is stored during simulations into *text files* to be then processed. This is managed through a VHDL process, which writes upon event the numerical information produced by the receiver into the file. Additionally, the system response can be inspected graphically through the *waveform viewer*, as the simulator calculates and plots the corresponding waveforms.

[2]The device to be fully inspected is the mixer only, while the receiver represents an ideal hardware support for the analysis based on VHDL functional simulation.

The test-bench and the RTL (register transfer level) description of the DUT are implemented in VHDL and simulated with Modelsim, from Mentor Graphics. The system is tested for different combinations of f_1, f_2, and $\Delta\phi$, and for each of them a text file is generated.

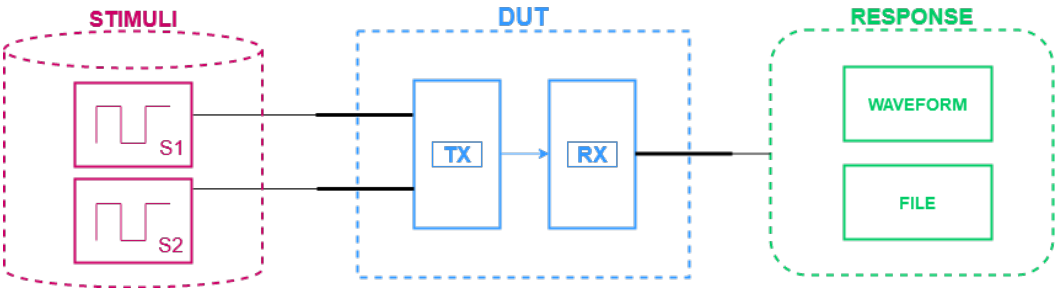


FIGURE 4.5: Conceptual structure of the Testbench for the test of the error performance in digital mixing. The input stimuli are applied to the DUT, consisting of a TX (mixer) and a RX (additional block for the time-to-code conversion), and they are quasi-digital signals. The numerical information produced by the receiver related the mixed signal are written into a text file. The waveform viewer allows for a visual inspection of the signal involved in the test-bench.

4.2.2 *Scripting* for test automation

In general, the higher the number of combinations of f_1, f_2, and $\Delta\phi$ considered, the higher the accuracy during the processing stage. Therefore, to assure a good accuracy for the analysis, it may be necessary to run many simulations, and doing it *manually* can be extremely onerous in terms of time. To speed-up the process, it is possible to automatize the test procedure with the aid of **scripting languages**, such as *python* and *bash*. It essentially serves to work with the Electronic Design Automation (EDA) tools in **batch mode**. It is convenient to analyse the Simulation stage and the Processing stage separately to appreciate the efficiency of *scripting*.

Automation in the Simulation stage - *bash, Tcl*

Figure 4.6 demonstrates the flow of the automation process for the Simulation stage, where the relevant steps are modelled by specific files. "*Conf. Script*" is a bash script (*.sh*) that performs *in loop* the following steps:

1. **Modify** the values of f_1, f_2, and $\Delta\phi$ in the *Testbench* file (*.vhd*).
2. **Launch** the "*Run Script*" (*.sh*), which is a bash script file that is used to configure and run the VHDL simulator (i.e. Modelsim) under UNIX environment, thus allowing to work in *batch mode* without the GUI (Graphical User Interface).

The "*Run Script*" in turns launch the Tcl (.tcl) file "*Sim Script*", which is used to define the operation required for the simulation[3]. This script, in particular, uses

[3]Many HDL simulator tools, such as Modelsim, support Tcl script language as an application programming interface, enabling control of the simulation using Tcl commands [40]

the Testbench with the modified parameters (f_1, f_2, and $\Delta\phi$). The codes of "*Conf. Script*", "*Run Script*", and "*Sim Script*" are reported in Appendix A to clarify the flow of the operations required for the execution the entire simulation process.
As previously anticipated, during each simulation the results are stored into text files (*.txt*). The outcome of the entire process is a set of text files, each corresponding to a combination of f_1, f_2, and $\Delta\phi$. The total number of simulations is defined in "*Conf. Script*" through the depth of the loop.

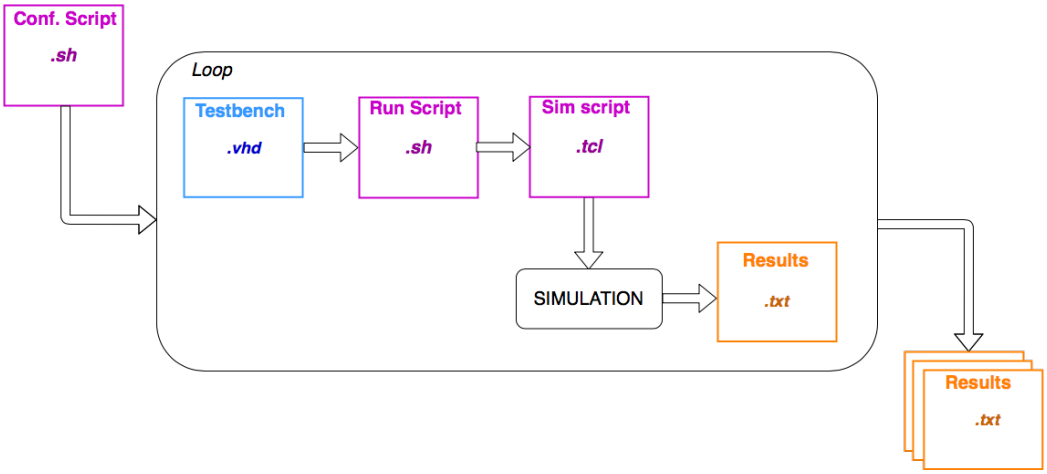


FIGURE 4.6: Conceptual representation of the flow of the entire simulation stage with the relevant files involved in the automation process.

In the following it is reported an example of the format of the *Results* file:

```
1265140  10370
1265140  10370
1265140  10370
1265140  10370
1265140  10370
1254560  10370
1253730  10370
1252900  10370
1252070  10370
1251240  10370
1250410  10370
1259950  10370
1265140  10370
1265140  10370
1265140  10370
1265140  10370
1265140  10370
1265140  10370
1254140  10370
1253310  10370
```

The first column refers to the numerical information associated to T_2 measured in the mixed signal, the second column represents the numerical information associated to T_1. The number of rows corresponds to the total number of samples collected in each simulation and it is related to the simulation time configured in the *Run Script*.

Automation in the Processing stage - *python*

Figure 4.7 demonstrates the flow of the automation process for the Processing stage. The main scripts here are Python scripts (*.py*), which are used to extract the results of the text files form the Simulation stage and generate some plots, showing the trend of the error as a function of the error sources, that are $\frac{f_1}{f_2}$ and $\Delta\phi$. Based on the way the data are extracted different analysis can be carried out.

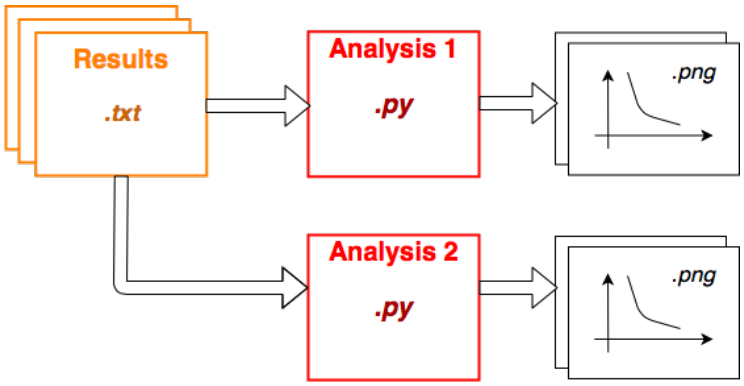


FIGURE 4.7: Conceptual representation of the flow of the processing stage for the results from simulations with the relevant files involved in the automation process.

4.3 Design of the Receiver Architecture

As previously specified, this additional block serves to perform the *time-to-code* conversion in the mixed signal, in order to provide a useful numerical information to be used during the processing stage for the error calculation. It is to remark that the main objective of the test methodology is the error characterization of the mixer, and for this reason the main requirements for the design of the receiver are:

- High Accuracy
- Hardware Simplicity

Accuracy is crucial, because a significant error contribution to the measurement produced by the receiver would make the results inconsistent for analysis. Additionally, it is fundamental to keep the hardware complexity of the receiver low in order to speed-up the simulation process: this additional block has to be verified before being used in the Testbench, and it is well known that the design complexity influences the time required for the verification of the design itself.

4.3.1 Internal blocks of the receiver architecture

The method exploited for the measurement of the informative parameter in the mixed signal is *the edge-to-edge time difference to code conversion*, which means that the receiver acts essentially as a **pulse-length counter**. This method is often referred to as "indirect frequency counting method" in literature [36].
Figure 4.8 showcases the top level view architecture of the receiver. It consists of four main building blocks, listed in the following:

1. Synchronizer
2. Edge Detector
3. Fast-Frequency Detector
4. Slow-Frequency Detector

All these blocks are synchronous circuits and they share the same reference clock signal with frequency f_{ck}. The input is the quasi-digital mixed signal, while the output signals are two binary codes, that represents the period information of the sources extracted from the mixed signal.

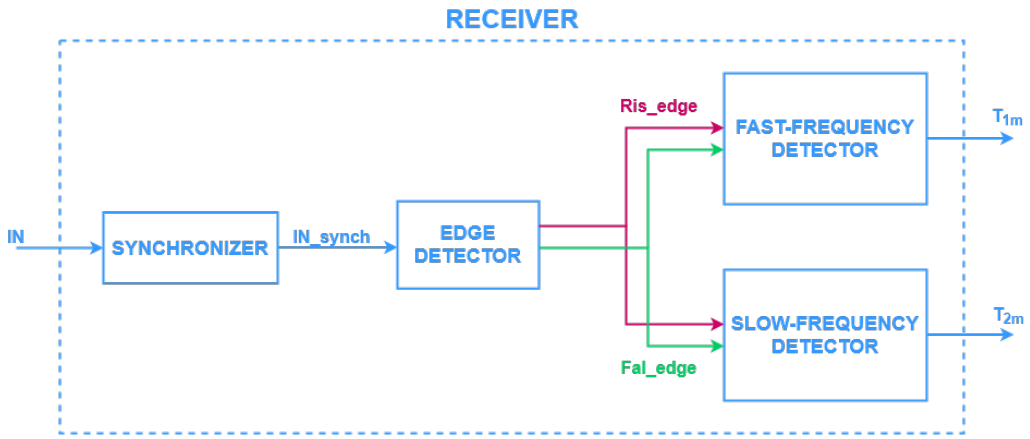


FIGURE 4.8: Top level view of the architecture of the receiver, which consists of four building blocks: (1) synchronizer, (2) edge-detector, (3) fast-frequency-detector, and (4) slow-frequency-detecor.

Synchronizer

In general, the role of the synchronizer consists in (1) receiving the input signals arriving at arbitrary times with respect to its system clock domain, and (2) minimizing the probability transition of the output signal within the setup-and-hold window of the receiving latch or flip-flop [41], which means reducing the problem of metastability. In this analysis, the main focus is (1), as the design of the receiver serves essentially for the characterization of the mixer, rather than for a true application. For this reason the problem of metastability is temporarily neglected.
The synchronizer (Fig. 4.9) drives the input signal (D) into the timing domain of the system clock (Clk). It essentlially provides the synchronous notification of the incoming asynchronous edges, when producing the output Q.

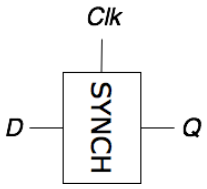


FIGURE 4.9: Relevant signals of the synchronizer

The main requirement is to prevent data from being lost during the acquisition of the input signal. This means that the following equation must be satisfied [42]:

$$\frac{1}{f_{ck}} = T_{ck} < min\{T_{IN}\} \tag{4.5}$$

where f_{ck} is the clock frequency of the whole receiver, and $min\{T_{IN}\}$ is the minimum pulse of the asynchronous input signal. This condition ensures that the input signal is sampled before any combinational operation is carried out, so that there is no risk to loose the edge detection, performed by the downstream block.

Edge Detector

The RTL view of the edge-detector is depicted in Figure 4.10. It consists of (1) a flip flop and of (2) some additional combinational logic, as well as two pairs of AND gates and inverters. The first flip-flop stores the state of the signal at the last rising clock edge, and compares it to the value of the signal at the current clock edge, while the additional combinational logic is used to detect the low-to-high transition and viceversa. If the state change matches one of the conditions, a "flag" (falling_edge/rising_edge), which is essentially a pulse synchronous with the internal clock, is generated. The pulse width of this flag is equal to the internal clock period, as it is shown in Fig. 4.11.

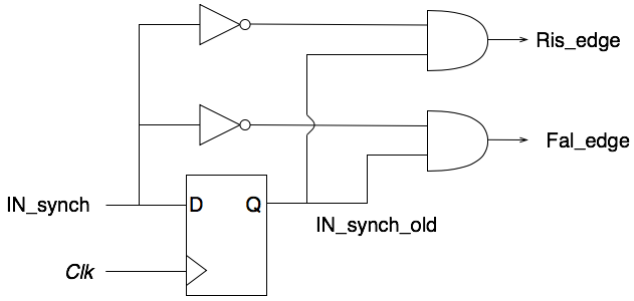


FIGURE 4.10: RTL view of the edge detector.

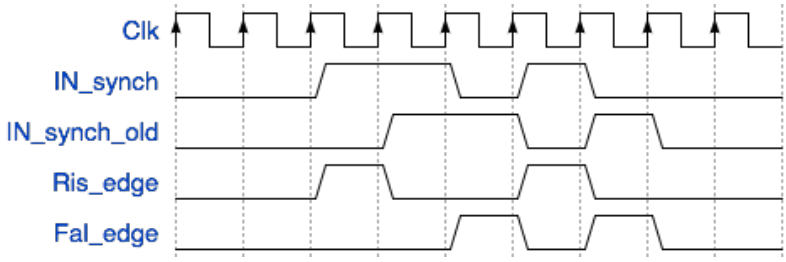


FIGURE 4.11: Example of timing diagram with all the signal involved in the edge detection. IN_synch is the output of the synchronizer. IN_synch_old represents the state of IN_synch at the previous clock cycle. Ris_edge and Fal_edge are the actual output signals of the edge detector, and they are active when either a rising or a falling edge is detected.

4.3.2 Fast-frequency detector and Slow-frequency detector

As shown in Figure 4.8, the "Ris_edge" and "Fal_edge" signals are conveyed to the Fast-frequency detector and to the Slow-frequency detector, which act respectively as a high-pulse counter and a low-pulse counter. In particular, the "Ris_edge" signal enables the fast-frequency detector, while the "Fal_edge" signal disables and then reset it synchronously for next conversion. This behaviour is similar for the slow-frequency detector, which is instead a low-pulse counter: the "Fal_edge" signal enables the counter, while the "Ris_edge" signal disables and reset it. The use of two dedicated counters allows the simultaneous measurement of the two periods within the mixed signal.

4.4 Analysis of the Accuracy of the Receiver

As previously anticipated, the main requirement for the design of the receiver is to ensure a good accuracy in the time-to-code conversion. This section deals with the analysis of the main sources of error concerning such conversion, while outlining the means to improve accuracy. This is achieved considering the error contribution arising in the whole conversion chain (Fig. 4.8).

4.4.1 Synchronization stage

Figure 4.12 shows the error arising from the synchronization stage. "IN" is the input asynchronous signal, "IN_synch" is the output of the synchronizer, as well as the synchronize version of "IN", and "CK" is the reference clock signal of the receiver.

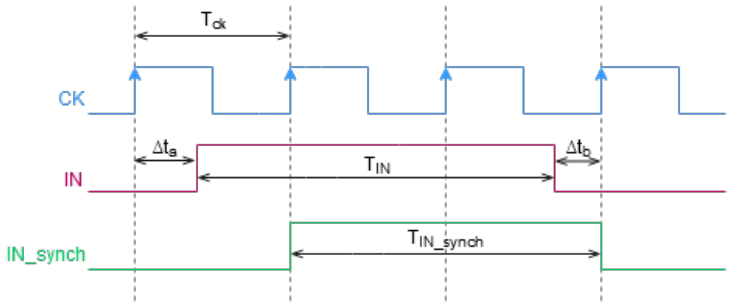


FIGURE 4.12: Example of timing diagram concerning the synchronization stage and the error generated on the synchronized input signal . T_{ck} is the period of the reference clock signal (CK), T_{IN} is the period of the input signal (IN), and $T_{IN_{synch}}$ is the period of IN_synch. Δt_a is the time distance between the rising edge of CK and the rising edge of IN, while Δt_b is the time distance between the falling edge of IN and the rising edge of CK.

Assume that the input period T_{IN} lies in between the following interval:

$$2\ T_{ck} < T_{IN} < 3\ T_{ck}$$

Consider then four successive rising-edges of "CK", as depicted in Figure 4.12. "IN" can have a low-to-high transition a time-interval (Δt_a) after the first edge of "CK",

and a high-to-low transition a time-interval (Δt_b) before the third edge of clock signal. The period of the asynchronous input signal can be expressed in terms of T_{ck}, Δt_a and Δt_b, according to the following equation:

$$T_{IN} = 3\ T_{ck} - \Delta t_a - \Delta t_b \tag{4.6}$$

While the period of the synchronized input signal can be written as:

$$T_{IN_{synch}} = 2\ T_{ck} \tag{4.7}$$

The error concerning the period of the input signal generated at the synchronization stage can be evaluated in absolute terms as:

$$\epsilon_{T_{IN}} \leq \mid T_{IN} - T_{IN_{synch}} \mid = 3\ T_{ck} - \Delta t_a - \Delta t_b - 2\ T_{ck} \mid \tag{4.8}$$

$\epsilon_{T_{IN}}$ can assume different values, depending on Δt_a and Δt_b:

1. If $\Delta t_a, \Delta t_b \longrightarrow 0, \epsilon_{T_{IN}} \longrightarrow T_{ck}$
2. If $\Delta t_a, \Delta t_b \longrightarrow T_{ck}, \epsilon_{T_{IN}} \longrightarrow \mid -T_{ck} \mid = T_{ck}$
3. If $\Delta t_a + \Delta t_b \longrightarrow T_{ck}, \epsilon_{T_{IN}} \longrightarrow 0$

The above-listed case can be summarized with a more general expression as follows:

$$\epsilon_{r_{T_{IN}}\%} \leq \frac{T_{ck}}{T_{IN}} \cdot 100 \tag{4.9}$$

where $\epsilon_{r_{T_{IN}}\%}$ is the relative error generated on T_{IN} during the synchronization stage. Assuiming as acceptable an error of 0.1%, T_{ck} should be at least three order of magnitude smaller than T_{IN}. This demonstrates that the clock frequency fixes the accuracy of the system.

Edge detection and time-to-code conversion

Figure 4.13 shows the timing diagram concerning the whole conversion chain for a high-pulse counting process[4].

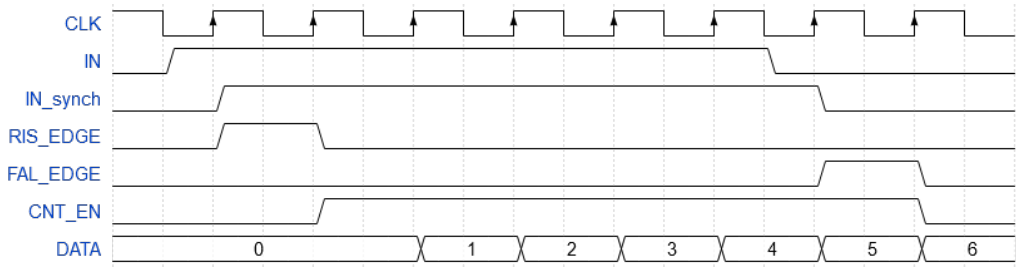


FIGURE 4.13: Example of timing diagram for a high-pulse counting process. IN is input signal. IN_synch is the synchronous input signal. CNT_EN is the counter enable signal and it has the same pulse length as IN_synch. This shows that no further error is generated after the synchronization stage

[4]The timing diagram is the same for a low-pulse counting process

After the synchronization stage, where the input signal is synchronous with the internal clock, the following stages, i.e. the edge detection and the pulse to code conversion, happen coherently with the reference clock, so no further error occur in the conversion chain for what concerns the logical behavior. However it is possible to improve further the accuracy of the whole receiver, by setting the following parameters for each counter:

1. The period of the clock signal (T_{ck}), which determines the resolution of the counter, as the smaller is the period, the higher the resolution in the measurement.

2. The number of bits of the counter (N), which is linked to the clock and to the input signals, and which determines the count length.

If T_{ck} is set to limit the error during the synchronization stage (Eq.4.8), N can be set accordingly as follows:

$$N > \log_2 \frac{\Delta t}{T_{ck}} \tag{4.10}$$

where Δt is the minimum pulse length of the input signal and T_{ck} is the period of the reference clock.

4.5 Functional simulations and analysis of the error performance

As remarked in section 4.2.1 , the system is simulated for different combinations of f_1, f_2, and $\Delta\phi$, and for each combination the results are collected into dedicated text files. These results refers to the counter ticks of the fast-frequency detector and of the slow frequency detector. During the processing stage, the data stored in the text files are converted into the corresponding time information, and the error is evaluated following Eq. 4.1. This is carried out automatically and efficiently with the aid of *python scripts*, as specified in section 4.2.2. The final objective of the analysis of the error performance of digital mixing is to quantify the dependency of the error generated in the mixed signal on the corresponding sources, i.e. f_1/f_2 and $\Delta\phi$.
The analysis will be built upon two main issues:

- study of the trend of the error generated on T_2 in the mixed signal ($e_{r_{T_{2\%}}}$) as a function of the input frequency ratio (f_1/f_2).

- study of the impact of relative phase shift of the input signals ($\Delta\phi$) on the trend of $e_{r_{T_{2\%}}}$ versus f_1/f_2.

In general, this trend is expected to be *decreasing* with the increase of the input frequency ratio, and the impact of $\Delta\phi$ is expected to be significant towards lower values of f_1/f_2.
The analysis starts from the work developed by Ma et al [35], where it was assumed that to have a maximum acceptable error on T_2 equal to the 10%, f_1 should have been at least one order of magnitude greater (x10) than f_2.

4.5.1 Results

Fig. 4.14 shows with three different graphs the trend of the error on T_2 as a function of the input frequency ratio. Each graph showcases that the general trend of $e_{r_{T_{2\%}}}$ in logarithmic scale is exponentially decreasing with f_1/f_2, and in particular from the fourth decade, where f_1 is about three orders of magnitude greater than f_2, the error decreases to almost 0% in any of the three cases.
The graphs depicted in Fig. 4.14a, 4.14b, and 4.14c differ with one other according to the way the phase shift is considered for the calculation of the error, which establishes in particular the dot distribution in the scatter-plot of each graph.
Figure 4.14a exhibits the trend of the average error on T_2: the dots in the scatter-plot represent the error evaluated for a specific input frequency ratio and averaged over all the $\Delta\phi$ for that specific $\frac{f_1}{f_2}$.
Figure 4.14b, showcases the trend of the maximum error on T_2, where each dot represents the maximum error evaluated for a specific input frequency ratio, resulting from a specific phase shift. It is interesting to notice by comparison between the graphs in Fig. 4.14a and in Fig. 4.14b, that the dots distribution of the maximum error changes towards lower values of the input frequency ratios with respect to the average case. In particular, for the same input frequencies the limit error of the 10% previously assumed is overcome, such that the error on T_2 reaches a maximum value of the 12%, as highlighted in red.
Fig. 4.14c exhibits the trend of the minimum error on T_2, where, similarly to the case of the maximum error, the dots of the scatter plot represent the minimum error evaluated for each f_1/f_2, resulting from a certain phase shift. In this particular case, as a results of some specific combinations of frequency ratio and phase shift, $e_{r_{T_{2\%}}}$ reduces to 0% even when the f_2 is only one order of magnitude greater than f_1, that is in the second decade of the plot.
In conclusion, the plots in Fig. 4.14a, 4.14c, and 4.14b demonstrates that the phase shift alters the error towards lower frequency ratios, as you can appreciate from the dot distribution of each scatter plot in the second and the third decade. However, no significant variations occur from the fourth decade on between the plots, as the error reduces exponentially to the 0% in any case.

4.5.2 Analysis

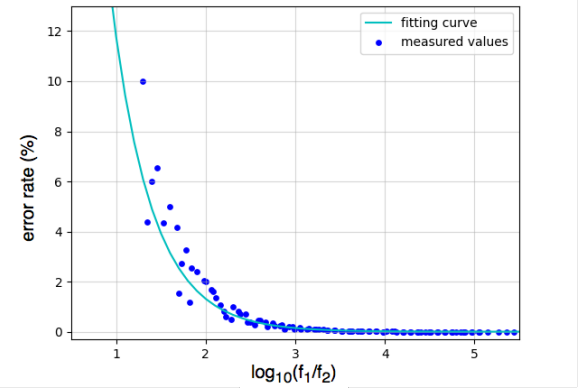
This last section is the core of the analysis, as it aims to (1) quantify the dependency of the error on T_2 on the input frequency ratio, and to (2) justify the alteration of the dot-distribution between the different scatter plots resulting from specific combination of f_1/f_2 and $\Delta\phi$.

(1) Mathematical equation linking the error to the input frequency ratio

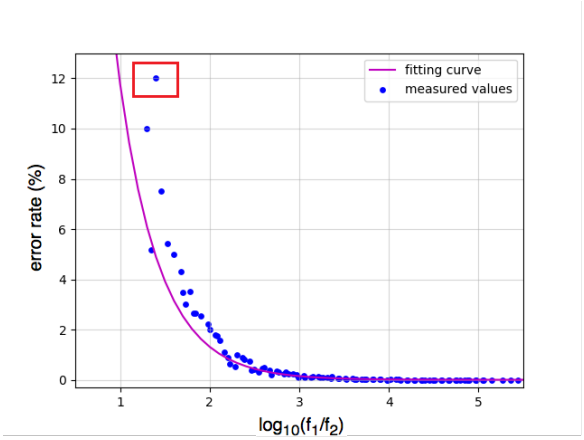
The mathematical relation linking the error rate to the input frequency ratio is defined by a decaying exponential, which can be expressed in a parametric form as:

$$e_{r_{T_{2\%}}} = A \cdot e^{(-B \cdot log_{10}(\frac{f_1}{f_2}))} + C \tag{4.11}$$

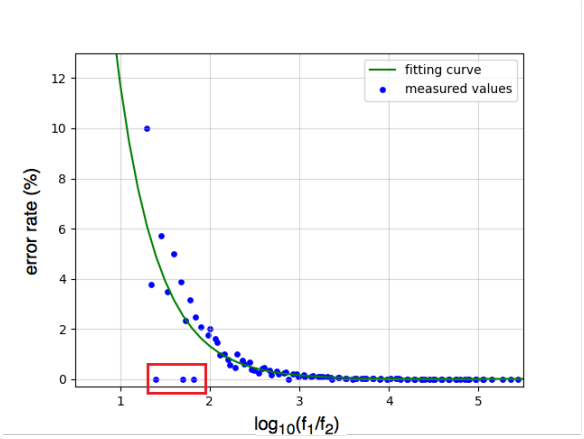
where $e_{r_{T_{2\%}}}$ is the relative percentage error concerning the period T_2 in the mixed signal, f_1/f_2 is the input frequency ratio and A, B and C are the coefficients.



(A) Trend of the average error, showing the average error from the results collected in the simulations over different $\Delta\phi$ for each f_1/f_2. The fitting curve shows the general trend.



(B) Trend of the maximum error on T_2, showing the maximum error from the results collected in the simulations for specific $\Delta\phi$ giving each value and f_1/f_2. The sample within the box in red shows that the error overcomes the limit of the 10% assumed at the beginning.



(C) Trend of the minimum error on T_2, showing the minimum error from the results collected in the simulations for specific a $\Delta\phi$ giving each value and f_1/f_2. The samples within the box in red in red shows that the error due to may reach the 0% even for sufficiently small frequency ratios.

FIGURE 4.14: Trend of the error on T_2 as a function of f_1/f_2, along with the effect of $\Delta\phi$.

From the fitting curve of the average error on T_2 in Fig.4.14a, it is possible to retrieve such coefficients, and Eq. 4.11 can be rewritten as:

$$e_{r_{T_{2\%}}} = 154.122 \cdot e^{(-2.281 \cdot log_{10}(\frac{f_1}{f_2}))} + 0 \tag{4.12}$$

With no surprise, the coefficient C is equal to zero, as the error reduces to 0% with $f_1/f_2 \to \infty$.

(2) Study of the effect of the phase shift on the error with the time-base analysis

Fig. 4.14b and 4.14c highlight in red sections some relevant variations of the error distribution arising for specific phase shifts with respect to the average case in Fig. 4.14a. It is possible to justify such a behavior in great detail with the aid of a time-base analysis.
Figures 4.15 and 4.16 shows two examples of timing diagram of digital mixing, where the input frequency ratio is the same and in particular equal to 25, while the relative input phase shift is different. The goal is to demonstrate the effect of the $\Delta\phi$ on the error generated on T_2 in the mixed signal. Specifically, in Fig. 4.14c, where $\Delta\phi$ is 0%, T_{2_m} equals T_2, so no error is generated in the mixed signal concerning the period T_2.

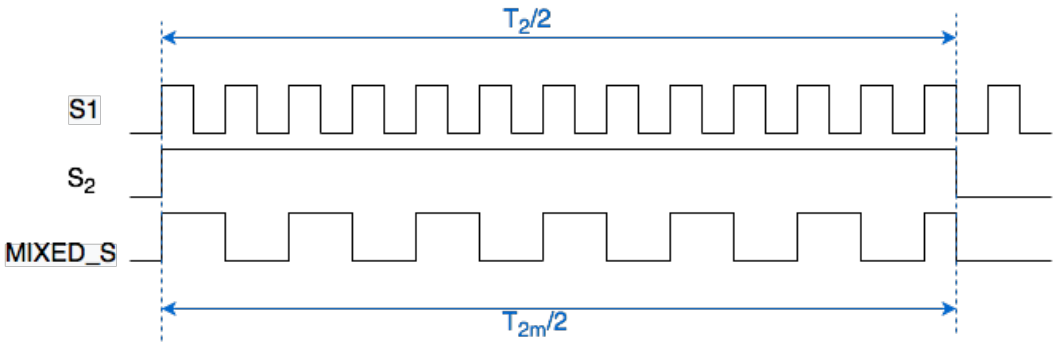


FIGURE 4.15: Example of timing diagram of digital mixing, where the input frequency ratio is equal to 25, and the relative phase shift among S_1 and S_2 is of the 0%. T_2 represents the period of S_2, and T_{2_m} is the period of S_2 in the mixed signal. In this case, the relative error on T_2 is of the 0%

On the contrary, in Fig. 4.16 where $\Delta\phi$ is of the 50%, an error arises, as in this specific case the last high pulse in MIXED_S present in Fig.4.15 disappears because of the non zero-input phase shift. It is possible to compute this error as follows:

$$T_2 = 12.5 \cdot T_1$$

$$T_{2_m} = 11 \cdot T_1$$

$$e_{r_{T_{2\%}}} = \left| \frac{T_2 - T_{2_m}}{T_2} \right| \cdot 100 = \left| \frac{12.5T_1 - 11T_1}{12.5T_1} \right| \cdot 100 = 12\% \tag{4.13}$$

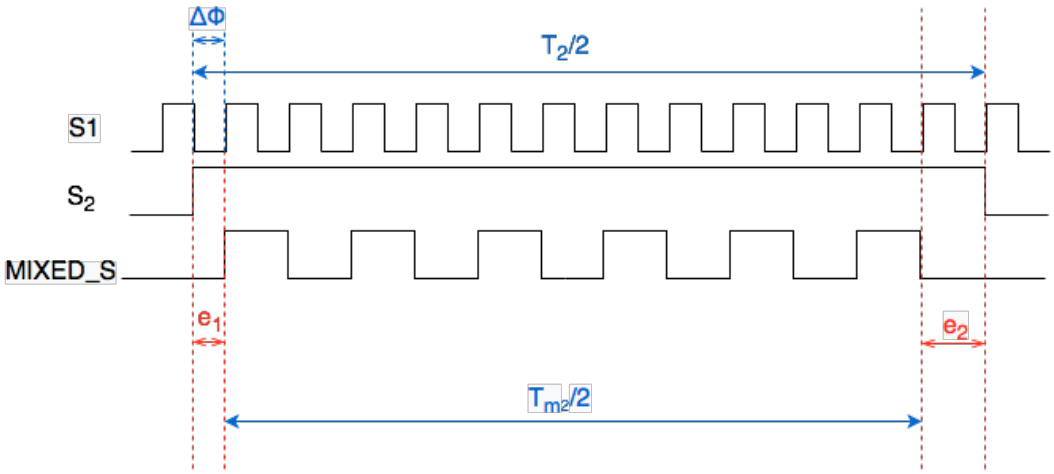


FIGURE 4.16: Example of timing diagram of digital mixing, where f_1/f_2 is equal to 25, and $\Delta\phi$ is of the 50%, . The error on T_2 in the mixed signal assumes a value of the 12%.

Chapter 5

Characterization of Digital Mixing in terms of power consumption

This chapter is dedicated to the study of the sources of power consumption in digital mixing, with a particular focus to the impact of the input frequency ratio on the system power dissipation.
As demonstrated in Chapter 4, the input frequency levels have to be carefully designed to limit the error generated in the mixed signal, especially for what concerns the information associated to T_2. Equation 4.11 shows that the error on T_2 in the mixed signal has an exponential (decreasing) dependency on the input frequency ratio, thus demonstrating that increasing f_1/f_2 represents a reasonable way to improve the error performance. However, this may have certain consequences on the power consumption of the system, and for this reason it is crucial to study the "cost" in terms of power of a specific input frequency ratio. The actual objective, indeed, consists in the analysis of the best trade-off in terms of choice of the input frequency levels between the error performance and the power consumption for a specific application.

5.1 Sources of power consumption in digital mixing

The analysis of the power consumed in digital mixing relies on a preliminary literature review of the sources of energy consumption in digital circuits, with a particular focus for the dependency of the dissipated power on *frequency*.

5.1.1 Power consumption in digital CMOS circuits

There exists several methods for the characterization of the power consumed in a digital system, and separation into different abstraction level is often assumed [43]. At the lower levels of abstractions, where logic gates and flip-flops or latches are considered, the characterization of the power consumption is related to the technology assumed for the circuit implementation [44]. As most logic today is based on CMOS logic families [45], the following analysis will assume CMOS technology for the implementation.
The sources of power consumption in a CMOS circuit can be classified in general as (1) static and (2) dynamic power dissipation [46], and the main difference between them is that (2) is frequency dependent, which is of great relevance for this analysis.
In a first order approximation, the dynamic power consumed by a CMOS circuit

can be estimated according to the following formula[1]:

$$P_{dyn} = C_{eff} V^2 f \tag{5.1}$$

where P_d is the power in Watts, C_{eff} is the effective switch capacitance in Farads, V is the supply voltage in Volts, and f is the frequency of operations in Hertz [46], which in the case of sequential circuits coincide with the reference clock signal.
The effective switch capacitance can be computed as:

$$C_{eff} = \alpha C_{Load} \tag{5.2}$$

where α is the switching activity, that denotes the number of changes in the output node during one time cycle of the clock signal, and C_{Load} the load (or output) capacitance being charged and discharged.

5.1.2 Dynamic power in the digital mixer implemented with the D-FF

Eq.5.1 demonstrates that the dynamic power consumed in the digital mixer is **linearly dependent** on the clock frequency, which coincides with frequency f_1, as depicted in Fig.5.1. For what concerns the reset signal instead, no significant variations of P_{dyn} are expected as its frequency (f_2) varies.
Figure 5.2 shows two examples of timing diagram of digital mixing with D-FF: in both Fig.5.2a and 5.2b, the clock frequency is assumed to be the same, while the frequency of the reset signal (S2) is higher in the case depicted in Fig.5.2b. You can observe that the the frequency of the reset signal determines the number of ON and OFF states in the mixing within a certain time interval, but it does not modify the number of state transitions ($0 \rightarrow 1$, and $1 \rightarrow 0$) in the output signal of the mixer. This means that the dynamic power is independent on f_2.
However, what makes the dynamic power vary in the mixer is the **duty-cycle** of the reset. In particular, the smaller the duty cycle, the longer the OFF states, the less the number of transitions in the output signal. Of course, this has a trade-off with the error performance in digital mixing, but the effect of the duty cycle is not going to be discussed throughout this analysis.
In conclusion, the expected trend of the dynamic power consumed in the D flip-flop as a function of the input frequency can be represented by a straight line, whose slope will be determined by the product $C_{eff} V^2$ (according to eq. 5.1), which is dependent on the technology for the circuit implementation. Additionally, if the total power is considered as a function of the input frequency ratio, which means to sum up the dynamic power and the static power, this straight line is expected to be shifted upward or downward depending on the target technology, as the static power is *frequency independent* [49].

[1] It is to remember that the total dynamic power in a CMOS digital circuit is the sum of the switching power and of the short-circuit power [47]. However, here it is assumed that there is no short-circuit current, so the total dynamic power is given by the switching contribution only [48]

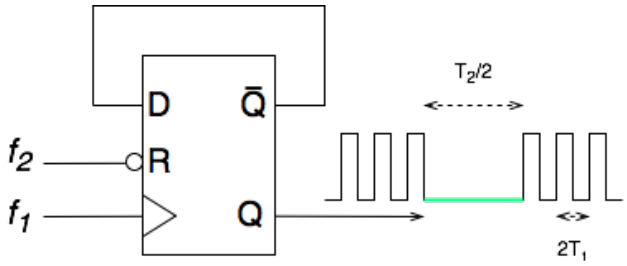
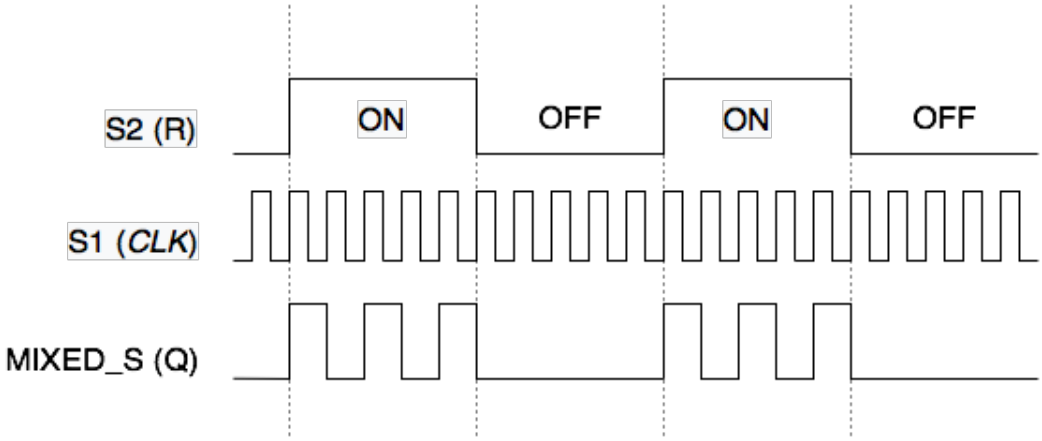
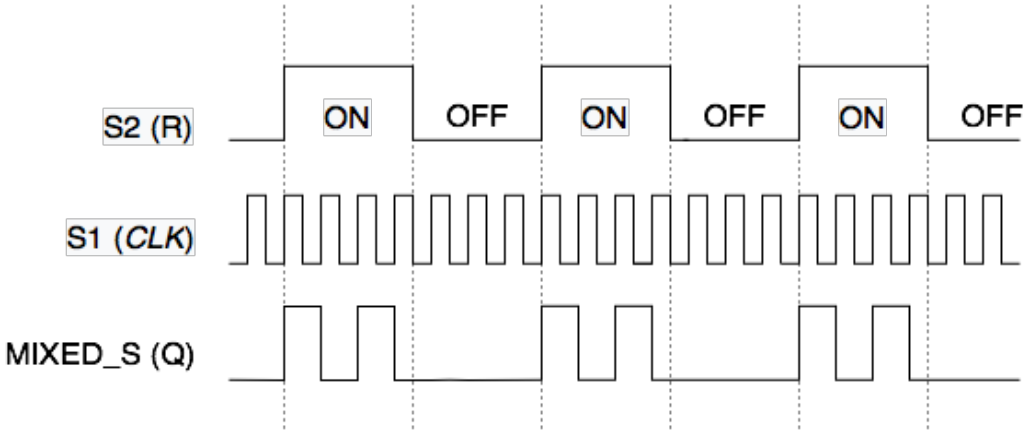


FIGURE 5.1: RTL view of the digital mixer implemented with the D flip-flop. The reset signal when active, determines the off state of the digital mixer, where no output transitions occur.



(A)



(B)

FIGURE 5.2: Timing diagrams of the signal mixing. The reset signal (S2) determines the number of ON and OFF states in a certain time interval. Assuming that the frequency of the clock signal (S1) is the same, any variation of the frequency of the S2 does not modify the number of transitions of the mixed signal (MIXED_S): in 5.2a and 5.2b the total number of transitions of the output is the same, so no variations of the dynamic power consumed in the mixed are expected

5.2 Analysis of the Dynamic Power versus input frequency ratio

As previously anticipated, the core of the analysis of the power consumption in digital mixing is the study of the dependency of the power dissipated on the input frequency ratio. A preliminary discussion on the subject has been carried out in the previous section, in particular for what concerns the dynamic contribution. The work developed in this section aims to analyse in great detail such a dependency. This is achieved through the synthesis of the VHDL description of the circuit.

5.2.1 Methodology

The method proposed is based on the synthesis of the D flip-flop mixer with the aid of *Synopsys Design Compiler*, and it can be subdivided into two different stages:

- **Synthesis Stage**, where the mixer is synthesised for different possible combinations of input clock and reset frequencies, and the power consumed is reported.
- **Processing Stage**, where the results of power reports are first extracted, and then processed, to examine in great detail the dependency of the dynamic power on the input frequency ratio.

As for functional simulations (Ch.4), the system has to be synthesized for different combinations of f_1 and f_2, so the use of scripting methods is essential to ensure efficiency during the entire process.

Automation in the Synthesis stage - *bash, Tcl*

Figure 5.3 demonstrates the flow of the automation process for the Synthesis stage. The relevant steps are modelled by dedicated files. In particular, "Conf. Synth. Script", which is a bash file (*.sh*), generates a loop structure where it (1) modifies the values of T_1 and T_2 in the "Run Script" file, before running it, and (2) configure Synopsys under the Unix environment to work in shell mode.
The "Run Script" file is a Tcl script (*.tcl*), which is essentially a set-up file for the Design Complier, where all the command for the synthesis are issued, as many Synopsys command shells uses **Tcl** as a scripting tool for automating the design processes [50]. For each synthesis, the results related to the power analysis are collected into a dedicated report (*.txt*). The outcome of the whole synthesis stage is a set of power report, each corresponding to a specific combination of T_1 and T_2. As for functional simulations, the total number of synthesis is defined in the "Conf. Synth. Script" through the depth of the loop. The codes of "Run Script" and "Conf. Synth. Script" are reported in Appendix B to clarify the flow of the operations needed to perform the entire synthesis stage.

Automation in the Processing stage - *bash, python*

Figure 5.4 showcases the flow of the automation process for the Processing stage. The power reports generated by Synopsys DC contains several data, and most of them are useless for the current analysis. To this end, with the aid of the "Filter

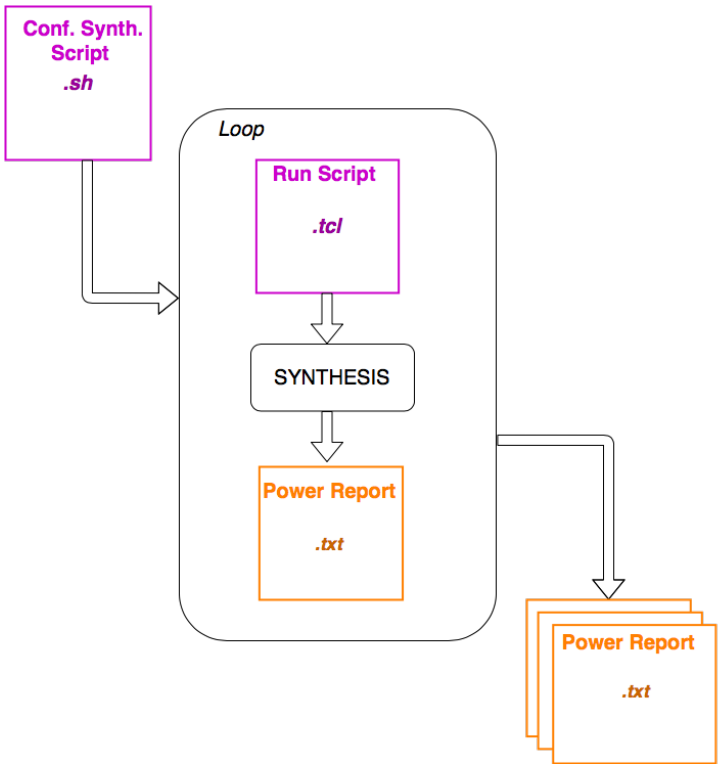


FIGURE 5.3: Conceptual representation of the flow of the synthesis stage with the relevant files involved in the automation process.

Reports" script (*.sh*), which generates a loop where it iteratively filters the data of interest in each report and stores it in a text file ("Extracted Data"). This file will be used by the "Analysis" script (*.py*) to create a plot of the dynamic power consumed in the mixer as a function of the input frequency ratio.

5.2.2 Results

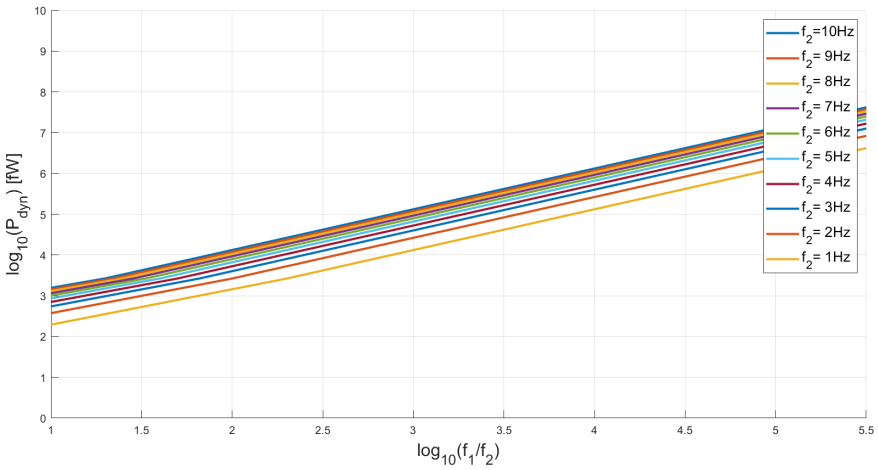


FIGURE 5.5: Dynamic Power consumed in the digital mixer versus input frequency ratio. Each curve refers to a specific f_2.

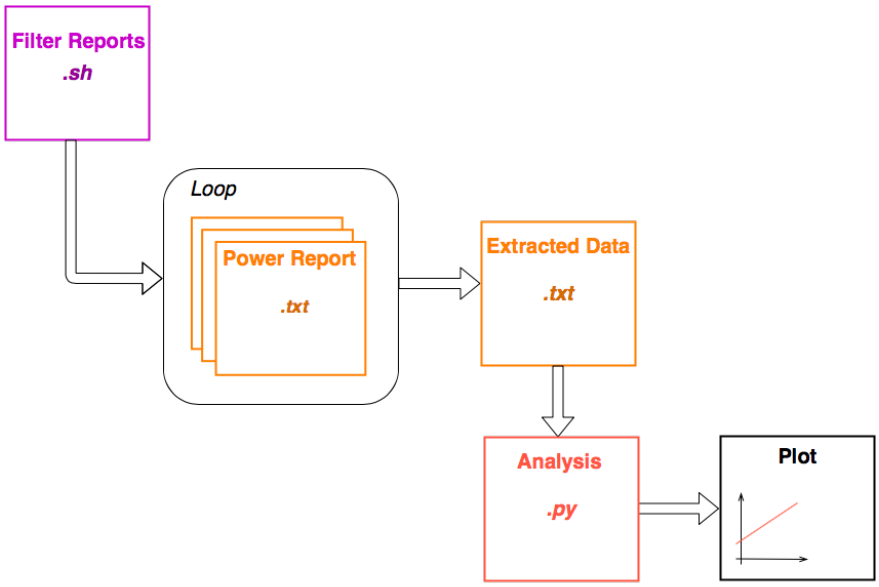


FIGURE 5.4: Conceptual representation of the flow of the processing stage for the results from synthesis with the relevant files involved in the automation process.

The results from the processing stage after synthesis of the mixer are shown in Fig.5.5, which showcases a collection of curves characterizing the trend of the dynamic power the digital mixer for different frequencies of the reset signal (f_2) as a function of the input frequency ratio (f_1/f_2). This trend, in particular, is linear, and it follows the expectations.
It is interesting to notice that as f_2 increases, each curve is progressively shifted upwards. This behaviour is due to the way the reset is created by the Design Compiler during the synthesis, and to the time it takes to compute the dynamic power. If the reset is assumed to start with a 1 and stops with a 0 logic value, the dynamic power estimation starts and stops with the ON state of the mixer. As a consequence, the compiler computes the dynamic power considering an odd number of ON states, and an even number of OFF ones, thus making the dynamic power increase with f_2. The percentage of increment will depend on the total number of periods (integer or fractional) of the reset considered by the compiler during the synthesis.

5.3 Error versus power trade-off in digital mixing

As specified multiple times throughout this project, the final objective of the characterization of the digital mixing is the study of the trade-offs in terms of power and error performances for the design of the input frequency levels. This can be achieved by combining the results obtained from the error and power analysis, as depicted in Figure 5.6, which showcases the integration of the power and error plots discussed respectively in sections 4.5 and 5.2.
The straight lines model the linear dependency of the dynamic power consumed in the mixer on the input frequency ratio, while the decaying exponential curve is the fitting curve of the average error on T_2 versus f_1/f_2, obtained according to Eq.

4.12. The intersection points represents the trade-off condition for the characterization of the digital mixing. They belong to the following interval:

$$1.5 < log_{10}\left(\frac{f_1}{f_2}\right) < 2$$

They are points of global minima of the combined power-error plot demonstrating the optimum operating condition of the mixer, where both power consumption and error performances are optimized. This trade-off is achieved with a careful design of the frequency ranges of the input signals for the target application.

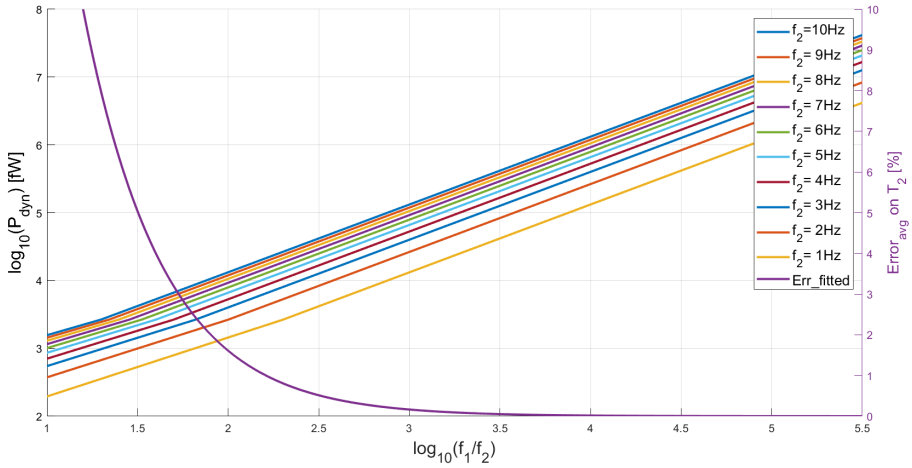


FIGURE 5.6: Combined (1) error and (2) power plots. (1) is the fitting curve of the average error on T_2 versus f_1/f_2 obtained through Eq.4.12. (2) is an array of straight lines, each referring to a specific f_2, and representing the dynamic power consumed in the digital mixer versus $\frac{f_1}{f_2}$.

Chapter 6

Digital Modulation Schemes

This chapter provides a deep insight into the concept of digital modulation. The subject will be approached from a general perspective, without taking into account the target application of the proposed thesis project. Section 6.1 introduces the concept of digital modulation, analysing the advantages over the analog counterpart. Section 6.2 provides a list of the most common digital modulation schemes, highlighting the most relevant features of each of them. Lastly, Section 6.3 is dedicated to the definition of useful Figures Of Merit (FOM), which allows for a comparative analysis of the existing methods, and for the choice of the most suitable modulation technique according to the requirements of the specific application.

6.1 Digital Modulation

In telecommunication systems, modulation is the method of conveying some information, usually a low-frequency signal, by varying one or more properties of a periodic signal, called "carrier signal", which is a sinusoidal signal with a high frequency [51], [52], [53]. These properties could be the envelope, the frequency or the phase of the carrier. The source information is the "modulating signal", while the carrier is the "modulated signal". When the modulating signal is analog, which means that the variation of the carrier property is continuous, the modulation technique is called specifically "analog modulation", otherwise when such a variation is discrete, the modulation technique is called "digital modulation" [54].
Digital modulation has several advantages over the analog counterpart:

- larger noise and external interference immunity. Analog signals can take an infinite number of shapes, whereas digital signals have two discrete transmission states, that are so more easily detected at the receiver side in presence of noise or external interference [55].
- improved data security [56], as the transmission of the information can be done in encoded form, thus allowing for signal encryption.
- easier implementation of error correction coding.
- reduced hardware for the implementation [54].

In a digital communication system, the **bit stream**, which represents the information to be sent through the wireless channel, is represented essentially by a binary voltage waveform, as it is shown in Figure 6.1.
These binary voltage waveforms are voltage pulses alternating between a high voltage level and a low one, and their frequency content is typically very low. For

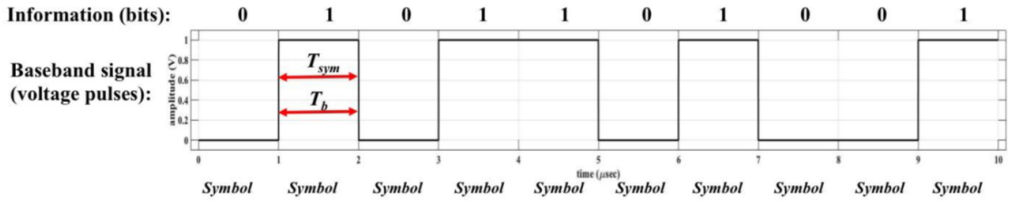


FIGURE 6.1: Binary voltage waveform representing the bit stream

this reason, it is often termed as "baseband" signal, as its spectrum extends from dc (i.e. at 0 Hz), where it exhibits the largest frequency content in magnitude, up to a certain value, as depicted in Fig. 6.2. At regular time intervals the magnitude of the spectrum goes to zero. To succeed in transmitting the baseband signal through the wireless channel, an impractically large antenna would be required. With digital modulation, the frequency spectrum of the baseband signal is "up-shifted" to allow for the wireless transmission with a reasonably sized antenna.

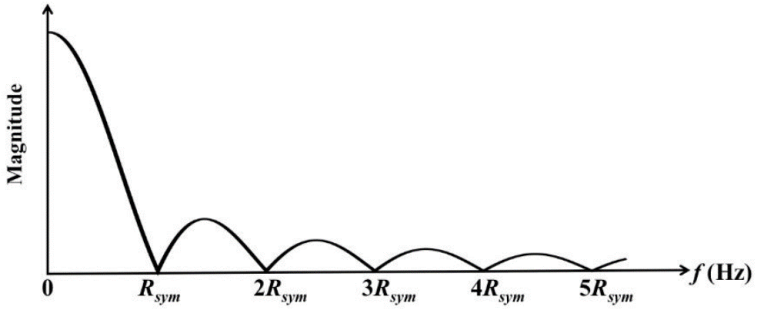


FIGURE 6.2: Shape of the spectrum of the base-band digital signal

6.1.1 Bit-to-Symbol Mapping

As previously stated, the goal of digital modulation is to transmit digital data across the wireless channel, which is an "analog medium" [52]. To this end, the digital modulator at the transmitter side converts the digital information into analog waveforms, called "symbols", which is discrete in time and represents a particular "state" of the carrier waveform. Depending on the type of digital modulation scheme, a bit or a set of bits can be mapped (or assigned) to a particular symbol, and this assignment process is called "*bit-to-symbol mapping*"[1] [57]. There exists a relationship linking the number of bits (N) used for the mapping to the number of possible symbols available at the transmitter (M), and it can be written as:

$$N = \log_2(M) \tag{6.1}$$

The **symbol rate** (R_{sym}) is the number of symbols per second being transmitted (or the time required to send one symbol) and it is often termed as **baud or baud rate**. The bit rate (R_b), instead, is the number of bits being transmitted per second.

[1]The concept of bit-to-symbol mapping will be discussed more in depth in Chapter 6 to introduce the method used for the simulation of the BER in MATLAB.

The symbol rate and the bit rate are related through the following equation:

$$R_{sym} = N \cdot R_b \tag{6.2}$$

with N being the number of bits per symbol.
As previously stated with reference to Figure 6.2, at regular intervals the frequency content of the base-band signal goes to zero magnitude. These intervals are typically multiples of the symbol rate. Moreover, the magnitude of the spectrum of the baseband signal drops drastically at certain frequencies. The transmission bandwidth can be computed as:

$$BW = f_2 - f_1 = R_{sym} - 0 = R_{sym} \tag{6.3}$$

With digital modulation, the spectrum of the transmitted signal assumes a shape like the one shown in Figure 6.3, where the signal to be transmitted is shifted up to the frequency of the carrier f_c.

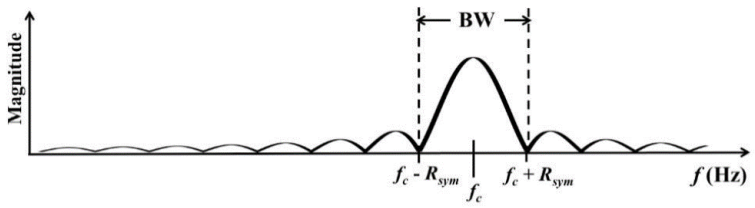


FIGURE 6.3: Shape of the spectrum of the up-shifted baseband signal

The transmission bandwidth now can be computed as follows:

$$BW = f_2 - f_1 = (f_c + R_{sym}) - (f_c - R_{sym}) = 2R_{sym} \tag{6.4}$$

6.2 Digital Modulation Schemes

The process of modulating one or more properties of the carrier with bits is usually termed as "shift-keying", as bit values “shift” between 0 and 1, and it is employed to differentiate digital modulation from the analog counterpart. The three basic types of digital modulations are (1) Frequency Shift Keying (FSK), (2) Phase Shift Keying (PSK), and (3) Amplitude Shift Keying (ASK). Moreover, digital modulation schemes can be classified into two sub-categories, depending on the number of bits used for the mapping, that are:

- Binary Digital Modulation Schemes
- M-ary Digital Modulation Schemes

In binary digital modulation two symbols can be transmitted, and each symbol carries 1 bit of information, whereas M-ary modulation schemes have more than two symbols that can be transmitted, and each symbol carries more than one bit of information.

6.2.1 Frequency Shift Keying

In Frequency Shift Keying (FSK) the carrier frequency is shifted between discrete values to transmit the digital data. The simplest FSK is binary FSK (BFSK), where the frequency of the carrier is specifically shifted between two discrete values, depending on whether a 1 or a 0 is sent. With this scheme, the 1 bit is mapped to the so called "mark frequency" (f_{mark}), while the 0 bit to the "space frequency" (f_{space}) [58].

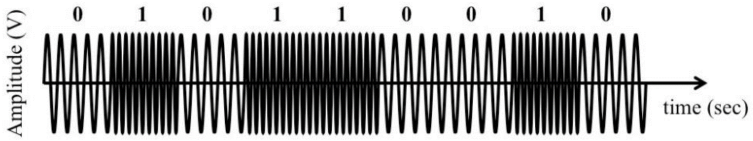


FIGURE 6.4: Representation of the FSK signal in the time domain

Figure 6.5 showcases the spectrum of a frequency modulated signal.

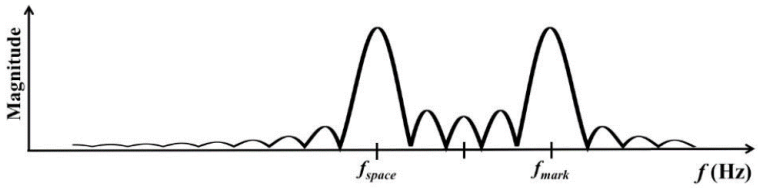


FIGURE 6.5: Magnitude of the spectrum of FSK signal.

The bandwidth of the FSK signal extends from the first zero to the left of space frequency up to the first zero to the right of the mark frequency, and it can be computed as follows:

$$BW = f_2 - f_1 = (f_{mark} + R_{sym}) - (f_{space} - R_{sym}) = f_{mark} - f_{space} + 2R_{sym} \quad (6.5)$$

Provided that $R_{sym} = R_b$ for a BFSK modulated signal, Eq. 6.5 can be rewritten as:

$$BW = f_2 - f_1 = (f_{mark} + R_b) - (f_{space} - R_b) = f_{mark} - f_{space} + 2R_b \quad (6.6)$$

6.2.2 Amplitude Shift Keying

In Amplitude Shift Keying (ASK), the digital data are represented as discrete variations in the amplitude of the carrier signal [58], [59]. The simplest ASK is Binary-ASK (BASK) where the amplitude of the carrier shifts between two discrete values, representing the 0 and the 1 bit respectively.
Figure 6.6 shows an example of the shape of an amplitude modulated signal in the time domain. One particular type of BASK is the so called "On-Off Keying", where the amplitude of the carrier shift between a certain value and the zero value, as depicted in Fig. 6.7, which means that the carrier is actually transmitted only to convey the 1 bit [58].

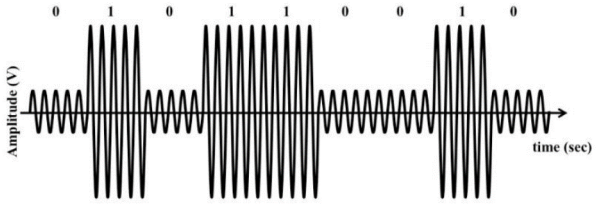


FIGURE 6.6: Representation of a BASK signal in the time domain.

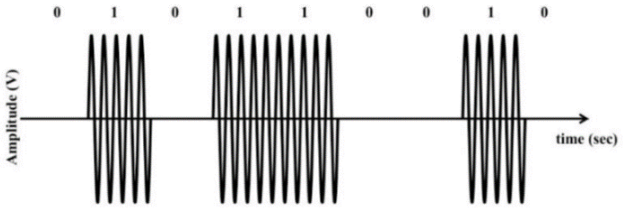


FIGURE 6.7: Representation of a OOK signal in the time domain.

6.2.3 Phase Shift Keying

In PSK, the digital data is conveyed by changing the phase of the carrier wave between discrete values. Each discrete value of the phase in PSK is assigned to a specific pattern of bits. The simplest form of PSK is binary phase shift keying (BPSK), where the two phases are shifted apart of 180°. For this reason PSK is often also be termed 2-PSK [58].

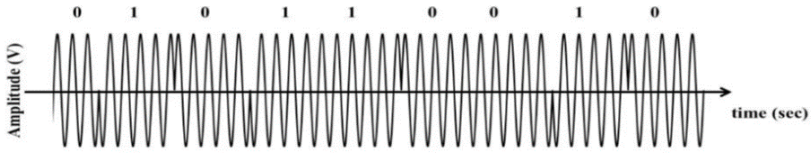


FIGURE 6.8: Representation of a BPSK signal in the time domain.

6.2.4 Bandwidth of BASK, OOK, and BPSK signals

For BASK, OOK and BPSK, the bandwidth is the same, as depicted in Figure 6.9, and it can be computed according to the following equation:

$$BW = 2Rsym \tag{6.7}$$

6.2.5 The Constellation Diagram

Digital modulation schemes can be are represented on the complex plane, which is also called "constellation plane" or "constellation diagram", or I-Q plane, using the radio-communication terminology [57]. Before introducing the constellation diagram, it is useful to consider some mathematical formalism behind the concept of modulation.

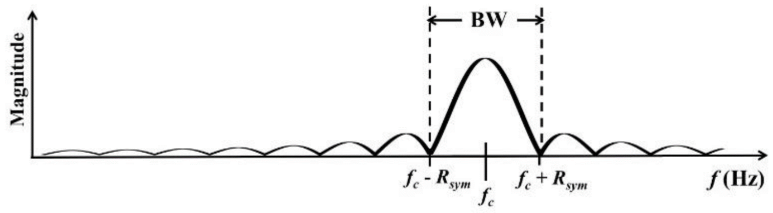


FIGURE 6.9: Spectrum of binary phase or amplitude modulated signals

6.2.6 Complex Envelope

The carrier is a cosine wave with frequency f_c, that follows the equation:

$$c(t) = 1.0 \cdot cos(2\pi f_c t) \tag{6.8}$$

The carrier itself does not carry the information to covey across the wireless channel. After modulation, the resulting signal of up-conversion with the carrier frequency, can be represented as a cosine being modulated by a baseband signal with its amplitude (A) and phase (ϕ).

$$s_x(t) = Acos(2\pi f_c t + \phi) \tag{6.9}$$

where A and ϕ are, in general, function of time.
Eq. 6.9 can be rewritten using the definition of the "complex envelope" [57], which is another useful representation of the modulated signal, according to the following equation:

$$s_x(t) = \text{Re}\left\{\left[A_I + jA_Q\right]e^{j2\pi f_c t}\right\} \tag{6.10}$$

where $C = A_I + jA_Q$ is the complex envelop, which can be considered a phasor in the complex plane, that changes dynamically accordingly to the baseband signal of amplitude A and phase ϕ.

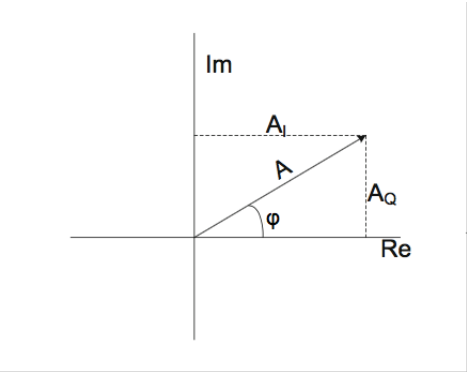


FIGURE 6.10: Phasor representation of the complex envelope in the complex plane

For digital communication, the complex envelope, which is a continuous signal, is generated from the bit stream [57]. In particular, one ore more bits are assigned to a symbol, that can be represented by a point on the complex plane, or constellation diagram. In other words, the constellation plane is a plot of the phase

and relative amplitude in polar coordinates, as depicted in Fig. 6.10 of the output symbols for a digital modulation system. A symbol with 0° phase lies along the axis of the complex plane, and an increase of the phase corresponds to a counter-clockwise movement around the plane. The relative amplitude of each symbol, instead, is the distance from the origin of the complex plane. The possible output symbols are represented with large dots, and adjacent to them are the bits they represent.

Example of constellation diagram for BPSK modulation

In BPSK, there are two possible output symbols, which have the same amplitude. In the constellation diagram, these symbols are represented by two different dots, both equidistant from the origin, and their phase is shifted apart of 180°. As depicted in Fig.6.11, there exists two possible combinations of symbols for the carrier phase:

- 0° and 180°
- +90° and -90°

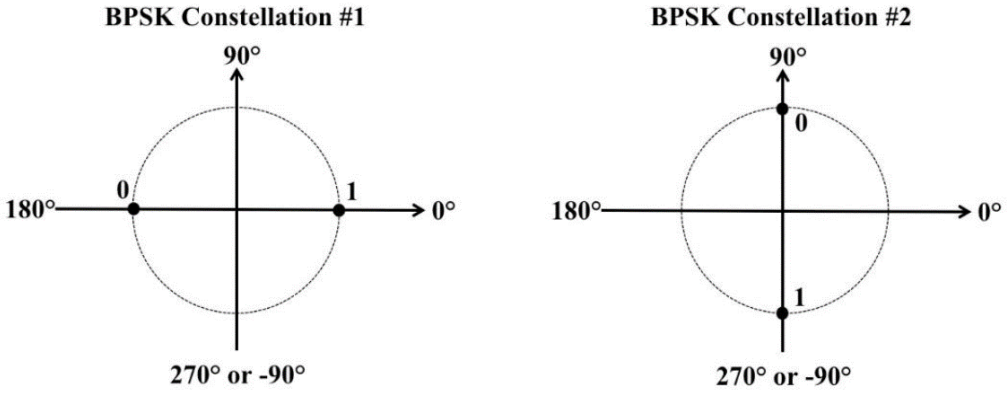


FIGURE 6.11: Examples of constellation diagram for BPSK modulation

However, as long as transmitter and the receiver use the same constellation pattern, the specific constellation employed for the transmission diagram is not relevant.

6.2.7 M-ary Digital Modulation Schemes

Modulation schemes that use more than one bit per symbol for the mapping, which means that more than one bit is transmitted at a time, are usually referred to as "M-ary modulation schemes". Other expression often used are "higher order modulation", "multilevel modulation" or "modulation with alphabet size M" [57].
M-ary modulation schemes are usually employed to increase the bit-rate (number of bit per second), or data-rate, while preserving the transmission bandwidth (one symbol at a time), if the symbol rate is the same, as more bit per symbol are used [59]. However, the cost is the increase of the transmission bandwidth, which can be expensive. This is of utmost importance, as increasing the transmission bandwidth

is often expensive, beside requiring more complex architectures for the implementation.
Figure 6.12 showcases the frequency spectrum of M-ary modulation schemes. It is interesting to notice that the shape is the same as the one of ASK, OOK, and BPSK 6.9. The difference with respect to the binary modulation schemes lies on the fact that, being the bit rate constant, the nulls of the frequency spectrum are closer to the carrier frequency, since $R_{sym} = \frac{R_b}{N}$.

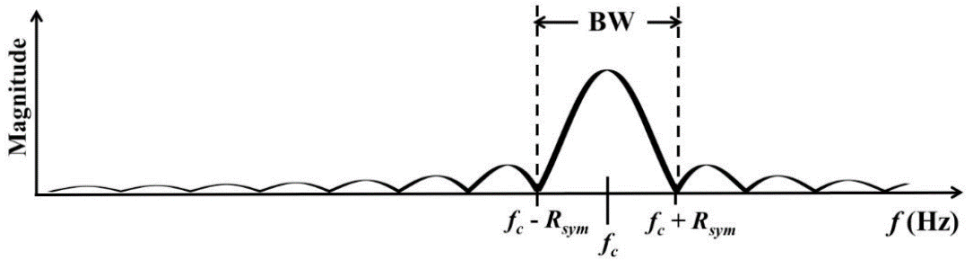


FIGURE 6.12: Frequency spectrum of a QPSK modulated signal

Therefore, the bandwidth of M-ary modulation techniques can be computed as:

$$BW = 2R_{sym} = 2\frac{R_b}{N} = 2\frac{R_b}{2} = R_b \tag{6.11}$$

QPSK - Quadrature Phase Shift Keying

One example of M-ary modulation scheme is Quadrature Phase Shift Keying (QPSK), where it is possible to transmit 4 symbols. This means that two bits per symbol are actually transmitted, according to Eq. 6.1.
Figures 6.13 show two examples of constellation diagram in QPSK, while Figure 6.14 showcases the corresponding carrier waveform in the time domain.

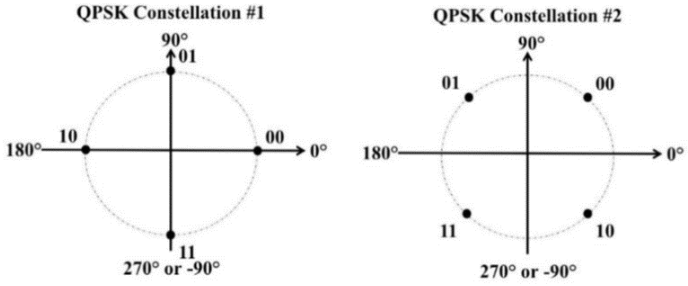


FIGURE 6.13: Examples of constellation diagrams for QPSK

6.2.8 M-ary digital modulation and noise corruption

It is possible to increase further the number of bits per symbol with respect to QPSK, thus increasing the total number of symbols to be transmitted. If, on one hand the bandwidth is further reduced as well, **the system will result more susceptible to noise**, and this can be demonstrated with the aid of the constellation diagram, as the symbols get closer to each other.

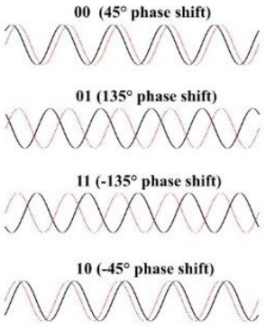


FIGURE 6.14: Shape of the QPSK signal in the time domain. The different phase corresponds to number of symbols available at the transmitter

Noise represents one of the main limiting factors in communication systems. In all transmissions, the received signal is degraded by noise, and it is harder the detection of the correct signal at the receiver end. Figure 6.15 illustrates the effect of noise degradation on the the shape of a BPSK signal in the time domain.

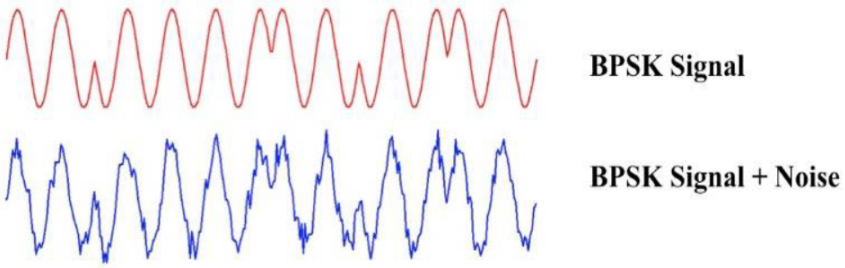


FIGURE 6.15: Noise corruption of a BPSK signal: illustration of the signal representation in the time-domain

The effect of the noise corruption on the modulated signal can be shown also in the constellation diagram, as displayed in Figure 6.16 for the case of a BPSK signal. As you increase the number of symbols, and so the order of the modulation, as depicted in Figure 6.17), you increase the probability of error when demodulating the signal at the receiver side as well.

6.2.9 QAM - Quadrature and Amplitude modulation

There exists a possibility of using more symbols, while reducing the probability of making bit errors at the receiver side and this requires to employ of symbols that have different **amplitudes** and **phases**. As a results, the points of the constellation diagram results more spread out. This is the case of the Quadrature and Amplitude modulation technique. An example of QAM modulation is shown in Figure 6.18, while in Figure 6.19 it is shown the related constellation diagram.

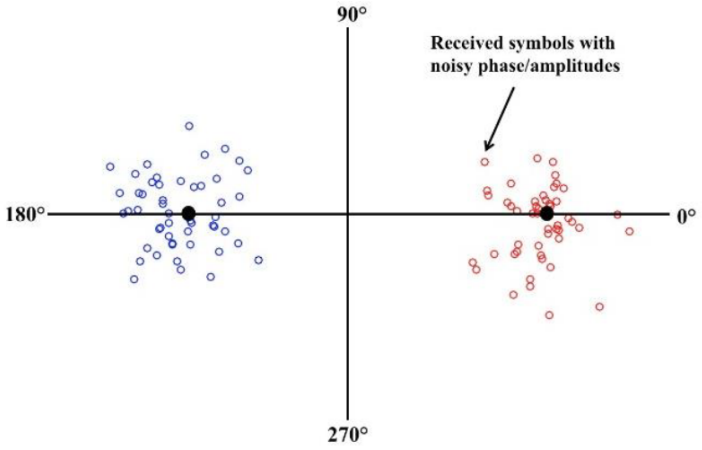


FIGURE 6.16: Representation of the effect of the noise corruption on a BPSK modulated signal in the constellation diagram: the two transmitted BPSK symbols are represented by two larger black circles, with phase respectively 0° and 180°. The surrounding blue and red coloured dots represents the "noisy symbols" at the receiver end.

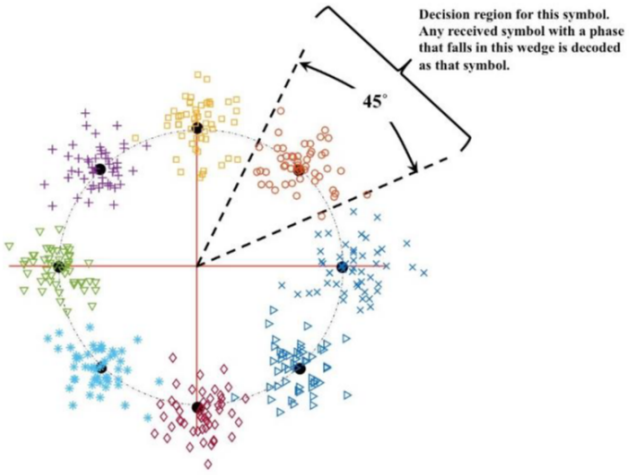


FIGURE 6.17: Representation of the effect of the noise corruption on a higher order digital phase modulated signal in the constellation diagram

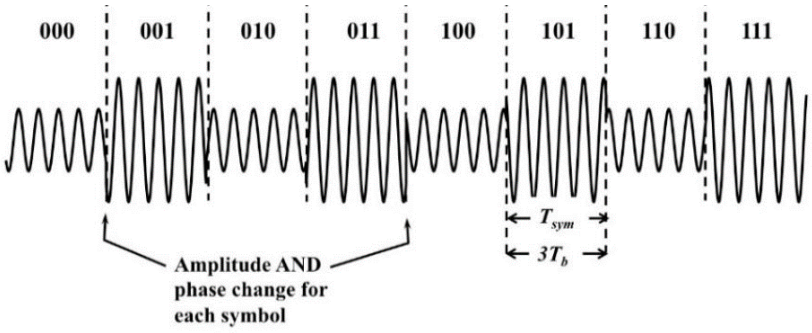


FIGURE 6.18: 8-QAM modulated signal in the time domain along with the corresponding bit used for the mapping

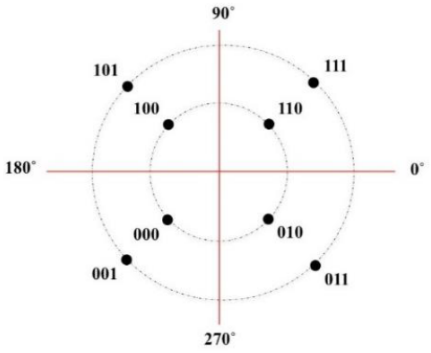


FIGURE 6.19: Example: constellation diagram of 8-QAM modulation scheme

The bandwidth can be evaluated according to the following formula:

$$BW = 2R_{sym} = \frac{2R_b}{N} = \frac{2R_b}{3} \tag{6.12}$$

Higher level QAM signals

Higher order QAM allows to transmits multiple bits, and so a larger number of symbols, thus allowing for much higher bit rate in the same bandwidth. Figure 6.20 showcases the constellation diagram of 64-QAM (to the left) and 256-QAM schemes.

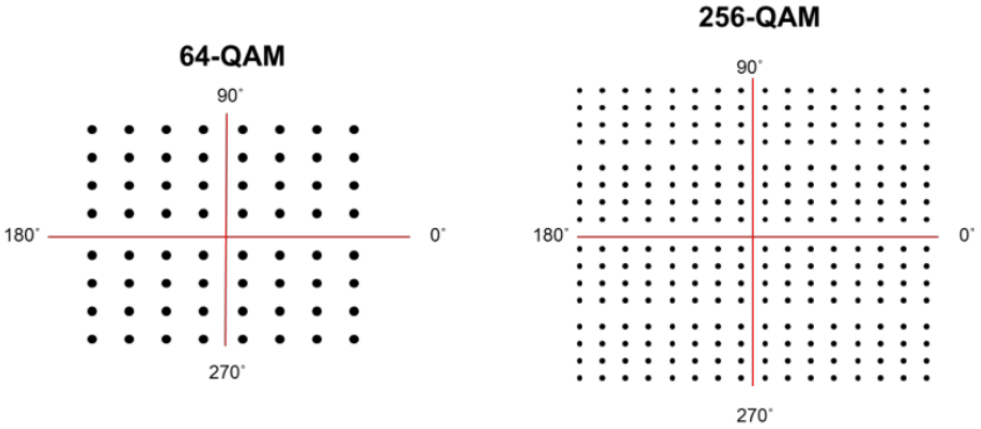


FIGURE 6.20: Example: constellation diagram of higher order QAM modulation scheme.

6.2.10 Summary of Types of Digital Modulation Schemes

Table 6.1 summarizes some of the most common types of digital modulations. There exist PSK and QAM with 32, 64, 128 symbols, etc. However, usually 32 is the highest modulation order used with PSK because of the effect of the noise degradation.

Summary of Digital Modulation Schemes			
Modulation	#Symbols(M)	#Bits/symbols ($N = log_2(M)$)	Bandwidth (BW)
FSK	2	1	$f_{mark} - f_{space} + 2R_b$
ASK	2	1	$2R_{sym} = 2R_b/N = 2R_b$
OOK	2	1	$2R_{sym} = 2R_b/N = 2R_b$
BPSK	2	1	$2R_{sym} = 2R_b/N = 2R_b$
QPSK	4	2	$2R_{sym} = 2R_b/N = R_b$
8PSK	8	3	$2R_{sym} = 2R_b/N = 2R_b/3$
8QAM	8	3	$2R_{sym} = 2R_b/N = 2R_b/3$
16PSK	16	4	$2R_{sym} = 2R_b/N = R_b/2$
16QAM	16	4	$2R_{sym} = 2R_b/N = R_b/2$
32PSK	32	5	$2R_{sym} = 2R_b/N = 2R_b/5$
32QAM	32	5	$2R_{sym} = 2R_b/N = 2R_b/5$
64PSK	64	6	$2R_{sym} = 2R_b/N = R_b/3$
64QAM	64	6	$2R_{sym} = 2R_b/N = R_b/3$
128PSK	128	7	$2R_{sym} = 2R_b/N = 2R_b/7$
128QAM	128	7	$2R_{sym} = 2R_b/N = 2R_b/7$
256PSK	256	8	$2R_{sym} = 2R_b/N = R_b/4$
256QAM	256	8	$2R_{sym} = 2R_b/N = R_b/4$

TABLE 6.1

6.3 Figure of Merits in Digital Modulation

The choice of a suitable modulation format relies on the analysis of some useful criteria, as well as figures of merit (FOM), listed in the following:

- Bit Error Rate (BER)
- Signal to Noise Ratio (SNR)
- Energy per bit-to noise power density ratio ($\frac{E_b}{N_o}$)
- Power efficiency
- Bandwidth-Efficiency
- Complexity and cost-efficiency (not discussed in the following).

This list will be analysed in detail in the following sections. An ideal modulation scheme provides low bit error rates (BER) at low received signal-to-noise ratios (SNR), and occupies a minimum bandwidth. However, the existing digital modulation techniques do not satisfy all these requirements at once. Depending on the requirements of a particular application, the choice of the most suitable candidate is often the result of some trade-offs.

6.3.1 Bit Error Rate (BER)

The concept of Bit-Error-Rate (BER) has been discussed several times in literature, as it represents one of the major parameters used for the measurement of the quality of transmission in a digital communication system [51]. As previously anticipated, the data stream over the wireless channel can be degraded during transmission because of different reasons, such as thermal noise, distortion, bit synchronization error, or interference. In practical tests, the bit-error-rate or bit-error-ratio (BER) is the number of bits in error divided by the total number of transferred bits during a certain studied time interval (Equation 6.13). It is a unit-less parameter, often expressed as a percentage[52].

$$BER = \frac{BIT_{Error}}{BIT_{Total}} \tag{6.13}$$

where BIT_{Error} is the number of "misinterpreted bits", and BIT_{Total} is the total number of bits being transmitted, both measured during the so called "*gating time*" (ΔT).
Actually, the *bit error rate* represents the number of bit in error per unit time $[s^{-1}]$, while the *bit-error-ratio* is the true dimensionless parameter, following Eq. 6.13. Another term often used is *bit error probability*, which refers to the case in which infinitely many bits are transmitted, as it estimates the probability that any bit transmitted through the system will be received in error. However, in literature there is a tendency in mixing-up the definitions of *bit error probability* and *bit error rate*. To be more precise, the quality of the BER estimation increases as the total number of transmitted bits increases. In the limit, as the number of transmitted bits approaches infinity, the BER becomes a perfect estimate of the true error probability. This will be demonstrated in Chapter 7.

However, the performance of each modulation scheme is often evaluated by calculating its probability of bit error (P_b), which is the expected value of the BER, with the assumption that system operates under Additive White Gaussian Noise [52]. The greater the symbol alphabet, the higher the probability of error, as anticipated in Section 6.2.8 [51].

6.3.2 AWGN Channel

There exist various models of the channel in a wireless communication system [60]. AWGN channel represents the simplest one, as well as the usual starting point for the system performance assessment. The term "noise" is often used to define "unwanted electrical signals" that may interfere with the signal being transmitted, thus limiting the capability of the receiver to perform the correct symbol decision during the demodulation process. Thermal noise represents one of the main sources of performance degradation at the receiver [61]. It originates from the electron thermal motion in dissipative electrical components [62].
AWGN (Additive White Gaussian Noise) Channel is a mathematical model commonly used to simulate the thermal noise of the channel while transmitting a signal in a communication system [59], which is linear and time-invariant (LTI). The origin of the acronym AWGN can be clarified in the following [62]:

- **Additive**, as the noise gets "added", not multiplied, to the received signal, when the signal passes trough.
- **White**, noise power spectral density is a flat for all frequencies.
- **Gaussian**, as the amplitude of the noise follows the Gaussian probability distribution function p(z) reported in Equation 6.14, where σ is the variance, and a is the mean.

$$p_z = \frac{1}{\sigma^2\sqrt{2\pi}} \exp\left[\frac{1}{2}\left(\frac{z-a}{\sigma}\right)^2\right] \tag{6.14}$$

The expression for the noise spectral power density follows Eq. 6.15:

$$G_n(f) = \sigma^2 = \frac{N_0}{2} \tag{6.15}$$

where the factor of $1/2$ indicates that $G_n(f)$ is a two-sided power spectral density, where half the power is associated with positive frequencies and the other half with negative frequencies [62]. The received signal can be expressed as:

$$r(t) = s(t) + n(t) \tag{6.16}$$

where r(t) is the actual received signal, s(t) is the source signal at the transmitted side, and n(t) is noise.

6.3.3 Signal to Noise Ratio (SNR)

The Signal to noise ratio (SNR) is defined as the measurement of the amount of signal divided by the amount of noise being received. [52] It is usually expressed

in decibel (dB), through the following expression:

$$SNR = log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) \tag{6.17}$$

A high SNR is always desirable as it implies getting more signal and less power. BER and SNR are inversely related [63].

6.3.4 "Energy per bit-to noise power density ratio" ($\frac{E_b}{N_o}$)

In digital communication, the "energy per bit - to noise power spectral density ratio" ($\frac{E_b}{N_o}$) is a normalized signal-to-noise ratio (SNR) measure, also known as the "Signal to noise per bit". It is especially useful when comparing the bit error rate (BER) performance of different digital modulation schemes without taking bandwidth into account [62], and it is the parameter actually used to measure signal to noise ratio for a digital communication system.
In particular, E_b can be computed with the following equation:

$$E_b = \frac{P_{signal}}{R_b} \tag{6.18}$$

where P_{signal} is the signal power and R_b is the bit rate. Noise power instead is expressed in the frequency domain, and its spectrum density is constant in frequency, compared to signal. If BW is the receiver noise bandwidth, and R_s the symbol rate, it is possible to express the noise power (P_{noise}) at the receiver front-end as [57]:

$$P_{noise} = N_O \cdot BW \tag{6.19}$$

Therefore, the relationship between SNR and E_b/N_O can be written as:

$$\frac{P_{signal}}{P_{noise}} = \frac{E_b}{N_O}\frac{f_b}{BW} \tag{6.20}$$

6.3.5 Power efficiency

With reference to equation 6.20, the optimum is when $BW = R_s$, i.e. when the noise bandwidth is equal to the symbol rate. When BW is greater than the optimum R_s, the SNR is reduced, thus meaning that more signal power is necessary to maintain the same performance [57].
Power efficiency, sometimes referred to as "energy-efficiency" is defined as "the capability of a modulation technique to preserve the bit error probability of the digital message at low power levels" [64]. The amount by which the signal power should be increased to maintain a certain BER depends on the modulation scheme. Power efficiency is often expressed as the Eb/No required at the input of the receiver for a specified probability of error (e.g. 10-6).

6.3.6 Bandwidth-Efficiency

Bandwidth efficiency is "the ability of a modulation technique to accommodate data within a limited bandwidth" [64], and it explains how proficiently the allocated bandwidth is utilized. In general, increasing the data rate implies reducing the duration of the digital symbol, which increases the bandwidth of the signal being transmitted. This demonstrates that there exists a relationship between data rate and bandwidth occupancy. Bandwidth efficiency is defined as "the ratio of the throughput data rate per Hertz ($\frac{bps}{Hz}$) in a given bandwidth". Modulation techniques with a higher bandwidth efficiency are capable to transmit more data in the allocated bandwidth [65]. So higher order modulation schemes are characterized by a higher bandwidth efficiency than binary ones.
There exists a theoretical limit to the bandwidth efficiency for each type of modulation scheme, and this limit refers to the "**Shannon Capacity Theorem**".

Shannon Capacity Theorem

For a specified digital communication system being studied, there are two different upper limits to the performance:

- one upper limit depending on the actual system.
- absolute upper limited associated *with each system*, which means that it represents a limit to the maximum rate at which data could be transmitted across a communication channel, that no digital communication system can exceed in any case. This theoretical upper limit was given by Shannon, in 1948.

In general, it can be stated that the closer a system comes to this upper limit, the better its performance.
For a channel corrupted by Additive White Gaussian Noise, the Shannon's bound is given by the following equation:

$$\eta_{B_{max}} = \frac{C}{B} = \log_2(1 + SNR) \tag{6.21}$$

where C is the channel capacity (bps), B is the bandwidth (Hz) and SNR is the signal-to-noise ratio.

6.3.7 Reducing the Bit-Error-Rate using an Error-control Coding Technique

Error control coding theory has been intensely studied since 1940s by a large variety of scientific disciplines, such as information theory, mathematics, electrical engineering and computer science. Today error control codes are being widely used in communication systems [66] to recover the information degraded by sorts of noise when the signal travelling through a communication channel, thus allowing for the enhancement of the system performance. They are used to increase the possibility of detecting and eventually correcting the error generated during the signal transmission over a noisy channel. The basic principle of error-control coding techniques consists in "*adding a certain degree of redundancy* as a form of **control digits** to the information carrying signal" [67], thus making the transmission of the

signal more robust to the disturbances in the communication channel. Typically, the channel encoder at the transmitter side adds these extra-bits, usually called "parity check bits", to the message bits. At the receiver end, the channel decoder interprets the received data, using the redundant symbols, and eventually correct the error occurred during the transmission. However, adding redundancy reduces the effective data rate through the channel, thus decreasing the bandwidth efficiency [68]. There exists a large variety of error-control coding, and the choice of the specific coding technique depends on the nature of the noise or data [67].

6.3.8 BW efficiency versus Power efficiency

Power efficiency and bandwidth efficiency usually counteract with each other, as it is hard to achieve good power and bandwidth performance at the same time. Very often there is trade-off between in the design of a digital communication system. In particular:

- adding error control coding reduces the bandwidth efficiency, as the added redundancy increases the bandwidth occupancy, but it makes the power efficiency increase.
- higher order modulation scheme increase the bandwidth efficiency, as they decrease the bandwidth occupacy, but require higher transmission power to keep the same BER, thus seriously affecting power-efficiency.

Chapter 7

Power-Error trade-off of digital modulation schemes for energy constrained applications

Similarly to the power and error analysis of digital mixing reported in Chapters 4 and 5, the final objective is the study of the power versus error trade-off in digital modulation. To this end, Section 7.1 is dedicated to a literature review of digital modulation schemes in the field of wireless micro-sensors application, as it is considered here as the "precursor" to the smart dust and body dust concepts, where energy efficiency is the most relevant FOM. This is essential to select a few modulation technique, as valuable candidates for the application under discussion. The last section (Section 7.2), is essentially a simulative part, which aims to investigate in depth the performances of the previously selected schemes, based on the definition of a method for the estimation Bit Error Rate (BER) in MATLAB.

7.1 Digital Modulation Schemes for Wireless micro-sensors applications

At present, a great variety of research works concerning the study of suitable communication strategies in the field of wireless micro-sensors applications exists. This chapter will be built upon this, while attempting to extend the whole study to the Body Dust concept.
Wireless micro-sensors are tiny and low-cost devices, equipped with sensing, processing and wireless communication capabilities. They are used in a large variety of applications, especially for those ones where deployment is difficult, wires impractical, and maintenance impossible [69]. The main characteristics of such devices can be listed in the following [70]:

- **relatively low data rate**, usually of the order of few K-bits per second.
- **transmission range of the order of few meters**, unlike other wireless applications.
- **energy efficiency** of the design of the communication systems for battery operating systems and even more when battery-less operation is targeted.
- for some specific bio-sensor applications, over-heating due to excess of power consumption leads to the damage of tissues [71].

In the case of Wireless Sensor Networks, which contain hundreds or thousands of sensing nodes, unlike the point-to-point communication, energy consumption in WSN can not be optimized by solely considering a sensor node, but considering overall network resources [72].

7.1.1 Energy Efficiency Requirement

It is evident that for such applications, the main requirement is to increase the autonomy of these small sensor node, which means that energy efficiency is a very important metric at all levels of system design. However, despite being the primary objective, it represents also the biggest challenge.
At the system level, energy efficiency must be achieved by considering:

- circuit complexity and circuit power consumption.
- power consumed for the signal transmission across the wireless channel.

Therefore, the choice of the modulation scheme has a significant impact on both the transceiver design and the demodulation circuit. For WSN, the transceiver has to remain as simple as possible and therefore only simple binary modulations are usually considered to ensure an energy-efficient demodulation [73].
It is well known that every design comes with a trade-off concerning **power, bandwidth or error**. Sensor applications characterized by very low data can afford to compromise on bandwidth requirements of the system, but definitely not on the power consumption [70]. This means that the best solution for the design of such systems consists in maximizing energy efficiency for the wireless communication systems at expense of bandwidth, while ensuring reasonably good error performance and keeping low the circuit complexity.
There are many research works in this field that discussed about how to bring energy efficiency both at system and circuit levels. However, the proposed research project considers only OOK and FSK digital modulations. The rationale behind this choice is discussed in the following.

7.1.2 On-Off-Keying

On-Off-Keying modulation represent a highly promising solution, especially in case of low data rates and extremely resource-constrained applications. An OOK transmitter consumes energy only when transmitting a high-bit, which is of great relevance as the energy used to transmit one single bit equals to the energy needed to perform thousand 32-bit calculations [74]. Moreover it consumes low power also a the circuit level, because of the reduced circuit complexity. Indeed, the circuit implementation for an OOK modulator is a switch, which makes this modulation scheme an excellent solution for miniaturization. However, like any ASK digital modulation scheme, it has a bad error performance. Typically, uncoded OOK requires double the power to to have the same performances of BPSK [70].

Minimum Energy Coding (MEC)

Minimum energy coding (MEC) represents an efficient way to improve further the advantage of OOK's feature of transmitting only 1-bits. The basic principle consists in (1) mapping the source data bits into *code-words* which have less 1 bits, and

(2) assign code-words with less number of high bits to the source message with a higher probability [75]. However, even if the average power during transmission is reduced, the bandwidth increases with the ME-coding, as a larger number of bits are actually used for the transmission. So, a trade-off must be done between power and bandwidth consumption of the system.
OOK modulation with ME-coding is widely employed in the design of "Wireless Nanosensor Networks" (WNSNs), which are collections of nanosensors with communication capabilities. Due to their extremely small size, nanosensor nodes can provide limited energy. This means that power and energy efficiency are the most critical system level metrics, especially for what concerns communication. Several research works in this field discussed the issues of minimizing power consumption when employing OOK and different possible ME-coding algorithm [76], [77], [78], [79]. This specific application can be of great interest as it share some common characteristics with the smart dust and body dust concepts.

7.1.3 Frequency Shift Keying

Another widely employed digital modulation scheme in wireless microsensor application is FSK. The main disadvantage of OOK modulation consists in the poor BER performance, compared to the other modulation schemes. In particular, to meet specific error performance, higher power may be required for the transmission of the signal across the wireless channel. FSK has proven to be a valuable trade-off, as it ensures better BER performance, and allows to reduce the power consumed in the circuit low.
Indeed, FSK has proven in different research works to be appropriate to reduce circuit cost and complexity with a high level of integration and great flexibility [80], [81], [82]. For example, an FSK modulator can be implemented by a simple direct modulation method such as a $\Sigma - \Delta$ modulator, in which power-hungry components such as DACs and quadrature VCOs are not needed [81]. There exist in literature several circuit implementation optimized to reduce the power consumption. However, the proposed thesis project aims to focus on a high level estimation of the performance of the selected digital modulation schemes.

7.2 BER comparison of FSK and OOK modulation using MATLAB

As stated in section 6.3.1, the generation of errors during data transmission across the wireless channel may compromise the integrity of a communication system. To this end, the estimation of the bit-error-rate (BER) would provide a reasonable way to asses the error performance of the system being studied, as it allows to analyse the "full end-to-end performance" of a system which includes (1) a transmitter (TX), (2) a receiver, and (3) the medium in between [83] [84]. This is of great relevance for the analysis developed in this chapter, whose main goal is to achieve a high-level estimation of the system performance rather than a low-level estimation approach. To model the effect of the noise degradation during the data transmission, it is assumed that FSK and OOK modulations operate under AWGN channel, which implies the use of statistical analysis techniques.

Before introducing the method employed, it is essential to recall some useful definitions. As anticipated in section 6.3.1, the bit-error-ratio and bit-error-probability represents two different ways of estimating the BER. However, in literature there is the tendency using these terms interchangeably. Actually, the manner is a perfect estimation of the latter only if the number of transmitted bits approaches infinity. Indeed, the bit-error probability (P_e) can be considered as a mathematical expression of the BER, while the bit error-ratio is an empirical method for the measurement of a system's actual bit error performance [85].
This analysis aims to assess the error performance of the above mentioned modulation schemes by first measuring the bit-error-ratio, and then comparing it with the theoretical model. For the definition on P_e, it is necessary to recall the definition of some useful variables, which are listed in the following:

- the error function, **erf**, which is different according to the type of modulation scheme being studied.
- the energy in one bit, E_b, which can be computed according to Eq.6.18, and it is expressed in $\left[\frac{W}{Hz}\right]$, i.e. in joule .
- the noise power spectral density, N_o, i.e. the noise power in 1 Hz of bandwidth, and therefore it has the dimension of power ($\left[\frac{W}{Hz}\right]$) per hertz, i.e. $[J]$.

It is to remark that the POE is proportional to the the ratio E_b/N_o, which is a dimensionless parameter, as all the physical dimensions of E_b and N_o cancel out with the computation.

7.2.1 The simulation tool: MATLAB

The tool used for the simulation of the BER of the selected digital modulation schemes is MATLAB. It is widely used for the simulation of digital communication systems, and it has been often employed in several research activities to simulate the performance of different modulation techniques [86], [87], [85], [88]. Testing the BER MATLAB is very simple and efficient, as a result of the easy scripting language and the excellent visualisation capabilities [83].

7.3 Method for the BER analysis

Figure 7.1 showcases the reference block diagram of the model of a digital communication system assumed in simulations. The analysis assumes a discrete-time representation of the signals involved. Following the flow suggested by the succession of the main building blocks of the block diagram in Fig. 7.1, the method employed for the estimation of the BER for both OOK and FSK can be summarized into the following steps:

1. Generation of the input bits
2. Bit-to-Symbol Mapping
3. Generation of the Noise
4. Summation of the noise to the mapped signal

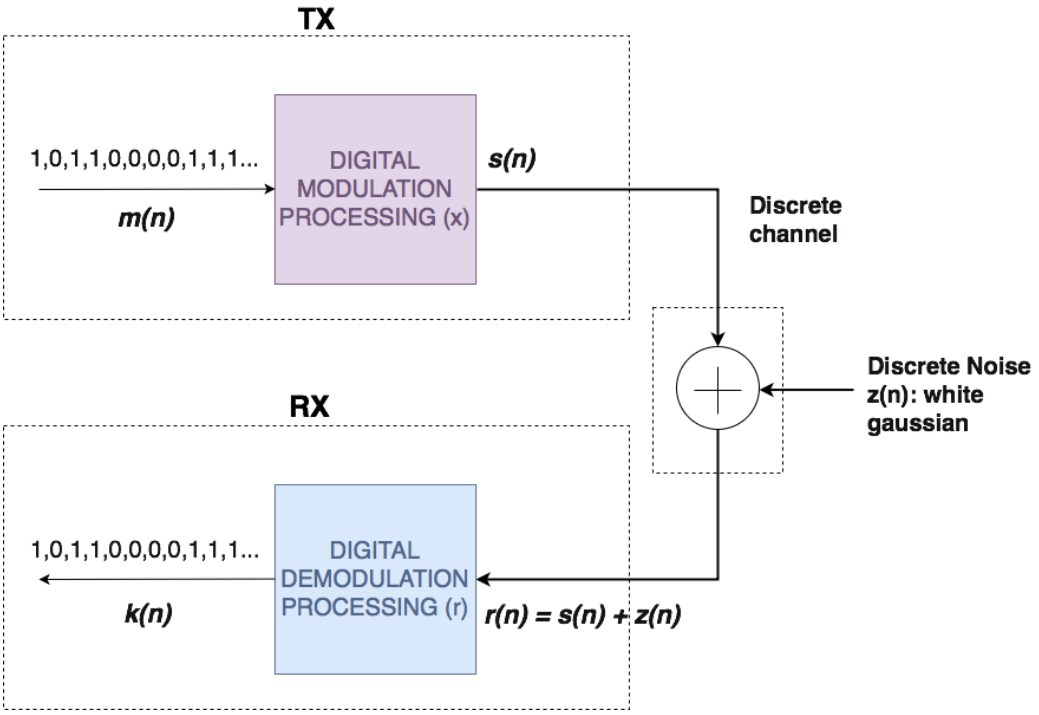


FIGURE 7.1: Digital modulation system block diagram including (1) TX, (2) channel, and (3) RX. The channel is assumed to be a discrete Gaussian noise channel.

5. Detection
6. BER estimation and plot
7. Comparison with the theoretical model

STEP 1: generation of the input bits

The signal to be transmitted in digital modulation consists of a bit stream $\{0,1\}$, which is actually a sequence of pseudo-random bits $x(n)$.

STEP 2: Bit-to-Symbol Mapping

This step represents the actual modulation process at the transmitter side, where the input bit sequence is converted into a symbol sequence ($s(n)$). It is important to remark that modulation symbols are in general complex number, with real and imaginary parts, but discrete in time. To explain in depth the bit-to-symbol mapping procedure, it is convenient to make use of constellation diagrams, in order to appreciate how these symbols are mapped to a discrete set of magnitude and phase values.

STEP 3-4: Noise addition

The effect of the noisy channel of the transmitted signal is simulated by adding an amount of noise to it. Under the assumption of a discrete-time representation, this is carried out by generating a noise vector ($z(n)$), which is then added to the

signal vector ($s(n)$), i.e. to the symbol sequence. The signal at the receiver input is a vector, which can be computed according to the following equation:

$$r(n) = s(n) + z(n) \tag{7.1}$$

STEP 5: Detection

The noisy signal ($r(n)$) is a sequence of received symbols. These passes through the demodulation block at the receiver and produces a bit stream ($k(n)$), representing the retrieved information. This process is performed by setting a "threshold", which is specific for each modulation scheme, and based on this a 0 or a 1 is generated.

STEP 6: BER estimation and final plot

In the final stage the transmitted bit stream is compared to the received one, while counting the error performed. At this step the bit-error-ratio is computed to estimate the final BER.
BER performance is usually represented on a two dimensional graph, where the abscissa is the normalized signal-to-noise ratio, i.e. E_b/N_o, expressed in decibels (dB), while the ordinate is the BER, which is a dimensionless quantity usually expressed in powers of ten. This plot is generated, running a simulation for each value of SNR in a specified array.

STEP 2, 3, 4, and 5 can change according to the specific modulation scheme being simulated.

7.3.1 BER simulation procedure for OOK

Figure 7.2, shows the block diagram model of a OOK-based communication system, with discrete AWGN channel.

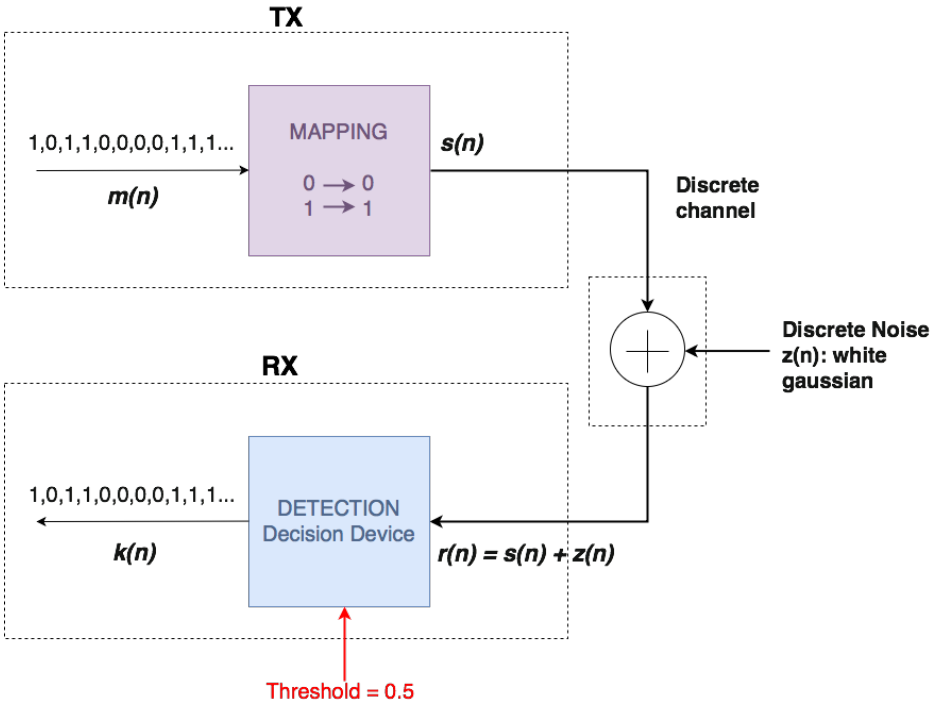


FIGURE 7.2: Reference block diagram of a digital communication system based on OOK modulation, assuming discrete AWGN channel.

Bit-to-Symbol mapping

In the case of OOK modulation, 1-bits are represented by the reference symbol voltage 1, while 1-bits by zero voltage, as depicted in Fig. 7.2. Each modulation scheme can be represented in the constellation diagram, as specified in Section 6.2.5.
Fig. 7.3 refers in particular to the constellation diagram of OOK modulation, where the dashed line circles represent the additive noise around the symbols being transmitted [89]. OOK is a 1D (1 dimensional) modulation technique, meaning that no Q-component is present [1], so the constellation points are on the real axis of the graph. Due to the noisy channel, the received symbols can deviate from the original position in the constellation diagram. This means that a decision device is required at the receiver side to interpret the original digital information carried by the received symbols. This is achieved by setting a threshold and comparing it the incoming noisy symbols. Therefore, the threshold choice is critical to the performance of the receiver.

[1] In other words, with reference to Section 6.2.5 the complex envelope has zero imaginary part

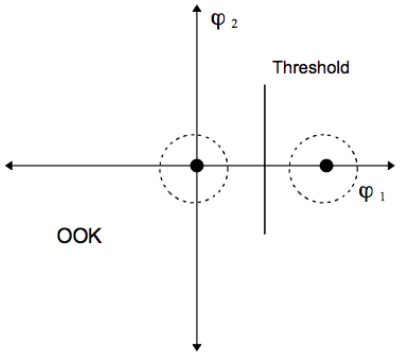


FIGURE 7.3: Representation of the bit-to-symbol mapping in the constellation diagram for OOK modulation

Noise generation

The objective is to obtain a plot of the BER versus SNR. The expression to compute SNR in the case of OOK modulation is:

$$SNR = \frac{signal \quad power}{noise \quad power} = \frac{E_b}{\frac{N_0}{2}} \tag{7.2}$$

As previoulsy anticipated, OOK modulation is a 1D (one-dimensional) modulation. This means that there is no Q-component. So, for the I-component it is accounted a noise of $\frac{N_0}{2}$. The idea is to define an array in Matlab with different values of the SNR, and then, assuming a E_b equal to 1, let the noise power (N_0) vary and run a simulation for each value of N_0. Therefore, it is possible to rewrite Eq. 7.2 as:

$$\frac{E_b}{\frac{N_0}{2}} = \frac{1}{\sigma^2} \quad \rightarrow \quad \sigma^2 = \frac{1}{\frac{2E_b}{N_0}} \quad \rightarrow \quad \sigma = \sqrt{\frac{1}{\frac{2E_b}{N_0}}} \tag{7.3}$$

where σ is the standard deviation. It is essential to remember that, since the noise has zero mean, its power and its variance are identical. Then it is essential to generate a noise vector that has the same length as our signal vector s(n), and this noise vector must have variance σ. This can be achieved with the following MATLAB command:

$$z = \sigma * randn(1, N) \tag{7.4}$$

where N is the length of m(n) [2].

Detection

Once the noise is added to the signal s, the decision device at the receiver side has to demodulate the signal coming from the channel. For OOK, it is sufficient to set a threshold to the value of 0.5, to determine whether the transmitted signal has been

[2]Note that the MATLAB function *randn* generates normally distributed random numbers with a mean of zero and a variance of one. So the output of this function has to be scaled in order to have the desired variance.

interpreted has a 1-bit or as a 0-bit.

After the detection, there are two sets of bits, the input bits (m) and the detected ones (k), and by comparison it is possible to determine how many received bits are in error.

7.3.2 BER simulation procedure for FSK

Figure 7.2, shows the block diagram model of a FSK-based communication system, with discrete AWGN channel.

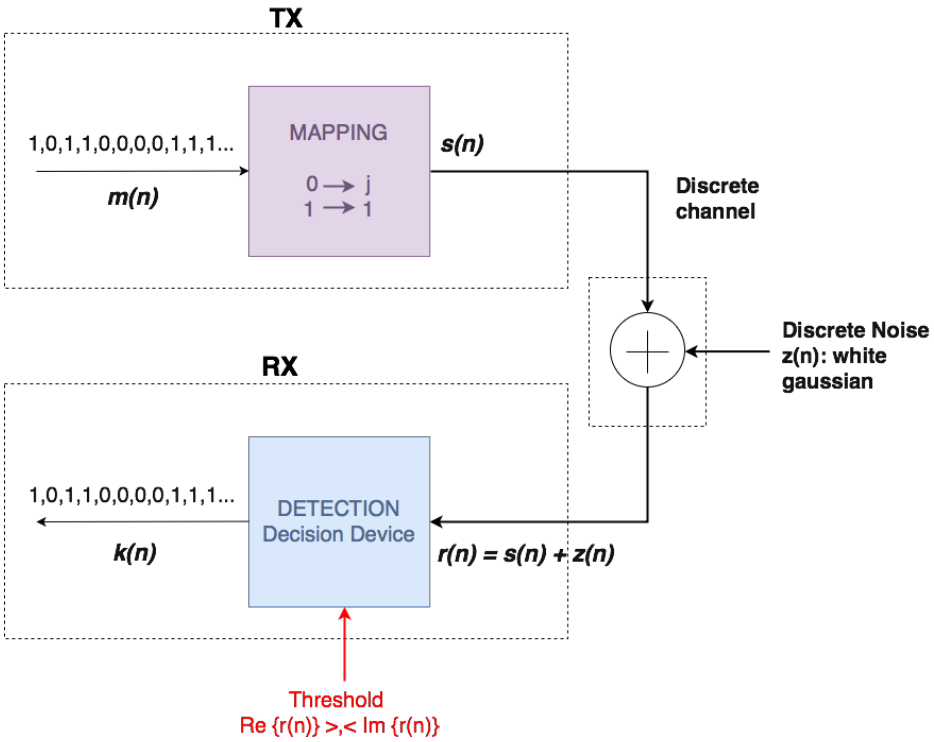


FIGURE 7.4: Reference block diagram of a digital communication system based on FSK modulation, assuming discrete AWGN channel.

Bit-to-Symbol mapping

Fig. 7.5 refers to the constellation diagram of orthogonal BFSK modulation [90], where again the dashed line circles represent the additive noise around the symbols being transmitted [89]. FSK is a 2D (2 dimensional) modulation technique, meaning that both I and Q-components are present, so the 1-bit will be mapped to symbol 1 and 0-bit to symbol j (Figure 7.4).

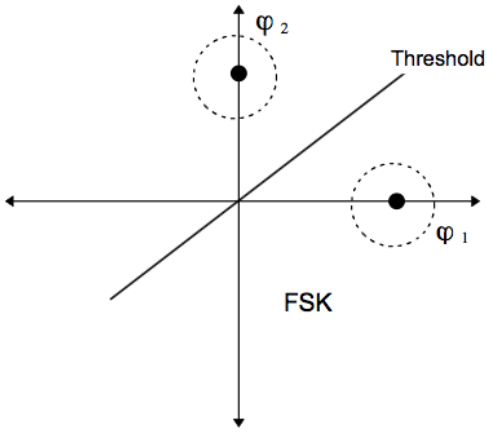


Figure 3

FIGURE 7.5: Representation of the bit-to-symbol mapping in the constellation diagram for FSK modulation

Noise generation

Being FSK a 2D modulation scheme, both I and Q components generates a power noise contribution equal to $\frac{N_0}{2}$. So, overall the noise has zero mean and N_0 variance. Assuming E_b equal to 1, from the predefined array of SNR, it is possible to evaluate the variance as:

$$\sigma = \sqrt{\frac{1}{\frac{E_b}{N_0}}} \tag{7.5}$$

The noise alter both the real and the imaginary parts of each symbol being transmitted, meaning that the noise is generated as a complex number as well. The corresponding MATLAB command will be:

$$z_0 = \frac{1}{\sqrt{2}} * [randn(1, N) + j * randn(1, N)] \tag{7.6}$$

The reason of the presence $1/\sqrt{2}$ comes from the fact that both the imaginary and real parts have a variance of $\frac{N_0}{2}$. From eq. 7.7 you can observe that, since the standard deviation is the square root of σ, when multiplying z_0 by the standard deviation, the only way to obtain a number with zero mean and variance of 1 ($\frac{1}{2} + \frac{1}{2}$) is to introduce that factor.

$$z = \sigma * z_0 \tag{7.7}$$

Detection

The last detail regards the decision device. The received signal is complex and it can be expressed as:

$$r = s + \sigma * z_0 \tag{7.8}$$

If the signal transmitted m is equal to 1, it can be rewritten as:

$$s = 1 + j \cdot 0 \tag{7.9}$$

With some noise:

$$r = (1 + z_1) + j \cdot 0 + j \cdot z_2 \tag{7.10}$$

Since z_1 and z_2 are approximately equal, it is possible to state that the detected signal is a 1 if it holds that:

$$\mathrm{Re}(z) > \mathrm{Im}(z) \tag{7.11}$$

For the zero detection the case is dual:

$$\mathrm{Re}(z) < \mathrm{Im}(z) \tag{7.12}$$

The expressions in the Eq. 7.11 and Eq. 7.12 represent actually the thresholds of the detection device.

7.3.3 Comparison OOK vs FSK: output plots

Figures 7.6 and 7.7 shows the final plots of the simulations in MATLAB. Each curve is compared with the corresponding theoretical model, represented by the erf (error function). With no surprise, FSK performs better in AWGN channels, matching the actual expectations. Moreover, given a certain required BER performance, which can be represented by the red straight line in Fig. 7.8, OOK modulation requires higher E_b/N_o, which implies higher power consumption for the signal transmission compared to FSK.
Moreover, when increasing the number of samples, the estimation of the BER is more accurate and the corresponding curve gets closer to the model, thus demonstrating that the bit-error-ratio represents a perfect estimation of the bit-error-probability only when the number of bit transmitted approaches infinity.

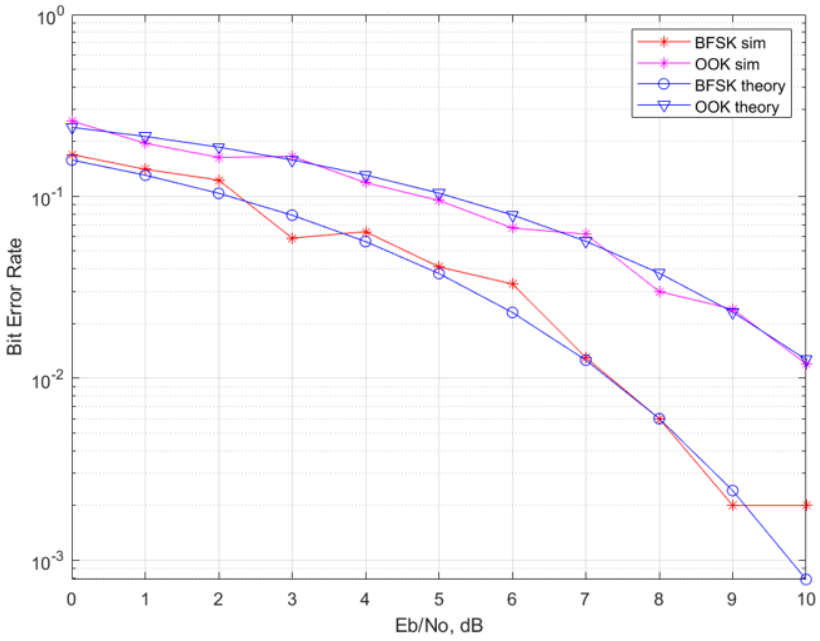


FIGURE 7.6: BER versus normalized SNR plots for both FSK and OOK. For each modulation scheme the theoretical BER is also plotted. Number of samples=1000

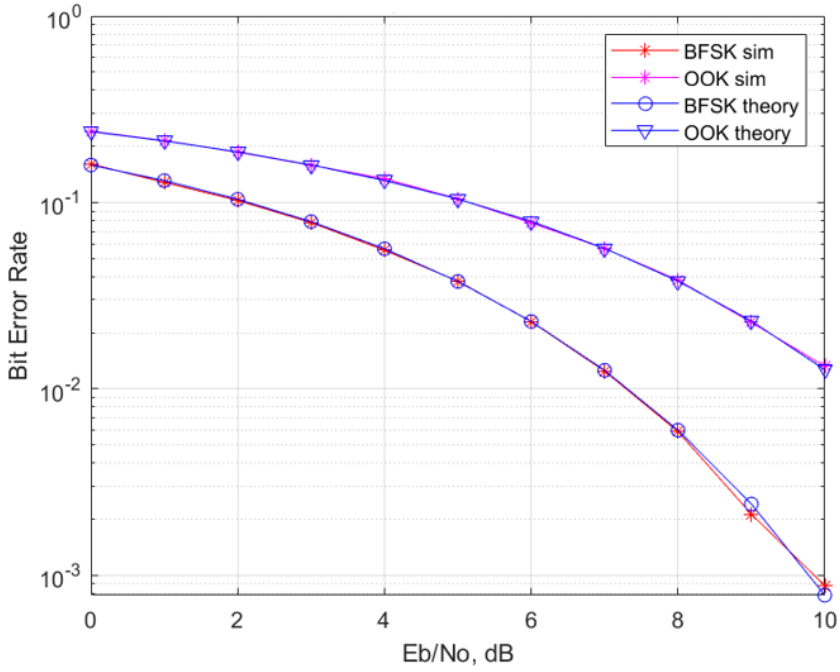


FIGURE 7.7: BER versus normalized SNR plots for both FSK and OOK. For each modulation scheme the theoretical BER is also plotted. Number of samples=1000

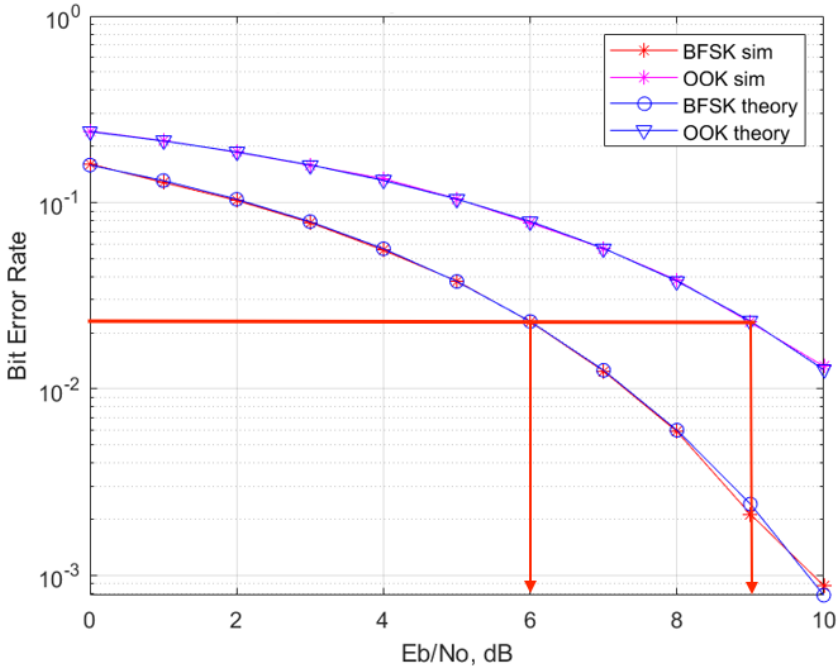


FIGURE 7.8: BER versus normalized SNR plots for both FSK and OOK. Number of samples=1000. The red lines demonstrates that to achieve the same BER, OOK requires higher E_b/N_o than FSK.

Chapter 8

Conclusion and future work

The proposed research project examined the feasibility of communication methods for the next-generation of wearable and implantable devices through a separate study of a (1) novel multiple-input communication method based on digital mixing, and (2) suitable digital modulation schemes. In both the cases, the main system level metrics for the analysis were power (and energy) efficiency and error performance.
In (1), a preliminary time-base analysis demonstrated that the main sources of error in digital mixing are the frequency ratio of the input signals and their relative phase shift. A novel test method based on automated VHDL functional simulations has proven to be highly accurate for the characterization of the error performances concerning the logical circuit behaviour. In particular, the analysis of the results collected showcased that the error performed on the input signal characterized by a lower frequency exhibits an exponential (decreasing) dependency on the input frequency ratio, while the phase shifts impact significantly such trend towards lower values of the frequency ratio. If on one hand increasing the the input frequency ratio represents a safe solution to improve the system error performance, on the other hand it may impact significantly the power consumed at the circuit level, as the dynamic power represents a frequency dependent contribution. With the aim of outlining the best power versus error trade-off in digital mixing, the circuit has been synthesized for different combination of the input frequencies with the aid of another automated procedure. The results collected from the power reports demonstrated that the dynamic power consumed in the mixer has a linear dependency on the input frequency ratio. The combination of the error and power plots allows for the research of the global minima, which represents the intersection points of the error and power curves, where the system is optimized for both power and error performance in therms of input frequency levels.
The aim of (2) consists in providing the means for the choice of a suitable modulation scheme for the class of devices under discussion. Due to the lack of research works in this field, this project analysed through a preliminary literature review the most common schemes employed in wireless microsensor application, which is conceived as a sort of "precursor" of the smart dust and body dust concept, where energy efficiency represents the major system level metric. In particular, energy efficiency must be brought both at the system and a the circuit level. This preliminary review showcases that OOK and FSK have been often employed to meet such requirement. On one hand OOK allowed for significant power savings at the system level, including both the power consumed in the circuit and the power required for the signal transmission. On the other hand, FSK ensured better BER performance and reduced power consumption at the circuit level. The final BER

performance analysis with the aid of MATLAB simulations demonstrated that to perform as much as FSK, OOK requires higher power to transmit the signal under Additive White Gaussian Noise (AWGN) channel.

8.1 Future work

8.1.1 Digital mixing

There are several open issues concerning digital mixing that can be study in the future research work. They can be listed as follows:

- characterize the effect of phase jitter on the error performance with reference to the D-FF circuit implementation. This would imply further constraints on the design of the input frequency levels.
- study of the impact of the choice of a specific technology, which would be suitable for the circuit implementation in terms of circuit are and power consumption. This could be coupled with a more accurate estimation of the power consumption at the circuit and RTL levels.
- study of the effect of the reduction of the duty cycle of the reset signal in the power consumption at the circuit level, still with reference to the D-FF circuit implementation.
- seek viable solutions to increase the number of inputs in digital mixing.

8.1.2 Digital Modulation

The future work concerning digital modulation schemes in this field implies a low level estimation of the system performance concerning both power and error. This requires to think in terms of the actual circuit implementation concerning both FSK and OOK. For what concerns OOK, it can be studied whether the implementation of ECC or ME-coding techniques can further improve the BER performance and the system power consumption. Indeed, this has a trade-off with the circuit complexity required for the implementation of the specific coding algorithm, which can increase in turns the power consumed at the circuit level.

Appendix A

Scripts for Test Automation

This appendix reports the most relevant scripts used to set and run automated functional simulations with Modelsim.

A.1 Script 1: Set the input parameters

To symulate the system for different combinations of clock period, reset period, and relative phase shift a dedicated bash script is employed to manipulate automatically and iteratively such parameters in the testbench. The code below, which refers to the "*conf_script.sh*", summarize the relevant steps required for the test manipulation. Moreover, in the inner loop, another script is run ("*run_script.sh*"). This script is discussed in section A.2.

```
#!/bin/bash

source /software/scripts/init_msim6.2g

cp tb_receiver.vhd tb_receiver_gold.vhd
rm tb_receiver.vhd

list_reset=(10 5 3.3 2.5 2 1.67 1.43 1.25 1.11 1.0)
list_ck=(50 20.74 8.6 3.57 1.48 0.6135 0.2544 0.1055 0.0438 0.0181)

for i in "${list_reset[@]}"
do
  sed -e 's/constant\ TB\_RESET\_PERIOD\ :\ time\ :=\ '.*'\ ms;/
    constant\ TB\_RESET\_PERIOD\ :\ time\ :=\ '$i'\ ms;/'
    tb_receiver_gold.vhd > tb_receiver_tmp1.vhd

  for j in "${list_ck[@]}"
  do
    sed -e 's/constant\ TB\_CK\_PERIOD\ :\ time\ :=\ '.*'\ us;/
    constant\ TB\_CK\_PERIOD\ :\ time\ :=\ '$j'\ us;/'
    tb_receiver_tmp1.vhd > tb_receiver_tmp2.vhd

    for k in {0..8}
    do
      h=$( echo 'scale=4;'${j}'*'${k}'/8' | bc | sed 's/^[.]/0./' )
      sed -e 's/constant\ phase\_shift\ :\ time\ :=\ '.*'\ us;/
    constant\ phase\_shift\ :\ time\ :=\ '$h'\ us;/' tb_receiver_tmp2.
    vhd > tb_receiver.vhd

      rm -r work
      ./run_script.sh $i
```

```
26        mv output_results.txt ./output_files/output_results_${i}_${j}_$
      {h}.txt
27
28      done
29    done
30
31 done
```

LISTING A.1: conf_script.sh

A.2 Script 2: Configure the simulator under the UNIX environment

The bash script "*run_script.sh*" is used to configure and run the VHDL simulator (i.e. Modelsim) under UNIX environment. The code is reported below.

```
1 #!/bin/bash
2 DURATION=$( echo 'scale=4;'$1'*22' | bc | sed 's/^[.]/0./' )
3 #DURATION=$(( $1*22 ))
4 vsim -c -do    "vlib work;
5 do script.do;
6 run ${DURATION} ms;
7 quit"
```

LISTING A.2: run_script.sh

A.3 Script 3: Define the steps of the simulations

The following code refers to the tcl script "*script.do*", which is used to define the operation required for the simulation.

```
1 vcom -reportprogress 300 -work work D_FF.vhd dff_ed.vhd
      edge_detector.vhd fast_frequency_detector.vhd
      slow_frequency_detector.vhd receiver.vhd tb_receiver.vhd
2 vsim -t 1ns -novopt work.TBRECEIVER
```

LISTING A.3: script.do

Appendix B

Scripts for Synthesis Automation

This appendix reports the most relevant scripts used to set and run automated synthesis with Synopsys.

B.1 Script 1: Set the input parameters and configure Synopsis under the UNIX environment

To synthesize the system for different combinations of clock and reset periods, the bash script "*conf_synth_script.sh*" is used and its code is reported in the following. In particular, it is used to manipulate the values of the two periods in the "*run_script.scr*", which is discussed in section B.2.

```
#!/bin/bash

#source /software/scripts/old/init_synopsys


list_reset=(1000000000 500000000 330000000 250000000 200000000
    167000000 143000000 125000000 111000000 100000000)
list_ck=(5000000 2074000 860000 357000 148000 61350 25440 10550 4380
    1810)

for i in "${list_reset[@]}"
do
  sed -i -e 's/create\_clock\ -name\ Clk\ -period\ '.*'/create\_clock
    \ -name\ Clk\ -period\ '$i'\ \{Clk\}/' run\_script.scr

  for j in "${list_ck[@]}"
  do
    sed -i -e 's/create\_clock\ -name\ Reset\ -period\ '.*'/create\
    _clock\ -name\ Reset\ -period\ '$j'\ \{Reset\}/' run\_script.scr
    sed -i -e 's/report\_power\ '.*'/report\_power\ \>\ .\/reports\/
    power'$i'\_'$j'.txt/' run\_script.scr
    dc_shell-xg-t -f run\_script.scr

  done

done
```

LISTING B.1: conf_synth_script.sh

B.2 Script 3: Define the steps of the simulations

The script "*run_script.scr*", whose code is reported below, is used to define the operation needed for the synthesis when working in batch mode with the EDA tool.

```
1
2 analyze -library WORK -format vhdl {/home/ms20.22/cap1/syndff/
    D_ff.vhd}
3 elaborate D_FF -architecture BEHAV -library WORK
4 ##setting constraints
5 create_clock -name Clk -period 100000000 {Clk}
6 create_clock -name Reset -period 1810 {Reset}
7 compile -exact_map
8 report_power > ./reports/power100000000_1810.txt
9
10 quit
```

LISTING B.2: run_script.scr

Bibliography

[1] S. Yury and E. Samoylova. "The internet of things as socio-technological institution of civil society in post-informational era". In: *2017 2nd International Conference on Computer and Communication Systems (ICCCS)* (2017), pp. 142–145.

[2] X. Li and L. Da Xu. "A Review of Internet of Things — Resource Allocation". In: *IEEE Internet of Things Journal* (2020), pp. 1–1.

[3] D. Serpanos and M. Wolf. "Internet-of-Things (IoT) Systems [electronic resource]: Architectures, Algorithms, Methodologies". In: *Springer International Publishing* (2018).

[4] F. Hu, D. Xie, and S. Shen. "On the Application of the Internet of Things in the Field of Medical and Health Care". In: *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing* (2013), pp. 2053–2058.

[5] S. Carrara. "Body Dust: Well Beyond Wearable and Implantable Sensors". In: *IEEE Sensors Journal* (2020), pp. 1–1.

[6] J. M. Kahn, R. H. Katz, and K. S. J. Pister. "Emerging challenges: Mobile networking for "Smart Dust"". In: *Journal of Communications and Networks* 2.3 (2000), pp. 188–196.

[7] B. Warneke et al. "Smart Dust: communicating with a cubic-millimeter computer". In: *Computer* 34.1 (2001), pp. 44–51.

[8] I. Chatzigiannakis and S. Nikoletseas. "A sleep-awake protocol for information propagation in smart dust networks". In: (2003), 8 pp.-.

[9] Ben Cook, Steven Lanzisera, and Kristofer Pister. "SoC issues for RF smart dust". In: *Proceedings of the IEEE* 94(6) (July 2006), pp. 1177–1196.

[10] Bryan Atwood, Brett Warneke, and Kristofer Pister. "Preliminary Circuits For Smart Dust". In: (Mar. 2001).

[11] Dongjin Seo et al. "Neural Dust: An Ultrasonic, Low Power Solution for Chronic Brain-Machine Interfaces". In: *Available at: ArXiv.org/abs/1307.2196.* (July 2013).

[12] Alexander Bertrand et al. "Beamforming approaches for untethered, ultrasonic neural dust motes for cortical recording: A simulation study". In: *236th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2014 (Aug. 2014), pp. 2625–8.

[13] Dongjin Seo et al. "Model validation of untethered, ultrasonic neural dust motes for cortical recording". In: *Neuroscience Methods* 244 (2015), pp. 114–122.

[14] Ryan M Neely et al. "Recent advances in neural dust: towards a neural interface platform". In: *Current Opinion in Neurobiology* 50 (2018). Neurotechnologies, pp. 64–71.

[15] et al. Seo. "Wireless Recording in the Peripheral Nervous System with Ultrasonic Neural Dust". In: *Neuron* (2016), pp. 529–539.

[16] et al. Cagnan H. "Emerging technologies for improved deep brain stimulation". In: *Nat Biotechnol. 2019 Oct;* (2019), pp. 1024–1033.

[17] D.G Hargreaves, S.J Drew, and R Eckersley. "Kirschner Wire Pin Tract Infection Rates: A randomized controlled trial between percutaneous and buried wires". In: *The Journal of Hand Surgery: British European Volume* 29.4 (2004), pp. 374–376.

[18] Mesut Sahin and Victor Pikov. "Wireless Microstimulators for Neural Prosthetics". In: *Critical reviews in biomedical engineering* 39 (Jan. 2011), pp. 63–77.

[19] Amanda Singer et al. "Magnetoelectric Materials for Miniature, Wireless Neural Stimulation at Therapeutic Frequencies". In: *Neuron* 107.4 (2020), 631–643.e5.

[20] Sandro Carrara and Pantelis Georgiou. "Body Dust: Miniaturized Highly-integrated Low Power Sensing for Remotely Powered Drinkable CMOS Bioelectronics". In: (2018).

[21] J. Snoeijs, P. Georgiou, and S. Carrara. "CMOS body dust — Towards drinkable diagnostics". In: *2017 IEEE Biomedical Circuits and Systems Conference (BioCAS)* (2017), pp. 1–4.

[22] Y. Liao et al. "A 3-μW CMOS Glucose Sensor for Wireless Contact-Lens Tear Glucose Monitoring". In: *IEEE Journal of Solid-State Circuits* 47.1 (2012), pp. 335–344.

[23] X. Zhou W. Liu K. Huang and S. Durrani. "Next Generation Backscatter Communication: Theory and Applications". In: *IEEE Journal of Solid-State Circuits* (Jun. 2018).

[24] N. Van Huynh et al. "Ambient Backscatter Communications: A Contemporary Survey". In: *IEEE Communications Surveys Tutorials* 20.4 (2018), pp. 2889–2922.

[25] C. Yao et al. "Backscatter technologies and the future of internet of things: Challenges and opportunities". In: *Intelligent and Converged Networks* 1.2 (2020), pp. 170–180. DOI: 10.23919/ICN.2020.0013.

[26] Gian Luca Barbruni et al. "Body Dust: Ultra-Low Power OOK Modulation Circuit for Wireless Data Transmission in Drinkable sub-100m-sized Biochips". In: (2019). arXiv: 1912.02670 [physics.ins-det].

[27] P. M. Ros et al. "Low-power architecture for integrated CMOS bio-sensing". In: (2017), pp. 1–4.

[28] K. P. Roby. "Flip-flop measures frequency difference between two signals". In: *Electronic Design* 17 (1969), pp. 101–103.

[29] S. Bruckenstein and M. Shay. "Experimental aspects of use of the quartz crystal microbalance in solution". In: *Electrochimica Acta* 30.10 (1985), pp. 1295–1300.

[30] I. Shankar, S. A. Morris, and C. G. Hutchens. "A novel frequency measurement technique for quartz microbalance systems and other resonator-based sensor system". In: *2nd ISA/IEEE Sensors for Industry Conference* (2002), pp. 134–138.

[31] M. P. Srinivasan L. Shivalingappa and S. Mohan. "Digital mixer for quartz crystal thickness monitor". In: *Chinese J. of Sci. instrument* 47 (1996), pp. 87–89.

[32] J. C. Bennett D. I. Shaw and A. M. Clements. "Analysis of 'D type' flip-flop frequency changers". In: *Electronics Letters* 26 (1990), pp. 1995–1997.

[33] D. Jinfeng Z. Weiyu and Z. Guoxiong. "Extended condition for measuring the differential frequency by D flip-flop". In: *Chinese J. of Sci. instrument* 28 (2007), pp. 260–263.

[34] Dong Jinfeng et al. "Study on the output characteristics of a D flip-flop differential mixer". In: *Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)* (2011), pp. 1552–1555.

[35] W. Zhang et al. "Choosing suitable logic gate mixer for measuring frequency difference". In: 26 (Jan. 2011), pp. 63–68.

[36] N V Kirianaki et al. "Data Acquisition and Signal Processing for Smart Sensors". In: *Measurement Science and Technology* 13.9 (Aug. 2002), pp. 1501–1501.

[37] D. Ma, S. S. Ghoreishizadeh, and P. Georgiou. "DAPPER: A Low Power, Dual Amperometric and Potentiometric Single-Channel Front End". In: *2020 IEEE International Symposium on Circuits and Systems (ISCAS)* (2020), pp. 1–5.

[38] V. A. Pedroni. "Simulation with VHDL Testbenches". In: *Circuit Design with VHDL* (2004), pp. 241–274.

[39] L. Molnar and A. Gontean. "Functional Simulation methodes". In: *2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC)* (2016), pp. 198–202.

[40] Lukasz Butkowski and Meric Karakurt. "CO-SIMULATION OF HDL USING PYTHON AND MATLAB OVER TCL TCP/IP SOCKET IN XILINX VIVADO AND MODELSIM TOOLS". In: (Oct. 2019).

[41] J. Cox et al. "Synchronizers and Data Flip-Flops are Different". In: (2015), pp. 19–20.

[42] "Chapter 8 - Acquisition of Asynchronous Data". In: (2015). Ed. by Hubert Kaeslin, pp. 445–472.

[43] I. Bulatova T. Grzes V. Salauyou. "Power estimation methods in digital circuit design". In: *Optoelectronics, Instrumentation and Data Processing* 45 (2009), pp. 576–583.

[44] Wolfgang Nebel. "Power Estimation at the Logic Level". In: (1997), pp. 135–177.

[45] Marilyn Wolf. "Chapter 3 - Logic". In: (2019), pp. 41–63.

[46] Paul J.M Havinga and Gerard J.M Smit. "Design techniques for low-power systems". In: *Journal of Systems Architecture* 46.1 (2000), pp. 1–21. ISSN: 1383-7621.

[47] Y. Nasser et al. "RTL to Transistor Level Power Modeling and Estimation Techniques for FPGA and ASIC: A Survey". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 40.3 (2021), pp. 479–493.

[48] Chi-Ying Tsui et al. "Power estimation methods for sequential logic circuits". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 3.3 (1995), pp. 404–416.

[49] B Dilip, Surya Prasad, and R Bhavani. "LEAKAGE POWER REDUCTION IN CMOS CIRCUITS USING LEAKAGE CONTROL TRANSISTOR TECHNIQUE IN NANOSCALE TECHNOLOGY". In: *International Journal of Electronics Signals and Systems* (Jan. 2012).

[50] Inc Synopsys. "Using Tcl With Synopsys Tools". In: (Dec. 2007).

[51] Andreas F Molisch. "Wireless communications". In: 34 (2012).

[52] T. Patra and S. Sil. "Bit error rate performance evaluation of different digital modulation and coding techniques with varying channels". In: (2017), pp. 4–10.

[53] Md Proloy, Razaul Hoq, and Syed Mushfiq Ahmed. "Comparative analysis of various wireless digital modulation techniques with different channel coding schemes under AWGN channel". In: (2016).

[54] Dr K Sharma, A Mishra, and Rajiv Saxena. "Analog & Digital Modulation Techniques: An overview". In: *International Journal of Computing Science and Communication Technologies* 3.1 (2010), p. 2007.

[55] Manoj Barnela and Dr Suresh Kumar. "Digital modulation schemes employed in wireless communication: A literature review". In: *International Journal of Wired and Wireless Communications* 2.2 (2014), pp. 15–21.

[56] Jaspal Bagga and Neeta Tripathi. "Study and comparison of various modulation classification techniques under noisy channel conditions". In: *IJETAE, ISSN* (2012), pp. 2250–2459.

[57] Sung-Moon Michael Yang. "Modern digital radio communication signals and systems". In: (2018).

[58] Harpreet Kaur Channi. "A Comparative Study of Various Digital Modulation Techniques". In: (Mar. 2016).

[59] Md Razaul, Syed Mushfiq, and Mohammad Asif Hossain. "Comparative Analysis of Various Wireless Digital Modulation Techniques with different Channel Coding Schemes under AWGN Channel". In: *International Journal of Computer Applications* 161 (Mar. 2017), pp. 30–34.

[60] Vinay Panwar and Sanjeet Kumar. "Bit error rate (ber) analysis of rayleigh fading channels in mobile communication". In: *International Journal of Modern Engineering Research (IJMER) Vol* 2 (2012).

[61] Golam Sadeque et al. "Bit error rate (BER) comparison of AWGN channels for different type's digital modulation using matlab Aimulink". In: *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)* 13.1 (2015), pp. 61–71.

[62] Rashmi Mongre and Monika Kapoor. "Comparative Analysis of Different Digital Modulation Techniques on the Basis of Their Bit Error Rate in VHDL". In: (2014).

[63] A Sudhir Babu and Dr KV Sambasiva Rao. "Evaluation of BER for AWGN, Rayleigh and Rician fading channels under various modulation schemes". In: *International Journal of Computer Applications* 26.9 (2011), pp. 23–28.

[64] Charles U Ndujiuba, Oluyinka Oni, and Augustus E Ibhaze. "Comparative analysis of digital modulation techniques in LTE 4G systems". In: *Journal of Wireless Networking and Communications* 5.2 (2015), pp. 60–66.

[65] Eric Sackey. "Performance evaluation of M-ary frequency shift keying radio modems via measurements and simulations". In: (2006).

[66] Dixit Dutt Bohra and Avnish Bora. "Bit error rate analysis in simulation of digital communication systems with different modulation schemes". In: *International Journal of Innovative Science* 1.3 (2014).

[67] Dauda E Mshelia, Peter Y Dibal, and Samuel Isuwa. "Reducing the bit error rate of a digital communication system using an error-control coding technique". In: *Int. J Sci. Eng. Res* 8.4 (2017), pp. 625–628.

[68] Akanksha Tiwari and VK Sahu. "Error correction channel coding practices for wireless communication systems". In: *Int. J. Adv. Res. Comput. Commun. Eng.* 4.10 (2015), pp. 151–159.

[69] Rex Min et al. "The hardware and the network: total-system strategies for power aware wireless microsensors". In: (2002).

[70] Y. Prakash and S. K. S. Gupta. "Energy efficient source coding and modulation for wireless applications". In: 1 (2003), 212–217 vol.1.

[71] Loren Schwiebert et al. "A biomedical smart sensor for the visually impaired". In: 1 (2002), pp. 693–698.

[72] H. Kim and J. Kim. "Energy-efficient resource management in Wireless Sensor Network". In: (2011), pp. 69–72.

[73] C. Enz, N. Scolari, and U. Yodprasit. "Ultra low-power radio design for wireless sensor networks". In: (2005), pp. 1–17.

[74] Kenneth C Barr and Krste Asanović. "Energy-aware lossless data compression". In: *ACM Transactions on Computer Systems (TOCS)* 24.3 (2006), pp. 250–291.

[75] C. Erin and H. H. Asada. "Energy optimal codes for wireless communications". In: 5 (1999), 4446–4453 vol.5.

[76] K. Chi et al. "Coding Schemes to Minimize Energy Consumption of Communication Links in Wireless Nanosensor Networks". In: *IEEE Internet of Things Journal* 3.4 (2016), pp. 480–493.

[77] M. Kocaoglu and O. B. Akan. "Minimum Energy Channel Codes for Nanoscale Wireless Communications". In: *IEEE Transactions on Wireless Communications* 12.4 (2013), pp. 1492–1500.

[78] K. Chi et al. "Energy-Efficient Prefix-Free Codes for Wireless Nano-Sensor Networks Using OOK Modulation". In: *IEEE Transactions on Wireless Communications* 13.5 (2014), pp. 2670–2682.

[79] Kaikai Chi et al. “Energy optimal coding for wireless nanosensor networks”. In: (2013), pp. 998–1002.

[80] X. Liang, W. Li, and T. A. Gulliver. “Energy Efficient Modulation Design for Wireless Sensor Networks”. In: (2007), pp. 98–101.

[81] SeongHwan Cho and A. P. Chadrakasan. “A 6.5-GHz energy-efficient BFSK modulator for wireless sensor applications”. In: *IEEE Journal of Solid-State Circuits* 39.5 (2004), pp. 731–739.

[82] A. Y. Wang et al. “Energy efficient modulation and MAC for asymmetric RF microsensor systems”. In: (2001), pp. 106–111.

[83] Irfan Ali et al. “Bit-error-rate (BER) simulation using MATLAB”. In: *International Journal of Engineering Research and Applications* 3.1 (2013), pp. 706–711.

[84] Vinay Panwar and Sanjeet Kumar. “Bit error rate (BER) analysis of Rayleigh Fading channels in mobile communication”. In: *International Journal of Modern Engineering Research (IJMER) Vol* 2 (2012).

[85] Oluseye A Adeleke and Robert O Abolade. “Modulation methods employed in digital communication: an analysis”. In: *Int J Electr Comput Sci IJECS-IJENS* 12.03 (2012), pp. 85–93.

[86] K. Choi and H. Liu. “Ber Simulation at the Waveform Level”. In: (2016), pp. 224–238.

[87] B. L. Ahlem, M. Béchir Dadi, and C. Belgacem Rhaimi. “Evaluation of BER of digital modulation schemes for AWGN and wireless fading channels”. In: (2015), pp. 1–5.

[88] B. Chirag, A. Lohith, and H. S. Prashantha. “Comparative performance analysis of various digital modulation schemes in AWGN channel”. In: (2017), pp. 1–5.

[89] John Anthes. “OOK, ASK and FSK Modulation in the Presence of an Interfering signal”. In: *RF Monolithics* (2004).

[90] Hamood Shehab Hamid, W Ismail, and M Singh. “Analyze BER performance of wireless FSK system”. In: *Microwaves and RF* 48 (2009).