Table of contents

List of tables

List of figures

List of acronyms

3G-LTE	3rd Generation-Long Term Evolution
4G	4th Generation
CP	Cyclic Prefix
CSI	Channel State Information
D-BLAST	Diagonal - Bell Laboratories Layered Space-Time Architecture
DSTBC	Differential Space Time Block Coding
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
ISI	Inter-Symbol Interference
LDPC	Low Density Parity Check
LOS	Line of Sight

MIMO	Multiple Input Multiple Output
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MRC	Maximal Ratio Combining
NLOS	Non Line of Sight
OFDM	Orthogonal Frequency Division Multiplexing
PAM	Pulse Amplitude Modulation
PAPR	Peak to Average Power Reduction
QAM	Quadrature Amplitude Modulation
QOSTBC	Quasi-Orthogonal Space Time Block Codes
QPSK	Quadrature Phase-Shift Keying
rms	root mean square
SFBC	Space Frequency Block Coding
SISO	Single Input Single Output
SNR	Signal to Noise Ratio
STBC	Space Time Block Coding

STFBC	Space-Time-Frequency Block Coding
SUI	Stanford University Interim
VBLAST	Vertical-Bell Laboratories Layered Space-Time
TAST	Threaded Algebraic Space-Time
WLAN	Wireless Local area Network

Chapitre 1 - Introduction

Over the last twenty years, wireless communication systems technology has expanded at a fast pace. Data rates and the quality of service (QoS) requirements are constantly reviewed and improved in order to ensure that users get the desired satisfaction of their wireless communication experience. Wireless communication systems and networks will have to become ever more efficient and flexible to be able to compensate for the limited availability of radio frequency (RF) spectrum due to various regulations. As such, wireless communication systems have to have the ability to generate a constantly expanding high spectral performance, more extensive data rates, and greater number of users employing the wireless system at the same time. The efficiency of the spectrum is increased by the utilization of multiple antennas at the transmitter as well as the receiver end. This creates what is known as a multiple-input/multiple-output radio channel (MIMO) [1][2], different from the customary single-input/single-output radio channel (SISO). MIMO in combination with orthogonal frequency-division multiplexing (OFDM) (MIMO–OFDM) have been identified as a promising approach for high spectral efficiency wideband systems. The demand for extensive data rates and high channel capacity requires improved receiver implementations and architectural design. In order to achieve this, a balance between hardware complexity and operational performance must be established.

The purpose of this thesis is threefold: to propose highly efficient algorithms of a reasonable complexity for MIMO-OFDM communication system ; to design a real-time Field Programmable Logic Array (FPGA) architectural model for the proposed transmitter and

receiver algorithms; and to put forward efficient design methodology to translate algorithms to FPGA architectural models. This first chapter will mainly be concerned with outlining the motivations behind the improved MIMO-OFDM transmitter-receiver model to support high data rate. In addition, the existing literature on MIMO-OFDM systems will be reviewed and the challenges of developing algorithms and FPGA architectural models will be identified, particular stress being put on the necessity of an efficient methodology with regard to the transformation of algorithms to architectural models. All these aspects will be drawn together to define the purpose of the study. Furthermore, the organization of the thesis will be presented with the summary of contributions in design methodology, algorithm optimization, FPGA architectures and joint optimization of algorithms and architectures.

1.1 Wireless system development

Wireless communication has developed over the past twenty years from being a limited technology used by a handful of specialists to becoming integrated into a wide variety of electronic devices available to the general public. With the advent of data-centric applications such as the Internet, mobile communication, or wireless local area networks (WLANs), in the early nineties, wireless communication started its way into everybody's daily life. New products (e.g., the iPhone or the iPad) and services (e.g., digital-TV or on-demand video streaming) call for higher throughput and better quality of service which require new concepts and standards for wireless communication. Third Generation (3G) wireless systems were first implemented in Japan in 2001 to meet the above requirements. The International Telecommunications Union endorsed the following third generation wireless networks: CDMA2000, wideband CDMA, and time division synchronous CDMA.

The introduction of 3G wireless networks led to the development of a wide range of multimedia applications, including online gaming, Internet browsing, video streaming. Nevertheless, despite its improved features, 3G has not yet manage to provide solutions for such issues as Multiple Access Interference (MAI) and Inter Symbole Interference (ISI). As a result, Fourth Generation (4G) wireless networks have begun to be developed in order to rectify the shortcomings of the previous generations of wireless systems. The main targets of 4G wireless systems are bandwidth expansion, data rate increase, extended coverage and reduced cost.

The International Mobile Telecommunications Advanced (IMT-Advanced) specifications requirements for 4G standards were published in March 2008 by the communications division of the International Telecommunications Union-Radio (ITU-R). The specifications included a maximum speed of 100 megabits per second in the case of high mobility communication and 1 gigabit per second for low mobility communication [3].

Mobile WiMAX and Long Term Evolution LTE systems are frequently classified as 4G by wireless service providers, despite the fact that they do not reach the established peak rate of 1 Gbit/s, and thus not abiding by the specifications of IMT-Advanced. ITU-R made a concession on 6th December, 2010, namely that the aforementioned systems, and other 3G systems, could be categorized as 4G, despite not meeting the specifications of IMT-Advanced, if they could be demonstrated to be precursors of the versions that abide by the specifications of IMT-Advanced and show "a substantial level of improvement in performance and capabilities with respect to the initial third generation systems currently deployed" [4].

Mobile WiMAX Release 2 (also known as Wireless MAN-Advanced or IEEE 802.16m') and LTE Advanced (LTE-A) are IMT-Advanced compliant backwards compatible versions of the

above two systems, standardized during the spring 2011, and promising peak bit rates in the order of 1 Gbit/s. Services are expected in 2013 [5].

The objectives of 4G wireless networks include increasing the data transmission rate, reduced latency, and high reliability (reduction of wireless disconnection), by adapting packet-optimized radio access systems that sustain bandwidth distribution. Furthermore, 4G systems seek to decrease the price of infrastructure equipment and user terminals, as well as to use a modulation structure with a higher performance than the CDMA scheme employed by 3G networks, to maximize the use of communication bandwidth. These objectives call for a complete reorganization of the physical layer and the system architectural model.

OFDM is the modulation scheme employed by 4G systems to improve spectral efficiency. It is a broadband multi-carrier modulation scheme where the available bandwidth is subdivided into orthogonal narrow-band sub-carriers. OFDM is known since the late 1960's [6], the developments in semiconductor and computer technology have made OFDM a useful and functional scheme, in contrast to earlier days when the hardware technology available in the 1960s was not suitable to the application of the OFDM scheme. Nowadays, however, OFDM has become the preferred option for wireless communication systems due to a number of reasons, including more straightforward channel equalization in contrast to single carrier schemes, stable against frequency selective fading, and high spectral efficiency. Systems that currently employ OFDM are digital audio broadcasting (DAB), digital video broadcasting terrestrial (DVB-T), digital video broadcasting-handheld (DVB-H), WLANs, WiMAX, the majority of the long-term evolution (LTE) 4G systems, as well as a number of short-range systems with ultra bandwidth (UWB). What is more, OFDM can also be applied to wired

systems, such as data modems for asymmetric digital subscriber line (ADSL) and very high-speed digital subscriber line (VDSL).

1.2 Background and Motivation

In order to increase the data rate and the communication link robustness, 4G systems employs MIMO schemes alongside OFDM. The advantage of MIMO schemes is that they can reach higher throughput than SISO systems at the same bandwidth and transmit power. Wireless MIMO systems send and receive information over two or more antennas often shared among many users in case of multi-user system. The signals reflect off of objects in the environment causing multiple paths. In conventional systems, these multi-paths cause interference and fading. However, MIMO systems combine the multiple fading paths and users' signals to overcome multi-user interference and fading, and thereby increase data throughput and reduce Bit Error Rate (BER) as compared to SISO systems. On the other hand, MIMO communication is targeted toward wideband systems which suffer from frequency-selective fading, and as a result the ISI will exist in the system. To mitigate this ISI effect and simplify the channel equalization, MIMO is combined with OFDM in order to convert the frequency-selective channel into a set of parallel frequency-flat fading channels. Transmission using MIMO-OFDM is used to either increase the robustness of the system or the data rate. In a richly scattered environment, transmit diversity play an important role to maintain the robustness of the wireless communication system. Transmission schemes that exploit diversity use spatial dimensions to add more redundancy, thus keep the data rate equivalent to SISO-OFDM system in order to increase the BER performance. Space-Time Coding is the principal of generating redundancy by coding across time and spatial dimensions [7]-[13], Space time Block Coding (STBC) [14] is the most widely used examples that employs STC scheme. On the other hand, Space Division

Multiplexing (SDM) is employed if the algorithm uses different antennas to transmit multiple data symbols over the channel. SDM schemes are used if high data rates are the main objective of the system [15]-[20].

Both STC and SDM coding schemes cannot achieve multipath diversity and were proposed for flat fading channel and not suitable for frequency selective fading channels. These two problems could be solved if more frequency diversity is introduced to the system. MIMO-OFDM provides the opportunity to code the transmitted symbols over different antennas (space) and sub-carriers (frequency), this coding scheme is known as Space-Frequency Block Coding (SFBC) and it can exploit the multipath diversity. Three dimensional coding over space, time and frequency is also known as Space-Time-Frequency Block Coding (STFBC). Both transmission schemes have recently been proposed in the literature [21]-[29]. However, the system complexity is a major obstacle and the decoding complexity problem has to be tackled. Additionally, most of the existing ST/SF codes are designed for single user systems only, for multiple access channels (MAC), the single-user ST/SF codes are always applied to each user independently, which results a reduced transmission rate. For example in conventional MIMO-OFDMA, users are separated in different frequency bands (sub-channels), and each user is coded separately using STBC or SFBC, leading to data rate reduction for each user when the number of users is increasing. The above reasons call for a new transmission scheme to enables multiple access by joint code design across multiple antennas, OFDM frames (time), subcarriers, and users.

The significant performance improvements of the MIMO-OFDM systems comes at the cost of increasing complexity of signal decoding at the receiver end. For example, in spatial multiplexing the linear increase in data rate with the minimum number of antennas at the transmitter and receiver end, is achieved with a more than linear increase in decoder complexity

irrespective of the nature of the used decoding algorithms. What is more, maximization of the potential benefits of multiple antennas technology necessitates even more complex algorithms, coming close to or surpassing the technological and economical limits of the integrated circuits technology.

According to Moore's Law, the chip's transistor density doubles every two years, which put a maximum limit on the system performance improvement rate. On the other hand, according to Shannon's Law, algorithms grow in complexity more rapidly than chips grow in density in order to reach the maximum channel capacity. This creates a gap between the algorithmic complexity and the hardware performance, the gap between the complexity of algorithms and battery capacity are even more pronounced, which calls for an efficient design of both more compact and more power efficient architectures.

The most complex component of a MIMO-OFDM receiver is the detector, whose role is to separate the spatially multiplexed data streams at the receiver end. Initially, only the complexity order of MIMO-OFDM receiver algorithms has been examined; however, this is appropriate only in the case of qualitative comparisons between different decoding algorithms, results of such an analysis are not particularly relevant to system implementation. On the other hand, a more thorough analysis of the level of algorithm complexity was developed for digital signal processor (DSP) implementation [30]. However, DSP implementations cannot meet the requirements (in terms of throughput) of currently emerging and future wideband MIMO-OFDM systems. As a result, FPGA architectural models are required for the implementation of highly complex decoding algorithms. However, to make sure that the only factor that influences the performance of the system is the wireless channel capacity, and not the receiver technology, additional developments of high-throughput wideband MIMO-OFDM systems are required.

Conventionally, the algorithm researchers and the hardware design teams work separately [31][32], this leads to the fact that many algorithms proposed are not realistic for real-time implementation due to high complexity and numerical stability problems. This thesis proposes a development environment that lets designers model an entire system accurately, including the behavior and interaction of hardware and software subsystems that represent the system platform parameters such as input data and wireless channel.

1.3 Thesis Objectives and Scope

The objective of this thesis is to propose high performance algorithms with realistic complexity and real-time optimized FPGA architectures for the MIMO-OFDM Transceiver. First, in order to reduce the detection algorithm complexity at the receiver side, and at the same time improve the MIMO-OFDM performance, a novel transmission scheme for MIMO-OFDM based on the parity bit selected and permutation block spreading methods is proposed. In this scheme, the transmitted data is coded across space, time and frequency domains. The coding is done using a spreading code where the choice of this code is determined by the parity bits of the transmitted message vector across the multiple antennas. The proposed scheme enables multiple access by joint code design across multiple antennas, OFDM frames, subcarriers, and users. It will benefit from the combined space, time and frequency diversity and allow users to share subcarriers with a manageable level of multi-user interference. Hence, better spectrum efficiency is achieved while improving bit error rate performance with respect to signal-to-interference rate.

The second objective is to develop platform architecture for real-time prototyping environment. In the proposed platform, the communication between Matlab and FPGA board is managed directly through the Universal Asynchronous Receive and Transmit (UART). In this

thesis, UART core functions are implemented using VHDL and integrated into the MIMO-OFDM FPGA chip to achieve compact, stable and reliable data transmission, which effectively represent a complete hardware design platform for MIMO-OFDM system.

The third objective is to develop an end to end Floating-Point FPGA architecture for the proposed MIMO-OFDM Transceiver scheme. The proposed architecture is divided into sub-modules where suitable optimization techniques are proposed for each sub-module in order to reach the overall optimized architecture.

1.4   Publications

1.4.1   Published

1. S. Moussa, Ahmed M.Abdel Razik' A.O. Dahmane, and H. Hamam "FPGA Implementation of Floating-Point Complex Matrix Inversion Based On Gauss-Jordan Elimination," CCECE 2013, May 5th to 8th, 2013, Regina, Saskatchewan, Canada.

2. S. Moussa, A.O. Dahmane, C. D'Amours and H. Hamam "MIMO-OFDM with Parity Bit Selected Block Spreading," 2nd international conference on consumer electronics, communications and networks (CECNET 2012), Three Gorges, Hubei, China, April 21st~23rd, 2012.

3. S. Moussa, A.O. Dahmane, C. D'Amours and H. Hamam "MIMO-OFDM with Permutation Block Spreading," 14th International Conference on Advanced Communication Technology, Pyeongchang, Republic of Korea., February 19~22, 2012.

4. S. Moussa, A.O. Dahmane, C. D'Amours and H. Hamam "OFDM with Permutation Block Spreading," GCC Conference and Exhibition , 19-22 Feb. 2011, Dubai, UAE.

5. S. Moussa, A.O. Dahmane, C. D'Amours and H. Hamam "OFDM with Parity Bit Selected Block Spreading," 10th IEEE ICT Conference, 14-18 April 2010, Doha, Qatar.

1.4.2 Submitted

1. S. Moussa, Ahmed M.Abdel Razik, A.O. Dahmane, C. D'amours and H. Hamam "FPGA Implementation of MIMO-OFDMA based on parity bit selected and permutation spreading", submitted to: *IEEE Transactions on Circuits and Systems I.*

1.5 Thesis Organization

Chapter 2 provides an overview of OFDM transmission systems including its mathematical model, and then its advantages and disadvantages are highlighted. Next, the combination of MIMO systems with OFDM is then described and MIMO-OFDM model is introduced, followed by a comprehensive review of the existing MIMO detection techniques and their associated BER performance and complexity analysis. Finally, MIMO-OFDM transmission schemes are categorized into three main categories Spatial Division Multiplexing (SDM), Space Time Coding (STC), and Space Frequency Coding (SFC), where the performance of these schemes are analyzed and compared.

Chapter 3 presents the new MIMO-OFDM scheme based on the parity bit selected and permutation block spreading. Mathematical model of the proposed technique is given and simulations are presented for different number of transmit/receive antennas, different modulation, different spreading code length, and different equalization techniques.

Chapter 4 presents FPGA design methodology for MIMO-OFDM systems, which allows converting the proposed algorithms onto the target prototyping platform in a systematic way. Additionally, detailed implementations for real-time prototyping environment based on UART

are also presented. Then, the RTL model for the individual blocks in the proposed MIMO-OFDM system is introduced. Detailed implementations and the potential drawbacks in each module are also provided. The synthesis results, which include the hardware resources usage, latency, and power consumption are presented and analyzed. Finally, functional verification results are introduced for major modules in the system.

Chapter 5 provides the optimization process for the proposed MIMO-OFDM FPGA architecture. Optimized and efficient architectures are proposed and designed for the key functional module of the systems. These efficient designs include pipelined architecture for IFFT/FFT modules, low complexity architecture for the despreading module, low complexity architecture for matrix inversion using GJ-elimination, and finally the complete design is converted to Fixed-Point representation, then the tradeoffs between BER performance and area reduction are investigated and the final results are introduced and analyzed.

Chapter 6 gives the thesis conclusion. The main results and conclusions are summarized. Moreover, some remaining open questions and directions for future research are discussed.

Chapitre 2 - MIMO-OFDM

2.1 Introduction

The main target of next generation wireless technologies such as 4G is to provide high speed data transmission rate to satisfy the needs of the emerging new applications. The requirements of 4G wireless communication could be summarized as, 100 Mb/sec data rate in outdoor environment and 1 Gb/sec in indoor channels. Hence a significant bandwidth efficiency improvement is required in order to maintain the frequency spectrum in the order of 100 MHz. The main challenge of high speed single carriers wireless transmission lies in the frequency selectivity of the channel, which means that the multipath delay spread of the channel is quite large due to the large bandwidth and leads to severe intersymbol interference (ISI). In order to overcome the ISI problem, the duration of the transmitted symbol must be much larger than the delay spread of wireless channels. In multi-carrier's transmission system such as OFDM, the entire channel is divided into many narrow-band subchannels, which are transmitted in parallel to maintain high-data rate transmission and, at the same time, to increase the symbol duration in order to mitigate the ISI effect. In addition to that, many advanced techniques, such as adaptive loading, transmit diversity, and receiver diversity, could be used with OFDM to improve transmission efficiency.

Recently MIMO communication which consists of multiple transmit and receive antennas are used extensively to increase the transmission rate, it is considered as the key solution for fading channels in rich scattering environment. Compared with SISO, a MIMO system can improve the capacity by a factor of the minimum number of transmit and receive antennas for

flat fading or narrow-band channels. For wideband transmission, it is natural to combine OFDM with MIMO to deal with frequency selectivity of wireless channels and to obtain diversity and/or capacity gains. Therefore, MIMO-OFDM has widely been used in various wireless systems and standards.

The coding structure of the transmitted signal plays a major role on the performance and capacity of MIMO-OFDM system. Several transmission schemes have been proposed for MIMO system to improve transmission performance and/or increase the throughput. These schemes of coding can be divided into two broad categories: Space Time Coding (STC) and Space Division Multiplexing (SDM). The former scheme is mainly used to increase the robustness of the system, while the later one is used to increases the maximum data rate attainable by the system.

This chapter begins with an overview of OFDM transmission systems including its mathematical model, then its advantages and disadvantages are highlighted. Next, MIMO-OFDM model is introduced, followed by a comprehensive review of the existing MIMO detection techniques and their associated BER performance and complexity analysis. A few examples of such algorithms include Maximum-Likelihood (ML) algorithm, the Zero-forcing (ZF) algorithm, the Minimum Mean Square Error (MMSE) algorithm, and the V-BLAST algorithm. Finally, MIMO-OFDM transmission schemes are categorized and the pros and cons of each scheme are described in details, in order to serve as a base for introducing the novel MIMO-OFDM transmission scheme based on parity bit selected and permutation, which will be introduced in the next chapter.

2.2   Conventional MIMO-OFDM system

2.2.1   OFDM system model

OFDM stands for orthogonal frequency division multiplexing. It is a subdivision of frequency division multiplexing in which multiple sub-carriers on adjacent frequencies are utilized in a single channel. In OFDM system, spectral efficiency is maximized by overlapping sub-carriers. Generally these sub-carriers can interfere with one another. But in OFDM, they do not interfere with each other because sub-carriers are orthogonal to each other. Due to this fact OFDM can maximize spectral efficiency without channel interference. The spectrum of OFDM system in frequency domain is represented in Figure 2-1.
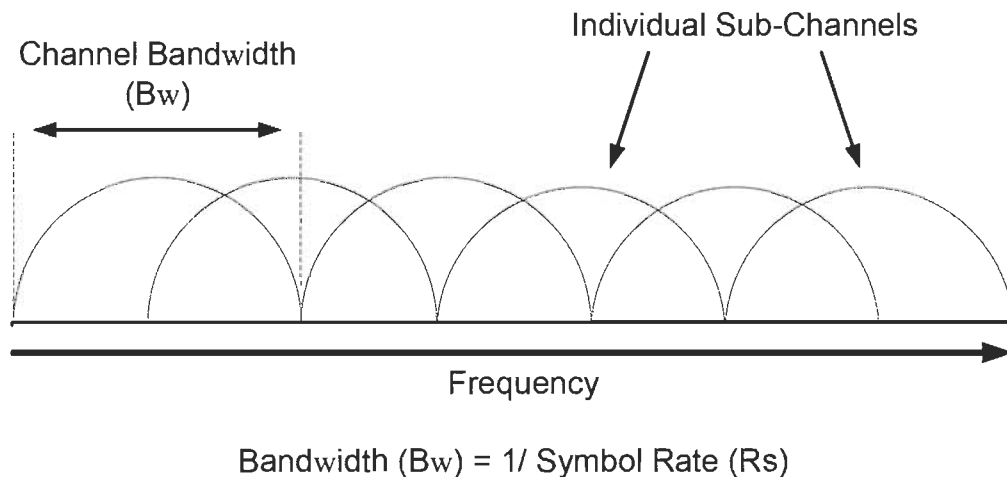
Figure 2-1 OFDM Frequency spectrum [33]

Orthogonality of Sub-Channel Carriers

As described above, the sub carriers are orthogonal in OFDM systems, which mean that each carrier spectrum in frequency domain has a null value at the center frequency of each of

the other carriers in the system. This allows these carriers to be as close as possible to each other, hence better spectral efficiency. In another words, orthogonality enables concurrent transmission on almost every sub-carrier in frequency space without interference as shown in Figure 2-2. So at the receiver side distinct sub-carriers can be easily extracted. In conventional FDM system on the other hand, this overlapping of sub-carriers is not possible, as a result a guard band between each carrier is used to avoid inter-carrier interference.
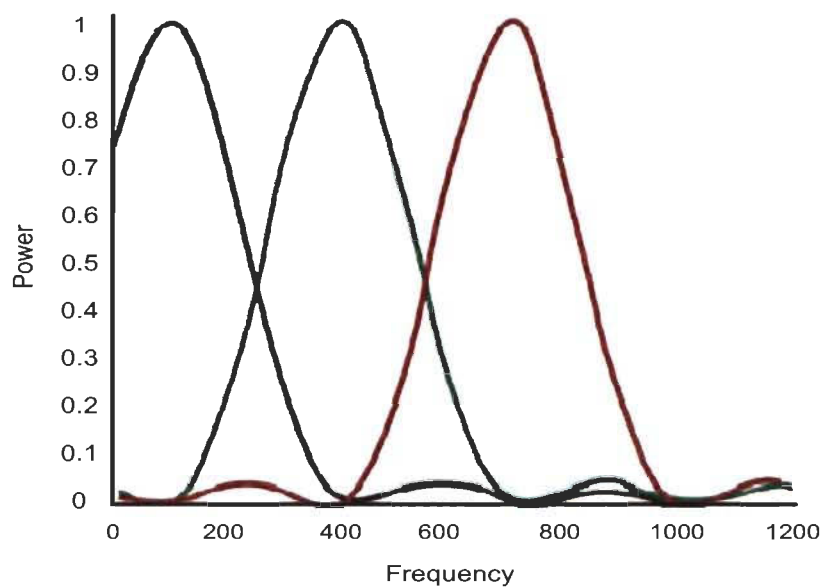
Figure 2-2 Orthogonal overlapping spectral shape for OFDM [34]

OFDM Transmitter/Receiver architecture:

Classic model of an OFDM transmitter and receiver is shown in Figure 2-3. Figure 2-3(a) shows OFDM transmitter and Figure 2-3(b) shows OFDM receiver. The transmitter converts digital data which is to be transmitted, into a subcarrier mapping of amplitude and phase and then using the Inverse Fast Fourier Transform (IFFT), digital data is converted into time domain signal representation from spectral representation and a cyclic prefix (CP) is added. Then to transmit the OFDM signal, the time domain signal is mixed with required frequency through

frequency multiplexing. The reverse operation is performed at the receiver end as shown in Figure 2-3(b). When the modulated OFDM signal arrives at the receiver, the RF signal is mixed with base band for processing and the CP is removed. Then, the signal spectrum is converted to frequency domain using Fast Fourier Transform (FFT). Then phase and amplitude of the subcarrier is extracted out and demodulated back to digital data. The IFFT and FFT are both the opposite functions of each other. The suitable term to describe them rests on whether the signal is being generated or received.

(a) OFDM Transmitter

(b) OFDM Reciever
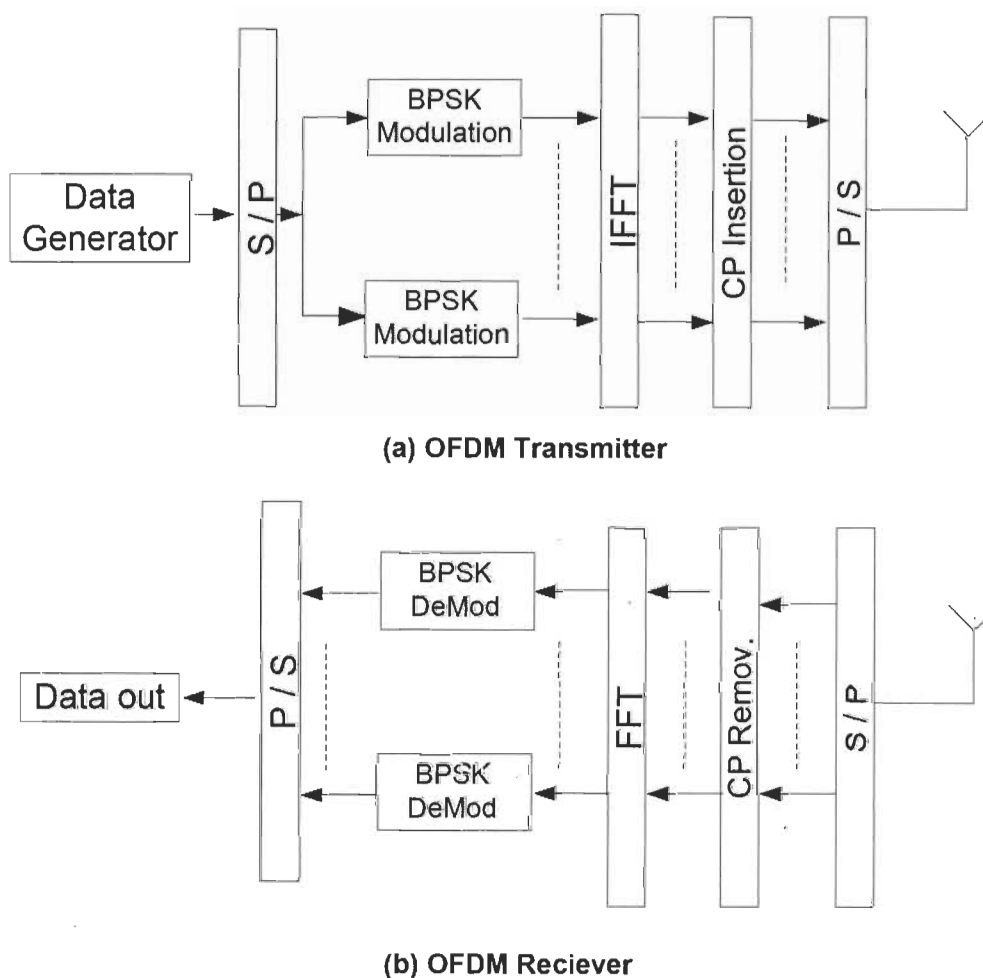
Figure 2-3 Conventional OFDM System

17

Cyclic Prefix Insertion:

In wireless transmission system, the radio signal gets reflected back from walls, buildings, mountains and all other objects in the transmission environment causing multiple signals to arrive at the receiver at different time. This phenomenon is known as multipath transmission. At the OFDM receiver side, multipath channel presents time distortion where the duration of each OFDM symbol is increased. As a result the received symbols interfere with each other and produce the intersymbol interference (ISI) which is very popular in OFDM systems. The symbol rate of OFDM signal is much less than a single carrier transmission technique. For example, in a single carrier system with BPSK modulation the bit rate directly determines the symbol transmission rate. But in OFDM, the whole bandwidth is subdivided into N_f subcarriers which results in N_f-times lower symbol rate than that of single carrier transmission. Thus the effect of intersymbol interference is reduced in multipath transmission through OFDM, making OFDM a natural resistant to ISI. The system can be further improved if we add guard periods, which is a cyclic copy, this is done by replicating the OFDM symbol end and appending it to the start of the OFDM frame as shown in Figure 2-4. The addition of this guard period results in the extension of symbol waveform but it greatly reduce ISI caused by multipath transmission.
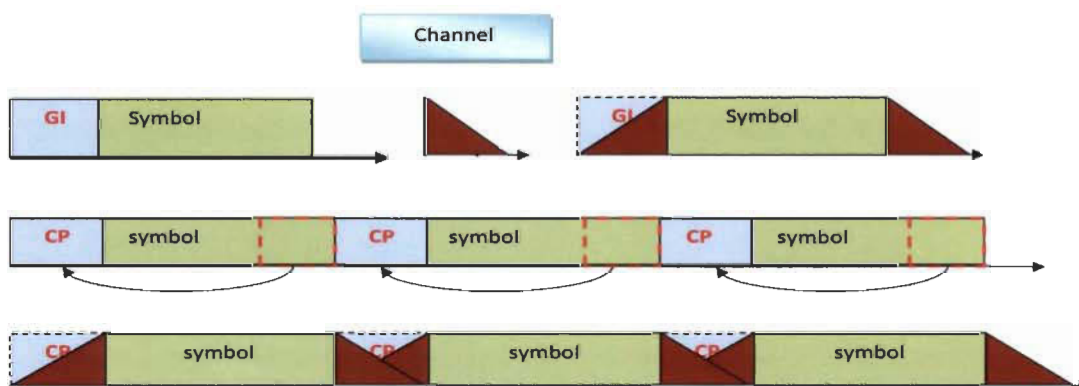
Figure 2-4 OFDM CP insertion

2.2.2 OFDM Mathematical model

In OFDM the information stream is converted into N_f parallel streams via serial to parallel (S/P) converter as shown in Figure 2-3, where N_f is the number of subcarriers before CP insertion. These parallel streams are modulated using BPSK. The kth OFDM symbol is given by [35]:

$$X_k(t) = \sum_{n=0}^{N_f-1} s_{k,n} p(t-kT) e^{j2\pi\frac{n}{T}t} \tag{2.1}$$

Where T is the OFDM symbol duration, and

$$s_k = \left[s_{k,0}, s_{k,1}, \dots, s_{k,N_f-1} \right]^T \tag{2.2}$$

is the N_f parallel data streams for one OFDM frame before CP insertion. The subscript T represents the transpose operator and $p(t)$ is the pulse shaping for each symbol. If we consider rectangular pulse shaping waveform:

$$p(t) = \begin{cases} 1, & 0 \leq t \leq T \\ 0, & \text{otherwise} \end{cases} \tag{2.3}$$

and that each subcarrier and the OFDM symbol are sampled N_f times per frame interval, the modulated waveform of (2.1) becomes:

$$X_k\left(\frac{mT}{N_f}\right) = \sum_{n=0}^{N_f-1} s_{k,n} e^{j2\pi nm/N_f} \quad , m = 0,1, \dots, N_f - 1 \tag{2.4}$$

Hence, the inverse fast Fourier transform (IFFT) is used to modulate these substreams to their respective subcarrier frequencies:

$$X_k = N_f IFFT(s_k) \tag{2.5}$$

The resulted parallel symbols after OFDM modulation is converted into a serial stream and a CP is inserted. CP duration is selected to be larger than the channel delay to avoid ISI. At the receiver, the signal is converted from serial to parallel and the CP is removed, then the FFT is used to demodulate and take decision according to the employed modulation. In case of BPSK, the decision function is simply the sign function.

2.2.3 OFDMA

OFDMA stands for Orthogonal Frequency Division Multiple Access and it is the multiple user version of OFDM. As shown in Figure 2-5, in this system a subset members of sub-carriers are assigned to individual user dynamically, through the use of TDMA (based on timeslots) or FDMA (separate channels), hence the system support simultaneous users transmission.

Although there are many advantages of OFDMA, its flexibility to resource allocation and robustness to frequency selective fading are considered the main benefit. By using OFDMA, every user has its own unique set of sub-channels as shown in Figure 2-5 and the base station can allocate the subcarriers to users dynamically [36]. If high quality of service (QoS) has been requested by specific user, more resources (more power, large number of sub-channels and higher level of modulation) can be applied to this user.

The major advantages of OFDMA could be summarized as follow:

- Deployment flexibility at different frequency bands with little modification needed with the air interface.
- Allows power control at either channel or sub-channel level.
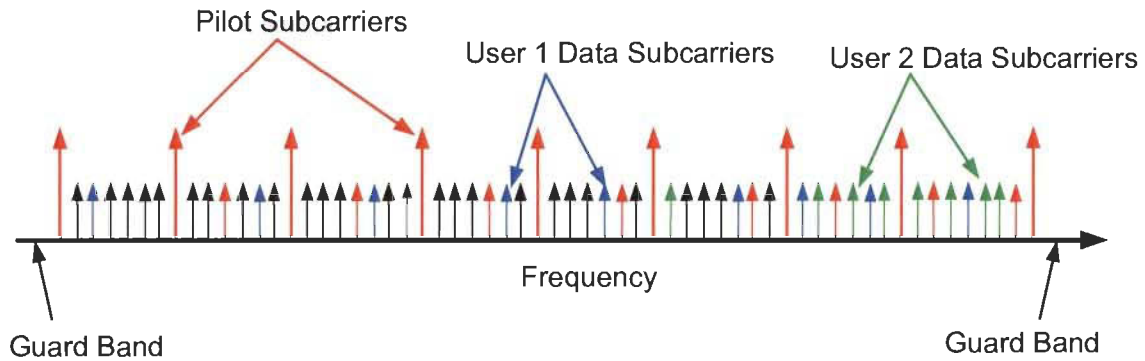
Figure 2-5 Multi-user OFDM system [37]

- By spreading the carriers all over the used spectrum, frequency diversity could be introduced to the system.

- Gives excellent coverage by enabling single frequency network coverage.

- By applying different carrier permutation between different users in different cells, interferences in neighboring cells is averaged.

- Cyclic permutation is used to overcome interference within the cell.

2.2.4 MIMO-OFDM Mathematical model

In MIMO-OFDM systems, the wireless link has its transmitter and receiver systems equipped with an arbitrary number of antennas as shown in

Figure 2-6. The basic idea of using MIMO-OFDM signals is to ensure that the quality of signal (measured by BER) and data rate (bits/sec) are improved. This is done by combining the signals on the transmit (TX) antennas at one end and the receive (RX) antennas at the other end. The result improvements in terms of link quality and performance can be used to significantly increase the operator's revenue as well as the wireless network quality of service.
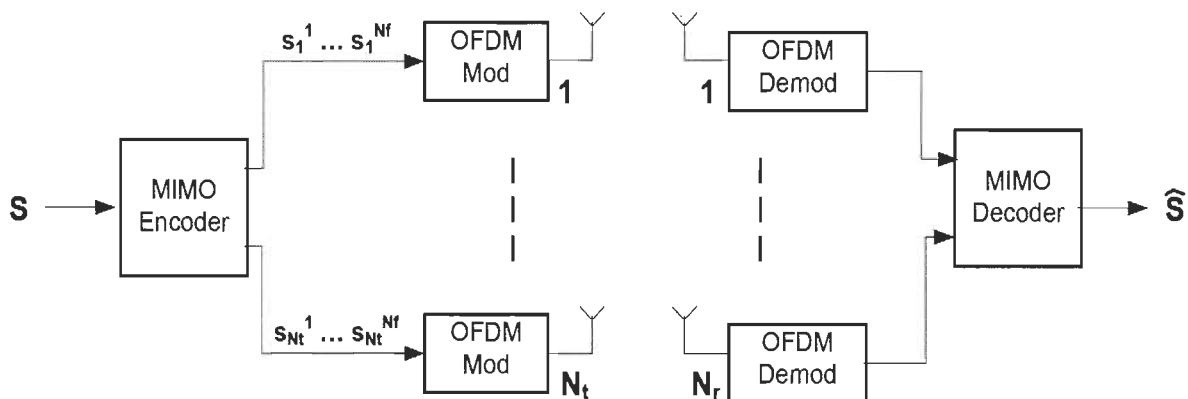
Figure 2-6 Simplified block diagram of MIMO-OFDM system

A general MIMO-OFDM system is shown in

Figure 2-6, where N_t transmit antennas, N_r receive antennas, and N_f-tone OFDM are used. First, the incoming bit stream is mapped into a number of data symbols via some modulation type such as BPSK. Then a block of N_s data symbols $[s_1, s_2, ., ., ., s_{N_s}]$ are encoded into a codeword matrix S of size TXN_t which will then be sent through N_t antennas in T OFDM frames, each frame consisting of N_f sub-channels. Specifically, $S_j^1, S_j^2, ., ., ., S_j^T$ will be transmitted from the jth transmit antenna in OFDM frames $1, 2, ., ., ., T$ respectively, where S_j^n denotes a vector of length N_f, for all $j = 1, 2, ., ., ., N_t$ and $n = 1, 2, ., ., ., T$ The codeword matrix S can be expressed as.

$$S = \begin{pmatrix} S_1^1 & \cdots & S_1^T \\ \vdots & \ddots & \vdots \\ S_{N_t}^1 & \cdots & S_{N_t}^T \end{pmatrix} \tag{2.6}$$

After appending the cyclic prefix on each OFDM frame, S_j^n will be transmitted from the jth transmit antenna in the nth OFDM frame.

We can identify X of size N_f x N_t as a subset of S that represents the transmitted symbols from all transmitting antennas for the duration of one OFDM frame, hence

$$X = \begin{pmatrix} s_1^1 & \cdots & s_1^{N_f} \\ \vdots & \ddots & \vdots \\ s_{N_t}^1 & \cdots & s_{N_t}^{N_f} \end{pmatrix} \tag{2.7}$$

Or it could be written as

$$X^{(k=1:N_f)} = \begin{bmatrix} X_1^{(k=1:N_f)} \\ X_2^{(k=1:N_f)} \\ \cdot \\ \cdot \\ \cdot \\ X_{N_t}^{(k=1:N_f)} \end{bmatrix} \tag{2.8}$$

Here $X_j^{(k=1:N_f)}$ is a vector represents all symbols transmitted from antenna j through all N_f subcarriers.

After passing through the MIMO channels, the received signals will be first sent to the reverse OFDM frame (cyclic prefix removal and FFT) and then sent to the decoder.

The received vector could be represented as

$$Y^{(k=1:N_f)} = \begin{bmatrix} Y_1^{(k=1:N_f)} \\ Y_2^{(k=1:N_f)} \\ \cdot \\ \cdot \\ \cdot \\ Y_{N_r}^{(k=1:N_f)} \end{bmatrix} \tag{2.9}$$

Without loss of generality, we can consider only one subcarrier (k=1), then equation 2.9 can be represented in a matrix form as

$$Y = HX + v \tag{2.10}$$

Where Y is the received vector with N_r dimension, H is an N_rxN_t complex propagation matrix that is assumed constant for the length of a frame transmission (i.e., a quasi-static channel) and assumed known at the receiver (e.g., via transmission of training sequences). It is assumed that the statistics of the channel transfer matrix H can be described by the fading statistics namely, Rayleigh fading, Ricean fading or AWGN. Furthermore, it is assumed that the elements of H have a variance of one, or, in other words, the average channel gain P_c is normalised to one.

2.2.5   MIMO Detection techniques

This section summarizes and compares the various algorithms designed to detect the MIMO signals, Channel State Information (CSI) is assumed to be perfectly known at the receiver side. A few examples of such algorithms include Maximum-Likelihood (ML) algorithm, the Zero-forcing (ZF) algorithm, the Minimum Mean Square Error (MMSE) algorithm, and the V-BLAST algorithm.

2.2.5.1   Maximum-Likelihood (ML)

The ML algorithm uses a received vector Y to detect the vector symbol most likely to be transmitted by the transmitter. Minimizing the difference between the transmitted signal affected by the channel coefficients and the received signal is done by using the following equation:

$$e = arg\ min_{all\ X} \|Y - HX\|^2 \tag{2.11}$$

The number of transmitting and receiving antennas plays an important role in defining the size of possible number of input vector $2^{N_t N_r}$ that need to be checked using the ML algorithm. If we consider the number of transmitting and receiving antennas as 2 with BPSK modulation, the total likely candidates turn up to be $2^{2*2}=16$ different signals. When the number of transmitting and receiving antennas becomes 4, the number of candidates becomes $2^{4*4}=65536$ different signals.

2.2.5.2 Zero Forcing (ZF)

The ZF algorithm gives its best results when Inter Signal Interference (ISI) becomes prominent in the performance of the system. The interference is cancelled out by the introduction of weight matrix into the system. By multiplying the N_txN_r weight matrix with the incoming signal at the receiver, we can get an estimated transmitted signal as shown in the equation.

$$\widehat{X} = W_{zf}Y \tag{2.12}$$

This detection technique is also known as spatial filtering. The condition $N_t \geq N_r$ should be maintained true in order to perform this linear demultiplexing. The simple process of ZF filtering is described in the following steps.

A weight vector of N_r dimensions $w_{zf,k}^T$, for the kth transmitted substream is given by

$$w_{zf,k}^T = g_k^T H^+ \tag{2.13}$$

In which, the Moore-Penrose (MP) inverse is denoted by the '+' symbol and an N_tx1-dimensional vector with all zeros except for the kth element (it's value is 1) is denoted by g_k.

The product of H^+ with g_k^T gives the kth row vector in the H^+. Assuming that $N_t \geq N_r$ as mentioned above, the pseudo inverse code of H is computed using

$$H^+ = (H^H H)^{-1} H^H \tag{2.14}$$

The weight matrix needed to calculate all transmitted substreams could be obtained by applying the MP pseudo inverse matrix H^+ as

$$W_{zf} = \begin{bmatrix} w_{zf,1}^T \\ \vdots \\ w_{zf,N_t}^T \end{bmatrix} = H^+ \tag{2.15}$$

we could choose inverse schemes other than MP pseudo inverse. The reason for choosing MP pseudo inverse in ZF is to obtain a norm-minimized and least-squares-type weight vectors. When the transmitted signal $Y(t)$ is multiplied with W_{zf}, the output vector signal(N_t-dimensional) is denoted by

$$\widehat{S} = W_{zf} Y \tag{2.16}$$

$$= W_{zf}\{HX + v\} \tag{2.17}$$

$$= X + W_{zf} v \tag{2.18}$$

$$= X + \begin{bmatrix} w_{zf,1}^T\, v \\ \vdots \\ w_{zf,N_t}^T\, v \end{bmatrix} \tag{2.19}$$

The ZF algorithm reduces the power of the interfering substreams to zero. This results in maximizing the SIR (Signal to Interference power Ratio) in the filtered output signal. As the power of received signal is sacrificed, ZF algorithm provides lower SNR compared with other schemes.

2.2.5.3 Minimum Mean Squared Error (MMSE)

The ZF algorithm doesn't consider the effect of noise in the channel while reducing the interference. This results in increasing the noise power in the desired signal, which causes degradation in the output signal quality. This deficiency is addressed in the MMSE (Minimum Mean Squared Error) algorithm. As the name suggests, this algorithm minimizes the error between the output signal $\hat{s}(t)$ and a reference signal (t). The Mean Square Error $J(w)$ is shown as

$$J(w) = E[|d - \hat{s}|^2] \tag{2.20}$$

$$= E[|d - w^T x|^2] \tag{2.21}$$

$$= E[|d|^2] - w^H \gamma_{rd} - w^T \gamma_{rd}^* + R_{rr} w \tag{2.22}$$

In this equation, $E[*]$ denotes the ensemble average (expectation) of the signal. γ_{rd} & R_{rr} respectively represent the correlation vector and the correlation matrix. They are defined using the equations

$$\gamma_{rd} = E[y^* d] \tag{2.23}$$

$$R_{rr} = E[y^* y^T] \tag{2.24}$$

At the minima of $J(w)$, the following condition holds true.

$$\frac{\partial J(w)}{\partial w} = 0 \tag{2.25}$$

Using differentiation for (2.22) with respect to w, we can arrive at the following conditions shown below.

$$\frac{\partial}{\partial w}(w^H \gamma_{rd}) = 2\gamma_{rd} \tag{2.26}$$

$$\frac{\partial}{\partial w}(w^T \gamma_{rd}^*) = 0 \tag{2.27}$$

$$\frac{\partial}{\partial w}(w^H R_{rr} w) = 2R_{rr} w \tag{2.28}$$

By substituting (2.22) and (2.26)–(2.28) into (2.25), we can obtain

$$\frac{\partial J(t)}{\partial w} = -2\gamma_{rd} + 2R_{rr}\mathbf{w} = 0 \tag{2.29}$$

this brings us to the resulting equation

$$w_{mmse} = R_{rr}^{-1} \gamma_{rd} \tag{2.30}$$

The vector derived from the above equation is the weighted MMSE vector for a single substream. When it is generalized to a system of $N_t \times N_r$, we get the equation

$$W_{mmse}^T = [W_1 \quad \cdots \quad W_{N_t}] = R_{rr}^{-1} \gamma_{rd} \tag{2.31}$$

W_{mmse}^T is the matrix of size $N_t \times N_r$, and γ_{rd} represents the correlation matrix in the received signal calculated by the received signal vector Y and an $N_t \times 1$ dimensional reference signal vector D using the equation

$$\gamma_{rd} = E[Y^* X^T] \tag{2.32}$$

By defining the P_t as the average transmission power per antenna, we can represent the correlation vector as well as correlation matrix using the equations (2.23) & (2.24), as follows.

$$\gamma_{rd} = E[(HX + v)^* X^T] = H^* E[x^* d_k] = P_t H^* \tag{2.33}$$

$$R_{rr} = E[(HX + v)^* (HX + v)^T] = P_t \left(H^* H^T + \frac{\sigma^2}{P_t} I_{N_r} \right) \tag{2.34}$$

In the above equation σ^2 represents the power of thermal noise and I_{N_r} denotes a unit matrix of size N_r.

The optimal matrix W_{mmse} can be derived as follows.

$$W_{mmse}^T = R_{rr}^{-1}\gamma_{rd} \tag{2.35}$$

$$= \frac{1}{P_t}\left(H^*H^T + \frac{\sigma^2}{P_t}I_{N_r}\right)^{-1} P_t H^* \tag{2.36}$$

$$= \left(H^*H^T + \frac{\sigma^2}{P_t}I_{N_r}\right)^{-1} H^* \tag{2.37}$$

Therefore, the optimum MMSE weight matrix is obtained by

$$w_{mmse} = (R_{rr}^{-1}\gamma_{rd})^T \tag{2.38}$$

$$= H^H\left(H^*H^T + \frac{\sigma^2}{P_t}I_{N_r}\right)^{-1} \tag{2.39}$$

As the MMSE algorithm takes the thermal noise into consideration, it improves the signal power of the desired component in order to improve the SNR performance of the system.
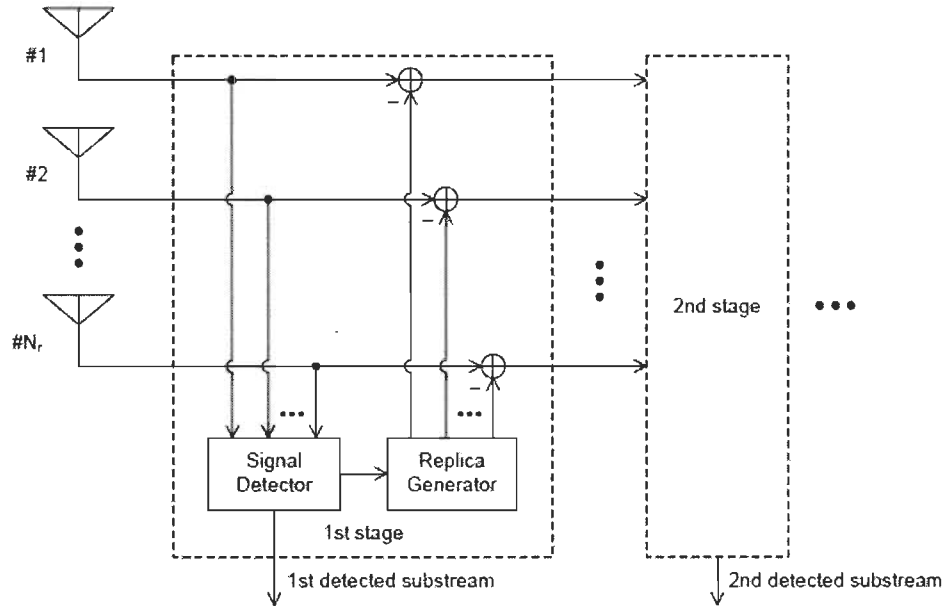
Figure 2-7 SIC architecture model

2.2.5.4 VBLAST

This nonlinear algorithm is very popular due its usage of SIC (Successive Interference Cancellation) depicted in the Figure 2-7. One of the above described algorithms (ZF or MMSE) is useful in detecting the strongest component in the received signal that will be used for interference cancellation. VBLAST uses the strongest signal detected at the receiving end and subtract it from the received signal. It then proceeds to the detection of the second most powerful signal, since it has already cleared the first and so forth. The final vector is provided after all interferences added in the channel are cancelled.

SIC-ZF

This section gives an initial description of the SIC algorithm when it is combined with the ZF method. This combination is known as VBLAST architecture. Using 'i' as an indicator for the ith iteration, we can work on the following equations.

$$w^{(i)}_{zf,k} = w_{zf,k} \tag{2.40}$$

$$Y^{(i)} = Y \tag{2.41}$$

$$H^{(i)} = H \tag{2.42}$$

Symbols in the first stage, or those with (i=1), are given as follows:

$$w^{(1)}_{zf,k} = w_{zf,k} \tag{2.43}$$

$$Y^{(1)} = Y \tag{2.44}$$

$$H^{(1)} = H \tag{2.45}$$

By assuming that $w_{zf,k}$ has the smallest normalized vector, we estimate the vector $\widetilde{X}^{(1)}_k$ with the help of ZF spatial filter $\widehat{X}^{(1)}_k$ for the k$^{\text{th}}$ substream of the output. This estimated vector is cancelled from the signal vector $Y^{(1)}$ by using its corresponding channel vector h_k.

$$Y^{(2)} = Y^{(1)} - \widetilde{X}^{(1)}_k h_k \tag{2.46}$$

Using this method for all the corresponding vectors in the channel matrix, we arrive at the following conclusions.

$$H^{(1)} = \underbrace{\left[h_1 \; ... \; h_{k-1} h_k h_{k+1} \; ... \; h_{N_t}\right]}_{N_t \; colum \; vectors} \tag{2.47}$$

$$H^{(2)} = \underbrace{\left[h_1 \; ... \; h_{k-1} h_{k+1} \; ... \; h_{N_t}\right]}_{N_t-1 \; colum \; vectors} \tag{2.48}$$

The above process converts the size of the channel matrix to $N_r \times N_t - 1$. The MP pseudo inverse matrix of $H^{(2)}$ is calculated to findout the normalized minimum weight vector $w^{(2)}_{zf,l}$

which is used to estimate the lth substream of the signal. By using the minimum vector, we can cancel h_l and $\widetilde{X}_l^{(2)}$ from $H^{(2)}$ and $Y^{(2)}$ respectively, which gives us the following equations.

$$Y^{(3)} = Y^{(2)} - \widetilde{X}_l^{(2)} h_l \qquad (2.49)$$

$$H^{(2)} = \underbrace{\left[h_1 \ldots h_{l-1} h_l h_{l+1} \ldots h_{N_{t-1}}\right]}_{N_t - 1 \; colum \; vectors} \qquad (2.50)$$

$$H^{(3)} = \underbrace{\left[h_1 \ldots h_{l-1} h_{l+1} \ldots h_{N_{t-1}}\right]}_{N_t - 2 \; colum \; vectors} \qquad (2.51)$$

This procedure is iterated N_t times in order to find the complete estimate of the received signal. The spatial diversity is increased as we progress through the detection stages; hereby SIC-ZF gives better performance than the traditional ZF algorithm.

2.2.5.5 SIC-MMSE

The performance of SIC could be further improved when MMSE is used instead of ZF because the former algorithm maximizes the output SNR (Signal to Noise Ratio). When SIC uses MMSE for substream separation, it is called SIC-MMSE. Although the process is similar to that of SIC-ZF, it differs when the substream quality is evaluated. the SIC-MMSE gives better spatial diversity than the SIC-ZF for an equal number of antennas.

In Figure 2-8 the performance of each MIMO detection algorithm is simulated for 2-transmit and 2- receive antennas using BPSK modulation. Although the ML is the most complex than any other algorithms, it has the best performance; the results also show that nonlinear SIC algorithms with either ZF or MMSE gives better performance than linear algorithms such as ZF and MMSE.
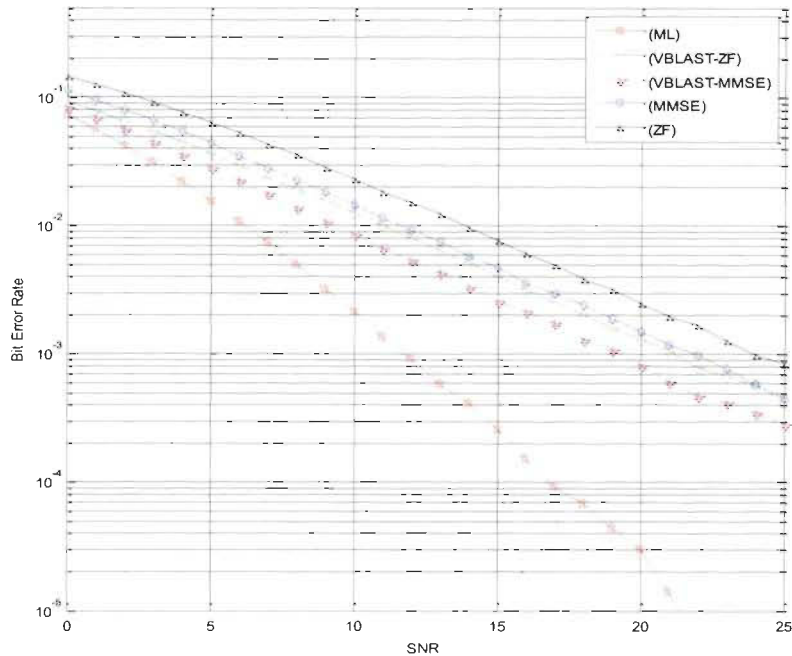
Figure 2-8 Performance comparison for MIMO detection algorithms (2 Tx, 2Rx and BPSK modulation)

2.3   MIMO-OFDM coding techniques

The performance and capacity of a communication system depends on the structure of the transmission side as much as it depends on the channel condition. The complexity of the transmitter as well as the receiver depends on the transmitted signal design. This particular fact has guided the direction of various researches into the design of suitable MIMO coding techniques.   These schemes of coding can be divided into two broad categories: Space Time Coding (STC) and Space Division Multiplexing (SDM). The advantage of STC lies in its ability to increase the robustness of the system, while SDM increases the maximum data rate attainable by the system. STC improves the robustness by using different antennas to transmit the coded symbols over different sub-carriers, while SDM uses a single sub-carrier to transmit independent data streams over different branches. These approaches can be combined in order

to obtain various types of transmission schemes and give a number of options for the systems designer to choose from. The choice depends on various parameters such as bit-error-rate of the channel, sensitivity to channel/interference, computational complexity/simplicity and overall performance of the system. The classification of MIMO-OFDM systems using various algorithms based on the above schemes could be summarized as follow.

Open-loop versus closed-loop techniques:

In open loop transmission schemes the wireless communication systems assume no knowledge of the channel response at the transmitter. On the other hand, the closed loop systems provide the channel information back to the transmitter side by using some kind of feedback mechanism. The information provided via feedback is used to select proper structure of the transmitted signal, i.e. the transmitted power per antenna, coding rate, type of space-time mapping, and/or constellation size.

Transmit diversity versus spatial multiplexing algorithms:

In a richly scattered environment, transmit diversity plays an important role to maintain the robustness of the wireless communication system. Transmission schemes that exploit MIMO diversity use spatial dimensions to add more redundancy, thus keep the data rate equivalent to SISO system in order to increase the BER performance. Space-Time Coding is the principal of generating redundancy by coding across time and spatial dimensions. On the other hand, Space Division Multiplexing (SDM) is employed if the algorithm uses different antennas to transmit multiple data symbols over the channel. SDM scheme is used if high data rates are the main objective of the system. These schemes are not mutually exclusive in nature, and therefore are used in different combinations to produce hybrid schemes that combine transmit diversity and spatial multiplexing and partly benefit from both robustness and data rate enhancement.

Theoretical analysis for the trade-off between diversity gain and Spatial Multiplexing gain was carried out in [38], and an example was given in [15].

Joint Coding (JC) versus Per-Antenna Coding (PAC):

In MIMO system Joint Coding or vertical coding [39] refer to the transmission scheme where the bit stream is encoded first and then separated into different sub-streams of which each is modulated and mapped onto the corresponding transmit antenna. On the other hand, in Per-Antenna coding (or vertical coding) the bit stream is first divided into different sub streams, which are coded with separate codes and then modulated for transmission from the transmission antennas [40]. The first method uses the time and space dimensions to produce redundancy and improve the performance of the signal, while the PAC simplifies the receiver architecture due to the separate encoding provided for the spatial and time dimensions.

The combination of the above transmission approaches result in many possible transmission schemes, which cater for all the needs of communication systems. Various parameters like capacity performance, bit error-rate performance, computational complexity/simplicity and output SNR vary from one scheme to another resulting in a wide range of trade-offs that need to be considered. An overview of some standard MIMO transmission techniques is given in the next subsections.

2.3.1 Space-Time coded MIMO-OFDM

The Space-Time coding techniques use the space and time dimensions to code the message data symbols. The resulted transmission scheme exploits the spectral efficiency of the MIMO system by adding more redundancy in order to improve robustness. In order to code the signal over space and time dimensions without compromising its efficiency, the criteria defined in [52]

could be used. The algorithm defines the upper bound analysis on the pair-wise error probability. Considering two code words C and E, the following quantities should be maximized

- Diversity order, which is the measure of the exponential decrease of error-rate, with respect to the SNR (in a linear scale). For channels with independent fading, the diversity order is quantified as $r \cdot N_r$, with r representing the least rank possible for the codeword difference matrix $C - E$. Thus, to maximise the diversity order, r must be maximised. Therefore, this criterion is called the rank criterion.

- The SNR gain with respect to uncoded scheme with similar diversity order is known as coding gain. This can be calculated by the minimum product of the nonzero eigenvalues of $(C - E)(C - E)^H$ over all possible pairs of codewords. The product of eigenvalues produces the determinant of the matrix, hence naming this criterion as determinant criterion.

These criteria are difficult to relate to traditional code designs [42]. Recently, however, it has been shown that under certain properties the traditional code design criterion of maximising the minimum Euclidean distance ($\|C - E\|$) between any pair of code words is more appropriate [43][44].

The rank criterion, the determinant criterion, and, later, the Euclidean distance criterion, have stimulated various design activities, which have resulted in different space-time codes, among these codes, Space-Time Block Codes (STBCs) are considered the most widely used in today's wireless technology.

Space-Time Block Codes (STBC)

In [14] Alamouti presents a remarkably simple scheme to achieve transmit diversity, for an array of two elements, without any loss of bandwidth. The scheme transmits two symbols over two time periods. In the simplest case, the receiver has only a single element, though extensions are possible to receivers of multiple elements as well.

The 2 x 1 MIMO system shown in Figure 2-9, denotes two symbols to be s_1 and s_2. In the first symbol interval, transmit s_1 from the antenna #1 and s_2 from the antenna #2. In the next symbol interval, transmit $(-s_2^*)$ from antenna #1 and $(s_1^*))$ from antenna #2 where the superscript * represents conjugation.
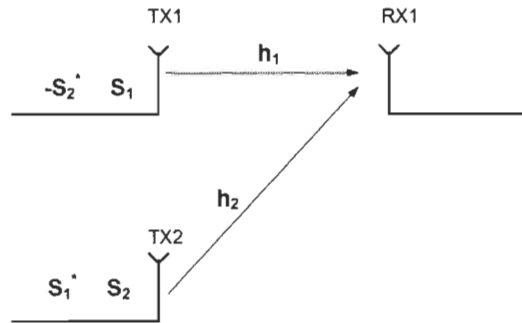
Figure 2-9 Alamouti STBC 2 Tx and 1 Rx

The channel from the two antennas to the receiver is assumed constant over both intervals $(2T_s)$. The two transmit antennas have a total energy budget of E_s, each symbol is transmitted with half the energy. Overall, the received signal over the two symbol intervals (y_1 and y_2) can be written as

$$y_1 = \sqrt{\frac{E_s}{2}}[h_1 s_1 + h_2 s_2 + v_1] \tag{2.59}$$

$$y_2 = \sqrt{\frac{E_s}{2}}[-h_1 s_2^* + h_2 s_1^* + v_2] \tag{2.60}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \sqrt{\frac{E_s}{2}} \begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \Rightarrow Y = H \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + V \tag{2.61}$$

Note that the second entry in vector **x** is the conjugate of the data received on the second symbol interval.

$$\widehat{S} = \begin{bmatrix} \widehat{s_1} \\ s_2 \end{bmatrix} = H^H Y = \sqrt{\frac{E_s}{2}} H^H H \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + H^H V \tag{2.62}$$

$$\Rightarrow \widehat{s_1} = \sqrt{\frac{E_s}{2}} [|h_1|^2 + |h_2|^2] s_1 + h_1^* v_1 + h_2 v_2^* \tag{2.63}$$

$$\widehat{s_2} = \sqrt{\frac{E_s}{2}} [|h_1|^2 + |h_2|^2] s_2 + h_1^* v_2^* + h_2^* v_1 \tag{2.64}$$

Note that the equations for y_1 and y_2 include the squares of the two channel magnitudes, i.e., the received signal incorporates order-2 diversity. In addition, two symbols are transmitted over two symbol intervals and no bandwidth is wasted. This is the beauty of Alamouti's scheme. With a remarkably simple arrangement of transmitted and received data coupled with purely linear processing, order-2 diversity is achieved without any loss in bandwidth. The only 'penalty' is the halving in transmitted energy per symbol.
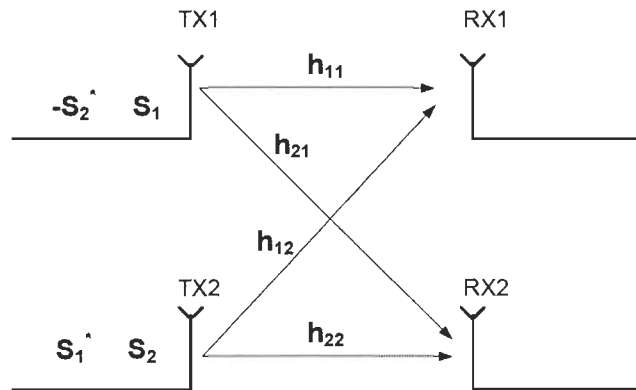
Figure 2-10 Alamouti STBC 2 Tx and 2 Rx

Now let us consider the 2 x 2 MIMO system shown in Figure 2-10

The received signal in the first time slot is,

$$\begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} v_1^1 \\ v_2^1 \end{bmatrix} \tag{2.65}$$

.Assuming that the channel remains constant for the second time slot, the received signal is then

$$\begin{bmatrix} y_1^2 \\ y_2^2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} -s_2^* \\ s_1^* \end{bmatrix} + \begin{bmatrix} v_1^2 \\ v_2^2 \end{bmatrix} \tag{2.66}$$

where

$\begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix}$ are the received information at time slot 1 on receive antenna 1, 2 respectively,

$\begin{bmatrix} y_1^2 \\ y_2^2 \end{bmatrix}$ are the received information at time slot 2 on receive antenna 1, 2 respectively,

h_{ij} is the channel from i^{th} receive antenna to j^{th} transmit antenna,

s_1, s_2 are the transmitted symbols,

$\begin{bmatrix} v_1^1 \\ v_2^1 \end{bmatrix}$ is the noise at time slot 1 on receive antenna 1, 2 respectively and

$\begin{bmatrix} v_1^2 \\ v_2^2 \end{bmatrix}$ is the noise at time slot 2 on receive antenna 1, 2 respectively.

Combining the equations at time slot 1 and 2,

$$\begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} v_1^1 \\ v_2^1 \\ v_1^{2*} \\ v_2^{2*} \end{bmatrix} \tag{2.67}$$

Let us define $H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix}$

To solve for $\begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$, we know that we need to find the inverse of H.

We know, for a general m x n matrix, the pseudo inverse is defined as,

$$H^+ = (H^H H)^{-1} H^H \tag{2.68}$$

The term,

$$(H^H H)^{-1} = \begin{bmatrix} |h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2 & 0 \\ 0 & |h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2 \end{bmatrix} \tag{2.69}$$

Since this is a diagonal matrix, the inverse is just the inverse of the diagonal elements, i.e

$$(H^H H)^{-1} = \begin{bmatrix} \frac{1}{|h_{11}|^2+|h_{21}|^2+|h_{12}|^2+|h_{22}|^2} & 0 \\ 0 & \frac{1}{|h_{11}|^2+|h_{21}|^2+|h_{12}|^2+|h_{22}|^2} \end{bmatrix} \tag{2.70}$$

The estimate of the transmitted symbol is,

$$\cdot\begin{bmatrix} \widehat{s_1} \\ s_2^* \end{bmatrix} = (H^H H)^{-1} H^H \begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} \tag{2.71}$$

As the code matrix is orthogonal, simple single-symbols detection could be applied on Alamouti code due to the inherited fast decoding ML property. Using the theory of orthogonal designs, Alamouti STBC was extended to more than two antennas, however, in this case the maximum data rate is ¾ and no extra coding gain is achieved. This limitation is due to the fact that STBC was initially designed for channels exhibiting flat fading rather than frequency selective fading where the large delay spread cannot be tolerated by STBC [45]. This problem is solved in [26],[53], where the proposed design was able to achieve the multipath as well as space diversity in a single MIMO-OFDM system. The Space-Time codeword is copied over other sub-channels. Although this impacts the rate of the system, the orthogonality of the matrix allows the single-symbol to be decoded. This set up can exploit the multi-path diversity of the system by repeatedly transmitting it over various sub-channels. The theoretical representation of a Quasi-Orthogonal STBC (QO-STBC) is introduced on basis of an array processing technique in [46]. This technique uses the null spaces of split-up channel matrices to decompose the received QO-STBC OFDM symbol array. By using a parallel decoder, the system can harness the full diversity of the frequency selective channels. However, the complexity of this scheme is fairly high.

2.3.2 Space Division Multiplexing (SDM)

Instead of space-time coding, which utilizes the space dimension by using redundancy to enhance the robustness of the system, the data rate can be increased by using multiple antennas. This can be implemented by using a single carrier frequency to transmit multiple data streams over different antennas. Even though these parallel data streams are mixed up in the air, multiple antennas at Rx end can be used to successfully recover all the transmitted data by using suitable algorithms. This technique of using several antennas is called Space Division

Multiplexing (SDM). While this method provides the best possible data rates, the main disadvantage is that no redundancy is added and, thus, it might suffer from poor link reliability. To overcome this problem additional channel coding can be introduced. This, however, reduces its data rate advantage.

The low level of complexity on the receiving end of the system adds to the advantages of SDM in addition to its high data rates. This is especially helpful when the system uses a huge number of transmit antennas. The complexity of Space-time receiver is due to the fact that they require joint detection of space and time dimensions simultaneously from the received signal. The SDM removes such complications by separating the space and time dimensions of the transmitting signal. However, this leads to lack of redundancy of signal, ending in low performance of the system. One method of overcoming this deficit is shown in [47][48][49] which shows that system must be designed to alternate the usage of Space coding as inner code and Time coding as the outer one. This kind of coding is denoted as JC (Joint Coding) or PAC (Per-Antenna Coding). In the concept of PAC, the TX antennas can be either co-located or not. The latter option can be seen as a multiple access scheme and is generally called Space Division Multiple Access (SDMA). From a reception point of view, the two options are not very different and the detection at the receiver can be performed by multi-user detectors operating in the spatial domain. One of the main contributions of [15] was to recognize that, because each transmit-antenna encounters a different propagation channel, PAC incurs a capacity penalty. Hence, a diagonally layered architecture was proposed (Diagonal Bell-Labs Layered Space-Time (D-BLAST)) in which successive symbols of a given encoded data stream are sent on a different TX antenna, by cyclically selecting another TX antenna per symbol period. In this way, each data stream is exposed to the distinct propagation channels within the MIMO

channel. In fact, this eliminates the capacity penalty compared to cases in which no cycling is used [50].

While the D-Blast covers the deficit of capacity, it cannot prevent an increase in the complexity of receiver end in the communication system. If this cyclic mapping system is not used, the complexity of the receiver instantly decreases. This brings us back to the original problem of drop in channel capacity. As the receivers in the PAC model can operate in the spatial domain, it can switch between linear and non-linear forms of processing. MMSE and ZF are examples of SDM algorithms with linear processing while V-BLAST and ML are non-linear ones.

2.3.3 Space-Frequency Block Coding MIMO-OFDM

The strategy of using multidimensional coding across different antennas (space) and OFDM sub-channels (frequency) is called Space Frequency Block Coding (SFPC) [23]. This coding can be easily realized by using two antennas to spread the Alamouti code over two sub-channels inside a single OFDM block. Although this approach is useful in achieving space diversity gain, the maximum diversity gain cannot be accomplished for frequency selective MIMO channel which is denoted by $N_t N_r L$, where N_t is number of Tx antennas, N_r is the number of Rx antennas and L is the number of the channel paths. The work in [24] was derived to achieve the full diversity in MIMO multi-path fading channels using SFBC coding. The DFT matrix is used as a multiplying factor to modulate the input stream in order to achieve the full diversity [21]. The codes generated using this principle have full diversity but experience a huge loss in bandwidth efficiency. The upper limit of symbol rate is set at $1/(N_t L)$. The scheme in [25] was proposed to address this deficiency. The design changes enabled the SFBC to attain full diversity by duplicating the rows of ST matrix on different sub-channels inside the same OFDM block. The

repetition of codes on L different sub-channels increases the data rate of the SFBC codes beyond the value in [21]. These proposed changes cannot increase the rate beyond 1/L. The recent proposal for achieving full diversity SFBC at a rate of (rate-1) was described as a part of [29] which was designed for any number of transmitting antennas with arbitrary values in power delays. Consider the information denoted by vector S is coded with a rotation matrix, resulting in a coded vector X ($X = \Theta S$). This vector X is now spread across different antennas and OFDM sub-channels before transmission. This splitting allows the code to achieve the desired rate of 1. The design of rotation matrix Θ is done with an eye on the signal space diversity it can produce when rotated with the signal constellation [51]. This design change allows a multi-path diversity gain of 2. The latest advancements given in [28] regarding the high rate SFBC, allow the system to achieve any rate as well as the full diversity of MIMO-OFDM system for any number of antennas used for transmission. The slight drawback of using a zero padding matrix when N is a rational multiple of $N_t L$ does not ensure a steady transmission rate of N_t. In [52] a universal design is proposed, which shows SFBC as a special case of STF coding which can ensure a steady rate at N_t as well as the full diversity for an arbitrary number of antennas and power delay profiles. The construction, which is based on design of TAST (threaded algebraic space-time) code [53], modulates the input signal with algebraic component codes. This modulated signal is divided into threads based on the component codes and modulated again over space and frequency. This method contributes to an increase in the complexity on the decoding end of transmission. It can be reduced by using Sphere decoding [54], which is a slightly modified version of ML decoding.

Multiple user space-frequency coding

Gärtner and Bölcskei conducted a number of recent researches in the direction of increasing the multi-user data rate on MIMO-OFDM systems [27]. The results indicated a necessity of using joint code designs for achieving high rates with multiple user facility. If they are not used, one should depend on the traditional ST/SF codes for a single user to obtain the best possible results. Extensive studies have indicated that the point to point MIMO communication systems use the joint code design across all the transmission antennas. However, the co-ordination of transmission between individual users becomes a sticking point in implementation of multiple access systems. The work in [27] describes a specific case of using a 2 user MAC (Multiple Access Channel), which is designed by using a modified version of Alamouti matrix i.e. a column was rotated and swapped in the matrix. The minimization of error rate in SFC is done by allowing the users to specify their own codebooks, which allows them to have multiple transmissions at the same time. However, it must be noted that the above study had used only 2 users in its scenario where as in real time we have a much higher number of users using the system at the same time. Recent results in this field of research indicate that the diversity gain of any multi-user MIMO-OFDM cannot be maximized beyond the gain attained by a single MIMO-OFDM, if each user gets its own codebook. The multi-user SF code can be used to increase the efficiency of the communications system by minimizing the multi-user interference in the channel. The study in [55] aims to table a proposal to investigate a systematic design to use multi-user SF codes, which helps in implementing a system based on MIMO Frequency Selective Fading MAC. The structured code is rotated twice, once using a constellation rotation followed by a phase rotation (the phase value is unique). The usage of constellation rotation guarantees the complete signals space diversity for each unique user, while the multi-user interference is minimized by employing the phase rotation to the mix. The data rate of every

user in the OFDM system is increased by allocating the coded symbols to all subcarriers. This step is done only after each user employs a unique SF coding. The high data rates and the targeted diversity gain of N_tN_rL can be achieved by each and every user in the system by using the proposal for multi-users discussed above. This result can be reached by assuming perfect knowledge of CSI at the receiver and the usage of high complexity detection algorithm at the receiver such as ML.

2.4   Conclusion

The combination of OFDM with MIMO is very important technique that has been considered as the preferred air-interface technique for current and future wireless communication standards. OFDM provides efficient spectrum utilization and simplified implementation using fast Fourier transform. Additionally, it reduces the equalization techniques at the receiver by converting the wide-band frequency selective channel into a set of narrow band flat fading channel. On the other hand, MIMO is used to mitigate the signal fading due to multiple path transmission by coherently combining the wireless signals at the receiver in order to increases the receive signal-to-noise ratio (SNR).

This chapter provided a detailed background for MIMO-OFDM systems, in which the system and mathematical models were introduced. MIMO detection algorithms were also described along with simulation comparison. It has been showed that ML detection provides the best performance. However, it has the highest implementation complexity. On the other hand, nonlinear SIC algorithms showed better performance than linear algorithms such as ZF and MMSE.

Finally, MIMO-OFDM coding schemes were divided into two main categories. Diversity schemes such as STBC and SFBC are employed to combat fading transmission where the transmitted signal is coded over multiple independent fading paths in space, time, and frequency. On the other hand, spatial multiplexing schemes such as V-BLAST are used to transmit distinct data signals from different antennas to provide a linear increase in the capacity or data rate. Recently, multiple access transmission scheme for MIMO-OFDM is obtained by combining STBC and SFBC. However, its system and decoding complexity is a major obstacle and require further innovative solutions. In the next chapter, a novel STFBC transmission scheme is proposed in order to support multiple access and alleviate the complexity problem.

Chapitre 3 - MIMO-OFDM with parity bit selected and permutation spreading

The idea of employing a linear transform to spread energy of transmitted symbols over OFDM subcarriers in order to obtain diversity advantage was first proposed in [56]. It was further developed in [57] by splitting the full set of subcarriers into smaller blocks and spreading the data symbols across these blocks via short block spreading matrices. By mapping the symbols on other sub-channels, it can exploit the multiple path diversity. However, the system complexity is a major obstacle and the decoding complexity problem has to be tackled. For example, when the spreading block size M is not too large, maximum likelihood (ML) method can be used for detection. When M is large, the complexity of ML detection grows exponentially with M. To reduce complexity, some suboptimal ML detection methods such as sphere decoding are used.

In addition to that, most of the existing ST/SF codes are designed for single user systems only, for multiple access channels (MAC), the single-user ST/SF codes are always applied to each user independently, which results a reduced transmission rate.

In this chapter, a new transmission scheme based on Space Time Frequency (STF) coded MIMO-OFDM family combined with parity bit selected and permutation block spreading methods is proposed. In this scheme, the data symbol to be transmitted across each antenna is spreaded by a spreading code; the choice of this code is determined by the parity bits of the transmitted message vector across the multiple antennas. At the receiver side, the data detection

relies on correlators matched to the different spreading codes used by the transmitter, once the spreading code is identified in the first stage, the probability of error in determining the correct block of information bits is very low. In other words the probability of error is dominated by the errors caused by incorrect spreading sequence determination. This technique was originally proposed in [58] and was recently adapted for MIMO-CDMA in the presence of frequency selective fading channel [59]. A novel approach is developed here to effectively combine parity bit selected and permutation block spreading techniques with MIMO-OFDM to obtain improved bit error rate performance in the existence of frequency selective fading channel and greatly lower system complexity. In addition, the system allows flexible data rates and efficient user multiplexing which are required for next generation wireless communications systems. In MIMO-OFDMA, users are separated in different frequency bands (sub-channels), and each user is coded separately using STBC or SFBC, leading to data rate reduction for each user when the number of users is increasing. The proposed new scheme enables multiple access by joint code design across multiple antennas, OFDM frames, subcarriers, and users. It will benefit from the combined space, time, and frequency diversity and allow users to share subcarriers with a manageable level of multi-user interference. Hence, better spectrum efficiency is achieved while improving bit error rate performance with respect to signal-to-interference rate.

3.1 MIMO-OFDM with parity bit selected and permutation spreading

Let us consider the K user MIMO-OFDM downlink, employing Binary Phase Shift Keying (BPSK) modulation. Figure 3-1 shows the proposed MIMO-OFDM system. The data of each user is spatially separated into N_t transmitting antennas. The substream to be transmitted on antenna i on time interval n is spread by a spreading code vector $c_{k,i}^{(n)}$. The spreading code vector is of dimension $N_c \times 1$:
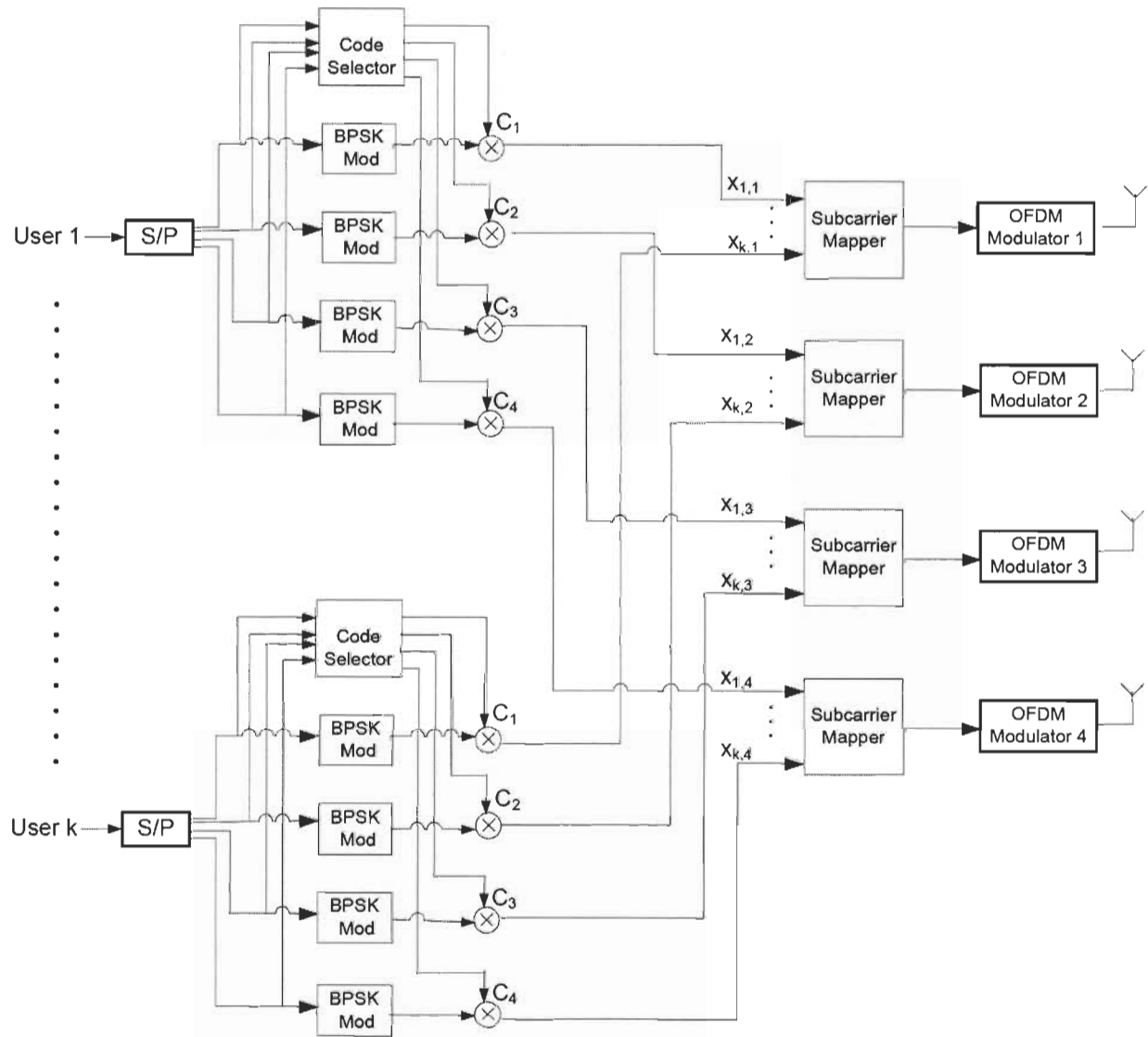
Figure 3-1 4X4 MIMO-OFDM transmitter with Parity Bit Selected Spreading

$$C_{k,i}^{(n)} = \left(c_{k,i}^{(n)}(1) \;\; c_{k,i}^{(n)}(2) \ldots \ldots \ldots \ldots \ldots . c_{k,i}^{(n)}(N_c) \right)^{T} \tag{3.1}$$

Where N_c is the spreading factor and the superscript T represents the transpose operator, $N_c = T_s/T_c$ is an integer number where T_s is the symbol period and T_c is the chip period. The spreading code vector is selected from a set of N spreading vectors $\{c_{k,1}, \; c_{k,2}, \ldots, c_{k,N}, \} \in C_k$.

The wireless media is considered to be a discrete-time baseband channel model with chip-spaced channel taps. We assume block fading chip-spaced channels perfectly known by the receiver. Thus, assuming the same channel order L for all single input single output (SISO) channels, the sampled channel response from the transmit antenna i to the receive antenna j of user k is given by the $L \times 1$ vector:

$$\boldsymbol{h}_{i,j,k} = \left[h_{i,j,k}^{(0)} \quad h_{i,j,k}^{(1)} \quad \cdots \quad h_{i,j,k}^{(L-1)} \right]^T \tag{3.2}$$

For T OFDM frames, we denote the transmitted symbols after spreading from all transmitting antennas for user k as

$$X = \begin{pmatrix} x_{1,k}^1(1) & \cdots & x_{1,k}^1(N_c) & \cdots & x_{1,k}^T(1) & \cdots & x_{1,k}^T(N_c) \\ & \vdots & & \vdots & & \vdots & \\ x_{N_t,k}^1(1) & \cdots & x_{N_t,k}^1(N_c) & \cdots & x_{N_t,k}^T(1) & \cdots & x_{N_t,k}^T(N_c) \end{pmatrix} \tag{3.3}$$

Where

$$x_{i,k}^u(1 \text{ to } N_c) = \boldsymbol{C}_{i,k} s_{i,k}^u$$

is the kth user spreaded symbol s for antenna i in OFDM frame u with spreading vector $\boldsymbol{C}_{i,k}$, by considering one OFDM frame with N_f sub-carriers, X could be written as

$$X = \begin{pmatrix} x_{1,k}^1 & \cdots & x_{1,k}^{N_f} \\ \vdots & \ddots & \vdots \\ x_{N_t,k}^1 & \cdots & x_{N_t,k}^{N_f} \end{pmatrix} \tag{3.4}$$

The symbols in each row of $\mathbf{X}$ are then OFDM modulated with cyclic prefix (CP) insertion, and transmitted from the respective antenna.

Code selection in parity bit selected case:

For each data block of N_t symbol length, the set M of all possible message vectors has 2^{N_t} different elements. Each data block (prior to IFFT processing block) is input to a systematic linear encoder which produces a set of parity bits $P = [p_1, p_2, \ldots, p_k]^T$, where $K = log_2(N)$. The set of message vectors that produce the all 0 parity vector is denoted by M_1. This set is closed under modulo-2 addition. The set of message vectors that produces parity vector p_i is denoted M_i. For example, for 4 transmitting antennas, the set M has 16 elements. Choosing $N = 8$ we can partition M into 8 different cosets. We choose the coset leader to have the greatest possible minimum Hamming distance, therefore: $M_1 = \{0000, 1111\}$ $M_2 = \{0001, 1110\}$ ...., $M_8 = \{0111, 1000\}$. For each bit in the code word to be transmitted, we use the same code. Hence:

$$C_1 = C_2 = \cdots . C_{N_t} = C_m.$$

Code selection in permutation case:

Unlike the previous technique, in permutation spreading each transmitting antenna is allocated different spreading code. By determining which coset the message comes from, a unique permutation of the spreading codes are used to spread the message symbols. Each permutation employs N_t of the N spreading waveforms, to minimize the number of spreading codes that each permutation has in common, the design of different spreading permutations is based on t-designs which are used in permutation modulation schemes. Table 3-1 lists the spreading permutations when 4 transmitting antennas are used.

Table 3-1 Spreading permutations for MIMO-OFDM with 4 antennas

Coset	Message vectors	C_1	C_2	C_3	C_4
M_1	0000 1111	c_1	c_3	c_5	c_7
M_2	0001 1110	c_8	c_1	c_4	c_5
M_3	0010 1101	c_2	c_4	c_3	c_8
M_4	0011 1100	c_5	c_2	c_6	c_3
M_5	0100 1011	c_6	c_7	c_1	c_4
M_6	0101 1010	c_3	c_6	c_8	c_1
M_7	0110 1001	c_7	c_8	c_2	c_6
M_8	0111 1000	c_4	c_5	c_7	c_2

Time-Frequency mapping

Figure 3-2 shows the time and frequency mapping for the data symbols before IFFT modulation, here the Time-Frequency mapping is described for one user at a particular transmit antenna. Without loss of generality all users will use the same mapping method at each antenna.

Let's consider the mapping for the first row of the matrix X in (3.3), and also consider a number

of OFDM frames equal to the length of the spreading code N_c, assume $x_1^{c_1}$ occupies OFDM

frame 1 at subcarrier c_1, $x_2^{c_2}$ occupies OFDM frame 2 at subcarrier c_2,..., and $x_{N_c}^{c_{N_c}}$ occupies

OFDM frame N_c at subcarrier c_{N_c}. The next transmitted symbol $x_1^{c_1+1}$ occupies OFDM frame 1

at subcarrier $c_1 + 1$, $x_2^{c_2+1}$ occupies OFDM symbol 2 at subcarrier $c_2 + 1$,..., and $x_{N_c}^{c_{N_c}+1}$

occupies OFDM frame N_c at subcarrier $c_{N_c} + 1$. The process is repeated until all symbols are

mapped in N_c OFDM frames.
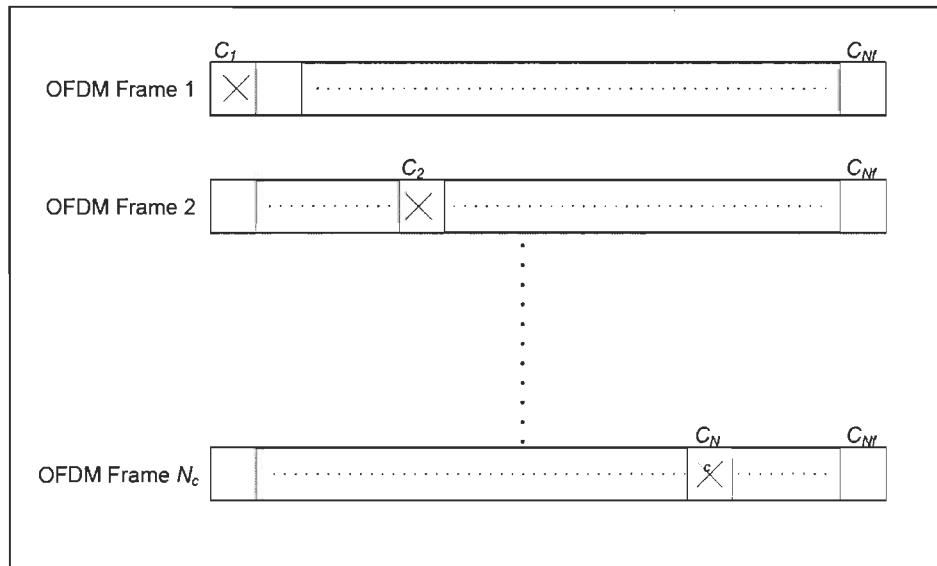
Figure 3-2 Time-Frequency mapping

The above mapping of the subcarriers is based on the analysis introduced in [60]. Here the

OFDM transmitted data for frame 1 is $F = [x^{c_1}, x^{c_2}, ..., x^{c_{N_c}}]^T$ where $F^T \subset$ IFFT matrix with

size N_f. F is a wide matrix $N_c x N_f$ where the rows are picked from an IFFT matrix and complex

transposed (Hermitian), those rows are picked as N_f/N_c in order to guarantee the orthogonality

among the rows and columns to achieve independent fading for each signal and hence

maximizing frequency diversity.

At the receiver, since we are interested in single user detection, the received signal at antenna j can be written as:

$$y_j^{(n)} = \sum_{i=1}^{N_t} X_{k,i}^{(n)} h_{i,j,k} + v_j^{(n)} \tag{3.5}$$

where $X_{k,i}^{(n)}$ is the transmitted data by antenna i of user k at instant n , $v_j^{(n)}$ encompasses the complex Gaussian noise with variance σ_n^2.
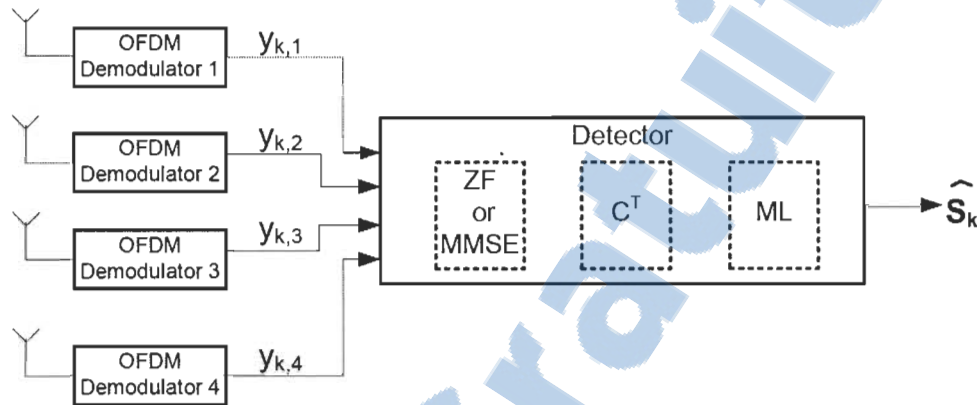
Figure 3-3 MIMO-OFDM receiver for parity bit selected and permutation spreading for Nr = 4

The received signal shown in Figure 3-3 will be first sent to the reverse OFDM frame (cyclic prefix removal and FFT), then in the presence of full channel knowledge, the spreaded data $\hat{X}$ are estimated by linear detection algorithm such as Zero Forcing or MMSE. In the first case $\hat{X}$ is given by

$$\hat{X}^{(c=1:N_f)} = H^+ Y^{(c=1:N_f)} \tag{3.6}$$

Where H is an $N_t N_r L$ channel matrix for each OFDM sub-channel c and the superscript $^+$ denotes the Moore-Penrose (MP) pseudo inverse where

$$H^+ = (H^H H)^{-1} H^H \tag{3.7}$$

In case of MMSE $\widehat{X}$ is given by

$$\widehat{X}^{(c=1:N_f)} = arg\ \min_W \left\| X^{(c=1:N_f)} - WY^{(c=1:N_f)} \right\|^2 \tag{3.8}$$

Where W is the mean square error matrix defined in equation (2.39).

For each transmit antenna, we apply a correlator matched to the signature used by the transmitting antenna. The output of the correlator matched to the transmit antenna i of user k using the received signal at receiving antenna j for N_f sub-channels is

$$\mu_{i,j,k}^{(c=1:N_f)} = \sum_{i=1}^{N_r} C^T X_i^{(k=1:N_f)} \tag{3.9}$$

Where

$\mu^{(k=1:N_f)}$ is the output of the matched filter

C^T is the transpose of codes matrix

The index of the highest power in μ will determine the original used code at the receiver, after determining the coding sequence; the receiver can make use of the fact that each block of information symbols across the multiple antennas is carried by a specific coding sequence.

Once the code is identified in the first stage, the maximum likelihood is used to detect the original data; hence the probability of error in determining the correct block of information symbols is very low. In other words the probability of error is dominated by the errors caused by incorrect coding sequence determination.

In the above transmission scheme, we have introduced three diversity sources to the system, the first one is frequency diversity introduced by the block spreading, the second one is the time diversity obtained by mapping the spreaded data over different OFDM frame, and finally the spatial diversity introduced by the MIMO configuration, this combination improves further the

performance of the whole system as we will see in the simulation result. On the other hand, this spreading improves MIMO detection since the inherited code within the received MIMO vector will be used to identify the original transmitted message.

3.2 Simulation set-up

From practical implementation aspects, some simulation issues may need to be considered and appropriate simulation parameters should be accordingly defined. This section discusses some of the crucial simulation issues according to the proposed system design. Then the parameters used for the simulations are given in detail.

3.2.1 Power requirements

If the system is radiation power limited, we assume that the total transmit power from the multiple antennas for the proposed MIMO-OFDM system is the same as the transmit power from the single transmit antenna for the conventional OFDM system with and without spreading. In order to address this assumption, we normalize the power of signals transmitted from each antenna of the proposed system. Although the reduction of the signal power for each transmit antenna incurs a penalty in the BER performance, it provides a fair environment for comparison with known single antenna systems. Moreover, the reduction of the power in each transmit chain leads to cheaper, smaller, or less linear power amplifiers in practice. It is often less expensive to employ lower power amplifiers rather than a single full power amplifier.

3.2.2 Channel conditions

In practice, the primary requirement for diversity improvement is that the signals transmitted from the different antennas be sufficiently uncorrelated and that they have almost equal average power. For this purpose, we assume that the amplitudes of fading from each

transmit antenna to each receive antenna are mutually uncorrelated and the average signal powers from each transmit antenna to each receive antenna are normalized to be the same. Consider the proposed system using two transmit antennas and two receive antenna, we use the tapped delay line model to create four frequency selective multiple path channels and the multiple path diversity order for each channel is assumed to be L_P. In order to obtain the full diversity in practice, the antennas should be sufficiently separated and therefore each channel for a specific link can be considered to be independent. MIMO brings diversity gain to systems in practice due to the advantage of multiple path channels. In our simulation work, however, the diversity gain generated by channels is unified among different systems in order to make performance comparisons under the normalized SNR. Therefore, the channel normalization is required. The coefficients for each channel of the two transmit antennas and two receive antenna system are modeled as complex Gaussian random variables with zero-mean and variance $1/4L_P$.

3.2.3   Parameters for simulations

An important issue about the simulation parameters is how to select a proper spreading code size for spreading. Intuitively, the larger the spreading code size is, the better the system performance will be. However, a larger spreading code size also implies higher implementation complexity and lower system performance. On the other hand, if the spreading code size is too small, the available diversity provided by the multiple path channels cannot be fully exploited. Therefore, determining a proper spreading code size according to the available multiple path diversity order is of significance to the proposed spreaded MIMO-OFDM scheme. In order to make a comparison between different symbol group sizes, we use $N_c = 8$ or $N_c = 16$ to be the size of spreading code, respectively. For MIMO configuration two or four antennas are applied at the transmitter and at the receiver; either two or four antennas are used respectively to achieve

different diversity orders. We use the BPSK modulation as the signal constellation where information bits are mapped to modulated symbols. The parameters used for the simulations are given in Table 3-2.

Table 3-2 Simulation parameters

Parameter	Value
IFFT frame length	256
Cyclic Prefix length	64
Antenna configuration N_t x N_r	2 x 2 and 4 x 4
Modulation	BPSK
Number of simulation runs	10000
Spreading Code Length	8 or 16

3.3 Numerical simulation results

In this section, numerical simulation results are provided. First, a BER performance comparison between the conventional STBC MIMO-OFDM system, and the proposed MIMO-OFDM with parity bit selected or permutation spreading system is presented for 2X2 antenna configurations, where the benefits provided by spreading can be easily recognized. Then, we increase the antenna configuration to 4X4 case for the proposed system and consequently the related performance comparison is shown. It demonstrates that the system performance can be further improved by increasing the number of antennas. Next, BER performance for systems with a larger spreading code size is shown to demonstrate that a further performance improvement can be obtained by increasing the spreading code size for the proposed scheme. After that, we present the performance comparison between the systems using different linear equalizations. According to our previous analysis, the MMSE equalization always achieves better performance than the ZF equalization. Finally, a comparison between the proposed MIMO-OFDM scheme and the conventional STBC MIMO-OFDM is presented for multi-user

access. The simulation results proves that the proposed scheme is achieving better BER performance as the separation between users is improved due to the increased diversity of the proposed system.

Simulation results for the proposed system

Figure 3-4 shows the BER performance comparison between the conventional STBC MIMO-OFDM system and the proposed MIMO-OFDM system with both parity bit selected and permutation spreading approach. In this case, the spreading code size is M = 8 with two transmit antennas and two receive antennas are used for MIMO transmission.

As we can see from Figure 3-4, both MIMO-OFDM systems with parity bit selected and permutation spreading provides improved BER performance over the conventional STBC MIMO-OFDM system. For example, the proposed MIMO-OFDM system provides 5dB SNR gain in case of parity bit selected spreading and 7dB SNR gain in case of permutation spreading over the conventional STBC MIMO-OFDM system at BER= 10^{-3}. This is because the new scheme greatly mitigates the effect of frequency selective fading channels by spreading data symbols across all subcarriers for each user. We can also observe that permutation spreading is achieving better results than parity bit selected due to the fact that using different spreading codes for each antenna in the first case greatly reduces the correlation among symbols sent by each antenna while in the second case we used the same spreading code. This anticipated performance improvement is approved by the simulation result. As illustrated, a 2.5dB SNR gain is achieved by the permutation spreading system at BER= 10^{-3}, compared with the parity bit selected spreading system.
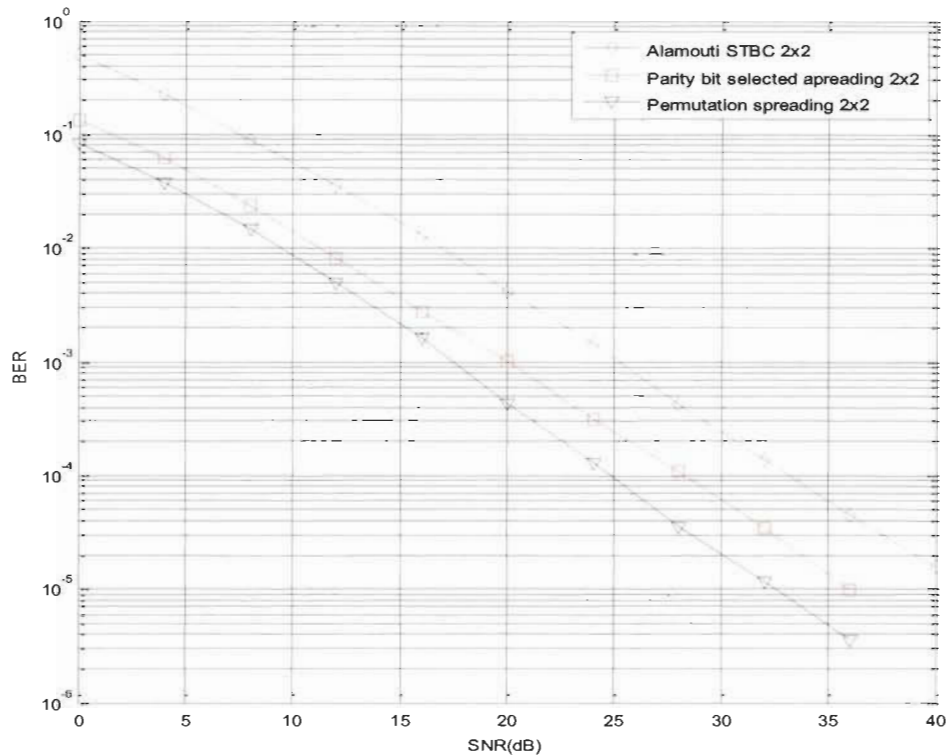
Figure 3-4 BER performance for 2X2 MIMO-OFDM with parity bit selected spreading, permutation spreading, and Almouti STBC

As we described earlier, increasing the number of antennas at the receiver provides MIMO system with a higher order of diversity, therefore achieving a further performance improvement. At the same time by increasing the number of antennas at the transmitter side improves the multiplexing gain, hence increasing the spectral efficiency. Simulations are carried out to demonstrate this anticipated performance improvement. In this case, we use 4X4 antenna configurations and maintain other parameters unchanged. Figure 3-5 presents a BER performance comparison between conventional STC MIMO-OFDM and the proposed system with spreading for both 2X2 and 4X4 configurations. As we can see, the 4X4 system further improves system BER performance gain over the 2X2 system as shown in Figure 3-4. Therefore, we may conclude that increasing the number of receiving antennas can provides a

higher diversity order and as a result, the system performance can be continually improved. On the other hand the multiplexing gain is doubled by using four antennas at the transmitter instead of only two.
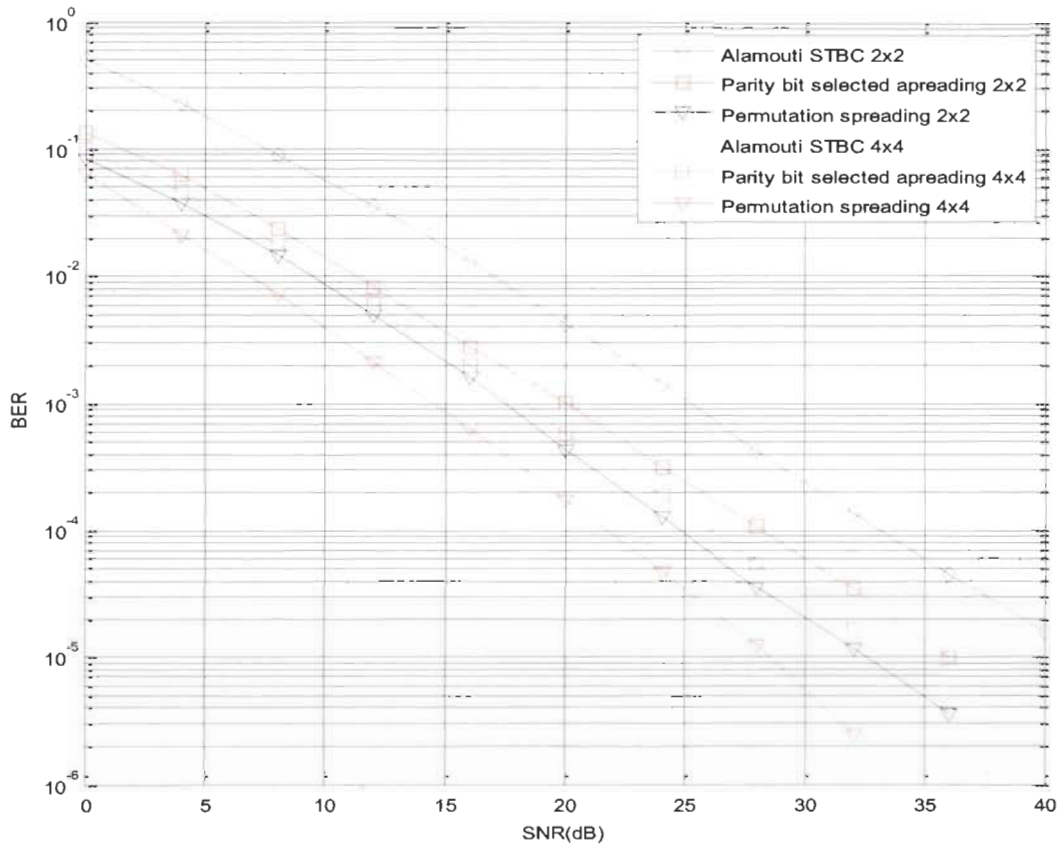
Figure 3-5 BER performance for MIMO-OFDM schemes with 2X2 and 4X4 configurations

Simulation results for a larger spreading factor

The selection of a proper spreading code size or simply the spreading factor N_c usually is based on the number of the multiple path channels presented in the system. If the spreading factor is too small, the available diversity provided by the multiple path channels cannot be fully exploited. Normally, the larger the spreading factor is, the better the system performance will

be. However, a larger spreading factor reduces the spectral efficiency of the system. In our simulations, the proper spreading factor is strictly related to the mulitpath diversity order, which is defined as L_P. The principle that we use to determine a proper spreading factor for our proposed system is that N_c should be equal to or slightly larger than the multiple path diversity order L_P. For example, given the multiple path diversity order $L_P= 8$, the proper spreading factor should be 8 or 16. As shown previously, Figure 3-4 and Figure 3-5 and present the simulation results, where $N_c= 8$ is used for the spreading factor. Then, we increase the spreading factor to $N_c= 16$ and as a result, different simulation results are shown as following.
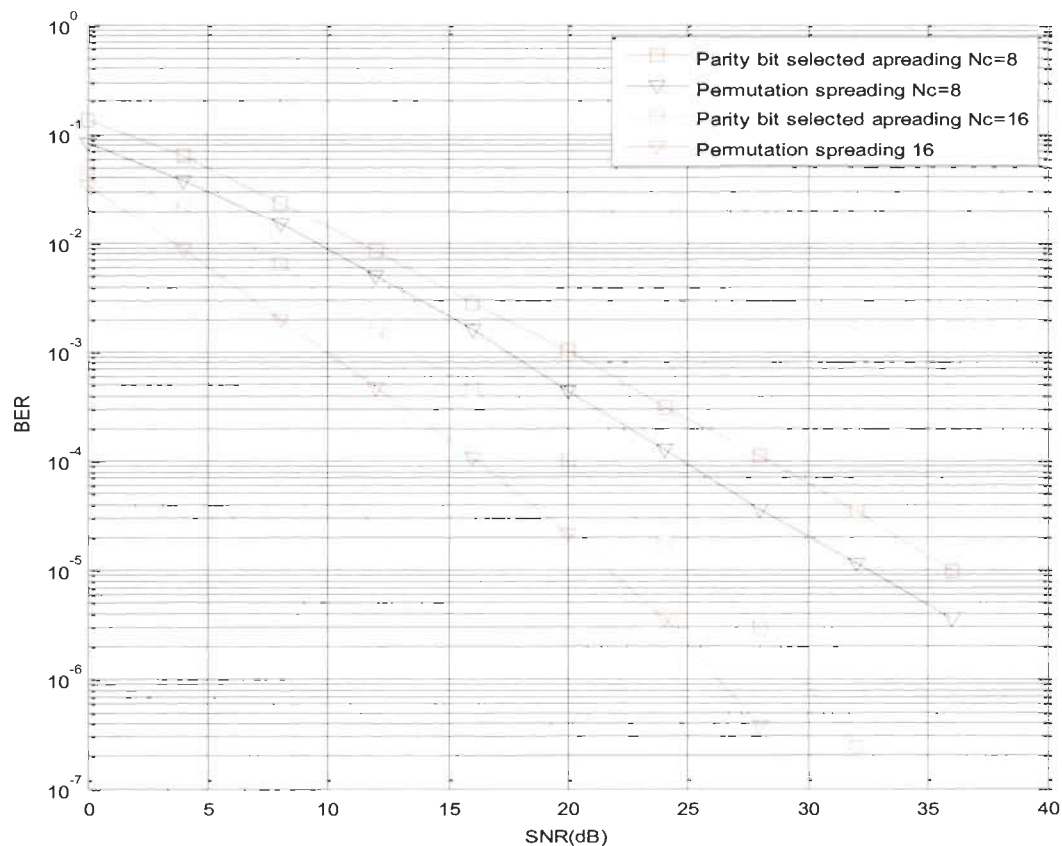
Figure 3-6 BER performance comparison for MIMO-OFDM with permutation spreading, when Nc = 8 and Nc = 16

Figure 3-6 presents a BER performance for the proposed system with spreading factor N_c=16, compared to N_c=8, in this case, we can see that the performance curves of the proposed system have a significant improvement when the spreading factor increases. For example, to achieve BER= 10^{-3}, MIMO-OFDM system with N_c= 8 requires 20 dB SNR for parity bit selected and 17.5 dB SNR for permutation scheme. On the other hand, for N_c= 16 , the parity bit selected scheme requires only 13.5dB SNR, while the permutation scheme requires 10 dB SNR. Therefore, the system with a larger spreading factor achieves better BER performance over the system with a smaller spreading factor, the above simulation was set for 2X2 MIMO configuration.

Simulation results for different linear equalizations

The system performance does not only depend on the proposed system spreading schemes, which although are the most important factors, but also relies on the different equalizations used for detection process. As we described before, the ML equalization increases computation complexity exponentially, therefore it is not widely used in practice even though it is able to theoretically provide the best system performance. In our project, we used linear equalizations, which include the MMSE equalization and the ZF equalization. Of the two linear equalization techniques, MMSE equalization performs better as it considers the effect of noise in channels while the ZF equalization assumes no noise is present. Consequently, the MMSE equalization is expected to result in better system performance than the ZF equalization. This anticipated performance improvement by MMSE equalization over ZF equalization is demonstrated by the following simulation result. Figure 3-7 presents a BER performance comparison between systems with different linear equalizations. In this case, we apply two linear equalizations, MMSE and ZF equalizations, for Alamouti STBC, parity bit selected, and permutation

spreading systems respectively. As we can see, for all systems the MMSE equalization achieves better BER performance than the ZF equalization does.
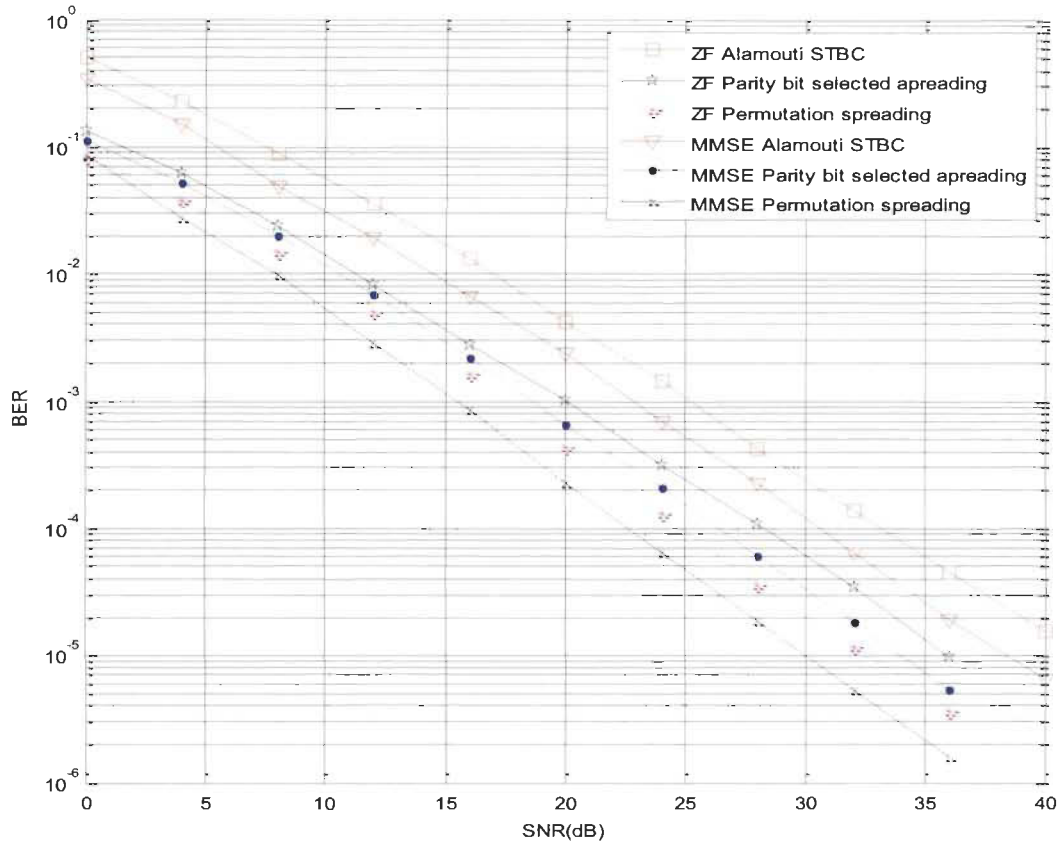
Figure 3-7 BER performance for MIMO-OFDM schemes with MMSE and ZF equalizations

Simulation results for multi-user

Figure 3-8 shows the BER performance comparison for multi-user MIMO-OFDM with parity bit selected and permutation spreading versus the conventional MIMO-OFDMA with STBC. The antenna configuration is set to 2X2, and the simulation is carried out for 1, 4 and 8 users respectively. It is clear that the proposed scheme with both spreading techniques has better performance in multi-user environment as we were able to maintain maximum achievable

diversity due to the signal spread over space, time and frequency. In addition to that, the proposed scheme is capable of achieving better spectrum utilization due to the fact that subcarriers are shared among users and spreading codes are used for user's separation. Furthermore, spreading the signal on the three domains provides extra flexibility in user multiplexing and scheduling without the need of CSI at the transmitter side.
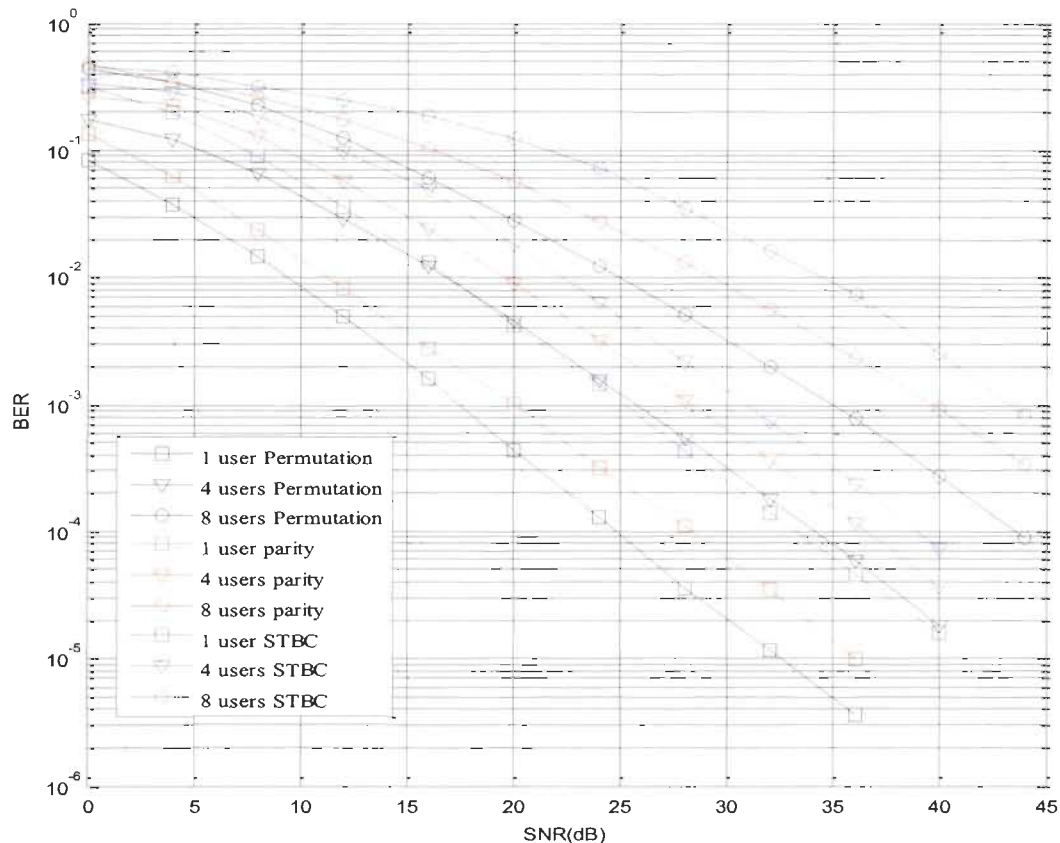
Figure 3-8 BER comparison for multi-user MIMO-OFDM with parity bit selected and permutation spreading Vs MIMO-OFDMA with STBC

3.4 Conclusion

In this chapter, a novel open-loop transmission scheme for MIMO-OFDMA with parity bit selected and permutation spreading is proposed, the transmitted data stream is spread over

space, time, and frequency in the presence of frequency-selective channel. Compared with conventional MIMO-OFDM with STBC, a significant performance improvement is achieved since spreading in the three domains greatly mitigate the effect of frequency selective fading channels by providing more diversity at the receiver. Using parity bit selected or a permutation technique for selecting the spreading code at the transmitter has greatly reduced the detection at the receiver side. This reduction comes from the fact that, by acquiring the spreading code at the receiver, the transmitted data block could be identified. The mathematical model is presented to describe the proposed MIMO-OFDMA scheme in detail. The receiver design is also presented where channel equalization is applied first, then despreading is performed by using matched filter to determine the spreading code and separate the data for each user.

After introducing the simulation parameters and conditions, the BER performance of the proposed system has been evaluated for different antenna configurations, spreading code lengths, channel equalizations, and multi-user access methods. Simulation results have showed that the new scheme provides better performance as compared to conventional MIMO-OFDM with STBC because it is capable of maintaining maximum achievable diversity on the receiver side. The FPGA implementation of the proposed scheme will be introduced in the next chapter, the implementation process starts by introducing the design methodology and the implementation platform. Then the algorithms of the proposed scheme are broken down into smaller mathematical functionalities, and then Register Transfer Logic (RTL) approach is used to map each function into hardware using either VHDL code or IP cores. Finally, the functional validation and synthesis results will be introduced.

Chapitre 4 – Design & Implementation of MIMO-OFDM system

In this chapter the implementation details for the proposed MIMO-OFDM system are presented for 2x2 antenna configurations, the same design parameters that was introduced in Table 3-2 are used here for implementation, i.e. OFDM frame length of 256 in addition to 64 CP is used. First in section 1 the rapid prototyping design methodology for Floating-Point representation is introduced, by which MIMO-OFDM algorithms are taking all the way from theory to the final FPGA implementation. The reason for starting with Floating-Point model is two folds. First, Floating-Point implementation gives high precision and resolution in calculations, which allows verifying the functional operation of the implemented design against those of Matlab model. Second, data dependencies could be identified using the initial Floating-Point RTL model, which in turn could be used in the optimization phase. Next, a real-time hardware design platform is proposed in section 2; it supports hardware-in-the-loop testing for MIMO-OFDM algorithms. In this platform, UART module is designed and integrated on the same FPGA chip to provide serial communication between Matlab and the FPGA board. In section 3, the FPGA architecture for the proposed MIMO-OFDM Transceiver is introduced. The architecture is divided into smaller sub-modules and the detailed design for each sub-module is proposed, then the implementation results such as hardware resources usage and timing analysis is presented and discussed. Finally the verification results are introduced in section 4.

4.1   Design methodology:

To design and implement MIMO-OFDM transmitting and receiving systems on FPGA, the process shown in Figure 4-1 starts by identifying the characteristics of the transmitted signal, the status of the transmission channel and the received signal model taking into account the anticipated signal impairments. This stage ends by developing a high-level simulation model for the system on Matlab [61]. After completing the Matlab model successfully, the complete Matlab design is translated to a hardware design using VHDL code and IP cores, using a RTL design approach. Then Maltlab/VHDL co-simulation is carried out to verify that the hardware design is performing exactly the required function. This step is associated with performing several advanced design optimization techniques including design for timing performance, pipeline techniques and designing for area optimizations and resource sharing.

After the RTL design and optimizations, the post place & route simulations are carried out to make sure that the optimized design is meeting the design constraints. Finally the onboard verification is required to ensure that the hardware design is performing the expected function and producing the expected results.

4.1   Implementation platform

In this project Xilinx Virtex 5 FPGA was chosen for design prototyping as it provides a suitable amount of resources that can be used for implementing various designs including Fixed-point and Floating-point arithmetic operations, pipelined architectures are also possible due to the availability of Block RAMs that can be utilized for storing the data during a pipeline processing also it can operate under a frequency up to 400 MHz which is above the target operating frequency that has been set for this project as 100 MHz.
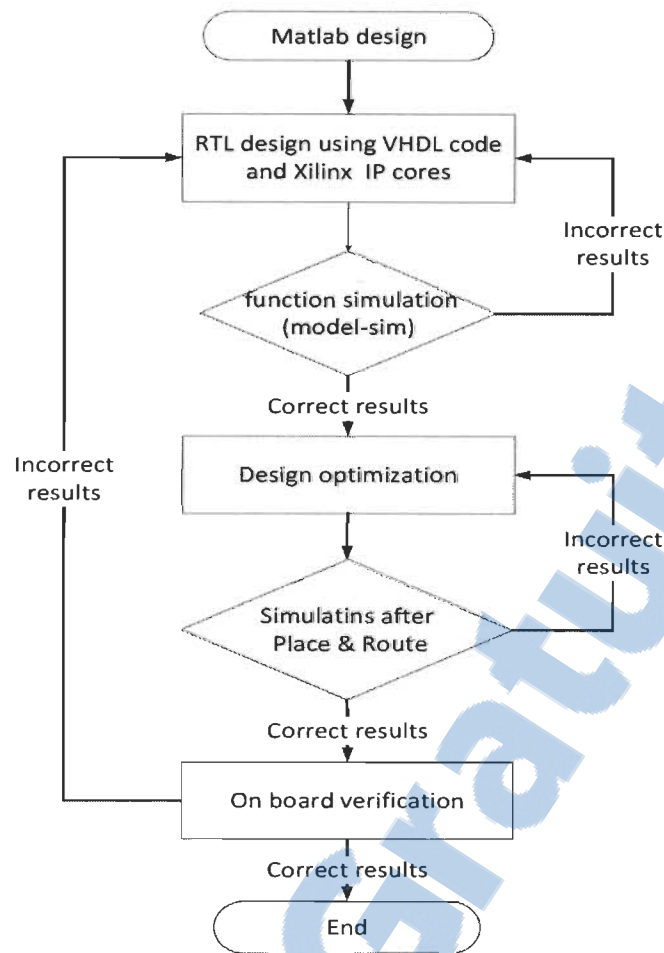
Figure 4-1 Design methodology flow chart

It contains 480 KB distributed RAM, 48 DSP84E slice, and 2160 KB block RAM divided as 120 of 18 KB RAM or 60 of 36 KB RAM. Genesys FPGA development board shown in Figure 4-2 provided by DIGILENT is used in this project and it contains one chip VIRTEX 5 XC5VLX50T.

In order to perform co-simulation with Matlab code, the on board design datapath needs to communicate with sources and sinks of the user data. Matlab is used to set the simulation parameters such as modulation scheme, algorithm choice, input data, channel condition, and so forth; it is also used to start and stop the simulation. Most of the reported design in the literature

[62][63] use Simulink as hardware design platform, because it provides the designer with a library of components which have a hardware equivalent.
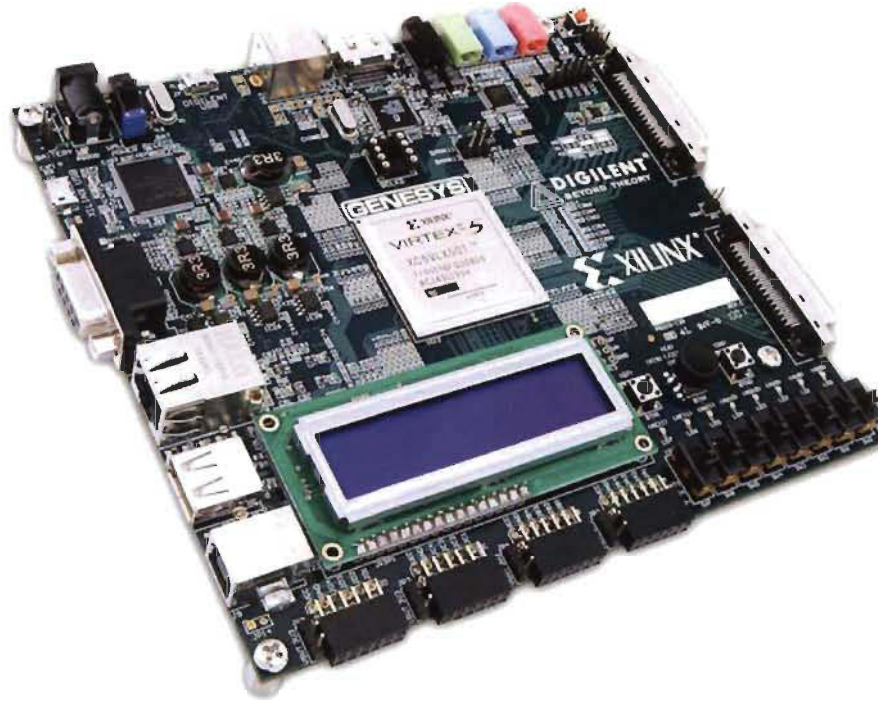
Figure 4-2 Genesys board

However, Simulink is suitable for systems which are not too complex because it is preferred for DSP calculations, not for systems with sophisticated control. MIMO-OFDM system have complex control signals and I'ts not easy to describe it in Simulink. Furthermore, the timing parameters need to be added to the design during the RTL development; clock is supported by Simulink by using "$z-1$" delay blocks. However, in order to model the right number of clock cycle delays the process is considered a time consuming and error-prone task. Hence, Simulink is not suitable platform to develop cycle-true behavior for high complexity systems such as MIMO-OFDM base station.

For the above reasons, a direct, manual conversion from Matlab code to RTL-VHDL is used and the communication between Matlab and FPGA has to be managed directly through the Universal Asynchronous Receive and Transmit (UART). UART allows full-duplex communication using serial link, and it is widely used in the data communications and control system. Building the UART function using separate interface chip is a waste of hardware resources; hence it's better to integrate the UART function inside the same FPGA [64]. In this thesis, UART core functions are implemented using VHDL and integrated into the MIMO-OFDM FPGA chip to provide compact, stable and reliable data transmission, which effectively represent a complete hardware design platform for MIMO-OFDM system.

4.1.1 UART algorithm

UART module consists of transmitter and receiver modules. The transmitter is built as a shift register that accept parallel data and then produce it serially at a specific rate. The receiver, on the other hand, shifts in data bit by bit and then produces it in parallel at the output. Figure 4-3 shows that the transmitter serial output is' 1' during the idle status. Then a start bit of '0' is used to indicate the beginning of transmission, then 6, 7, or 8 data bits are sent, followed by an optional parity bit. Finally it sends '1' bit to indicate the stop of data transmission. The parity bit is set to '0' when the data bits have an odd number of 1's, if odd parity is used. In case of even parity, it is set to '0' if the data bits have an even number of 1's.

Figure 4-3 UART serial bit structure

Figure 4-3 shows a UART transmission system that uses 8 data bits with no parity bit and 1 stop bit. In this system it could be noted that the data least significant bit is transmitted first. Since clock information is not included through the serial line, the transmitter and receiver must agree on the communication parameters before transmission starts. These parameters include the baud rate (i.e., number of bits per second), the number of data bits and stop bits, and use of the parity bit. The design of the receiving and transmitting subsystems is described in the following sections. The design is customized for a UART with a 19,200 baud rate, 8 data bits, 1 stop bit, and no parity bit.

UART receiver architecture

The architecture of the UART receiving subsystem consists of receiver, baud rate generator, and interface modules as shown in Figure 4-4. Due to the fact that the clock information is not included in the transmitted signal, the receiver uses the preconfigured parameters in order to retrieve the data bits. A baud rate oversampling is used to estimate the middle points of transmitted bits and then retrieve them at these points accordingly.
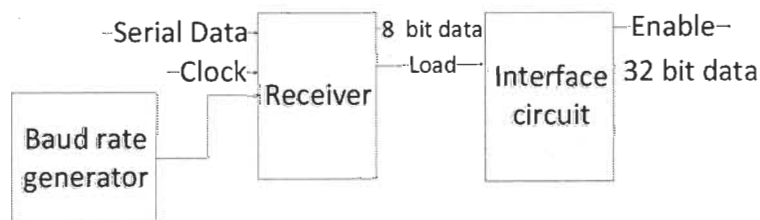
Figure 4-4 Block diagram of a UART receiving subsystem

The baud rate generator module generates an oversampling signal with frequency equal to16 times the configured baud rate. This signal is employed as enable ticks to the UART receiver in order to avoid creating a new clock domain and violating the synchronous design principle.

Assume that the communication uses N data bits and M stop bits. The oversampling scheme works as follows:

1. Wait until the receiving of the start bit then start the sampling counter.

2. After the counter reaches 7, this indicates the middle point of the start bit. Clear the counter to 0 and restart.

3. When the counter reaches 15, this indicates the middle point of the first data bit. Retrieve its value and shift it into receiving register, then restart the counter.

4. Repeat step 3 N-1 more times to retrieve the remaining data bits.

5. If the optional parity bit is used, repeat step 3 one time to obtain the parity bit.

6. Repeat step 3 *M* more times to obtain the stop bits.

The receiver block consists of a finite state machine that has 4 states as shown in Figure 4-5, at state S0 the receiver waits until the data pin equals to zero. This indicates the start of transmission then it goes to state S1. At this state the receiver counts from 0 to 7 to be sure that the sampling will be exactly at the middle of the received bit as discussed earlier , then it goes to state S2 where it stores the incoming bit in a shift register each time the counter reaches 15. After storing the 8 data bits which is indicated by the bit counter the state machine goes to state S3 where it transfers the 8 bit data to the output port and a load signal is activated to indicate the presence of new data at the output port.
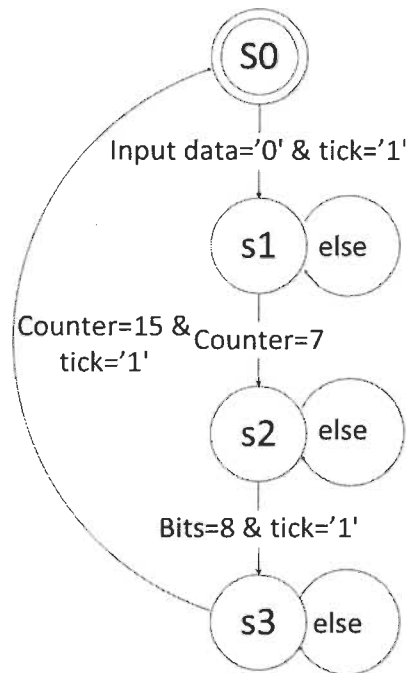
Figure 4-5 UART receiver FSM

The receiver interface module shown in Figure 4-6 consists of a FSM and 4 RAMs each of 32 width and 320 depth (256 data + 64 cyclic prefix). As the data received from UART is arranged in 8 bits a FSM is required for arranging the incoming data into 32 bit words to be stored in each location of the RAM. Therefore the finite state machine waits until a load signal from the receiver is activated then it stores the incoming 8-bit data at the input port in a 32bit shift register. This process is repeated 4 times until the 32 bit shift register is full then it asserts a write enable signal to the corresponding RAM to store the 32 bits data and increments the address for the next iteration. If the address reaches 320 the FSM switch to the next RAM, until all RAMs are full after storing all data the FSM goes to the reading state where it starts reading from all RAMs at the same time from address 0 to 319. This gives the MIMO receiving sub-

system a stream of data which is exactly the same as the data arriving from the two antennas in case of 2×2 MIMO-OFDM system.
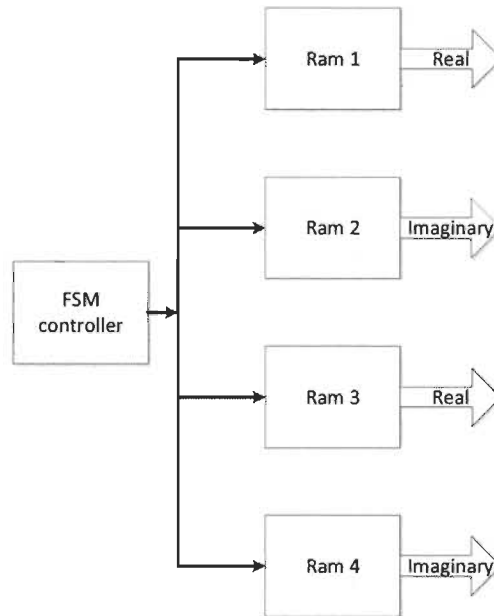
Figure 4-6 Interface circuit block diagram

UART transmitter architecture

Similar to that of the receiving subsystem, the architecture of the UART transmitting subsystem is shown in Figure 4-7. It consists of a transmitter, baud rate generator, and interface circuit. The transmitter is built as a shift register that shifts out data bits at a rate equal to the baud rate. The baud rate generator produces one-clock-cycle enable ticks to control the transmission rate. The frequency of the ticks is 16 times slower than that of the UART receiver. Instead of introducing a new counter, the UART transmitter usually shares the baud rate generator of the UART receiver and uses an internal counter to keep track of the number of enable ticks. A bit is shifted out every 16 enable ticks.
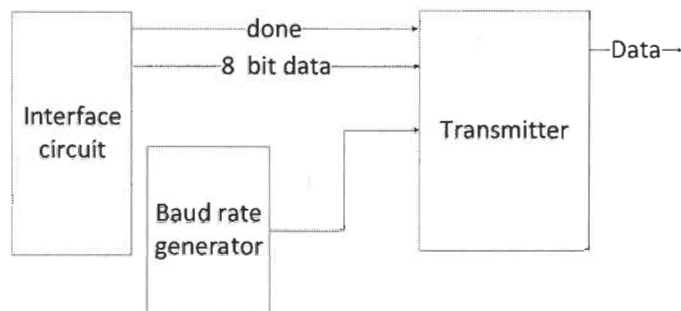
Figure 4-7 Block diagram of a UART transmitter subsystem

The transmitter consists of an FSM with only 2 states, at the initial state if the done signal is received from the interface circuit is activated, the transmitter stores the 8-bit data at its input port in a 9 bit shift register, and it also sets the LSB in the register to 0 to be the start bit. At the same time the data output port is set to 1 to indicate that there is no transmission operation yet.

After this, the state machine switch to the next state where it waits until it receives a signal from the baud rate generator, which means that it has started to shift out the data in the shift register bit by bit to the data port. A counter indicates how many bits are shifted out from the shift register when it reaches 9 (this means that all data has been shifted out) and returns back to the initial state then reset the data port to 1.

The interface module for the transmitter sub-system takes the output of the receiver which is the binary data symbols and arranges them in a shift register that represents the received data from both antennas. Then, it starts sending the data stored in the register to the UART. This is done using a FSM controller that receives both output streams from the MIMO receiver-subsystem and stores them in the shift register. At the initial state it waits until data ready signal is received from the MIMO receiver subsystem. Then, it starts shifting the received data in a shift register 2 bits by 2 bits in case of 2X2 MIMO-OFDM system and a counter indicates the number of shifts. When it reaches 48, then it starts sending 8-bit packets of data to the UART

transmitter, it also set the enable signal to trigger the transmitter to start sending. Then it waits for the done signal from the transmitter that indicates the completion of transmitting the 8-bit data, after that it sends the next 8 bits. This process is repeated until all the data are transmitted.

4.1.2 UART Implementation results

The UART module is implemented and integrated with complete MIMO-OFDM Transceiver . It is used to send and receive real time data in order to verify the function of the implemented system. Table 4-1 shows the consumed resources by the UART sub system from the Virtex 5 FPGA. It could be observed that the UART module consume very small percentage (1%) of the total available resources.

Table 4-1 Consumed Resources for UART Module in Virtex 5 FPGA

Resource	Consumed Number	Percentage of Virtex 5 Resources
320x32-bit single-port RAM	4	1%
4-bit adder	1	
8-bit adder	1	
9-bit adder	3	
12-bit adder	1	
1-bit register	9	1%
12-bit register	1	
32-bit register	1	
4-bit register	1	
1-bit latch	1	1%
8-bit latch	1	

The UART subsystem is tested using VHDL test benches for both transmitter and receiver subsystem. Figure 4-8 shows the receiver subsystem simulation results. A VHDL test bench sends the input data to both antennas for a 2 x 2 MIMO-OFDM subsystem, the data is sent from the Matlab through RS232 serial interface, the simulation results for data 11010110 of antenna 1 at baud rate of 19200 shows that the receiver successfully extracted the 8 bit signal from the received waveform. The transmitter subsystem simulation result is shown in Figure 4-9, the test bench shows the UART transmitter sending data received from antenna1 to be displayed in Matlab. The result shows that the data is transmitted successfully using RS232 protocol.
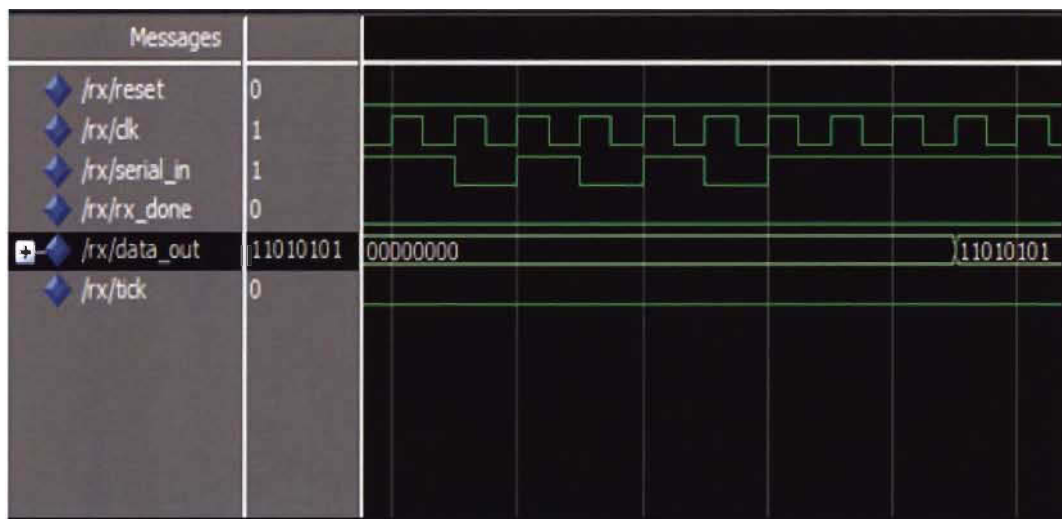
Figure 4-8 UART receiver subsystem simulation results

4.1.3   Matlab interface

For sending and receiving data from PC to the FPGA through serial port a Matlab code is written to open the COM port and perform the write/read operation then close the port after finishing. The main challenge is that the required data to be transmitted to and from the FPGA board is single precision floating-point which is 32-bit data for each word, while the UART

support only 8-bit word, this problem was solved in the FPGA by the interface circuit in the UART sub-system, a similar procedure is used in the Matlab code to enable sending and receiving the data in a correct format.
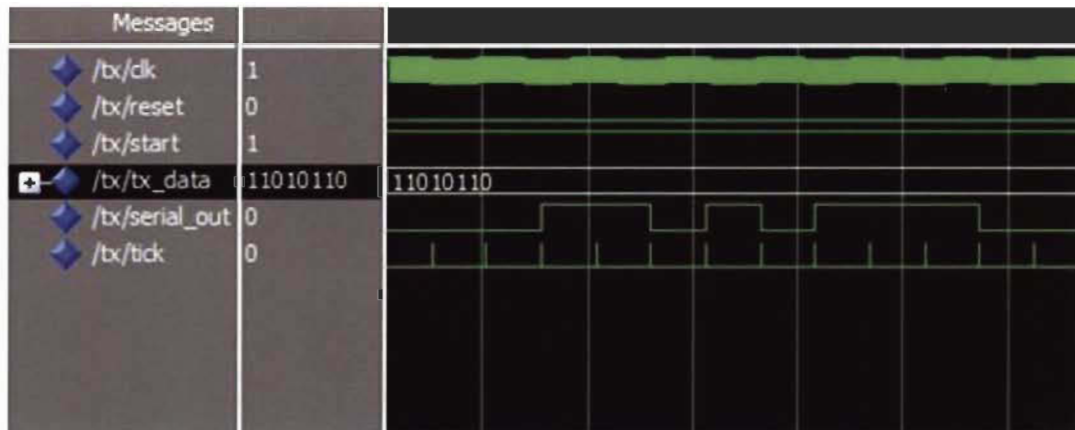
Figure 4-9 UART transmitter subsystem simulation results

If each number is defined as a single precision floating-point and sent directly to the UART , then the data word will not be 32-bit as Matlab automatically reduces the number of bits to the minimum number that is required to represent each number, this will result in an error when the data is sent to the FPGA as the MIMO system implemented on the FPGA is a single-precision floating point system. To solve this problem the number in the Matlab is converted to hexadecimal representation , this step will force the number to be represented in 6 characters each is 8-bit length, this step ensures that each number is sent to the FPGA as 32 bit number in 6 iterations without losing any bits even if the number itself doesn't need 32-bit for representation.

4.2 Design & Implementation of MIMO-OFDM system

The high level Transceiver architecture is shown in Figure 4-10. It supports both Parity bit and Permutation spreading, and the architectures for both schemes are nearly identical except

that the coding memory contents and the despreading mechanism are different. The transmission starts by constellation mapping or modulation, followed by spreading the data symbols either by Parity or Permutation encoding. Next, the spreaded symbols are converted to parallel in order to perform IFFT operation. The IFFT is done using the Xilinx FFT core with different architectures which will be discussed later in this chapter and in the next chapter. Parallel to serial is done after the IFFT transform in order to arrange the OFDM frame for transmission, then a cyclic prefix insertion is done in which the last 64 data samples are added at the beginning of each frame in order to have a total 320 data symbols per frame each of them is represented in 32-bit.

Figure 4-10 MIMO Transceiver block diagram
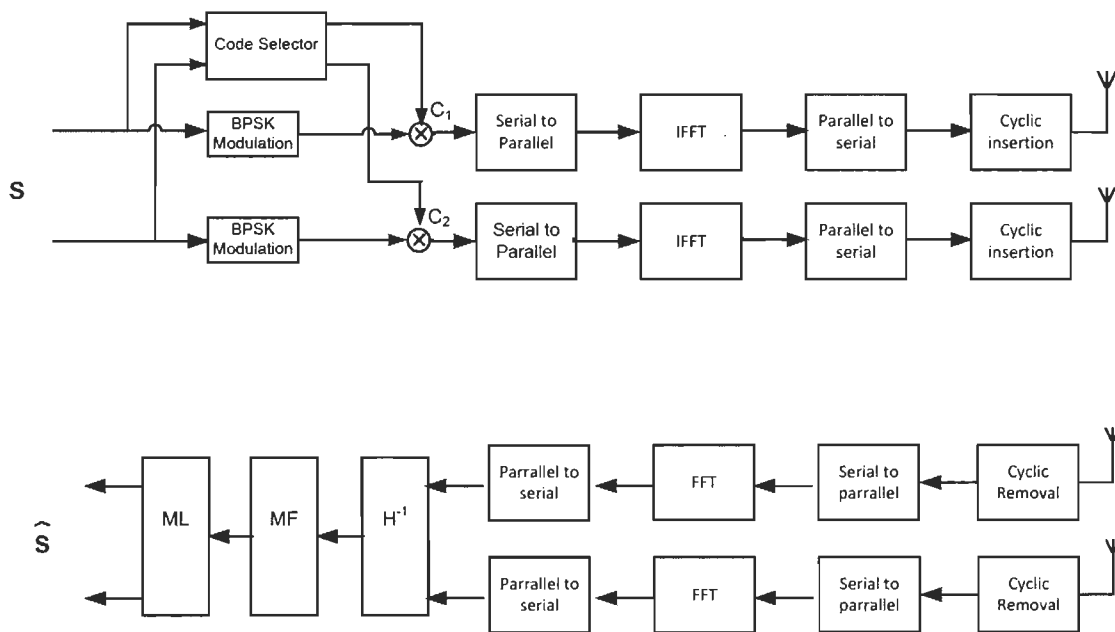
The receiver starts by removing the cyclic prefix, then the received data is converted to parallel in order to be sent to the FFT module. After the data is transformed, it is converted back to serial and sent to the detection module where the channel effect is removed and despreading is carried out either by parity or permutation despreading depending on the used scheme at the

transmitter. The implementation details of each block in the above architecture are given in the following sub-sections.

4.2.1 Spreading code selection:

For single user 2 x 2 MIMO system only two codes are needed for spreading, here two RAMs are used for storing those codes where each code is stored in one RAM. Each RAM has a width of 32 bits and a depth of 8 bits, a counter that counts from 0 to 7 is used to provide the reading address for the two RAMs. The counter clock is 8 times faster than the input data stream clock as each data symbol is encoded by 8 bits code. Therefore, a clock divider is used to divide the input clock by 8.

Parity code:

In the parity code block shown in Figure 4-11 the output of the 2 RAMs is applied as an input to a 2 x 1 Multiplexer which choose 1 of the 2 codes according to the incoming data stream, i.e. for to input samples of 00 or 11 code 1 is selected by sending a select signal to the multiplexer to choose input C1, and for input sample of 01 or 10 code 2 is selected by sending a select signal to the multiplexer to choose input C2.

Permutation code:

The permutation code block is almost similar to the parity spreading; however, in case of parity spreading only 1 code is used for both antennas based on the input data stream as explained above, but in case of permutation different codes are used for each antenna according to the input data. For example, for input 00 or 11 code 1 is selected for antenna 1 and code 2 is selected for antenna 2, and in case of input data of 01 or 10 code 2 is selected for antenna 1 and code 1 is selected for antenna 2.
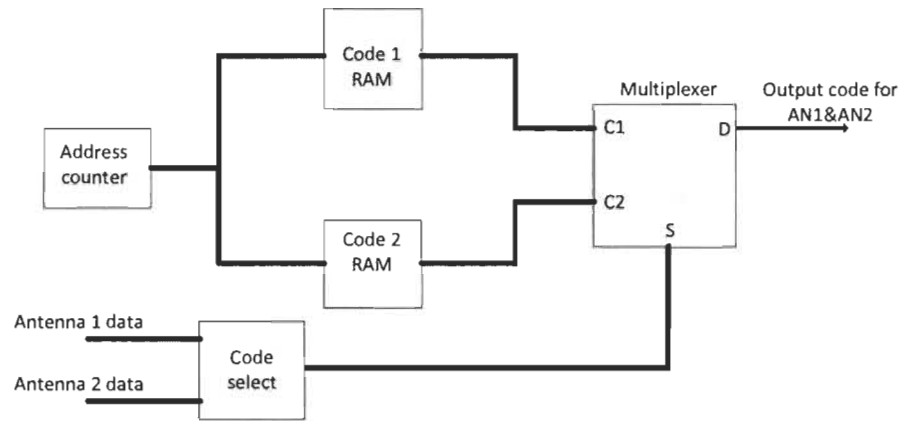
Figure 4-11 Parity code selection block

Figure 4-13 shows the realization of permutation code block in hardware as two 2 x 1 multiplexers which means one multiplexer per antenna, both take code1 and code2 as their 2 inputs and provide their output to the corresponding antenna. Therefore the spreading code selection block is actually a simple set of multiplexers that selects which code to be used, in case of parity one multiplexer is used and in case of permutation two multiplexers are used.

4.2.1   Modulation and data spreading

BPSK is used for modulation, therefore input value of 0 is mapped to 1 and input value of 1 is mapped to -1 this can be translated easily to hardware as a 2 x 1 multiplexer whose 2 inputs are the code and its inverse, to provide the negative of the code only single bit inverter is needed as we use Floating-Point representation shown in Figure 4-12.

S	Exponent e	Unsigned Mantissa m

Figure 4-12 Floaring-Point representation structure
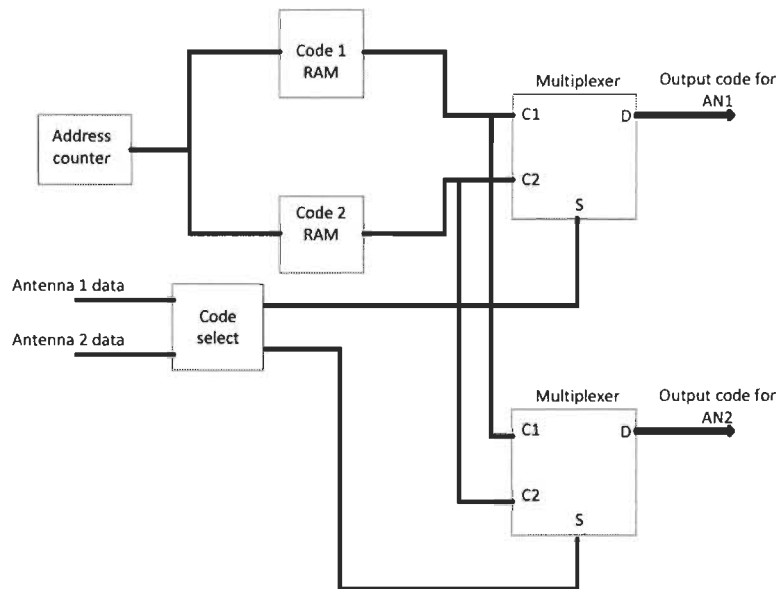
Figure 4-13 Permutation code selection block

Thus inverting the sign bit only gives the negative of the number. The select signal of the multiplexer should be the initial original bit value before spreading for each antenna and to avoid timing violation a D flip-flop is used to provide a one clock cycle delay for each antenna so that it can be used as a select signal to the multiplexer. Figure 4-14 and Figure 4-15 depicts the block diagram for BPSK modulation and spreading for both parity and permutation schemes respectively.

4.2.2   Serial to Parallel circuit:

The serial to parallel block is used to arrange the data stream after spreading in order to construct frame to be applied to the IFFT module. Since pilot signal is not considered in this project it is replaced by zeros, therefore for each 24 input bits per antenna, 192 spreaded data samples are produced, then this frame is padded with 32-sample of zeros at the beginning and 32-sample of zeros at the end in order to give frame length of 256 for the IFFT.
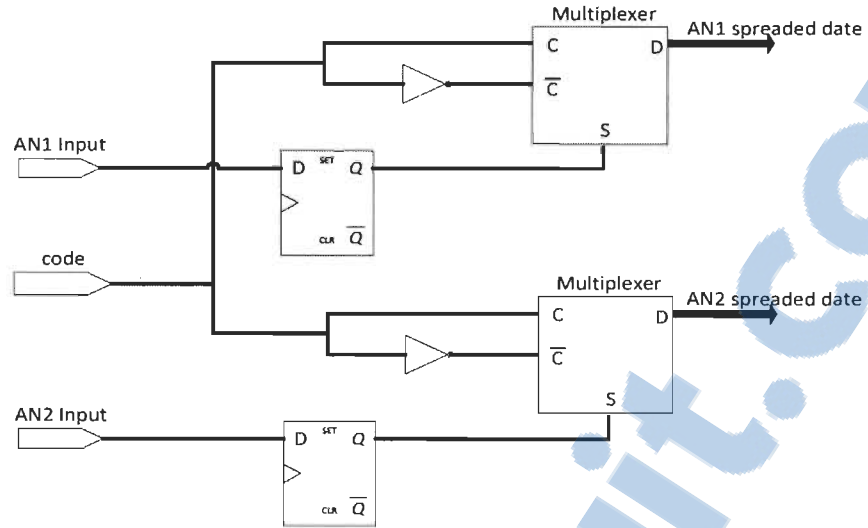
Figure 4-14 BPSK modulation and spreading for parity scheme
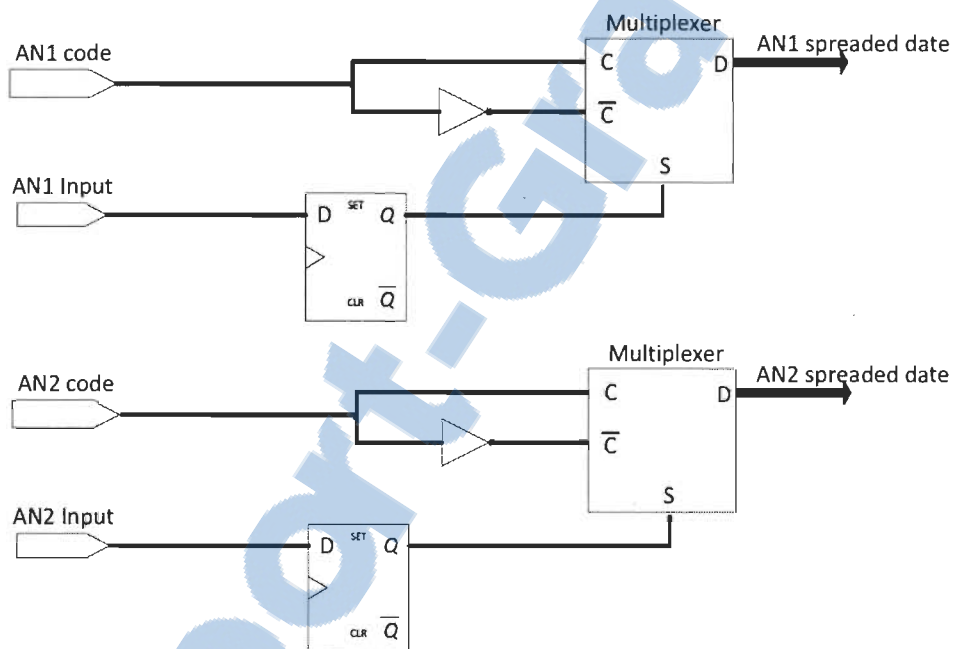
Figure 4-15 BPSK modulation and spreading for permutation scheme

The frame construction is implemented by using a 32-bit width 256 depth RAM and a control unit as shown in Figure 4-16. This block complete the required task in 2 phases (reading phase and writing phase). During the reading phase, the control unit stores the incoming data

stream in the RAM at addresses starting from 33 up to 224 leaving other locations of the memory filled with zeroes, and during the reading phase the control unit reads the data from the RAM starting from address zero to 256 and apply it as an input to the IFFT. This ensures that a 256 data sample is applied to the IFFT module with zero padding at the correct location.
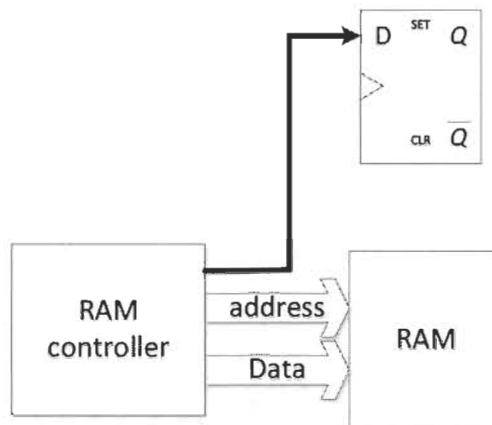
Figure 4-16 Serial to Parallel block

RAM controller:

The RAM control unit is a finite state machine that consists of 3 states, initial state, reading state and writing state as shown in the state diagram in Figure 4-17. During the initial state, the FSM waits until it receives enable which indicates the start of input data. Then it resets the address of the RAM to 32 and moves to the writing phase, during the writing phase the control unit writes the incoming data into RAM addresses 32 to 192. Then, it sends an enable signal to the IFFT module and moves to the reading phase. In the reading phase it reads the data stored in the RAM starting from address 0 to address 255 which are applied the enabled IFFT including the zero padding. After that, it returns back to the initial state.
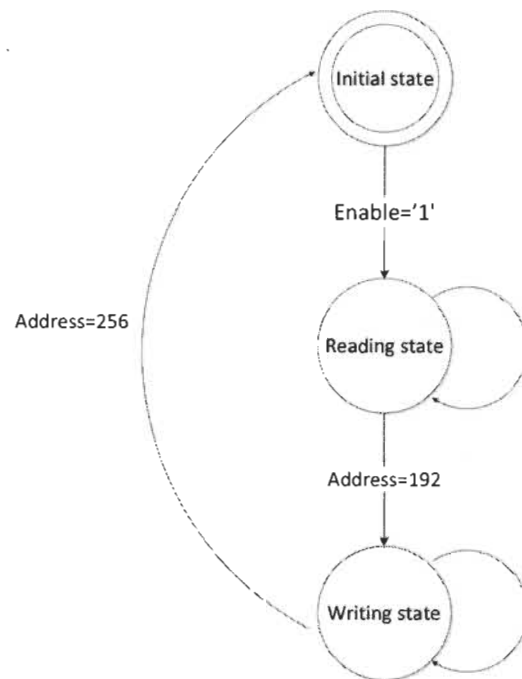
Figure 4-17 RAM control unit FSM

4.2.3 IFFT block

The IFFT is implemented using the Xilinx FFT IP core with frame length of 256 and output data width of 32. The FFT IP core normally computes the FFT of the given input. The IFFT is computed by conjugating the phase factors of the corresponding forward FFT. This will produce the IFFT of the input multiplied by 256. Therefore a divider circuit is required to divide the output of the core module by 256 to get the correct value. However, to avoid the division operation the same function is done by subtracting 8 from the exponent of the floating-point number.

4.2.4 Cyclic Prefix insertion

The cyclic prefix insertion block shown in Figure 4-18 is implemented using two RAMs for storing the real and imaginary values and a control unit. The control unit FSM is shown in

Figure 4-19. It first waits for the enable signal from the IFFT circuit that indicates the presence of transformed data frame and then it moves to state S1. It then starts to store the incoming Real and Imaginary data in their corresponding RAMs from address 0 to 255. After that it goes to state S2 where it starts reading from the RAMs starting from address 192 to address 255, this is done to append the last 64 data at the beginning of the transmitted frame (CP insertion). After that the FSM moves to state S3 where it starts reading from address 0 to address 255, then it goes back to state S0.
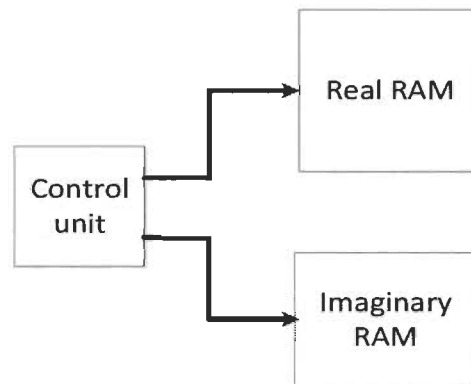
Figure 4-18 Cyclic Prefix insertion block

4.2.5 Cyclic Prefix removal:

The cyclic prefix removal is performed using a simple counter circuit, this circuit starts counting from 0 to 319 when it receives enable signal that indicates the start of data receiving. The counter enables the receiving of data after it reach 64 this mean that the first 64 data samples are ignored. After reaching 319 the counter goes back to zero and it waits the next enable. This circuit depends only on single 9-bit counter and there is no need for using memory and control unit which results in a significant reduction in the overall resources. After cyclic prefix removal, the frame is fed to FFT block similar to the one used at the transmitter to perform FFT operation.
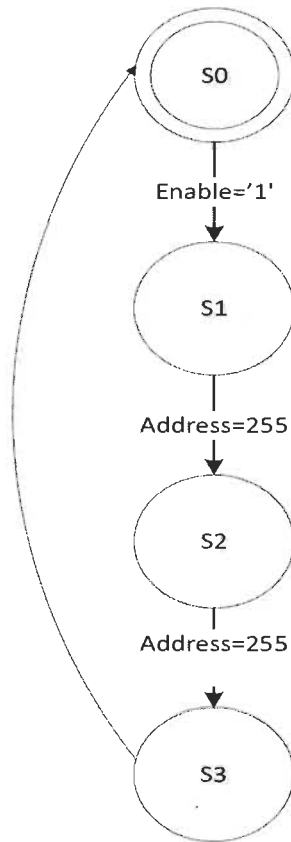
Figure 4-19 Cyclic Prefix FSM

4.2.6 Channel effect removal:

ZF algorithm is used in this design to remove the effect of the transmission channel, the architecture for channel equalization is shown in Figure 4-20. This task is implemented by taking the inverse matrix of the channel effect and performing matrix multiplication between the incoming signal and the inverted channel matrix. Both channel matrix and received data are stored in 4 RAMs each. A control unit is used to store the output data from the FFT block into the RAMs. It then provides the data and the channel effect RAMs with address at the appropriate time. After that, the multiplication result is stored in a RAM in order to be sent to the despreading circuit. There are 2 main challenges in this part, the first one is the matrix multiplication of the complex floating-point numbers, and the second one is the matrix

inversion of the values represented in complex floating point representation. These tasks are explained in the next subsections.
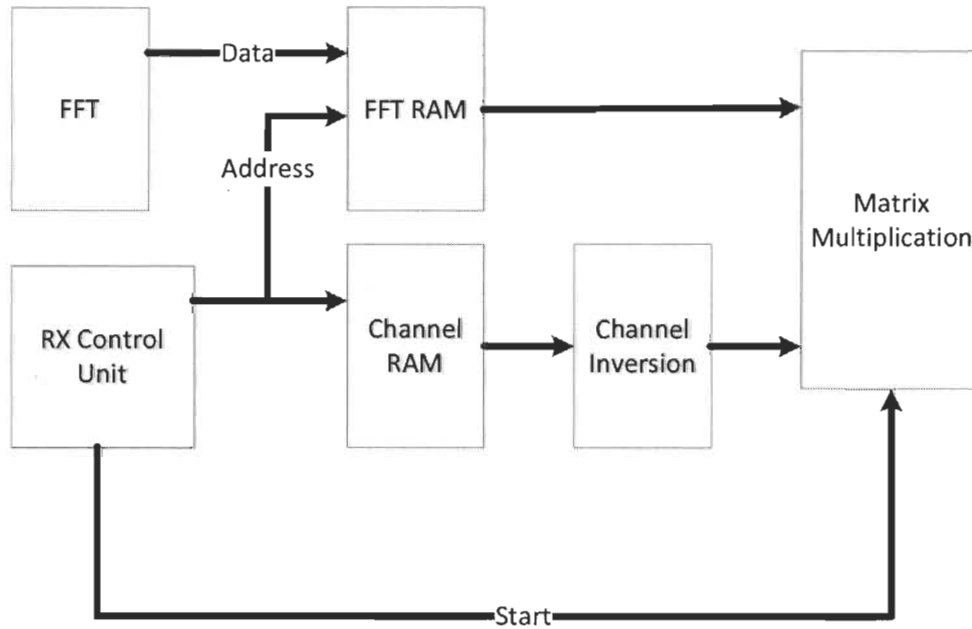
Figure 4-20 Architecture for channel equalization block

4.2.6.1 **Matrix Inversion:**

In order to perform matrix inversion for 2×2 MIMO systems the analytical matrix inversion method is used. However this method is not suitable for larger systems due to the increased complexity. Therefore, Gauss-Jordon elimination method is used for 4X4 MIMO system which will be discussed in details in the next chapter. The analytical method for matrix inversion requires calculating the determinant of the matrix then dividing the matrix with its determinant after swapping element positions, for example for matrix

$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ the inversion is done as follow:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \tag{4.1}$$

Therefore the main part of the inversion block is calculating the determinant and the elements division. For matrix A the reciprocal of the determinant will be equal to

$$\frac{1}{ad-bc} \qquad (4.2)$$

Where a, b, c and d are complex numbers, assume the result of complex multiplication of

$$ad = Xr + jXi \qquad (4.3)$$

And

$$bc = Yr + jYi \qquad (4.4)$$

Therefore the reciprocal of the determinant will be equal to

$$\frac{1}{(Xr-Yr)+j(Xi-Yi)} \qquad (4.5)$$

This can be represented as

$$\frac{(Xr+Yr)-j(Xi+Yi)}{(Xr-Yr)^2+(Xi-Yi)^2} \qquad (4.6)$$

Therefore each real value in the matrix will be divided by the value

$$\frac{(Xr-Yr)}{(Xr-Yr)^2+(Xi-Yi)^2} \qquad (4.7)$$

And each imaginary number will be divided by the value

$$\frac{(Xi-Yi)}{(Xr-Yr)^2+(Xi-Yi)^2} \qquad (4.8)$$

The block diagram of the determinant calculation is shown in Figure 4-21
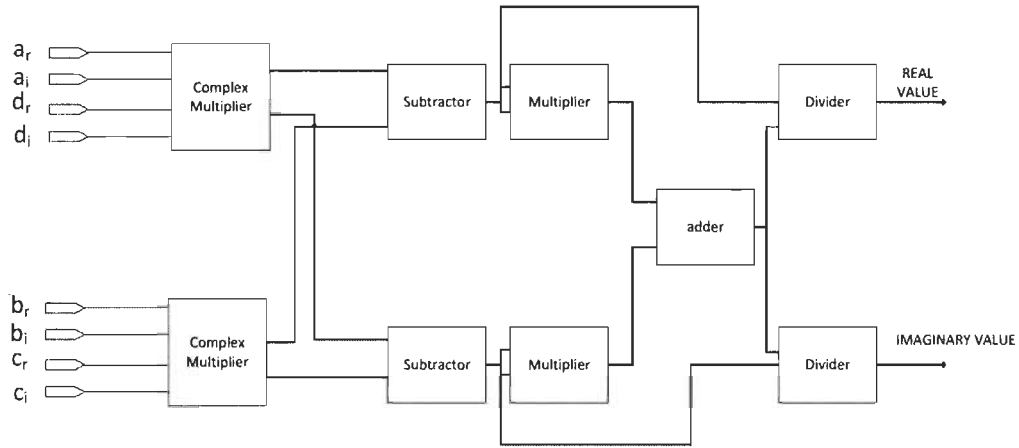
Figure 4-21 Determinant calculation circuit

The reciprocal of the determinant is multiplied by each value of the matrix and the output ports are mapped in such a way to give the result of swapped matrix as shown in Figure 4-22, inverter is used to provide the negative sign for the required elements
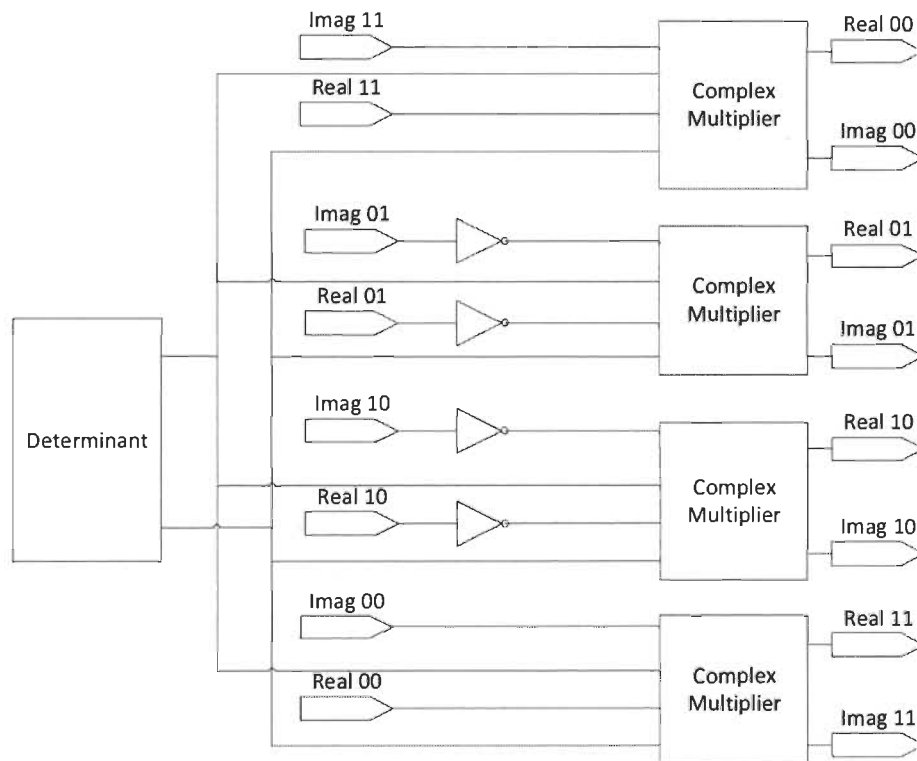
Figure 4-22 Matrix Inversion block

Complex Multiplication:

In order to complete the matrix inversion calculation, complex multiplication must be done for the single-precision floating point complex numbers. Consider two complex numbers $a1 + jb1$ and $a2 + jb2$ the multiplication operation between these 2 numbers is done in hardware by using 4 single-precision floating point multipliers and 2 single-precision adders as shown in Figure 4-23
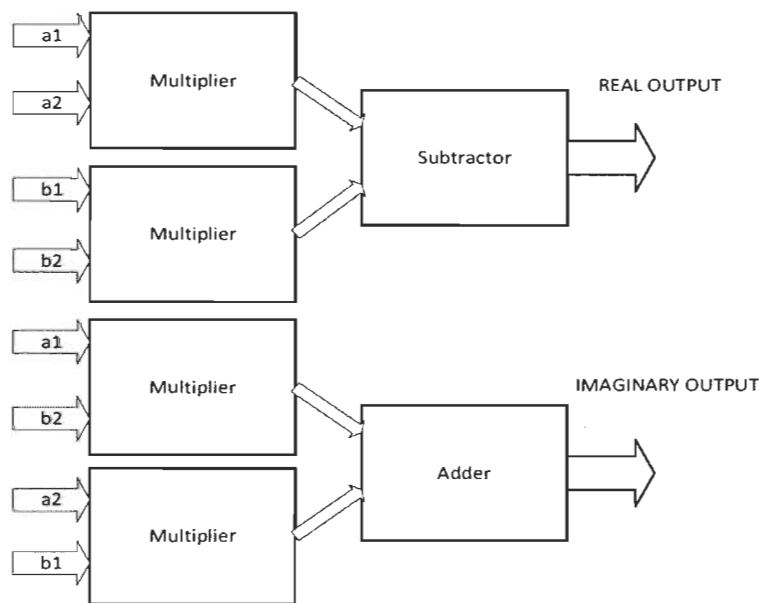
Figure 4-23 Complex multiplication circuit

Matrix Multiplication:

The control unit provides the address for the data RAMS and the channel RAMS. These 2 addresses are provided at the same time and are incremented only after the channel matrix inversion operation is completed in order to perform a new inversion operation. In other words,

4.2.6.2 Channel removal control unit:

The control unit is the most important block in the channel removal process as it is responsible for arranging and removing the zero padding from the incoming data and providing the channel and the data RAMs with the addresses at the required time to perform the matrix multiplication. It also enables the buffering RAMs in order to store the output of the matrix multiplication at the correct time and provides the stored outputs to the despreading. The control unit FSM is shown in Figure 4-25. It consists of 5 states. During the state S0, the FSM waits for the enable signal which indicates the start of the data samples coming from the FFT then it goes to state S1. In state S1, it sends enable signal to the data RAMs and starts storing the incoming data in the data RAMs until it reach address 255. Then, it goes to state S2. In state S2, the control unit starts reading from the data RAMs starting from address 32 as the data from 0 to 31 are zero padded during transmission. During the data reading, the address is incremented each 88 clock cycles. This gives both matrix inversion and matrix multiplication circuits enough time to complete its tasks. Then an enable signal is asserted to the buffering RAM to store the result of the channel inversion. This process is repeated until it reaches address 244 which is the last data value before the zero padding. Then, the FSM goes to state S3 where it starts sending the results to the despreading unit. During this step, the FSM oscillates between state S3, and S4 until it reach address 191 which means that all data are sent to the despreading unit
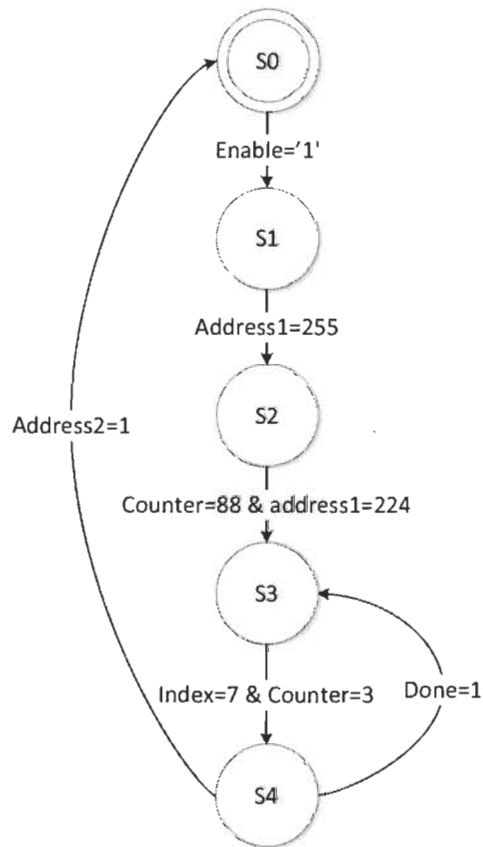
Figure 4-25 Channel removal control unit FSM

4.2.7   Code Despreading:

The despreading is carried out after channel effect removal is completed, either parity or permutation despreading is used depending on the used scheme at the transmitter to recover the original modulated signal, this is done by matching the incoming data with the original codes and calculating the power resulted by each code where the correct code should give the maximum power.

Parity code matching:

The matching circuit consists of 4 matching filters for a certain code each two filters are used for single antenna (real value and imaginary value). The filter is mainly an FIR filter that multiplies and accumulates the input data sequence with a given code as shown in Figure 4-26 and Figure 4-27. Since the used codes are a sequence of 1 and -1 thus no multiplication operations are needed; only inverter is used in place where multiply by -1 is required.
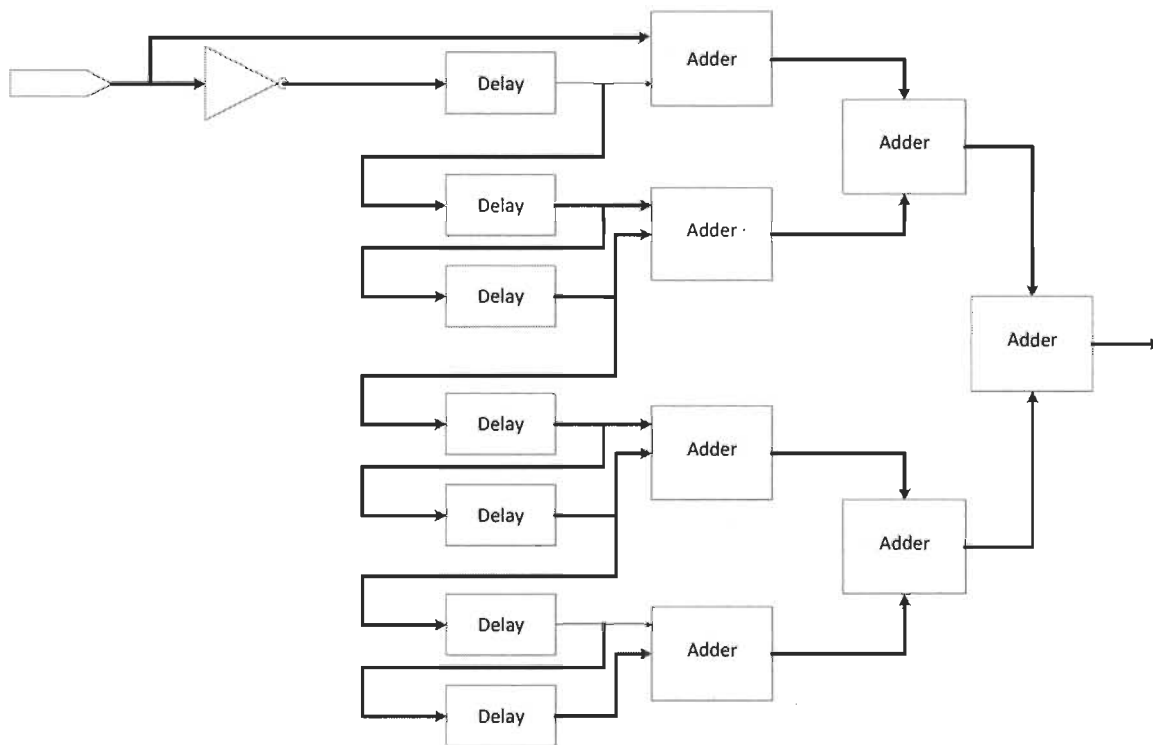
Figure 4-26 Code-zero matching filter

In case of parity code matching, the data from both antennas are applied to code-zero and code-one matching filters at the same time. After this, the absolute value is calculated for the

outputs of code-zero filter for both antenna 1 and antenna 2. Then, these two outputs are added as shown in Figure 4-28.
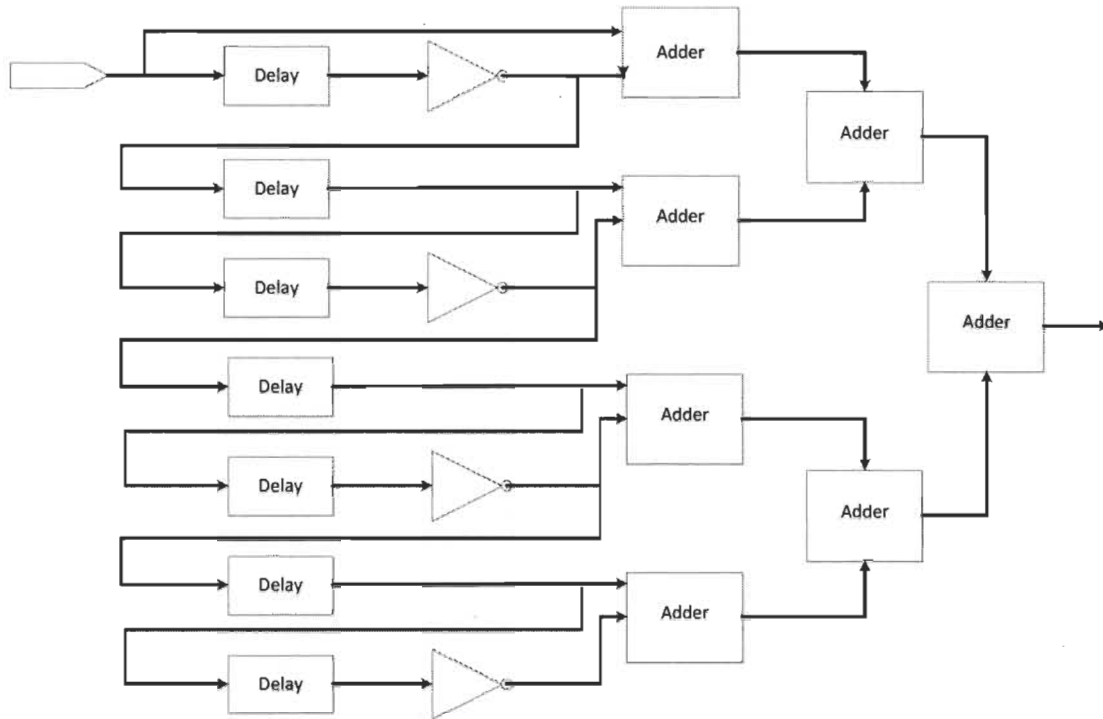
Figure 4-27 Code-one matching filter

Similarly, the absolute value for the outputs of code-one filter for antenna 1 and antenna 2 are calculated then added. The results from both matching circuits are compared as shown in Figure 4-29 and the maximum power indicates the index of the used code. The output of the matched filters is stored in order to be used by the ML block. The control unit used to manage these operations will be described later in this section.

Permutation matching:

The permutation matching circuit is nearly similar to the parity code matching. However, in the formal, one different codes is used for each antenna; i.e. for coset-zero antenna 1 data is matched to code-zero and antenna 2 data is matched to code-one. For coset-one, antenna 1 data

is matched to code-one and antenna 2 data is matched to code-zero. After matching the power, outputs for each coset is calculated and compared using the control unit so that the maximum power corresponds to the correct coset to be identified.
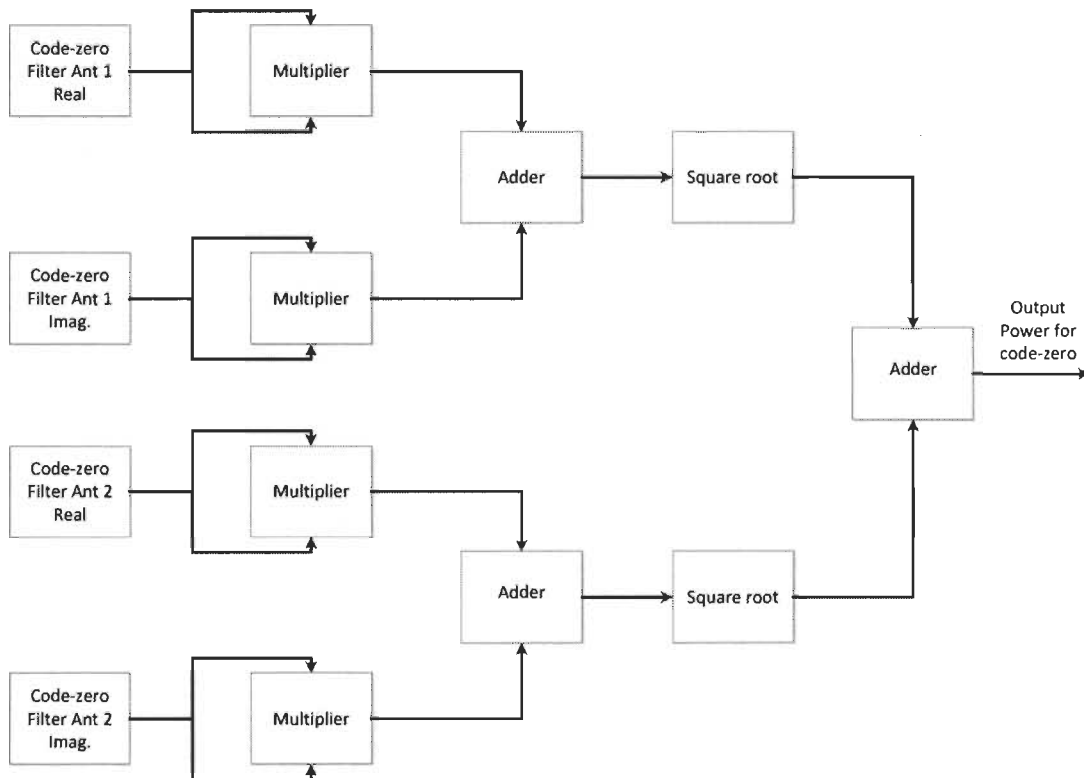
Figure 4-28 Code-zero absolute power calculation block
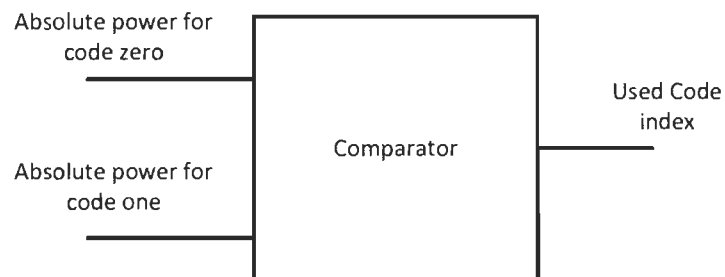
Figure 4-29 Absolute compare

4.2.8   Maximum Likelihood Detection:

ML detection is used to find the actual data that has been transmitted after the matching block discussed in the previous section has found the used code in case of parity spreading or the used coset in case of permutation. Now, each code or coset gives us the possibility of two constellation points that has been transmitted, i.e in case of parity spreading code-zero suggests that either 11 or -1-1 was transmitted by the two antennas while code-one means that either -11 or 1-1 was transmitted. Hence the ML block is used to find the minimum distance between the matched filter output and the two possible points in order to determine the transmitted date.

ML block is realized in hardware as shown in Figure 4-30 and Figure 4-31. Applying constant 1 and constant -1 as two inputs for a multiplexer and the result of the code or coset absolute power comparison is used as the select of this multiplexer.
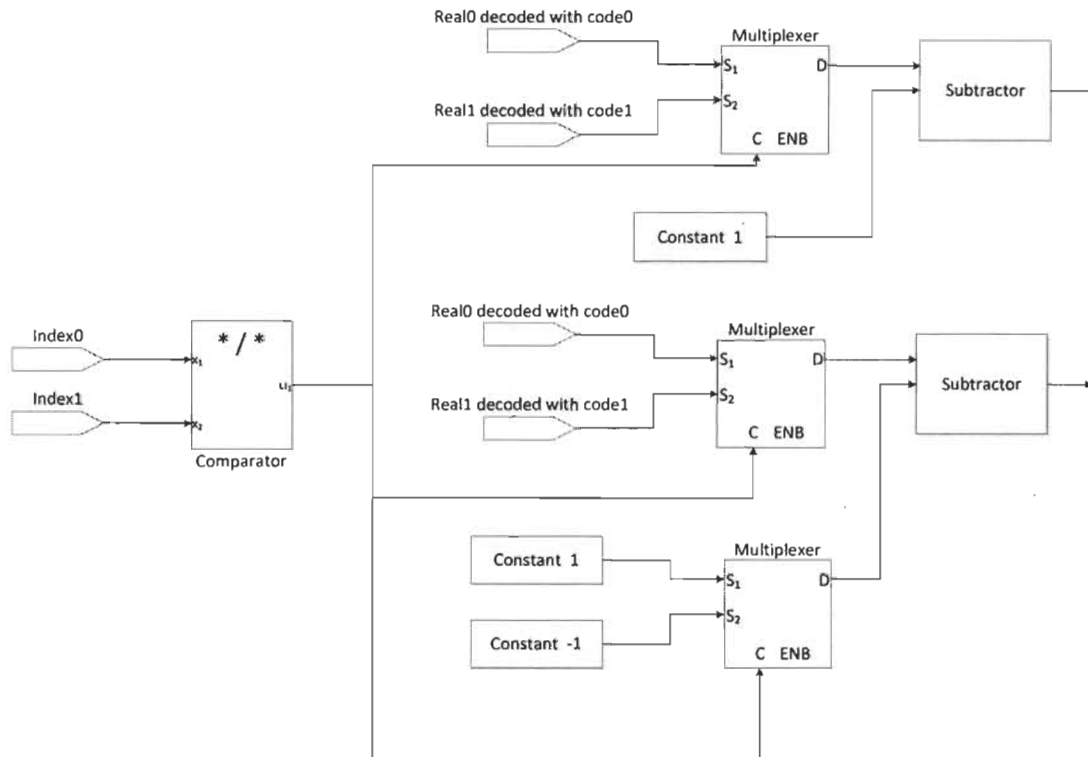
Figure 4-30 ML circuit for 00 and 01 code set

The selected output of the multiplexer is applied as an input to the subtraction unit which subtracts this input from the antenna received value after the matched filter. Another multiplexer is used to switch between the outputs of the matched filters depending on the comparison result. This means, in case of parity coding, if the power of code-zero is greater than the power of code-one then the multiplexer select the real and imaginary values of antenna 1 coded by code-zero and real and imaginary values of antenna 2 coded by code-zero and vice versa. On the other side, in case of permutation coding, if the power of coset-zero is greater than the power of coset-one, then the real and imaginary values of antenna 1 coded by code-zero and the real and imaginary values of antenna 2 coded by code-one are selected and vice versa.
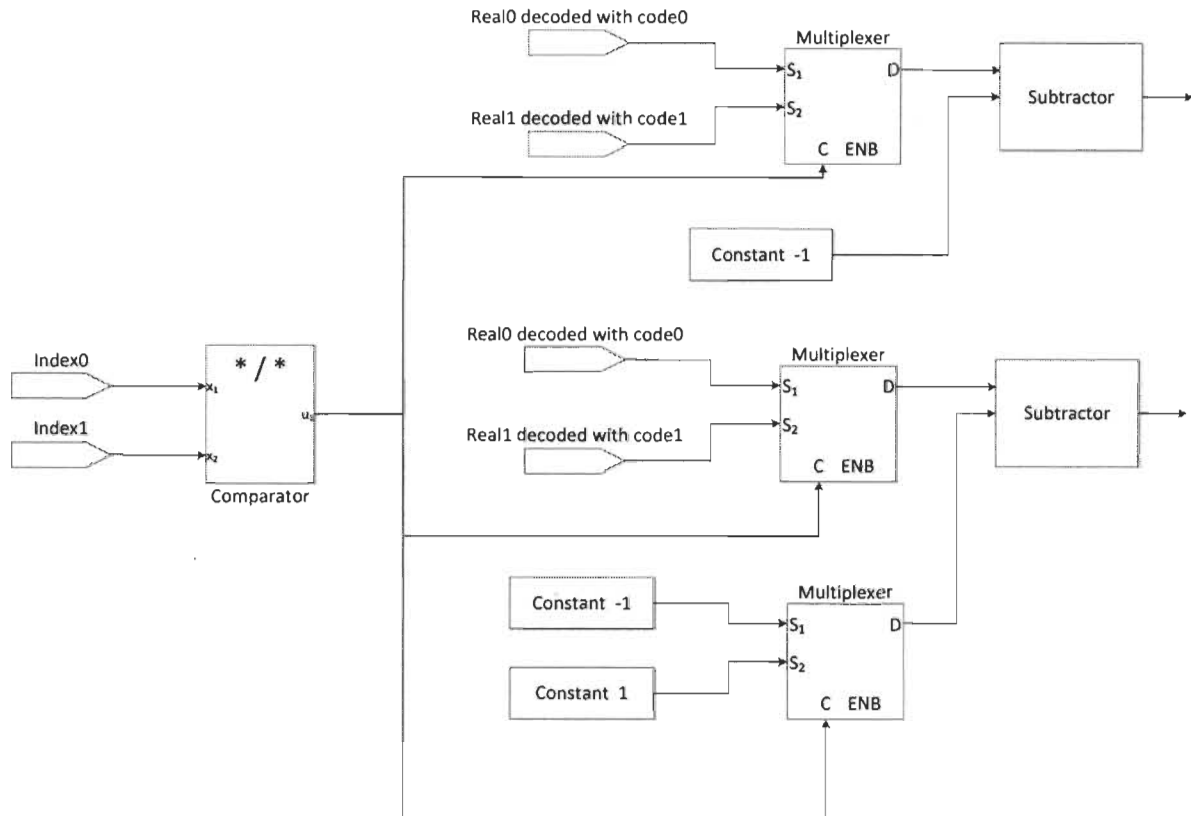
Figure 4-31 ML circuit for 11 and 10 code set

Despreading control unit:

The process of determining the maximum code power and passing code index to the ML unit is managed by despreading control unit. It also passes the outputs of the matching filters to the ML circuit in order to recover the original transmitted signal. The control unit FSM has 8 states as shown in Figure 4-32. During state S0, the FSM waits until the enable signal which comes from the receiver control unit is equal to 1. This means that the data is available at the input of the matching circuit. Then it goes to state S1. In S1, the state machine gives a delay of 42 clock cycles which is the required time for the matching circuit to complete the filter

processing. Then it stores the outputs of the matching filters in the registers for further use. In state S2, the FSM gives a delay of 34 clock cycles which is the required time by the matching circuit or to decide which code or which coset is used. In state S3, the FSM waits for 2 clock cycles until the select signal arrive from the comparator to the ML multiplexers. Then, it provides the multiplexers with data stored in state S1 as an input. In state S4, the FSM decides whether to go to state S5 or S6 according to the comparison result which is registered in state S2. In state S5, it recovers the original signal to 00 or 11 according to the comparison result given from the absolute compare circuit. Finally, in state S6, it recovers the original signal to 01 or 10.

4.1 Function validation

Function simulation was carried out to verify that each module is performing the required function correctly and gives the same results as the Matlab model. The same inputs used in Matlab are used as inputs for the hardware design. The design function was then simulated using Modelsim software. An I/O buses were inserted to monitor the internal signals and to
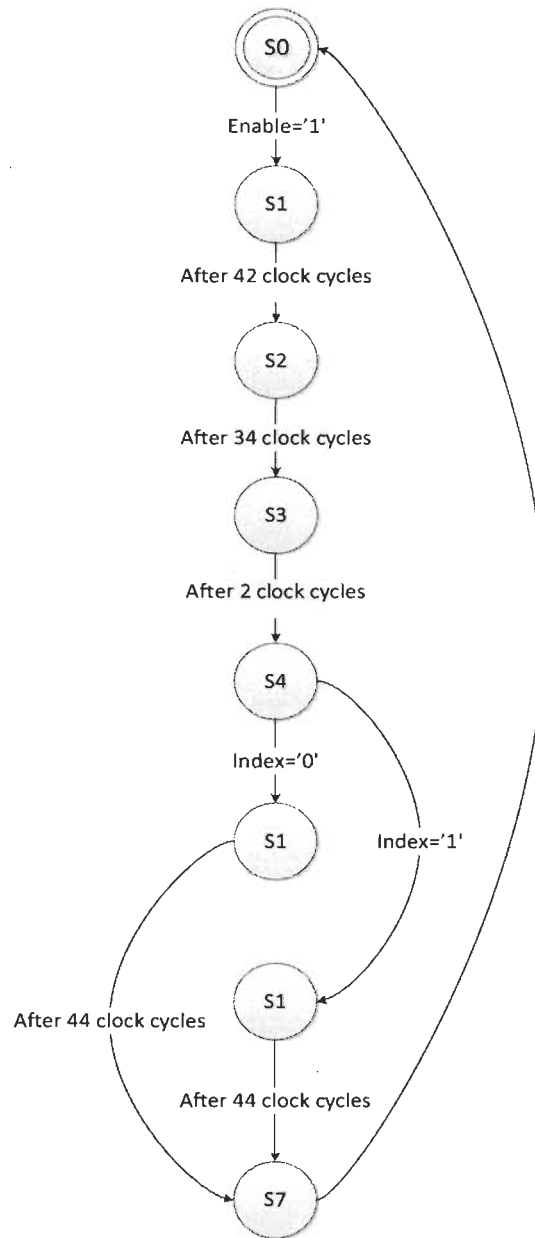
Figure 4-32 Despreading control unit FSM

trace any malfunction in the processing of data inside each module. These I/O busses were removed after verifying that the system is functioning correctly as the presence of excess I/O ports in the design will prevent the generation of the final programming file. The functional simulation results are introduced in Appendix A.

4.2 Synthesis results

After the verification step, the design is synthesized for the target Xilinx XC5VLX50T FPGA chip, and then the hardware resources consumption and the timing analysis reports are extracted. The results of the major blocks in the Transceiver system are presented bellow. First, the hardware resources utilization for the transmitter module shown in Table 4-2 indicates that all DSP48E resources are consumed by this module only. The main reason for such high requirements of hardware resources comes from the fact that separate IFFT IP core module is used for each transmitting antenna. Therefore, the transmitter module need to be optimized in order to reduce the hardware resource utilization as our main target is to put the entire Transceiver on a single chip. The timing analysis report showed in Table 4-3 reveals that the transmitter design support clock speeds up to 404 MHz, which satisfy our target frequency of 100 MHz. The second major module in the Transceiver is the channel equalization at the receiver side. The resource utilization and timing analysis results are shown in Table 4-4 and Table 4-5. It could be noticed that this module is consuming around 50% of both slice register and LUT, which is quite high due to the complexity of the matrix inversion block. Optimized matrix inversion algorithm will be presented in the next chapter in order to reduce the resource utilization of the channel equalization module. Next, the dispreading module results are presented in Table 4-6 and Table 4-7, while the timing constrains are satisfied, the hardware resources requirements are far beyond the capability of Xilinx XC5VLX50T FPGA chip. An intuitive algorithm for despreading and symbol detection will be proposed in the next chapter to reduce the computational complexity of this module. Finally the results for the receiver side and the whole Transceiver are shown in Table 4-8 to Table 4-11. It is clear that hardware resources requirements are extremely higher than the available resources in Xilinx XC5VLX50T FPGA. The main reasons for this result are the high complexity of those modules that has been presented above and the fact that we used Floating-Point representation. In the next chapter, the Fixed-Point representation is also developed in order to reduce the hardware requirements

for the proposed MIMO-OFDM design. It is worth noting that this design could be implemented using

larger FPGA chip such as Virtex 6. Table 4-12 summarizes the synthesis results for the proposed design

in Xilinx XC6VLX195T FPGA.

Table 4-2 Hardware resources consumed by Transmitter in XC5VLX50T

Resource	Used	Available	Utilization
Slice Registers	11627	28800	40%
Slice LUTs	10529	28800	36%
Block RAM/FIFO	9	60	15%
DSP48E	48	48	100%

Table 4-3 Timing summary for the Transmitter in XC5VLX50T

Minimum period	2.475ns
Maximum frequency	404.008MHz
Maxim delay	4.271ns

Table 4-4 Hardware resources consumed by channel removal in XC5VLX50T

Resource	Used	Available	Utilization
Slice Registers	14368	28800	49%
Slice LUTs	14708	28800	51%
Block RAM/FIFO	1383	60	18%
DSP48E	12	48	25%

Table 4-5 Timing summary for the channel removal in XC5VLX50T

Minimum period	3.506ns
Maximum frequency	285.229MHz
Maximum delay	2.775ns

Table 4-6 Hardware resources consumed by despreading module in XC5VLX50T

Resource	Used	Available	Utilization
Slice Registers	26273	28800	91%
Slice LUTs	35312	28800	122%
Block RAM/FIFO	15	60	25%
DSP48E	136	48	283%

Table 4-7 Timing summary for the despreading module in XC5VLX50T

Minimum period	4.332ns
Maximum frequency	230.816MHz
Maxim delay	2.877ns

Table 4-8 Hardware resources consumed by Receiver module in XC5VLX50T

Resource	Used	Available	Utilization
Slice Registers	75913	28800	263%
Slice LUTs	84771	28800	294%
Block RAM/FIFO	15	60	25%
DSP48E	184	48	383%

Table 4-9 Timing summary for the Receiver module in XC5VLX50T

Minimum period	6.047ns
Maximum frequency	165.378MHz
Maxim delay	5.622ns

Table 4-10 Hardware resources consumed by Transceiver in XC5VLX50T

Resource	Used	Available	Utilization
Slice Registers	87540	28800	303%
Slice LUTs	95300	28800	330%
Block RAM/FIFO	24	60	40%
DSP48E	232	48	483%

Table 4-11 Timing summary for the Transceiver in XC5VLX50T

Minimum period	6.047ns
Maximum frequency	165.378MHz
Maxim delay	5.622ns

Table 4-12 Hardware resources consumed by Transceiver in XC6VLX195T

Resource	Used	Available	Utilization
Slice Registers	11625	249600	34%
Slice LUTs	96540	124800	76%
Block RAM/FIFO	24	344	6%
DSP48E	232	640	36%

4.3   Conclusion

A systematic design methodology and real-time platform to translate the algorithms for MIMO-OFDM Matlab model into a real-time wireless prototype is introduced in this chapter. The proposed MIMO-OFDM Transceiver has been divided into smaller blocks and the Floating-Point baseband RTL architecture for each one has been described in details. Matlab/VHDL co-simulation was conducted for each block to verify the functional and behavioral validity of the code-mapping. In addition to behavioral VHDL simulations, syntheses results for the design were also introduced. The results reveal an increase of the resource utilization especially at the receiver side due to high computational complexity. IFFT/FFT, matrix inversion, and dispreading modules are identified as potential blocks that need further optimization in order to be able to fit the design in a single Virtex 5 FPGA chip from Xilinx.

Chapitre 5 - Design optimization

5.1 Introduction

After introducing the FPGA implementation for the proposed MIMO-OFDM Transceiver scheme; in this chapter, we will investigate and propose several optimization options in order to reduce area, power and time. Among those optimization methods that are proposed, a pipelined architecture in which only one IFFT/FFT block is shared among all transmitting/receiving antennas is proposed. Another high computationally challenging module is the despreading unit. An efficient low complexity algorithm for despreading unit based on counters and comparators only is introduced. While the despreading unit based on matched filter consumes a great amount of resources, because it incorporates several arithmetic operations such as multiplication, division and square root calculation; the proposed despreading algorithm greatly reduces the hardware resources requirements because it only uses counters, comparators and basic control logic. Next, an optimized architecture for complex matrix inversion using Gauss-Jordan elimination (GJ-elimination) is proposed. The proposed architecture performs the GJ-elimination for complex matrix element by element. Only critical arithmetic operations are calculated to get the needed values without performing all the arithmetic operations of the GJ-elimination algorithm. The algorithm results in a reduced hardware resources and execution time. Finally, Fixed-Point FPGA architecture is developed, where the maximum allowed performance loss due to quantization is defined, then the tradeoffs between BER performance and area reduction are investigated.

5.2   Pipelined Architecture

The FFT/IFFT processor is one of the kernel modules that have high computational complexity in the physical layer of the MIMO-OFDM system. The optimized FFT library implemented on FPGA is based on pipelined architecture [65]. Thus, if we wish to take full advantage of this library, we have no other choice but to design our system using a pipelined architecture. The basic design introduced in chapter 4 was implemented by using as many IFFT (FFT) kernels as the number of antennas in the transmitter (receiver), same methodology was also reported in [66], [66], and [67]. Figure 4-10 illustrates this idea. In this figure, two FFT blocks are used for two receiver-antennas and two IFFT blocks for two transmitter-antennas. It is assumed that the performance of the FFT block exactly meets the requirement of the MIMO-OFDM data rate. However, a faster FFT operation can lead to a less resource requirement.

The pipeline can save resources by utilizing the concept of logic reuse. This technique, for area optimization, can be used whenever a number of identical processing circuits are used in the design. These circuits can be reduced to a single processing unit shared by all input signal, and a control unit is used to multiplex the input signals to the processing unit. This is exactly the case for the IFFT unit in the transmitter where a single IFFT is shared by both transmission paths, and the two FFT modules in the receiver can also be reduced to a single FFT module shared by both paths. A detailed explanation for the pipelined architecture is given in the next subsections.

5.2.1   IFFT with pipelined architecture:

On the transmitter side the IFFT module is shared by all MIMO-OFDM paths as shown in Figure 5-1, i.e. path 1 send its data to the IFFT module first and path 2 waits for the IFFT to

finish transformation. When the IFFT finish the transformation process for the first path, it sends a done signal to the second path serial to parallel circuit. This circuit contains a FSM which organize the communication between the paths and the IFFT. This FSM has 3 states as shown in Figure 5-2. In state INIT, the FSM wait for enable signal which indicates the presence of coded data ready to be transformed, the FSM sets the start address of the storing RAM to 32 while leaving the first locations in the memory empty for zero padding then it goes to state Dwrite. In this state the FSM stores the incoming coded data until it reach address 224 which is the address of the last data value to be stored leaving the locations from 225 to 255 empty for zero padding. Then, if done, the signal is present. This indicates that the IFFT has finished transforming the data for the other path. Therefore, the FSM goes to state Dread in this state the FSM sends the stored data to the IFFT to be transformed.
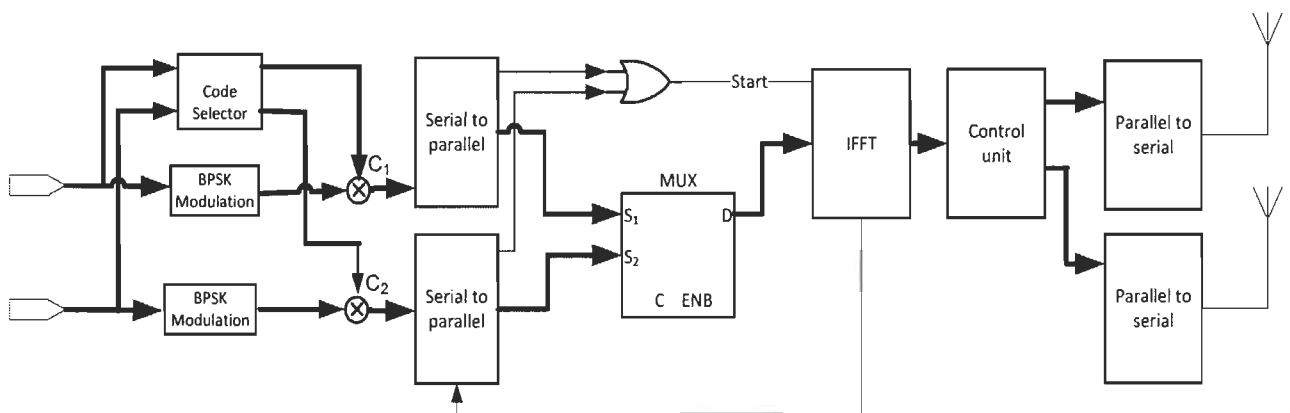
Figure 5-1 Pipelined transmitter for 2X2 MIMO-OFDM
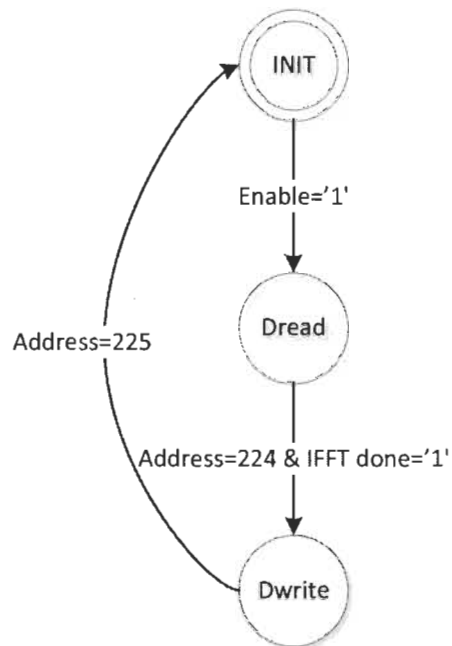
Figure 5-2 Pipelined IFFT FSM

In order to send the transformed data of both paths to the two antennas at the same time, the transformed data from the first path needs to be stored and waits the data from the second path to be transformed. This is done through FSM and 4 RAM blocks as shown in Figure 5-3 and Figure 5-4. This FSM consists of five states, in state S0 it waits for the done signal from the IFFT, which indicates that it finished transforming data from the first path. Then it goes to state S1 where it stores the data in the first path RAM. After that it goes to state S2 where it waits for the second done signal from the IFFT, which indicates the complete transformation of data from the second path. Then it goes to state S3, where it stores the transformed data into the second path RAM. After that, it moves to state S4 where it starts sending the data from both RAMs at the same time.
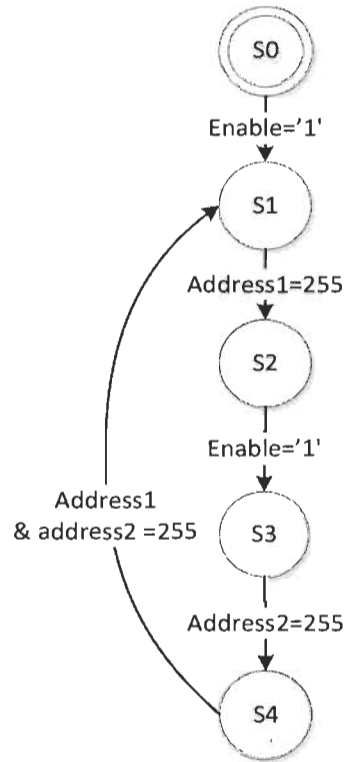
Figure 5-3 Output FSM for pipelined IFFT
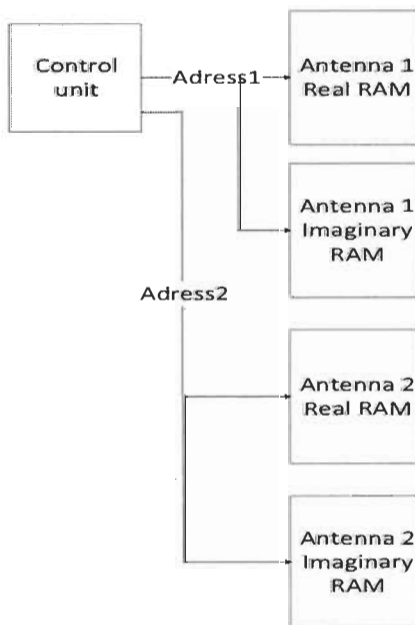
Figure 5-4 Output Ram for the pipelined IFFT

5.2.2   FFT with pipelined architecture:

The pipelined architecture for FFT at the receiver side is shown in Figure 5-5 , two RAMs are used to store the data from both antennas. Then, the control unit sends the data stored in the first path RAM to the FFT module. After the FFT module finishes data transformation, it sends a done signal to the control unit. The control unit starts sending the data from the second path RAM after it receives the done signal from the FFT module. Multiplexers are used to select the real and imaginary values from the first path RAM or the second path RAM according to the select signal coming from the control unit. The control unit is a FSM that consists of 5 states as shown in Figure 5-6. In state S0, it waits until it receives an enable signal to indicate the start of incoming data from the antennas. It sets the initial addresses of the RAMs to 0 and the WE=1. Then it goes to state S1, state S1 increments the address of both RAMs until it reaches 255 which means that the data is completely stored then it goes to state S2. In state S2, the FSM sends the data stored in the first antenna RAM to the FFT. After that, it goes to state S3 where it waits until the done signal from the FFT model is asserted. This indicates that the data from the first antenna is transformed. Then, it goes to state S4. In state S4, the FFT sends the data from the second antenna to the FFT module until it reaches address 255. Then it goes back to state s0.
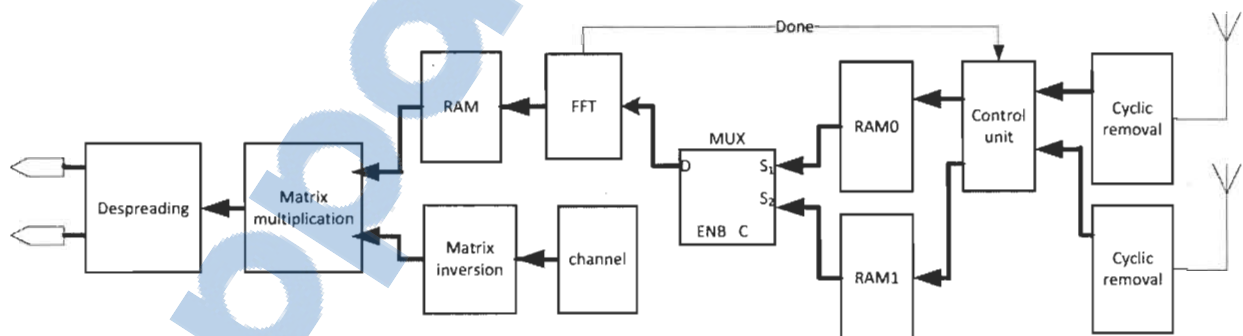
Figure 5-5 Pipelined receiver for 2X2 MIMO-OFDM
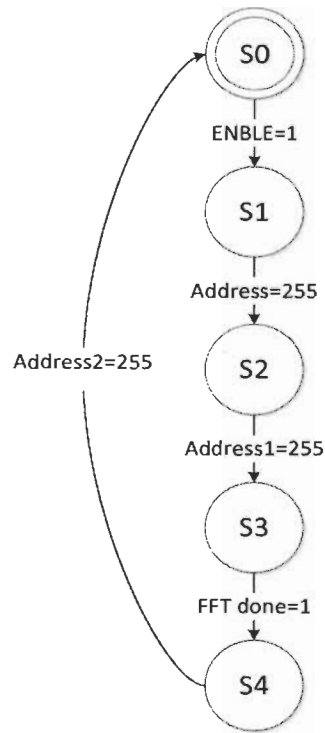
Figure 5-6 Pipelined FFT FSM

5.2.3   Implementation results for pipelined architecture:

Table 5-1 to Table 5-2 shows the synthesis results for pipelined transmitter. The proposed architecture reduces the consumed slice registers by 20%, the slice LUTs by 18%, and the DSP48E by 50%. The block RAM/FIFO exhibits a slight increase by 1% due to the nature of the pipeline. Additionally, the time analysis results reveal a reduction for time requirements.

Table 5-1 Hardware resources consumed by pipelined transmitter

Resource	Non-Pipelined transmitter		Pipelined transmitter	
	Consumed number	Percentage of Virtex resources	Consumed number	Percentage of Virtex 5 resources
Slice Registers	11627	40%	5,830	20%
Slice LUTs	10529	36%	5,132	17%

amount of resources because it incorporates several arithmetic operations such as multiplication, division and square root calculation. The process has been fully explained in Chapter 4. In order to determine the used code for spreading at the transmitter side, the received signal from each antenna is matched to the code matrix at the receiver side. The matching circuit includes 7 adders for each code and each antenna is matched to all codes. Thus, in case of 2×2 MIMO system, 56 adders are needed in addition to 8 multipliers, 4 square root and 2 dividers. After determining the used code, a number of mathematical operations are needed to recover the original signal. These operations are translated in hardware to 8 multipliers, 6 adders, 2 square roots and 1 comparator. In order to reduce this huge amount of resources, a novel detection algorithm is proposed here.

The new despreading algorithm is based on comparator and two up/down counters; counter0 is used to count the matching if the received data is 0, and counter1 is used to count the matching if the received data is 1. The comparator and the two counters are shared among all antennas using pipeline technique. Since BPSK modulation is used at the transmitter side, where 0 is represented with 1 and 1 is represented with -1, the phase of the received data chips is compared against the phase for each local spreading code chips. If they have the same phase, counter0 is incremented and at the same time counter1 is decremented. On the other hand, if they have different phase, counter0 is decremented and at the same time counter1 is incremented. After each code is matched with the incoming data, the counters values are stored in a temporary vector. Then the maximum value in this vector is located. If it belong to counter0, then the received signal represent 0 otherwise it represent 1, the index of the maximum value in the temporary vector will represent the index of the used code. Figure 5-7 shows the flowchart for this detection algorithm.
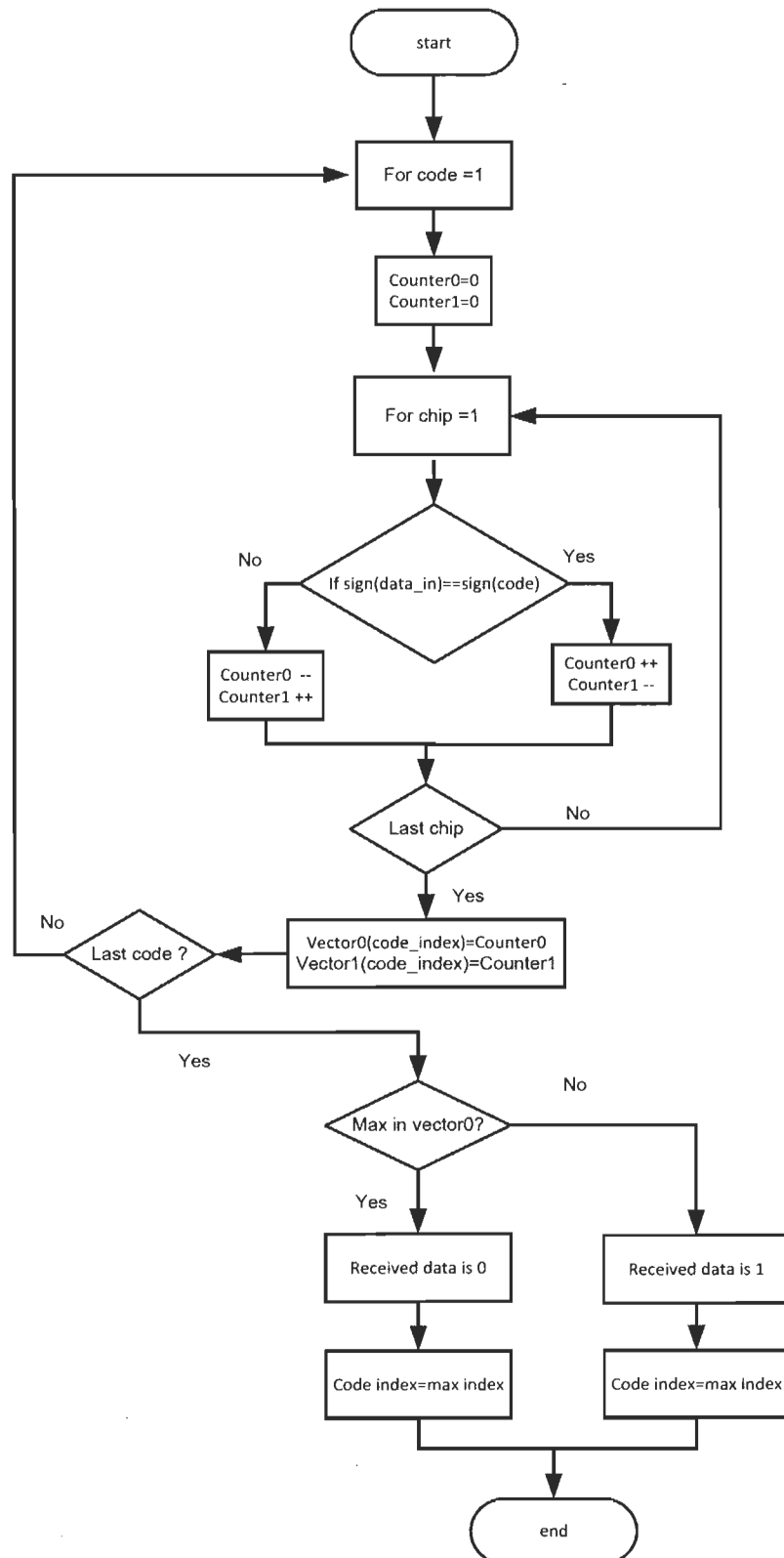
Figure 5-7 Flow chart for the proposed despreading algorithm
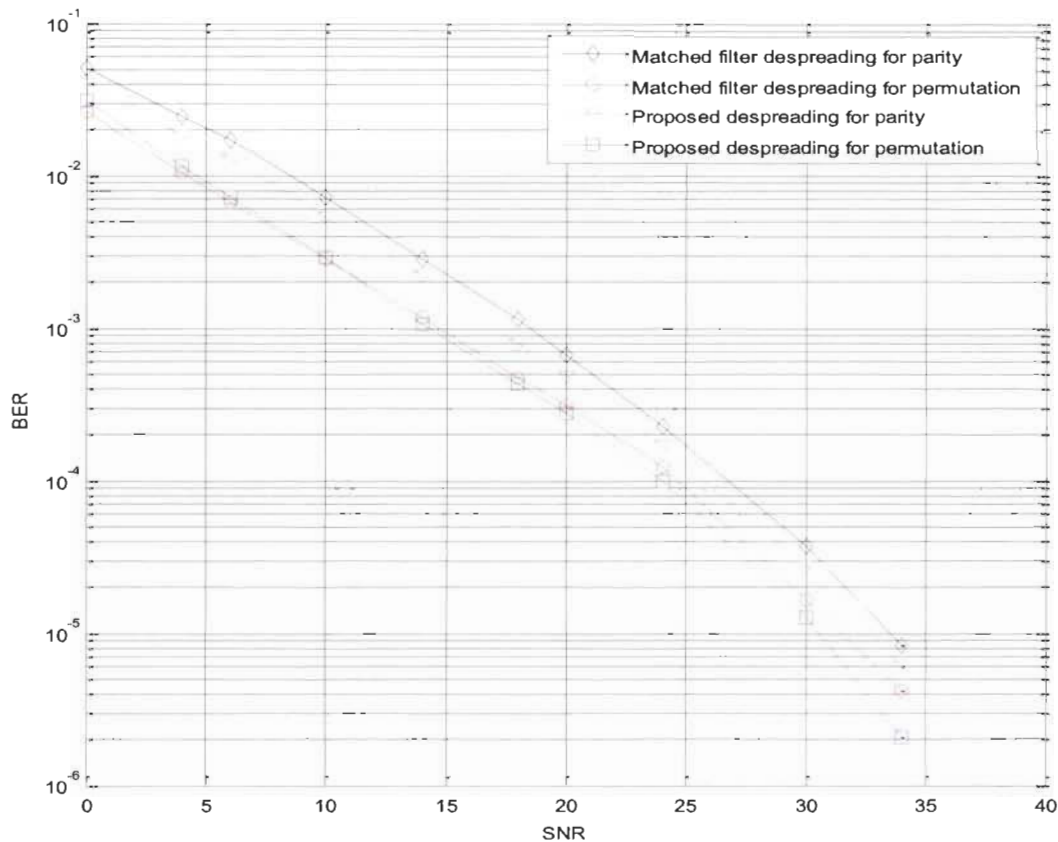
Figure 5-8 Simulation comparison for MF despreading and optimized despreading algorithm

Since the detection is done for each antenna individually, the above algorithm is suitable for any number of antenna configuration and spreading code sequence. It can be used with other modulations such as quadrature phase shift keyed (QPSK) by changing the comparison operation only. It is also suitable for despreading both parity and permutation sequences without any modification. To validate the algorithm, a Matlab simulation is carried out for parity and permutation spreading. The simulation result is shown in Figure 5-8. It can be noticed that the proposed algorithm gives better results than detection with matched filter.

The synthesis results for the proposed despreading algorithm shown in Table 5-3 reveals a significant reduction for the required hardware resources. Unlike the MF despreading where

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

multiplication, division and square root calculations extensively used, the new architecture uses only addition operations.

Table 5-3 Consumed resources for optimized despreading

	Despreading with matched filter		Optimized dispreading	
Resource	Consumed number	Percentage of Virtex 5 resources	Consumed number	Percentage of Virtex 5 resources
Slice Registers	26273	91%	230	0%
Slice LUTs	35312	122%	600	2%
Block RAM/FIFO	15	25%	1	1.6%
DSP48E	136	283%	0	0%

5.4   Matrix inversion optimization:

Matrix inversion is a complex operation that involves several steps with many mathematical operations to be done. Analytically [68] matrix inversion is done by using the adjoint matrix, $adj(A)$, and determinant, $det\ A$ to solve the following equation:

$$A^{-1} = \frac{1}{detA} \times adj(A) \tag{5.1}$$

Adjoint Matrix of a square matrix is the transpose of the matrix formed by the cofactors of elements in the determinant. It is calculated in three steps (a) calculate Minor for each element of the matrix, (b) form Cofactor matrix from the minors calculated and finally, (c) form Adjoint from cofactor matrix. For 4×4 or larger matrices the hardware implementation of the analytic

matrix inversion would be very complex requiring large resources and a lot of iterations to be executed which would reduce the overall performance of the system. As a result, other matrix inversion algorithms based on decomposition methods such as QR, LU and Cholesky have been proposed. However, these algorithms also present a high complexity for its hardware implementation. This fact is due to the use of at least two matrix operations, in addition to the decomposition. For instance, the QRD Gram-Schmidt Ortho-normalization method [70] uses square root, whereas the QRD Givens-Rotations [71] uses sine and cosine operations.

GAUSS-JORDAN elimination (GJ-elimination) algorithm is chosen for this work as it is a direct method that requires three element-based arithmetic operations, namely multiplication, addition/subtraction and division. In this case, no square root operations are used; also no matrix multiplication is required, which in turn significantly reduce the hardware complexity. This has been reported lately in [68] for floating-point real numbers. In this section, more reduction in resources and execution time will be achieved by performing arithmetic operations on critical elements only and ignoring the calculation of the by-product elements that are known to be either 0 or 1 after computation.

5.4.1 GAUSS-JORDAN algorithm

GJ-elimination calculates the inverse of a square matrix. By augmenting the matrix with the identity matrix of the same dimensions. Let $\mathbf{A}$ be $n \times n$ matrix, $\mathbf{I}$ the $n \times n$ identity matrix and $\mathbf{X}$ an $n \times n$ matrix of unknowns. The solution of the linear system $\mathbf{AX} = \mathbf{I}$ gives as result $\mathbf{X} = \mathbf{A}^{-1}$. This is done in a number of iterations which is equal to the dimension n of the square matrix e.g. for 4×4 matrix it will need 4 iterations. In each iteration i the operations done on the augmented matrix $[\mathbf{A}\ \mathbf{I}]$ are:

Check the values of the pivot elements to make sure that they are nonzero elements and in case of a zero pivot element swap the row with zero in the pivot position with a row having a nonzero value in same position.

For iteration i divide row i of the augmented matrix $[\mathbf{A}\ \mathbf{I}]$ by the pivot element in this row resulting in 1 at the pivot element location.

Eliminate all the elements above and below the pivot element by multiplying row i by each of these elements and subtracting it from their rows respectively until all the elements above and below the pivot element are zeros

Increment i and repeat the above 2 steps until $i = n$

For example consider the $2\ x\ 2$ matrix $\mathbf{A}$, where

$$\mathbf{A} = \begin{bmatrix} 2 & 13 \\ 8 & 9 \end{bmatrix}$$

Matrix A is written on the left and the Identity matrix I on the right as follows. The result is called an augmented matrix.

$$\begin{bmatrix} 2 & 13 & 1 & 0 \\ 8 & 9 & 0 & 1 \end{bmatrix}$$

Divide Row [1] by [2] (to give us a "1" in the desired position) this gives:

$$\begin{bmatrix} 1 & 6.5 & 0.5 & 0 \\ 8 & 9 & 0 & 1 \end{bmatrix}$$

Row [2] − 8 × Row [1] (to give us 0 in the desired position):

This gives us our new Row [2]:

$$\begin{bmatrix} 1 & 6.5 & 0.5 & 0 \\ 0 & -43 & -4 & 1 \end{bmatrix}$$

Divide Row [2] by -43 (to give us a "1" in the desired position):

$$\begin{bmatrix} 1 & 6.5 & 0.5 & 0 \\ 0 & 1 & 0.0934 & -0.02331 \end{bmatrix}$$

Row [1] − 6.5 × Row [2] (to give us 0 in the desired position):

This gives us our new Row [1]:

$$\begin{bmatrix} 1 & 0 & -0.1047 & 0.15120 \\ 0 & 1 & 0.0934 & -0.02331 \end{bmatrix}$$

Finally the Identity matrix is produced on the left. So we can conclude the inverse of the matrix **A** is the right hand portion of the augmented matrix:

$$\mathbf{A}^{-1} = \begin{bmatrix} -0.1047 & 0.15120 \\ 0.0934 & -0.02331 \end{bmatrix}$$

In case of 4 × 4 input complex matrix conventional, GJ-elimination needs to perform 8 complex division operations, 24 complex multiplication operations, and 24 complex add operations. These operations are repeated 4 times to produce the inverse matrix. However, the inverse matrix could be obtained by performing operations on critical elements only. The proposed architecture shown in Figure 5-9 consists of:

- Complex multiplier.

- Complex divider.

- 2 Rams of 32 bit width and 32 depth

- Control unit.

The complex multiplier and divider modules are build up using single-precision floating point multiplier, adder and divider IP cores.
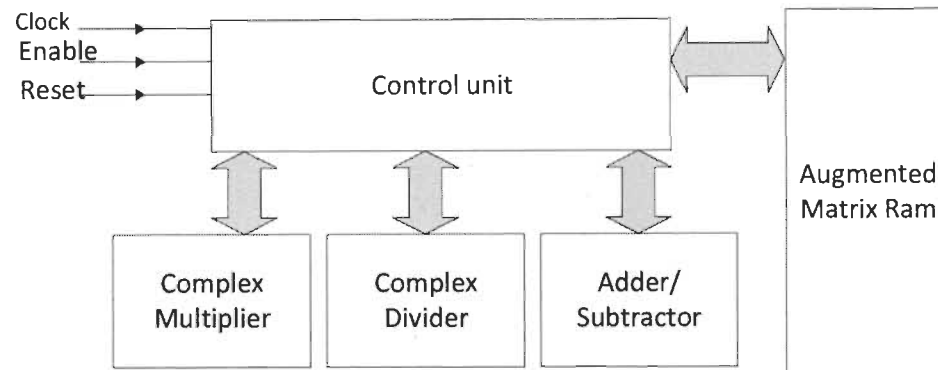
Figure 5-9 Proposed architecture for optimized GJ-elimination

a. Complex Multiplier:

Consider two complex numbers $a_1 + jb_1$ and $a_2 + jb_2$. The multiplication operation between these 2 numbers is done in hardware by using 4 single-precision floating point multipliers and 2 single-precision adders as shown in Figure 5-10.

b. Complex Divider:

The division operation for two complex numbers is done in hardware by using 6 single-precision floating point multipliers, 3 single-precision adders and 2 single-precision Floating-Point dividers as shown in Figure 5-11.

c. RAM:

Two Rams are used to store the real and imaginary values of the input matrix and the identity matrix during the initialization phase and the results during the calculation phase. For 4×4 Matrix the RAM depth is 32 and the width is 32 bit as single precision-floating numbers are used.
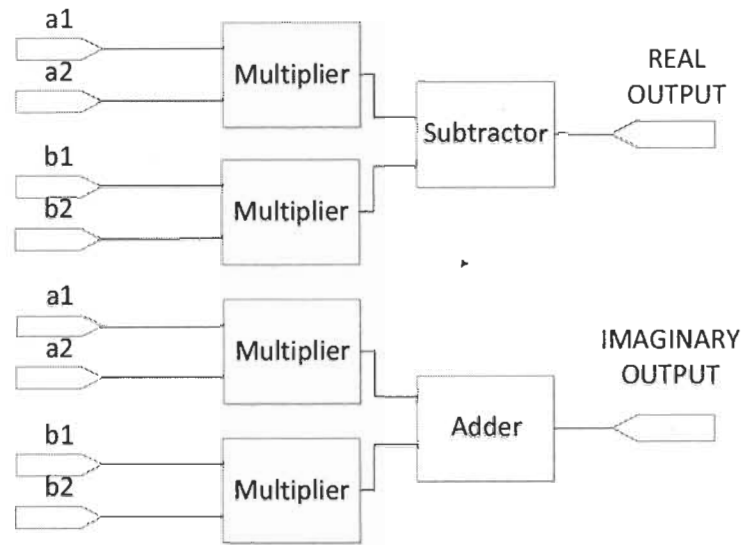
Figure 5-10 Complex multiplier architecture
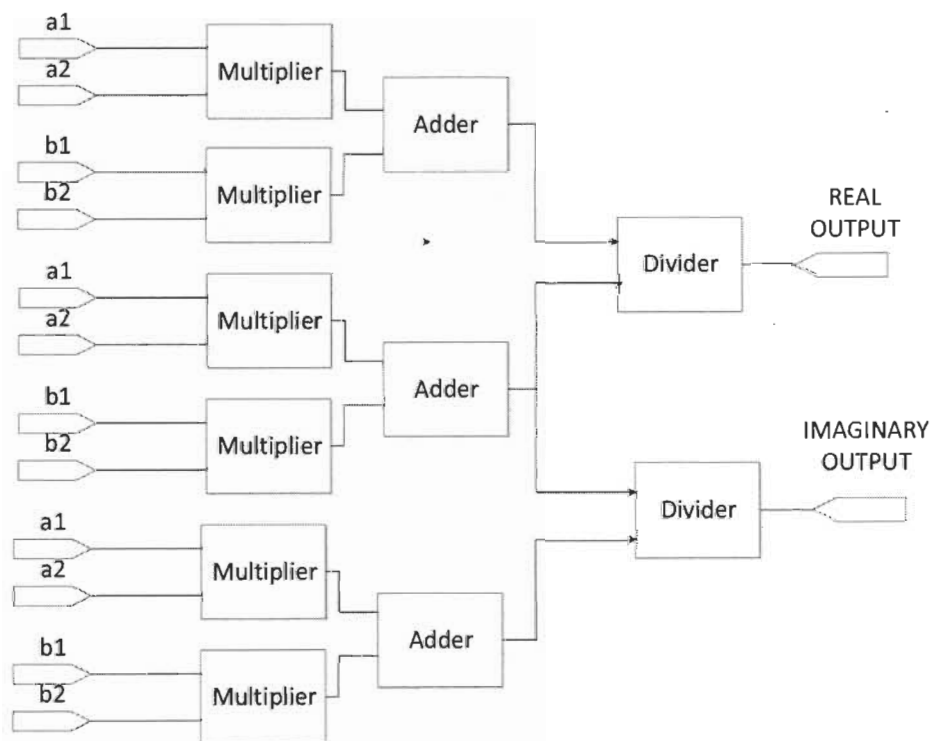
Figure 5-11 Complex Divider architecture

d. Control unit:

This is the main module in the architecture. It consists of a finite state machine (FSM) that points to the address of the critical elements and ignore the addresses of the elements which are known to produce either 1 or 0 results according to GJ-elimination. Therefore, the reduced numbers of arithmetic operations per each iteration are 4 complex division operations, 12 complex multiplication operations, and 12 complex add operations. These operations are exactly half the operations done in conventional GJ-elimination algorithm, for more illustration consider input matrix A, where

$$A = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{5.2}$$

In which a_{ij} is the complex number in row i and column j. During the initialization phase, the control unit stores this input matrix and the identity matrix in the Ram so that the data stored in RAM are

$$[AI] = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} & 1 & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & a_{13} & 0 & 1 & 0 & 0 \\ a_{20} & a_{21} & a_{22} & a_{23} & 0 & 0 & 1 & 0 \\ a_{30} & a_{31} & a_{32} & a_{33} & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

The whole process would fail if there is a zero value in the pivot element location. Therefore, the control unit start by checking the values of the pivot elements to make sure that they are nonzero elements. In case of a zero pivot element, the problem is solved by swapping the row with zero in the pivot position with a row having a nonzero value in same position (This process is compensated at the end by swapping the columns whose numbers are the same as the swapped rows). Then the GJ algorithm starts.

As discussed earlier, the control unit would then perform the arithmetic operations only on the critical elements. Thus the operations for iteration number one are

Division for iteration 1:

$$[AI] = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} & 1 & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & a_{13} & 0 & 1 & 0 & 0 \\ a_{20} & a_{21} & a_{22} & a_{23} & 0 & 0 & 1 & 0 \\ a_{30} & a_{31} & a_{32} & a_{33} & 0 & 0 & 0 & 1 \end{bmatrix}$$ (5.4)

The element a_{00} is the divisor element and the shaded elements are the dividends

Multiplication for iteration 1:

$$[AI] = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} & 1 & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & a_{13} & 0 & 1 & 0 & 0 \\ a_{20} & a_{21} & a_{22} & a_{23} & 0 & 0 & 1 & 0 \\ a_{30} & a_{31} & a_{32} & a_{33} & 0 & 0 & 0 & 1 \end{bmatrix}$$ (5.5)

The element a_{10} is the multiplier and the shaded elements in row 1 are the multiplicands

Addition for iteration 1:

$$[AI] = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} & 1 & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & a_{13} & 0 & 1 & 0 & 0 \\ a_{20} & a_{21} & a_{22} & a_{23} & 0 & 0 & 1 & 0 \\ a_{30} & a_{31} & a_{32} & a_{33} & 0 & 0 & 0 & 1 \end{bmatrix}$$ (5.6)

Row 1 is subtracted from Row 2 to eliminate a_{10}, then the multiplication and addition is repeated to remove a_{20} and a_{30} after that we move to iteration 2.

The control consists of 6 phases and a load state. Each phase consists of 3 states. These phases are: initialization phase, division phase, multiplication phase0, multiplication phase1, multiplication phase2 and multiplication phase 3. In the initialization phase, the FSM stores the input data (input matrix). Then, it sets the zeros of the identity matrix and the ones of the identity matrix. By the end of the initialization phase, the input matrix is stored with the identity

matrix. During the division phase, the FSM stores the pivot element of a given row then divides the main 4 elements of the row with the stored element. It sends the result to each of the 4 multiplication phases where each phase represents a row. It is multiplied by the negative of the pivot element in each row and added to the same row. The whole process is repeated 4 times. The inverse of the 4 × 4 matrix is calculated. Then, finally, the FSM goes to the load state where a done signal is asserted to indicate the completion of the inversion process.

The proposed architecture can be used for any square matrix size. The only change in the architecture will be the RAM size and the time required to finish the operations. A timing constrain of 10 ns was set on the system clock as we used a clock of 100 MHz. The algorithm has been validated within Matlab and the design successfully met the constraint. It has also showed the possibility of operating at higher frequency as the synthesize report shows Clock period: 3.801ns (frequency: 263.116MHz). The suggested architecture performs the 4×4 matrix inversion in 608 clock cycles which is less than the number of clock cycles offered by QR decomposition for 32-bit in [68] and [72].

[68] Shows the same matrix inversion done analytically in nearly more than 700 clock cycles. Finally [72] used QRD for 4x4 matrix inversion and the design needed more than 1000 clock cycles to finish the inversion.

Table 5 shows the required resources for building the architecture for different matrix sizes and the percentage of consumed resources from Virtex 5 LX50T.

Table 5: Consumed resources for proposed matrix inversion architecture

Resource	4×4 Matrix	8×8 Matrix	16×16 Matrix
DSP48E	10	10	10
16× 32 single port RAM	2	4	8
64× 32 dual port RAM	2	4	8
Slice Registers	3048	4321	5430
Slice LUTS	4476	7396	10316
Number of clock cycles	608	1204	2420

The consumed DSP48E's are fixed as the complex arithmetic circuits are the same for all matrix dimensions. The Table also shows nearly half the resources used in reference [73] where it consumes total of 22 DSP48 and 8549 LUTs for 4X4 matrix.

Before introducing the Fixed-Point architecture, the proposed optimization techniques have been applied for the implementation of complete Transceiver system using Floating-Point representation. Table 5-4 shows hardware resources requirements comparison between optimized and non-optimized design. The indicated results clearly show that the proposed optimizations have greatly reduced the resources consumptions and made it possible to port the design on a single Virtex 5 FPGA chip. The timing requirements have also been reduced as shown in Table 5-5, this is due to the elimination of multiplication operations in the despreading unit and the division operations in the matrix inversion unit. Further reduction for resources

requirements will be introduced in the next section by using Fixed-Point representation to implement the Transceiver architecture.

Table 5-4 Consumed resources for Floating-Point non-optimized Vs optimized Transceiver

Resource	Non-optimized		Optimized	
	Consumed number	Percentage of Virtex 5 resources	Consumed number	Percentage of Virtex 5 resources
Slice Registers	87540	303%	26712	92%
Slice LUTs	95300	330%	26800	92%
Block RAM/FIFO	24	40%	24	40%
DSP48E	232	483%	48	100%

Table 5-5: Timing summary for Floating-Point non-optimized Vs optimized Transceiver

	Non-optimized	Optimized
Minimum period	6.047ns	4.84 ns
Maximum frequency	165.378MHz	206.6 MHz
Maxim delay	5.622ns	4.12 ns

5.5　Fixed point architecture:

Hardware implementation for Floating-Point arithmetic has generally been considered much less efficient than fixed-point designs. Most of the MIMO-OFDM algorithm implementations

are done in fixed-point representation due to its smaller size. However, quantizing a floating-point model causes a loss in performance, and the maximum allowed performance loss needs to be defined before starting the quantization of the RTL model. For this reason the floating-point Matlab model needs to be converted into fixed-point model, in order to measure this performance loss and investigate the tradeoffs between BER performance and area reduction.

Initially the signals amplitude is determined by the floating-point Matlab model. This in turn gives an upper bound for the word length of the fixed-point model. The maximum allowed performance loss should be considered as an "error budget." Almost each signal, which is quantized, consumes a part of the budget. A big part of the budget can be distributed over the various top-level signals proportional to the hardware cost of the functions, which are influenced by the quantization.

Figure 5-12 shows the relationship between the fixed-point word length and BER performance. Optimum word length can be selected from the graph. Large word length reduces error. However, it increases hardware costs. On the other hand, short word length decreases costs but it increases the error.
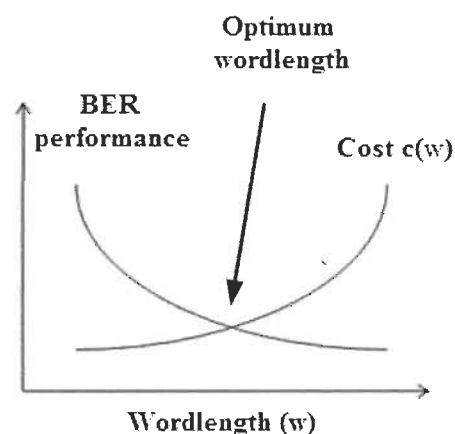
Figure 5-12 Word length VS BER performance for MIMO-OFDM quantization

Generally there are two approaches for word length optimization; the first one is the analytical approach where the error model for feedback systems is quantized. However, it is difficult to develop analytical quantization error model of adaptive or non-linear systems. The second approach and the one used in this project is simulation-based where the word length is chosen while observing error criteria and the process is repeated until word length converges.

Since FPGAs IP cores are optimized for certain quantization word length, when it is used in the implementation of specific modules, it is not necessary to reduce the quantization of the internal signals of these modules to a number of bits less than the IP core size. For example, In case of Xilinx Virtex5 FPGAs, the multipliers are 25 X 18 bits. Hence, the first attempt for quantizing the internal signals of the multipliers modules is therefore 25 bits. The same FPGA family also supports dedicated RAMs which are 32-or-64 bit wide. Hence, data stored in these RAMs is allowed to be 64-bit wide without additional hardware cost.

Figure 5-13 shows the fixed-point design methodology. It starts by performing the quantization of the top-level signals. One signal at a time is quantized. Each attempt needs to be simulated. After all top-level signals are quantized, the top-level entities are quantized. Each top-level entity is assigned a part of the rest of the performance decrease budget, based on the hardware cost function. Quantizing the internal signals of a top-level entity is almost independent of the internal quantization of other top-level entities. Again, the quantization values which are not easily derived from already quantized signals need to be checked by simulation.

Quantization in Matlab is performed using function calls. It is very important that identical functions are available for hardware implementation. The two design representations should be identical at the bit level. Each top-level block which is converted to hardware needs to be

simulated and compared with the fixed-point Matlab model. The maximum performance degradation is set to 0.5 dB when the Bit Error Rate (BER) is less than 0.001. Each block was quantized separately and the word lengths with the corresponding degradation were noticed. After that, the fixed-point blocks were simulated together and the word lengths were changed slightly according to the degradation constraints mentioned above. Figure 5-14 shows the simulation results for MIMO-OFDM receiver with 8, 10 and 12 bits Fixed-Point representation, it could be observed that 12 bits representation satisfy the maximum performance degradation which is set at 0.5dB.

A complete real-time MIMO-OFDM Transceiver is implemented using the above proposed optimizations and 12 bits Fixed-Point representation. Table 5-6 introduce the hardware resource requirements comparison between Floating-Point and Fixed-Point models for the complete Transceiver design. The reduction in recourses requirements due to quantization could be clearly identified. Post place & route simulations are carried out to make sure that the optimized design is meeting the design constraints. Finally, the onboard verification is conducted using Matlab co-simulation and the proposed design platform. In this final step, the data is generated by Matlab and sent to the FPGA through the proposed UART interface module for processing, then received back in Matalb for verification. The co-simulation result has proven to be identical to the result obtained early using only Matlab simulation.
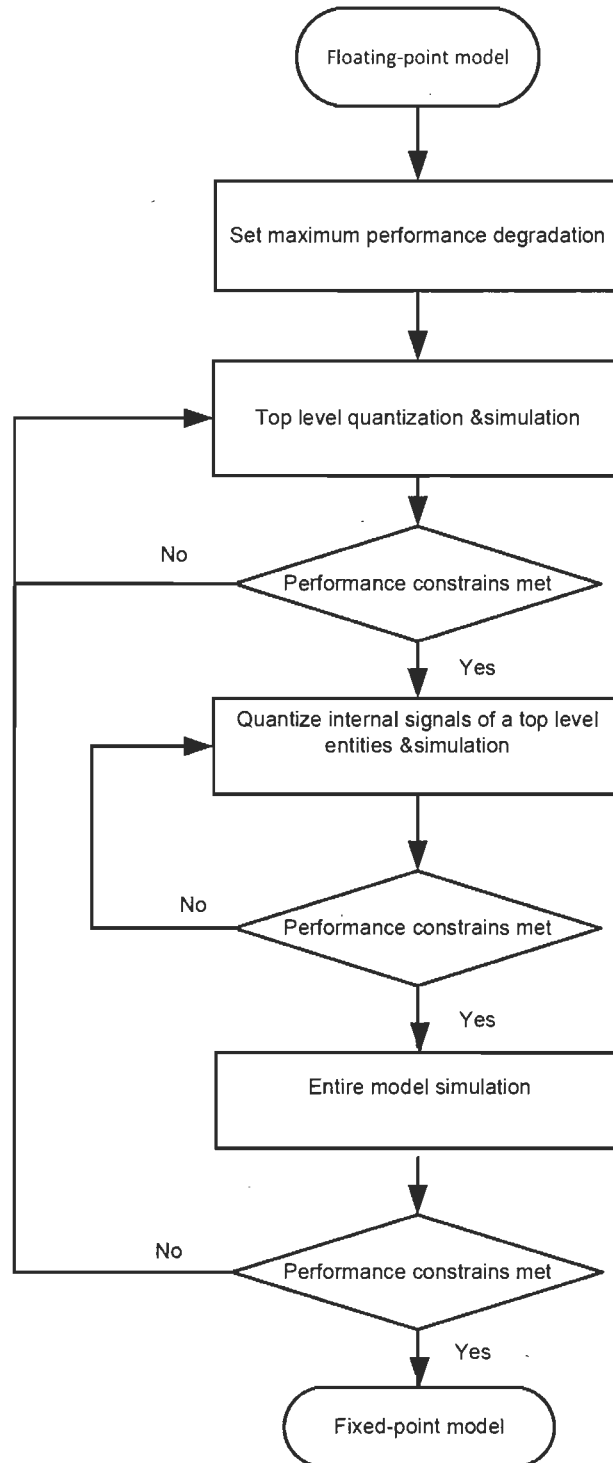
Figure 5-13 Fixed-Point Design flow for MIMO-OFDM
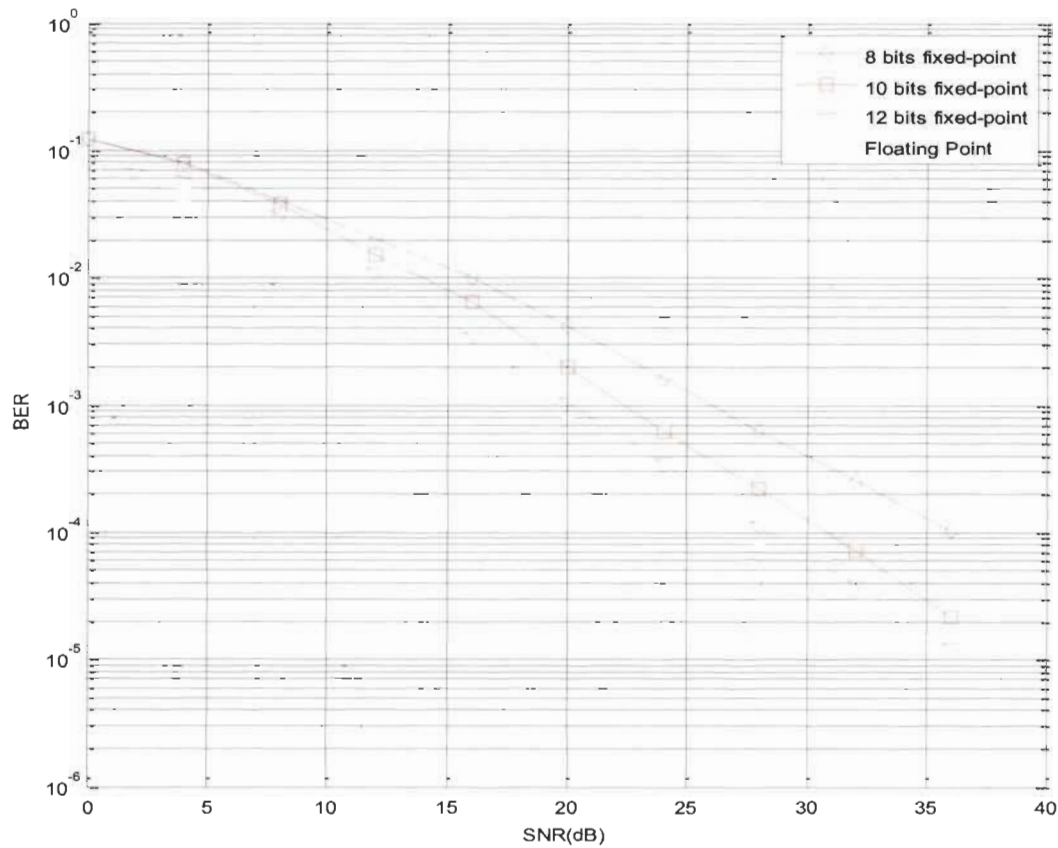
Figure 5-14 Matlab simulation for MIMO-OFDM receiver with Fixed-Point representations

Table 5-6: Consumed resources for optimized Transceiver with Floating-Point Vs Fixed-Point

Resource	Floating-point		12-bit Fixed point	
	Consumed number	Percentage of Virtex 5 resources	Consumed number	Percentage of Virtex 5 resources
Slice Registers	26712	92%	18364	63%
Slice LUTs	26800	92%	25070	83%
Block RAM/FIFO	24	40%	20	33%
DSP48E	48	100%	28	58%

5.6   Conclusion

In order to reduce the hardware resources requirements for the proposed MIMO-OFDM architecture, several optimization techniques have been introduced in this chapter. First, a pipelined architecture in which only one IFFT/FFT block is shared among all transmitting/receiving antennas saves more than 30 percent of the hardware resources while maintaining the same data rate. Second, an efficient low complexity algorithm for despreading unit based on counters and comparators is proposed. While the despreading unit based on matched filter consumes a great amount of resources, the proposed despreading algorithm produced a significant reduction for hardware resources requirements. Third, to reduce the channel equalization, an optimized architecture for complex matrix inversion using GJ-elimination is introduced. The proposed architecture performs the GJ-elimination for complex matrix by calculating the critical elements only it results in a reduced hardware resources and execution time. Finally, Fixed-Point FPGA architecture is introduced. the maximum allowed performance loss due to quantization is defined, then the tradeoffs between BER performance and area reduction are investigated and the final results are introduced and analyzed.

Chapitre 6 - Summary and future work

6.1 Summary

The increasing demand of high speed data transmission over wireless communication channel calls for advanced wideband transmission techniques as well as suitable detection algorithms at the receiver side. MIMO is combined with OFDM in order to use the limited radio spectrum more efficiently. MIMO-OFDM has the ability to provide more reliability and robustness to transmission in the wireless environment. Many coding schemes are proposed in literature to code the transmitted MIMO-OFDM symbols over space, time, and/or frequency in order to increase the system diversity and mitigate the wireless channel impairments. However, most of these reported schemes are not realistic in terms of hardware implementation due to the high computational complexity of the decoding algorithms. In addition to that, those schemes are designed for single user and not suitable for multiple access.

In this thesis, a novel transmission coding scheme is proposed for MIMO-OFDMA to support multiple access and greatly reduce the detection algorithm complexity. In this scheme, each user is assigned a unique set of orthogonal spreading codes. The user's data are spread over multiple antennas, OFDM symbols, and subcarriers. The selection of the spreading code is done using either parity or permutation techniques. This in turn reduces the detection complexity; since the determination of the spreading code at the receiver side directly identifies the transmitted data block from the corresponding user. In addition to that, the increased diversity obtained by spreading the transmitted signal over three domains has produced a

significant BER performance improvement in the existence of frequency selective fading channels. Furthermore, the system allows flexible data rates and efficient user multiplexing. Hence, better spectrum efficiency is achieved.

Matlab simulations are carried out to evaluate the BER performance of the proposed system for different antenna configuration, spreading code length, channel equalizations, and multi-user access. Simulation results have showed that the new scheme provides better performance as compared to conventional MIMO-OFDM with STBC duo to its capability of maintaining maximum achievable diversity on the receiver side.

MIMO-OFDM is computationally challenging system and requires a development environment that enables modeling the entire system accurately while taking into consideration large number of parameters. The second contribution of this thesis is the introduction of a systematic design methodology and real-time prototyping platform. Unlike most of the reported design in literature where Matlab model is converted into Simulink (RTL) model by means of schematic entry to validate the architecture functionality, then Simulink model is converted to VHDL to be ported to FPGA chip. In the proposed methodology, the Matlab model is directly converted to VHDL and ported to the FPGA chip, as a result significant time has been saved in the design process due to the elimination of intermediate conversion. UART is used to effectively manage the communication between Matlab and the FPGA board to perform co-simulation between the ported design and Matlab. UART core functions are implemented using VHDL and integrated into the MIMO-OFDM FPGA chip. The function of the UART interface is validated and the synthesis results showed that it consumes less than 1% of the total hardware resources of the target FPGA chip.

Floating-Point FPGA architecture for the proposed MIMO-OFDM algorithms has been developed and integrated with the proposed platform. It has been divided into smaller blocks and the Floating-Point baseband RTL architecture for each module is described in details. Matlab/VHDL co-simulation was conducted for each block to verify the functional and behavioral validity of the code-mapping. The synthesis results for the initial proposed architecture reveals huge resources requirements. Hence, the third contribution of this thesis is the proposal of different optimization techniques in order to reduce the system complexity and save hardware resources, time and power requirements. Among those optimization methods that are introduced :

- A pipelined architecture in which only one IFFT/FFT block is shared among all transmitting/receiving antennas. The proposed architecture saves more than 30 percent of the hardware resources while maintaining the same data rate.

- An efficient low complexity algorithm for dispreading unit based on counters and comparators in order to be used in the receiver for data decoding. While the dispreading unit based on matched filter consumes a great amount of resources, because it incorporates several arithmetic operations such as multiplication, division and square root calculation. The proposed dispreading algorithm greatly reduces the hardware resources requirements because it only uses counters, comparators and basic control logic.

- An optimized architecture for complex matrix inversion using GAUSS-JORDAN elimination (GJ-elimination) to be used in MIMO-OFDM receiver is proposed. The proposed architecture performs the GJ-elimination for complex matrix element by element. Only critical arithmetic operations are calculated to get the needed values

without performing all the arithmetic operations of the GJ-elimination algorithm. The algorithm results in a reduced hardware resources and execution time.

- Finally, Fixed-Point FPGA architecture is developed, where the maximum allowed performance loss due to quantization is defined, then the tradeoffs between BER performance and area reduction are investigated.

6.2 Future work

The work in this thesis has focused on the design of MIMO-OFDM transmission scheme and data detection algorithms in addition to the hardware architecture optimization. While the research is comprehensive, there remains further work which could be done to further enhance system performance and real time operation. Specifically, adaptive coding, adaptive modulation, and integration with channel estimation.

6.2.1 Adaptive coding

The proposed transmission scheme is most effective in highly mobile and harsh channel conditions, which in turn justify the data rate reduction due to spreading. However, in situations where the channel condition is not severely scattered, the system could use different coding scheme that has less effect on the data rate reduction and lower computational complexity at the receiver side. The channel status could be acquired by the transmitter through a feedback from the receiver to the transmitter.

6.2.2 Adaptive modulation

If the modulation order at the transmitter is adjustable, the system will have the capability to achieve the best performance through the channel. The system adapts the modulation scheme to

suit the transmission environment in order to maximise the channel capacity. In this thesis, the proposed algorithms and implementations are investigated for BPSK modulation. However, the effect of other modulation scheme such as QPSK and QAM on both algorithmic and hardware architectural needs further analysis.

6.2.3 Integration with channel estimation

The proposed system assumed the existence of CSI at the receiver side, however channel estimation module has not been considered in this thesis. Blind channel estimation technique is suitable for slow time varying channels and requires high complexity algorithms at the receiver. On the other hand, pilot aided channel estimation algorithms have lower implementation complexity and could be used with different types of channels. However, the pilot insertion within the transmitted signal reduces the transmission data rate. As a result, pilot aided channel estimation algorithms are considered more suitable for the proposed MIMO-OFDM system. A comparative analysis between different algorithms need to be conducted, while the trade-offs between bandwidth efficiency and accurate estimation are considered. Finally, the hardware architecture of the selected algorithms needs to be developed and integrated with the proposed architecture for MIMO-OFDM.

References

[1] Foschini GJ and Gans M.J., "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, Kluwer Academic Press, no. 6, PP. 311–335, 1998.

[2] Paulraj AJ, Gore DA, Nabar RU & Bolcskei H, "An overview of MIMO communications - A key to gigabit wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 198–218, Feb. 2004.

[3] Hongwei Yang, "A road to future broadband wireless access: MIMO-OFDM-based air interface," *IEEE Communications,* Vol. 43, No. 1, 2005, pp. 53–60.

[4] International Telecommunication Union. (Geneva, 6 December 2010). *ITU World Radio communication Seminar highlights future communication technologies* [Online]. Available: http://www.itu.int/net/pressoffice/press_releases/2010/48.aspx.

[5] Zachary Lutz. (Nov 8th, 2011). AT&T commits to LTE-Advanced deployment in 2013, Hesse and Mead unfazed [online]. Available: http://www.engadget.com/2011/11/08/atandt-commits-to-lte-advanced-deployment-in-2013-hesse-and-mead.

[6] R. W. Chang, "Synthesis of band-limited orthogonal signals for multi channel data transmission," *Bell System. Tech. J.*, vol. 45, no. 10, pp. 1775–1796, Dec. 1966.

[7] A. Wittneben, "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," *in Proc. IEEE ICC93*, Vol. 3, Geneva, Switzerland, 1993, pp. 1630-1634.

[8] V. Tarokh, N. Seshadri and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, Vol. 44, pp. 744-765, Mar. 1998.

[9] V. Tarokh, H. Jafarkhani and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inform. Theory,* Vol. 45, pp. 1456 - 1467, July 1999.

[10] G. Raleigh and J. M. Cioffi, "Spatial-temporal coding for wireless communications," *IEEE Trans. Communications*, Vol. 46, pp. 357-366, 1998.

[11] C. Fragouli, N. Al-Dhahir and S. Diggavi, "Pre-filtered space - time equalizer for frequency selective channels," *IEEE Trans. Communications*, Vol. 50, pp. 742-753, May 2002.

[12] Ait-Idir, T.; Saoudi, S.; Naja, N., "Space–Time Turbo Equalization With Successive Interference Cancellation for Frequency-Selective MIMO Channels," *IEEE Transactions on Vehicular Tech.*, Volume: 57 , Issue: 5 pp 2766 - 2778 , 2008.

[13] Firag, A.; Garth, L.M., "Adaptive Joint Decoding and Equalization for Space-Time Block-Coded Amplify-and-Forward Relaying Systems," *IEEE Transactions on Signal Processing*, Volume: 57, Issue: 3, pp : 1163 – 1176, 2009.

[14] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Select Areas in Communications,* vol. 16, pp. 1451–1458, August 1998.

[15] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. Journal.* pp. 41-59, 1996.

[16] Cong Xiong; Xin Zhang; Kai Wu; Dacheng Yang, "A simplified fixed-complexity sphere decoder for V-BLAST systems," *IEEE Comm. Letters*, Volume: 13, Issue: 8 , pp. 582 - 584, 2009.

[17] W. J. Choi; K. W. Cheong and J. M. Cioffi, "Iterative soft interference cancellation for multiple antenna systems," *IEEE Wireless Communications and Networking Conference*, no. 1, pp. 304 - 309, September 2000.

[18] Karjalainen, J.; Veselinovic, N.; Kansanen, K.; Matsumoto, T., "Iterative Frequency Domain Joint-over-Antenna Detection in Multiuser MIMO," *IEEE Trans on Wireless Communications*, Volume: 6 , Issue: 10 , 3620 - 3631, 2007.

[19] N. Seshadri and J. H. Winters, "Two schemes for improving the performance of frequency-division duplex (FDD) transmission systems using transmitter antenna diversity," *Int. J. Wireless Information Networks*, Vol. 1, pp. 49-60, Jan. 1994.

[20] A. van Zelst, "Space division multiplexing algorithms," *in Proc. of the 10th Mediterranean Electrotechnical Conference (MELECON) 2000*, vol. 3, May 2000, pp. 1218-1221.

[21] H. Bolcskei and A. J. Paulraj, "Space-frequency coded broadband OFDM systems," *in IEEE Conference on Wireless Communications and Networking*, vol.1, pp. 1-6, 2000.

[22] H. El Gamal, A. R. Hammons, L. Youjian, M. P. Fitz and O. Y. Takeshita, "On the design of space-time and space-frequency codes for MIMO frequency-selective fading channels," *IEEE Transactions on Information Theory,* vol. 49, pp. 2277-2292, 2003.

[23] K. F. Lee and D. B. Williams, "A Space-Frequency Transmitter Diversity Technique for OFDM Systems," *Proc. IEEE Global Commun. Conf.*, Nov. 27–Dec. 1, 2000, vol. 3, pp. 1473–77.

[24] H. Bölcskei and A. Paulraj, "Space-Frequency Codes for Broadband Fading Channels," *Proc. IEEE Int'l. Symp. Info. Theory*, Washington, DC, June 24–29, 2001, p.219.

[25] W. Su *et al.*, "Obtaining Full-Diversity Space-Frequency Codes from Space-Time Codes Via Mapping," *IEEE Trans. Signal Processing*, vol. 51, Nov. 2003, pp. 2905–16.

[26] W. Zhang, X.-G. Xia, and P. C. Ching, "High-Rate Full-Diversity Space-Time-Frequency Codes for Broadband MIMO Block Fading Channels," *IEEE Trans. Commun.*,vol. 55, Jan. 2007, pp. 25–34.

[27] M. E. Gärtner and H. Bölcskei, "Multiuser Space-Time/Frequency Code Design," *Proc. IEEE Int'l. Symp. Info. Theory*, July 9–14, 2006, pp. 2819–23.

[28] T. Kiran and B. S. Rajan, "A Systematic Design of High-Rate Full-Diversity Space-Frequency Codes for MIMOOFDM Systems," *Proc. IEEE Int'l. Symp. Info. Theory*, Adelaide, Australia, Sept. 4–9, 2005, pp. 2075–79.

[29] W. Su, Z. Safar, and K. J. R. Liu, "Full-Rate Full-Diversity Space-Frequency Codes with Optimum Coding Advantage," *IEEE Trans. Info. Theory*, vol. 51, Jan. 2005, pp. 229–49.

[30] R. Cmar, L. Rijnders, P. Schaumont, S. Vernalde, and I. Bolsens, "A methodology and design environment for DSP ASIC fixed point refinement," in *Proceedings of Conference on Design, Automation and Test in Europe (DATE '99)*, pp. 271–276, Munich, Germany, March 1999.

[31] J. Dowle,1 S. H. Kuo,2 K. Mehrotra,1 and I. V.McLoughlin1, "An FPGA-Based MIMO and Space-Time Processing Platform," *EURASIP Journal on Applied Signal Processing*, Volume 2006, Article ID 34653, Pages 1–14.

[32] T. Kaiser, A. Wilzeck, M. Berentsen, and M. Rupp, "Prototyping for MIMO-systems: an overview," in *Proceedings of 12th European Signal Processing Conference (EUSIPCO '04)*, Vienna, Austria, September 2004.

[33] OFDM and Multi-Channel Communication Systems, *National Instruments Measurement Fundamentals series*, Publish Date: Feb 02, 2012.

[34] Jim Geier, *Enabling Fast Wireless Networks with OFDM* [online]. Available: *http://www.eetimes.com/design/ communications-design/4140000/Enabling-Fast-Wireless-Networks-with-OFDM*.

[35] Rodger Ziemer and William Tranter, *Principles of Communications - Systems Modulation and Noise*, fifth edition, John Wiley and Sons Ltd, NJ, 2002.

[36] K. B. Letaief and Y. Zhang, "Dynamic Multiuser Resource Allocation and Adaptation for Wireless Systems," *IEEE Wireless Commun.*, vol. 13, no. 4, Aug. 2006, pp. 38–47.

[37] S. Srikanth, V. Kumaran, C. Manikandan et al., "Orthogonal Frequency Division Multiple Access: is it the multiple access system of the future," *AU-KBC Research Center*, Anna University, India

[38] A. van Zelst, R. van Nee, and G. A. Awater, "Turbo-BLAST and its performance", *in Proc. of IEEE Vehicular Technology Conference (VTC) 2001 Spring*, vol. 2, May 2001, pp. 1282-1286

[39] Xiaodong Li, H. Huang, G. J. Foschini, and R. A. Valenzuela, "Effects of iterative detection and decoding on the performance of BLAST", *in Proc. of the IEEE Global Telecommunications Conference (GLOBECOM) 2000*, vol. 2, pp. 1061- 1066.

[40] A. van Zelst, "Per-antenna-coded schemes for MIMO OFDM", *in Proc. of the IEEE International Conference on Communications (ICC) 2003*, vol. 4, May 2003, pp. 2832-2836.

[41] J.-C. Guey, M. P. Fitz, M. R. Bell, and W.-Y. Kuo, "Signal design for transmit diversity wireless communication systems over Rayleigh fading channels", *in Proc. of the IEEE Vehicular Technology Conference (VTC) 1996*, vol. 1, May 1996, pp. 136-140.

[42] J. G. Proakis, *Digital Communications*, Third Edition, New York, McGraw-Hill, 1995, McGraw-Hill Series in Electrical and Computer Engineering.

[43] E. Biglieri, G. Taricco and A. Tulino, "Performance of space-time codes for a large number of antennas", *IEEE Transactions on Information Theory*, vol. 48, no. 7, July 2002, pp. 1794-1803.

[44] Jibing Wang, E. Biglieri, and Kung Yao, "Asymptotic performance of space-frequency codes over broadband channels", *IEEE Communications Letters*, vol. 6, no. 12, Dec. 2002, pp. 523-525.

[45] H. Yang, "A road to future broadband wireless access: MIMO-OFDMBased air interface," *IEEE Communications Magazine*, vol. 43, no. 1, pp. 53–60, 2005.

[46] Haixia Zhang; Dongfeng Yuan; Hsiao-Hwa Chen," On Array-Processing-Based Quasi-Orthogonal Space–Time Block-Coded OFDM Systems ," *IEEE Transactions on Vehicular Technology*, Volume: 59 , Issue: 1 , Page(s): 508 – 513

[47] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel", *IEEE Transactions on Communications*, vol. 51, no. 3, March 2003, pp. 389-399.

[48] M. Sellathurai and S. Haykin, "TURBO-BLAST for high-speed wireless communications", *in Proc. of IEEE Wireless Communications and Networking Conference (WCNC) 2000*, vol. 1, Sept. 2000, pp. 315-320

[49] A. M. Tonello, "Space-time bit-interleaved coded modulation with an iterative decoding strategy", *in Proc. of the 52 nd IEEE Vehicular Technology Conference (VTC) 2000 Fall*, vol. 1, Sept. 2000, pp. 473-478.

[50] S. L. Ariyavisitakul, "Turbo space-time processing to improve wireless channel capacity", *IEEE Transactions on Communications*, vol. 48, no. 8, Aug. 2000, pp. 1347-1358.

[51] J. Boutros, and E. Viterbo, "Signal Space Diversity: A Power and Bandwidth Efficient Diversity Technique for the Rayleigh Fading Channel," *IEEE Trans. Info. Theory*, vol. 44, July, 1998, pp. 1453–67.

[52] W. Zhang, X.-G. Xia, and P. C. Ching, "Full-Diversity and Fast ML Decoding Properties of General Orthogonal Space-Time Block Codes for MIMO-OFDM Systems," *IEEE Trans. Wireless Commun.*, vol. 6, no. 5, May 2007, pp. 1647–53.

[53] H. E. Gamal and M. O. Damen, "Universal Space-Time Coding," *IEEE Trans. Info. Theory*, vol. 49, May, 2003, pp. 1097–19.

[54] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Info. Theory*, vol. 45, July 1999, pp. 1639–42.

[55] W. Zhang and K. B. Letaief, "A Systematic Design of Multiuser Space-Frequency Codes for MIMO-OFDM Systems," *Proc. IEEE Int'l. Conf. Commun.*, Glasgow, Scotland, UK, June 24–28, 2007.

[56] A. Bury, J. Egle, and J. Lindner, "Diversity comparison of spreading transforms for multicarrier spread spectrum transmission," *IEEE Transactions on Communications*, vol. 51, no. 5, pp. 774–781, 2003.

[57] M. L. McCloud, "Analysis and design of short block OFDM spreading matrices for use on multipath fading channels," *IEEE Transactions on Communications*, vol. 53, no. 4, pp. 656–665, 2005.

[58] D'Amours, C.; "Parity bit selected spreading sequences: a block coding approach to spread spectrum," *IEEE communications letters*, vol. 9, no. 1, january 2005 , page(s): 16 – 18.

[59] Dahmane, A.O.; D'Amours, C.; "Parity Bit Selected Spreading for MIMO-CDMA Using Hadamard Codes and Gold Scrambling Sequences," *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, Page(s): 536 – 540.

[60] Dahman and Shayan, "Performance evaluation of space-time-frequency spreading for MIMO OFDM-CDMA systems," *EURASIP Journal on Advances in Signal Processing*, 2011, 2011:139.

[61] Nikolaos Bartzoudis, Oriol Font-Bach, Antonio Pascual-Iserte and David López Bueno, "A Real-Time FPGA-based mobile WiMAX transceiver supporting multi-antenna configurations," *Argentine School of Micro-Nanoelectronics Technology and Applications (EAMTA)*, 2011.

[62] Jeoong Sung Park; Ogunfunmi, T.," FPGA implementation of the MIMO-OFDM physical layer using single FFT multiplexing," *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, PP. 2682 – 2685, 2010.

[63] Veena M.B. , M.N.Shanmukha Swamy," Implementation of Re-configurable Digital Front End Module of MIMO-OFDM using NCO," *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 5, No 2, September 2011

[64] Fang Yi-yuan; Chen Xue-jun, "Design and Simulation of UART Serial Communication Module Based on VHDL," *3rd International Workshop on Intelligent Systems and Applications (ISA)*, PP. 1 – 4, 2011.

[65] W. Han, T. Arslan, A. T. Erdogan, and M. Hasan, "Multiplier-less based parallel-pipelined FFT architectures for wireless communication applications," *2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005)*, vol. 5, pp. v/45-v/48, Mar 2005.

[66] V. Jungnickel, T. Haustein A. Forck, S. Schiffermueller, H. Gaebler, and C. von Helmolt, "Real-time concepts for MIMO-OFDM," *in proc. CIC/IEEE Global Mobile Congress*, pp. 47-52, Oct 2004.

[67] Transit Access Points. Rice University TAPs research group [Online]. Available: http://taps.rice.edu/.

[68] UCLA MIMO wireless communication research group [Online]. Available: http://www.ee.ucla.edu/~mimo.

[69] Ali Irturk, Bridget Benson, Shahnam Mirzaei, Ryan Kastner, "An FPGA Design Space Exploration Tool for Matrix Inversion Architectures," *SASP '08 Proceedings of the 2008 Symposium on Application Specific Processors*, Pages 42-47.

[70] Davide Cescato, Moritz Borgmann, Helmut Bölcskei, Jan Hansen, and Andreas Burg, " Interpolation-based QR decomposition in MIMO-OFDM systems," *IEEE Workshop on Signal Processing Advances in Wireless Communications - SPAWC* , 2005.

[71] Zheng-Yu Huang and Pei-Yun Tsai," Efficient Implementation of QR Decomposition for Gigabit MIMO-OFDM Systems," *IEEE Trans. on circuits and systems*, vol. 58, no. 10, October 2011.

[72] Senthilvelan, M. ; Iancu, D. ; Glossner, J. ; Moudgill, M. ; Schulte, M., " Software Solutions for Converting a MIMO-OFDM Channel into Multiple SISO-OFDM Channels," *Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications,. WiMOB*, 2007.

[73] Janier Arias-Garc´ıa, Ricardo Pezzuol Jacobi, Carlos H. Llanos, Mauricio Ayala-Rinc´on,"A suitable FPGA implementation of floating-point matrix inversion based on

149

Gauss-Jordan Elimination," *2011 vii southern conference on programmable logic (SPL)*, pp. 263–268, 13-15 April 2011.

Appendix A - Functional Simulation

Transmitter function simulation:

In order to verify the operation of the transmitter module a test bench was written for the transmitter. The same input data used in the Matlab model were used as an input for the transmitter hardware model. In Matlab model the data sequences from antenna 1 and antenna 2 for the first iteration were given as :

1 0 1 0 1 0 1 1 0 1 1 1 1 1 1 1 0 0 1 1 0 1 0 0

1 1 0 1 1 1 0 0 1 1 0 1 1 0 0 0 0 1 0 0 0 1 0 1

This data sequence was written in the test bench as an input for the MIMO-OFDM transmitter sub-system and the system clock was set to 100MHZ (10 ns). The function simulation output is presented in hexadecimal format for single-precision Floating-Point numbers. Figure A-1 and Figure A-2 show that the hardware model of the transmitter sub-system gives exactly the same results were compared to the Matlab model.
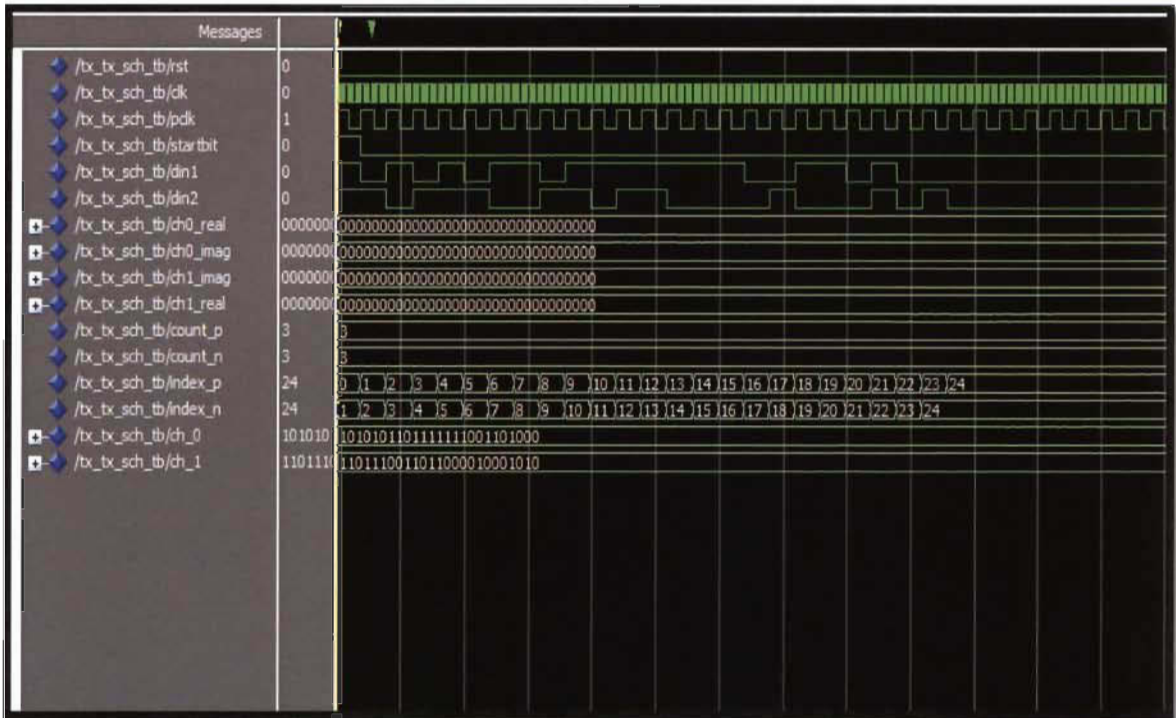
Figure A-1 Simulation input to the MIMO transmitter Sub-system
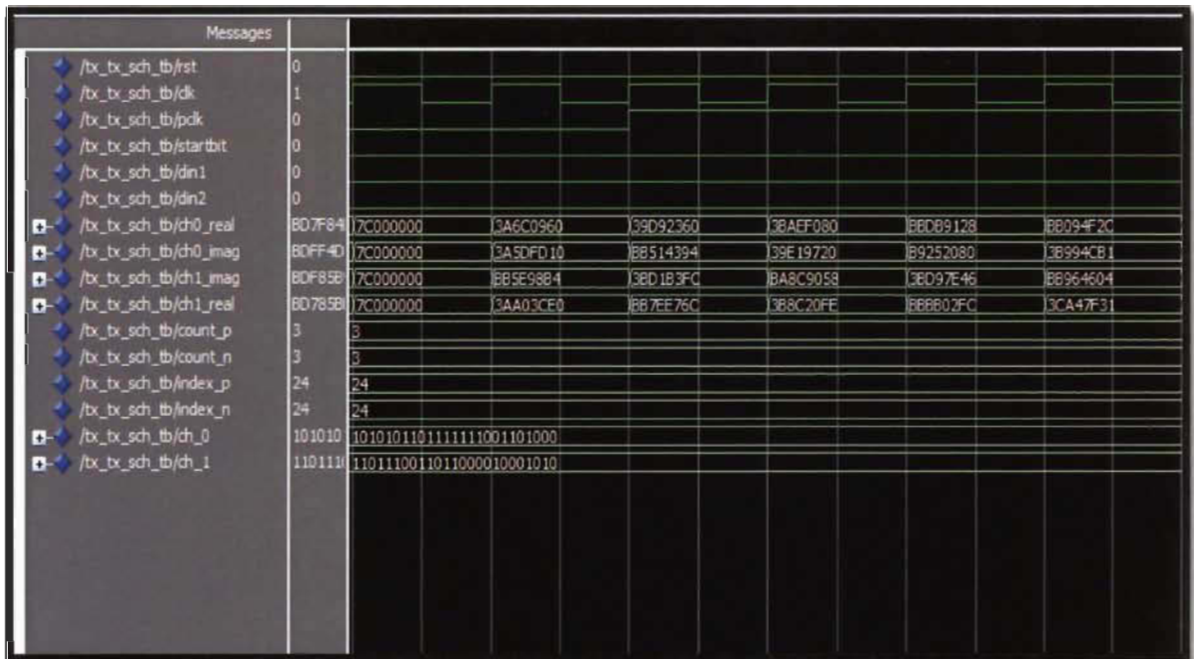
Figure A-2 Simulation output of the MIMO transmitter sub-system

Receiver function simulation:

At the receiver side, the transmitted data in addition to the channel noise generated by Matlab was used in the VHDL test bench as an input for the receiver sub-system. Both parity and permutation despreading are tested separately. First each despreading unit was simulated before simulating the whole receiver. This was done by writing a test bench to feed the despreading unit with spreaded data and observe the output. Figure A-3 to Figure A-6 show that the despreading unit is producing the expected output.
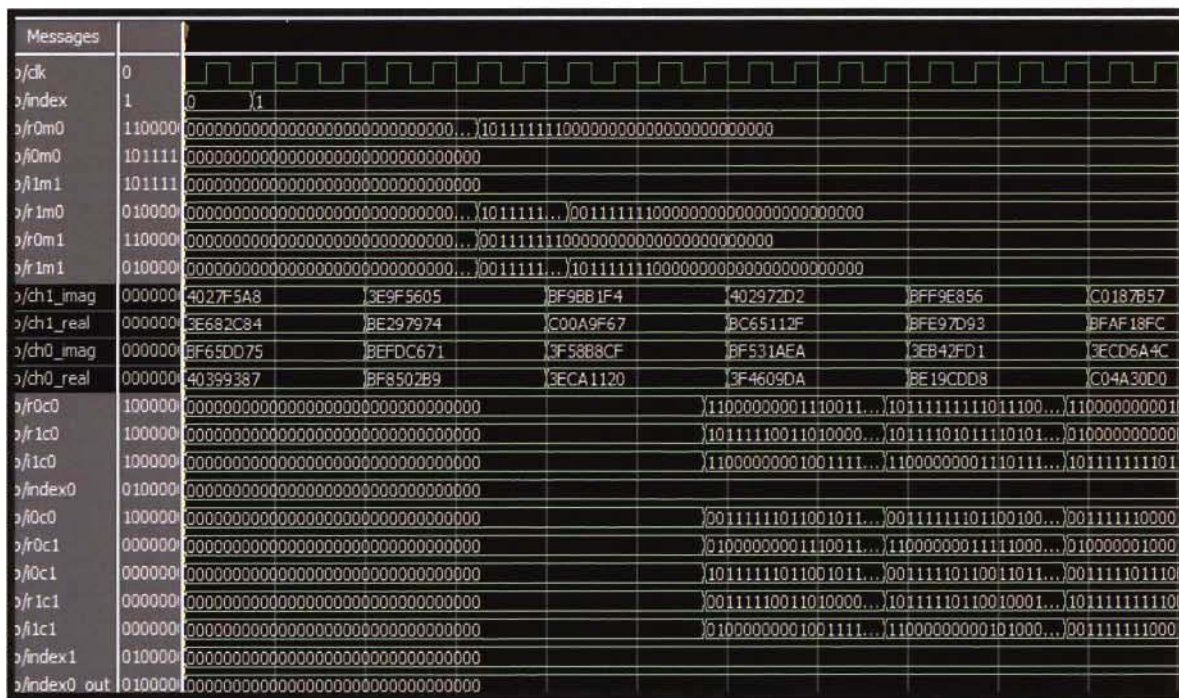
Figure A-3 Simulation input to the parity despreading

Figure A-4 Simulation output from the parity despreading

Figure A-5 Simulation input to the permutation despreading
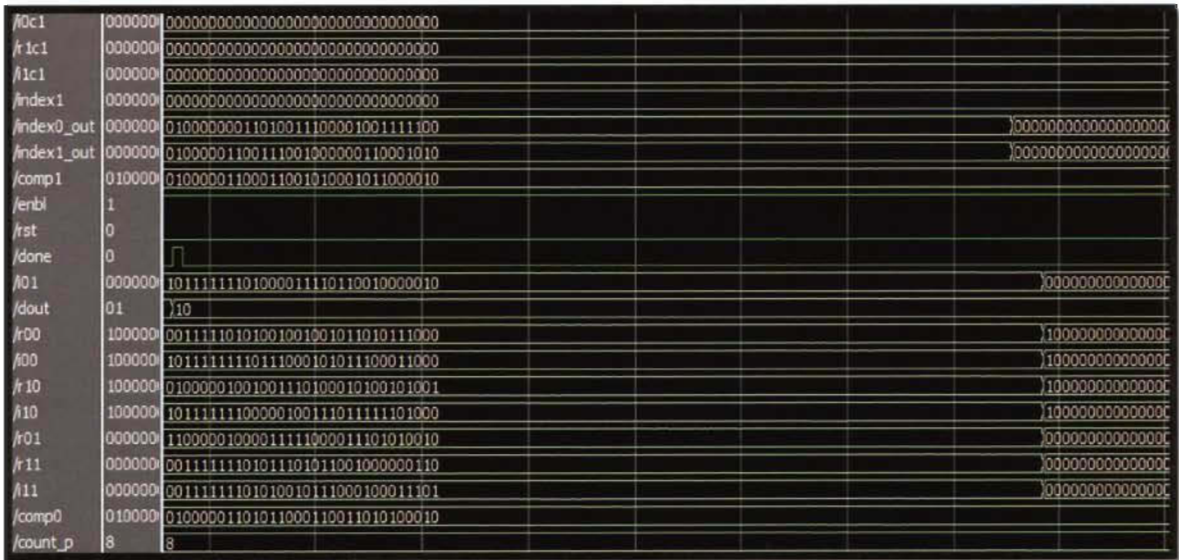
Figure A-6 Simulation output of the permutation despreading

After verifying that the despreading units are functioning correctly, simulation for the whole receiver system was done. Figure A-7 to Figure A-10 show that the receiver unit is producing the expected output.

Figure A-7 Simulation input to the parity receiver sub-system
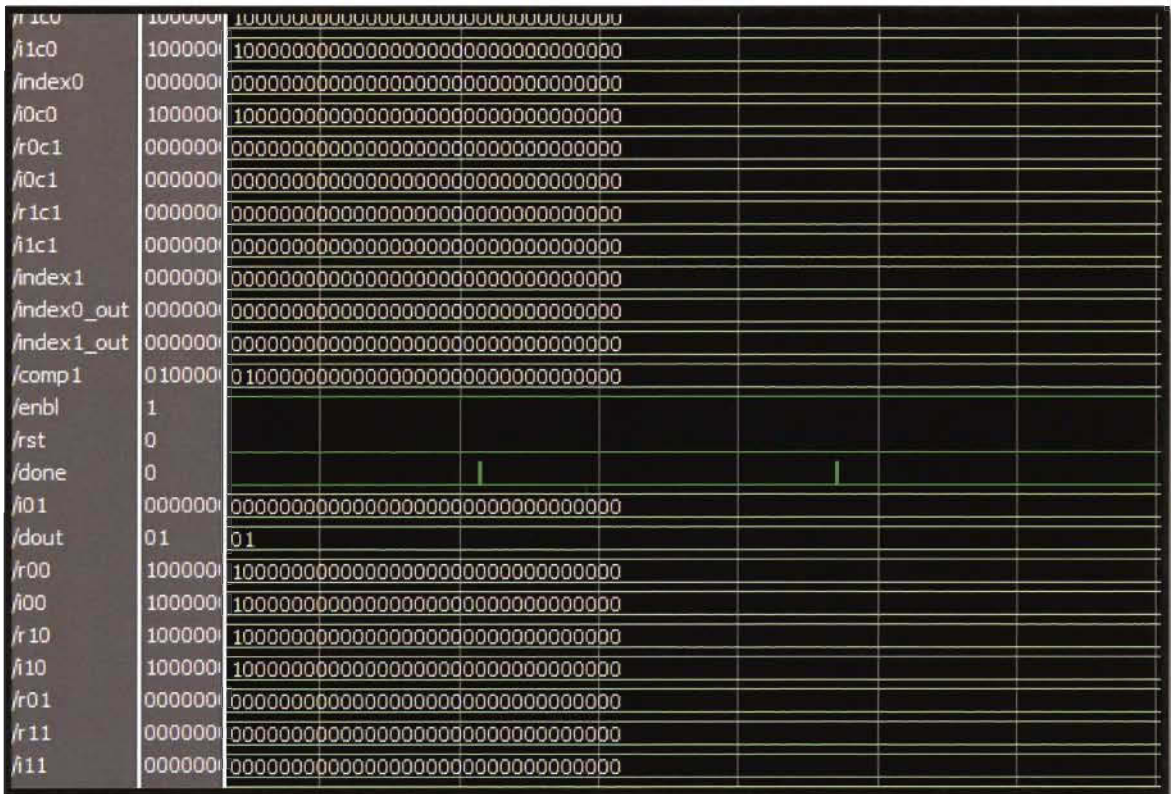
Figure A-8 Simulation output of the parity receiver sub-system
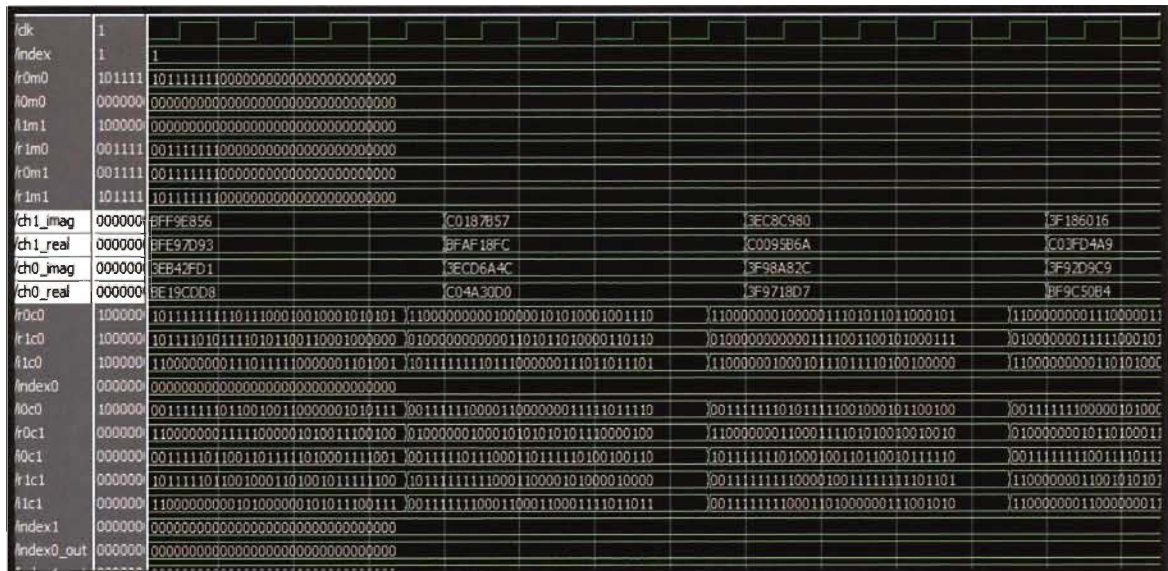

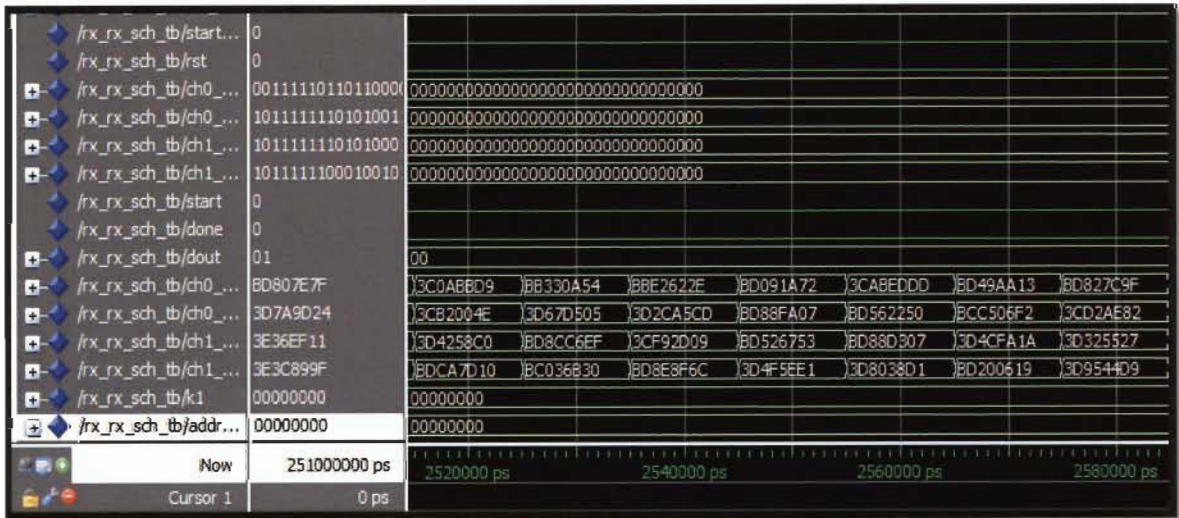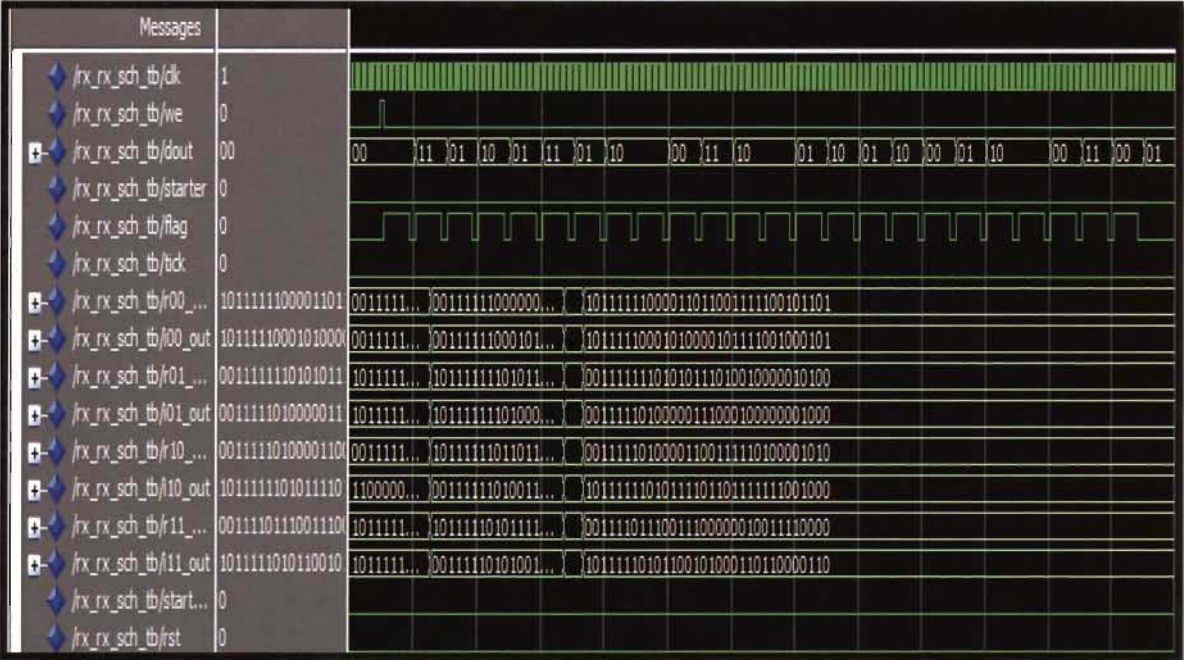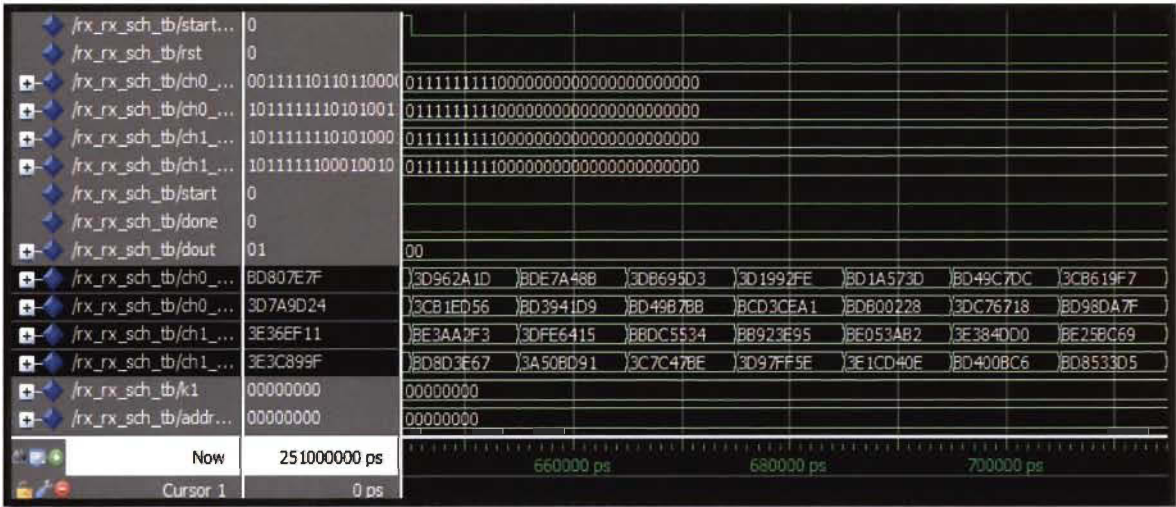Figure A-9 Simulation input to the permutation receiver sub-system
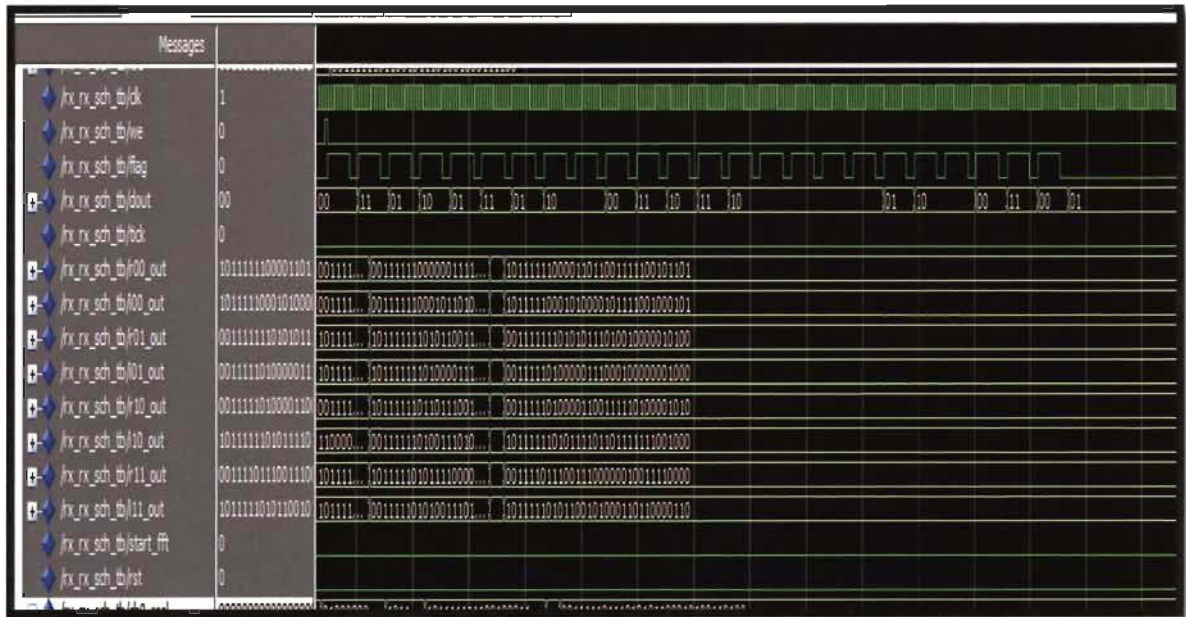
Figure A-10 Simulation output of the permutation receiver sub-system

Annexe B - Résumé de la thèse en français

Dans cette thèse, un nouveau transmetteur/récepteur MIMO-OFDM (*Multiple-Input Multiple-Output - Orthogonal Frequency Division Multiplexing*) a été développé afin de permettre l'accès multiple à travers de multiples antennes, sous-porteuses, trames OFDM, et utilisateurs via un code commun. Cela permet d'obtenir une meilleure efficacité spectrale tout en améliorant la performance du taux d'erreur (BER). Le plan proposé utilise le bit de parité sélectionné ou bien les techniques de permutation pour choisir le code d'étalement sur le transmetteur. Ainsi, la complexité de détection au récepteur se réduit significativement sous l'effet d'une identification du code d'étalement en obtenant directement les symboles de données transmises. De plus, cette thèse propose une implémentation matérielle des algorithmes proposés. La seconde contribution de cette thèse consiste à introduire une méthodologie de conception systématique ainsi qu'une plateforme de prototypage temps réel. Une conversion de chaque algorithme suggéré de Matlab en programme prêt à implémenter sur la plateforme de prototypage cible FPGA est alors possible d'une façon systématique. En visant la réduction d'espace, de temps et de consommation, des techniques d'optimisation de l'architecture matérielle sont abordées. Parmi celles-ci, on propose notamment une architecture pipeline où un seul bloc IFFT/FFT est partagé par toutes les antennes émettrices/réceptrices ; un algorithme efficace à basse complexité pour le désétalement à base de compteurs et comparateurs ; et une architecture optimisée pour l'inversion de matrices complexes utilisant l'élimination de Gauss-Jordan (élimination GJ).

Finalement, une architecture FPGA à virgule fixe pour l'émetteur/récepteur MIMO-OFDM est développée.

B.2 Introduction

B.2.1 Problématique

Afin d'augmenter le débit binaire et la robustesse des liaisons montantes, les systèmes 4G utilisent les systèmes MIMO combinés à l'OFDM. L'avantage d'un système MIMO réside dans le fait qu'il peut atteindre un débit plus élevé que celui marqué par un système SISO sur la même bande passante et pour la même capacité de transmission. Les systèmes sans fil MIMO envoient et reçoivent les informations via deux ou plusieurs antennes. Les signaux se réfléchissent contre les objets de l'environnement en créant des chemins multiples. Dans les systèmes usuels, ces multi chemins engendrent des interférences et affectent la portée et la qualité du signal. En revanche, les systèmes MIMO combinent les chemins multiples nuisibles et les signaux d'utilisateurs pour diminuer les interférences ; et par conséquent, augmenter le débit de transmission de données et réduire le taux d'erreur (BER) d'une façon plus efficace que les systèmes SISO. De plus, la communication MIMO est destinée aux systèmes large bande au sein desquels les évanouissements sélectifs en fréquence sont présents ; ce qui rend inévitable la présence d'interférences intersymboles. Pour diminuer l'effet de ce dernier et simplifier l'égaliseur, MIMO est combiné à l'OFDM afin de transformer le canal sélectif en fréquences en un ensemble de canaux parallèles à évanouissement uniforme. La transmission par MIMO-OFDM est utilisée soit pour augmenter la robustesse du système soit pour améliorer le débit binaire. Dans un milieu avec beaucoup de dispersion, la diversité de transmission joue un rôle important pour maintenir la robustesse du système de communication sans fil. Les systèmes de

transmission qui se servent de la diversité utilisent la dimension spatiale pour ajouter de la redondance et maintenir, en conséquence, le débit binaire équivalent à celui d'un système SISO-OFDM afin d'accroître la performance BER. Le codage spatio-temporel génère une redondance en codant à travers les dimensions temps et espace ; le STBC (*Space-Time Block Code*) représente sans doute le codage le plus commun qui emploie le codage spatio-temporel (STC). D'autre part, le multiplexage spatial (SDM) est employé au cas où l'algorithme utilise différentes antennes pour transmettre des symboles à travers le canal sans redondance. Les systèmes SDM sont utilisés si l'on vise essentiellement l'augmentation des débits binaires.

Ni le codage SDM ni le codage STC ne peut réaliser une diversité multi chemins et les deux ont été proposés pour des canaux à évanouissement uniforme. Ils ne sont pas appropriés aux canaux à évanouissement sélectif en fréquence. Ces deux problèmes peuvent être résolus en introduisant plus de diversité en fréquences au système. MIMO-OFDM offre l'opportunité de coder les symboles transmis à travers différentes antennes (espace) et diverses sous-porteuses (fréquence). Ce codage est connu sous le nom SFBC (*Space-Frequency Block Code*). Il permet l'exploitation de la diversité multi chemins. Le STFBC (*Space-Time-Frequency Block Code*) est un autre procédé de codage, tridimensionnel, à travers l'espace. Les deux types de codage ont été récemment proposés dans la littérature. Néanmoins, la complexité du système demeure un obstacle majeur compte tenu de la complexité du décodage qui en résulte. De plus, la majorité des codes ST/SF existants sont destinés aux systèmes à utilisateur unique seulement. Pour les canaux à accès multiple, les codes ST/SF sont alors alloués séparément à chaque utilisateur, ce qui réduit le taux de transmission. Dans le cas d'un MIMO-OFDM conventionnel, par exemple, les utilisateurs sont séparés puis distribués sur différentes bandes de fréquences (sous-canaux), et chacun d'eux est codé séparément via STBC ou SFBC. Ceci mène à une chute du débit

binaire directement proportionnelle au nombre d'utilisateurs. Les raisons susmentionnées impliquent l'introduction d'un nouveau schéma de transmission qui permet l'accès multiple par l'intermédiaire d'une conception de codes conjoints sur multiples antennes, sous-porteuses OFDM, et utilisateurs.

L'amélioration significative de la performance des systèmes MIMO-OFDM est au détriment d'un décodage complexe à la réception. Par exemple, l'accroissement linéaire du débit binaire en fonction du nombre minimal d'antennes sur le transmetteur et le récepteur, dans un multiplexage spatial, n'est pas accompagné d'une simple augmentation linéaire de la complexité du décodeur, indépendamment de la nature des algorithmes utilisés. En outre, maximiser les avantages potentiels de la technologie d'antennes multiples nécessite de faire appel à des algorithmes plus complexes, approchant voire surpassant les limites technologiques et économiques de la technologie des circuits intégrés.

Selon la loi de Moore, la densité des transistors double chaque deux ans, ce qui limite le taux d'amélioration de la performance du système. D'un autre côté, et selon la loi de Shannon, l'évolution de la complexité des algorithmes est plus rapide que celle de la densité des puces visant à atteindre une capacité de canal maximale. Cela crée un vide entre la complexité des algorithmes et la performance matérielle, ce qui rend inévitable de penser à un design efficace assurant des architectures aussi compactes et puissantes qu'économiques.

Le détecteur, s'occupant de la séparation des flux de données multiplexées spatialement, est la composante la plus complexe d'un récepteur MIMO-OFDM. Seul l'ordre de complexité des algorithmes du récepteur MIMO-OFDM a été examiné ; toutefois, cela n'est adéquat qu'en cas de comparaisons qualitatives entre les différents algorithmes de décodage. Les résultats d'une telle analyse ne sont pas particulièrement pertinents à l'implémentation du système. D'un autre

côté, une analyse plus approfondie du niveau de complexité des algorithmes a été développée pour l'implémentation dans un processeur de signal numérique (DSP). Cependant, les implémentations DSP ne répondent pas aux exigences (par rapport aux débits) des systèmes MIMO-OFDM à large bande actuels et émergents. En conséquence, une mise en œuvre sur FPGA est requise pour l'implémentation d'algorithmes de décodage complexes. Également, des développements additionnels de systèmes MIMO-OFDM à haut débit et large bande sont requis afin de s'assurer que le seul facteur à influencer la performance du système est la capacité du canal sans fil et non la technologie. Habituellement, les développeurs d'algorithmes et les équipes de conception matérielle travaillent indépendamment les uns des autres. Ceci explique l'impossibilité d'implémenter à temps réel beaucoup d'algorithmes proposés, jugés irréalistes pour ce genre de mises en œuvre à cause de leur niveau de complexité ainsi que leurs problèmes de stabilité numérique. Cette thèse propose un environnement de développement permettant aux concepteurs de modéliser, d'une façon précise, un système complet. Cela comprend également le comportement et les interactions des sous-systèmes matériels et logiciels qui représentent les paramètres de la plateforme système.

B.2.2   Objectifs de la thèse

Cette thèse vise à proposer des algorithmes performants avec un niveau de complexité réaliste ainsi que des architectures FPGA optimisées pour un émetteur-récepteur MIMO-OFDM. Tout d'abord, pour réduire la complexité de l'algorithme de détection au récepteur et améliorer la performance du MIMO-OFDM, un nouveau schéma de transmission pour ce dernier basé sur l'étalement à bit de parité sélectionné et à bloc de permutation est proposé. Les données transmises, dans ce schéma, sont codées en espace, en temps et en fréquence. Le codage se fait par l'intermédiaire d'un code d'étalement dont le choix est déterminé par les bits

de parité du vecteur message transmis à travers les antennes multiples. Le schéma proposé permet l'accès multiple par l'intermédiaire d'une conception conjointe des codes à travers les antennes multiples, les sous-porteuses OFDM, et les utilisateurs. Une diversité combinée en espace, temps et fréquence permettent aux utilisateurs de partager les sous-porteuses à un niveau acceptable d'interférence multi utilisateurs. Ainsi, une meilleure efficacité spectrale est atteinte tout en améliorant la performance en taux d'erreur sur les bits (BER) en fonction du rapport signal sur interférence.

Le deuxième objectif consiste à développer un environnement de prototypage temps-réel. Dans la plateforme proposée, la communication Matlab-FPGA est gérée directement par l'entremise du protocole UART (*Universal Asynchronous Receive and Transmit*). Dans cette thèse, les fonctions de base de l'UART sont implémentées à l'aide du VHDL puis intégrées au système afin d'obtenir une transmission de données compacte, stable et fiable ; et obtenir ainsi une plateforme de conception matérielle complète pour un système MIMO-OFDM.

Le troisième objectif est de développer une architecture FPGA à virgule flottante pour le système émetteur-récepteur MIMO-OFDM proposé. L'architecture proposée est divisée en sous-modules où des optimisations adéquates sont suggérées afin d'atteindre une optimisation globale de l'architecture.

B.2.3   Organisation de la thèse

Le premier chapitre traite de la motivation et des objectifs de la thèse. Le deuxième chapitre fournit une vue d'ensemble des systèmes de transmission OFDM y compris leurs modèles mathématiques, leurs avantages et inconvénients. Ensuite, la combinaison MIMO-OFDM est décrite et le modèle qui en résulte est introduit, suivi par une vue exhaustive des techniques de

détection MIMO, leurs performances en terme de BER ainsi que leurs analyses de complexité. Enfin, les schémas de transmission MIMO-OFDM sont abordés. Le chapitre 3 présente le nouveau plan MIMO-OFDM basé sur l'étalement à bit de parité sélectionné et à bloc de permutation. Un modèle mathématique de la technique proposée est fourni et des simulations sont présentées pour de différentes antennes d'émission et de réception, de différentes modulations, de diverses longueurs de code, et de différentes techniques d'égaliseur. Dans le chapitre quatre, une méthodologie de conception FPGA pour les systèmes MIMO-OFDM est présentée permettant la conversion des algorithmes proposés pour qu'ils soient exploitables sur la plateforme de prototypage. De plus, des implémentations détaillées pour un environnement de prototypage temps réel basé sur l'UART sont également présentées. Parallèlement, les désavantages potentiels de chaque module sont fournis. Les résultats de la synthèse, incluant l'usage des ressources matérielles, la latence, et la consommation sont présentés puis analysés. Finalement, les résultats de la vérification fonctionnelle des principaux modules du système sont introduits. Le chapitre 5 propose et décrit le processus d'optimisation. Des architectures efficaces et optimisées sont proposées et conçues pour les modules fonctionnels clés du système. Ces designs efficaces comportent une architecture pipeline pour les modules IFFT/FFT, une architecture à faible complexité pour le module de désétalement, et une architecture à faible complexité pour l'inversion des matrices par élimination GJ. Finalement, le design est converti, en sa totalité, en une représentation à virgule fixe ; les compromis performance - réduction d'espace sont examinés. Une conclusion générale de la thèse est faite dans le chapitre six qui récapitule les principaux résultats, mais aussi traite quelques questions ouvertes qui feraient l'objet de futures recherches.

B.3   MIMO-OFDM avec étalement à bit de parité sélectionné et à permutation

L'idée d'appliquer une transformée linéaire pour étaler l'énergie des symboles transmis à travers les sous-porteuses de l'OFDM, afin de profiter des avantages de la diversité, a été réalisée en divisant l'ensemble des sous-porteuses en plusieurs blocs à travers lesquels les symboles de données sont étalés. L'exploitation de la diversité multi chemins est possible en représentant les symboles sur les sous-canaux. Néanmoins, la complexité du système demeure un obstacle majeur compte tenu de la complexité du décodage qui en résulte. Par exemple, lorsque la taille du bloc M n'est pas très grande, l'estimation du maximum de vraisemblance (ML) peut être utilisée pour la détection. Lorsque M est grande, la complexité de détection ML augmente exponentiellement avec M. Afin de réduire la complexité, des méthodes sous-optimales spécifiques sont utilisées, telles que le décodage sphérique. De plus, la majorité des codes ST/SF existants sont destinés aux systèmes à utilisateur unique seulement. Pour les canaux à accès multiple, les codes ST/SF sont alloués séparément à chaque utilisateur, ce qui réduit le taux de transmission.

Dans cette thèse, un nouveau schéma de transmission basé sur un système MIMO-OFDM codé en STF combiné à des méthodes d'étalement à bit de parité sélectionné et à bloc de permutation est proposé et illustré dans la Figure B-1. Le symbole de données à transmettre par chaque antenne est étalé grâce à un code d'étalement dont le choix est déterminé par le bit de parité du vecteur du message transmis à travers les antennes multiples. Du coté récepteur, illustré dans la Figure B-2, la détection de données se fait grâce aux corrélateurs attribués aux différents codes utilisés par le transmetteur. Une fois le code d'étalement est identifié au premier stade, la possibilité d'erreur relative à la détermination du bon bloc d'informations devient très

réduite. Autrement dit, la probabilité d'erreur est largement dominée par les erreurs dues à la

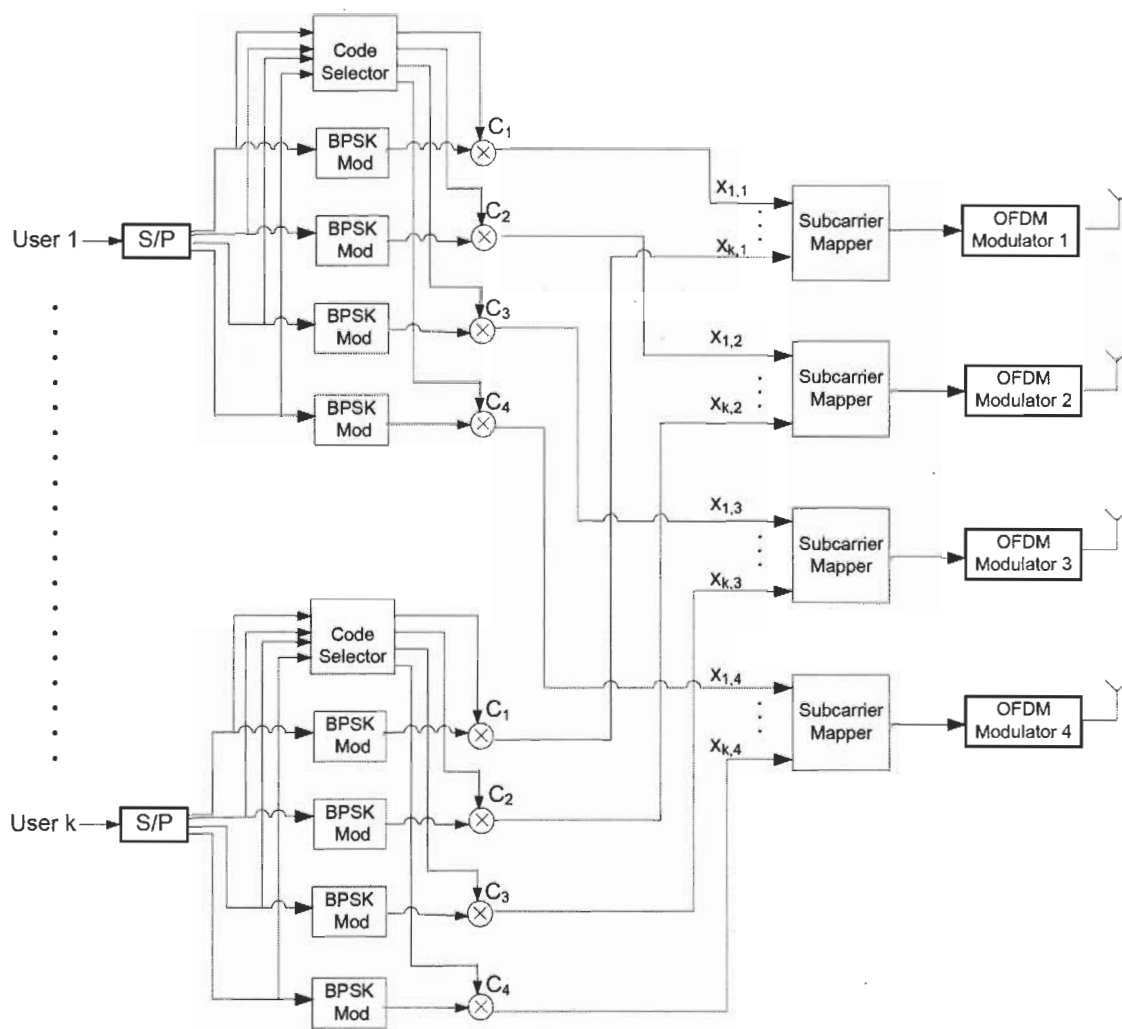détermination erronée d'une séquence d'étalement.

Figure B-1 4X4 Transmetteur MIMO-OFDM avec étalement à bit de parité sélectionné
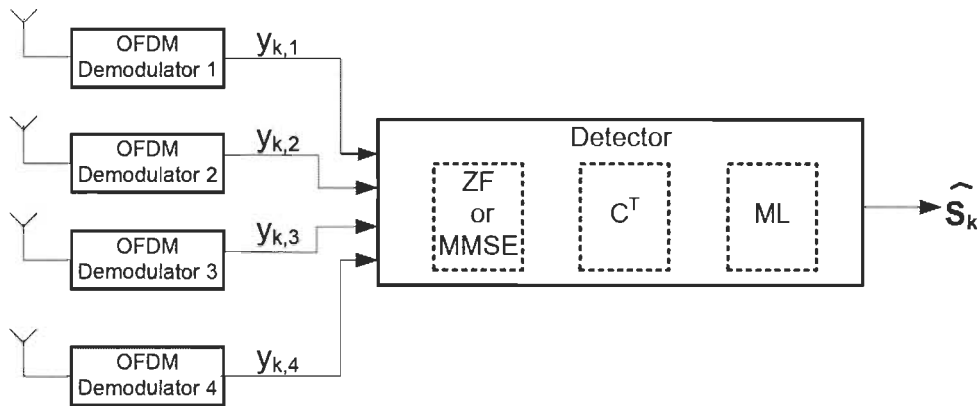
Figure B-2 Récepteur MIMO-OFDM pour étalement à permutation et à bit de parité sélectionné pour Nr = 4

Le système proposé permet des débits binaires flexibles et un multiplexage utilisateur efficace, requis pour les systèmes de communication sans fil de la prochaine génération. Dans le cas d'un système MIMO-OFDM, les utilisateurs sont séparés sur différentes bandes de fréquences (sous-canaux), et chacun d'eux est codé séparément via STBC ou SFBC, ce qui mène à une réduction du débit binaire lorsque le nombre d'utilisateurs augmente. Le nouveau schéma permet une diversité combinée de l'espace, du temps et de fréquence en permettant aux utilisateurs de partager les sous-porteuses à un niveau acceptable d'interférence multi utilisateurs. Ainsi, une meilleure efficacité spectrale est atteinte lors de l'amélioration de la performance du taux d'erreur (BER) en fonction du rapport signal sur interférence.

B.3.2   Résultats de simulation numérique

La Figure B-3 présente une comparaison de la performance BER d'un système conventionnel STBC MIMO-OFDM à celle du système MIMO-OFDM proposé (système d'étalement à permutation ou à bit de parité sélectionné) pour une configuration 2x2. Les courbes démontrent clairement l'avantage du schéma proposé. D'autres scénarios de simulation sont discutés par la thèse.
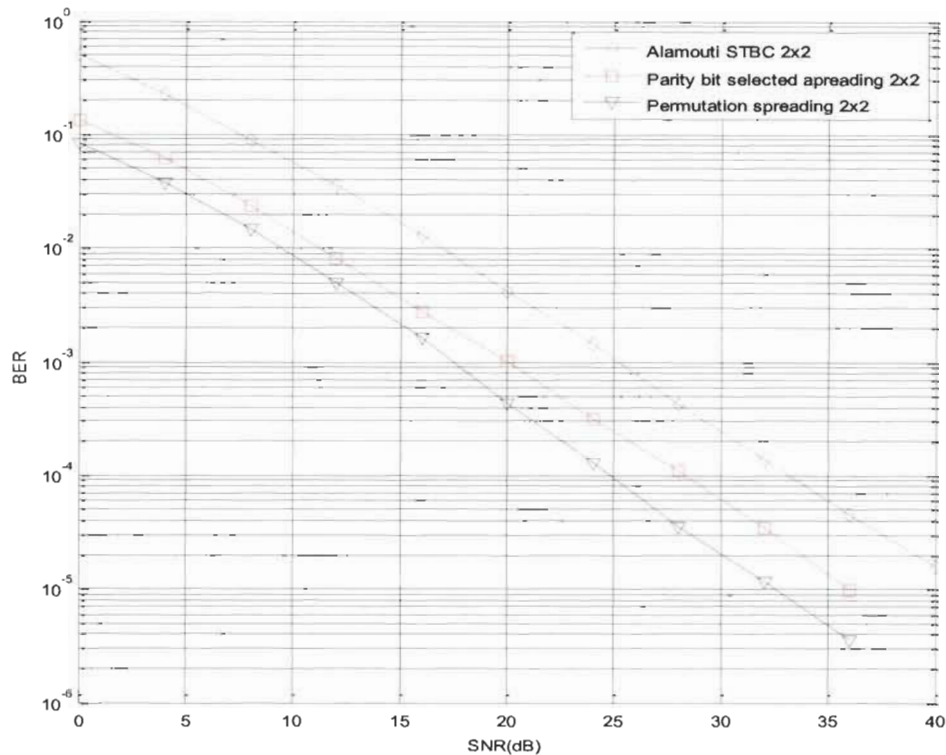
Figure B-3 La performance BER pour un OFDM MIMO 2X2 avec étalement à bit de parité sélectionné, à permutation, et STBC Almouti

B.4   Conception et Implémentation FPGA du système MIMO-OFDM proposé

Les détails de l'implémentation du système MIMO-OFDM proposé sont présentés pour une configuration 2x2. La méthodologie de conception du prototypage rapide pour une représentation à virgule flottante est introduite. Cette implémentation offre une haute précision, ce qui permet de vérifier le bon fonctionnement du design implémenté comparé au modèle Matlab. De plus, les dépendances de données peuvent être identifiées par l'intermédiaire du modèle initial RTL à virgule flottante, avant la phase d'optimisation. Par la suite, une plateforme de conception matérielle temps réel est proposée. Elle supporte la simulation HIL (Hardware-in-the-loop) pour les algorithmes MIMO-OFDM. Sur cette plateforme, le module

UART est conçu et intégré sur la même puce FPGA pour mettre en place une communication en série entre Matlab et la carte FPGA. Ensuite, l'architecture FPGA de l'émetteur-récepteur MIMO-OFDM est introduite. Cette architecture est divisée en sous-modules et le design détaillé de chacun est proposé. Par après, les résultats de l'implémentation tels que l'utilisation des ressources matérielles et l'analyse temporelle sont présentés et discutés. Une fois l'implémentation FPGA de l'émetteur-récepteur MIMO-OFDM proposé est introduite, plusieurs options d'optimisation sont proposées pour réduire l'espace, la consommation et le temps d'exécution. On recommande également, dans ce contexte, une architecture pipeline dans laquelle un seul bloc IFFT/FFT est partagé par toutes les antennes émettrices/réceptrices. L'unité de désétalement est un autre module gourment en terme de ressources. Alors que l'unité de désétalement basée sur filtre attribué consomme les ressources en grande quantité, car elle inclut beaucoup d'opérations arithmétiques telles que la multiplication, la division et la racine carrée, l'algorithme de désétalement proposé réduit significativement la quantité de ressources matérielles requises. Ensuite, une architecture optimisée pour l'inversion de matrices à valeurs complexes utilisant l'élimination Gauss-Jordan (GJ) est proposée. Seulement les opérations arithmétiques critiques sont calculées pour obtenir les valeurs voulues sans exécuter toutes les opérations arithmétiques de l'algorithme d'élimination GJ. Le résultat est une réduction du temps d'exécution et de la consommation de ressources matérielles. Enfin, l'architecture FPGA à virgule fixe, où le maximum de perte de performance est défini grâce à la quantification, est développée, puis les compromis performance BER - réduction d'espace sont analysés.

B.4.2 Résultats d'implémentation

Les techniques d'optimisation proposées ont été appliquées pour une implémentation complète du système émetteur-récepteur utilisant la représentation à virgule flottante. Le

Tableau B-1 illustre une comparaison entre les ressources matérielles requises pour un design optimisé et celles pour un design non optimisé. Les résultats indiqués montrent clairement l'influence des optimisations proposées qui se reflètent par une réduction importante de la consommation et la possibilité de transférer le design vers une carte FPGA basée sur le Virtex 5.

Tableau B-1 Ressources consommées pour un émetteur-récepteur à virgule flottante non optimisé vs. optimisé

Ressources	Non optimisé		Optimisé	
	Nombre consommé	Pourcentage de ressources Virtex 5	Nombre consommé	Pourcentage de ressources Virtex 5
Registres Slice	87540	303%	26712	92%
LUTs Slice	95300	330%	26800	92%
Bloc Ram/FIFO	24	40%	24	40%
DSP 48 E	232	483%	48	100%

Un émetteur-récepteur temps réel MIMO-OFDM est implémenté utilisant les optimisations proposées et une représentation à virgule fixe 12-bits. Le Tableau B-2 présente une comparaison entre les ressources matérielles requises pour un modèle à virgule flottante et un autre à virgule fixe d'un design émetteur-récepteur complet. La réduction de ressources requises due à la quantification peut être clairement identifiée. Des simulations post placement et routage sont conduites pour s'assurer que le design optimisé respecte les contraintes. On mène, par la suite, une vérification sur carte assistée par une co-simulation Matlab et la plateforme du design

proposé. Au stade final, les données sont générées par Matlab et envoyées à la carte FPGA via le module d'interface UART proposé pour le traitement, puis reçues par Matlab pour vérification.

Tableau B-2 Ressources consommées pour un émetteur-récepteur optimisé à virgule flottante vs. virgule fixe

Ressources	Virgule Flottante		Virgule Fixe 12-bit	
	Nombre consommé	Pourcentage de ressources Virtex 5	Nombre consommé	Pourcentage de ressources Virtex 5
Registres Slice	26712	92%	18364	63%
LUTs Slice	26800	92%	25070	83%
Bloc Ram/FIFO	24	40%	20	33%
DSP 48 E	48	100%	28	58%

B.5 Conclusion

Dans cette thèse, un nouveau schéma de codage de transmission est proposé pour un système MIMO-OFDM pour supporter le multi accès et réduire significativement la complexité de l'algorithme de détection. Ce plan attribue à chaque utilisateur un ensemble de codes d'étalement orthogonaux, et les données de chacun sont diffusées à travers de multiples antennes, symboles OFDM, et sous-porteuses. La sélection du code d'étalement se fait soit par les techniques de parités soit à l'aide de celles de permutation. Ce schéma réduit la complexité de détection. De plus, l'accroissement de la diversité obtenue en étalant le signal transmis sur trois domaines a engendré une amélioration importante de la performance en présence de

canaux à évanouissement sélectif de fréquence. Les résultats de simulation ont montré que le nouveau schéma donne une meilleure performance comparée à celle d'un MIMO-OFDM conventionnel avec STBC due à la capacité du premier à maintenir un maximum de diversité à la réception.

La deuxième contribution de cette thèse consiste à l'introduction d'une méthodologie de conception systématique et une plateforme de prototypage temps réel. Dans la méthodologie proposée, le modèle Matlab est directement converti en programme VHDL et transféré à la puce FPGA, ce qui permet d'épargner un temps important comme toute conversion intermédiaire est éliminée. L'UART est utilisé pour gérer la communication entre Matlab et la carte FPGA d'une façon efficace afin d'exécuter une co-simulation entre le design transféré et Matlab. Les fonctions UART sont implémentées en utilisant le VHDL et intégrées à la puce FPGA du MIMO-OFDM. La troisième contribution de cette thèse s'illustre dans la proposition de plusieurs techniques d'optimisation afin de réduire la complexité du système et épargner les ressources matérielles.